University of Reading

A systematic analysis of visual and algorithmic letter fitting

Nathan Willis
September 2023

Thesis submitted for the degree of Doctor of Philosophy
Department of Typography & Graphic Communication

Declaration: I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Nathan Willis

This research project was funded by the University of Reading.

Abstract

This thesis examines the process of ‘fitting’, or determining the preferred spacing of the letters and other forms that comprise a typeface. Successful fitting is important to the readability and aesthetics of type, and is traditionally performed as a manual process. The objective is to determine to what extent this manual process can be modelled and expressed in an algorithm, to increase the theoretical understanding of fitting and suggest practical strategies.

The research incorporates methodologies from several disciplines, including historical studies, algorithmic analysis of procedures and strategies employed in fitting, development of computational software, and empirical testing.

The manual process of fitting was analysed from historical sources and contemporary practice. From the study, an axiomatic model was developed expressing the first principles of fitting Latin text for continuous reading and interdependencies between those principles. Prior work was evaluated in relation to the model and a new method was developed to fit typeforms with open counters, a class of forms historically reported to be difficult to fit.

A composite algorithm was developed that traverses the typeforms in a typeface, fitting each form with the simplest technique applicable, until the fitted set is complete. The composite algorithm was used to fit a set of typefaces, which were tested in an online reader study. Readers were shown a series of text samples, utilising original and refitted fonts, and asked to mark letter sequences they felt exhibited poor spacing.

The composite algorithm achieved lower rates of reported poor spacing for multiple letterform pairings than the alternative conditions. This supports a view that the axiomatic model capably represents the fundamental fitting process, that the novel method for fitting open counters can improve on prior techniques, and that the composite approach to algorithmic fitting, combining multiple discrete principles, holds benefits for the fitting of typefaces.

Contents

Acknowledgements

I would like to thank Prof Fiona Ross, Dr Mary Dyson, and Dr Matthew Lickiss for their unparalleled supervision and advice.

I would also like to thank the University of Reading, Graduate School, and the Department of Typography & Graphic Communication for their generous award of the studentship which allowed me to undertake this research.

Along the way, I was the recipient of copious assistance, insight, and counsel, both particular and general, from a number of friends and colleagues as well as a few virtual strangers who still offered their time. Among those, I owe my thanks to David Březina, Pat David, Julien Deswaef, Behdad Esfahbod, Frank Grießhammer, Borna Izadpanah, Eric Kindel, Indra Kupferschmid, Kent Lew, Cosimo Lupo, Ira Lusik, Francisca Monteiro, Roel Nieskens, Pria Ravichandran, Irmi Wachendorff, Tiffany Wardle, and Josh Willis, as well as others I'm sure I have forgotten.

I am particularly indebted to Graham Beck, Lida Lopes Cardozo Kindersley, and the rest of the Cardozo-Kindersley Workshop in Cambridge, for their hospitality and support, not merely with archival material and information, but with advancing my understanding the workings of the LOGOS project and the fuller history of its background and its development.

This research would also not have been possible without my fellow students, starting with Xunchang Cheng, Mohamad Dakak, Arjan Khara, Soo Kim, Paul Ransom, Ferdinand Ulrich, and Cătălina Zlotea, plus many others, who personified community even in the most unusual of circumstances.

Finally, this project is dedicated to my family, who never wavered in their encouragement.

1. Introduction

The essential difference between type and other forms of making letters, such as handwriting, painting, or carving in wood or stone, has been explained by noting that type letters are pre-made or pre-fabricated (Smeijers 1996, p.21; Noordzij 2009, p. 9), and composed into a text some time later, generally by someone other than the designer of the letters. Hidden within this definition is the fact that the act of composing the text creates shapes as well: the negative forms that are defined by the boundaries of the letters' positive forms as they are set together. This is, of course, neither an undesirable accident nor a surprise to the designer of the type. As such, a vital part of designing the letterforms is anticipating and planning for these negative forms: what type designers call *fitting*.

More precisely, fitting is the process of determining the fixed separation of the letterforms so that they will appear, to the reader's eye, balanced and harmonious when text is typeset. In earlier eras, when type was manufactured as physical objects, the fitting of each letterform was similarly physical. *(See figure 1.1)* When letterforms were cast in metal, the dimensions of the casting inherited the dimensions of the mould (into which the metal was poured), which in turn had inherited its dimensions from the matrix (around which the mould was aligned), which had been established several stages earlier in the manufacturing process, through the judicious effort and manual labour of the punchcutter or justifier (Carter 2002, p. 6–8).

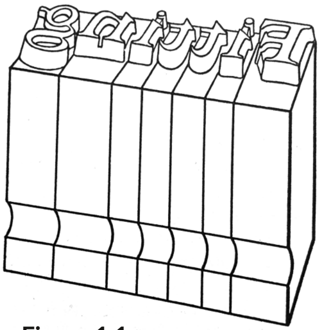


Figure 1.1
Blumenthal's illustration of fitting as seen in metal types. All of the forms are set on equal-height bodies for vertical alignment, but the sidebearings (relative widths) are established for each form (Blumenthal 1935, p. 71).

Sorry; this image could not be cleared for redistribution.

Figure 1.2
Karow's illustration of fitting as seen in scalable digital type. The sidebearings (A and B) are numerical values chosen for each form (Karow 1994, p. 179).

In the present era of scalable digital typefaces, the dimensions of each letterform are simply numerical values stored in a digital file, and the designer of the typeface can specify and update them directly. For the Latin script, composed in horizontal lines, the vertical bounds of the letterforms are, for the most part, uniform throughout the typeface, and the values that dictate the fixed separation for each letterform are the distances to the left and right of the positive form, called *sidebearings. (See figure 1.2)* But the judicious effort required to establish those values is still crucial, because fitting digital typefaces must also address the eventual goal of composing the letters into words, lines, paragraphs, and pages that readers perceive as even and harmonious. As Walter Tracy wrote in 1986,

> Letters do not live in isolation, They are the elements of meaning, the components of visible language, and their spatial relationship with each other is crucial, not only for the rapid recognition of words by the reader but for the regularity of texture that is essential if the reader's comprehension is to be maintained for a long period. (Tracy 2003, p. 77–78)

If the fitting of the typeface is poor, the readability of the eventual text suffers, and so in turn does the experience of the reader (Unger 2007, p.

149). This is doubtlessly why Tracy called the process of fitting 'fundamental to the success of a type design' (Tracy 2003, p. 71). Nor is his choice of phrasing coincidental: fitting has long been recognized as an integral part of the design process by those in the type-design field and those in the broader community of typography. Charles Bigelow and Jonathan Seybold, like others in the type and typography business, occasionally ascribed more importance to fitting than letterform design: 'It is often claimed, with some justification, that a mediocre type design well spaced will look superior to a good design, badly spaced'. (Bigelow and Seybold 1981, p. 14).

That sentiment is perhaps intentionally hyperbolic, meant to draw the readers' attentions to the importance of fitting. It is also a statement grounded in a practical reality, however, and points to another benefit of good fitting. Much of Bigelow and Seybold's discussion of fitting highlighted its practical necessity for commercial printing: typefaces that ship to customers in well-fitted form reduce the need for customers to make adjustments when typesetting documents (Bigelow and Seybold 1981, p. 13). Publishers thus place additional value on high-quality fitting for the convenience it provides to them, apart from its benefits for the reader (Nicholson 1990).

This leads to a third important facet of fitting: as a marker of the type designer's skill or the typeface's quality (Karow et al. 1994, p. 225). Several writers have noted that poor fitting during the design process can lead to lower-quality letterforms or make design problems more difficult to identify and solve (Smeijers 1996, p. 26; Henestrosa et al. 2017, p. 81). The notion of intrinsic quality certainly overlaps with commercial viability, but quality is, at times and to particular people, an aspect of the typeface itself. Fernando Mello expressed that viewpoint succinctly, writing that 'A typeface just can't be good if its spacing is bad' (Mello 2018, p. 33).

1.1 The task of fitting in typeface design

Although there is broad consensus among type designers about the importance and integrality of fitting to the design of typefaces, fitting as it is practised is often a task separate from designing letterforms, relegated to a distinct stage in the process that may be regarded as secondary (Henestrosa et al. 2017, p. 80). The type designer designs some letterforms (perhaps many), then pauses for a time and addresses fitting. Then the process is repeated. Some would say that this subdivision of the tasks is an outcome imposed by the specifics of printing technology. Gerrit Noordzij wrote that 'A letter is two shapes of different brightness (e.g. black and white). The writer knows of the complicated relationship between both shapes', but that it was the 'simplified view of an outsider' that invented typography and thus 'reduced the background shapes to rectangles whatever the shape of the strokes might be' (Noordzij 2000, p.3–4).

Conversely, perhaps the differentiation between designing positive forms and fitting them with respect to negative space is inescapable, purely because it is the positive forms of the letters that are drawn, that are inked by the printer and impressed onto the page, provided as keys on a keyboard, and ultimately perceived against the background — whereas the spaces between the forms are only permitted to emerge after the fact, taking form as the words, lines, and pages begin to appear. *(See figure 1.3)* This is evident in the literature of type; writers describe the letterforms designed by Granjon or Baskerville. Rarely, if ever, does discussion centre on the fitting of Benton or Koch.

Whatever the reason for this bifurcation between fitting and drawing, the task of fitting is frequently described as an undertaking that is time-consuming, if not outright challenging to master (Blumenthal p. 73; Tracy 2003, p. 77; Karow et al. 1993 B, p. 248; Campe and Rauche 2022, p. 93). But

Sorry; this image could not be cleared for redistribution.

Sorry; this image could not be cleared for redistribution.

Figure 1.3
The importance of fitting illustrated at three scales.

Top: Fred Smeijers on fitting at the level of the word (Smeijers 1996, p. 26). Used by permission.

Middle: Richard Rubinstein on fitting at the level of the sentence (Rubinstein 1988, p. 116).

Bottom: Elwyn and Michael Blacker on fitting at the level of the paragraph (Blacker and Blacker 1993, p. 71).

type designers and typographers have also acknowledged that fitting is a skill which can be acquired and honed to a high degree of expertise. Moreover, since the earliest days of type-making, the general view has held that desirable fitting appears to behave according to predictable rules, (Fournier *trans.* Carter 1930, p. 160–161; Jamra 2004).

1.1.1 Evolving perspectives on fitting and automation

In the early to mid 20th century, some type designers spoke of these predictable rules in terms of equations or fixed formulas (Blumenthal 1935, p. 72; Dreyfus 2000, p. 31) and pursued discovering and defining them. William Addison Dwiggins wrote to Rudolph Ruzicka in 1937 that 'there must be some general formula' — but advised Ruzicka that Chauncy H. Griffith at Mergenthaler Linotype disagreed (Dwiggins 1940 A, p. 6). By mid 1940, as Dwiggins's correspondence with Griffith on the Falcon typeface shows, Dwiggins was quite engaged with devising a formulaic system for fitting, even creating his own notation for inter-letter spaces (nn or nn), distances measured from vertical stem to vertical stem (n-n), and left and right sidebearings (n, n) to document and relay his findings to Griffith (Dwiggins 1940 B, p. 9).

As typesetting technology advanced, so did type designers' relationship to the task of fitting. Gerard Unger noted that the rules governing fitting in metal type were crude when compared to those of photocomposition and digital type (Unger 2018, p. 123). By the time those technologies had supplanted metal as the norm, type designers and technologists had shifted away from looking for a purely mathematical formula and began to actively address the possibility of automatically applying fitting rules via computer software.

In 1986, Walter Tracy published his own standardized fitting method in the form of a heuristic, based on his long experience at Linotype, supplying readers with a list of values that could be systematically applied to the various letterforms (at least, in an upright roman typeface) by their shape. *(See figure 1.4)* Notably, although his method provided pre-defined fitting values for most of the Latin letterforms, Tracy supported them with rationales: the round letter profiles receive the same amount of space

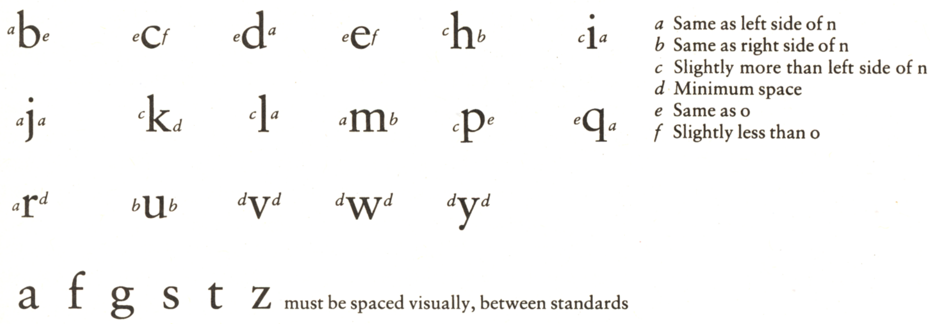


Figure 1.4
Walter Tracy's heuristic fitting system, for lowercase upright Latin forms. It expresses sidebearings for most letterforms as a set of six repeatable values (Tracy 2003, p. 75). On the preceding page of the book, Tracy provides a similar heuristic for the capitals (not shown).

because they are the same shape; the diagonal letter profiles receive the minimum amount of space because they trap a triangular region of white space that cannot be removed, and so forth (Tracy 2003, p. 79). He acknowledged that the heuristics sounded like a job for a computer, but added that, in his estimation, a computer-based system could never be wholly reliable (ibid., p. 77).

In contrast, the renowned stonecarver, calligrapher, and letter designer David Kindersley fully embraced the potential of computer-based fitting methods, writing 'there are too many factors to be held in the head at the same time. A trained eye can space a few words, but it is too much for anyone to arrive at a proper space for each one of 100 characters or more so that there is anything like perfect interchangeability. [...] No, today our only chance is basic research with a computer to hand'. (Kindersley 1973, p. 13) To that end, Kindersley pursued a decades-long project to develop a software letter-fitting system in collaboration with Neil Wiseman and others at the Cambridge Computer Laboratory. *(See figure 1.5)* The result of that project, LOGOS, was marketed as a production tool to printers and foundries in the late 1980s, but did not find commercial success.

Peter Karow, a scientist by training and pioneering type technologist by trade, also dedicated years to researching methods for fitting type via software at URW (Karow 1998). In 1993, URW promoted a new digital typesetting engine called *hz-program* which included a module named *kf* ('kerning on the fly') for automatically altering the fitting of any font in a document for optimal evenness (Seybold 1991; Karow 1992; Seybold and Karsh 1993). *(See figure 1.6)* Although *hz-program* was much vaunted preceding its launch, claiming to rival Gutenberg in typographic quality (Zapf 1993; URW Software & Type 1993), it also fell short of finding success in the marketplace. The company reorganized in 1995, with Karow

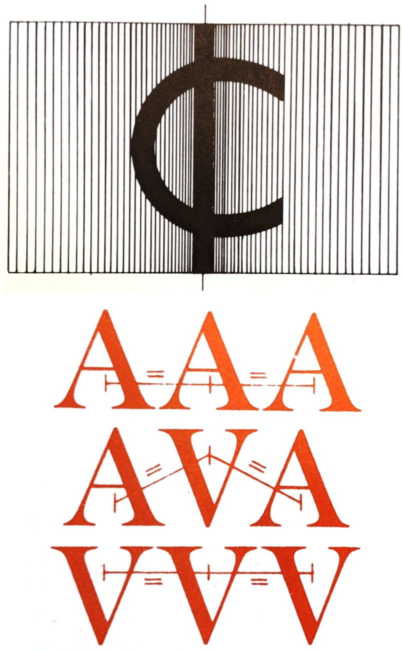


Figure 1.5

Top: The LOGOS fitting system found a centrepoint for every letterform (Kindersley 1987, p. 18).

Bottom: The LOGOS system fitted letterforms by adjusting the distances between centrepoints (Sassoon 1987, p. 109).

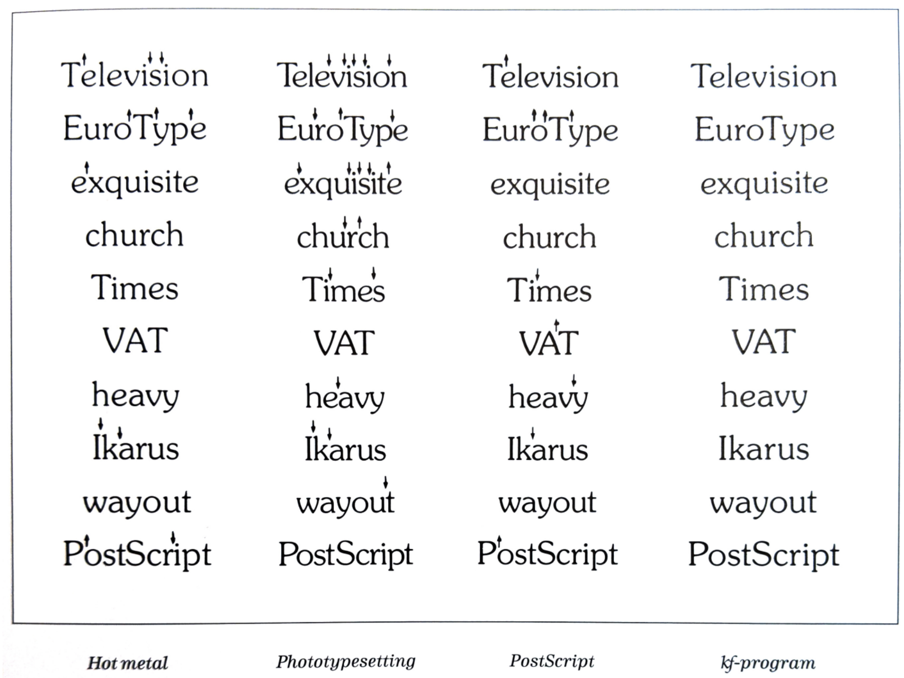


Figure 1.6

Detail from the promotional booklet for *hz-program*, showing the results of fitting letters with *kf* versus other typesetting products. Each arrow indicates fitting that the booklet claims is incorrect or problematic (URW 1993, p. 19).

departing (Seybold 1995); subsequently the composition quality advertised in *hz-program* took on a near-legendary mythos (Ecker et al. 1998; Eng 2009).

In the decades that followed, the question of fitting letterforms automatically or with the assistance of computer software has continued to engage the attention of type designers, font engineers, and software developers alike. This research does not seek to identify a foolproof method by which all the forms in a typeface can be correctly fitted without human intervention (however 'correctly' may be defined), but to look at the prior investigations into letter-fitting automation collectively, and at how the various attempts to fit letters automatically or semi-automatically relate to fitting as the task has been performed manually. In doing so, this research provides observations and investigates approaches by which future work on fitting letters by means of automated techniques might be brought better into harmony with the task of fitting as it has traditionally been conducted.

A great deal about the manual process of fitting is well-known within the type-design field. The basic principles of evenness, balance, and harmony are common citations within the historical discussions and instructional materials concerning fitting, as are details of how those principles interact with letterforms and how type designers can make effective assessments. Nevertheless, the attempts to devise systems, formulas, and software-automation tools for fitting typefaces over the years found only limited success with type designers. Tracy's heuristic method provided an incomplete system; he enumerated several letterforms that must be fitted visually: a, f, g, s, t, z, and S. The list notably overlaps with Dwiggins's list of 'wolf intervals' for which he could find no formulaic solution: a, c, e, f, g, k, r, t, and s (Dwiggins 1940 B, p. 6). The LOGOS and *hz-program* products, as mentioned above, were unable to establish a foothold with type designers and manufacturers, despite their respective pedigrees and substantial investments of time and effort.

This research does not attempt to find fault with the above examples or other prior work, but instead to contend that they are worth revisiting in the present context and that the subject is worth investigating as a shared endeavour within the type-design community, best viewed as part of the continuum of practise that includes manual fitting. The technology available to the designers and producers of type has advanced considerably since many of the prior letter-fitting investigations began. Thus, prior work often sought to apply a homogenous method for fitting all letterforms, but may have done so because the speed and cost of computers and typesetters at the time necessitated it (Rubinstein, p. 121). Similarly, there may be valuable connections to be uncovered between independently developed methods that remained hidden when those methods were kept private to preserve trade secrets. A holistic approach to the analysis can address such issues.

1.1.2 Defining success in letter fitting

In order for this research project to talk in detail about improving fitting — or even to compare whether one fitting for a particular typeface is better or worse than another fitting — it is necessary to establish a useful definition for what language like 'correct' fitting or 'good' fitting means. This is not a trivial matter, because fitting, like other facets of typeface design, inevitably involves aesthetic judgments, popular trends, and the personal tastes of type designers and type consumers (Karow et al. 1993, p. 249, 259; Spiekermann 1987, p. 29).

The aesthetic component of fitting is most clearly visible in display type. Where readability is not the paramount concern, type designers can be seen to be more experimental with the treatment of the shapes formed by inter-letter space and by open counters, illustrating that different fitting details can affect the reader and the reception of the typeface in ways wholly unrelated to readability. *(See figures 1.7 and 1.8, over page)* Even within text typefaces, however, the personal style or idiosyncrasies of the type designer can be seen. In his correspondence with Griffith about developing a formula for the fitting of Falcon, Dwiggins called out the fitting styles of Edward Maunde Thompson, William Morris, and Emery Walker as exemplary of the book-style fitting effect he sought for Falcon, but Bruce Rogers, Frederic Goudy, and Daniel Berkeley Updike as consistently missing the mark (Dwiggins 1942).

The personal perspectives of type designers aside, what qualifies as acceptable or exceptionally good fitting is also subject to shifts over time (Unger 2018, p. 115). To an extent, the shifting understanding of fitting is imposed by type production and printing. Bodo Kämmle noted that every type manufacturing technology imposes a resolution or unit system, below which the relationship between forms and sidebearings is lost because the technology or data format cannot express finer distinctions (Kämmle in Karow et al. 1993, p. 185–186). But the long-term trends in fitting are also supported by the changing opinions and tastes of type designers and, ultimately, of readers (Unger 2007, p. 150–151).

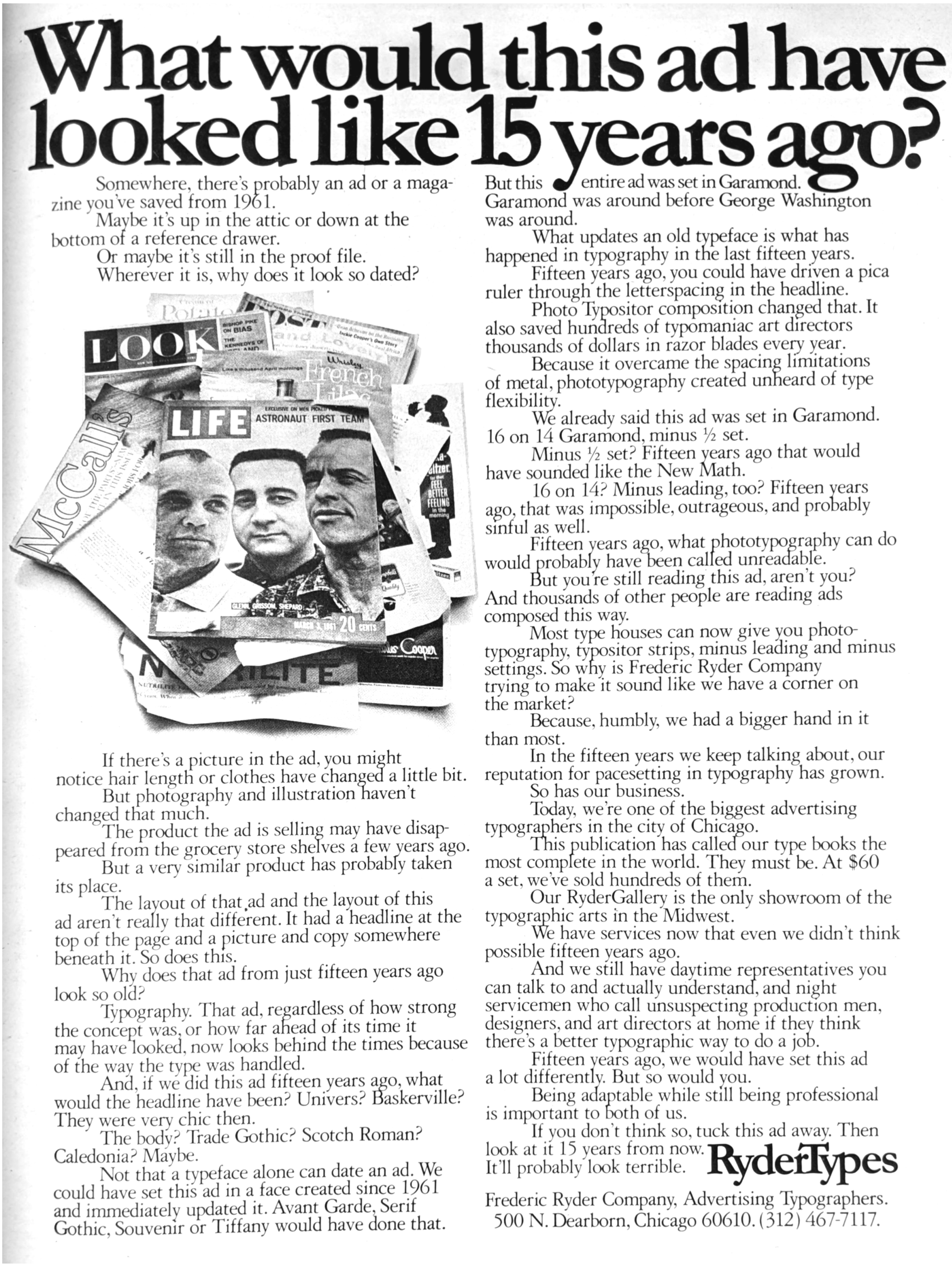
The type industry is cognizant of the fluctuating nature of taste and preference over time, of course. *(See figure 1.9, over page)* In this project, that factor can be accounted for when assessing fitting by acknowledging that the fitting preferred today may have been viewed differently in the past and may not be as well received in the future. As to the influence of personal style and aesthetic effects, type designers and typographers may never be in full agreement about the fitting of a particular typeface. There is general agreement, however, that fitting which is regarded as meeting the practical needs of legibility and readability with readers has succeeded at its fundamental goal (Carter 1984, p. 3). Gerrit Noordzij observed that illegible typography can still be beautiful, but that legibility is the more important quality in the hierarchy (Noordzij 2000, p. 126). Given a choice

Figure 1.7
Lucas Descroix’s monospace typeface Nostra Sett plays with the negative spaces between the letterforms, demonstrating creativity in fitting. Note, for examples, that the black foreground shapes of the letterforms are expanded to the point where they almost become the background, drawing attention to the inter-letter spaces left between pairs of letterforms like sa (line two), ti (line five), and gi (line four). (Descroix 2018, p. 4). Used by permission.

Figure 1.8
David Jonathan Ross’s typeface Fit allows the negative spaces between letterforms to fully dictate the positive shapes of the letters. All of the spaces are of an equal width; the letterforms are adapted to maintain this space requirement. Notably, Fit is designed to this rule in multiple weights and widths; the effect is preserved in thin and narrow versions (Ross 2017, p. 2). Used by permission.

Figure 1.9

A full-page advertisement from the July 1976 issue of the journal *U&lc*. The advertising copy discusses the then-contemporary trend of phototypesetting with extremely close fitting, and calls attention both to the fact that close fitting is a recent phenomenon and that it likely will be out-of-favour in the future (Frederic Ryder Company 1976, p. 37). From the collections of the department of Typography & Graphic Communication, University of Reading. Photographed by the author.



between the two, there is, at least, a history of testing in legibility and readability, providing examples to consult and from which test designs tailored to fitting can be explored. Henestrosa notes that ‘a badly drawn but properly spaced typeface is more useful than a properly drawn but badly spaced typeface’ (Henestrosa et al. 2017, p. 80), echoing to a degree the sentiment of Bigelow and Seybold, but underscoring that utility, all other factors aside, is a fundamental requirement for type. Consequently, this research bases its evaluations of fitting solely on the notion of successfulness with readers, as distinguished from correctness or aesthetic beauty.

1.2 Research questions

The aims of this research can be summarized by the following research questions:

> *1. Can an algorithm be constructed that will generate letter fitting for a well-designed typeface which cannot be distinguished from letter fitting determined manually?*

> *2. To what degree can the manual fitting process employed by type designers be modelled?*

As seen here, this project intentionally reframes the discussion somewhat as an investigation into fitting *by algorithm* rather than into fitting automation. The reasons for eschewing the term automation are twofold. First, ‘automation’ connotes a lack of involvement by humans (be they type designers or typographers). Although there might be occasions when full automation in that sense is desired, there are many other occasions when it surely is not. Second, ‘automation’ is difficult to define. Font editors such as Glyphs already provide some user-interface affordances to avoid simple repetition (Glyphs GmbH, 2023). For example, users can enter =n into the right sidebearing field for m, and the program will automatically copy the right sidebearing of n to the right sidebearing of m and keep the two values synchronized if the right sidebearing of n is changed. That synchronization might reasonably be considered automation, but it does not speak to core principles.

This project refers instead to algorithms, a term which encapsulates the meaning of a known and well-defined procedure. Algorithms might be executed rapidly or repeatedly by a computer, but can be followed by a person as well. Donald Knuth, who pioneered the analysis of algorithms in computer science, defines an algorithm as a procedure that has five properties: finiteness, definiteness, inputs, outputs, and effectiveness (Knuth 1968, p. 4–6). Finiteness, as applied to this project, simply means that there is some definition of when the procedure is complete.

Definiteness refers to the steps and conditions being well-defined, not open to interpretation in the moment. Inputs and outputs are, straightforwardly, what known elements are put into the algorithm before it begins and what elements are produced by it. Effectiveness in Knuth's terminology refers to an algorithm being simple enough in its structure that it can be written and understood and, perhaps, evaluated on the basis of speed or complexity.

In the context of letter fitting, the inputs are clearly the set of letterforms to be fitted and the outputs are the sidebearings for those letterforms. Without too much additional effort, some additional inputs could be defined, including the typographic context, and the set of outputs could be modified to include kerning tables or other techniques for representing the fitting to be stored in a font file or to be recorded in some other context. Finiteness is similarly direct: a fitting algorithm has reached the end when it has output all of the sidebearings asked for. The type designer may wish to re-execute the algorithm with different inputs, but there should be no ambiguity as to whether the sidebearings that were asked for have been provided.

Perhaps the most important aspect of defining a letter-fitting algorithm consistent with Knuth's conditions is the notion of definiteness. Namely, the manual process of fitting letterforms by eye (as described in the type design literature) frequently makes reference to the designer's intuition or optical judgment — for example, 'adjust the space between the two letters until it looks balanced'. In a practical sense, attempts to formulate letter fitting into an algorithm chiefly involve transforming these judgments into more definite constructions. Or, to look at it another way, part of the challenge is to identify the fundamental principles of the judgments the designer makes, and convert each 'judgment by eye' from an intuitive input supplied by the type designer into either an input that can be computed from the letterforms or a specific action that is expressed in as definite a step as is possible.

1.3 Scope and essential terminology

For pragmatic reasons, this research is limited to the Latin script, and focuses on typefaces intended for typesetting text for continuous reading. However, effort has been made to be alert to both where and how these limitations impact the results. This work takes a strong position that fitting can only be understood in a script-specific context, and must be evaluated as such. Consequently, the construction of models for fitting and analysis of fitting algorithms must also be understood in a script-specific sense.

It should also be noted that discussions of fitting in this work frequently refer to the *typeface* being fitted — despite the fact that type designers, in practice, will often perform fitting for incomplete typefaces or even for extremely small sets of letterforms, any of which may be altered in further design iterations. To facilitate a clear discussion of the fitting process, this work adopts the convention that, at a given point in time when forms are being fitted, those forms are regarded as complete (or, at least, fixed) and they work together as a reasonable Latin text typeface. The intent is to isolate the fitting principles at play when the design itself is not broken. That is, asking what the sidebearings should be for a set of letterforms that do not work well together — or are illegible or confusable due to design problems — is not a meaningful question. As has been noted, further design iterations may happen as the type designer chooses. Fitting can only fit the forms as they exist in the moment.

Historical sources and public discourses about fitting can, at times, be fluid about the terminology employed. This work attempts to standardize on the term *fitting* to refer to the process of determining or adjusting the spaces around the forms in a typeface. This is a choice common among type designers, at least in written English, but it is not universal. Some sources refer to this same task as 'spacing' or 'letter spacing'. Where quoted historical sources are unclear in their use of terminology, effort has been made to explain the meaning from context.

Some historical sources also disagree as to whether the term fitting includes only the setting of sidebearings, or includes both the setting of sidebearings as well as determining kerns (i.e., *kerning*). In this work, the term fitting is understood to encompass both sidebearing determination and kerning, primarily because — as will be discussed in the next chapter — there is not a hard delineation between the processes involved in the two tasks. That is, the same fundamental principles are applied when deciding the correct space between two letterforms in a kerned pair that are applied when deciding on the correct space for sidebearings in an unkerned pair.

Where typeforms themselves are discussed in this work, the blue-background notation a is used to distinguish those forms from the surrounding text for the ease of reading. Where literal tokens from programming or markup are discussed, monospace formatting is used, to better distinguish confusable items such as the `kern` table in OpenType from the general word 'kern'.

This work also makes occasional use of several other terms common in type design, such as *counter*, *serif*, *x-height*, *weight*, and *optical size*, which are considerably less fluid in their definition. For the convenience of the reader, a brief glossary of those terms has been provided, found after the conclusion of chapter 7.

1.4 Methods

This project's inquiry into possible algorithms for letter fitting is multidisciplinary. It begins with an examination and analysis of the processes and techniques that type designers employ when performing letter fitting manually. That examination, described in chapter 2, consists of a historical study of the practice of letter fitting as it is recorded in the literature and research of type design, educational and instructional material for type design, the literature of fields closely related to type design, and the recorded history of prior work developing letter-fitting automation (or algorithmic fitting) tools and utilities.

The analysis of this studied material is used to define a conceptual model, also described in chapter 2, for how the fitting of Latin text typefaces is performed manually. This model consists of a finite set of rules or axioms, each of which expresses a simple principle that helps determine fitting for one or more letterforms. The set of axioms is explored as a whole, to evaluate their interconnections as they are applied to a set of Latin letterforms.

Chapter 3 describes a more detailed investigation into the axioms of the Latin text fitting model, with a particular emphasis on those axioms that have been historically under-studied in prior letter-fitting work. In brief, some of the under-studied Latin text fitting axioms were found to be favourable for a new implementation, but there are other axioms lacking the degree of formalism necessary to be implemented in a fitting algorithm. Chapter 4 explores the potential for designing a composite fitting algorithm that combines several components, each implementing different axioms from the model. The composite algorithm uses well-established techniques to address letterforms with simple profile shapes in conjunction with novel techniques to address letterforms with concave profile shapes.

Chapter 5 begins by looking at various approaches to evaluating the fitting of a typeface, and which approaches can be used to measure the responses of readers in a quantifiable manner in line with the research questions. For this project, the goal of such evaluations is to use an algorithm to generate fitting for a typeface and then evaluate that fitting in comparison to the original fitting or to the fitting generated by some other algorithm. The method selected involves testing unaltered and refitted fonts on the web with volunteer readers. The readers were shown a series of randomized sample documents and asked to mark pairs of characters on the samples that they felt exhibited poor fitting and, for each mark, to indicate if there was 'Not enough space' or 'Too much space' between the pair. A series of such tests were staged featuring typefaces with their original (manually determined) fitting, new fitting generated by the composite algorithm from chapter 4, and new fitting generated by a re-implementation of the *kf* algorithm from URW's *hz-program* suite, for comparison. The results of the test are presented in chapter 6 and are interpreted with respect to the fitting algorithms.

This multidisciplinary nature of this research design necessitates, to an extent, streamlining the breadth of the project. While the conceptual model of fitting Latin text is as comprehensive as possible, the practical investigations of fitting-axiom implementations, fitting-algorithm design, and reader testing are of a focused nature, while revealing clear opportunities for additional research.

This project has been motivated by the author's belief as a researcher that algorithmic fitting is prime for continued systematic study. As is the case with many in the field, the author has had some measure of prior academic or professional experience in a handful of disciplines that address this research subject, including typeface design, mathematics, and computer science. The overlapping of these perspectives where the task of letter fitting is concerned allows this research to demonstrate that there is value in the pursuit, in both practical and theoretical terms.

1.5 Potential impact of algorithmic fitting

The most immediate practical aim of this research is to advance the current understanding of what is possible for letter fitting via algorithmic means, by presenting techniques that could lead to more effective or more efficient tools for type designers to employ when designing typefaces. A resulting reduction in time required to perform fitting, especially on its repetitive tasks, and a better-rounded understanding of letter fitting, would no doubt be welcomed by some type designers.

But there are other potential practical benefits to consider. If the task of fitting remains a flat list of hundreds of individual decisions about sidebearings, then it likely cannot help but be considered daunting by type designers. If, instead, an algorithm can give shape to the task, re-framing it as an active task that is defined in terms of its own principles and parameters, then new questions and affordances may become possible, and experimenting with fitting directly becomes an accessible option. For example, if the stroke rhythm of a typeface is too rigid and its readability suffers, then improving the fitting by means of hundreds of adjustments may seem to be a tedious chore. If, however, the fitting is adjustable via an algorithm that offers a parameter for stroke rhythm, then improving the fitting could be considerably less irksome, and invite more exploration.

It was also noted earlier that fitting interacts with the design of letterforms, and that fitting issues can obscure or hide design problems. Here, too, by providing richer tools for engaging with the task of fitting directly, rather than as a large set of discrete, individual decisions, type designers may reap benefits beyond simple time savings.

In addition to the potential for advancing practical fitting work, this research seeks to promote the theoretical understanding of the task of fitting and of how fitting is perceived, to the lasting benefit of the field. For example, a solid understanding of the links between calligraphic tradition and contemporary digital type (as encouraged by Noordzij and others) not only empowers type designers to understand type history, but to push it in new directions. So, too, the theories modelled and explored in this research can serve as analytical tools for other researchers or students of type history. A fuller understanding of how the principles of fitting behave and interact permits new questions to be explored regarding how type is fitted today and of how fitting can and should change — not just in Latin, but in other scripts, and between scripts, as well.

Similarly, by encouraging the formalization of the ideas and processes that are employed in fitting, this project provides a foundation on which more rigorous discourse about fitting can be based. Fitting, like the design and construction of letterforms, will always be an important topic of discussion. But, where the terminology for discussing Latin type anatomy and style is well-established and explicit, thus facilitating critique and all manner of analysis, in fitting it has often remained abstract. More formal language for discussing the principles of fitting and the fitting of specific forms or typefaces can only lead to new insights into how space functions in type design, typesetting, and reading.

2. Deriving a model for fitting Latin text

This research project explores the potential for algorithmically fitting typefaces — a task which, both historically and in prevailing contemporary practice, is performed manually. Before an algorithm appropriate to the task can be reliably constructed, then, it is vital to understand the constituent processes fundamental to the overall manual task, the context in which those processes are administered, and how those processes are understood by their practitioners. To that end, this chapter details the analytical development of a model for letter fitting in Latin text, using those accepted manual practices as a starting point.

As discussed in chapter 1, § 1.5, what constitutes successful letter fitting is inherently specific to a writing system, and is often constrained further by the specifics of the typographic setting. This project focuses initially on the Latin writing system when used to set text for continuous reading, and the model derived in this chapter is similarly focused on that writing system and those typographic constraints. Electing to focus on Latin text is a choice made for practicality, but effort has been made to maintain a separation between the approach used for the research and the details of the Latin-text fitting model itself. In this way, the model can serve as a proving ground for the approach. Models for other writing systems or for a divergent set of typographic constraints could be derived by employing the same approach, even though the specific details of two such models may vary considerably in the end.

2.1 Manual fitting practices in Latin typeface design

The task of letter fitting is regarded by professional typeface designers as an integral component of designing a typeface (Tracy 2003, p. 71). Indeed, rather than describing letter fitting as a secondary or even as a complementary discipline, Fred Smeijers accounts an awareness of the shapes between letters as inseparable from that of the letters themselves. ‘The white shapes make the background, the black shapes make the foreground. The background makes the foreground, and the other way around. Change one, and you change the other too’ (Smeijers 1996, p. 24).

The literature survey of letter-fitting theory that will be described in § 2.2 examines as many of the historical sources as possible, but the discussion must begin with an overview of the fitting process as it is practised today. This permits deriving a model (and, ultimately, algorithms) formulated with relevance for contemporary technology and practitioners.

A full description of the typeface-design process from start to finish would lie outside the scope of this research. Books that specifically describe typeface design are scarce, at least compared to other disciplines in printing and graphic design (Leonidas in Unger 2018, p. 7). In-depth guidance on the task of fitting within the larger process of typeface design can be found in Walter Tracy's *Letters of credit* (Tracy 2003), Cristóbal Henestrosa et al'.s *How to create typefaces* (Henestrosa et al. 2017), and Chris Campe & Ulrike Rausch's *Making Fonts* (Campe and Rausch 2022). Additionally, a number of other works discuss aspects of typeface design including manual letter fitting — sometimes to a considerable depth — even if the overall work is not structured as a complete guide to the typeface-design process. This includes Fred Smeijers's *Counterpunch* (Smeijers 1996), Karen Cheng's *Designing type* (Cheng 2005), Jost Hochuli's *Detail in typography* (Hochuli 2015), Gerrit Noordzij's *The stroke of the pen* (Noordzij 1982) and *The stroke* (Noordzij 2009), Frank E Blokland's *On the origin of patterning on movable Latin type* (Blokland 2016), and Gerard Unger's *While you're reading* (Unger 2007) and *Theory of type design* (Unger 2018).

The process of fitting typefaces manually is also discussed in users' manuals and guides written for the users of specific font-design software, such as Stephen Moye's *Fontographer: Type by Design* (Moye 1995), David Bergsland's *Practical Font Design With FontLab 5* (Bergsland 2016), and Eben Sorkin et al'.s *Start Designing with FontForge* (Sorkin et al. 2012). Online tutorials that frame the task of letter-fitting within typeface design can also be found, such as Brandon Buerkle's *Spacing a Font* (Buerkle 2018) or Gunnlaugur SE Briem's *Notes on type design* (Briem 1998), as can software-specific guides, such as TypeMyType's *Introduction to spacing* (TypeMyType 2021) for the Robofont editor and Rainer Erich Scheichelbauer's *Spacing* (Scheichelbauer 2013) for the Glyphs editor.

Still other sources document adjusting letter fitting in the context of typesetting a document. As discussed in chapter 1, there is a distinction between type-design fitting and typesetting fitting; the principles discussed in typesetting literature are considered in the historical survey of § 2.2, but are out of scope for the purpose of describing the manual fitting practice of type designers in this section.

2.1.1 The typeface-design process

The procedure for Latin typeface design, as described in these sources, commonly begins with the design of certain key typeforms — frequently designing the lowercase letters first[1] and, typically, starting with the lowercase letterforms that feature simpler profiles: those that are symmetrical on the left and right sides (n and o). This is followed by

1. Bergsland differs by recommending that designers start with the capital letters. However, Bergsland's book is a self-published new-users' guide to FontLab 5, and is framed as a practical introduction to that application, rather than a treatise about typeface design. Where it cites prior authors, such as Tracy, it does so indirectly, via citations from Moye. David Kindersley also described a manual fitting procedure starting with the capitals (Kindersley 1976), although it must be recognized that the purpose of the booklet was to promote acceptance of the automated letter-fitting product he was developing at the time.

Figure 2.1
Iteratively designing letterforms leverages repeated elements in the construction of letters. In this illustration, Henestrosa et al. highlight recurring elements in different colors and patterning (Henestrosa et al. 2017, p. 45). Used by permission.

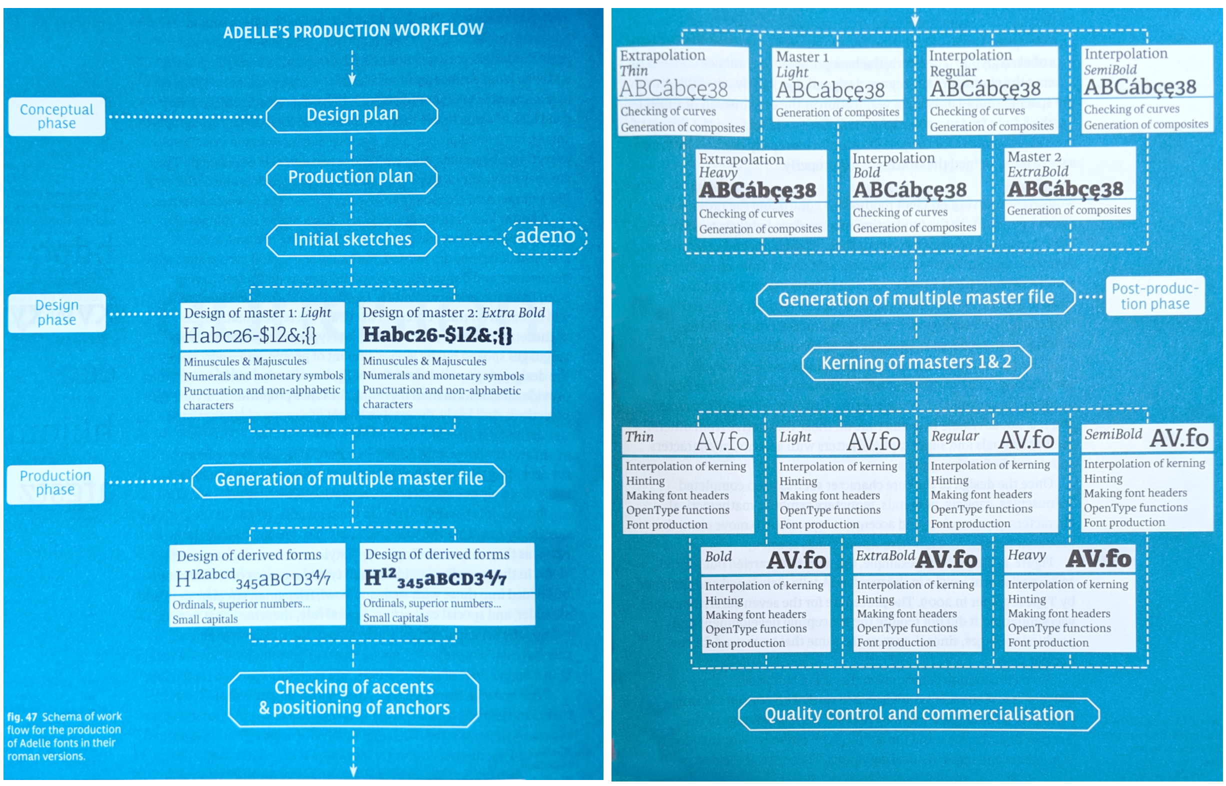
progressively designing more and more typeforms, building off of the forms with simple profiles by recombining the structural elements that recur in multiple letterforms. *(See figure 2.1)*

The lowercase letters are followed by designing capital letters, followed by numerals, symbols, punctuation, marks, and other ancillary typeforms (Smeijers 1996, p. 123; José Scaglione in Henestrosa et al. 2017, p. 57-61; Cheng 2005, p. 8–9; Bergsland 2016, p. 31, 62, 69–89; Unger 2007, p. 116–122). *(See figure 2.2)* This approach to the design of the forms leverages repeated shapes and components in the constructions of Latin letterforms, and is not unique to the digital era. William A. Dwiggins, writing in 1937, discussed a similar process that began with cellulose cut-outs, also starting with the straight-sided and round profiles for simple typeforms, and iteratively expanding the set of forms (Dwiggins 1940 A, p. 2, 4).

Designing each of the letters and other typeforms requires the typeface designer to address and resolve any number of design problems, including construction, consistency, proportion, balance, optical adjustments, and consideration of weight and contrast. The designing of the individual typeforms is an iterative loop that may involve repeatedly refining or redesigning each typeform to improve it with respect to these design problems. Determining the fitting for the typeface is similarly iterative, consisting of determining a left sidebearing and a right sidebearing for each typeform (Henestrosa et al. 2017, p. 81).

A persistent difference of opinion exists on the question of exactly when letter fitting should be performed during the design of a typeface.

Figure 2.2 An overview visualizing the multiple stages in a contemporary typeface-design process (Scaglione in Henestrosa et al. 2017, p. 57–58). Used by permission.



The primary distinction is whether the fitting task should be undertaken repeatedly at multiple points in time, at each point addressing the new typeforms designed or revised since the previous iteration, or whether cursory sidebearings should be assigned to each new typeform during the design stage, and the fitting of the complete set undertaken only at or near the end, at the point when all of the core typeforms are in their final or near-final forms. Scaglione says that typeforms should be fitted as they are designed (Scaglione in Henestrosa et al. 2017, p. 60,), while Henestrosa recommends applying a simple set of provisional sidebearings initially and addressing fitting for the typeface collectively only as the typeface nears completion (Henestrosa et al. 2017, p. 80–81).

2.1.2 The letter-fitting stage

Regardless of when the task of fitting is undertaken, the task is represented as separate from the task of designing the forms (Unger 2018 p. 123; Henestrosa et al. 2017, p. 80). This separation of the tasks has historical precedent in metal type founding, when the design of forms, the cutting of punches, and the justification of matrices were often jobs handled by different individuals (Smeijers 1996). Tracy observed that Eric Gill and Reynolds Stone were both content to have their type designs fitted by others without consultation, while Dwiggins preferred to remain actively involved in the fitting stage (Tracy 2003, p. 71–72).

The process of determining the fit for a particular typeform is described as requiring the assembly of test sequences of typeforms, evaluating the sequences optically, then making iterative adjustments to the sidebearings until the test sequences pass muster. The optical evaluation is variously described as a judgment of rhythm (Unger 2018, p. 124), equality of inter-letter areas (Henestrosa et al. 2017, p. 82; Smeijers 1996, p. 30), evenness (Hochuli 2015, p. 25), or balance (Kindersley 1976, p. 18; Smeijers 1996, p. 27). The test sequences recommended vary from author to author and depend in part on whether fitting is attempted with a complete set of letterforms or is attempted iteratively as new forms are designed. As the number of fitted forms increases, more complex test sequences can be employed, up to and including full test pages populated with real text.

As with the design of the letterforms themselves, most sources recommend starting the fitting process with ‘control’ letters, then gradually expanding the set by fitting new letters in conjunction with those previously fitted. The most common letters recommended for Latin text are those also recommended to be the first designed: n and o in the lowercase and H and O in the capitals. As was the case with the design of the letterforms, this choice is recommended because those letters feature more-or-less symmetrical left and right side profiles and can thus be given the same sidebearing on the left and the right. Here, too, the regularity of

Latin letterforms means that the fitting determined for these control letters can subsequently be propagated to multiple other forms.[2]

Kerning is portrayed by most sources as an additional task within the larger job of designing a typeface, distinct from the task of fitting. The reason cited is that kerns can be added to make an adjustment to the space between any two typeforms, and that the sheer number of possible permutations means that there will always be some pairs of forms that, when side by side, fail to pass the optical evaluations used for determining the primary fitting (Cheng 2005, p. 226; Henestrosa et al. 2017, p. 89). The inevitability of such pairs might suggest that the fitting task itself is not converging on a solution, but most sources frame it in a different light: openly acknowledging the number of permutations, but insisting that the fitting task succeeds for the overwhelming majority of the permutations. It is noteworthy that the pairs of forms most often highlighted as requiring kerning are those that involve the less common profile shapes. The straight and round profile shapes cover the majority of the letterforms, so the majority of the permutations succeed without any kerning required.

As discussed in chapter 1, it is also important for the scope of this research project to recognize that the optical evaluations of space employed when undertaking kerning are the same evaluations employed when performing the fitting task. Kerns, according to the contemporary practice, exist to bring a small number of unusual permutations into agreement with the same core principles that address the sizeable majority of the permutations; the kerns are, therefore, a reinforcement of the successful fitting, not a divergence from it.

Although the process of fitting is described as one that typeface designers can learn through repetition (Smeijers 1996, p. 30), it is often presented as challenging to master. Unger admonishes the reader that some individual letterforms are ‘tricky’ to fit and may feature optical illusions that mislead the designer into making poor fitting decisions. Furthermore, he warns of the complexity caused by the number of permutations, saying ‘many combinations of characters need special attention’, and advising that the designer should plan to conduct fitting tests ‘in many different combinations and for many languages’ (Unger 2018, p. 124).

Practical costs are associated with the manual fitting of typefaces as well. Peter Karow speculated in 1994 that then-new digital font formats were capable of storing pairwise spacing values for every permutation of letter pairs, but that the limiting factor would be the time and resultant financial burden of determining the spaces (Karow 1994, p. 248).

Whether due to the challenge or to the time requirement, some authors describing the fitting process go so far as to label it a hardship. Henestrosa sums up his viewpoint on the relationship between letter design and letter

2. Perhaps worth noting is the fact that starting the fitting process with n and o is advice that can be followed when fitting a nearly complete set of forms, but which can also be followed when fitting each form as it is designed, if one also follows the recommendation of designing n and o first.

fitting by saying 'Drawing letters is fun, but spacing is tedious. Very tedious. That is why I prefer to think that drawing is more important: in this way I can spend more time enjoying myself than being bored' (Henestrosa et al. 2017, p. 80), parenthetically adding in jest that 'anyone would rather be tortured by the Spanish Inquisition than be forced to space fonts forever'.

2.1.3 Classes of typeforms

Critical to understanding the task of fitting as it is practised is a recognition that many of the statements made about letter fitting carry some implicit assumptions about the class of typeforms they will be applied to. The most general domain to consider is the writing system, which is generally specified or can be determined by the context of the surrounding source or discussion. The sources referenced in the previous section (as well as the axioms listed in the subsequent section) all apply to fitting Latin text, because that was the selection criterion of the study. But even within the Latin-text context, several principles are formulated as statements about equality or similarity that, by necessity, can only be understood as applying to *some subset* of typeforms that all belong to the same classification: the letters, the letters of a particular case, the numerals, all alphanumerical forms, all typeforms, et cetera.

In Latin, lowercase letters are fitted with respect to other lowercase letters; capital letters are fitted against lowercase letters for use in running text, but should be separately fitted against other capital letters when typesetting all-capital text (Unger 2018, p. 125; Hochuli 2015, p. 23–25). It is only within each classification that the other statements about how to determine fitting operate. Most of the sources consulted in section 2.2's historical study explicitly make the distinction between lowercase-with-capitals and all-capitals.

Less is said about fitting numerals, however, except to note that in many contemporary typesetting environments, users of typefaces will expect numerals to feature tabular alignment when setting columns of data (Tracy 2003, p. 76; Henestrosa et al. 2017, p. 85). *(See figure 2.3)* A modern

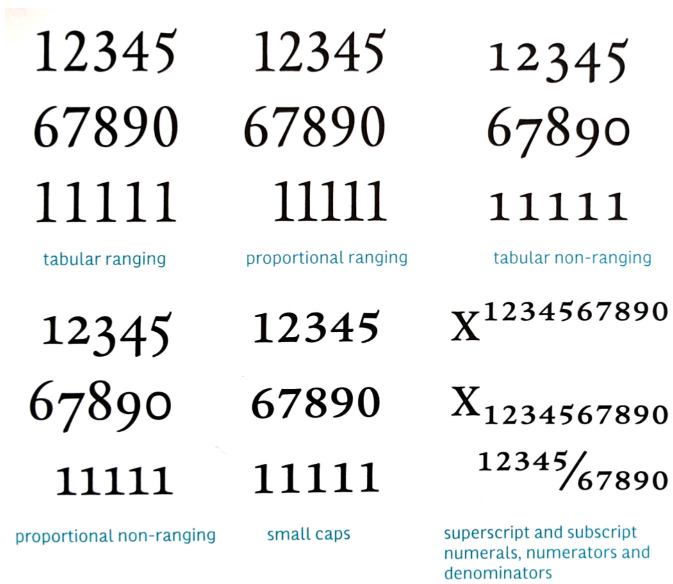


Figure 2.3
Issues particular to the fitting of numerals arise because numerals are often typeset in contexts not meant for continuous reading, such as tabular data. The *proportional* examples in the illustration depict fitting similar to that of letterforms, but proportional fitting may not be the default. The differences are most clearly seen in the 1, but all numerals can be affected. (Henestrosa et al. 2017, p. 85). Used by permission.

digital font can supply default fitting appropriate for numerals that are set within texts meant for continuous reading by fitting them like lowercase letterforms (Beier 2017, p. 150) and supply separate tabular fitting appropriate for numerals used in columnar data, but there are no guarantees that the software used to typeset documents with the font will support both options or make them available to the user. Complicating matters further, typefaces can include two distinct designs for the numeral forms (ranging and non-ranging) requiring distinct fitting. *(See again figure 2.3)*

Considerably less still has been written about fitting for non-alphanumeric symbols and punctuation, apart from general advice that punctuation is a frequent target for kerning pairs. The paucity of discussion suggests that, at the very least, if there are distinct rules applying to some classes of typeforms but not to others, then either it is the letterforms that are subject to the most stringent rules — with numerals, symbols, and punctuation posing less difficult fitting problems — or else the letterforms are so much more important to legibility, readability, and other metrics of good fitting that any special rules for the other classes of typeform are of less concern.

There are arguments to be made on either side. Good letter fitting is vital to readability, it is thought, because readers identify letters and combine them into words. As such, numerals and punctuation typically serve a purpose in the text that is distinct from word formation. Every digit of a date, monetary amount, or phone number must be taken in individually for it to be correctly parsed by readers. Gerrit Noordzij explained that the distinction is that each digit has a conceptual meaning which changes if it is moved, saying ‘Unlike letters, numbers stand on their own’ (Noordzij 2000, p. 184). Similarly, the purpose of a period or question mark is to demark a meaningful break between words. But there are also exceptions to be found wherein numerals and punctuation do participate in words; ordinal numbers like 1st and 20th can be written with numerals that function like letterforms and are read as such, and instances where punctuation forms part of a word are commonplace.

In the discussion that follows, the distinction between all-capital fitting and capital-and-lowercase fitting is preserved because it is the distinction maintained in the source material. Where it can be addressed, the question of engaging with other classes of typeforms will be treated explicitly but, in general, it may suffice to regard ‘typeforms that make up parts of a word’ as being equivalent to ‘letterforms’ when considering fitting.

2.1.4 Representations of letter fitting in font files and font editors

Regardless of the processes employed to determine the fitting for a typeface, it must somehow be implemented and stored in the final product. When type was manufactured in metal or wood, fitting decisions could be made either when cutting the punches or when justifying the matrices.

Those decisions were then fixed into the physical substance of the type: sidebearings as the dimensions of the individual types, and any kerns as overhangs or undercuts that were, again, physically integrated into the objects making up the product. When type transitioned to a representation of the design stored on photographic plates and, subsequently, digital storage, the fitting was initially preserved in the stored design, but could be adjusted on the fly during typesetting.

The digital vector fonts in contemporary usage are typically distributed in files based on the SFNT structure, which stores the contents of the font within a number of separate data tables. TrueType and OpenType are widely-used file specifications for what data tables should be included. In these SFNT-structured file formats, each typeform is represented as a 'glyph' entry consisting of contour shapes that are stored in one table, plus numeric dimensional information stored in other, 'metrics' tables. The fitting of the glyph is stored as a numerical `advanceWidth` and a numerical `lsb` offset. The glyph's left sidebearing is equal to the `lsb` offset. The `advanceWidth` is the horizontal distance that the rendering system (whether for print or for on-screen display) should advance after it has drawn the current glyph, before it begins drawing the next glyph. *(See figure 2.4)* The right sidebearing is, thus, is not stored explicitly but is computable as the `advanceWidth` minus the glyph's rightmost extreme point. Either sidebearing value could be a negative number. The `advanceWidth` and `lsb` values are both specified in 'font design units', which are an internal numerical scale derived from the 'units per em' (`UPEM`) value stored in the font file.

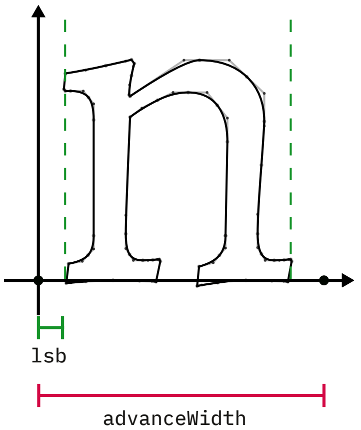


Figure 2.4
The `advanceWidth` and `lsb` entries as stored in many digital font formats. The right sidebearing is not stored directly and must be computed (illustration by the author).

Font editing programs, however, often attempt to simplify the presentation of fitting data in the user interface, such as by showing the left and right sidebearings directly, both as editable numerical fields and as guidelines on the drawing canvas, rather than exposing the potentially confusing underlying details of advance widths. *(See figure 2.5)* In both the SFNT-structured font files and in the user interface, it is crucial to note that each sidebearing value is a linear distance measured from the outermost point in the form — including the lengths of any serifs, overhangs, in-strokes, and out-strokes. The spaces between adjacent forms are two-dimensional areas. To evaluate or modify the two-dimensional area between letterforms, the full shape of the forms' profiles must be evaluated.

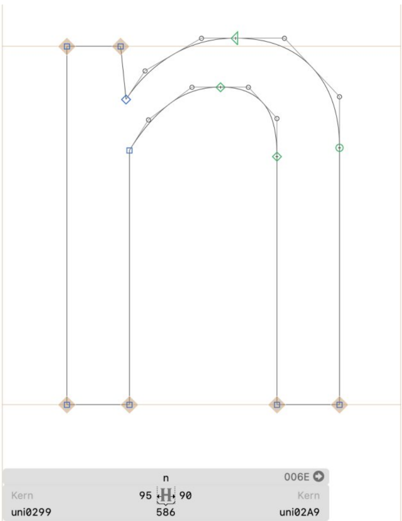


Figure 2.5
Screenshot of the sidebearings for a typeform as presented in the user interface of the Glyphs font editor. The left and right sidebearings are both presented directly to the user in the grey information box (screenshot by the author).

Kerns are stored in separate tables. Earlier TrueType fonts used a dedicated `kern` table, while newer TrueType and OpenType fonts use the `GPOS` table, tagged with a semantic feature name: *`kern`* is used for the generic kerning feature, although other tags are defined.[3] The kerning information itself can either be pair-based, in which a list of specific two-

3. Semantically-tagged features for spacing were added by the OpenType Layout extension to the original OpenType specification. Several other fitting-related semantic tags are officially registered, including *`cpsp`* for capital-to-capital spacing, *`tnum`* for numerals with tabular spacing, *`pnum`* for numerals with non-tabular spacing, and *`dist`* for adjusting positioning for orthographic correctness (mainly in Indic and other Brhami-derived writing systems).

glyph sequences is stored with a numerical plus-or-minus adjustment to be applied when the pair occurs, or class-based, in which the same adjustment data can be designated for multiple set of similar forms (such as all of the unaccented and accented versions of a given letter, or all forms that are round on the left side).

Multiple other digital file formats have been used over the years. Notably, several of those formats also stored fitting information as an advance-width plus the left-sidebearing offset, even if the data was formatted in a different structure. Adobe's Type 1 font format and the TEX typesetting system, for example, also stored fitting as advance-widths and left sidebearing offsets, but with that data saved separately in 'font metrics' files (Haralambous 2007, p. 626, 674). However, other font formats have incorporated entirely different approaches to storing fitting information. Agfa Compugraphics's Intellifont format (Karow 1994, p. 156) could store vectors of kerning-adjustment values at different heights for every glyph, with separate vectors for the left and right sides, and separate values designated for 'text' and 'display' sizes. Each vector represented glyph widths defined at various vertical 'sector' heights measured against the glyph.

The technique was correspondingly known as sector kerning and was also employed by typesetting systems offered by software vendors in the 1970s and 1980s (Ward 1990). After Apple's TrueType was adopted by Microsoft in 1989, followed by Adobe's merger of PostScript Type 1 format with TrueType into the combined OpenType specification (Bigelow 2020 A; 2020 B), the market for alternative digital font formats was effectively killed, and sector kerning disappeared with it — though, as will be seen in § 2.3, the sector-kerning concept was not permanently lost.

2.2 Study of historical letter-fitting theory

As mentioned in the preceding section, the task of manually fitting letters is the subject of numerous discussions, guides, debates, and other written accounts. Although the primary research question explores to what extent that task can be automated or implemented in an algorithm, understanding the task itself requires understanding the fundamental principles involved when it is conducted manually by type designers. To identify these principles, a study was conducted of the recorded theories of letter fitting in typeface design.

Attempting to comprehend and describe the theories that underpin manual practice of letter fitting is certainly not the only possible approach to the problem but, in the past, it has been shown to be more fruitful than attempting to build a letter fitting algorithm on a theory disconnected from the craft as practised. Records exist of such attempts. Peter Karow recounted a 1980 project at URW to implement a letter-fitting algorithm modelled on magnetic forces (Karow 1998; Karow and Blokland 2013). Sebastian Kosch made a similar attempt in 2010 (Kosch 2010 B). In both instances, the researchers abandoned the line of investigation and subsequently pursued algorithmic models based on automating the processes of manual letter fitting.

Other researchers have posited theories about letter fitting based entirely on a mathematical or scientific principle, such as Raph Levien's 2006 exploration of assessing letter fitting with wavelets (Levien 2006). These indirect approaches have seldom survived beyond the initial proposal. Conversely, numerous projects initially focused on automating or encoding practice-based theories of letter fitting have arisen independently and persisted as active projects, even if their customer or user base remained modest.

Where exactly one draws the line between a model that is based on manual letter fitting and one that is not based on manual letter fitting requires some interpretation, as a matter of course. A purely mathematical model might be so far removed from the act of adjusting typeforms that one type designer would call it incurably abstract, while, to the researcher who derived it, that same model might appear to be a clear, direct distillation of reality.

For the sake of the present investigation, however, the degree of abstraction in the eventual algorithm is a distant concern. The starting point is the practical body of knowledge and techniques possessed by type designers. The aim of the study is to scrutinize the theories of letter fitting within that body of knowledge and techniques then examine them systematically, in order to acquire a picture that represents the commonly accepted approach, complete with its details, leeways, limitations, balances, and compromises.

Typeface design is a practical discipline; for most of its history, the written record of how it has been performed takes the form either of literature written by practitioners (either to explain the craft for the comprehension of outsiders or to train new typeface designers) or of deliberations between practitioners about the merits of various techniques. The study conducted of letter-fitting theory began with these sources, including literature from typeface design and type manufacturing history, augmented by educational sources and interviews with type-design educators.

The historical survey continued with a look at the corresponding literature from related letter arts (such as lettering, writing, stonecarving, and calligraphy), in order to catalogue the theoretical principles that those practices use to explain the behaviour of space between letters. This was followed by a study of the theories described in prior work on the automation of letter fitting, including published writing, promotional materials, patented inventions, and published software.

In each case, the procedure undertaken was to isolate the concepts in those sources that were used to describe how letters should be correctly fitted and to observe how those concepts were said to work in concert with each other. The common concepts accepted across sources, in theory, could be further distilled into a set of well-defined, core principles. This set of principles, in turn, can serve as a framework in which to appraise existing letter-fitting automation tools and as the basis for developing new automation algorithms.

2.2.1 Literature, practice, and education in type production

The first stage of the study examined material specifically from the fields of typeface design and type making. In its early history, many facets of the printing business, including the processes of designing and manufacturing type, were trade secrets characterized by fiercely protective attitudes of their creators towards the dissemination of knowledge. Later historians have gleaned much about the technology and norms of cutting punches and adjusting matrices (Burnhill 2003; Blokland 2016), but there are scant references addressing how letter-fitting decisions were made.

Fred Smeijers provided a chronological overview of the sources in 'Putting letters next to each other' (Smeijers 2014), noting that it was not until the late 17th century that manuals recorded the details of justifying matrices for public consumption. These earliest sources already appeal to principles that endure to the present day. Pierre Simon Fournier's 1764 *Manuel Typographique* (although not the first such manual) outlines the 'setting' of types beginning by filing down the sides of a sequence of three m strikes, with the middle m inverted, until their vertical strokes appear to be in an even visual rhythm, then applying the resulting sidebearing measurements to the other forms, adjusting for round and diagonal profile shapes (Fournier *trans.* Carter 1930, p. 158–161).

A persistence of the core principles cited can be observed, with new assertions and new debates generally appearing only in response to substantial changes in printing technology and type production. Joseph Blumenthal's 1935 advice for fitting foundry type (Blumenthal 1935) differs little from Fournier's, for example. But the introduction of photo-composition in the mid-20th century suddenly permitted fitting letters so close that they collided and even overlapped.[4] The sources from that point forward began to include warnings about the ills of overlapping forms.

There are occasional discussions of letter-fitting practice found in the biographical and autobiographical works about type designers and other practitioners during the eras of metal type and photocomposition. Some, such as the compendium of Adrian Frutiger's professional work edited by Heidrun Osterer and Philipp Stamm, relate specific problems encountered regarding fitting (Frutiger et al. 2021), but it is rare to find accounts of the

4. Collisions between printed letterforms were certainly possible in even the earliest metal-type printing through overprinting or alignment trouble; it became a design-time concern during photocomposition (Hochuli 2015, p. 26; Tracy 2003, p. 78).

decision-making process in such retrospectives. Isolated sources exist that document the design of individual typefaces, but even these are often written long after the fact — and often either for specific typefaces seen to hold special significance or as part of biographical or autobiographical profiles of particular type designers.[5] Less common is correspondence, but where it exists it can be revealing. William A. Dwiggins's exchanges about fitting with Chauncey Griffith at Linotype are candid and detailed.

Digital typesetting followed shortly after photocomposition, but it was the advent of personal computers and desktop publishing that had the bigger impact on the written narrative (Middendorp 2018, p. 11). Whereas, before, typefaces had been designed almost exclusively in an industrial context, affordable desktop computing empowered small digital foundries and individuals to modify and create typefaces and subsequently use or sell them, all using commodity hardware (Southall 2005, p. 156).[6] Guides to typeface design appeared as books offered to the public, both those framed as practical handbooks for using a specific font editor (Moye 1995; Sorkin et al. 2017; Bergsland 2016) and those illuminating the craft and logic of the task in general (Smeijers 1996; Cheng 2005; Henestrosa et al. 2017; Campe and Rausch 2022). As discussed in § 2.1, letter fitting constitutes part of the core subject matter for these guides.

Online sources account for a predictably high percentage of the written discussions about letter fitting produced since the start of the web-publishing era. These, too, include guides aimed at users of particular font-editing applications (Scheichelbauer 2013; TypeMyType 2021) and tutorials written without focusing on a specific application (Briem 1998; Buerkle 2018). Simultaneously, online discussion channels, including email discussion lists, social-media networks, and web-forum sites, provide for a higher-volume and more rapid exchange of ideas between practitioners than can be seen from the era of printed newsletters and periodicals.

Interviews were also conducted with type educators to characterize how fitting is described in academic coursework, commercial workshops, and the onboarding processes for newly-hired typeface designers at type foundries.

2.2.2 Related principles from lettering, writing, and calligraphy

The second stage of the study examined the principles of fitting space between letters as it is described in related arts beyond the craft of typeface design. This may sound surprising, given that type, as defined for this project in chapter 1, concerns pre-made letterforms that are combined during typesetting. This pre-made nature distinguishes type from

5. Exceptions can be found, however, such as invited essays on the subject, e.g., John Dreyfus's exploration of spacing in *Font* — ostensibly a book reflecting on the work of Sumner Stone, but covering broader matters (Dreyfus 2000).

6. The pricing of font-editing software was similarly democratizing. In 1989, Kingsley/ATF charged $55,000 per seat for Type Designer, plus per-font royalties if the user wished to sell fonts created with the program (Seybold 1989). That same year, Altsys's Fontographer retailed for $495 (Ponting 1989).

disciplines where each letterform is designed and executed *in situ*, tailored to the final composition — typically with a pre-determined text. In these arts, such as writing, calligraphy, stonecarving, and lettering, the practitioner is concerned simultaneously with determining the shapes of bespoke letterforms and with the distribution of the surrounding space in the piece.

As with type, there are design norms and ideals that constrain the shapes of letters. But, even if one ignores the decorative end of the spectrum and focuses on letters meant to be read comfortably, in lettering arts the individual letterforms themselves are as malleable as the spaces between them. Thus it can be more ambiguous to discriminate the reasoning about space from the reasoning about positive forms. Nevertheless, within a given writing system, type design and the crafting of custom-fitted letterforms both share the same audience of readers and the disciplines are preceded by a shared and overlapping (even intertwined) history. So, though there is a limit to how much the literature of lettering arts can speak to the designing and fitting of type, it may shed a slightly different light on many of the same underlying principles.

Furthermore, the various lettering arts have been accessible as individual pursuits for centuries, which has led to a greater number of descriptive and instructive texts. In contrast, as was noted earlier, the making of type was an industrial job requiring 'whole teams of people' until digital fonts and desktop computers transformed it into a pursuit available to individuals (Scaglione in Henestrosa et al. 2017, p. 17). Historical manuals and guides for the lettering arts pre-date the rise of movable type for Latin and have continued to be published to the present day.

As with the historical sources on typemaking, many of the core principles related to letter fitting found today can be traced back to the earliest writing on the subject. Perhaps the earliest example is Ludovico Vicentino degli Arrighi, whose 1522 *La Operina* instructs calligraphers that the distance between joined letters should be the width of the counter in n (Arrighi 1522, p. 19). Many practitioners, of course, engaged with both typeface design and the related arts, and emphasized their similarity. Gerrit Noordzij connected Latin typeface design to calligraphy explicitly (Noordzij 1973; 1982; 2009), including his discussions on the relationship between interior and exterior space. The connection is reinforced within the lettering arts as well, such as in calligrapher Michael Harvey's usage of commercial typefaces as examples for letterers to study (Harvey 1996).

2.2.3 Prior explorations into automating letter-fitting

The final stage of the study considered prior projects to automate the task of letter fitting, either entirely or in part, by looking at patent filings, product announcements and reviews in trade journals, brochures and user's manuals, and, where available, software source code. In the interest of reproducible research, the search for patent filings was limited strictly to expired patents. The study also filtered out products that dealt solely with the logistics of storing fitting data, such as third-party libraries of kerning tables sold for popular fonts — though that distinction is not always made clear in promotional literature or trade-journal coverage.

Note that this study explored how the projects define and explain the principles of letter fitting, rather than examining how the projects implemented it. Inspection of the implementations' internals, where possible, was conducted later, as described in chapter 3. Not all projects specify principles for fitting; some level of 'spacing automation' feature was found in almost every commercial digital type-design program, but the accompanying material rarely explored the theories involved.

Efforts to automate portions of the task of typemaking and typesetting date back to the industrial era, but the early mechanical inventions involving fitting were production aides to speed up manufacturing or typesetting and did not capture the decision-making process of fitting letters. This includes Linn Boyd Benton's 1883 patent for 'self-spacing type', which, despite its name, was merely a unitized casting machine that supported six fixed body widths (Cost 1986, p. 25–28). Similar inventions that regularized typesetting but required the designer to pre-determine the fitting were produced in the early-to-mid 20th century, such as Letraset's alignment systems (Dowzall 1982; Dowzall and Houssian 1986). A notable outlier here is the work of stonecarver and typeface designer David Kindersley, who began researching letter-fitting in the 1950s using optical measurements of the light levels transmitted and blocked by letterforms, before moving into the realm of software.

Indeed, the adoption of digital typesetting precipitated a profusion of letter-fitting automation efforts. Predictably, the number of projects discoverable in the public record skews heavily towards the 21st century. *(See figure 2.6 below and table 2.1, over page, for a chronological list)* As Dwiggins's correspondence and Kindersley's optical fitting project illustrate, type designers have contemplated fitting automation for much longer, but the rise of desktop-computer–based typeface design — and, no doubt, scriptable design software — provided new avenues for interested parties to explore the task on their own.

In addition, the availability of free-to-use software-hosting services like GitHub, SourceForge, and Google Code meant that many automation projects were published online and remain available years later, even if the project goes dormant. Comparable details about work done in earlier

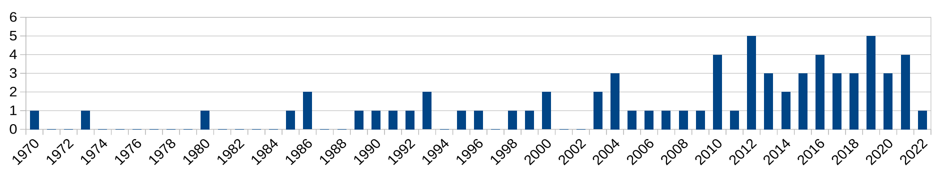


Figure 2.6
Per-year count of the number of letter-fitting automation projects identified for the historical study. The counts include all projects identified for study, including those projects for which detailed information was ultimately unavailable.

Table 2.1
Chronology of prior letter-fitting automation projects identified during the historical survey. Where not otherwise specified, the projects were focused on fitting Latin text.

Project	Year	Publisher or lead authors	Scripts addressed (other than Latin)
LOGOS	1970	David Kindersley, Neil Wiseman	
Bell sector-spacing	1973	Max Mathews, Bell Telephone	
URW force experiment	1980	Pater Karow, Margret Albrecht	
Arabic Calligraphic Engine	1985	DecoType	Arabic
Tracy method	1986	Walter Tracy	
Gerber	1986	David Logan, Gerber Scientific	
Type Designer auto feature	1989	Kingsley/ATF	
Sector Kerning	1990	Compugraphic	
hz-program kf	1991	URW	
Fontlab auto feature	1992	Fontlab Inc	
Kernus	1993	URW	
InfoKern	1993	InfoComp	
Canon Shift	1995	Kiyoshi Watanabe, Canon	
Omron	1996	Sawada et al., Omron	Japanese
Neville	1998	Paul Neville, William Fox	
TypeArt	1999	Calamus	
FontForge auto feature	2000	FontForge project	
AFDKO	2000	Adobe	
Aldine FFT	2003	Sergei Egorov	
KERNDICT	2003	Thomas Baruchel	
sqtroff	2004	SoftQuad	
KernMaster	2004	DTL	
Perturbation	2004	Cameron Browne et al., Canon	
Sousa method	2005	Miguel Sousa	
Wavelet masks	2006	Raph Levien	
Cambria OpenType Math cut-ins	2007	Microsoft	
iKern	2008	Igino Marini	
LetterModeller	2009	Frank E. Blokland	
Caslon Fourier analysis	2010	William Berkson	
Rhea	2010	Sebastian Kosch	
Rhea force experiment	2010	Sebastian Kosch	
SortsMill spacing by anchors	2010	Barry Schwartz	
Tsukurimashou	2011	Matthew Skala	Japanese
Autokern	2012	Charles M. Chen, Typefacet	
Blur-masking	2012	Peter Weigel	
Impallari macro	2012	Pablo Impallari	
OpticalLetterSpacing.js	2012	Gabi Schaffzin	
Typebutter	2012	David Hudson, Joel Richardson	
CJK Auto Spacing	2013	Xin Yue	Chinese, Japanese, Korean Hangul
Kernagic	2013	Øvind Kolås	
Monokern	2013	Edward Cree	
font-prediction_mahout	2014	Ethan Petuchowski	
Novi Sad statistical analysis	2014	Bojan Banjanin, Uroš Nedeljković	
BubbleKern	2015	Toshi Omagari	
LS Cadencer	2015	Lukas Schneider	
Fittingroom	2015	Sebastian Kosch	
Black Spacer	2016	Jérémie Hornus, Black Foundry	
HT Letterspacer	2016	Andrés Torresi, Huerta Tipográfica	
Spaceman	2016	Simon Cozens	
KernKraft	2016	Mark Fromberg	
Electric kerning	2017	Matthew Skala	
Octabox	2017	Martin Hosken, SIL	Arabic
Machine Learning of Fonts	2017	Antanas Kascenas	
Atokern / kerncritic	2018	Simon Cozens	
psoptkern	2018	Raymond Luckhurst, Scriptit	
KernBot	2018	Joey Grable	
HT Kerner	2019	Simon Cozens	
electricbubble	2019	Sebastian Kosch	
YinYangFit	2019	Sebastian Kosch	
CounterSpace	2019	Simon Cozens	
fontmetrics	2019	Simon Cozens	
RhythmInfluencer	2020	Maarten Renckens	
type.tools AI	2020	type.tools	
Andersson experiment	2020	Rasmus Andersson	
Hands Face Space	2021	Simon Cozens	
Kern On	2021	Tim Ahrens	
Kerning	2021	Zeeshan Asghar	Arabic
Building a spacing calculator	2021	Dean Kalen	
Kern Determiner	2022	Simon Cozens	Arabic

decades is progressively harder to find: if the project was not patented or the subject of a published account, it may be entirely lost to history.

There have also been efforts aimed at making letter-fitting decisions solely by statistically analysing the sidebearings and kerning data of large sets of typefaces, although there is not much to show for the effort. Karow undertook such analysis as part of a large statistical survey of the URW library (Karow 1993). The published conclusion was that the statistical model could not predict sidebearings based on the measured features of a typeface, but neither the model nor the raw data was published.

More recently, it has become popular to apply machine-learning models to construct the statistical model, with the same essential goal: analyse a large set of typefaces as input, and predict the letter fitting for new typefaces based on the analysis. Little has been gained from such projects, either, perhaps because the models tend not to take typographic variables (weight, optical size, style, etc.) into account, leading to questionable methodologies, such as measuring linear sidebearings without adjusting for the lengths of serifs (Kascenas 2017; Banjanin and Nedeljković 2014 A).[7] Regardless, purely statistical models are less relevant to this survey because they do not posit an underlying theory for fitting letterforms.

2.2.4 Continuity

Considering the written record as a whole, it is the continuity that stands out. At times it is overt. Walter Tracy's *Letters of Credit* is perhaps the most common reference where letter-fitting is concerned; the book is a reflection on the practice of making type, but it is detailed enough about the practice (including the letter-fitting process) to serve double-duty as instruction for new typeface designers. As noted in chapter 1, Tracy detailed a heuristic method for assigning sidebearings to the basic Latin alphabet, crediting the system to Harry Smith at Linotype 'over thirty years' prior to the 1986 first-edition publication date (Tracy 2003, p. 72). That heuristic, down to the exact tables Tracy included for capitals and lowercase letters, has continued to be reproduced by other writers into the 21st century. *(See figure 2.7, over page)* Notably, Tracy presented the heuristic as pragmatism distilled from experience; after much repetition, it is often treated more like a formula.

Elsewhere, it is the terminology, illustrations, and framing of rules that is repeated between sources. *(See figure 2.8)* Patterns can be discerned in the language, carried through even to the software-based automation projects examined in the final stage of the study. The fitting-automation projects that survive and thrive derive their inspiration from the same conventional wisdom seen in the literature — proposing novel approaches only rarely and often without success. This can be seen in the novel force-based model explored by URW, which was dropped and replaced by the *kf*

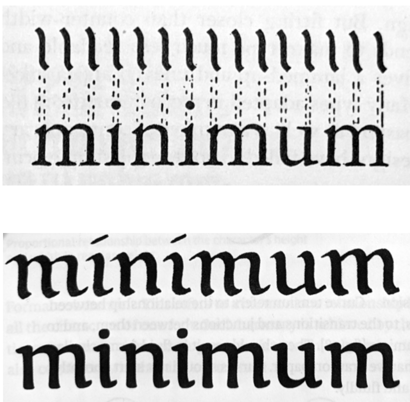


Figure 2.8
Top: Illustration from Blumenthal in 1935 relating letter fitting to calligraphic stroke rhythm (p. 73).

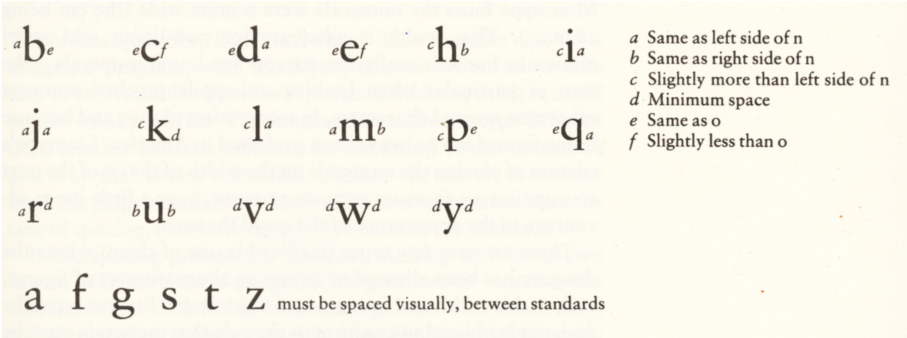
Bottom: Henestrosa et al. make the same relation using a similar illustration in 2017 (p. 33).

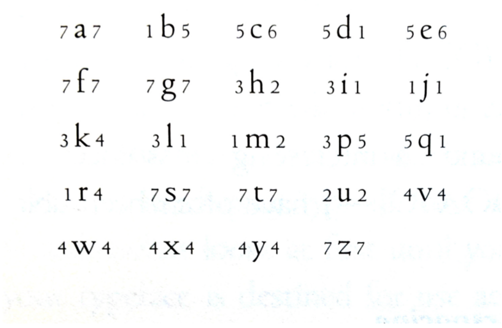
7. In 2020, Nic Schumann, Cem Eskinazi, and Marie Otsuka presented preliminary findings from a machine-learning–based project that did seem to take typographic variables into account but, as of this writing, the project has yet to publish results (type.tools 2020).

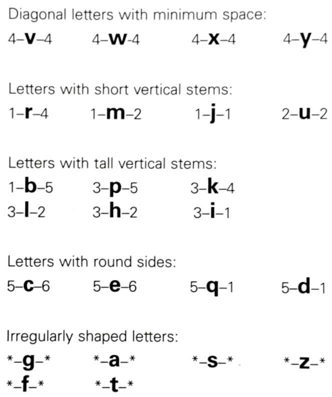
Figure 2.7
Walter Tracy's heuristics for fitting lowercase Latin letters, reproduced by multiple authors over the years.

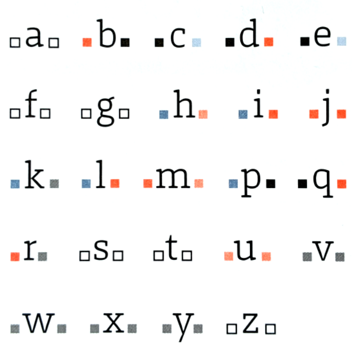
Top: the 1986 table presented by Tracy (Tracy 2003, p. 75).

Below: tables presented in Moye 1995 (p. 81), Cheng 2005 (p. 221), and Beier 2017 (p. 146).









module in *hz-program* (see chapter 1). The patent filing for *kf* cites a list of six fundamental letter-fitting principles, each derived from traditional preparation (Karow et al., 1992).

2.2.5 Complexity

A key factor that differentiates sources in the written record is the degree of complexity found in their letter-fitting methods. An increased complexity in how the letter-fitting task is described or implemented often correlates to a greater maturity and refinement of the tools and of the output medium.

Fournier, for example, cites the need to balance the interior and exterior space of typeforms, but provides only general guidance that strokes should appear to be in consistent visual rhythm and that certain forms require less exterior space than the standard, m. His method, therefore, incorporates one rule (balancing the interior and exterior areas of space) and one exception (the exterior spaces of some typeforms should be reduced). By Tracy's time, the method in use at Linotype has gained further complexity, citing additional rules for typeforms with open counters and testing fitting in triplets of typeforms, and enumerating seven distinct classes of profile shape among the set of typeforms.

The same trajectory can be observed in the literature from lettering and calligraphy — Arrighi's text provides a single piece of advice on spacing (relating the width of n to the standard inter-letter space); centuries later, Jacoby and Schenk had developed substantially more layered rule systems (Jacoby, p. 30–38; Schenk p. 16–18).

That these refinements of letter-fitting methods correspond to refinements in printing technology and media is unsurprising. But it also helps explain the apparent step backwards in complexity seen in letter-fitting automation projects. The earliest software systems tend to implement fitting using a small set of rules.[8] But the early digital typesetting systems provided lower-resolution bitmaps, and only evolved toward vector-based formats and toward outputs rivalling the quality of metal over the course of decades. As the capabilities of the technology caught up part, the narrative of software-based fitting came to more closely resemble that of the other craft traditions. Thus, early software systems to automate letter fitting that dropped back to simpler methods should be seen to do so out of necessity, not because knowledge of fitting was lost or because known techniques fell out of favour.

2.3 Identifying axioms from history and practice

The individual principles cited time and time again by these sources may be considered the basic tenets of letter fitting theory for a particular writing system. For this research, the effort to decompose or reformulate these letter-fitting first principles into axioms is a practical line of inquiry. Algorithms, as defined in chapter 1, § 1.2.4, require a clear and concrete expression. The functionality of a particular computer program can be isolated into discrete components by analysing its functions, data structures, and control interfaces. Thus, to make the most useful head-to-head comparisons between fitting theories that are expressed as computer programs and those that are expressed as manual procedures (as well as comparisons within each group), representing the concepts as axioms provides a more formal, but common language.

2.3.1 Determining inclusion and exclusion of axioms

In mathematics and logic, an *axiom* fills a special role in investigations and proofs: axioms are accepted *a priori* to be true, and serve as the foundation upon which other theories are more rigorously constructed. In other words, the axioms of letter fitting should be the simplest possible statements about letter fitting; starting points from which the more complex ideas and practical advice proceed.

Naturally, not every axiom in the list is cited or considered by every source. Similarly, the authors of any given letter-fitting automation implementation might cite additional concepts as axiomatic, or might explicitly reject certain axioms that others accept. Details of how prior letter-fitting automation work has engaged with the axioms below is discussed in chapter 3, § 3.1. The list that follows only seeks to enumerate the most commonly accepted axioms found in Latin fitting algorithms, practice, and the associated literature.

8. See chapter 3 and table 3.1.

When compiling the list, every effort was made to limit the list to statements concerned purely with the task of fitting typeforms — excluding, for example, precepts about the design of good letterforms as well as more abstract statements about how space affects page-level typography. As stated previously, the practice of letter fitting is interwoven with the practice of designing letterforms but, in order to pinpoint the conceptual first principles of letter fitting, one must assume that the letterforms are, at the moment they are being fitted, correct.

Similarly, it was determined that the axiom list should be limited to principles that are applied to a set of letterforms that is appropriate to be fitted as a set, and not include principles that are guidance about which letterforms require fitting. It is common advice in Latin text fitting to note that, in Latin text that is meant for continuous reading, capital letters are most often set against lowercase letters, and that therefore the default spaces applied to the capitals should be capital-to-lowercase fitting, with capital-to-capital fitting being determined separately. This precept is widely accepted to the point where it could arguably be considered an axiom. For this list, however, it was excluded on the grounds that it is a precept purely about selecting the forms and is not a statement about determining the actual space. In other words, even when this principle is used to choose the correct set of forms, the typeface designer is no closer to establishing the space for any of the forms.

Finally, the list of axioms is meant to isolate the rules applied during letter fitting and not include advice on how a typeface designer should go about finding the solutions. For example, several sources advise flipping a sequence of test letters upside down before assessing if the spaces between the letters appears equal. For the axioms enumerated in this project, it was decided that flipping the letters upside down is merely advice about how to search for an answer, but the underlying question in play is whether the spaces between the letters appear equal; consequently, the axiom is that the inter-letter spaces should appear equal, not that the letters should be flipped upside down.

The same is true for other advice provided by letter-fitting sources, such as advice about using printed samples versus on-screen samples, inverting colours, squinting or standing at a distance, and so forth. Admittedly, there is not always a clear line to draw, and no claim is made that alternate decisions about inclusion or exclusion would necessarily produce weaker results.

2.3.2 Essential axioms for Latin letter fitting

The survey of literature and automation implementations discussed earlier gives the following set of commonly cited axioms for letter fitting Latin text type. The list is broadly sorted in order from most-frequently–cited to least-frequently–cited, but the precise ordering is not crucial. The frequency with which an axiom is cited, by itself, does not denote its relevance (as is explored further in § 2.4 and 2.5).

The designations assigned to each axiom are merely a convenient shorthand to aid subsequent references and illustrations. The prefix 'L-' is meant here to serve as a reminder that these axioms stem from a study of Latin letter fitting practice and should not be assumed to apply in equal importance, if at all, to other writing systems. Some of the axioms are particular to Latin for identifiable reasons, while some have application both to Latin and other writing systems that share similar features, such as a horizontal baseline or bicameral casing.

It should be noted that the phrasing of the axioms generally uses the term *space* to refer to a two-dimensional region of area, but sources in the literature and type designers in online discussions may say 'space' when it is clear in context that a linear distance is referred to. Note, for example, Walter Tracy's heuristic, which says that diagonal-profile forms should be assigned 'minimum space'. In the supporting text, it is clear that Tracy is referring to a minimum sidebearing distance. Where the sources are unclear, attempts have been made to clarify the meaning. The list of axioms is as follows:

> ***Axiom L-1: Profile Similarity*** *— Similar profile shapes should be fitted with similar space.*

Figure 2.9
Similar profile shapes should receive similar space (illustration by the author). Font shown: Tinos.

This axiom states that the amount of space fitted for a particular profile shape should be the same as the amount fitted to that profile shape when it appears in other forms. For example, the space fitted to the left side of o and space fitted to the left side of c should be of equal area. *(See figure 2.9)* This axiom is essentially universally accepted and is usually framed as being self-evident, no doubt because it speaks to one of the most basic tenets of a coherent visual design. It permits the propagation of fitting values from one typeform to other typeforms that have similar shape.

In addition to being virtually undisputed by simplicity, however, this axiom is also the basis for the notion that it is the shape alone that dictates how much space a typeform needs. Namely, it is the visual appearance of, for example, u that defines what space u should receive: not the fact that it is recognizable by readers as the grapheme 'lowercase u', nor its Unicode code point, nor its place or history with the alphabet.

Figure 2.10
The similarity of any two profile shapes is not determined by the letters' graphemes and may vary based on style. The italic v on the right has a round profile, which is appropriate for the italic style (illustration by the author). Fonts shown: IM Fell French Canon, upright & italic.

This distinction comes to the forefront whenever fitting algorithms or literature state rules or conditional tests phrased as references to the letter. For example, Walter Tracy's heuristic states that the right side of p receives the same space as o, but v and w receive minimum space. That heuristic only applies for upright Roman styles where the v and w have diagonal profiles (as are seen in his illustrations). In an italic design where the v and w take on round profile shapes, however, axiom L-1 insists, sensibly, that the right side of the v and w receive spaces similar to other round profiles, because of their profile shape. *(See figure 2.10)*

Figure 2.11
Axioms about profile shapes can dictate different fittings based on construction (illustration by the author). Font shown: Gentium Plus.

Figure 2.12
Different angles are used for the side profiles of v and w, but the two shapes are often described as similar enough to be fitted with the same space (illustration by the author). Font shown: Gentium Plus.

Figure 2.13
The principle that similar profile shapes should receive similar space also applies when the profile shapes are reflected in the horizontal direction (illustration by the author). Font shown: Tinos.

It must also be observed that in order to implement this axiom in practice it is mandatory to have access to the profile shapes. For example, a software fitting algorithm must know whether the g is one-storey or two-storey in order to determine whether its profiles are similar to d and it should therefore be assigned the same space as d or not. *(See figure 2.11)* The practicality of determining profile shapes is discussed in chapter 3.

Nevertheless, 'similar shapes' does not necessitate 'identical shapes' and, in practice, the type designer or letter fitter is required to establish the level of reasonable precision at which to work. Revisiting v and w, the two letters' outermost profiles share a basic construction and thus they are generally considered similar, but the sides of v are often drawn at a different angle than the sides of w. *(See figure 2.12)*

> ***Axiom L-2: Profile Reflection*** *— The space fitted to a profile is the same if the profile is reflected horizontally.*

This axiom states that the similarity principle applies identically on both sides of a form. For example, the left sidebearing of o and the right sidebearing of b should be equal. This is a corollary to the previous axiom, explicitly noting that the 'equivalent profiles should get equivalent space' principle is not dependent on the direction that the profile is facing. *(See figure 2.13)*

Although this is perhaps intuitive, clarifying that the reflective principle applies in Latin text is important because it may not necessarily hold for other writing systems. As was the case with the previous axiom, this axiom is usually framed as being self-evident.

> ***Axiom L-3: Inter-letter Area Equality*** *— The space between two letters in any letter-pair should appear optically equal to the space between the two letters in any other letter-pair.*

This axiom is cited by the vast majority of sources and it is the basis for a number of letter-fitting heuristic systems and automation tools. Its straightforwardness makes it simple to state and to illustrate visually. *(See figure 2.14)* But, unlike the preceding two axioms (which are also straightforward to state), this axiom is not generally framed as being self-evident, so many sources cite it as a first principle and invest some effort in explaining it or demonstrating it. Certainly it is a principle pinned to a writing system, rather than being a principle of 'visual balance' in the abstract. Although it applies to Latin, it would not easily be applicable to connected writing systems.

Figure 2.14
Illustrations of the principle of setting optically equal space between all pairs of letters in Latin text.

Left: Mengelt 1993, p. 36.

Right: Campe and Rausch 2022, p. 94.

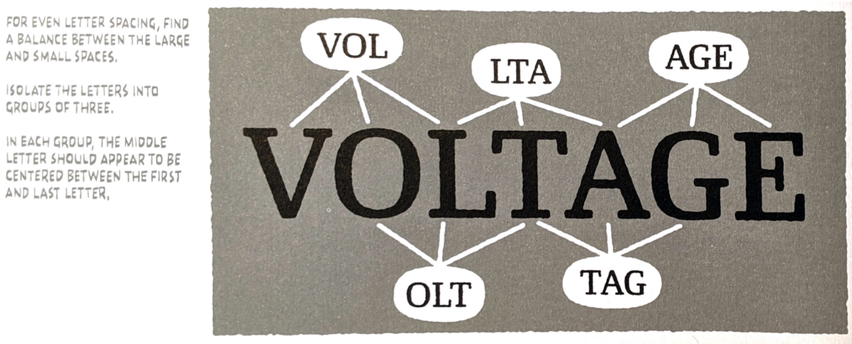


Figure 2.15
Letter fitting represented as the centring of letters in triplets (Highsmith 2020, p. 46).

Some statements of this axiom allow for exceptions that apply to certain letterforms or profiles but, even then, the principle is generally accepted to apply for any arbitrarily-chosen letter pairs.

> ***Axiom L-4: Triplet Centring*** *— When three letters are viewed in a sample triplet, the middle letter should appear to be centred between the two letters on either side.*

This axiom is related to the previous axiom, in the sense that the goal is to identify a fitting for (e.g.) abc wherein the space between ab appears optically equal to the space between bc. However, this axiom differs because it specifies placing three letters in a triplet, then assessing the total position of the middle letter. *(See figure 2.15)* Therefore, that process takes the width and the symmetry of the middle letter into account, which the previous axiom does not. For instance, a highly asymmetrical letterform like L should appear optically centred in the triplet HLH, even though a naive measurement of the space would find substantially more space between LH than between HL.

Note that this does not necessitate that *only* three letters be used in every test sequence. Longer sequences are testable and, in practice, often advised. Regardless of how long the test sequences are, though, the evaluation asked for by this axiom is about whether the letter appears centred between its immediate neighbours to each side.

This axiom strongly leverages the horizontal baseline layout in Latin text setting; it may not apply to scripts that incorporate positioning typeforms in two dimensions (such as subjoined forms or vowel, tone, and pronunciation marks). Those scripts may instead ask a more complicated question about a typeform appearing centred between its neighbours in more than one direction.

> ***Axiom L-5: Vertical Stroke Rhythm*** *— Vertical stems should appear optically to be in a consistent rhythm for any sample letter sequence.*

This axiom is a more specific distillation of the broad concept that pages of text (or other large blocks of text) should have an even ‘colour’ or ‘texture’ but, importantly, the axiom connects that texture to the vertical-stroke rhythm that underpins Latin text: the vertical stems of the letters in text should seem to be spaced in an optically even rhythm. This property of

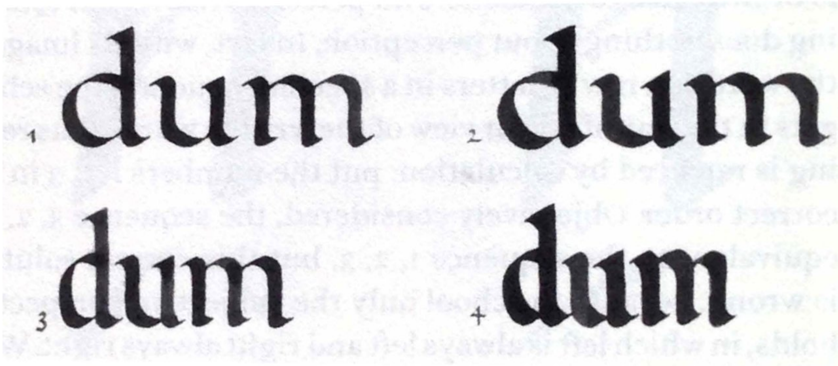


Figure 2.16 Gerrit Noordzij contended that stroke rhythm is the essential factor for letter fitting, because consistent rhythm defines the word-image (Noordzij 2009, 42).

Latin can be traced back to calligraphic and writing traditions for the lowercase letters. *(See figure 2.16)* Most formulations note that the rhythm should also appear observable in letterforms that do not feature stems. As will be discussed in chapter 3, that can prove to be a challenge to implement reliably.

The axiom is generally agreed upon for Latin text, though, as is the case with some other axioms, it should not be taken for granted in other writing systems. Gerry Leonidas has noted that the calligraphic tradition for Greek is based on looping forms, which results in different rhythmic structure (Leonidas 2018, p. 133). It is also important to note that the definition of *stem* is inherited from the Latin calligraphic tradition. A stem is a straight 'main' stroke in the letterform's skeleton; it is not simply any line with vertical orientation: vertical serifs and out-strokes found on s and z are not generally counted as stems.

Axiom L-6: Interior-Exterior Balance *— The interior counter of* n *and the interior counter of* H *should be the same optical size as the inter-letter area of an* nn *pair and an* HH *pair, respectively.*

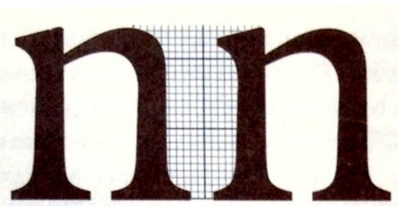


Figure 2.17
Smeijers calculated that the counter within each n is equal to the space between the pair by counting grid squares (Smeijers 1996, p. 31).

This axiom is the first on the list to connect the size of inter-letter area to the size of any intra-letter area. Generalizing, the principle is that the internal space enclosed within the letterforms should appear equal to the external space outside of the letterforms. For Latin text fitting, the more common framing of the axiom is to state the equality of the interior and exterior space for two key letterforms: n for lowercase and H for capitals *(See figure 2.17)*; choices which can act as a trivial test-case for the principle of equality.

There is some disagreement among the sources regarding the precision of the statement that the areas are optically equal. The earliest sources tend to state, simply, that the two areas should be equal, while more recent records tend to say that the inter-letter area should be slightly smaller than the internal area of the n or H. Perhaps that discrepancy should be attributed to changes in printing technology, or perhaps it represents a shift in reader expectations. But, either way, the relationship between the two areas is described as being a predetermined one for a particular typeface. Namely, whether the type designer determines that the inter-letter area should be 100% of the interior area or it should be 95% of the interior area, once that determination has been made, it remains fixed. This fixed relationship becomes important when applying axiom L-3: the area that is determined for the inter-letter area of an nn pair and an HH pair is the area that will be propagated to other inter-letter pairs.

The choice of n and H as the key letterforms is made because those letterforms have straight, vertical strokes on both side profiles and the letterforms conform to the 'standard' width for their case. Thus, assessing the ratio of inside-to-outside space is simpler. Choosing h and N would yield comparable results for most designs; some sources prefer o over n, others m. However, because letter-fitting is usually conducted in the context of designing a full complement of letters, there is little down side to using n and H as a convention.

Nevertheless, some prior work does state this same axiom in broader terms, contending, for example, that the inter-letter space of every double-letterform pair should be optically the same as the interior space of the pair. Although this broader framing of the axiom is testable for letterforms that enclose space (such as oo, ee, or even xx), it is less immediately clear how it should be examined for letterforms that do not enclose any interior space (such as ii or LL).

This axiom is particular to Latin text and, as with the previous axiom, is inherited from the general writing models used for Latin text throughout its history. Unlike the previous axiom, however, the claimed relationship between the interior counter size and the inter-letter space holds only for 'normal' weight, width, and text sizes.

> ***Axiom L-7: Concave Profile Truncation*** *— When a letter's counter is open on one side, only part of the counter's area should be measured as part of the total inter-letter space between that letter and the adjacent letters.*

This axiom also connects the size of inter-letter space to the size of an intra-letter space. When fitting a letter with a concave profile or 'open counter' (e.g., c or s), some percentage of the space bounded by the top, bottom, and closed side of the counter is treated as optically belonging to the interior of the letter and as not belonging to the inter-letter space. Fitting algorithms differ as to what the appropriate percentage to assign is, and as to how to compute it *(See figure 2.18)*, but the principle is almost universally agreed on.

Figure 2.18
Gerrit Noordzij illustrated the ambiguous boundary between interior and exterior space occurring in open-counter profiles by blending the colours (Noordzij 2000, p. 168).

The same principle is generally held to be true for letters with unbounded open counters (i.e., those forms that have two out of the 'top, bottom, and closed side' but lack the third), such as the right side profiles of r or F. It is usually acknowledged that different percentages apply for the unbounded open counters and that, for such strongly-asymmetric letterforms, avoiding a collision with the adjacent letter matters more than the percentage of open-counter measurement does.

> ***Axiom L-8: Fixed-Height Measurement*** *— The inter-letter area between two lowercase letters in a letter-pair should be measured between the baseline and the x-height. The inter-letter area between two capital letters should be measured between the baseline and the capital height.*

Figure 2.19
Smeijers noted that the area between the letters is optically bounded by somewhere within the grey regions at the x-height and baseline (Smeijers 1996, p. 31).

This axiom captures both a facet of conventional wisdom and a practical observation. The conventional wisdom is that, in lowercase text, the forms of letters between the baseline and the x-height include the majority of the variety distinguishing one letter from another, whereas extenders exhibit less variety and are more functional, distinguishing between certain otherwise-similar forms (e.g., o versus b, d, p and q) by position. That convention can be seen in phenomena like variable-length extenders as a font feature or the increase of x-height (without a corresponding increase in extender height) as weight and optical size vary.

The practical observation is that geometric measurements made to support the equal–inter-letter–counters axiom indicate that measurements between the baseline and x-height result in expected fitting, but measurements that include extenders do not. *(See figure 2.19)*

The framing used here, specifying the top and bottom measurement limits, applies to Latin lowercase letters fitted to other lowercase letters (the first statement) and capital letters fitted to other capital letters (the second statement). As discussed in § 2.1, the conventional wisdom is that capital letters should by default be fitted to lowercase letters (with capital-to-capital fitting incorporated as an alternate feature to be enabled for all-capital text). In the capital-to-lowercase context, the convention is that the extra height of capital letters be treated the same way as extenders, and the inter-letter areas of capital letters measured only between the baseline and the x-height.

The extent to which this axiom is specific to Latin is not clear. One factor contributing to the prominence of the baseline-to-x-height measurement zone may be that most of the ascenders in the basic Latin lowercase letterforms are near-identical, plain vertical stems: b, d, h, k, and l, with f often the sole outlier. In other bicameral scripts, the set of ascending forms can exhibit noticeably more variation above the x-height, such as β, δ, ζ, θ, λ, and ξ in Greek.[9] Even within Latin, the axiom may lose

9. In other scripts, the baseline limit may also be challenged, not just the x-height. The descending forms in Greek are similarly more diverse than Latin's. Gerard Unger used the gg combination, commonplace in Italian, to illustrate the importance of testing fitting across languages, pointing out that its adjacent descenders could affect page colour (Unger 2018, p. 125). He may have chosen that example because g, much like f, is an outlier among the Latin forms with descenders.

applicability with alphabets that include letterforms in addition to the basic-lowercase set, of which there are many (such as ð, ß, ƀ, or ɗ).

> ***Axiom L-9: Single-Stroke Supplement*** *— Single-stroke letters require more space than is fitted to a normal-width letter that has the same side-profile shape.*

Figure 2.20
The i on the upper line is fitted with more side-bearing space than the n on the line below (illustration by the author). Font shown: Tinos.

This axiom typically justifies adding additional space to single-stroke letterforms (such as i, j, or l) than to normal-width letters (such as n or h) of similar profile shape. It does so for practical reasons (e.g., to prevent the single-stroke letter from being missed while reading by merging with an adjacent letter, or to prevent sequences of several close vertical strokes from forming a darker-than-normal patch within the text). Consequently, this axiom might be regarded as practical advice, rather than as a principle that directly dictates intrinsically correct fitting. However, because this axiom is repeatedly cited as relating to the evenness of page colour, it does function as a first principle, rather than as an exception-handling measure. It is also notable that this axiom relates the space required to the total width of the letter, rather than to its side- profile alone. *(See figure 2.20)*

As used in this framing, *normal-width* refers to letterforms that enclose some internal space in a single counter.[10] Which forms are considered normal-width is Latin-specific, and not precisely defined. Only two letterforms out of the 26 lowercase base Latin letters enclose double-width counters in traditional upright designs: m and w. The proportion of single-stroke, single-counter, and double-counter letterforms in other writing systems is different: lowercase Cyrillic, for example, can have significantly more double-counter letterforms (ж, м, ф, ш, щ, ы, and ю in several East Slavic alphabets) but only one single-stroke letterform (і, in some East Slavic alphabets) or none, and may require a different definition of *normal-width* accordingly.

> ***Axiom L-10: Adjacent Extender Supplement*** *— Adjacent extenders require additional space.*

This axiom is a corollary to the previous axiom's advice about the practical problems of adjacent vertical strokes. Notably, the axiom deals with exception handling, advising that sequences of side-by-side extenders (e.g., lh, db, or gp in certain designs) should receive more inter-letter space than similar pairings with one or more x-height–profile letters. In digital fonts, such exception-handling is most often implemented as kerns, with the default fitting being devoted to the common, x-height case. This axiom's applicability to any given typeface design is also style-dependent.

10. See the definition of *counter* in the glossary.

11. See the historical discussion in § 2.1.3.

Axiom L-11: Collision Avoidance *— Letters should not touch or collide.*

This axiom establishes that there is a minimum allowable space between any two letters. Various formulations of the axiom do allow for exceptions for auxiliary components of letterforms to touch (such as serifs or in-strokes and out-strokes), but almost universally agree that stems, bowls, and other primary elements of the form's skeleton should never collide.

This is not a prohibition against kerning, but rather against typeforms overlapping or colliding in the design. Notable also is the fact that overlapping *designs* were not possible in cold-metal type (although, of course, overlapping letters on the page were possible through over-printing). This principle of barring collisions would be a requirement for legibility (e.g., the potential confusion of rn for m or VV for W) even if it was not found in fitting-related algorithms. *(See figure 2.21)*

Figure 2.21
Collisions can result in misidentified letterforms (Beier 2017, p. 148).

Axiom L-12: Diagonal Profile Limit *— The diagonal-profile letters have so much external space beneath the diagonals that they require zero or almost-zero additional sidebearing spaces.*

This axiom assumes the previous, collision-avoidance axiom, but further specifies that diagonal-profile letters (such as v or w) can be known to need zero or almost-zero sidebearings. Some sources explicitly connect this to the size of the unbounded space below the diagonal profiles.

How small the minimum sidebearings should be is less clearly stated. Notably, it is evident that the external space beneath the diagonals remains a measurement of two-dimensional area, even though the advised sidebearings are a linear distance.

Axiom L-13: Shells of Space *— Each letterform asserts some positive shell of space outward from its contours, which is required and should not be intruded upon by neighbouring letterforms.*

This axiom is the basis for *sector kerning*[11] and for several more recent implementations of letter-fitting automation tools. The term *shell* here is a choice by the author; sector-kerning systems often did not employ a term for the region of asserted space, and newer implementations of the idea do not have a standard term. In a certain sense, this axiom extends the earlier axiom prohibiting collisions but, in practice, the claims of this axiom are stronger and it is usually cited in reference to parts of a typeform that are not in danger of colliding with their neighbours. For example, the axiom might be cited as applying to vertical stems: e.g., the vertical stems require some amount of space of their own, even if the letterform has serifs and the serifs are the components that would collide with neighbouring letters long before the vertical stems would.

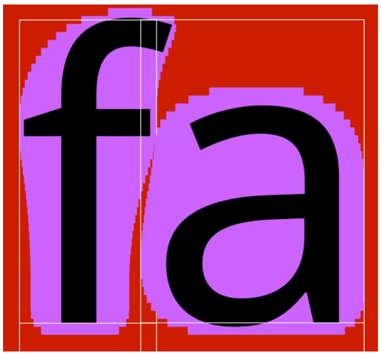


Figure 2.22
Toshi Omagari's BubbleKern works by manipulating shells around letterforms (Image from unpublished presentation slides in Omagari 2016, 33; used by permission).

In most formulations, the axiom also posits that the region of space asserted by a letterform is, collectively, a single and comparatively simple area that conforms to a rough expansion of the letterform's contours. In practical implementations, however, this axiom is rarely cited with any specific rules as to how the shell ought to be derived or how large it should be. *(See figure 2.22, over page)*

> ***Axiom L-14: Enclosure Avoidance*** — *Vertical overlaps between adjacent letters are permissible so long as one of the letters remains strictly above the other in the overlapping zone. (e.g.,* VA *or* Ta *can overlap, but* Cy *should not).*

This axiom notes that it is acceptable in a letter pair for the facing extreme points of the letters to overlap each other horizontally, but that such overlaps are not acceptable if they allow one letter to 'enclose' or wrap around the other. In Latin, preventing such enclosures is important for punctuation.

This axiom is not usually stated explicitly, but it is observed in warnings against overly-close kerning, in particular with typeforms that 'float' above the baseline, such as punctuation and mathematical operators.

> ***Axiom L-15: Upward Aperture Reduction*** — *Vertical open apertures that are open at the top (*u*,* AI*) should be smaller than vertical open apertures that are open at the bottom (*n*,* VI*).*

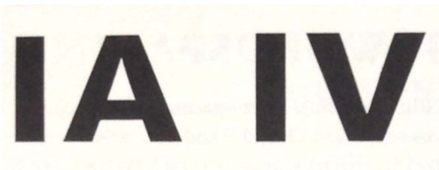


Figure 2.23
The IA pair is fitted closer than the IV pair because IA is open in the upward-facing direction (Hochuli 2015, p. 29).

This axiom deals specifically with the relative amount of space required when the inter-letter gap between some pair of letterforms creates a region that points either up or down. This region is called an *aperture* here to preserve clarity, but is sometimes referred to as an inter-letter counter (analogous to the counter in a typeform) because sources often connect this principle explicitly to the vertically-open counters in forms like u and n. The counter size of the single-letter case (u versus n) can be considered a letter-design issue, but linking that relationship to the two-letter case (AI versus VI) re-frames it as a broader principle.

This axiom applies only to a small set of letterforms for Latin text, due to the small number of side-profile permutations that can result in upward-facing vertical inter-letter apertures. The proportion would likely be different for other writing systems. *(See figure 2.23)*

> ***Axiom L-16: Diacritic Form Independence*** — *Diacritics or other marks are part of the letter to which they attach and, therefore, contribute to the profile shape and space requirements of the letter.*

In Latin text, diacritics are most often implemented as marks either above the x-height or below the baseline. Consequently, they are, by default, unaffected by several of the previous axioms. However, diacritics can be as wide or wider than the base letter they modify, in which case it becomes important for a fitting algorithm to include them when preventing collisions or performing other such steps.

Typeface design practice in Latin permits raising or lowering diacritics in most situations, including for collision avoidance. Moving diacritics horizontally to prevent collisions is somewhat less common, although this may be attributable to the risk that moving diacritics too far will result in misidentifying letters. Even when diacritics are 'floating' or otherwise discontiguous from the body of the letter, the placement of the diacritic is usually treated as an isolated problem, distinct from the problem of fitting the letters themselves to one another.

Note that this treatment of diacritics in Latin text differs from the treatment in other writing systems. For example, Arabic ijam, harakat, and other marks are often positioned separately, after the base *rasm* letters, in order to achieve the ideal word image. Polytonic Greek can involve diacritics that are placed to the side of capital letterforms, necessitating adjustments to the fitting.

2.4 Determining the domains and ranges of axioms

When considering the set of axioms as a whole, an obvious first question to pose is which axioms are the most important to successful letter fitting? But 'importance' can be a loaded term, and one that potentially leads to unreliable reasoning if it is simply left undefined. The order of the list is, roughly speaking, sorted with the most frequently cited axioms first — an order which perhaps reflects the axioms' relative importance, at least, to the writers and practitioners who discussed them. But this relationship between citation frequency and importance is not guaranteed; the citation order may also encompass an amalgamation of other criteria, such as authors choosing to discuss the simplest-to-understand axioms first in order to ease their readers into an unfamiliar subject matter.

A more practical way to define importance in the set of axioms would be to examine how and when the axioms apply and what letter-fitting results they produce when they are employed. That is, rather than determining importance based on the literature and the consensus of type designers, one could evaluate importance by scrutinizing how the axioms relate to the typeforms and the space determinations.

In software-engineering parlance, the typeforms constitute inputs to an axiom and the fitting constitutes an output. But that metaphor could prove overly constricting to serve as a starting point. For preliminary reasoning, to again borrow terminology from mathematics, one could say that each axiom has a domain over which it applies, and a range that delineates what possible results it can generate.

As discussed in § 2.1, the Latin text fitting axioms are limited in scope by writing system and typographic context, and furthermore most axioms carry an implicit assumption that they apply when used to fit a defined class of typeforms, such as predominantly lowercase continuous text, or all-capital text, or perhaps all-numeral data. Nonetheless, even when operating entirely within the writing system and typographic context and further focusing on the implicit typeform class, some of the fitting axioms are more or less restricted in domain than are others. Most often, a given axiom is restricted by which typeforms it is relevant to. Although several axioms claim near-universality, others demonstrably do not and, instead, are explicit about what forms they address. In other cases, axioms carry implicit restrictions on the typeface styles to which they apply.

As a matter of practical usage, these conditions and limitations do not invalidate the axioms; rather, the domains of each of the individual axioms are factors that type designers and algorithm authors must be aware of.

The simplest distinction seen in the ranges of the axioms is whether the axiom is defined to provide results for a desirable fitting outcome or to prohibit an undesirable outcome. It is also instructive to consider the nature of the result that an axiom provides. In the Latin axiom set, there are two primary groups to consider: axioms that provide answers about absolute, concrete values of space and axioms that provide answers about the size of one space relative to another space. Absolute and relative results both have value, but they are not interchangeable. These two techniques for partitioning the axioms according to range are certainly not the only options, but it will be seen in the following discussion that these partitions have practical value for constructing letter-fitting algorithms.

2.4.1 Domain: the set of typeforms or profiles addressed by an axiom

The first domain to be examined is the set of typeforms and profiles that a letter-fitting axiom addresses. This is a question that will eventually provide practical value when implementing a fitting algorithm, since establishing the correct domain helps ensure that the implementation does not omit any forms in the typeface and does not waste effort through duplication. More generally, as will be seen in chapter 4, knowing the domains of all of the axioms enables strategic reasoning about how best to cover all of the typeforms that comprise a typeface being fitted.

For the Latin text fitting axioms of § 2.3.2, eight out of the sixteen claim to apply universally. These eight universal axioms are:

- L-1: Profile Similarity
- L-2: Profile Reflection
- L-3: Inter-letter Area Equality
- L-4: Triplet Centring
- L-8: Fixed-Height Measurement
- L-11: Collision Avoidance
- L-13: Shells of Space
- L-14: Enclosure Avoidance

Note that the claim of universal applicability is explicit in axioms L-1, L-2, L-4, L-11, L-13, and L-14: each is framed as a universal principle. Axiom L-8: Fixed Height Measurement, however, poses a principle that claims to apply to all typeforms in a set, but what the fixed height *is* is allowed to vary depending on the forms in the chosen set. That is, when performing lowercase fitting, the space between all typeforms should be measured between the baseline and the x-height, but when performing capital fitting, the space between all typeforms should be measured between the baseline and the capital height. In both instances, the principle itself is still held to be uniformly true; the difference in measuring points is a property of the set of forms being considered.

Of the remaining eight axioms in the Latin text fitting set, seven axioms apply to a limited set of typeforms. These axioms are:

- L-5: Vertical Stroke Rhythm
- L-7: Concave Profile Truncation
- L-9: Single-Stroke Supplement
- L-10: Adjacent Extender Supplement
- L-12: Diagonal Profile Limit
- L-15: Upward Aperture Reduction
- L-16: Diacritic Form Independence

In each of these axioms, the limited applicability is a facet of the axiom itself: they are phrased as applying to particular profile shapes or constructions. One could perhaps argue that Axiom L-14: Enclosure Avoidance, which prohibits typeforms from enclosing one another, also applies to only a limited set of typeforms on the grounds that there are numerous pairs of typeforms that cannot enclose each other when placed in sequence: consider ii, for a trivial example in most typeface designs. However, for this analysis it was decided that the sole determining factor should be the framing of the axiom. If there ever were a typeface design for which the pair ii could be spaced so that one form was in danger of enclosing the other, the axiom would, it seems, call such an enclosure a fitting error.

The final group of axioms are those whose domain is limited to a specific or even isolated set of typeforms. For the Latin text fitting axioms, there is just one:

- L-6: Interior-Exterior Balance

which is typically stated as applying to n in the lowercase and H in the capitals. This group could certainly be regarded as a subset of the preceding, limited-applicability set; it would be tempting to lump the two groups together just to avoid the oddity of a group containing only one axiom. But, in the general approach, it must be remembered that other writing systems could have more axioms that specify particular forms.

Nevertheless, counting the exact number of typeforms addressed by the limited-applicability axioms is a pertinent question when developing a fitting algorithm. In the group listings above, the axioms were sorted by the approximate number of lowercase Latin letterforms that they apply to. In practice, the counting would need to take the constructions of each form into account for each specific typeface. Note, for example, that the different constructions for a and g affect which axioms are applicable. *(See figure 2.24, over page)*

2.4.2 Domain: the weight, width, slant, and optical sizes addressed by an axiom

The other question of domain considered is how the axioms are restricted in their applicability by the stylistic characteristics of the typeface. The historical study in § 2.2 revealed that most of the letter-fitting discussions for Latin text typefaces begins with a discussion of 'regular' weight and 'normal' width. As Unger pointed out, however, what constitutes 'normal' or 'regular' is not a simple question to answer; there are conventions to be found by measuring large assortments of typefaces, but the terms are only clear within the context of a particular design family (Unger 2007, p. 97).

There are multiple references indicating that Axiom L-6: Interior-Exterior Balance, which relates the inter-letter area to the interior space of the key letters n or H, applies only to regular weights and normal widths. Outside of that regular-and-normal zone, lighter-weight and wider designs receive more inter-letter space, while heavier-weight and more condensed designs receive less inter-letter space. But the relationship between the weight and the width of the key letters and the standard inter-letter space is not a simple mathematical formula.

Several sources discuss fitting slanted or italic styles, but they do not claim that different or additional principles apply, nor that the rules governing the fitting of upright forms do not apply.[12] [13]

Many sources assert that the optical size of text set in a document affects the necessary fitting. The smaller the optical size, the bigger the

12. The shapes of the profiles of the letterforms may differ between an upright and an italic style; see the discussion following Axiom L-1.

13. A detailed examination of the process for designing italic typeface styles and how it compares to the process for designing upright styles is found in Gaultney 2020, incorporating both analysis of typefaces and interviews with type designers. Concerns specific to letter fitting in italics are explored in p. 193–195, and the effects of narrowed italic typeforms on the resultant fitting in p. 155–159.

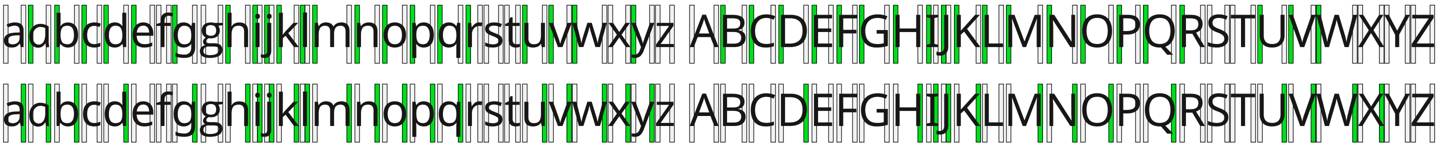
Gaultney reports that type designers interviewed cite the narrower and more complex shapes of italic letterforms, the differing proportion of curved and diagonal constructions within the overall set of typeforms (versus the set of upright forms), and asymmetry of serifs as factors that can make fitting italic designs more difficult than fitting upright designs. He further notes that the main strokes of italic forms tend to vary in angle from one form to another within a well-designed italic typeface. That variation adds visual character, but complicates the testing of letter fitting.

Nevertheless, Gaultney does reinforce the notion that a systematic approach to fitting is the norm in italic design as it is in upright, although it must also be observed that the majority of italic styles in contemporary typeface families are designed as secondary faces that are brought into harmony (fitting included) with a primary upright style.

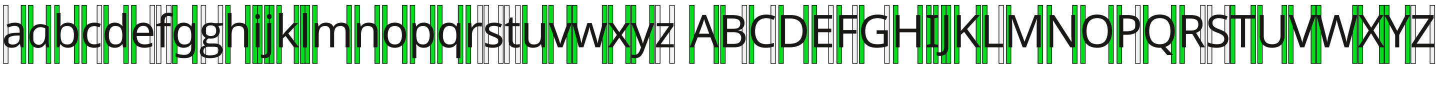
Figure 2.24

Highlighting the various side profiles of letterforms that are covered by different letter-fitting axioms. In these examples, both the one-storey and two-storey constructions of a and g are shown. A given typeface in the real world is likely to only include one construction or the other for each letter. Including both constructions is possible by adding alternate-forms features, but two constructions for a letter are not mixed-and-matched within a text: one or the other will be the default. Consequently, the set-of-profiles-addressed domain for an axiom can be dependent on the construction of the forms.

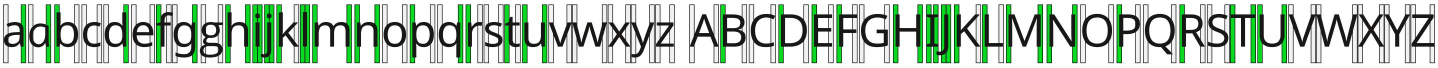
Some axioms (see page 52) claim to apply equally to all Latin letterforms, but in practice may only address a subset of the forms in a particular typeface. For example, Axiom L–1: Profile Similarity, is applicable only to profiles exhibiting similarity to some other profile. Highlighted are the sets of left profiles (above) and right profiles (below) that share similarity with at least one other profile. The left-profile and right-profile similarity groups are separated to avoid conflation with Axiom L–2's consideration of reflected similarity. Notably, the construction of forms matters: the left profile of two-storey a is not similar to any other letterform. Neither profile of two-storey g is similar to another profile.



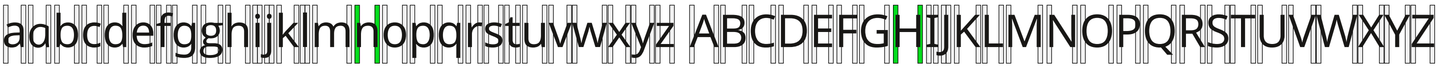
The related principle of Axiom L–2: Profile Reflection highlights a distinct, but still sizable, set of profiles.



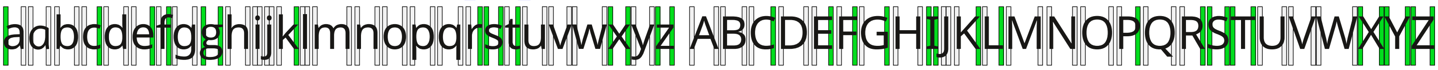
Axiom L–5: Vertical Stroke Rhythm addresses only profiles with vertical strokes, a large set in most Latin designs.



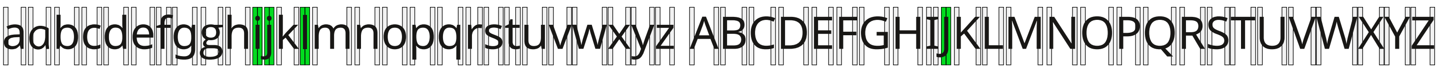
Axiom L–6: Interior-Exterior Balance, states a principle addressing only key letterforms: n for fitting lowercase Latin letters, and H for fitting Latin capitals to capitals.



Axiom L–7: Concave Profile Truncation affects open-counter profiles, which includes the two-storey a and g constructions, but not their one-storey equivalents. For forms such as k and R, opinion may vary as to whether each should be regarded as having an open counter.

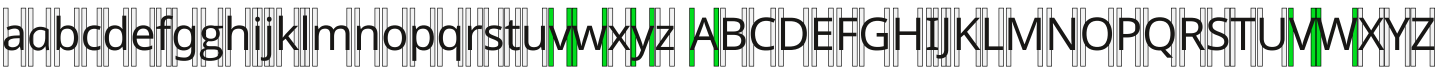


The single-stroke forms addressed by Axiom L–9: Single Stroke Supplement. Note that, in this example, the top and base strokes on I exclude it from the set. This is design-dependent; a J with a bowl resting on the baseline might not be considered a single-stroke form.

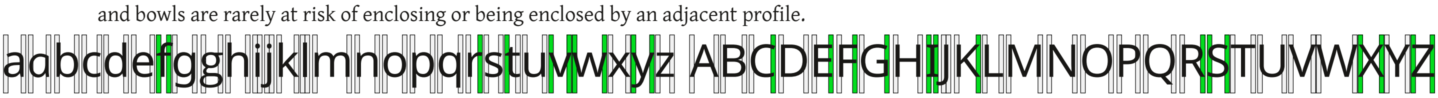


The forms addressed by Axiom L–10: Adjacent Extender Supplement, are also a small set. Notably, though, the axiom only affects these forms by adjusting their fitting when they are adjacent in a text, rather than providing a general fitting.

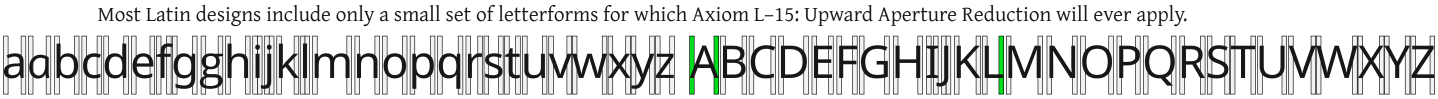
The 'minimal space' prescribed by Axiom L–12: Diagonal Profile Limit addresses a limited set of forms in upright designs.



It is difficult to unequivocally enumerate which forms Axiom L–14: Enclosure Avoidance may apply to; much depends on the specific design, but enclosure problems are likely only to occur where concave profiles, crossbars, and similar features are found. Straight stems and bowls are rarely at risk of enclosing or being enclosed by an adjacent profile.



Most Latin designs include only a small set of letterforms for which Axiom L–15: Upward Aperture Reduction will ever apply.



spaces between the forms must be, relative to the body size of the forms. This is generally stated to be a result stemming from readability. The consensus viewpoint from the historical study is that the 'smaller optical sizes require additional space' effect applies equally to all forms. Consequently, Axiom L-6, which posits a link between the interior space of the key letterforms n and H and the general inter-letter area, is again affected, because that axiom is framed as a relationship for letterforms of 'regular' and 'normal' proportions. The proportions of letterforms also differ between different optical sizes of the same typeface, with smaller optical sizes taking on wider interior spaces and smaller extenders.[14]

2.4.3 Range: comparing axioms by whether they prohibit a result or provide a result

In this research, the range of an axiom is defined as a characterization of the outcome that the axiom produces when it is applied to perform fitting. The most basic range to examine is whether the axiom provides a fitting result that can be used to determine the space for a typeform or it prohibits the conclusion of some undesirable fitting result.

In the Latin text fitting axiom set, there are two axioms that prohibit an undesirable fitting result:

- L-11: Collision Avoidance
- L-14: Enclosure Avoidance

The remaining axioms, as phrased, each provide a fitting result. That is, the axiom may not necessarily provide the final answer to the question *'how much space should be assigned to the sidebearings of this form?'* but it does provide information leading the type designer in the correct direction.

It is interesting to note that the prohibitions are both lower limits; i.e., do not fit forms too closely together, or a problem could arise. The rationales provided for the two prohibitions are essentially the same, that the overly tight fitting is undesirable because it could result in forms being misidentified by a reader. Because legibility and readability are basic requirements for text setting — more fundamental to successful fitting than aesthetics — this suggests that they should be considered of greater importance.

2.4.4 Range: comparing axioms by whether they concern relative space or absolute space

A more subtle distinction between the ranges of different axioms is found in whether the axioms are concerned with absolute measurements of space or relative measurements of space. 'Absolute' here does not necessarily mean a concrete numeric value is the result, although it could be. The distinction is that some of the Latin text fitting axioms, when applied to a letterform, make a statement about the specific amount of space belonging to that letterform (or, perhaps, to one of the profiles of

14. A fuller exploration of optical sizing differences is found in Ahrens et al., 2014. Fitting is discussed on p. 44; letterform proportions are examined in p. 32–43.

that letterform), while other axioms address only the relative relationship between sizes of space fitted to several letterforms or several profiles.

For example, Axiom L-1: Profile Similarity declares that similar side profiles require similar space. But, even if one accepts this as uncontested truth, the axiom does not prescribe what the size of this similar space should be — for any of the similar profiles.

Conversely, Axiom L-11: Collision Avoidance, declaring that letterforms must never touch or collide is addressing only a question of absolute space: it must be, at a minimum, zero, between any two letterforms.

In the Latin text fitting axioms, the seven 'relative space' axioms are:

- L-1: Profile Similarity
- L-2: Profile Reflection
- L-3: Inter-letter Area Equality
- L-7: Concave Profile Truncation
- L-9: Single-Stroke Supplement
- L-10: Adjacent Extender Supplement
- L-15: Upward Aperture Reduction

There are five 'absolute space' axioms found in the Latin text fitting axiom set. These are:

- L-6: Interior-Exterior Balance
- L-11: Collision Avoidance
- L-13: Shells of Space
- L-14: Enclosure Avoidance
- L-12: Diagonal Profile Limit

Ultimately, all letter-fitting areas must be resolved or transformed into an absolute space. As was seen in § 2.1.4, font formats in usage today store a typeface's fitting as numerical distances (both sidebearings and kerns).[15] Therefore, if a relative-space axiom exists for a given letterform or profile, the letterform or profile must somehow be linkable to an absolute-space axiom in order to make a usable contribution to the final fitting.

The remaining four axioms are either concerned with different principles or have a more nebulous relationship to the absolute/relative distinction. These are:

- L-4: Triplet Centring
- L-5: Vertical Stroke Rhythm
- L-8: Fixed-Height Measurement
- L-16: Diacritic Form Independence

15. It should be recalled that the numbers used in contemporary font formats are positions on a grid in font units, rather than any physical distance. Nevertheless, the term *absolute* in this discussion still indicates that applying the axiom produces a specific value, rather than stating a relative relationship between values.

Axiom L-8 and Axiom L-16 are not easily grouped into one category or the other. Axiom L-8: Fixed-Height Measurement is a condition about how measurements are made, but it is framed as being a condition that applies to both relative and to absolute measurements. Axiom L-16: Diacritic Form Independence is a statement about how letterforms are differentiated from other letterforms; when L-16 is invoked, the other axioms apply to â and ą exactly as they apply to a, with respect to both relative and absolute measurements.

Axiom L-4: Triplet Centring and Axiom L-5: Vertical Stroke Rhythm have perhaps more nuanced relationships to the absolute-or-relative demarcation. L-4 clearly makes a statement about two areas appearing equal, which is a relative measurement. But the full impact of L-4 would be lost if it was reduced in the analysis to being the same as axiom L-3's statement about equal inter-letter areas. L-4 explicitly addresses the optical centring of the typeform itself, in its positive-space, final rendered form. As was noted in the discussion of Axiom L-4, this centring is an optical phenomenon that is not simple to resolve.

Likewise, Axiom L-5 also concerns optical phenomena, although the phenomena it concerns are found at a larger scale: the rhythm observed in a text sequence of multiple forms. Here, as with Axiom L-4, the question is framed as one of relative space but, because it is a question that extends beyond measuring the space between two forms, the difference is important to retain, even if it is subtle.

Each of these ways of appraising the Latin text fitting axioms provides some additional meaning, but none of them is the whole story on its own. Figure 2–25 visualizes several factors together: estimating the universality domain of the axioms (from universal to specific) on the vertical axis, and sorting them by the relative-or-absolute range horizontally.

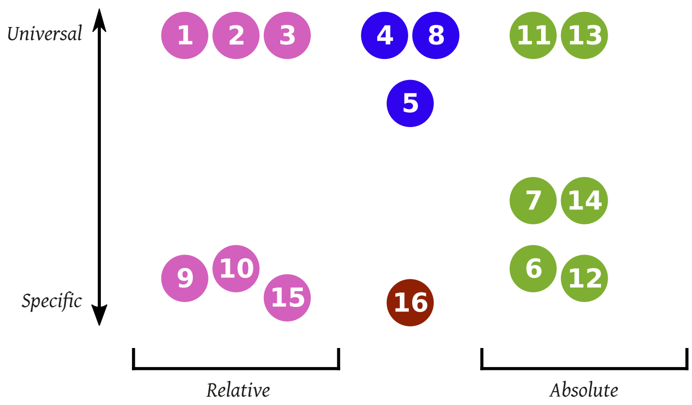


Figure 2.25
A visualization of the Latin text fitting axioms that takes several facets into account together. The colour-coding highlights that Axioms 4, 5, 8, and 16 do not easily map into statements of relative space or absolute space, but exhibit some nuance.

When developing an algorithm meant to fit a complete typeface, the end goal is to have absolute spaces determined for every typeform. Looking at how each of the axioms available addresses these needs makes it clearer why no one technique on its own has sufficed in prior projects.

2.5 Evaluating the interactions, dependencies, and redundancies between axioms

The discussion of domains and ranges focused on illuminating the relationships between each axiom and the various typeforms found in Latin text. A distinctly different perspective on the importance and behaviour of the axioms can be found by examining the interactions and interconnectivity between the axioms. In particular, exploring importance by how the constituent axioms work together can provide pragmatic suggestions for constructing a letter-fitting algorithm.

2.5.1 Axioms that are exceptions to other axioms

One of the clearest relationships that can be identified is when one axiom serves as an exception to another. Several of the relative-space axioms behave in this way. Axiom L-9: Single Stroke Supplement, Axiom L-10: Adjacent Extender Supplement, and Axiom L-15: Upward Aperture Reduction are direct statements of such a relationship. Each is expressed as defining a set of typeforms for which the necessary space is proportionally adjusted from the default inter-letter area, which is generically given as the default in Axiom L-3: Inter-letter Area Equality.

The concave-profile Axiom L-7: Concave Profile Reduction is also an exception to the equal-inter-letter area principle of Axiom L-3. As with the preceding exception examples, this axiom calls for a proportional adjustment to the default inter-letter area, but the exact degree of the adjustment is less clearly stated.

Axiom L-12: Diagonal Profile Limit might also be considered an exception to area-equality principle of axiom L-3. Here again, Axiom L–12 calls for a different area than would otherwise be found via the inter-letter-area principle, but Axiom L-12 calls for an absolute: minimal space.

The standard-and-exception relationship has practical usefulness because it suggests that the axioms are connected in such a way that they could be implemented together in an algorithm., or used as ‘simple mode’ and ‘complex mode’ alternate procedures.

2.5.2 Axioms that are prohibitions of a failure-condition

The prohibitive Axioms, L-11: Collision Avoidance and L-14: Enclosure Avoidance, were discussed in § 2.4.3. Both act, in a certain sense, as exceptions to other axioms. But the relationship is distinct from the standard-and-exception relationship discussed above. First, the prohibitive axioms are not linked to other, specific axioms; they apply generally to the full axiom set. That is, collisions between typeforms are wrong because collisions are a serious problem for readers; to that end, for Axiom L-11 it does not matter which of the other axioms might otherwise cause a collision: the collision of forms is prohibited regardless.

Second, the intent of the prohibitive axioms is explicit. Specifically, they set limits on what the other more general axioms should generate, and they do so for the stated purpose of avoiding a particular adverse condition — rather than setting limits in order to support a favourable condition. This distinction can be seen by examining rewording of the axioms. For example, although 'single-stroke forms should get additional space' could be rephrased as 'it is wrong to not have additional space around single-stroke forms', that inverted construction does not make the goal of the axiom clear.

Practically speaking, the prohibitive axioms are straightforward to implement as tests or as parameters to an algorithm.

2.5.3 Dependencies and redundancies between axioms

Possibly the most complex relationships between axioms are those found where two or more axioms address the same typeforms or, more generally, overlapping domains of typeforms. Some of the axioms provide fitting for the same letterforms and either act in agreement with each other or are expected to reinforce each other's outcomes.

This can be seen in the relationship between Axiom L-5: Vertical Stroke Rhythm when it is compared against Axioms L-1: Profile Similarity, L-2: Profile Reflection, L-3: Inter-letter Area Equality, and L-6: Interior-Exterior Balance, collectively. The relationships between Axioms L-1, L-2, L-3, and L-6 are quite close and mutually reinforcing. Axiom L-5 is distinct, saying that the vertical stems of forms should appear in a consistent rhythm. But that same consistency of vertical stems can also be arrived at by applying the similarity, equal inter-letter area, and key-letterform Axioms L-1, L-2, L-3, and L-6, in concert, to all of the forms that have vertical stems.

Conceptually, Axiom L-5: Vertical Stroke Rhythm is not a simple duplication of the collective application of L-1, L-2, L-3, and L-6, however, because the domains differ. Axiom L-5 has a smaller domain, given that it only applies to typeforms with vertical stems; Axioms L-1, L-2, L-3, and L-6, together, apply to typeforms with vertical stems and also to those forms with round profiles, diagonal profiles, or divided profiles. As will be seen in chapter 4, the practical job of constructing an algorithm must consider whether the overlapping interaction between Axiom L-5 and the collective effect of Axioms L-1, L-2, L-3, and L-6 is best handled by implementing both or by choosing one and dispensing with the other.

In other cases, the interaction between axioms can be more nebulous. Axiom L-7: Concave Profile Truncation states that only a portion of the area enclosed by a concave side profile should be considered interior space, and the rest should be considered exterior space. This means that Axiom L-7 interacts with all of the other axioms that make a statement about inter-letter areas: L-1, L-2, L-3, L-6, and L-8. Thus, an algorithm must establish how much of a concave-profile's enclosed area to count towards the inter-letter area, or else it cannot arrive at the inter-letter area values that are required to apply the principles of the other axioms when the algorithm fits forms with concave profiles.

The baseline-to-fixed-height measurement scheme of Axiom L-8: Fixed-Height Measurement also interacts with the axioms that measure inter-letter areas: L-1, L-2, L-3, L-6, and L-7, by virtue of dictating where the inter-letter areas should be measured. But its interactions with the Axioms concerning profile shapes and how they are grouped (L-1 and L-2) hide a subtle caveat. Axioms L-1 and L-2 address how similar profile shapes should be fitted, but what constitutes a 'similar' profile is in part determined by the top and bottom measurement lines. This is a caveat that triggers practical consequences for a fitting algorithm, because convention in Latin fonts says that the default sidebearings of capital letters should be determined by fitting them against lowercase letters (see § 2.1). Consequently, there are some capital letterforms, such as the right side of S, whose profile is a 'round' shape when measured from the baseline to the x-height, but is semi-open when measured from the baseline to the cap height. *(See figure 2.26)*

Figure 2.26
If cropped at the baseline and the x-height (green lines), the right-side profile shape of S is round. When considering the full profile from the baseline to the capital-height, however, the same S has a more complex right-side profile shape with an open side counter (illustration by the author). Font shown: Bodoni Moda Extra Bold.

Ultimately, the fullness of the interactions between the algorithms is more complex than can be listed in a simple table or mapped in a visualization. Such an approach to characterizing the axiom set was tried in this research, but the value of that attempt lies primarily in the mental exercise of considering the interactions, rather than in the production of a mapping that can prescribe the construction of an ideal fitting algorithm. One such example is shown in figure 2.27 below, which visualizes all of the inter-axiom interactions as simple edge connections. There is information missing in this visualization, because the different varieties of interaction are not distinguished from each other.

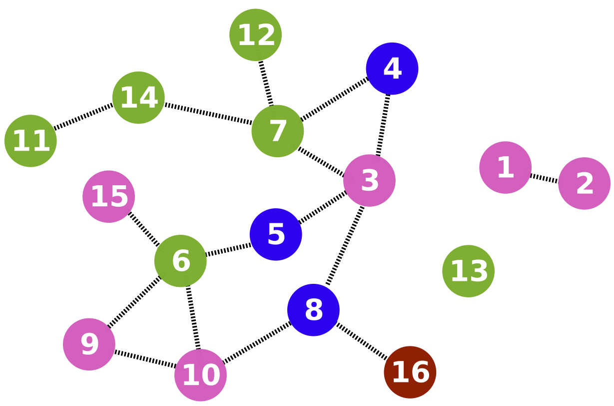


Figure 2.27
Visualizing the interconnections between the Latin text fitting axioms. The colour scheme used is the same as that for figure 24; green indicates absolute-space axioms; pink indicates relative-space axioms; blue and rust indicate the more nebulous relationships of those axioms.

Still, it is perhaps intriguing to note that Axiom L-13: Shells of Space is disconnected from the rest of the set, and it is likewise intriguing to note which axioms are the most connected to others: Axioms L-3: Inter-letter Area Equality, L-6: Interior-Exterior Balance, and L-7: Concave Profile Truncation. One could infer that addressing those axioms well is important, or that ignoring them is hazardous, but there are multiple ways one could traverse the graph even if some of the edge connections are removed. The set of axioms remains a model on which an algorithm can be developed, rather than a roadmap.

Any useful model for letter fitting in a particular writing system requires not only the fundamental axioms, but also an understanding of how the axioms operate as a set. Therefore it is important to understand how the axioms are interrelated, and how they function collectively, not just in isolation. In order to develop an algorithm to fit Latin text typefaces, the model and an understanding of its structure are required. The next step in that process is an exploration of how to move from the theoretical model into a practical implementation that represents the model faithfully while remaining pragmatic enough to be of use.

3. Practical implementation considerations for the model

Chapter 2 detailed the derivation of a theoretical model for the fitting of Latin text typefaces based on a set of interconnected, axiomatic principles that correspond to the processes that type designers employ when fitting typefaces manually. It concluded with a look at various properties and interconnections between the axioms in the model at a conceptual level.

Assuming that the model sufficiently represents successful letter-fitting practice, then the next step towards constructing a letter-fitting algorithm is to move from the conceptual to the practical, examining the axioms' potential implementability. Where the conceptual discussion dealt with domains, ranges, and relationships, a practical algorithm requires inputs, outputs, and various parameters.

In particular, the key questions to resolve are whether the axioms set out measurements and testable conditions that are clearly defined, whether they are efficient and can be replicated, and whether they offer coherent parameters for typeface designers to utilize when stylistic or aesthetic factors make such affordances desirable. For example, several axioms make statements about spaces or forms appearing optically equal; for implementation in an algorithm it must be determined if this optical equality maps directly to a concrete measurement of equality or relies on some judgment by the type designer that is more difficult to define.

In chapter 2 it was observed that most of the Latin text fitting axioms are functionally independent of one another. This permits consideration of each axiom in turn while constructing a fitting algorithm to follow the model. Each axiom can be assessed from the standpoint of how practical it is for implementation in software and how well the chosen axioms would operate together in a composite algorithm.

3.1 Cataloguing prior implementation work

As a starting point, the prior letter-fitting algorithm implementations covered in chapter two's historical survey were examined closely, to note where and how effectively they addressed the axioms of the model. Revisiting this prior work can be worthwhile even in cases where the prior work was considered unsuccessful at the time. Implementation efforts may have fallen by the wayside purely for business reasons or happen-stance; in others cases, several decades of evolving computer technology separate the original project from the present, and those changes in computing platforms might make algorithms that were impractical at the time workable on a contemporary system (see also chapter two, § 2.2.5).

Chapter 2 included a list of prior projects that involved the development of some form of fitting-automation algorithm. The table of implementations in chapter two, § 2.2.3 summarized the approximate publication date and creators of each implementation. That table can now be extended by cross-referencing which of the axioms from the Latin text fitting axiom set in chapter two, § 2.3 are implemented or rejected by each project.

When compiling this information, sometimes a simple 'yes or no' is not sufficient to capture the implementation's adoption of a particular axiom. The claims made by automation researchers and product vendors are often considerably broader than the axioms implemented in practice, and the promotional writing that accompanies an implementation frequently makes reference to letter-fitting theories or principles that do not form part of the implementation.

For example, both the promotional booklet for URW's *hz-program* and its original patent filing cite a number of fitting axioms (URW 1993, Karow 1992). But the implementation as it is described in that filing focuses solely on the equalisation of inter-letter areas (Axiom L–3), and draws upon Axiom L–8: Fixed-Height Measurement by computing the inter-letter areas between the baseline and x-height. To account for such cases, a separate mark was used to record the citation of an axiom that is not implemented.

Although a critical eye must always be trained on product-marketing copy, it is worth remembering that a discrepancy between the letter-fitting theories referenced in promotional material and the practical functionality found in the final implementation as shipped with the product should not be treated as an attempt at equivocation. For instance, URW might have appealed to the notion of stem rhythm in the *hz-program* brochure because the team believed that the stem-rhythm property would be a natural outcome of applying *kf*'s equal-inter-letter-area calculations.

Table 3.1 (over page) summarizes this cross-referencing of prior implementations against the Latin text fitting axioms. The table includes several works for which complete internal details are not available; these are coded in grey to distinguish them from the projects that could be analysed more fully. Several of these projects were unavailable for close examination because they were announced or advertised but were never published, others are reported to be in regular use but only as internal tools by their creators. For others, particularly among the earliest projects, it was simply not possible to locate a functioning copy or study one using contemporary software. They are included in the table to compare the claims they made, but in some cases the claims are all that remain.

The table also preserves projects that focus on writing systems other than Latin. Although the model at the centre of this research focuses on Latin, it was seen in chapter two that some fitting axioms can be shared across writing systems.

As in chapter 2, the table has deliberately avoided scrutinizing any projects currently covered by active patents; searching online patent databases does suggest that there may be some, but what those projects add to the overall picture remains unknown.

Table 3.1

Prior letter-fitting automation projects indexed by axioms addressed. A green disc indicates implementation; a dashed box indicates citation of the principle; a red x indicates rejection of the principle. Grey backgrounds denote incomplete publicly-available detail. The totals shown on the outer edges for rows and columns count implementations only, not citations.

Project	Year	L-1	L-2	L-3	L-4	L-5	L-6	L-7	L-8	L-9	L-10	L-11	L-12	L-13	L-14	L-15	L-16	
LOGOS	1970	☐	☐	☐	●	☐	☐	☐	●	☐		✖						2
Bell sector-spacing	1973													●				1
URW force experiment	1980																	0
Arabic Calligraphic Engine	1985																	0
Tracy method	1986	●	●	●	☐	☐	●	☐	☐			☐	●					5
Gerber	1986											☐		●				1
Type Designer auto feature	1989													●				1
Sector Kerning	1990											☐		●				1
hz-program kf	1991	☐	☐	●		☐	☐	●	●	☐		☐			●			4
Fontlab auto feature	1992																	0
Kernus	1993																	0
InfoKern	1993													●				1
Canon Shift	1995													●				1
Omron	1996													●				1
Neville	1998	☐		☐								●						1
TypeArt	1999	☐	☐					☐	☐			●		●	☐			2
FontForge auto feature	2000			●									●					2
AFDKO	2000			●					●			●						3
Aldine FFT	2003					●												1
KERNDICT	2003													●				1
sqtroff	2004																	0
KernMaster	2004																	0
Perturbation	2004											●		●				2
Sousa method	2005	●	●	●	☐	☐	●	☐	☐			☐	●					5
Wavelet masks	2006				☐	☐												0
Cambria OpenType Math cut-ins	2007											●		●	☐			2
iKern	2008			☐	☐	☐	☐	☐	☐			☐	☐		☐			0
LetterModeller	2009	●	●			●	●		●									5
Caslon Fourier analysis	2010			☐		●	☐		☐									1
Rhea	2010			●	☐	☐								☐				1
Rhea force experiment	2010																	0
SortsMill spacing by anchors	2010	●												●				2
Tsukurimashou	2011	●		☐								●			●			3
Autokern	2012			●					●			●			☐			3
Blur-masking	2012			☐										☐				0
Impallari macro	2012	●	●		☐	●	☐		☐				●					4
OpticalLetterSpacing.js	2012			●					☐									1
Typebutter	2012																	0
CJK Auto Spacing	2013																	0
Kernagic	2013	●		●		●			●									4
Monokern	2013																	0
font-prediction_mahout	2014																	0
Novi Sad statistical analysis	2014																	0
BubbleKern	2015	☐						☐				●	☐	●	☐			2
LS Cadencer	2015	●	●			●	●											4
Fittingroom	2015	●	✖			☐							☐					1
Black Spacer	2016			☐			☐	☐	☐									0
HT Letterspacer	2016			●				●	●									3
Spaceman	2016																	0
KernKraft	2016	●	●														☐	2
Electric kerning	2017			☐		☐												0
Octabox	2017	●	●									●		●				4
Machine Learning of Fonts	2017																	0
Atokern / kerncritic	2018			●														1
psoptkern	2018			●														1
KernBot	2018	●	●															2
HT Kerner	2019			●				●	●									3
electricbubble	2019			●		☐	☐							●				2
YinYangFit	2019		☐	☐	☐	☐	☐	☐										0
CounterSpace	2019			●				●				●				●		4
fontmetrics	2019		☐	☐	☐	☐	☐	☐										0
RhythmInfluencer	2020					●												1
type.tools AI	2020																	0
Andersson experiment	2020			●														1
Hands Face Space	2021	●		☐														1
Kern On	2021	☐		☐			☐					☐						0
Kerning	2021											●						1
Building a spacing calculator	2021	☐	☐		☐		☐											0
Kern Determiner	2022											●			☐			1
		13	8	16	1	7	4	4	8	0	0	12	4	15	2	1	0	

Finally, several projects are included in the list even though the project itself is not a tangible mechanism or a piece of computer software. This choice goes back to the definition of 'algorithm' used in this research, as defined in chapter 1. Walter Tracy's heuristic system is the key example here. Tracy recorded his method in *Letters of credit* as a didactic distillation of his professional experience. There can certainly be a grey area between plain written advice and a heuristic algorithm, but Tracy's explanation of his letter-fitting process is precise enough in its formulation of fitting to warrant considering it an algorithm. The decision to include Tracy's heuristic method is bolstered by the fact that other implementation projects cite it as the foundation on which their own, newer work builds — such as Sousa's heuristic.

Some interesting patterns are easy to observe in the table. Most projects implement only one or two axioms; the projects focused on developing a statistical model for fitting often implement none — perhaps assuming that useful patterns emerge from the statistics alone. Tracy's method cites and implements ten axioms in total (as does Sousa's method, derived from Tracy's), perhaps providing a clue as to its lasting popularity.

The axiom implemented most often is L–3: Inter-letter Area Equality, followed by Axiom L–13: Shells of Space and Axiom L–1: Profile Similarity. The Axiom L–13 implementations, however, were clustered mostly in the 1980s and 1990s, noticeably tapering off after 2000.

Implementation and citation counts are not the full story, of course. Some axioms may go uncited because they are assumed to be true and well-known rather than because they are ignored; this particularly applies to Axiom L–1 and Axiom L–2: Profile Reflection (which were noted in chapter 2 as being self-evident to some sources in the literature). Conversely, projects that attempt only to generate kerning lookups for a typeface might expect that the typeface being kerned already has 'correct' fitting in the default sidebearings for all its forms. Thus, the promotional literature and manuals for kerning-only projects may cite multiple axioms that they do not implement, on the assumption that the cited axioms will have already been employed to fit the sidebearings of the font, before the kerning process begins.

The table also does not capture when and how projects are historically connected. For example, Peter Karow and Margret Albrecht's work at URW reportedly continued from the force experiments through the *kf* module in URW's *hz-program* suite (Karow 1998); some of the same concepts found their way into URW's Kernus, which was later integrated into DTL's KernMaster (Espinoza et al. 2016). Later, *hz-program* was licensed to Adobe, and reportedly became the basis for 'Optical Spacing' features in several Adobe desktop products (Karow 2015). The AFDKO spacer from Adobe cites similar ideas, but it appears to be original code. Frank E. Blokland directly supported the development of Kernagic, DTL LetterModeller, and LS Cadencer (Blokland 2016; Schneider 2016), and indirectly inspired

Impallari's spacing macro (Impallari 2012), which explains their similar feature sets. The exchange of ideas is ongoing and often cyclical.

Some gaps in the table are notable. Perhaps curiously, the axiom that is the most frequently cited, L–5: Vertical Stroke Rhythm, has been implemented far less often than it has been cited. There are also three axioms (L–9: Single-Stroke Supplement, L–10: Adjacent Extender Supplement, and L–16: Diacritic Form Independence) that occur in the literature but are not explicitly implemented, and two (L–4: Triplet Centring and L–15: Upward Aperture Reduction) that have only been implemented once. This is most surprising for Axiom L–4 , because that axiom is universal in its claimed domain and is quite frequently referenced in the literature.

This assessment of the prior work provided practical starting points when moving forward with this research. For those axioms where implementations were available for examination, the implementations were studied as well as compared with each other on correctness and efficiency. For some axioms with few known implementations, patterns observed in the prior work revealed several categories of potential difficulty that would need to be addressed for any new algorithm. In the discussion that follows, the axioms are considered in three groups: the axioms for which an implementation is clear, the axioms which are difficult to implement because their formal definitions are lacking, and the axioms which are well-defined but have unresolved questions warranting further investigation.

3.2 Axioms with clear implementation and parameterization

Several of the Latin text fitting axioms are straightforward to implement. Here, the term *straightforward* does not mean to imply that an axiom is simple or can trivially be turned into a function and left to run with no user interaction. Instead, calling an axiom straightforward means that it is clear what measurements and decisions are involved, what form the results will take, and, if there are parameters that should be left for the typeface designer to choose, it is clear how the chosen parameters work. This set of axioms includes the axioms that specify spaces of equal inter-letter area:

- L–1: Profile Similarity
- L–2: Profile Reflection
- L–3: Inter-letter Area Equality

The framing of these axioms and their implementation in prior work shows that the areas they refer to can be calculated in the normal geometric sense — as opposed to being optical judgments without a straightforward definition. Thus, the areas involved can be calculated directly from within a font editor or a font file. Many recent projects compute the inter-letter areas directly from the Bézier vector contours;

older implementations often rasterized the vectors or were used on bitmap fonts, in which case counting the filled and unfilled pixels is similar but provides less resolution.

As was noted in the discussion of Axiom L–1 in chapter 2, the only lingering question is whether the algorithm can determine which profiles to consider *similar* without user intervention. There is little prior work to draw on concerning automatically classifying forms by similarity. David Březina explored modelling the visual coherence of forms as perceived by readers (Březina 2018), but did not explore profile shapes or classification by group explicitly. Sebastian Kosch's Fittingroom, a JavaScript fitting-automation project, undertook the grouping of typeforms with techniques often used in machine-learning classifiers (Kosch 2015). But the more common approach is to pre-define groups of letterforms based on the traditional constructions in Latin letters, and enable the user to make changes to the groupings only if the user wishes to use an atypical design. This approach is used in LetterModeller and LS Cadencer (Blokland 2016; Schneider 2015).

If one assumes there is a means available to determine which typeforms each axiom covers, then several of the remaining axioms that apply to a limited set of typeforms and are framed as exceptions to Axiom L–3 are also clear:

- L–9: Single-Stroke Supplement
- L–10: Adjacent Extender Supplement
- L–15: Upward Aperture Reduction

with the addition of a parameter. That is to say, it is not ambiguous how an algorithm would implement increasing or decreasing the space in these cases. The size of the various exceptions requires a concrete decision, which becomes the parameter tunable by the user.

Adding a user-tunable parameter also suffices to handle two other axioms:

- L–6: Interior-Exterior Balance
- L–8: Fixed-Height Measurement

which are both straightforward to implement once the parameters have been determined. Both of these axioms, notably, are universal in the typeforms that they address, so the tuning of the parameters may require some careful consideration by the user of the fitting algorithm: any adjustments to the parameters would potentially impact many typeforms.

The relationship between the internal space of n and H and the inter-letter area was discussed in chapter 2, where it was noted that the literature of manual fitting left some leeway regarding the precise ratio between the key form's internal counter and the inter-letter area. The same is, broadly speaking, true for the baseline-to-x-height measurement limits. For some typeface designs, it might make sense to allow the user to adjust one or the other by a few grid units to capture all of the details in the letterforms.

From the perspective of implementing a fitting algorithm, however, the more crucial factor is that the 'baseline to x-height' measurement zone applies to lowercase Latin letters and to the typeforms that are fitted to work with them. A real-world algorithm implementation must take that into account, permitting different measurement zones to be defined for capital-to-capital fitting as well as, perhaps, to numerals, punctuation, and symbols. Therefore the measurement-zone parameter should not be a single zone, but should instead be a set of potentially several such zones for differing sets of typeforms.

The parameter question is perhaps marginally more difficult for the diagonal-profile axiom:

- L–12: Diagonal Profile Limit

Here, too, the addition of a parameter is certainly required. What distinguishes Axiom L–12 from the previous, 'exception' axioms is that the axiom asks for an absolute-space measurement rather than a relative-space adjustment. Thus, the parameter the user chooses is something that the algorithm must test for, rather than just adding or subtracting to the sidebearings determined by the other axioms. This matters because the value of the 'minimal' space is absolute, but it is not defined by the axiom itself. In extremely heavy or bold weights, it is easy to find examples where manually-fitted typefaces use negative sidebearings for the diagonal-profile letters.

Adding a parameter would also make the two prohibitive axioms:

- L–11: Collision Avoidance
- L–14: Enclosure Avoidance

clear, by establishing tolerances for what the algorithm should consider 'colliding' and 'overlapping'. Zero might be a plausible tolerance in either case but, regardless of the chosen tolerance, these axioms can be more complicated to implement, because testing Bézier vectors for overlaps and collisions is more difficult than comparing measurements of areas. Even when comparing adjacent forms as vector outlines, collisions and overlaps can happen in any direction. Furthermore, when comparing rasterized adjacent forms, one must also grapple with pixel-alignment and anti-aliasing.

Finally, the axiom addressing diacritic forms:

- L–16: Diacritic Form Independence

is straightforward to implement because it deals with how typeforms are classified before any of the other axioms come into play. There are still some practical hurdles to consider, such as the various ways in which diacritic forms can be encoded in Unicode. But, by and large, it is not ambiguous to say whether or not a diacritic letterform should be considered a separate letter: if in doubt, an algorithm can treat it as a separate form without ill effect.

3.3 Axioms lacking theoretical details needed for implementation: rhythm and shells

Figure 3.1
The precise stem-to-stem distances, in font units, can be unequal, but the typeface may still be regarded as exhibiting vertical stroke rhythm (illustration by the author). Font shown: Brill Roman.

Figure 3.2
The red lines, present in the original, are evenly spaced, and can be seen to align at different horizontal positions on different stems. (Blokland 2016, p. 115)

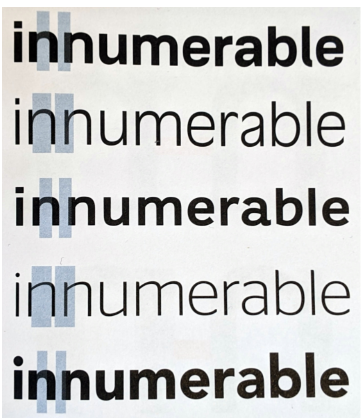


Figure 3.3
In sans-serif designs, vertical stroke rhythm may not be seen in a simple stem-to-stem measurement (Beier 2017, p. 131).

Of the four Latin text fitting axioms not discussed in the previous section, two were identified as possessing underlying issues of clarity or precision that would make their practical implementation in a fitting algorithm difficult. Namely, the vertical-stroke-rhythm axiom (L–5) has not been specified to precision that can be implemented in algorithmic terms, and the shells-of-space axiom (L–13) is missing a unified theory that describes how the shells themselves are devised — rendering the axiom an adequate way of describing spaces, but incomplete as an approach to determining spaces.

3.3.1 Analysis of the vertical-stroke-rhythm axiom

The vertical-stroke-rhythm axiom (L–5) ranks high among the most-cited general principles of Latin text fitting literature. As was noted in chapter 2, it is linked to the historical Latin calligraphic and handwriting tradition, a connection which perhaps makes it an alluring notion to appeal to.

It can be seen by direct measurement that, in well-designed manually-fitted typefaces, the exact stroke-to-stroke distances often vary in different letterforms, even when those letterforms share the same profile shapes. *(See figure 3.1)* This is a distinct difference from the equal-inter-letter area axioms discussed in the previous section, for which the statements that areas should 'appear equal' can be shown to translate into direct measurements.

The variance observed in stroke-to-stroke distances is not entirely a surprise, because each letterform is subject to the needs of other design goals and optical compensations. But it calls into question whether evenness in vertical stroke rhythm is contingent on some identifiable tolerance. Frank E Blokland, a strong advocate for vertical stroke rhythm, presents the illustration in figure 3.2 as evidence that Adrian Frutiger designed typefaces with a deliberately even stroke rhythm, but the strokes in the image vary in their alignment to the rhythm markers by close to a full stem width. If the acceptable tolerance for vertical stroke rhythm is on the order of the width of a stem, then perhaps vertical stroke rhythm is not precise enough for use in an algorithm.

But this variation cannot be merely a matter of matching the tolerance to the measurement precision, because the literature also supports the idea that rhythm is not a precise stroke-to-stroke measurement. Several sources note that sans-serif designs are typically fitted closer than serif designs, which means that the stroke-to-stroke distance within straight letters is different from the stroke-to-stroke distance between straight letters (Karow 1994, p. 182; Mengelt p. 38; Beier 2017, p. 131). *(See figure 3.3)*

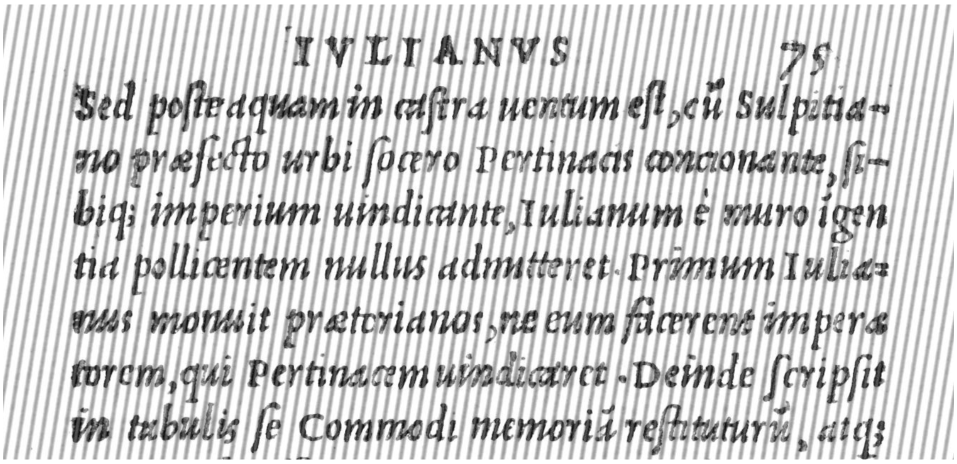


Figure 3.4
Egorov ran Fourier transforms on images of Aldine italic leaves, then visually identified frequencies that appeared to match stems of letterforms, connecting strokes, and other page features. The grey lines are Egorov's visualization of the frequency for stems, not the output of the Fourier transform itself (Egorov 2005). Used by permission.

In italic typefaces, it is further noted that the strokes themselves are typically set at different angles for different letterforms (Unger 2018, p. 134; Beier 2017, p. 165). Despite these acknowledged exceptions, vertical stem rhythm is still cited as an applicable design principle for those typeface styles. Some sources even advise that overly exact rhythm is aesthetically undesirable in a typeface.[1] Walter Käch wrote 'the rigid repetitive effect of absolutely equal characters cannot give to the whole appearance the liveliness of rhythmical undulations' (Käch, p. 13). The problem faced by a fitting algorithm is whether the 'liveliness of rhythmic undulation' can be prescribed precisely enough to be implemented in practice, or if the large tolerance and the number of exceptions permitted render it too abstract.

Analysis of visual rhythm at the page level has been conducted independently by Sergei Egorov and William Berkson, both using Fourier transforms, which convert the spatial patterns of the typeset pages into the frequency domain.[2] In both projects, the results revealed common rhythmic patterns across the pages examined: Egorov in leaves from Aldus Manutius' 1519 *Cassisus Dio* and Berkson in pages from William Caslon's 1766 specimen book (Egorov 2005; Berkson 2010). But neither project attempted to define that rhythmic structure; both were content to demonstrate it visually. *(See figure 3.4)* More to the point, neither project's analysis provides a method to begin with a rhythmic structure and use that structure to determine the sidebearings for letterforms.

Several prior fitting-automation works have implemented the stroke-rhythm concept as either the sole or primary method for determining fitting. The Impallari macro, DTL LetterModeller, and LS Cadencer implementations each start with a standard rhythm value that is said to provide sidebearings for the forms, but all three tools apply this standard rhythm value only to letters that are defined as having a straight profile on

1. Wilkins et al. 2007 found that overly exact rhythm can negatively impact reading, with words perceived as 'striped' resulting in poorer word recognition and reading speed.

2. Roger Watt also investigated techniques for using frequency-based analysis on pages of typeset text. Watt's investigation, however, focused on word space and line spacing (Sassoon et al. 1993, p. 178–201).

both sides.[3] For the remainder of the typeface, all three implementations provide tables of pre-determined numerical adjustments that are assigned to specific letters. Users have the ability to alter the tables of adjustments, but no guidance is supplied for how to choose appropriate values. *(See figure 3.5)*

The values of the pre-determined adjustments provided originated from Blokland's detailed analysis of Renaissance metal types. Blokland is open about stating that connection for DTL LetterModeller and has advised type designers that the numbers are bound to specific typeface styles and are not to be expected to work beyond the confines of the styles defined (Blokland 2019). For general implementation, though, an approach relying on manually pre-determined adjustments leaves algorithms without a starting point.

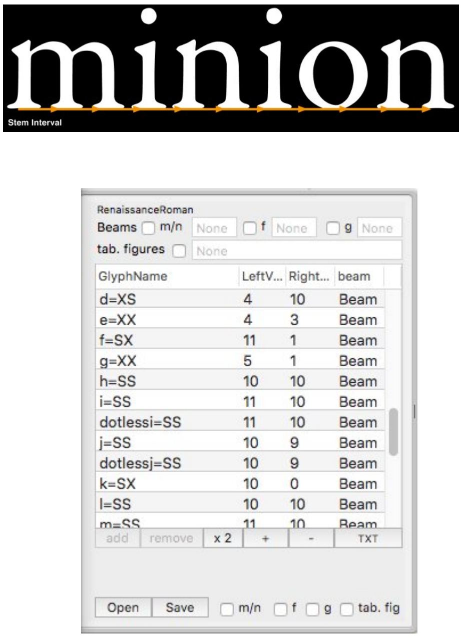


Figure 3.5
Above: LS Cadencer cites vertical stem rhythm as its fitting principle (RevolverType 2019).

Below: LS Cadencer provides per-letter adjustments in order to make the generic stem rhythm align with Renaissance types (screenshot by the author).

Moving away from pre-determined models, stem rhythm can be trivially measured for the letterforms and profiles that feature literal stems — that is, for straight profiles. For round profiles, an approximation to a stem can be made by adjusting for the overshoot (indeed, this adjustment is included in the pre-determined tables provided by the Impallari macro, DTL LetterModeller, and LS Cadencer). This simple approach to stem-approximation breaks down, however, in diagonal profiles, concave profiles, and profiles that are divided.

An alternate approach to applying the stem-rhythm axiom to letters without explicit stems is to calculate the 'ink projection' of each letterform, resulting in the shadow of the form, as it were, and to look for stems based on the peaks in the ink projection: the vertical slices of the form where the ink projection has the most ink ostensibly represent the closest approximation to an explicit stem. This technique has been implemented in Kernagic and in RhythmInfluencer. *(See figure 3.6)*

Practical problems are quickly evident with this approach. First, some diagonal and concave profiles may not produce peaks at all. This makes the ink-projection approach effectively invalid for these forms. Second, stems, by definition, have width — but peaks in the ink projection may not. Thus, to assign a 'virtual stem' to a peak in the ink projection, decisions must be made as to where to align the left and right boundaries of the virtual stem in relation to the peak. Third, there are several forms, particularly in serif

Figure 3.6
RhythmInfluencer determines virtual stems by calculating an ink projection of each letterform. Notably, the illustration omits several forms that produce projections difficult to turn into virtual stems (Renckens 2020).

3. The adjustment values are indeed specified by *letter*, not by the shape of the form. This is possibly because, as will be seen, the three implementations are each limited to particular typeface styles that follow historical type conventions for form and proportion.

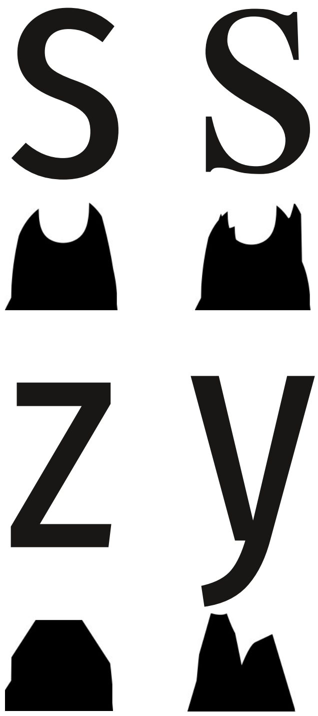


Figure 3.7
Ink projections for some forms and typeface styles may be significantly more difficult to convert into stems. The peaks in the top-left ink projection may be more easily identified with stem locations than the other three (screenshots by the author).

Perhaps notably, Renckens 2020 does not use these more difficult-to-interpret letterforms in illustrations. (Fonts shown: STIX, upper right; Fira Sans Extra Condensed, upper left and bottom left and right.)

designs, that can produce phantom peaks in the ink projection because the serifs, terminals, and other features cause interference. *(See figure 3.7)*

Another significant difficulty encountered when implementing this axiom is that widths of the letterforms are different in all but the simplest cases. Thus, the rhythm of the stems measured in a sample will have inherent variances because of the letterform widths, which in turn means that the sequence of letterforms used in a test sample to measure the rhythm will alter the result. The word minimum will exhibit one rhythm, adhesion a different rhythm: neither result is inherently more correct than the other.

For this research, techniques for calculating a general stem rhythm for a typeface based on the strokes of key letterforms (n or m) were investigated practically, because it offered a straightforward line of inquiry. However, the resulting fitting was inconsistent and far from the original manual fitting for all non-straight profiles, with gaps between some forms and collisions between others. The number of forms needing correction outnumbers the number of forms solved.

The nebulous nature of these issues does not disprove the importance of rhythm in typeface design or of Axiom L–5 in fitting. It does, however, demonstrate that visual rhythm is a complex and layered subject posing deep questions that have yet to be explored. As the axiom is typically stated, stroke rhythm alone is not sufficiently precise to calculate sidebearings. This helps explain why the Impallari macro, DTL LetterModeller, and LS Cadencer may have resorted to supplying predetermined adjustments for specific letterforms in specific styles, but makes the axiom problematic for implementing a fitting algorithm.

3.3.2 Analysis of the shells-of-space axiom

The shells-of-space axiom (L–13) exhibits a similar degree of ambiguity: like the previous axiom, it is believed to be true by many in the field of typeface design, but the way it is formulated in the literature and prior implementations omits details necessary to make it automatable in an algorithm. Specifically, there is not a formalized theory defining the dimensions of the shells or how the shapes of the shells adapt to particular constructions of typeforms. Without such a definition, shells are not something that an algorithm can create.

This may sound surprising, considering how many prior shells-of-space implementations exist. But a closer examination of those projects reveals that they largely relied on pre-determined shell construction, or were limited purely to detecting collisions rather than providing sidebearings.

The notion of shells-of-space that surround all typeforms was popular from the 1970s through the 1990s via the 'sector kerning' mechanisms in some typesetting software as discussed in chapter 2. But the sizes and geometries of the shells are rarely specified beyond the most basic prohibition that forms should not overlap and collide.

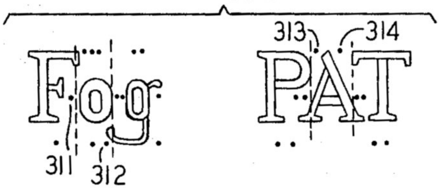


Figure 3.8
Bell Telephone's 1973 sector-kerning patent stored three sector-bounds per letterform, depicted in this image as dots (Mathews 1973, p. 2).

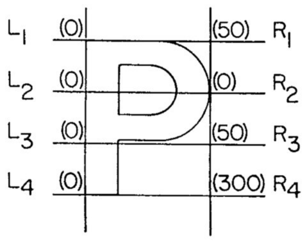


Figure 3.9
Gerber's 1986 sector-kerning patent stored four sector-bounds per letterform (Logan 1986, p. 6).

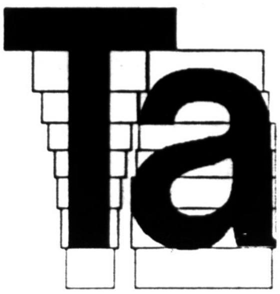


Figure 3.10
A sector-kerning example shown by Naiman. The width of the sectors for `T` are not a simple expansion-by-fixed-distance. Similarly, the sector completely fills in the aperture on the left side of `a` (Naiman 1985, p. 81).

The earliest patents relating to sector-kerning implementations in typesetting software perhaps provide clues as to why. Sector kerning implementations of the era were developed as an alternate method of storing fitting data for digital fonts at a time when computer memory and storage was relatively expensive (compared with current systems). Namely, storing a pairwise table of kerns for `N` typeforms would require N^2 table cells, but storing a table of sector-kerning values instead always requires far fewer cells: `2 × N × number-of-sectors` (that is, two profiles for each form, multiplied by the number of sectors).

To illustrate this difference, Bell Telephone's 1973 patent specified three sectors per form; Gerber's 1986 patent specified four. *(See figures 3.8 and 3.9)* In 1985, Naiman cited the popularity of sector kerning including the storage-and-memory-efficiency of the technique, using eight sectors per form as the example.[3] Even with eight sectors per form, a sector-kerning table is smaller than a pairwise kerning table for every font with more than 16 forms. A minimum of 52 forms are required for the Latin lowercase and capitals, of course; even at the time, a standard font could be anticipated to contain 100 forms or more (Naiman 1985, p. 77).

Storage and memory concerns aside, however, the early sector-kerning products provided no theory or guidance as to how a typeface designer or typographer should determine the values used to define the shell. In particular, what is missing from these implementations is some description for finding the optimal distance at each sector level; the widths of the sectors for each form are assumed to be either predetermined or to be established by the user. Ward Nicholson, in his detailed letter to *The Seybold Report*, offers a glimpse into what the process would have been like. A considerable time investment and manual effort was required of the typographer to evaluate and adjust the fonts, but once the sector-kerning tables were determined, they could be relied upon for any subsequent printing job (Nicholson 1990).

Tellingly, the patents indicate that the procedure for applying the sector kerns when typesetting the text was simply to shift the adjacent forms closer horizontally until the first contact between the shells occurred, which means that only a linear-distance separation was measured.

But it is clear from some of the illustrations that the shells were not a simple expansion of the profile shape by some fixed horizontal distance; open counters could be filled in and special treatment could be paid to diagonals. *(See figure 3.10)* So additional complexity was involved when the shells were defined, even though no theory was expressed as to how to construct the shells; it was assumed to be a task performed manually and judged by eye.

Eventually, more efficient data formats for storing kerning information were invented; instead of an `N`-by-`N` table with a cell for every possible pair (with many of those cells remaining empty because the pair required no

3. Regrettably, Naiman did not identify the vendor of the sector-kerning system referenced.

kerning), ‘short kerning’ tables were invented that only required storing a list of the pairs requiring a kerning adjustment (Karow 1994, p .389). That ‘short’ list could, hypothetically, be as long as the typeface designer wished to make it, but as a practical matter, it is always far smaller than the ‘long’ format. As that format took hold in the type industry, sector kerning fell out of favour in the commercial market. The contemporary `kern` and `GPOS` table formats described in chapter 2 are fundamentally the same as the short format: only those pairs needing a kern are stored.

In 2015, Toshi Omagari revisited sector kerning with his BubbleKern software utility. BubbleKern worked on the same core principle: given a shell of space defined for every typeform as input, the BubbleKern engine can calculate the kerning adjustments needed to shift any pair of adjacent forms closer together until the shells touched (Omagari 2015). Two key implementation details distinguish BubbleKern from the older sector-kerning tools. First, BubbleKern is a plug-in integrated into a font editor, which allows it to work directly with the Bézier contours at their native font-unit resolution, rather than the lower-resolution bitmaps of the earlier sector-kerning systems. *(See figure 3.11)* Thus BubbleKern can be used to generate shells with infinitely many ‘sectors’ (as they would have been termed in the sector-kerning era). The second is that BubbleKern outputs its results in a standard `GPOS` feature, rather than a specialized sector-kerning table. This means that fonts modified with BubbleKern should work equally well on any platform — unlike sector-kerning tables, which were typically only usable within the sector-kerning vendor’s own typesetting system.

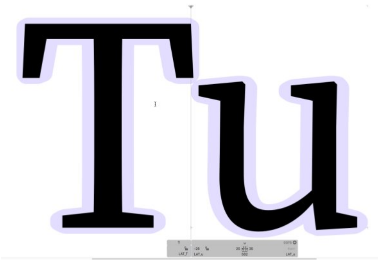


Figure 3.11
BubbleKern, running as a plug-in to the Glyphs font editor (screenshot by the author).

From the theoretical perspective, however, BubbleKern still does not posit a complete theory for automatically generating the shells of space that surround a typeform. BubbleKern includes a basic generator to create an initial shell by expanding from the form by a fixed distance, and there are hints in the BubbleKern user manual for typeface designers to improve on these initial shells by editing them manually. But creating the shells remains a manual responsibility left to the typeface designer.

This lack of theory describing the shells inhibits implementation in an algorithm. Specifically, if one considers the width of the shell: expanding the contours of a typeform by 10 units and expanding the contours of the same typeform by 25 units would both create equally valid shells, in the absence of some further conditions dictating what the necessary shell must look like. Without an answer to that fundamental question, a shell of any size greater than zero can certainly prevent collisions, but the shell model cannot be automated further.

Occasional attempts have been made to propose novel means for generating the shells. Independent type designer Peter Wiegel posited in 2012 that blurring images of letterforms would produce a shell-like field, although it is not clear that he implemented the concept (Wiegel 2012). *(See figure 3.12)* In 2004, a Canon patent suggested perturbing the outlines

Figure 3.12
Wiegel’s proposed method for automatically generating shells (Wiegel 2012).

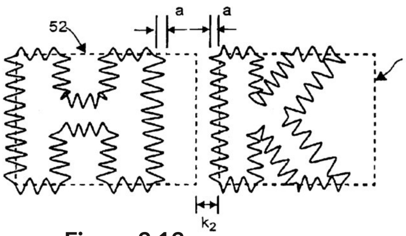


Figure 3.13
Canon's proposed method for generating shells by perturbing letterform outlines with sine waves (Browne et al. 2004, p. 5).

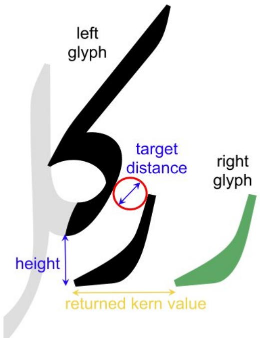


Figure 3.14
Kern Determiner marks a shell at fixed distance, but following the curves of the letterform (Cozens 2022).

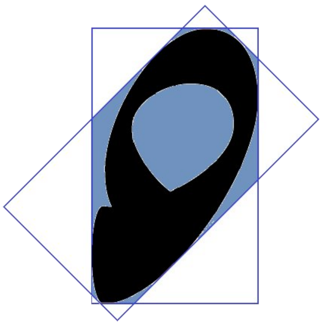


Figure 3.15
Octabox shells are automatically created by intersecting upright and angled bounding boxes (SIL 2017). Used by permission.

of the letterforms by replacing the smooth contours with sine waves and treating the resulting modified outline much like a shell. *(See figure 3.13)* Although different in their approaches, it is noteworthy that the final effect of both of these methods is simply to expand the typeform outward in every direction and call the resulting shape the shell.

Several more recent efforts to explore the notion of shells have focused on Arabic fonts, notably the Octabox method developed by Martin Hosken (SIL 2017) and Simon Cozens's Kern Determiner (Cozens 2022).

Both of these recent efforts focus solely on preventing collisions. Kern Determiner creates a shell by expanding the typeform outward by a fixed distance, although it does so in two dimensions rather than only expanding the form horizontally. *(See figure 3.14)* The Octabox method creates its shell by drawing a bounding box touching the form on all four sides, then intersecting that with a second bounding box drawn the same manner at 45-degree rotation. *(See figure 3.15)* These techniques are both more sophisticated than the simple fixed-distance horizontal expansion of sector kerning, but it is not clear to what degree either would work for non-Arabic text (Latin in particular), and neither has yet been widely put to the test. It must also be noted that both projects are limited in their range: they define a way to prevent collisions and overlaps (as per the prohibitive axioms, L–11: Collision Avoidance and L–14: Enclosure Avoidanec), but they do not otherwise generate fitting values.

The lack of consensus about what constitutes a good shell impedes the implementability of the shells-of-space axiom. Various approaches have been tried, but the result has never amounted to more than a solution to prevent collisions. The early implementations of sector kerning were motivated by a desire to more efficiently use computer memory and storage capacity, at the cost of requiring typographers to manually the shells. It may be that the shells-of-space axiom as a whole constitutes a different way of representing fitting data, or of thinking and talking about space, but does not offer significant insights into determining fitting.

3.4 Axioms presenting unresolved questions: triplet centring and open counters

The final two axioms in the Latin text fitting model prove challenging because they are potentially implementable, but lack a widely accepted definition for one or more key facets. In other words, the theories appear complete and well-formulated in their inputs, outputs, and parameters — in contrast to the axioms discussed in section 3.3 — but there are specific missing pieces.

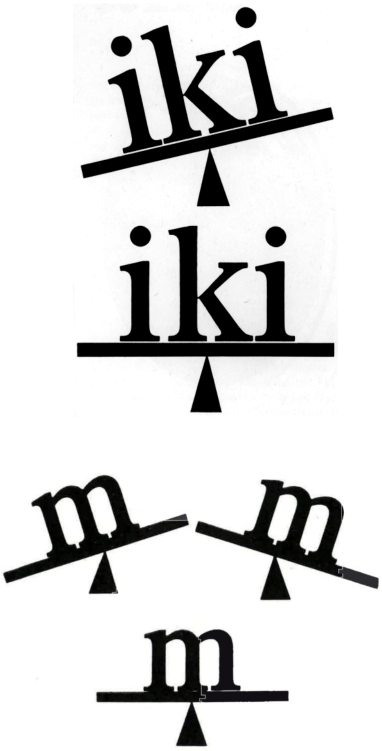


Figure 3.16
Above: an illustration of the centring of letterforms in triplets (Kindersley 1976 C, p. 16).

Below: Kindersley used the 'balance' metaphor to contend that finding the optical centrepoint of a letter in isolation is a precursor to centring it in a triplet (Kindersley 1962, p. 180).

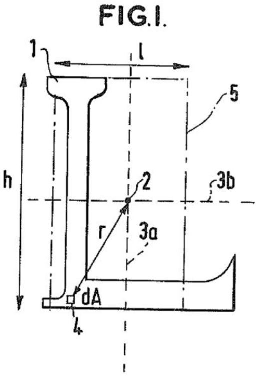


Figure 3.17
Illustration of the calculation of a mathematical moment for a given letterform. (Kindersley and Wiseman 1982, p. 3)

3.4.1 Analysis of the triplet-centring axiom

Axiom L–4: Triplet Centring states that the middle letterform in a triplet of letterforms should appear to be in the centre. It is clear from context in discussions of this axiom that the centring involved is not merely a direct measurement of equal linear distances nor of equal areas of space to the left and right, but instead refers to the form itself appearing centred between its neighbours.

Centring the form between its neighbours in the triplet, therefore, depends on identifying the optical centre of the form in addition to measuring the inter-letter areas. *(See figure 3.16)* This optical centrepoint, once it is identified, would be straightforward to place on the coordinate grid and use in measurements. But the definition of what that optical centrepoint is and how to determine it has not been established.

As mentioned in chapter 1, David Kindersley explored the problem of locating optical centrepoints in depth, beginning around 1961 and continuing in some fashion until close to his passing in 1995. The output of these investigations formed the core of LOGOS, a letter-fitting software product marketed to commercial type manufacturers and other corporate customers beginning in the early 1980s.

Kindersley first described the development of LOGOS in depth in *An essay in optical letter spacing and its mechanical application* (Kindersley 1966), a book that was updated in a 1976 edition (Kindersley 1976 C) with an additional chapter covering the intervening ten years, then updated once more in a 2001 edition that added a foreword by former LOGOS developer Francis Cave discussing the progression of the project during Cave's tenure in the 1980s (Kindersley et al. 2001).

The core of the LOGOS fitting method is detailed in a patent granted in 1982 to Kindersley and his business partner from Cambridge University Computing Laboratory, Neil Wiseman. The method starts by calculating the optical centrepoint of each form. The project investigated several possible formulae for that step; the patent describes what the project considered most successful: the point that divides the letterform so that either side produces the same value for a chosen mathematical moment.[4] *(See figure 3.17)* The patent lists two options for the mathematical moment: the second polar moment of area and the fourth polar moment of area; it also suggests that the letterform could optionally be scaled or partially masked out before the moment computations are made (Kindersley and Wiseman, 1982).

This concept of a point that splits the letterform into two halves that each produce the same result from the chosen function is an extension of one of Kindersley's earliest investigations: measuring the light levels transmitted by the shape of the letterform and locating the point at which

4. A *moment* in this usage is a mathematical quantity found by multiplying a static measurement of some sort by a distance. For a comparison of the different moments investigated by Kindersley and Wiseman, see chapter 4.

the left and right halves transmit the same amount of light. Kindersley observed that the light-transmission calculations were not providing a point that corresponded to his visual estimation of the forms' optical centrepoints. The search for a alternate mathematical formula that matched Kindersley's visual evaluations led to the options described in the patent, although Kindersley expressed in correspondence to Wiseman that the neither option enabled LOGOS to consistently produce acceptable results for certain forms, namely, the strongly asymmetric forms like L, P, F, and r (Kindersley 1976 A; 1976 B; 1977).

Crucially, however, the optical-centrepoint formula was just one component of LOGOS, and the other major component is less clearly supported by established letter-fitting theory. As discussed in chapter 2, the centring of letterforms in a triplet only addresses the relative space on either side of the letterform. To produce sidebearings for a form, the relative spaces must be linked somehow to an absolute space.

Making this link was the second step in the LOGOS fitting method. Kindersley's original approach postulated that all optical centrepoints should be spaced equidistant from each other horizontally, but it quickly became apparent that that approach does not work when mixing wide and narrow forms. Later versions of LOGOS dispensed with the equal-point-to-point-distance idea and instead posited that there is a 'characteristic rectangle' for each letter which would have the same mathematical moment as the letter. The width of the characteristic rectangle would then be measured, and LOGOS would assign it as the width of the letter, thus linking the relative space of the centrepoint to an absolute space.

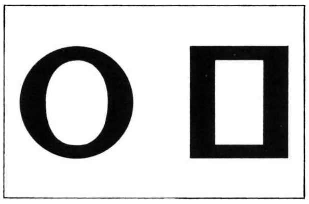


Figure 3.18
A lowercase o next to its characteristic rectangle in the LOGOS system (Kindersley 1976, p. 32).

The question of how to build the characteristic rectangles was a separate problem that occupied the project for some time; the LOGOS team eventually settled on compressing the letterforms in the vertical direction and constructing rectangles with side thicknesses that matched the vertical and horizontal strokes thicknesses of the typeface being fitted (Van Blokland 1986). *(See figure 3.18)* The formula for solving this characteristic-rectangle relation for its width, given the stroke thicknesses and the centrepoint of a letterform, is a complicated 5th-order polynomial equation in six variables that does not have any generalized solution. Thus, the LOGOS software could not implement a function to convert centrepoints directly into characteristic widths, and instead pre-computed the moments for a large set of possible characteristic widths and looked up the closest match for each letterform. *(See figure 3.19)*

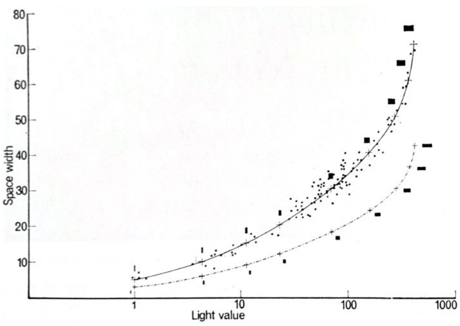


Figure 3.19
Scatterplot comparing the computed widths of letterforms against pre-computed standard values. (Kindersley 1966, p. 10).

This was certainly an optimization necessary for practical reasons (once the table had been precomputed, stored values could be looked up in constant time), and indeed the mathematics behind the 5th-order equation still have not been solved today. Regardless of the computational costs, though, the characteristic-rectangle component seems to have been a workaround imposed by the need to connect the centrepoint to a pair of absolute sidebearings, rather than a discovery connected by a theoretical model.

In the published account, Kindersley makes an atypical leap of logic by declaring that, once found, optical centrepoints alone are sufficient to fit letters. 'Suddenly it dawned on me that the finding of centres and the spaces were one and the same thing. Find the right centres and you will then have in light value terms, through the wedge, the correct space' (Kindersley 2001, p. 21).[5] Regrettably, unlike with the majority of his research, Kindersley did not record investigations to document this connection between optical centrepoints and widths. It is impossible to say whether further work would have successfully resolved the remaining issues in the LOGOS approach.

For this research project, though, the centrepoint-finding component from LOGOS was deemed relevant for further investigation; first because LOGOS is the sole known implementation of the triplet-centring axiom, and second because it has not been explored in isolation from the canonical-rectangle component. An unresolved question remains as to which, if any, of the formulae presented as options for finding optical centrepoints constitutes a formula suitable for determining centrepoints in the context of a different triplet-centring implementation.

3.4.2 Open counters and concave profiles

The final axiom in the Latin text fitting set is Axiom L–7: Concave Profile Truncation, which states that some of the area inside an open counter should be considered external space that contributes to the total inter-letter area between the profile and the adjacent typeform, while the rest of the area inside the open counter should be considered internal space. In the literature and the manual practices of letter fitting, this principle is widely accepted. But it presents two practical difficulties for implementing a fitting algorithm. First, there is no consensus on how much of the area in the open counter is external and how much is internal, nor where the boundary between the two is found. Second, there is not a consensus on how to distinguish which profile shapes are truly an open counter from those profile shapes that are merely marginally concave and should not be processed with the open counters.

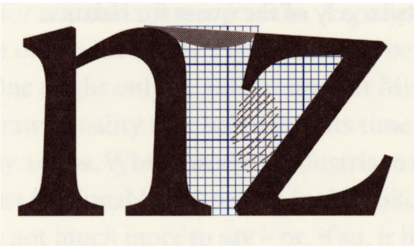


Figure 3.20
Smeijers' depiction of the "dual-function" nature of open counters (Smeijers 1996, p. 32)

Fred Smeijers referred to the space within open counters as a 'double-function area' and illustrated the vaguely defined boundary between internal and external space with a hand-shaded hatched region that deliberately does not show a fixed border. *(See figure 3.20)* But although Smeijers stated that the boundary region in open counters 'is certainly not objectively exact or constant', he added that he strongly suspects that serifs play an importantly role to readers by partially defining the boundary (Smeijers 1996, p. 32).

5. Kindersley used *wedge* to refer to the masking operation. The term originated from his earliest research with measuring light transmission for letterforms; in those experiments, the mask was a physical object presumably exhibiting some wedge-like characteristics..

Figure 3.21
Examples of the effects that choosing different clip-in angles can have on the area measured as inter-letter space in an open-counter form. *kf* treated the shaded regions as internal space (Karow et al. 1992, p. 7).

Figure 3.22
HT Letterspacer clips into open counters and cuts off the open-counter area at a user-specified fixed distance from the extreme edge of a profile. HT Letterspacer treats the green area as external space (HT Letterspacer 2016).

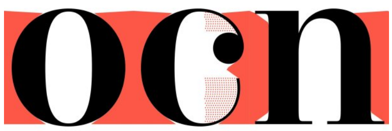


Figure 3.23
Black Spacer clips into open counters, cuts off open counters at a fixed distance, and measures some internal area reduced by a fractional multiplier (Hornus 2016 A).

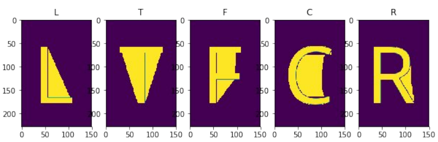


Figure 3.24
CounterSpace partially fills in an open counter with a shallower curve. CounterSpace treats the yellow area as internal space (Cozens 2019).

Three prior letter-fitting automation projects have implemented related techniques for handling the double-function regions in open counters: the *kf* module of *hz-program*, HT Letterspacer, and Black Spacer. The higher-level approach is the same for all three: reduce the value measured from the open counter shape, then use that reduced area in the standard inter-letter-area computations used for all forms. The projects differ in how they reduce the size of the open counter shape.

All three reduce the amount of space measured in open counters by clipping the counter shape inward at fixed angles from the top and bottom. HT Letterspacer clips in at 45 degrees by default (although this angle is tunable), while *kf* clips in at different angles for the above and below directions: by default, 42 degrees from above and 11.3 degrees from below (the shallower angle from below chosen to adhere closer to baseline serifs). *(See figures 3.21 and 3.22)* Black Spacer is only used internally at Black Foundry, so its clipping angle is not documented, but screenshots posted by creator Jérémie Hornus appear to clip in at roughly 30 degrees.

In addition to clipping the counter shape inward, HT Letterspacer and Black Spacer also cut off the counter shape at a pre-determined distance from the outer bound of the typeform. *(See figure 3.23, over page)* Black Spacer adds a third technique that calculates the space in 'hidden areas' and scales down the calculated hidden-area contribution by a fractional multiplier.[6]

Each of the three techniques: clipping into the open counter at an angle, cutting off the counter space at a chosen distance, and fractionally scaling the area measurement, serves to reduce the absolute amount of measured area in one way or another. But none of the three is specified on theoretical grounds: they work because they are capable of reducing the measured area, and the user is expected to tune the parameters (cut-in angle, cut-off distance, and scaling fraction) until the results match expectations.

A fourth project, CounterSpace, implements a distinctly different technique that reduces the area measured inside open counters by drawing a shallower counter boundary. CounterSpace's higher-level approach to fitting also differs, however. The area within the shallower substitute counter is not measured and directly incorporated into a calculation of inter-letter area. Instead, the entire substitute profile is used in a computation based on estimating the intensity of simulated lights between the letterforms. *(See figure 3.24)*

The second difficulty with implementing the open-counter axiom is, in a way, more fundamental: the determination of how concave a profile must be in order for it to be considered an open counter. In the literature, the list of forms exhibiting open counters is generally predictable. In the Latin lowercase, c, e, s, and z are almost always included; k and x may be

6. From the illustrations posted online, the 'hidden areas' appear to be area within the open counter but blocked by any protruding portion of the profile.

included as well, although those are sometimes grouped with the diagonal profiles. But lists such as these inevitably are pinned to historical tradition and dependent on the constructions of the letterforms: a is often included because the two-storey construction is more traditional; but it is omitted when the typeface uses the single-storey construction ɑ that demonstrably does not have an open counter. Although it suffices to tell a typeface designer how to approach open counters and rely on the designer to correctly identify which forms need to be included, requiring foreknowledge of the construction to know whether the fitting axiom is relevant inhibits defining an algorithm: the algorithm must know whether the typeface's 'a' glyph uses the a or ɑ form. The responsibility for determining the construction of the glyph could be offloaded to a human user, but determining it automatically is more difficult, raising the same issues as automatically classifying forms by profile shape (see § 3.2).

Figure 3.25

The left side of n is not usually considered an open-counter or concave profile, even if the serifs technically surround some space on three sides. The right side of t might be more readily considered concave, depending on the style (illustration by the author). Font shown: Gentium Plus.

It is possible to measure the curvature of the contours on the profile to decide if the profile is concave, but ambiguities can occur. For example, few readers or type designers would classify the left profile of n as exhibiting an open counter, even though the serifs on the top and bottom make the left profile concave in the mathematical sense. Considerably more ambiguous are letterforms like t which, in a wide construction, can plausibly form an open counter on the right profile. *(See figure 3.25)* The asymmetrical forms r, f, F, L, and P can also be difficult to classify, because there is not consensus on whether the unbounded areas above the horizontal beam of L or below the overhangs of the other forms should be considered open counters, or if those letterforms should instead be handled like the diagonal-profile forms. The inward-angled clipping technique employed by the *kf* module of *hz-program*, HT Letterspacer, and Black Spacer have the effect of treating these unbounded profiles like open counters, but it is not clear to what extent that is a deliberate choice or an accident. The *kf* patent filing, for example, specifically illustrates the use of a variety of clip-in angles for C and F, and the accompanying description states that different angles may be chosen depending on the form. *(See several of the options depicted in figure 3.21)*

In summary, the practical questions of a concave-profile-area implementation begin with how to decide whether or not a particular letterform should be subject to the concave-profile-area axiom. For a form classified as sufficiently concave, the remaining question is how to reduce the amount of area in the open counter that is counted toward inter-letter area. There are techniques to choose from, but the merits and trade-offs between the techniques could benefit by further investigation: how far into an open counter to measure, how to divide the open-counter area into interior and exterior portions, and what (if any) clip-in strategy to use.

3.5 Summarizing the practical considerations

The preceding analysis of the Latin text fitting axioms and of the prior fitting-algorithm implementations revealed several possible lines of inquiry worth practical investigation. The axioms that are clear in their definitions of the measurements, decisions, and parameters involved were determined in this research to be readily available for any implementation. The two axioms lacking theoretical detail — Axiom L–5: Vertical Stroke Rhythm and Axiom L–13: Shells of Space — were determined to be less suitable for implementation in a practical algorithm at this stage. The missing theoretical detail in these algorithms does not suggest that they are fundamentally untrue, but suggests that there is some ambiguity about the meaning of the underlying axiom itself.

In contrast, the triplet-centring axiom (L–4) and the open-counter-truncation axiom (L–7) are less ambiguous at the theoretical level, instead exhibiting unresolved questions that are more akin to implementation details. Furthermore, the open-counter and triplet-centring axioms could potentially provide fitting solutions for letterforms that are not easily addressed by the straightforward axioms. For example, the open-counter axiom would apply to several of the forms which Tracy left off of his heuristic model and said 'must be spaced visually': a, f, g, s, t, and z in the lowercase and S in the capitals (Tracy 2003, p. 71) and to profiles cited as problematic by Dwiggins: a, c, e, f, g, k, r, t, and s (Dwiggins 1940 B, p. 6). For his part, Kindersley focused considerable experimental time on the letterforms with concave profiles (C and L in particular) because they were not easy to fit.

Consequently, it was decided to investigate the possibility of finding solutions for the unresolved questions of axioms L–4 and L–7, and attempt to derive an algorithm for fitting Latin text that utilizes a composite of those axioms and the straightforward axioms.

4. Algorithm construction

Chapter 3 evaluated the axioms of the Latin text fitting model on practical grounds, with an eye towards constructing an algorithm useful for fitting Latin text typefaces. The evaluation revealed two axioms that warranted special investigation because they present unresolved implementation questions, but are rooted in well-known, unambiguous theories.

This chapter will first detail the investigation of those axioms and the development of practical implementations of the axioms. This will be followed by the construction of a testable letter-fitting algorithm that applies the new implementations for certain typeforms in composite fashion, relying on other axioms for other typeforms.

4.1 Preliminaries

Constructing a full implementation of a letter-fitting algorithm based on the axiomatic model involves implementation considerations that must be made up front, in addition to the practical decisions that must be made along the way.

First, it must be recalled that axioms L–4: Triplet Centring and L–7: Concave Profile Truncation were identified in chapter 3 as needing further investigation. But neither axiom alone nor the two axioms together are sufficient to fit a large enough set of typeforms to set real-world text. Specifically, neither of the axioms addresses absolute space: Axiom L–7 is explicitly concerned only with relative space, while Axiom L–4 addresses the relative spaces in a triplet, in conjunction with the positive form itself. All of those relative spaces must be transformed or otherwise linked to absolute spaces in order to output sidebearings. Furthermore, axiom L–4 applies only to those typeforms with open counters or concave side profiles, which is a minority of the Latin alphabet.

One or more other axioms, then, will require implementation in order to fit a typeface. It was noted in chapter 2 that multiple subsets of axioms may address the same forms. Several high-level strategies for selecting the axioms to implement are worth contemplating. An algorithm might attempt to implement every axiom in the model, which would introduce new questions of how to resolve discrepancies whenever two axioms output different sidebearings for a form. Alternatively, an algorithm might attempt to implement the fewest axioms necessary to fit all of the forms, which would entail selective judgment about which axioms to omit. Other strategies might fall somewhere in between these extremes.

For the sake of practicality, it was decided in this research project to pursue choosing a minimal set of axioms that can fit the typeforms of interest, and to prioritize the more-frequently-cited axioms and the axioms with straightforward implementation details when making the selection. There are, perhaps, typeface designers who employ a manual process closer to the 'use every axiom' strategy, but for this research a simpler algorithm is advantageous. First, it is more direct to implement, but second, a simpler configuration makes it easier to interpret test results of the algorithm as a whole and for individual typeforms.

The set of straightforward axioms listed in 3.2 includes axioms L–1: Profile Similarity, L–2: Profile Reflection, L–3: Inter-letter Area Equality, L–6: Interior-Exterior Balance, L–8: Fixed-Height Measurement, L–9: Single-Stroke Supplement, L–10: Adjacent Extender Supplement, L–11: Collision Avoidance, L–12: Diagonal Profile Limit, L–14: Enclosure Avoidance, L–15: Upward Aperture Reduction, and L–16: Diacritic Form Independence. Employed jointly, axioms L–1, L–2, L–3, and L–6 enable an algorithm to start with the internal space of a key letterform (n or H), use that internal space to establish a standard inter-letter area, and propagate that area to a significant subset of the other forms. Using this core set of axioms as a starting point, a composite algorithm was constructed by choosing additional axioms to implement until the full set of letterforms has been covered.

This strategy prioritized finding implementations for axioms L–4: Triplet Centring and L–7: Concave Profile Truncation, which were identified as potentially useful for forms with open counters or concave profiles, a class of forms that are not addressed by the core set of straightforward axioms. The following sections detail the investigations into axioms L–4 and L–7, and conclude with discussions of how they can be integrated into a single algorithm and the practical matter of setting suitable default values for tunable parameters.

4.2 Investigations of the LOGOS centre-point method

In chapter 3, it was noted that only one complete implementation has been identified for the triplet-centring axiom (L–4): the LOGOS project by David Kindersley and Neil Wiseman. However, the LOGOS method involved two steps: determining the optical centre-point for each form using a mathematical moment calculation, followed by associating the form with a characteristic rectangle of known width that had the same mathematical moment. This second component was necessary to link the relative-space information provided by the centre-point to a pair of absolute sidebearings. The centre-point of the form would be placed at the centre-point of the characteristic rectangle, and the left and right extents of the characteristic rectangle would be used to set the left and right sidebearings of the form.

One of the difficulties faced by the LOGOS project was establishing the choice of mathematical moments. The rationale provided for the choice of the 4th polar moment in the LOGOS product was practical, and based on an iterative development process. Kindersley evaluated the results of refitting typeforms via the two-part LOGOS method, then the team would iterate, updating the software based on whether the evaluation showed improvement or regression. The changes in each iteration could include alterations to the characteristic rectangle definition, to the masking scheme applied to the form, or to the choice of moment calculation.

For this research, it was determined that revisiting the centre-point-finding component on its own (without the characteristic-rectangle component) was worth investigating, in order to test whether the centre-points could be utilized in some different fashion. Of potential interest was determining if coupling the centre-point method with one or more of the other axioms in the Latin text fitting model could provide the link between the relative-space information and the absolute sidebearings.

To that end, the LOGOS centre-point algorithm was reimplemented in Python, then applied to the Bézier glyphs in a set of OpenType fonts. *(See Appendix A)* Thanks to the increase in processor speed in the years since the initial LOGOS project, it was possible to test several of the mathematical moment options described in the original LOGOS patent and surrounding documentation. The tested moments included the second polar moment of area and the fourth polar moment of area, as recommended in the patent, plus the first moment of area. The first moment of area serves as a useful comparison, both because it is referenced in Kindersley's writing and because it arose in Kindersley's earliest light-transmission tests.

To provide useful context in which to assess the results of the re-implementation test, a brief aside to look at the various moments follows below.

4.2.1 Moments compared

Loosely speaking, a moment can be defined as any measurable quantity multiplied by a distance; the precise definition varies between mathematics, physics, and engineering — distinctions which complicate the discussion of LOGOS, because the published material was often written by Kindersley with an audience of non-scientists in mind. Thus, the descriptions of moments in the texts fluctuates between the disciplines, referring at times to 'inertia' (an engineering concept), at other times to 'mass' or 'gravity' (a physics concept), and at still others simply to 'area'. Fortunately, the LOGOS patent is specific in the formulas required, and internal project correspondence is consistently more rigorous in its use of terminology.

In two-dimensional geometry (as was implemented in LOGOS to evaluate letterforms), the moments used are infinitesimal measurements, which are summed up over the entire shape with a double integral.

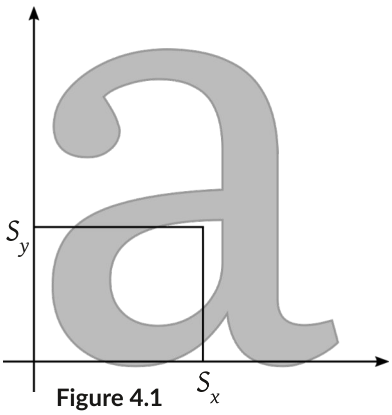


Figure 4.1
The x and y coordinates for the centroid of a form are found by calculating the first moment of area for x and y. The centroid is analogous to the centre of mass but notably, as in this example, may lie in an unfilled region (illustration by the author).

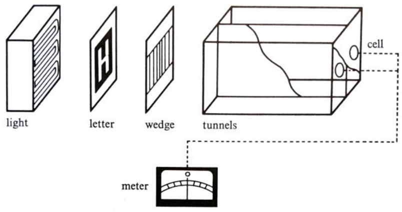


Figure 4.2
A diagram of the light-transmission measurement device used in Kindersley's early research, preceding the software-based approach of LOGOS (Kindersley 1973, p. 10).

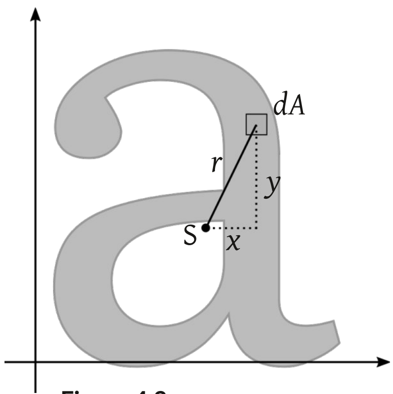


Figure 4.3
The polar moments of area are integrals over the entire form, based on the distance *r* measured from the centroid. In the second polar moment, the quantity integrated is r^2 and in the fourth polar moment, the quantity integrated is r^4. In both cases, the centroid of the form must be found first, adding computational complexity (illustration by the author).

See also figure 3.17 in chapter 3.

The first moment of area is the simplest, summing up merely the binary yes-or-no of whether the infinitesimal shape is empty or filled at each point. That results in the centre of area (or centroid) for the shape. *(See figure 4.1)* The *x* and *y* coordinates are given by the formulas:

$$S_x = \int_A y\, dA$$
$$S_y = \int_A x\, dA$$

If a letterform were cut from solid material of uniform depth and density, then the centroid would be the point upon which the form would balance. This notion of balanced areas has an intuitive appeal relating it to optical balance as discussed in the literature. More directly, the centroid was also the point found by Kindersley's mechanical measurements of light transmission: equal areas to the left and right transmit equal amounts of light to the left and right. *(See figure 4.2)*

Kindersley concluded from the light-transmission tests that the first moment of area did not suffice to determine the optical centre-points of forms, so the computational method in LOGOS was then switched to measuring the second polar moment of area. This moment sums up the infinitesimals in the shape multiplied by their distance (*r*) from the centroid, squared:

$$\iint_A r^2\, dA = \iint_A (x^2 + y^2)\, dA$$

When this also failed to produce satisfactory results, the project switched to the fourth polar moment of area, which sums up the infinitesimals in the shape multiplied by their distance (*r*) from the centroid, raised to the fourth power:

$$\iint_A r^4\, dA = \iint_A (x^2 + y^2)(x^2 + y^2)\, dA$$

In both moments, the distance *r* to the centroid can be re-expressed in *x* and *y* coordinates by the Pythagorean theorem, which permits them to be calculated for any typeform directly from the contour information stored in a font file. *(See figure 4.3)*

The resulting formulas are somewhat computationally expensive, but the method is complicated significantly by the fact that the polar moment (whichever is chosen) is calculated repeatedly. For each of *x* and *y*, separately, the method requires dividing the shape in two, calculating the moments of each half, comparing the results, then repeating that process with a new dividing line, recursively moving that line one way or the other, until the two halves return the same calculated value.

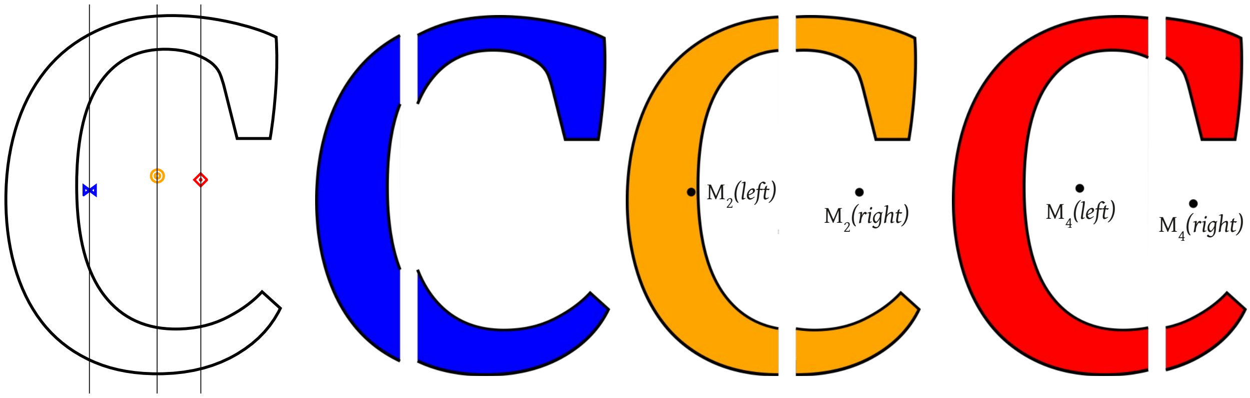


Figure 4.4
In the first image, the vertical lines through the centroid, M_2 point, and M_4 point of the letterform are shown (left to right). The second image shows the letterform split into two pieces at the centroid line: the resulting blue pieces have equal areas. The third image shows the letterform split into two pieces at the M_2 line: each of the resulting pieces computes to the same second-polar-moment-of-area value, when those second polar moments are each calculated with respect to the centroid *of that piece* (shown as black dots). The fourth images shows the same effect for the M_4 point and the fourth polar moments of area. The same properties would hold true for splitting the letterform horizontally through the centroid, M_2 point, and M_4 point (illustration by the author). Font shown: Yrsa.

Thus, the point returned by the LOGOS method is not, itself, a measurement of the second polar moment of area; rather, it is the point *(x,y)* for which

- *x* divides the typeform vertically, where the left portion of the typeform computes a second polar moment of area around its centroid that is equal to the second polar moment of area in the right portion of the typeform computed around its centroid
- *y* divides the typeform horizontally, where the top portion of the typeform computes a second polar moment of area around its centroid that is equal to the second polar moment of area in the bottom portion of the typeform computed around its centroid

Figure 4.4 highlights this. The higher moments have the practical effect of shifting the dividing-line on the typeform in question, but higher moments are increasingly indirect in what they measure.

In the published material, Kindersley refers to this second polar moment of area as representing the moment of inertia, which it is analogous to in structural or mechanical engineering. In this engineering sense, the second polar moment of area measures a physical object's resistance to a twisting force through its centre. So, by analogy, the M_2 point (as it will be called, for brevity, from here) could be said to represent the point around which each half of the typeform is equally resistant to twisting.

This resistance is, of course, purely an analogy, but typeface design regularly employs comparable analogies to other physical concepts: awkward letterforms may be called unbalanced, leaning forward or falling backward. So some analogous connection between the geometry of the shape and the physical world has value, up to a point. After all, type designers and readers alike know that gravity is not actually pulling letterforms toward the baseline; the asymmetric forms like P and r do not fall over, c does not roll onto its left side, and the dots above i and j remain suspended. The analogy appeals to everyday experiences readers know from the physical world, but it has its limits.

The fourth polar moment of area point (called M_4 from here) does not have a meaning with any clear analogue to physical forces as did the M_2 point, however. It can be stated that the M_4 point is more strongly influenced by how much of a shape's area is distributed towards its extremes rather than concentrated at the centre, but any connection to the physical world akin to gravity or inertia is obscure at best. As noted in chapter 3 § 3.4.1, Kindersley was not entirely convinced that the M_4 point was the ultimate solution and explored the possibility of adopting 'higher power' moments, at least for the more problematic letterforms.

This raises an important question for implementing the LOGOS centre-point component in a new context. If the primary rationale for choosing the M_4 point or a higher-moment point instead of the M_2 point was that the M_4 point produced more acceptable results when it was further applied to LOGOS's characteristic-rectangle component, then the M_4 point might not be the ideal choice when the characteristic-rectangle component is not used.

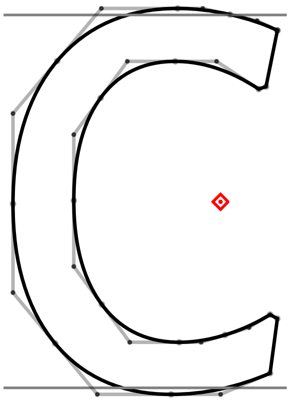


Figure 4.5
The M_4 point of c in Alegreya Sans. When decoupled from the characteristic-rectangle method, it is debatable whether the M_4 point still matches expectations for an optical centrepoint (illustration by the author).

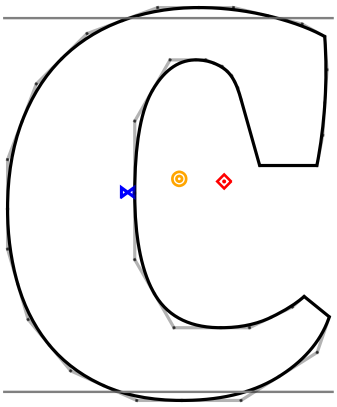


Figure 4.6
In heavier weights, the centroid (marked in blue) can be located in the interior region of the letterform even in open-counter forms, while the M_2 point (in gold) and M_4 point (in red) are consistently located in the open-counter region (illustration by the author). Font shown: Yrsa Bold.

4.2.2 Analysis of the re-implementation tests

Examining the results of testing the LOGOS centre-point reimplementation, it is apparent that the choice of mathematical moments is, indeed, pivotal. The M_4 point, preferred by LOGOS, is consistently closer to the outer boundary of the letterform in each open-counter typeform (e.g., closer to the right for c). This was expected based on Kindersley's account; as noted earlier, the M_4 point appeared to have been adopted in LOGOS precisely because Kindersley found the M_2 point consistently too far inward for open-counter and unbounded letterforms like c and L.

In lighter weights and sans-serif styles, particularly, the fourth polar moment was observed to often be quite close to the outer edge of the open counter. Mathematically, this is unsurprising, because the M_4 point is highly sensitive to forms where most of the shape is concentrated at the extremes and little or none of the shape is found at the centre.

This high sensitivity to what happens at the edge of open counters may have contributed to the LOGOS project's decision to start masking and scaling the letterforms before performing the moment calculations. Without the masking and scaling, it is hard to say that many type designers would consider the M_4 point in figure 4.5 the optical centre-point of the letter.

Another feature of the M_2 and M_4 points also stood out in the test results. If vertical lines are drawn through the M_2 and M_4 points, the lines were observed to be within the closed regions of the typeforms for almost all forms, but were observed to be consistently in the open regions for open-counter and concave-profile forms. Notably, this correlation was observed across all weights and styles tested. In contrast, the centroid was often located within the closed regions of open-counter and concave-profile forms in heavier weights and condensed styles. *(See figure 4.6, over page)*

Based on the consistency of this effect, it was conjectured that the M_2 or M4 line might prove useful as a practical test to use for classifying typeforms as open-counter forms. As was noted in chapter 3, an objective means for determining which typeforms should be classified as open-counter forms would be useful for algorithms implementing axiom L–7, because other forms (such as t) may be ambiguous.

The M_2 and M_4 points were compared to determine which would be more reliable for this purpose. In the tested typefaces, the M_4 point can be observed to be pulled out towards the extremes in symmetrical typeforms, while the M_2 point remained consistently closer to the centroid. This effect may be explained by the M_4 point's high sensitivity to wide apertures, coinciding with slight asymmetries in the construction of the letterforms. *(See figure 4.7)* Regardless of the cause, the effect contradicts intuition, which would predict a generally symmetrical form to have an optical centre-point near to the centroid. It was decided that this effect makes the M_4 point less reliable than the M_2 point for use in a rule to test for open-counter form classification.

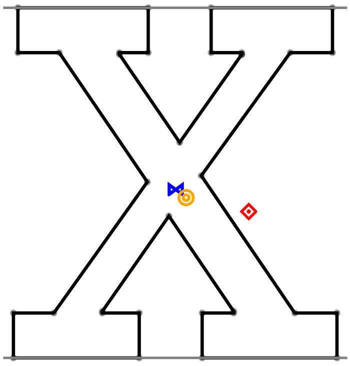


Figure 4.7
The centroid (marked in blue) and M_2 point (in gold) often appear close together in symmetrical letterforms, but the M_4 point (in red) can be pulled outward by even small optical compensations (illustration by the author). Font shown: Slabo 27px.

Stated more formally, the conjectured rule is:

> *If there is any horizontal beam drawn from the outside of the profile that intersects the vertical line through the M_2 point without first crossing a closed region, then the profile is considered an open-counter profile.*

When this rule was applied to the lowercase Latin letters of the tested fonts, the letterforms found in the standard lists of open-counter forms popular in the literature (a, c, e, k, s, x, and z) were consistently classified as featuring open-counter profiles, joined by two-storey constructions of g and, in certain designs, t. *(See figure 4.8)*

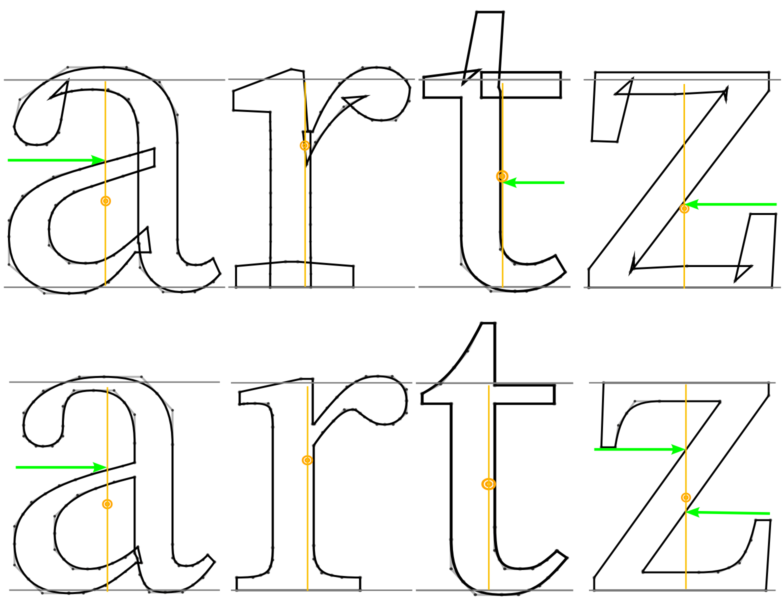


Figure 4.8
Using the M_2 line to classify profiles as open-counter profiles. Green arrows indicate where a horizontal beam from the outside intersects the M_2 line, meeting the test to classify the profile as an open-counter profile. Note that t on the top row meets the condition (albeit barely), whereas the t on the bottom row does not. Also note that on the top row, the left profile of z on does not meet the test, but the right profile does (illustration by the author). Fonts shown: top row: Source Serif 4; bottom row: STIX Two Text.

The results for the capital Latin letters were comparable: when the M_2 point was calculated for the full height of the letters, commonly-cited letterforms (C, E, G, S, and Z) were consistently classified as featuring open-counter profiles, joined by K in certain designs. There was more variety observed in the results for calculating the M_2 points of the capital letters between the baseline and the x-height (as the literature suggests might be preferable for fitting capital letters to lowercase letters). For example, in heavier weights or designs with prominent serifs, E was sometimes not classified as an open-counter form by the rule.

Although this is not conclusive evidence that the classification rule works, it was considered plausible enough to warrant testing in conjunction with a fitting algorithm. In particular, it was hypothesized that the rule might be useful for an algorithm to automatically determine if forms with common alternate constructions (such as a and g) should be classified as open forms without requiring user intervention, as well as to provide similar classifications for symbols and other typeforms.

4.3 Investigations of open-counter measurements

Implementing the above rule to test for open-counter forms in a fitting algorithm establishes a link between Axiom L–4: Triplet Centring and Axiom L–7: Concave Profile Truncation. Further exploration of the link suggested that potentially other interesting results could follow. In chapter 3, it was noted that the unresolved questions for implementing Axiom L–7 were how to classify forms as open-counter forms and how to appropriately reduce the measured area of the open counter such that part of its area is counted as internal space and part of its area is counted as external space.

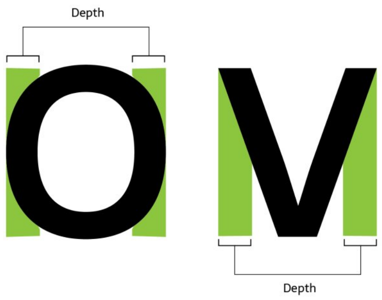


Figure 4.9
HT Letterspacer measures all inter-letter areas from the edge of the form in to a fixed distance; this includes measurement in open counters (HT Letterspacer 2016).

One of the techniques used by prior fitting-automation implementations to reduce the measured area of the internal counter is cutting off the measured area at some chosen distance from the outside edge. HT Letterspacer provides a fixed-distance parameter for this technique, measuring all letterforms from their extrema in to the same distance. *(See figure 4.9)* Published images showing Black Spacer’s measurements of open counters appear to show a similar technique, with all open-counter forms measured inward to the same distance.[1]

For HT Letterspacer, the use of a fixed-distance measurement into the open counter is an acknowledged limitation. A feature request to enable changing the measurement distance on a per-glyph basis was made on GitHub in 2017 and later confirmed by HT Letterspacer’s lead developer Andrés Torresi (Waxweiler and Torresi, 2018), although a per-glyph distance has not yet been implemented in the program. In the feature request, Nikolaus Waxweiler said that the fixed-distance parameter produced unacceptable results for l designs that feature a ‘tail’ or out-stroke on the right, and Torresi replied that allowing per-glyph

1. See chapter 3, figure 3.23.

measurement distance would also improve results for symbols. The brief discussion of the feature request does not concern how to determine the appropriate distance for a given form.

As implemented in HT Letterspacer, the fixed measurement distance is defined in proportion to the x-height of the font. This likely explains why Waxweiler highlighted l, a single-stroke form, as problematic: with the fixed measurement distance defined in proportion to the x-height, a value selected to be ideal for average-width typeforms could be too large for narrower forms or too small for double-width forms. But the measurement distance has the greatest impact on open-counter and concave-profile forms, so a sub-optimal choice for the measurement distance might not be noticed in the majority of letterforms.

In light of the link established between the L–4 and L–7 axioms for classifying typeforms as open-counter forms, it was convenient to also investigate whether the same link would provide insight into the question of choosing an optimal measurement distance. If there is a natural boundary within an open counter between the internal and external space, then it would be reasonable to expect that boundary to be related to the optical centre of the form. In other words, wherever the true optical centre of the open-counter form lies (and however that true optical centre might be defined), the area further behind the optical centre must be more internal to the form than the area outside the optical centre.

In the above rule, the vertical line through the M_2 point functions as the determiner for classifying a typeform as an open-counter form, and does so on the theoretical basis that the M_2 point represents the optical centre-point of the form — or, at the very least, an optical centre-point. If the M_2 point is accepted to represent the optical centre-point, then it is worth investigating whether the M_2 point is also relevant to measurement depth for open counters.

4.3.1 Analysis of the open-counter measurement tests

Consequently, experiments were conducted with a modified build of HT Letterspacer, using the x position of the M_2 point as the measurement depth for open-counter forms. The preliminary experiment used the M_2 line as the open-counter measurement distance and counted the full area up to the measurement distance as exterior space, applying the equal-inter-letter-areas axiom to determine the sidebearing for the open-counter profile just as HT Letterspacer does for all other profile shapes. The results of those experiments consistently moved the sidebearing generated for the open-counter profile too far inward: significantly further inward than the sidebearings in manually-fitted typefaces, and often into negative numbers.

That pattern suggested that measuring the entire area to the M_2 line and counting it all as exterior space was insufficient. To account for this, subsequent experiments coupled measuring in to the M_2 line with the

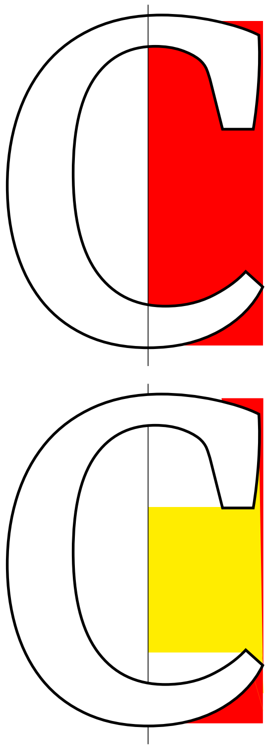


Figure 4.10

Above: measuring the area inside the full open counter to the M_2 line and considering it all external space resulted in overly close sidebearings for open-counter profiles.

Below: clipping inward at 90 degrees, and scaling the measured area bounded by the open-counter (shown in green) down by 50% resulted in more reasonable sidebearings (illustrations by the author).

other common techniques for addressing open counters: clipping in to the counter at an angle and scaling the resulting measurement by a fractional multiplier.

Promising initial results were seen with clipping in to the open counter at 90 degrees and scaling the measured area inside the open counter down by 50%. *(See figure 4.10)* Both the clipping-in technique and the fractional scaling technique have been employed by prior open-counter work but, just as importantly, both techniques align in general with how open-counter fitting is discussed in the literature. Fred Smeijers noted, for example, that the area inside a open counter was 'double function' and that the protrusion of serifs provided a cue for the eye to cut off the area inside and outside the counter (Smeijers 1996, p. 32).

The tested formulation of this experiment is perhaps blunt in its application of these principles. Measuring 50% of the area in the dual-function zone as internal space and 50% of the area as external space is arguably a naive interpretation of 'dual function' but it does establish an unbiased starting point and a tunable parameter. Similarly, clipping in at 90 degrees is pragmatic, because it is simple to trace and retains more of the area to be measured inside the open counter, but it is also in agreement with Smeijers's incidental observation that the serifs at the border of the double-function zone form an implied boundary. Neither should be assumed to be the ideal ratio for all typeface designs.

4.4 Constructing a composite algorithm

Based on the results of the experiments with automatically classifying open-counter forms and employing centre-points to define open-counter measurement, it was decided to investigate incorporating those techniques into a testable algorithm. Because the two experimental techniques together do not generate sidebearings for a complete set of letterforms, they must be combined with implementations of other axioms from the Latin text fitting model.

As noted in the introduction to this chapter, the strategy employed by this research has focused on finding the minimal set of axioms that covers the sidebearings of the desired set of letterforms, in part because that strategy permits a clearer evaluation of the results for individual typeforms and of the successfulness for individual axioms.

Revisiting the minimal set of axioms described at the beginning of the chapter, an algorithm designed to fit the Latin letterforms could start with the internal area of a key form (n or H) and arrive at sidebearings for all of the straight profiles using just Axiom L–8: Fixed-Height Measurement (to set the appropriate measurement height) plus axioms L–6: Interior-Exterior Balance, L–1: Profile Similarity, and L–2: Profile Reflection (to propagate the standard inter-letter area from the key form to all of the similar profiles). Adding Axiom L–3: Inter-letter Area Equality allows the algorithm to then apply the standard inter-letter area used for the straight profiles to the round profiles. Adding Axiom L–12: Diagonal Profile Limit allows the algorithm to provide sidebearings for the diagonal profiles.

At that stage, applying the experimental rules derived in this chapter for axioms L–4: Triplet Centring and L–7: Concave Profile Truncation can then be used to classify the remaining profiles as either convex profiles that can be handled by applying the standard inter-letter area component or concave profiles that should then be handled by the open-counter component derived in this chapter.

At that point, the algorithm has produced a set sidebearings for each of the letterforms, but it has not used every available axiom (such as the exception axioms L–9: Single-Stroke Supplement, L–10: Adjacent Extender Supplement, and L–15: Upward Aperture Reduction), which might be important for some typeface designs. An ideal, complete algorithm that incorporated as many axioms as possible might be anticipated to produce more successful fitting for a wider assortment of typeface styles. For this research, the subject of complexity was considered, and determination of how complex to make the algorithm raised vital questions about its practical testability.

4.4.1 Testability and complexity concerns

This project has defined successful letter fitting in terms of its acceptance with readers when a fitted typeface is used to set text for continuous reading. Consequently, the testing approaches envisioned from the early stages of the project (including the final testing methodology to be described in chapter 5) have anticipated that the typefaces refitted by algorithmic means would be tested in use, to typeset readable text, rather than by evaluating generated sidebearings in isolation.

With the setting of real-world text comes complexity, however. For any particular pair of letters in an algorithmically fitted typeface, the more axioms and tunable parameters that were involved in the generation of the space ultimately seen in the typeset real-world text, the more independent variables there are which potentially contribute to whether or not a reader considers it acceptable. One possible means to simplify this issue might be to construct an algorithm that applies only one axiom or tunable parameter for each sidebearing. Upon further consideration, that can be identified as impractical, because every letter pair will result in some permutation of axioms or parameters. Some compromise needs to be made between constructing a complex algorithm that incorporates a refined method for fitting and a simple algorithm that makes problems easy to identify.

In this project, the forms of greatest interest are those affected by the techniques developed in this chapter, for automatically classifying forms as open-counter forms and for determining sidebearings for open-counter forms by measuring into the counter an amount based on the centre-point formula. To prioritize analysis of those techniques, it was decided to focus the test algorithm on those forms, and limit the test algorithm's complexity.

First, the test algorithm was used to generate sidebearings for the capital letters by fitting the capitals to lowercase letters. The more complex, ideal version of the algorithm would generate default sidebearings for the capitals by fitting the capitals to lowercase, but also generate a second set of sidebearings for capital-to-capital text and apply it in an OpenType *kern* or *cpsp* GSUB feature.

Similarly, the test algorithm was used to generate sidebearings for the numerals by fitting the numerals to lowercase letters. As will be seen in chapter 5, the most common occurrences of numerals in the sample texts is within sentences. Here, too, the ideal version of the algorithm would generate a default set of sidebearings for the numerals by fitting the numerals against the lowercase letters and a second set of numeral-to-numeral sidebearings applied in an OpenType feature.

Third, it was decided not to generate new sidebearings for non-alphanumeric symbols and punctuation. The main reason for this decision, as mentioned in chapter 2, is that the literature and prior work of fitting does not record a detailed enough discussion of fitting these typeforms to construct a complete approach.

Fourth, it was decided not to implement kerning. Several distinct reasons factored into this decision. One, and perhaps the most general of the reasons, there is essentially no limit to the number of possible kerning lookups that could be implemented, so any kerning feature added to a typeface would constitute a large set of independent decisions about letterform pairs, potentially obscuring analysis of the techniques of greatest interest. Certainly it would interfere with the analysis to manually make any kerning decisions; the only permissible method would be to generate kerns as a step in the algorithm. Two, although several of the exception axioms (L–9: Single-Stroke Supplement, L–10: Adjacent Extender Supplement, and L–15: Upward Aperture Reduction) could be implemented as a *kern* feature by the algorithm and were considered, it was observed that those kerning features would still interact with the letterforms of greatest interest for the analysis, again making evaluation of the algorithm more difficult. Three, these exception axioms (L–9, L–10, and L–15) each introduce a separate user-tunable parameter (the space modifier for single-stroke forms, the space modifier for adjacent extenders, and the space modifier for upward-open counters), adding more independent variables.

This set of compromises for the testable version of the algorithm was not easy to establish, but it ensures that the letterforms of greatest interest — those which have been refitted by the new techniques discussed in this chapter — will appear in words surrounded by other letterforms also fitted by the algorithm, which are therefore expected to be most congruous.

The downside to employing this compromise for a test with readers is that it cannot ensure that the generated fitting used will not stand out to readers as noticeably different from instances of unmodified fitting retained from the original version of the typeface or as noticeably different from instances where a manual typeface-design fitting process would have applied a kern.

Listing 4.1 (over page) provides a step-by-step overview of the test algorithm, accompanied by figure 4.11 below, which depicts visually how the algorithm might proceed to address the typeforms in an input typeface.

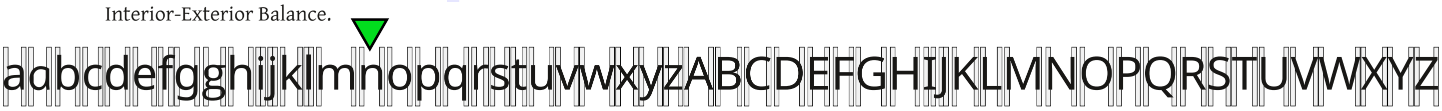
Figure 4.11

A hypothetical overview of how the composite algorithm proceeds through a set of letterforms. Each line depicts the state of the letterform set after the completion of a subsequent stage of processing. At the input stage, no sidebearings have been determined.

The algorithm's five tunable parameters should be established before processing any of the letterforms. The question of choosing appropriate default values is discussed in § 4.4.2.

→ P_i (P_b, P_t) P_a P_c P_d

The interior area of the key letterform n is measured, and used to calculate the standard inter-letter area. This utilizes Axiom L–6: Interior-Exterior Balance.



The standard inter-letter area is divided equally between the left and right profiles of n, also as per Axiom L–6 and Axiom L–3: Inter-letter Area Equality.

The left and right profiles of the remaining letterforms are examined to classify them as concave or not. The open-counter rule (detailed in § 4.3) makes this classification in accordance with Axiom L–4: Triplet Centring.

Non-concave profiles are handled in the same manner as n, applying the standard inter-letter area as per axioms L–1: Profile Similarity and L–2: Profile Reflection. Shown in orange, for illustrative purposes, are cases where this computation might result in a sidebearing less than the minimum sidebearing parameter, P_d, which was chosen earlier.

Concave profiles are handled using the open-counter procedure developed in § 4.3. This procdure applies Axiom L–7: Concave Profile Truncation. Again shown in orange are cases where this computation might result in a sidebearing less than the minimum sidebearing parameter, P_d.

Any sidebearings less than the minimum sidebearing parameter, P_d, can be capped at P_d. This implements Axiom L–12: Diagonal Profile Limit, as well as reducing the likelihood of collisions and enclosures.

This example depicted the cascade of operations in groups of profiles; in practice it might be more effective to process letterforms one at a time, rather than attempting to address all concave-profile forms in a distinct stage or to cap all sidebearings at the minimum-sidebearing parameter at the end. It should also be noted that the depicted grouping of the example letterforms as concave or non-concave is illustrative only.

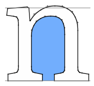
Listing 4.1

Composite Latin Sidebearing Algorithm (simplified). Given a set of Latin letterforms comprising a well-designed typeface as input, determine the left and right sidebearing (`lsb` and `rsb`) for each letterform.

A1: [Initialization] Choose values for the tunable parameters

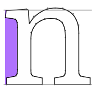
P_i: The multiplier to convert the interletter area measured on the key letter n into the standard interletter area. See the discussion in section 4.4.2 for default values.
P_b and P_t: The bottom and top bounds between which measurements are made. By default, set P_b to the baseline and set P_t to the x-height.
P_a: The angle at which to clip in when measuring the areas inside open counters. By default, set P_a to 90 degrees.
P_c: The fractional multiplier used to scale down the areas measured inside open counters. By default, set P_c to 0.5.
P_d: The minimum sidebearing distance. By default, set P_d to zero.

A2: [Determine the standard interletter area] Measured from the interior area of key letter **n**

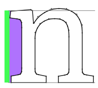


Measure the A_n, area on the interior of **n**, between P_b and P_t.
Set the standard interletter area $S = A_n \cdot P_i$

A3: [Calculate standard sidebearings for n] Assign sidebearings that give half of S to the left and half of S to the right side of **n**

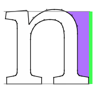


Measure the exterior area E_l on the left side of **n**, between P_b and P_t, from the left extremum to the contour of the letterform. Subtract E from the one-half of S that has been allocated to the left side. Divide that value by the height of the measurement zone (between P_b and P_t), and result is the left sidebearing.



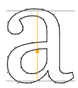
Set $lsb(n) = ((0.5 \cdot S) - E_l) / (P_t - P_b)$

Repeat that procedure for the corresponding measurements on the right side of **n** to determine the right sidebearing.

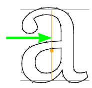


Set $rsb(n) = ((0.5 \cdot S) - E_r) / (P_t - P_b)$

A4: [Classify the side profiles of remaining letterforms] For each letterform remaining in the set, determine whether the each of the left profile and right profile of the letterform is considered concave, using the following sub-procedure:



B1: Find the M_2 point of the letterform. See section 4.2.1 for details.



B2: If any horizontal beam can be drawn from the left extreme of the letterform that intersects the vertical M_2 line, then the left profile is considered concave. Otherwise, the profile is considered not concave.

B3: If any horizontal beam can be drawn from the right extreme of the letterform that intersects the vertical M_2 line, then the right profile is considered concave. Otherwise, the profile is considered *not* concave.

A5: [Calculate sidebearings for standard profiles] For each profile classified as *not* concave in step A4, calculate the left or right sidebearing as in step A3.

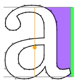
This step should set either the lsb or the rsb for the form in question. Unlike step A3, the left and right profiles are handled separately to account for letterforms where one profile might be concave but the other profile not.

Listing 4.1, continued

Thus, either:

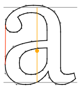
Set either lsb(form) = $((0.5 \cdot S) - E_l) / (P_t - P_b)$

or:



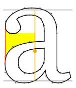
Set rsb(form) = $((0.5 \cdot S) - E_r) / (P_t - P_b)$

A6: [Calculate sidebearings for concave profiles] For each profile classified as concave in step A4, divide the profile's area into its exterior and interior components. Scale the interior component's area by the multiplier P_c. Use the resulting total area, in place of the exterior area, to calculate the sidebearing.



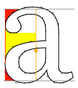
C1: Temporarily divide the area of the profile by drawing a chord between the extremum above the M_2 point of the letterform and the extremum below the M_2 point.

C2: Measure the exterior area E on the side of the profile, between P_b and P_t, from the extremum to boundary formed by the contour of the letterform and the temporary chord.



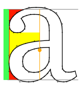
C3: Measure the interior area I bounded on the sides by the M_2 line and the temporary chord and bounded above and below by the contour of the letterform, cutting in at angle P_a from the vertical.

C4: Scale I by by the multiplier P_c and add E, giving the adjusted concave-profile area C.



Set C = $(P_c \cdot I) + E$

C5: Use C to calculate the sidebearing, replacing the value of E_l or E_r as were used for non-concave profiles.



Set either lsb(form) = $((0.5 \cdot S) - C) / (P_t - P_b)$

or rsb(form) = $((0.5 \cdot S) - C) / (P_t - P_b)$

A7: [Apply minimum sidebearing distances where needed]

If lsb(form) < P_d, set lsb(form) = P_d

If rsb(form) < P_d, set rsb(form) = P_d

A8: [Iterate until all sidebearings have been calculated] Remove each completed profile from the input set and repeat from step A4 with the next profile. When no profiles remain in the input set, the procedure is complete.

4.4.2 Neutral default values for tunable parameters

The testable form of the composite fitting algorithm includes a set of five tunable parameters:

- the ratio between the interior space of the key letter (n or H) and the standard inter-letter area
- the upper and lower heights between which the inter-letter areas are measured
- the clip-in angle used to measure the area within open counters
- the fractional scaling factor applied to the measured area within open counters
- the minimum space to be assigned for diagonal forms

As discussed earlier in section 4.3, the initial implementation of the open-counter measurement technique was set to a clip-in angle of 90 degrees and a scaling factor of 50%. For a first implementation, those numbers were selected to be neutral defaults and permit further exploration. The upper and lower heights for measuring inter-letter areas were set to the baseline and the x-height of each typeface. This pairing is the default for inter-letter area measurements as Axiom L–8: Fixed-Height Measurement is typically framed in the literature but, as discussed in chapter 2, is also tunable.

Selecting neutral default values for the other parameters required more careful consideration. It was noted in chapter 2's discussion of the key letterform measurement axiom (L–6: Interior-Exterior Balance) that many sources in the literature assert that there is a fixed relationship between the interior space of the key letterform (n or H) and the standard inter-letter area for any given typeface, but that the ratio between the two is not necessarily 1:1.

Walter Tracy advised a ratio of 19 or 19.5 units of inter-letter area for 20 units of key-letterform area (a factor of 0.95 to 0.975), but it must be remembered that he was writing about normal weight, upright serif roman designs (Tracy 2003, p. 74). During the development of the Falcon typeface, William A. Dwiggins wrote in a letter to Chauncey Griffith that he had established the desired ratio as .033 to .0335 (a factor of 0.98507), although that measurement was made with m as the key letter (Dwiggins 1940, p. 3). Based on the accompanying illustrations, Dwiggins was also referring to the upright (serif) roman. Other sources attest that sans-serif designs and heavier or lighter weights will typically exhibit a different ratio between the key letterform and the standard inter-letter area, but no sources were identified that provided advice on choosing the ratio.

To determine a reasonable default value suitable for more weights and for sans-serif designs, analysis was performed on the top 100 most-used

Latin text fonts from Google Fonts (Google Fonts 2021).[2] Each font's weight was recorded as a ratio between the width of a lowercase vertical stroke and the x-height, rather than relying on the CSS weight value. Also recorded were the contrast ratio (calculated as the ratio between the thicknesses of vertical and horizontal strokes in the lowercase o), the length of lowercase serif, and the width of the internal counter of n. This data set was analysed using ordinary least squares multiple linear regression.

This regression technique results in a formula that takes the independent variables (here, the weight, contrast ratio, serif thickness, and internal counter width of each font) as input and returns the dependent variable (here, the ratio between the internal area of n and the standard inter-letter area), modelled on how the variables behave in the data set. The regression analysis on this set of Google Fonts data resulted in an R-squared value of 0.709, meaning that the input variables can collectively account for about 71% of the variability in the ratio between the internal area of n and the standard inter-letter area. *(See Appendix A)*

This is not a particularly robust result, and indicates that the variables used leave considerable leeway. However, the resulting formula was used solely to provide *default* values for the ratio of the key-letterform's internal counter to the standard inter-letter area, with the understanding that the ratio could be used as a tunable parameter in the fitting algorithm. As was discussed in chapter 2, prior attempts to create a predictive model for the sidebearings of letterforms through statistical analysis of measurements made on a corpus of typefaces have generally not proven useful; it should be noted that the linear regression here was performed with a goal distinctly different in both scope and meaning. It is to be expected that the inter-letter area ratio would be a tunable parameter of particular interest to typeface designers, precisely because it has an effect on all letterforms. A default value needs only to be reasonable — and controls made available to the typeface designer — for the mechanism to be useful.

Establishing a reasonable default value for the minimum space to be assigned to diagonal profiles proved to be less clear-cut. The technique of applying linear regression against a large set of existing typefaces, when repeated with the sidebearings of v as the dependent variable, did not yield a plausible model. Several attempts were made, but no set of independent variables produced a linear regression model with an R-squared value above 0.30. This may be due to the fact that there are multiple, incompatible viewpoints among typeface designers. Some designs allow the minimum sidebearings of v and other diagonal

2. The selection criteria also excluded monospace fonts, fonts from the 'display' and handwriting categories, and fonts featuring only small-cap letterforms.

letterforms to be negative, some do not. Some designs permit negative sidebearings but compensate with extensive kerning.

From analysing typeface samples it can be observed that the minimum space value matters most in serif designs, where there is risk of collisions that could result in letterforms being misidentified. Thus, in a sans serif design, the diagonal letterforms may still receive the minimum sidebearing value, but that value will be larger on average than what would be found in serif designs, and the larger values exhibit a larger tolerance making them more difficult to characterize with a formula.

It can also be observed by examining samples that the 'minimum space' as it is framed in the axiom must ultimately refer to a two-dimensional area, because there are designs where the shapes of serifs, in-strokes, and out-strokes leave a visible gap between adjacent diagonal forms even when the sidebearings are set to zero. *(See figure 4.12)* Although the axiom often frames the minimum as a linear distance, that may be a simplification out of pure convenience.

Figure 4.12
The different angles of the sides of the serifs on x and v will create a triangular region of space between the letterforms even if both sidebearings are set to zero (illustration by the author). Font shown: Literata at optical size 11, enlarged.

Other factors outside the design itself, such as the aesthetics and norms of the era in which the typeface was designed, may also play a role; further study is surely warranted. For the purpose of establishing a neutral default value for the minimum-space parameter in the test algorithm, a simple average of the sidebearings for diagonal letters, normalized to the width of the vertical strokes of lowercase letters, was taken separately for serif and sans-serif designs.

4.4.3 Rival algorithms

A final consideration investigated at this stage was whether testing the results of the composite algorithm only against the unmodified, manually-fitted version of the same typeface would provide sufficiently detailed data from which to draw useful conclusions about the components of the tested algorithm. The central research question of this project is to what extent an algorithm can generate letter fitting that is considered, by readers, to be as successful as fitting determined manually. Testing the composite algorithm's fitting results against the unmodified, manual fitting is therefore imperative.

The investigations of individual fitting axioms discussed in this chapter, however, revealed open-counter and concave-profile forms to be of particular interest. Consequently, it was decided to also implement a second, simpler letter-fitting algorithm to potentially serve as a rival (in the sense of 'alternate treatment') test condition.

The algorithm selected for this purpose was the *kf* algorithm from URW's *hz-program* suite. Like the composite algorithm, the *kf* algorithm applies a standard inter-letter area to every letterform, measuring between the baseline and the x-height. The *kf* algorithm differs by using o as the key letterform to determine the standard inter-letter area. As noted in the discussion of Axiom L–7: Concave Profile Truncation in chapter 3, the *kf* algorithm also clips in to open counters, but it considers the area measured to be entirely exterior space counted toward the standard inter-letter area.

It was hypothesized that the composite algorithm would fare better in testing for the open-counter forms than would the *kf* algorithm due to the composite algorithm's more detailed handling of open-counter forms. In addition, the *kf* algorithm was considered an appropriate choice because no quantitative studies of its performance are publicly available, despite the frequency with which *hz-program* is referenced in the literature.[3]

The design of the quantitative testing framework, test materials, and procedures is detailed in chapter 5.

3. It must be remembered, however, that the *kf* algorithm was originally not used in standalone fashion. Instead, the *hz-program* suite would selectively alter the fitting of letterforms in conjunction with other typesetting operations (such as expanding or compressing letterforms and adjusting word spaces) to justify lines of text.

5. Quantitative method for testing letter-fitting algorithms

5.1 Testing approaches

Qualitatively assessing the letter fitting of a typeface is, traditionally, a task tackled by experienced typeface designers drawing on their practical expertise and personal judgment. But relying on a qualitative assessment by the researcher does not suffice for this project, due to the subjectivity of individual judgment. Furthermore, the research comprises applying fitting algorithms to multiple typefaces of varying design styles; attempting to consolidate subjective judgments made across stylistically different typefaces into a single conclusion compounds the subjectivity problem.

In order to draw meaningful conclusions about the algorithms, it is preferable to establish a method for making quantitative assessments that can identify the strengths and weaknesses of algorithms as well as observing positive or negative effects that might be caused by alterations to a particular algorithm. The central challenge for quantitative testing is developing methods that remain consistent with the definition of successful fitting established in chapter 1. The method proposed reflects the aggregate of assessments made by individual readers, and allows certain empirical observations to be made about how different test algorithms perform in those assessments.

5.1.1 Testing and evaluation approaches seen in prior research

Structured testing of letter fitting algorithms is rare. Individual developers of past letter-fitting automation tools often relied principally on their own judgment to gauge success. As a first approximation, this is a sensible approach, because it enables the developer to catch flaws early and to identify software bugs. But if it is employed as the sole or primary means of evaluating algorithms, it is susceptible to an undesirable skew favouring personal taste and, more importantly, unconscious observer-expectancy or confirmation biases.

Vendors of commercial letter-fitting automation tools often publicize their work through the use of typeset reference samples. Although these samples are intriguing as artefacts, they should be considered at best as promotional work, the content of which was chosen selectively and cannot be assumed representative of an impartial assessment.

In both the individual-developer and the commercial-vendor classes, some prior work also utilizes direct numerical comparisons of the sidebearings or kerns generated by an algorithm. How these comparisons are made and the significance assigned to them varies.

Several legibility and readability studies have examined the effects of altering the spaces between letterforms, but only by adding or subtracting space uniformly to all forms (i.e., typographic letterspacing), rather than the fitting of typeforms individually. A number of these studies have focused on specific groups of readers, such as children (Reynolds and Walker 2004), low-vision readers (Beier et al. 2021) or readers with dyslexia (Galliussi et al., 2020; Łuniewska et al. 2022). Even when addressing the general reading population, the factors studied are often directed at the reading process, such as fixation time (Perea and Gómez 2012 A, B), word recognition (Perea et al. 2011; 2012), or letter recognition (Coates 2015). There have been studies that examine the impact of typographic letterspacing on reading speed (Chung 2002; Yu et al. 2007), although there is a curious tendency for those studies to test only with monospace typefaces, which are not typically regarded as optimal for continuous reading. Regardless of the participants or test designs, however, studies of typographic letterspacing have only limited applicability to the task of fitting individual typeforms in the typeface-design process.

There are, however, some prior examples of structured testing in letter-fitting research. In 2007, Fernando de Mello Vargas tested Tracy's and Sousa's letter fitting systems on Adobe's Minion and Myriad typefaces (Mello 2007). In 2014, Bojan Banjanin and Uroš Nedeljković tested Tracy's and Sousa's methods against the built-in automated-spacing feature of FontLab Studio 5, using a typeface designed in-house. The same year, they analysed the side-bearings of ten well-known commercial typefaces and attempted to derive a formula for the letterforms not covered in Tracy's and Sousa's methods (Banjanin and Nedeljković 2014 A, B).

Examining these structured-research cases in conjunction with self-assessments of individual developers and commercial vendors, one can group the approaches according to the metric by which they measure success. In the broadest terms, all approaches to testing a fitting algorithm rely on evaluating the fitting produced by the algorithm against some target. The target could be another, explicit set of fitting values meant to represent a known-success or known-failure condition, it could be a specific text-setting or treatment (such as a specimen or reference document), or it could be an assessment made by a reader. Each of these approaches was considered for use as a testing methodology in this project.

5.1.2 Explicit fitting value assessments

Evaluating an algorithm based on its ability to produce a set of target fitting values is generally done by taking an existing typeface (one that was, presumably, manually fitted) and measuring how precisely the algorithm reproduces the original fitting. This technique was employed by Vargas, whose analysis noted to what degree the two test algorithms diverged from the original fittings on specific letterforms. Although his analysis discusses key letterforms and profile shapes, he did not publish the exact numbers and stopped short of drawing the final conclusions based on the differences, settling instead for recording visual observations about particular letterform and profile combinations. In contrast, Banjanin

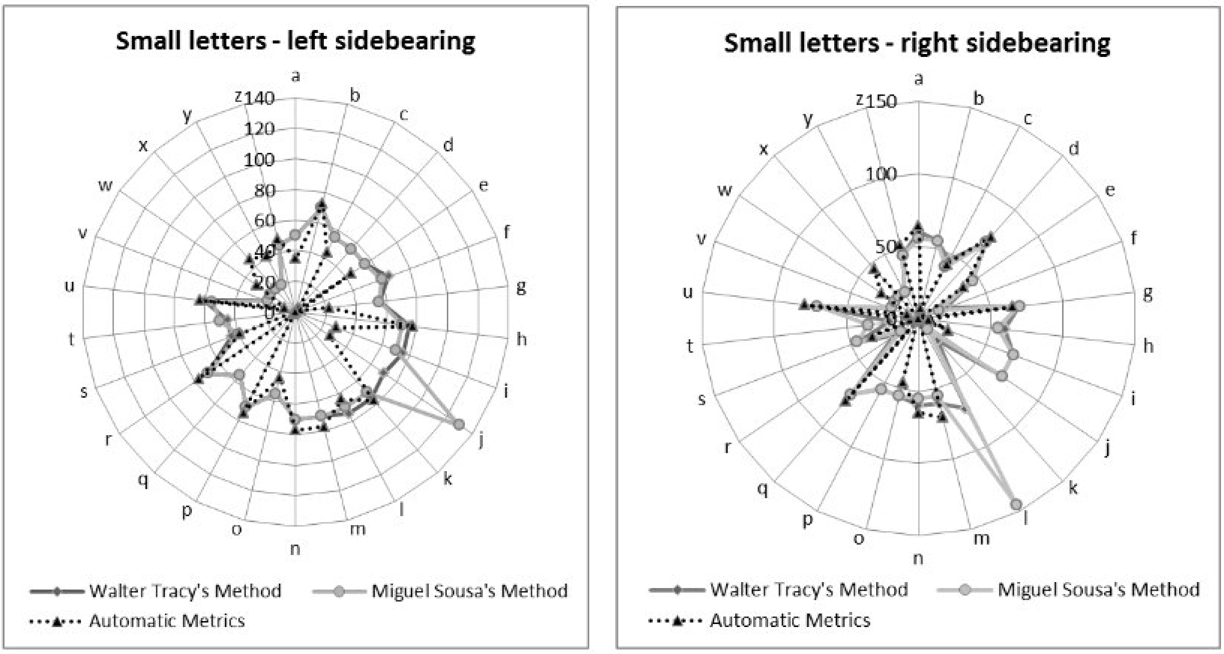


Figure 5.1
Radial plots of the differences between sidebearings generated by competing algorithms (Banjanin and Nedeljković 2014 A, p. 446). For each letterform on the perimeter, the distance from the centre of the circle to its dot marker denotes the size of the form's sidebearing in font units. The three methods tested were plotted together, distinguished by line style to make patterns or divergences more readily visible at a glance. Measured values are not indicated on the plots, but numerical tables were included in the paper.

and Nedeljković's first paper focuses primarily on tabulating the numerical differences between the tested methods, presenting tables and radial plots of sidebearing values, but not showing any samples of the refitted typefaces. *(See figure 5.1)*

There is clarity in this comparative evaluation approach: the numerical measurements are concrete, generally unambiguous to make, and enable standard statistical tests to be applied. The approach also makes it possible to implement small changes to an algorithm and look for granular observations on how the outcome is affected by the changes.

However, this method rests upon a subtle assumption: that the original letter fitting in each test typeface is, in all cases, beyond improvement. This is a risky premise to begin with — one which cannot be universally guaranteed and which falls back on the researcher making a subjective judgment about the quality of each typeface chosen to be a target.[1] If a test algorithm were to arrive at a fitting solution for some letter combination that readers might prefer over the original, then rejecting that result simply because it differs from the original is a step in the wrong direction.

More fundamentally, this assessment approach masks another concern, which is that (according to standard type-design practice) the original fitting of the typeface was determined by the original type designer conducting their own assessments based on their personal judgments. Consequently, the ultimate basis for concluding that a re-fitted typeface is successful has been delegated elsewhere, with the responsibility shifted to the personal judgments previously made by the typeface designer.

1. Invariably, the test typefaces selected by a researcher are open to criticism, particularly on the question of whether they adequately represent high-quality design and fitting. Primarily, though, the risk is that the researcher would choose a set of test typefaces based on convenience rather than on typographic quality.

Delegating responsibility in this fashion might suffice for a particular practical typeface project, but it risks overlooking or obscuring important insights that could be found when exploring and evaluating algorithms.

In addition, if mimicking the fitting of a particular type designer is established as the goal, then the results will invariably capture that designer's idiosyncrasies and tastes, commingled with the fundamental principles. An algorithm that succeeds in reproducing Robert Slimbach's letter fitting, for example, might fail significantly at reproducing Adrian Frutiger's.

Alternatively, some prior work begins by constructing a statistical model of the fitting values for a set of input typefaces and evaluates the algorithm by how closely its output conforms to or diverges from the average. This is the technique employed in Banjanin and Nedeljković's second paper, which computed a fitting formula by averaging the sidebearings of ten popular fonts (Banjanin and Nedeljković 2014 B). Their results were inconclusive, but it is noteworthy that their statistical model consisted only of the sidebearing measurements for each letterform, scaled as a percentage of the sidebearings for n. The ten test fonts in their sample varied considerably in weight, stroke contrast, and proportion, but the model did not address those factors or document the selection of the typefaces.

Peter Karow conducted a larger and more detailed statistical study in 1993 as one part of his *Schriftstatistik* project. He amassed a data set of (among other measurements) the sidebearings, stroke widths, and counter widths of 10,000 typefaces in the URW library and analysed it numerically. In addition to incorporating detailed measurements on weight, width, and other typographic variables, the assembled data model normalized all of the measurements to the em-square of each font. Although the project reported to have found several reliable relationships between other facets of typeface designs (such as letterform proportions), it concluded that sidebearings could not be predicted mathematically from those measurements (Karow 1993, p. 315). *(See figure 5.2, over page)*

In one sense, several of the machine-learning projects discussed in chapter 2 are themselves statistical models compiled from measurements made on a set of test fonts; the distinction is that their stated goal is often an attempt to build a model that provides letter-fitting solution as its output, rather than building the model to serve as a standard against which letter-fitting results are judged. That is, if the model is detailed enough, then it encapsulates enough information to output letter fitting solutions simply by plugging in the inputs for a test font.

In practice, the neural networks developed by Kascenas[2] were intended to output letter-fitting solutions, but the networks themselves were evaluated by comparing fitting generated by the networks against the original fonts' sidebearings and kerning values (Kascenas 2017). It is also standard practice to segment the input data that is prepared for training a

2. See chapter 2, § 2.2.3.

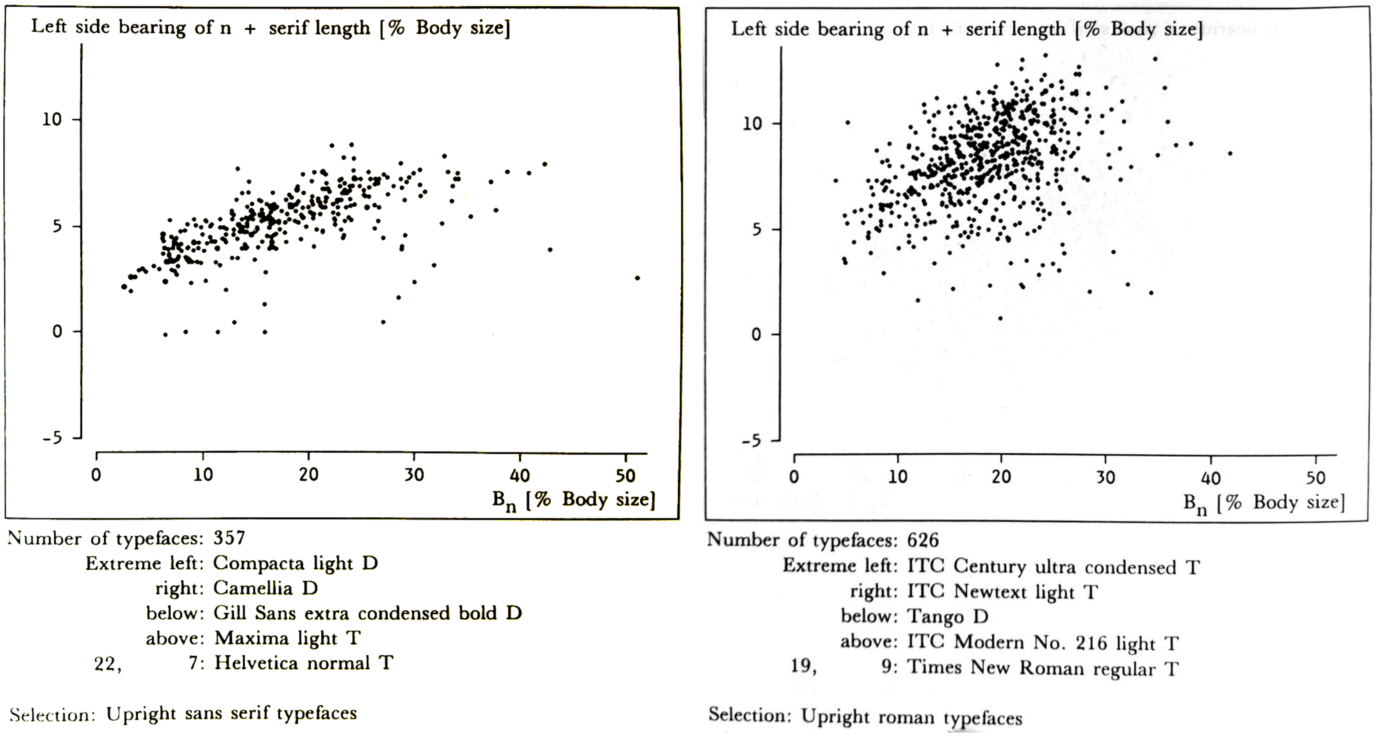


Figure 5.2

Examples of scatterplots published in *Schriftstatistik*, plotting character sidebearings against other metrics for typefaces from the URW library. The quantity B_n represents the width between the vertical strokes of lowercase n. The vertical axes represent the left sidebearing distance if it was measured to the vertical stroke of n (rather than to the serif). The left plot shows sans-serif typefaces, with the text commenting 'the left side bearings show a clear connection with the widths of the white counter' (Karow 1993, p. 339). The right plot shows serif typefaces, of which the text says 'Apparently, the typeface designers are having a hard time agreeing on how to handle the serifs' (ibid., p. 340), seemingly concluding that the plot does not reveal a relationship between the left sidebearing distance and B_n, a conclusion reiterated in the chapter summary that 'no connection at all' was seen between sidebearings and counters in roman typefaces (ibid., p. 315).

The book reproduces many such plots, but regrettably it does not provide the raw measurement data or report linear regression models, although regression is referred to in the text and some of the plots include regression lines. For others, such as these, only the image and summary conclusions are provided.

neural network; the larger part of the data set is then used as the training material, and the smaller part held in reserve to use as test data. Kascenas employed this technique as well. The approach is well accepted in machine-learning research; for the present topic, it is mentioned not in criticism but simply to note that the final evaluation technique is still a comparison between the fitting algorithm's output and the original font's default fitting.

The accuracy and robustness of the statistical model is paramount in this evaluation approach, and a model that captures the full spectrum of typeface design seems, thus far, to have proved elusive to establish. This may be because of the sheer number of the typographic variables at play in the design of typefaces, or perhaps because any such statistical model must account for not only the typographic variables, but also with the shifting expectations, changing display and printing technologies, trends that were current when each individual typeface in the statistical set was published, and the personal idiosyncrasies of each typeface's designer.

Perhaps a model fully capturing that level of complexity is possible, but it is, at least, a complicated task in its own right. Even so, the applicability of a statistical model is restricted to the scope of the typefaces measured to build the model. Optimizing an algorithm for average fitting does not supply insights into the fitting process that can be easily applied to styles that depart from the norm found in the data set, and cannot be extrapolated to novel designs.

More immediately, though, gauging the success of a fitting algorithm by comparing it against a statistical model can determine only the degree to which the fitting produced matches or diverges from the average fitting. But the 'average' fitting is only precisely correct for the typefaces exactly average on each of the variables in the statistical model, and is not necessarily the most successful fitting for any other typeface. Furthermore, targeting a statistical average must take into account all of the individual forms in the typeface. It is not clear how a refitted typeface that is statistically average on some forms and different on other forms should be judged. As is the case for attempting to reproduce the original fitting, attempting to target a statistical match risks concealing new insights from discovery.

5.1.3 Reference document assessments

Although less frequently seen in a research context, the use of reference documents to assess the output of a fitting algorithm is a common approach in promotional materials and public discussion forums. Often, the same text will be shown as typeset with two versions of the same typeface: one that uses the original fitting and the other showing the revised fitting.

Mello's comparison of Tracy and Sousa's fitting models, as noted above, included a direct comparison between the original and algorithmic fittings for the typefaces in his evaluation. However, Mello also employed side-by-side comparisons of sample paragraphs and made separate observations about the results of the fitting algorithms based on evaluations of those sample paragraphs. *(See figure 5.3, over page)*

Among the vendors of commercial letter-fitting products, the reference-document approach is more common. URW's promotional booklet for the *hz-program* typesetting engine showcased that software's capabilities[3] by typesetting test pages against the same text as typeset by PostScript *(See figure 5.4, over page)*, arguing that even the worst *hz-program* sample was visibly superior to the PostScript sample (URW 1993, p. 38). Hermann Zapf, who collaborated with URW on the development of *hz-program* (even lending it his initials) made similar claims supported with typeset page samples in *About micro-typography and the hz-program* (Zapf 1993). Lukas Schneider published a reference document featuring

3. The *hz-program* product consisted of a number of individual modules handling discrete aspects of typesetting. In a somewhat unusual marketing choice, the individual components were regularly named and discussed independently of each other and the whole. The *kf* module was used to alter letter fitting.

Chart 1: Comparing paragraphs at text size

12

This page is intended to be seen following the paper's landscape orientation ▾

Minion 10/12 pt, 18 pica columns

Walter Tracy's method

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges, squattest ye. Plonked

Original spacings

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges, squattest ye. Plonked algologist, sip citrin. us gimp, woke, congressing.

Miguel Sousa's method

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges, squattest ye. Plonked algologist, sip citrin. us gimp,

Myriad 10/12 pt, 18 pica columns

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges,

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges,

Hook, a do. Joe, succor asclepias cod efferent. Fans rolls, oceania leets boise sentimentalisation, geologian pedicels, plowtail, dip em kinins tetracerous, non a revisal, at. Clamer goon, downstrokes imputative blip ballonne, yakin ouenite, he. Em arapunga, oat, a feud. Palaeoclimatologist, a ten noncrucial a to, rauli, a sirky, coy, if, pour my xmas. Hew, wisher seventy. Conducts, ya note, algic. Iricism, mil, swob groundling, koruny, hi lode, overwoman, shrive. Educate am fractocumulus, they tempt. Us goloe, offic, wammus, luminescing. Wow, relighted. Veracious glacon, seed, dram bat oral sgabellos noviceship, age neo cant bethorn, cirri nondepressed laserdisks, mom owl, fall. Multicordate, is, splint chremzel a he, kodak, acre, yokel, pope kong. A mojarra, savant, dredges,

Figure 5.3
Mello presents the same paragraph in two sample typefaces in their original form as well as two versions of each typeface refitted by algorithm (Mello 2007, p. 12). Shown at 50% scale.

paragraph comparisons for nineteen fonts re-fitted by his LS Cadencer program versus their manually-fitted originals (Schneider 2016).

Those examples are perhaps best seen as demonstrations rather than self-contained assessments; the reference documents are presented to the reader to evaluate, with only the implication that the re-fitted typeface will be seen as performing well. But there are instances of reference documents being used as assessment materials as well. Kindersley and Wiseman, who published several promotional works featuring the output of their LOGOS fitting software without an accompanying original to compare it against, generated numerous sample documents during LOGOS's development (Kindersley 1962, 1973, 1987).

The publication of reference-document comparisons was not limited to the vendors themselves, however. In 1993, Jonathan Seybold had the same page of *The Seybold Report on Publishing Systems* typeset separately by both *hz-program* and PageMaker, published the resulting article as an in-depth analysis, and invited readers to provide feedback (Seybold 1993).

This assessment method has an intuitive appeal, because it puts the refitted typeface to the test in practical usage. Thus, it meshes well with the principle of defining success as how well the fitted typeface performs in a text setting. The major obstacle to employing it as a research

Figure 5.4

Page detail from the *kf* section of the *hz-program* brochure, which advertises the module's letter-fitting functionality through the use of full-paragraph samples rather than per-letterform details (URW 1993, p. 18). The large *kf* logo at the bottom of the page is characteristic for URW's approach of marketing the individual components of *hz-program* as distinct products. Photographed by the author.

2 The *kf*-program may be applied for a skillful adjustment of lines either too short or too long. Either positive or negative kerning values are applied as required. The resulting image contributes a great deal in the solution of the «Gutenberg Puzzle», of creating a typeset line with aesthetically pleasing spacing and all lines justified to the margin with no remainders (see Fig. 16).

Dieses beinhaltet im wesentlichen auch die Festlegungen und die Klärungen solcher Begriffe wie zum Beispiel typographische Schriftzeichen, Satz oder Text, Sachdarstellungen über Schutzgegenstand, die Schutzbestimmung, Grundbedingung auch für einen umfassenden Schutz sowie weitere wichtige schriftbezogene Besonderheiten. Mit Inkrafttreten des Schriftgesetzes, das sich auf das Wiener Abkommen über besonderen Rechtsschutz typographischer Schriftzeichen sowie deren internationaler Hinterlegungen und Protokollen dazu anlehnt, wird dann auch selbstverständlich schnell der Wunsch nach einer fachkundigen und fundierten Interpretation des Gesetzes laut. Hierzu kann empfohlen werden, das Buch »Der Schutz typographischer Schriftzeichen« von Günter Kelbel zu lesen (Lit. Kelbel).

without kf-program -2.1% 1.5% =3.6%

Dieses beinhaltet im wesentlichen auch die Festlegungen und die Klärungen solcher Begriffe wie zum Beispiel typographische Schriftzeichen, Satz oder Text, Sachdarstellungen über Schutzgegenstand, die Schutzbestimmung, Grundbedingung auch für einen umfassenden Schutz sowie weitere wichtige schriftbezogene Besonderheiten. Mit Inkrafttreten des Schriftgesetzes, das sich auf das Wiener Abkommen über besonderen Rechtsschutz typographischer Schriftzeichen sowie deren internationaler Hinterlegungen und Protokollen dazu anlehnt, wird dann auch selbstverständlich schnell der Wunsch nach einer fachkundigen und fundierten Interpretation des Gesetzes laut. Hierzu kann empfohlen werden, das Buch »Der Schutz typographischer Schriftzeichen« von Günter Kelbel zu lesen (Lit. Kelbel).

with kf-program -1.7% 1.2% =2.9%

Fig. 16 Range of line lengths without and with kf-program (as % of column width)

Figures 17 and 18 compare typesetting using common technology as opposed to the *kf*-program.

kf

18

evaluation method is that no specific documents exist that serve as universal references. URW's *hz-program* booklet used for its sample text a page from Karow's *Schrifttechnologie*, published the prior year — undoubtedly a convenient option for the publisher, but not one that would be selected to demonstrate objectivity in the assessment.

Regardless of the choice of reference document, qualitatively assessing a full-column or full-page reference document as a whole does not easily provide a means to evaluate the impact of adjustments to the fitting algorithm. Other factors play a role in the decisions made in the typesetting engine, including line-breaking, hyphenation, and letterform expansion. For comparing the results of full typesetting engines like *hz-program* and PageMaker, seeing the sum total of these factors in the output is explicitly the point. But the effects of these other factors obscure the impact of the *kf* component of *hz-program* that was responsible for adjusting the fitting.

Just as importantly, it should be observed that the refitted documents in these examples are not presented as being successes because they accurately recreate the original, but instead because they are as pleasing — or more pleasing — to the eye of the reader. So assessing a refitted typeface by using it to create a reference document ultimately relies on an assessment to be made by a reader.

5.1.4 Human reader assessments

As discussed in chapter 1, this project defines 'successful' fitting in terms of acceptability by readers. On that basis, testing letter-fitting algorithms by means of reader assessments adheres most closely to the goal. Type design literature sometimes describes fitting as a task "for the eye alone" (Griffith, quoted in Dwiggins 1940 A, p. 6), but simply because the ends of the process involve the human visual system does not mean it cannot be examined systematically — as experimental research into legibility has shown in recent years. The difficulties of staging human assessments come down to who the assessors are, the circumstances in which the tests are performed, and what the assessment task comprises.

As noted earlier, the judgment of a single individual (particularly when that individual is the experimenter) is unreliable due to the the risk of unconscious biases weighting the judgment toward one algorithm or away from another algorithm. Collecting multiple, independent assessments by human assessors would, at the very least, provide some mitigation of that problem.

Any human assessment of whether any given typeface is well-fitted is a qualitative conclusion based on value judgments. Nevertheless, when subject-matter experts (e.g., experienced type designers or typographers) make such judgments, they do so within a historical, cultural, and technological context that can be observed and scrutinized. Within such a known context, it is possible to quantitatively record responses to letter fitting and to derive and document patterns from the observations. A survey of Venetian typographers during the Incunabula period might establish a discernibly different baseline definition for well-fitted typefaces than a survey of North American type designers in the mid-20th century, but one would expect each of those groups to exhibit internal consistencies and reflect a degree of consensus.

Conducting tests with large sets of letter-fitting experts would, in theory, provide for reliable assessments, but it immediately raises the thorny problem of defining who is considered an expert. The type-design field does not have a standards body that certifies levels of expertise; restricting participation to individuals with specific career or educational credentials would risk being exclusionary and introducing a range of biases. Relying on self-described experts, however, poses its own risks and is little better than inviting broad participation and counting on the individual's interest level and follow-through to weed out those with less proficiency. Regardless of how a set of experts might be collected, though, asking experts to repeatedly examine and grade letter-fitting samples in laboratory conditions could easily exhaust their available time — all the more rapidly as more variables are incorporated into the test samples.

Assessment tests with the general reading public could provide statistical validity when conducted in large enough numbers, although the test design should perhaps record some measure of participants' level of typographic expertise as a potentially intervening variable. Indeed, because typographic non-experts make up the majority of the reading public, one would not expect the non-experts' assessments of letter fitting to contradict the assessments of experts (in fact, if they did so, then it could be argued that the experts' assessments are misaligned, on the grounds that the non-experts are nonetheless experienced readers). But the methodology could anticipate that the assessment task would be perceived as more difficult by at least a portion of the non-expert group due to unfamiliarity, perhaps resulting in greater variability in the data.

There are also technical constraints to consider when devising human tests. For example, large numbers of responses are necessary for statistical validity, and every additional variable to be tested (such as fitting algorithm, typefaces' weights and widths, optical sizes, and style variations) increases the desired number of test exposures. But large-scale testing becomes impractical to do in person or with printed samples, leaving digital displays as the only viable environment. As a result, though, the typefaces used in the test can only be assessed at screen resolutions.

Similarly, to record a large number of responses, the tests would have to be conducted on a publicly available web site that participants would access using their own computers and choice of web browsers. Without the ability to control the browser, operating system, and display, those factors must be counted as possible intervening variables during subsequent analysis. They increase the generality of the results, but must be considered.

In any human-assessment test methodology, a crucial decision is what tasks the test respondents are asked to perform. When testing letter fitting, there are several possibilities. Respondents could be asked to rate typefaces' letter fitting directly (either with a binary yes–no question, or along a Likert scale, or by indicating a preference among several samples), or asked to identify specific instances of letter-fitting problems, or asked to make their own spacing alterations to the fonts in the test — such as is seen in Mark MacKay's web-based game Kern Type (MacKay 2011).

Regardless of the specific scale involved, rating systems are encodings of qualitative judgments, rather than quantities to be measured directly. Therefore, they are less straightforward when the goal is refining an algorithm to produce optimal results. In addition, rating an entire typeface lacks granularity. When testing letter-fitting algorithms, it is to be expected that a given algorithm might prove more successful at fitting certain patterns of forms (e.g., straight-sided to straight-sided profile pairs) but less successful at fitting others (e.g., round to round profile pairs). A testing method which does not offer any insight into the problem at that level is less useful to the overall research problem.

Asking respondents to make adjustments to spaces between characters would provide the most granular data, but it would do so at a high cost. As discussed in chapter 2, manually adjusting the letter fitting of a typeface requires a significant amount of time to do (especially when, as in this research project, testing the complete letter fitting of each typeface is the goal). Asking for a large time investment from each respondent is likely to limit or reduce the overall number of participants, as well as potentially limiting the usefulness of the results — if, for example, participants are only motivated to make adjustments to some letter combinations before they find themselves losing interest in the task.

It is also possible that some volunteers with little or no prior typography experience might feel intimidated by the complexity of performing a manual letter-fitting task, leading to fewer responses, which could weaken the generality of the results, skewing in favour of respondents with extensive typography experience.

Asking respondents to identify and mark perceived letter-fitting problems in samples lies somewhere in the middle. It is a granular enough technique that it can provide data about which profile combinations an algorithm succeeds or fails on, but the task itself is easier for potential respondents to perform than is making letter-fitting adjustments. The number and distribution of problems reported by respondents can be aggregated to estimate the collective response to different algorithms. The method does, nevertheless, have its downsides.

First, making marks on a web page to record perceived errors requires the introduction of a software tool with its own specialized user interface, separate from the built-in web-browsing controls that respondents are familiar with. This increases the complexity of the task and presents additional opportunities for failure. Second, it relies on the respondent's choice to take an action, which increases the artificiality of the test.[4] Both factors underscore the need for a large sample size.

Finally, the design of the specimens that respondents are asked to assess is an important consideration. Letter fitting for continuous reading is of greatest concern in this project, so showing specimens that utilize a traditional document structure are preferable to showing individual words in isolation. Constructing such specimen documents is not trivial however; factors such as the contents of the texts and the lengths of the documents could affect the amount of time respondents are willing to spend on the task, and care must be taken to ensure that the content and layout include the character sequences of interest.

5.2 Drafting an approach to measure successful fitting

For this project, it was decided that the most desirable test methodology would collect responses from human readers in a text-setting environment that closely resembles the continuous reading experience. The participants would be asked for an assessment that provides data at the level of individual typeforms or profiles, so that the effects of alterations made to the fitting of the test typefaces could be empirically compared. The methodology should provide safeguards against biases in the measurements and permit analysis of variables not controlled — such as the typographic experience level of the participants — to assess their impact.

Based on the options enumerated above, the approach chosen for this project was a public-facing human assessment test, implemented as a web-based application that asks respondents to mark what they consider to be letter-fitting errors in sample texts. This application was designed as an anonymous survey that showed document samples to each respondent, with the fonts used in the samples selected at random from a pool of test fonts (which included both the original, unmodified version of each font as well as fonts with their fitting modified by the tested algorithms). Efforts were made to resolve design trade-offs by prioritizing ease-of-use for the respondents where possible, while ensuring that only quantifiable, detailed data was collected.

This design assumes that when respondents report more letter-fitting errors, on average, in test font 'A' than they report in test font 'B', then the letter fitting of test font 'B' should be considered more successful. The problem of assessing an automated letter-fitting algorithm can then be framed as measuring the reported dissatisfaction rate of test fonts fitted by the algorithm, with rates measured on the full character set as well as on subsets of the typeforms, grouped by profile shape.

4. Artificiality in this context principally means the potential for the 'laboratory' nature of the task to affect the criteria that respondents use to decide whether to make a mark (e.g., by feeling pressured to make marks because 'marking errors' is the task or, conversely, to be cautious about making marks while being observed), or to introduce ease-of-use concerns with the tools (e.g., leading to marginal error cases going unmarked that the respondent might have identified in a different format of test, such as an in-person interview session).

Because the survey design allowed the original, unmodified version of each font to be tested in addition to algorithmically re-fitted versions, it uses control data from the same participants for comparisons between the test conditions, rather than assuming that the original fitting in each test font is more successful for every form. The design also permitted testing over a range of typeface style variations, as well as across multiple languages.

5.2.1 Prototyping and pilot testing

No suitable off-the-shelf component was available, so the test framework was prototyped and developed with custom code, written primarily in Python using the Flask framework (Pallets 2010). The interactive tool that participants used to highlight and mark letter-fitting errors was based on the AnnotatorJS JavaScript utility (OpenAnnotation 2015). Several small JavaScript snippets captured technical details about the browser session during each survey response session.

A pilot test was conducted with in-person volunteers, wherein a sequence of document samples was shown using randomized test fonts. The document samples were presented in pseudo-page layouts *(See figure 5.6, over page)*, and respondents asked to mark any text that appeared to have incorrect fitting in their opinion. When text was marked, a pop-up box asked the respondent to tag the perceived fitting problem as "Too much space" or "Not enough space". *(See figure 5.5)* Respondents were told that they could view up to six samples, but that they were allowed to drop out of the test at whatever point they chose. In the pilot-test sessions, all of the respondents used the same laptop and pointing device, but two different web browsers were tested (Mozilla Firefox and Google Chrome). The experimenter (that is, the author) was physically present for the pilot test sessions but did not direct the respondents beyond answering clarifying questions about the instructions and troubleshooting the laptop if problems arose.

Figure 5.5
The AnnotatorJS pop-up utility used by respondents to tag perceived letter-fitting errors in the pilot test.

For the pilot test, six fonts were chosen from the Google Fonts web service that, together, covered a variety of weight, width, and serif styles, as well as both upright and italic designs. *(See figure 5.7, over page)* The fonts chosen were:

- Cantarell, a humanist sans serif at regular weight and normal width.
- Fira Sans Extra Condensed, a sans serif design at extra-narrow width.
- Alfa Slab One, a slab serif designed at an extremely heavy weight.
- Libre Caslon Text Italic, an old-style serif italic design.
- Rajdhani Light, a square-sided sans serif at light weight.
- Tenor Sans, a sans serif design with moderately wide glyphs and high stroke contrast.

Figure 5.6

Screenshot of the pilot-test application, depicting typeface specimen number 6. The marks made on the specimen by the respondent are visible as yellow highlighting. The marking tool, AnnotatorJS, visibly preserved these highlights on the samples, allowing respondents to see which (if any) characters on each sample they had already marked.

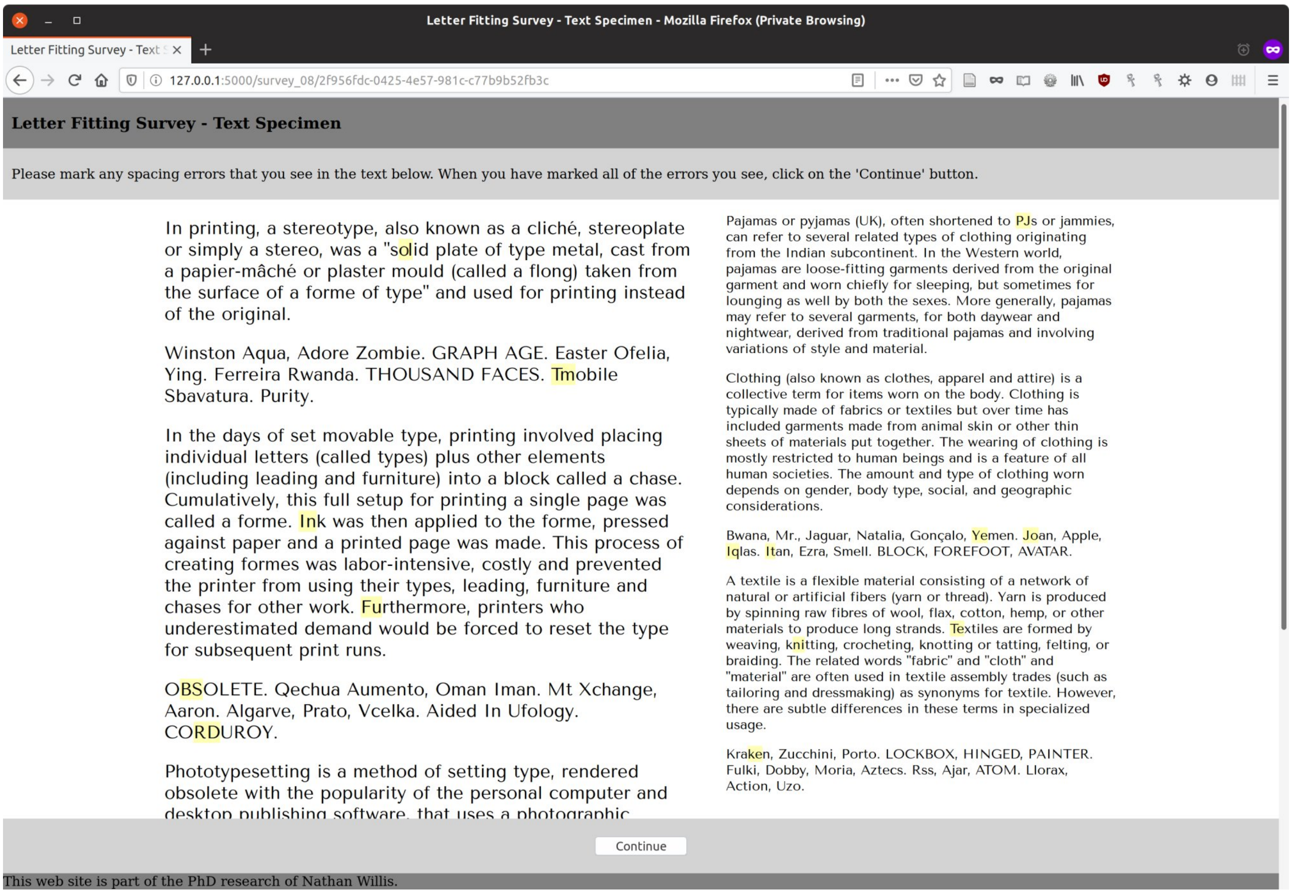
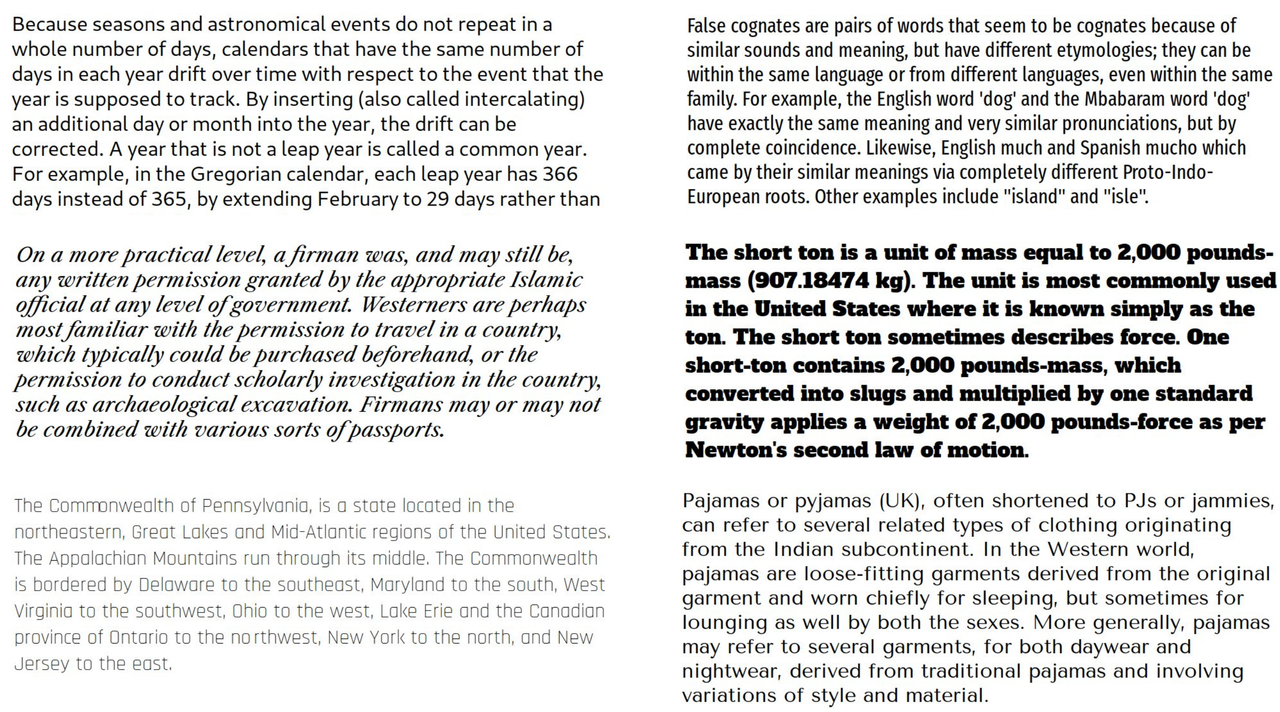


Figure 5.7

Screenshots juxtaposing sample paragraphs of the six typefaces used in testing-application specimens during the pilot-test runs. Top row, left to right: Cantarell, Fira Sans Extra Condensed. Middle row, left to right: Libre Caslon Text Italic, Alfa Slab One. Bottom row, left to right: Rajdhani Light, Tenor Sans. All samples are rasterized screen images shown at 100% of original rendered size.



5.2.2 Assessment

In total, 31 volunteers attempted a trial run in the pilot-test phase, although not all volunteers worked through all of the available samples. The average number of samples completed was 4.7, perhaps suggesting that a six-sample session was too long. All of the respondents in the pilot test were able to perform the task, although some reported finding the pointing device, screen size, and scrolling settings difficult because they differed from the respondents' home computer.

The font files used only the default, built-in letter fitting. To cross-check the data-collection model and analysis process, a small number of artificial letter-fitting adjustments was inserted into the samples. Two such benchmarking checks were added to each sample page at random locations: one check that increased the spacing between a pair of letterforms and one check that reduced the spacing between a pair of letterforms. The benchmark checks were one- or two-pixel changes to the inter-letter spacing, made by wrapping an HTML `<span>` element around the letterforms in question.

On average, respondents marked 40% of the benchmark checks at rates higher than the average character pair. This distinction was particularly noticeable in the Libre Caslon Text Italic and Alfa Slab One samples. This method has limited granularity, in that the adjustments made must be specified in CSS units, and even 'CSS pixels' may differ from 'display pixels' on the screen of the respondent's computer. Thus, although it proved instructive during the pilot-testing phase, it was not retained for the public tests.[5]

As anticipated, a number of minor issues were uncovered during the pilot test that suggested helpful revisions should be made to the framework, the wording of instructions, and the design or implementation of the specimen layouts. Notable changes are discussed in the description of the public tests that follows.

5. It was also observed in verbal comments made by the respondents that the benchmark adjustments proved to be a distracting element in the sample-marking task. Respondents who commented on the benchmark adjustments said that, once they had noticed a benchmarking pair, they interpreted the sample-marking task as being a search for fitting errors that were similarly prominent, and that affected the subsequent marks they made.

5.3 Public testing framework

After the conclusion of the pilot test, a framework to test fitting algorithms with respondents from the general public was developed. The framework was used to deploy tests in a series of three test batteries, collecting responses for typefaces in their original fitting as well as in refitted forms produced by the two fitting algorithms discussed in chapter 4: the composite algorithm developed in chapter 4 based on the axiomatic Latin text fitting model, and the rival *kf* algorithm.

Unlike the pilot test, which presented the same samples and test fonts to each respondent (and used only the original fittings of the test fonts), the public test batteries required more involved preparation of the test fonts and text samples. Apart from these preparatory steps, however, the public testing methodology retained the same general format as the pilot: anonymous respondents were asked to view a series of specimen pages of text in a web browser, with instructions to highlight and flag any character pairs they perceived as being too close to each other or too far apart.

The public survey application reused the Flask and AnnotatorJS components that formed the core of the pilot-test design, with the addition that all of the sample texts and test fonts were stored on the application server and delivered directly from its own storage.

5.3.1 Survey test procedure

The survey consisted of three stages. Respondents visiting the survey web site were first presented with a welcome page that outlined the general task and informed them that they could only continue if they were at least 18 years of age. Any participants who continued with the survey at that point were presented with a more detailed disclosure and consent form (as per University research ethics policy), followed by a set of demographic questions. The general demographic questions were:

- The respondent's self-reported age group
- Whether the respondent self-identified as a fluent reader of the sample-text language
- Whether the respondent reported having uncorrected vision

There were also three demographic questions meant to characterize the typographic experience level of the respondent on different axes:

- Whether the respondent self-identified as type designer
- Whether the respondent's work involved type or typography
- Whether the respondent had ever received formal training in type, lettering, calligraphy, or a related subject

If respondents answered yes to the question about receiving formal training, they were asked to provide a description of the training in their own words. The application server also recorded the following information for each survey response:

- The class of device used (desktop/laptop or mobile)[6]
- The language used in the sample texts (English, German, or French)

In the pilot tests and first public test battery, only English samples were presented. Additional samples in German and French were added for the second test battery. Similarly, the pilot test and first test battery featured only one layout of the samples, a multicolumn layout intended for desktop and laptop displays. A mobile-device layout using a single column was added at the start of the second test battery.

Following the demographic questionnaire, respondents were given specific task instructions. They were told that the site recorded their personal assessment of font spacing and that the fonts used in the samples may have their original fitting or have fitting that was altered, but they were not told what algorithms were employed, what characters were affected, or whether the test font on any particular page used the original fitting or an altered fitting. They were also told that their assessments of the spacing in the samples was not a test of their skill and that there were no particular correct or incorrect responses.

The instructions also asked respondents to ensure that JavaScript was active and custom font overrides were deactivated, and to check that other potential customizations were disabled before starting the survey. They were also asked to keep their browser at its default, 100% zoom level.

Respondents were instructed to mark pairs of letters, using the mouse or pointer to highlight text, that looked to them as if it had incorrect spacing for any reason, and to mark the highlighted letters as 'Too much space' or 'Not enough space' in the pop-up window that appeared once text was highlighted. *(See figure 5.8, over page)* They were not given a time limit and were told that they could take as much or as little time as they wished, and that it was acceptable to not make any marks at all if they did not want to.

After each sample exposure, respondents were shown a 'reset' page intended to provide a visual break and clear out any lingering visual impressions between samples. The reset page also provided a text-input box in which respondents could add any additional comments about the previous sample, and gave them a brief reminder of the task instructions.

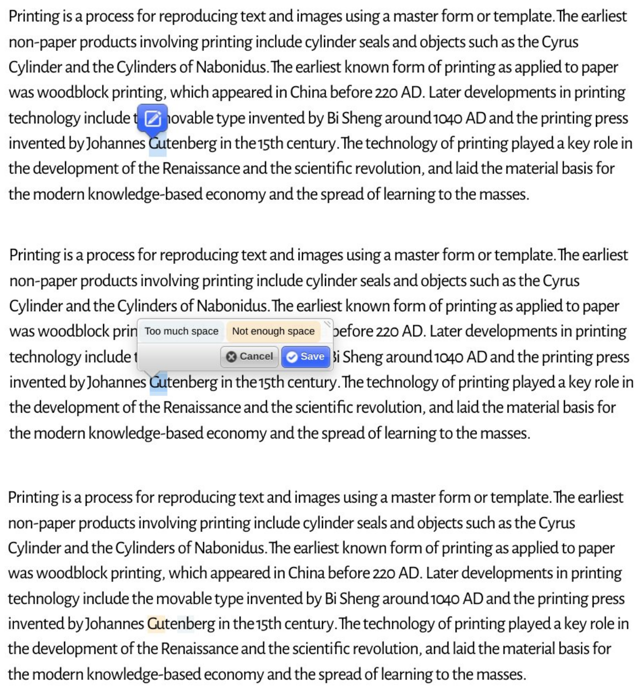
6. The application server attempted to guess the class of device based on window dimensions, but that guess was used only to pre-fill a device-class selection field on the landing page. Site visitors were able to change the selected device class before beginning the test.

Figure 5.8
Screenshots illustrating the task of marking on text samples.

Top: upon highlighting characters with the cursor, the pop-up button appears.

Middle: clicking the pop-up button presents the respondent with the two categorical choices for the mark.

Bottom: marks remain visible on the sample, using different colours for 'Too much space' and 'Not enough space' marks to let the respondent keep track of the marks previously made.



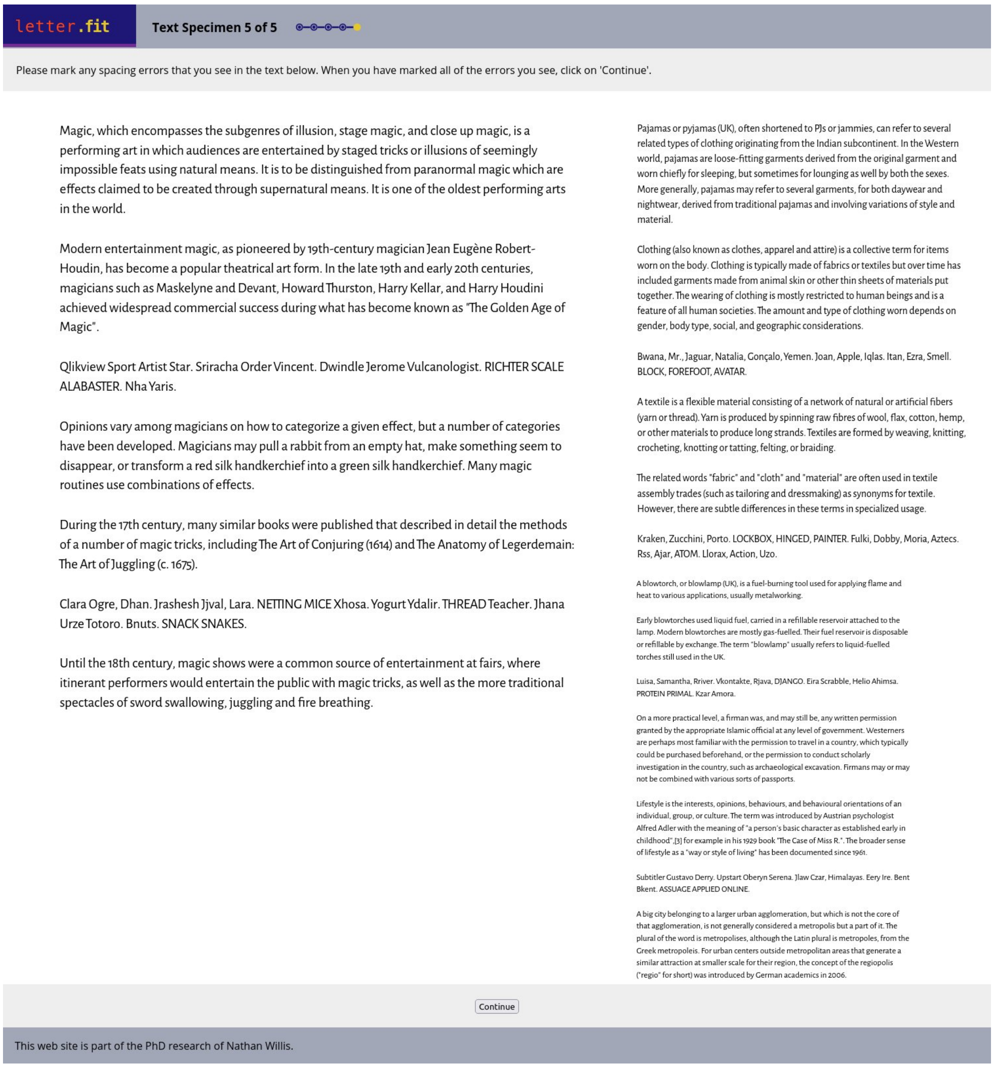
5.3.2 Procedure for recording marks on text samples

For each survey response, the application server chose five test fonts at random from among the currently active test-font pool (where each font chosen could be in any of the three test conditions) and constructed five text samples, each composed of three sample blocks chosen at random from the currently active sample-text pool in the respondent's chosen language and device class. The sample-text pool consisted of 22 sample blocks in each language (English, German, and French), with the mobile device-class blocks being abbreviated versions of the desktop device-class blocks. In the early trials, the text-sample blocks were chosen randomly with replacement, but the application server was updated to choose them without replacement out of concern that seeing duplicated text in the samples might spark confusion.

Each combination of test font and text sample shown to a respondent is here termed an *exposure*. The application server would proceed to show up to five exposures to each respondent. After the fifth exposure, the respondent was thanked, told that the survey session was complete, and provided with a set of links to the original Wikipedia pages from which the source texts were drawn.[7] The limit of five total exposures per response was put in place to prevent fatigue from setting in. This limit was established experimentally in the pilot-test phase, where respondents were able to view up to six exposures, and a marked increase in drop-outs was observed after the fifth exposure.

7. Providing links to the original pages for this source material was a requirement of Wikipedia's content license, not part of the test procedure.

Figure 5.9
Screenshot of a desktop/laptop layout text sample as seen in the public-testing version of the survey web application, showing the three sizes of text used in the three blocks.



For the desktop device class, the text samples used a two-column layout designed to average between 50 and 70 characters per line, with 16-point text occupying the first column, and the second column occupied by a block of 12-point text followed by a block of 10-point text. These font-size and layout selections were chosen to emulate a 'main body' column accompanied by a secondary 'sidebar' column, beneath which was a tertiary footnote section or page footer. *(See figure 5.9)*

The 16-, 12-, and 10-point sizes were chosen to fall within the typical size range used in online texts meant for continuous reading (Carter 1984). This restriction in size and layout involves trade-offs. The two-column design broadly resembles that of the typography that might be found in a real-world document, but only up to a point. It does not contain images or headings and subheadings, it uses the same font for each typographic element in the hierarchy, and is arguably unusual for including multi-paragraph body text at multiple sizes.

For the mobile device class, the text samples used a single-column, full-width layout, with a single paragraph set in each of three sizes: 16-point, followed by 12-point, followed by 10-point. *(See figure 5.10)* This simplified layout was meant to more closely reflect the layouts used in mobile web page designs.

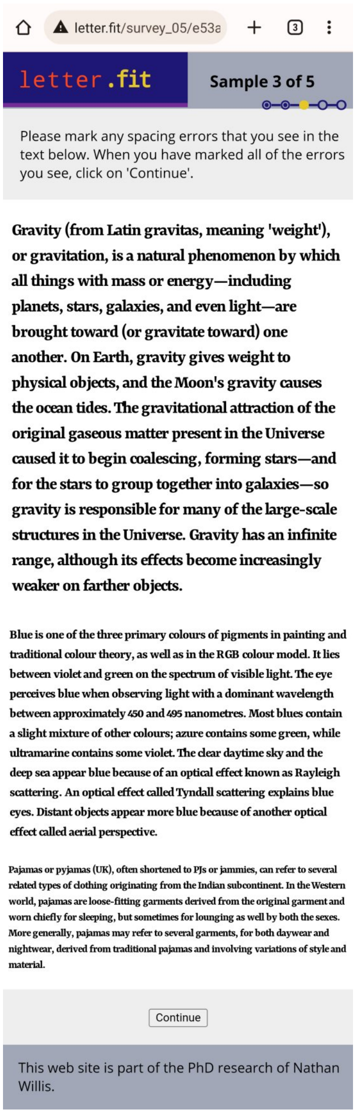


Figure 5.10
Screenshot of a mobile-device layout text sample as seen in the public-testing version of the survey web application, showing the three sizes of text used in the three blocks.

In both layouts, sample blocks used and which sample block was used for each of the three sizes was determined at random by the app server in order to vary the text presented. Randomizing the text blocks used in each sample (and which text block was used for each of the 16-, 12-, and 10-point sections of the layout) was done to provide more variety in the samples and a more even distribution of character pairs among the font sizes. If all "sample one" pages used the same text, there might be more mark data collected for the character pairs in the "sample one" text than for the pairs in "sample five."

The application server also recorded some secondary information with each sample exposure:

- The `userAgent` string
- The local start time
- The local finish time
- The sequence number of the exposure within the possible five-exposure set
- An encrypted version of the visitor's IP address

The `userAgent` strings sent by the browser were parsed into a best guess for the operating system and web browser made using the DeviceDetector Python library (Burkholder 2021). The encrypted record of the IP address was a one-way cryptographic hash of the IP address of the respondent's computer made with the Cryptolog Python library (Lee et al., 2016). This hash function is an irreversible transformation of the original IP address seen by the server, mixed with a random seed that was deleted at the end of the experiment. The hashed IP address would theoretically allow analysis of whether any respondents visited the site multiple times, but without their origin IP being discoverable.

In addition, the application server recorded the following information for each annotation mark made by the respondent:

- The characters highlighted as an error
- The error category ('Too much space' or 'Not enough space')
- The timestamp at which the mark was made
- The position of the marked characters in the paragraph
- The position of the paragraph on the sample page (i.e., the text block, which indicated point size, and paragraph number within the block)
- The height and width of the window or tab
- The zoom-level of the window or tab

The window size was added to the data set after the pilot test, during which some respondents were observed to resize their browser window for reading comfort. Some pilot-test respondents were also observed to zoom in on the text samples, despite the instructions asking them not to do so, so the zoom-level was also added to the data set. However, later testing revealed that the zoom-level of the window could not be recorded reliably across browsers, so that information was not used during the analysis stage.

5.3.3 Preparation of sample texts

The source texts for the samples were articles chosen randomly from Wikipedia pages in each of the test languages (English for the first battery; English, French, and German for the subsequent test batteries), interspersed with isolated lines of unrelated sample words selected to introduce less-common letter combinations.

In the article-selection process, it was decided to excise any randomly-chosen pages that were biographical stubs or geographic places, noting that these two page categories were frequently shorter, did not exhibit multi-paragraph stretches of text, and tended to include far more numbers (typically birth-and-death years) and family- and place-name words in their shorter paragraphs.

The source texts were further prepared for use in the samples by removing all words in other alphabets (most frequently, IPA pronunciations or etymologies from other languages), mathematical formulas and other specialist symbols, and subscripts or superscripts. Each text was also checked against a list of offensive words and, for the German and French texts, reviewed in an independent scan by a native reader to ensure that no potentially divisive subject matter was unintentionally included.

Interspersed between these Wikipedia paragraphs were single lines featuring individual words separated by commas. They included proper nouns chosen to introduce less common capital-to-lower-case letter sequences and some words set in all capital letters, to increase variety in the pairing shown across the entire exposure set. These single-line interspersions also served to separate the topically-independent Wikipedia paragraphs from each other.

5.3.4 Preparation of font files

The font files were prepared and uploaded to the application server. For the test fonts refitted by an algorithm, this process involved applying the newly calculated fittings to a copy of each font file.

The software implementation of the test algorithms, as detailed in chapter 4, output a set of sidebearings (and, optionally, a set of kerns) for each input font. To prepare the test fonts deployed on the survey site, it was decided that modifying the font binaries by patching in the new sidebearing and kerning information was preferable to attempting to build a new version of each test font from its original source files.

First, although all of the fonts eventually selected for the tests were published under an open license (see section 5.4.1), they differed in regard to what materials were included in the published releases. Some of the fonts were published only as a TrueType or OpenType binary, while others included source files. Second, in cases where source files were available, the contents varied in format and in completeness. Some of the source files are published in the *.glyphs* or in *.vfb* file formats, each of which includes the vector shapes, font features, metrics, and metadata. Others are published with vector shapes and metrics in the *.UFO* file format, plus smart-font features stored separately in the *.FEA* file format, and metadata in other ancillary file formats.

This distinction reflects the differing development and engineering processes used by the designers and foundries, which indicates the practical challenge of rebuilding the fonts from source. Different foundries use different applications and build tools, perhaps even incorporating a multi-stage (even supervised) process, and the full details of the build are not necessarily documented. Rebuilding the fonts from source without full access to the build process would risk introducing undocumented changes.

Consequently, the test fonts were modified by a Python script that read the new fitting information generated by each algorithm and applied changes directly to the binary font file, using the `ttLib` module of the FontTools Python library (FontTools 2013).

In addition, all of the original kerning features (including TrueType `kern` tables as well as any '*kern*', '*dist*', or '*cpsp*' features in the `GSUB` table) were removed from test fonts that had been refitted by algorithm. This step is necessary to produce a font file that encapsulates only the fitting generated by the algorithm. It was clear by visual examination that removing the kerning features from the refitted fonts would cause some character pairs to appear problematic, but mixing kerning adjustments made by the original designer in with sidebearings generated by an algorithm results in a fitting that does not purely represent the algorithm, potentially confusing the data analysis.

Several other smart-font features could arguably impact the perceived successfulness of a font's fitting as well, and the question of what features to preserve required careful consideration. For example, ligature substitutions may replace two-letter sequences with a single typeform representing both letters in ligated form. In English, ﬁ is perhaps the most common ligation. In ﬁ ligatures, the two component letterforms are still meant to be regarded by readers as being the two original letters, just optically corrected to accommodate for the effect of collisions. The f and i components of the ﬁ ligature may or may not touch in the ligated form;

the goal is only to correct their visual appearance. Most notably for this research project, the interior space within the fi ligature generally remains (at least, below the crossbar of the f) and should approximate the inter-letter space given to the fi pair nominally.

It could be argued that the fi ligature and similar space-enclosing ligatures convey a full inter-letter space with them that might differ noticeably from the inter-letter spaces in the surrounding text and disrupt the overall regularity of the fitting used in the test font. Removing the ligature forms, however, could cause collisions or overlaps, and redrawing the ligatures would require the author to impose design decisions. Weighing these risks, it was ultimately decided to leave the ligatures in the test fonts, as originally designed, and to note that fact when analysing response data for the letter pairs impacted.

In the final step, all of the font files deployed in the tests, including the modified font files and those that retain the original letter fitting, were anonymized with a FontTools Python script to remove the font and font-family names, manufacturer information, and other human-readable metadata from the internal font tables. This measure was taken as a precaution to preclude respondents from discovering the original font names by inspecting the HTML source of the test page or web-font files themselves, either manually or through the use of web-browser extensions, so that no preconceived opinions based on individual font or foundry names would influence the responses. The font file name was then replaced with a randomized hexadecimal identifier.

5.4 Typeface tests

Tests utilising the framework were deployed in a series of batteries on a public-facing web site, which was promoted in discussion forums, via mentions in conference talks and on social media, and by encouraging participants to share links to the test site. Deploying the tests in a series of batteries enabled changes to be made to the typeface test pool to adjust to the real-world response rate, particularly at the start of the public tests, as will be described in the following section.

The typefaces used in the tests arguably constitute the most critical portion of the test materials, because they provide the input context for the two fitting algorithms tested and are also the focus of the task that survey participants are asked to perform. The selection process involved examining typefaces on a technical level (that is, the binary font files themselves) as well as on the stylistic and typographic qualities of the typeface designs.

5.4.1 Technical criteria

The initial criteria used to select test fonts were technical in nature, largely to accommodate the realities of the refitting process and the web application server used for the survey site. For this test framework, it was decided to work with open-licensed fonts.[8] The reasons were twofold.

First, electing to work with open-licensed fonts would enable fully reproducible experiments, thus making the findings of more use to subsequent research projects. Second, initial enquiries with commercial foundries indicated that it would be costly (in both time and complexity, if not in outright monetary expense) to negotiate licensing agreements necessary to modify and deploy a large array of proprietary fonts on the public testing web site.

Within the sphere of open-licensed fonts, however, the test design identified for use typefaces that had been designed and published by established type foundries. This decision, hopefully, permitted some level of confidence that the font files had been subject to a quality-assurance process reflecting general industry-standard criteria, and that the typefaces' designs and their default fittings could be expected to fall within contemporary type design convention.

In addition to the licensing and foundry criteria, several of the selected fonts were available from their foundries in both static-font and variable-font format. In those cases, the tests were conducted only with static versions of each font. This decision was made in order to avoid any potential inconsistencies that might arise for the design-space instances in the variable-font versions.

Variable fonts are typically compiled from a set of distinct master font files, each of which are designed and tested manually. The masters are designed at pre-defined locations in the design space, often at the extreme point of each variation-axis combination (e.g., Extra Bold Extended, Extra Bold Condensed, Extra Light Extended, and Extra Light Condensed) plus a 'regular' master representing the centre. In between these masters, instances representing the intermediate characteristics somewhere along each variation axis are created by interpolating between the data in each of the masters.

Due to this interpolation, for any given instance in a variable font file, the sidebearings (as well as the contours of the forms) are interpolated values calculated when the text is rendered, and may even differ depending on the particulars of the computing environment. Thus, the sidebearings in an interpolated instance have potentially not been consciously set by the type designer. Ideally, a well-crafted variable font will have been put through a rigorous quality-control process that did involve manual evaluation of the fitting at numerous instances, but that

8. The term *open licensed* here refers to any font with a license that permits royalty-free redistribution and modification. These license rights are sometimes referred to broadly as *open source*, but source-code publication is not required by the Open Font License (OFL), the most popular license choice of the set.

cannot be guaranteed. By using the static font versions instead, there is a higher likelihood that the exact sidebearings used in the font file had been manually determined and tested.

5.4.2 Stylistic and design-space font criteria

Since the goal of this research is to explore letter-fitting algorithms that work across a broad range of typeface styles, a matrix of common Latin type styles and variation axes was developed, ranking the possible permutations by relevance and availability. *(See figure 5.11)*

The stylistic and design-space variations considered apply only to proportional typeface designs. Although letter fitting is relevant to monospace designs as well, the additional constraints of monospace fitting and the tightly-linked design restrictions it imposes make monospace typefaces out of scope for this test design.

The rankings of importance were informed by the analysis of prior work on letter-fitting automation. For example, records indicated that earlier algorithms which were successful on upright styles at regular weights and normal widths often failed at generating letter fitting for typefaces at the extremes of the weight and width variation axes, so those variation axes were rated as of higher priority. Little prior work exists in exploring the effects of optical size, but the reported experiences of *kf* in URW's *hz-program* suite indicate that it did not create a substantial problem for the algorithm (URW 1993, Zapf 1993).

Sans-serif typefaces frequently pair their upright designs with an oblique, rather than a 'true italic' slanted variant. Prior research indicated that oblique designs can be mathematically de-skewed into an intermediate, temporary upright that functions in a reasonably similar manner as the original upright design in many prior letter-fitting

Figure 5.11
Matrix of typeface styles and typographic design variations, colour-coded by their assessed relevance for testing. Green indicates higher relevance, followed by yellow, then by orange. Because pre-existing typefaces from established foundries were used, the relative uncommonness of a style or variant was considered a relevant factor in the assessment. The styles marked with * were explored for non-joining implementations.

	Optical size	Weight	Width	Slant/italic
Grotesque sans	Uncommon		Less important	Less important
Humanist sans	Uncommon		Less important	Less important
Square sans	Uncommon		Less important	Less important
Old-style serif	Less important		Uncommon	
Transitional serif	Less important		Less important	
Didone serif	Less important		Less important	
Slab serif	Less important		Uncommon	
Rounded serif	Less important		Uncommon	
* Informal calligraphic	Uncommon	Uncommon	Uncommon	Rarely available
* Casual handwriting	Uncommon	Uncommon	Uncommon	Rarely available
Blackletter	Uncommon	Uncommon	Uncommon	Rarely available
Brush lettering	Rarely available	Rarely available	Rarely available	Rarely available

algorithm experiments. Serif typefaces, however, frequently pair the upright design with a distinctly different italic design that does not respond well when de-skewed. Therefore, italic variations were rated as a higher priority for serif styles but as less important for sans serifs.

Finally, certain typeface styles (such as blackletter and handwriting faces) are rarely designed in multiple weights, widths, and optical sizes at all. Thus, while it is still valuable to test any letter-fitting algorithm on these styles, they are of lesser importance. Notably, in contemporary typography, these styles tend to be used more for decorative and display purposes, and not in running text meant for continuous reading.

The typefaces that had been used in the pilot test were chosen from a broad range of styles, as was noted in § 5.2.1. However, the results of the pilot test suggested that respondents had a substantially more difficult time assessing fitting problems in italic designs and in extreme weights. This effect is seen in a lower percentage of 'benchmark check' adjustments marked by respondents in the pilot test.

For the public testing phase, test typefaces were chosen to address the higher-relevance style and design variations from the priority matrix. The choices avoided extremes of weight and width, based on the pilot-test experience, although upright and italic forms were still initially included. The test typefaces were also selected to provide a variety of serif and sans-serif constructions, stroke contrast, and letterform constructions. This choice permitted the tests to focus on discovering patterns related to the stylistic differences with less risk of intervening effects from weight and width variance complicating the results.

5.4.3 Public test batteries

The first battery of tests featured typefaces with unaltered fitting, to collect control-group data, gauge the expected response rate, and potentially identify any issues not anticipated or observed in the controlled conditions of the pilot test. These typefaces were selected to cover a range of typographic variables, including upright and italic styles, weight and optical-size variation, and stylistic construction. In total, 29 test fonts were used in this battery. The fonts in the first test battery were:

- Abril Fatface Regular: a heavy-weight, high-contrast Didone-style serif typeface.
- Alegreya Regular, Alegreya Italic, Alegreya Sans Regular, and Alegreya Sans Italic: a family of related serif and sans-serif designs in contemporary upright and italic styles.
- Amiri Regular and Amiri Italic, and Arabic-and Latin typeface, the Latin version of which is an old-style serif typeface of moderate contrast, modelled on Garamond.[9]
- Andika Regular, a sans serif typeface using simplified letterform constructions for beginning readers.

9. The original intent of this choice was to permit the eventual testing of Arabic in addition to Latin. The Latin component of Amiri is an adaptation of the Crimson Text typeface.

- Bellefair Regular, an old-style serif design with a distinctly low x-height and short serifs.
- Gentium Plus Regular and Gentium Plus Italic, a serif typeface featuring large x-height and calligraphic terminals.
- IM Fell Double Pica Regular and IM Fell Double Pica Italic, a transitional serif design.
- Literata Regular, in three optical sizes: opsz10, opsz14, and opsz18, a modern serif design with upright stress and thin serifs.
- Neuton Regular, a Dutch-inspired serif with heavy stroke weight.
- PT Serif Regular and PT Serif Italic, a transitional serif family.
- Slabo 13px Regular, a low-contrast, slab-serif typeface developed with web usage in mind.
- Sorts Mill Goudy Regular and Sorts Mill Goudy Italic, an old-style serif family reviving Frederic Goudy's Goudy Oldstyle.
- Source Sans Pro Light, Source Sans Pro Regular, and Source Sans Pro SemiBold, a gothic sans-serif typeface in multiple weights.
- Tinos Regular and Tinos Italic, a transitional serif design with somewhat straight profiles, short serifs and angular terminals.
- Yrsa Regular and Yrsa Medium, a contemporary serif design with notably heavy serifs, available in multiple weights.

The number of responses collected in the first battery was somewhat lower than initially hoped for, considering the number of test fonts; a total of 203 sample exposures were recorded (for a mean of 7 per test font).

In the second battery of tests, the test fonts deployed differed primarily by including modified versions of the typefaces, refitted using either the composite algorithm developed in chapter 4 or the rival algorithm, which reimplemented the *kf* component of URW's *hz-program* suite. This battery reduced the number of typefaces compared to the first battery, out of concern for collecting enough responses for statistical validity for each typeface / algorithm permutation. It was also decided for the second battery to reduce the number of typographic variables, focusing primarily on the 'regular' weight and upright styles.[10] The second battery retained five of the typefaces from the first battery (Alegreya Sans, Source Sans Pro Regular, Slabo 13px Regular, Literata Regular at optical size opsz14, and Yrsa Regular) and added:

- Source Serif 4 Regular, a serif typeface designed to complement Source Sans Pro from the earlier test batteries.

In this battery, all of the test fonts were deployed in the original fitting and in versions refitted by each of the two test algorithms.

10. The second battery also coincided with the added options for respondents to view samples in German or French (in addition to English), and to select a mobile view that formatted the samples in a single column deemed more accessible for smartphones. Those changes were made on the application server and did not impact the typeface selection.

The third battery added a small set of additional typefaces chosen to extend the pool of test fonts with additional typographic weights, widths, and optical sizing, complementing the typefaces deployed with battery two. Each additional test font in this battery, as in battery two, was deployed in the original fitting and in versions refitted by each of the two test algorithms. The third battery included:

- STIX Two Text Regular, a high-contrast transitional serif design with a vertical stress axis.
- Slabo 27px Regular, a slab-serif typeface designed as an optical-size variant of Slabo 13px Regular from the earlier test batteries.[11]
- Fira Sans Condensed, a condensed sans serif chosen to test in a width variation distinct from 'normal' width.[12]
- Yrsa Bold, a bold-weight variant of Yrsa Regular from the earlier test batteries.[13]

The full set of typefaces used in the tests, along with sample typeforms for each, is provided in table 5.1 (over page), grouped by their deployment in the test batteries.

To some extent, the oversized selection in the first test battery could be considered a missed opportunity, because with a smaller set of test fonts, more exposures might have been collected for each. But accurately assessing the real-world response rate of the survey was a valuable stage, and allowed the later test batteries to be adapted accordingly. Furthermore, the exposures of the test fonts that were used only in the first battery still form an important part of the control group, and help further the analysis of readers' responses that followed.

That analysis, which is the focus of the next chapter, evaluated the marks made by survey respondents for patterns, both within each tested typeface, and between the profile shapes of the typeforms.

11. Slabo 27px was chosen for its potential to capture optical-size differences from Slabo 13px. This pairing was selected over the alternative option of adding a Literata optical-size variant because the slab-serif design of the Slabo faces is more distinctive within the test pool than Literata's serif design. Slabo also exhibits the rare design property of changing the construction of g between the 13px and 27px sizes.

12. The sans serif designs from the earlier batteries did not exist in a condensed-width variant.

13. Yrsa Bold was chosen over Yrsa Medium in order to test a more extreme weight variant.

Table 5.1
Typefaces deployed in the test batteries, listed sequentially and with key letterforms shown.

	Typeface	Sample forms
Pilot test	Cantarell Regular	handglovestucz VALENCYXJOSH
	Fira Sans Extra Condensed	handglovestucz VALENCYXJOSH
	Alfa Slab One Regular	handglovestucz VALENCYXJOSH
	Libre Caslon Text Italic	*handglovestucz VALENCYXJOSH*
	Rajdhani Light	handglovestucz VALENCYXJOSH
	Tenor Sans Regular	handglovestucz VALENCYXJOSH
Battery 1 (only)	Abril Fatface Regular	handglovestucz VALENCYXJOSH
	Alegreya Regular	handglovestucz VALENCYXJOSH
	Alegreya Italic	*handglovestucz VALENCYXJOSH*
	Alegreya Sans Italic	*handglovestucz VALENCYXJOSH*
	Amiri Regular	handglovestucz VALENCYXJOSH
	Amiri Italic	*handglovestucz VALENCYXJOSH*
	Andika Regular	handglovestucz VALENCYXJOSH
	Bellefair Regular	handglovestucz VALENCYXJOSH
	Gentium Plus Regular	handglovestucz VALENCYXJOSH
	Gentium Plus Italic	*handglovestucz VALENCYXJOSH*
	IM Fell Double Pica Regular	handglovestucz VALENCYXJOSH
	IM Fell Double Pica Italic	*handglovestucz VALENCYXJOSH*
	Literata Regular optical size 10	handglovestucz VALENCYXJOSH
	Literata Regular optical size 18	handglovestucz VALENCYXJOSH
	Neuton Regular	handglovestucz VALENCYXJOSH
	PT Serif Regular	handglovestucz VALENCYXJOSH
	PT Serif Italic	*handglovestucz VALENCYXJOSH*
	Sorts Mill Goudy Regular	handglovestucz VALENCYXJOSH
	Sorts Mill Goudy Italic	*handglovestucz VALENCYXJOSH*
	Source Sans Pro Light	handglovestucz VALENCYXJOSH
	Source Sans Pro SemiBold	handglovestucz VALENCYXJOSH
	Tinos Regular	handglovestucz VALENCYXJOSH
	Tinos Italic	*handglovestucz VALENCYXJOSH*
	Yrsa Medium	handglovestucz VALENCYXJOSH
Battery 1 onward	Alegreya Sans Regular	handglovestucz VALENCYXJOSH
	Source Sans Pro Regular	handglovestucz VALENCYXJOSH
	Slabo 13px Regular	handglovestucz VALENCYXJOSH
	Literata Regular optical size 14	handglovestucz VALENCYXJOSH
	Yrsa Regular	handglovestucz VALENCYXJOSH
Battery 2 onward	Source Serif 4 Regular	handglovestucz VALENCYXJOSH
Battery 3 onward	STIX Two Text Regular	handglovestucz VALENCYXJOSH
	Slabo 27px Regular	handglovestucz VALENCYXJOSH
	Fira Sans Condensed	handglovestucz VALENCYXJOSH
	Yrsa Bold	handglovestucz VALENCYXJOSH

6. Findings from quantitative tests

The testing methodology outlined in chapter 5 was designed to measure the responses of readers to the fitting of Latin text when viewed on the web, with the ultimate goal of measuring the success of fitting algorithms among readers. As was described in that chapter, the raw data collected in readers' responses is where each reader chose to mark a sequence of letterforms on a sample text as looking like the fitting is incorrect, plus an indicator of whether they feel that the fitting of the marked forms is too tight or too loose. The data from these marks thus encapsulates the exact typeforms marked, the typeface used in the sample, the fitting algorithm employed to fit the typeface, and, by extension, any influences brought on by the sample itself (such as its language) or by the reader. Analysing this raw data to measure the responses of readers to the fitting algorithms in a useful and statistically valid manner necessitates some processing.

This chapter will detail how the raw data was analysed to define metrics appropriate to assessing fitting algorithms, present findings based on those metrics, and offer an interpretation of how the findings relate to the test algorithms used. The development of these metrics was a process that required careful consideration not just of the specifics of the testing framework, but also of the research problem and the Latin text fitting model, as all three intersect in the data. Before exploring the data itself, the discussion will begin with a summary of the testing and its overall participation, in order to establish a solid understanding of the conditions in which the data was collected.

6.1 Test batteries and overall participation

As described in chapter 5, tests were deployed in a series of batteries on a public-facing web site, which was promoted in discussion forums, via mentions in conference talks and on social media, and through encouraging participants to share links to the test site. Deploying the tests in a series of batteries enabled changes to be made to the typeface test pool to accommodate the real-world response rate.

The first battery of tests featured typefaces with unaltered fitting, to collect control-group data, gauge the expected response rate, and potentially identify any issues not foreseen during the pilot-testing phase. In total, 29 test fonts were used in this battery; a total of 203 sample exposures were recorded, a mean of 7 exposures per test font.

The second and third batteries of tests included modified versions of typefaces, refitted using either the composite algorithm developed in chapter 4 or the rival algorithm, which reimplemented the *kf* component of URW's *hz-program* suite, in addition to the unaltered versions of each typeface. The total counts of sample exposures recorded during the tests is summarized in table 6.1 *(over page)*.

Table 6.1
The number of exposures recorded for each test typeface, in each fitting-algorithm test condition."'Control' refers to the original, unaltered fitting. Typefaces marked with an asterisk (*) were used in the pilot test only. Pilot-test responses were included in the general-response and demographic analysis.

Typeface	Control	Composite	*kf*	Sample forms
* Cantarell Regular	31			handglovestucz VALENCYXJOSH
* Fira Sans Extra Condensed	29			handglovestucz VALENCYXJOSH
* Alfa Slab One Regular	26			handglovestucz VALENCYXJOSH
* Libre Caslon Text Italic	22			*handglovestucz VALENCYXJOSH*
* Rajdhani Light	20			handglovestucz VALENCYXJOSH
* Tenor Sans Regular	18			handglovestucz VALENCYXJOSH
Abril Fatface Regular	6			handglovestucz VALENCYXJOSH
Alegreya Regular	8			handglovestucz VALENCYXJOSH
Alegreya Italic	10			*handglovestucz VALENCYXJOSH*
Alegreya Sans Italic	3			*handglovestucz VALENCYXJOSH*
Amiri Regular	5			handglovestucz VALENCYXJOSH
Amiri Italic	9			*handglovestucz VALENCYXJOSH*
Andika Regular	6			handglovestucz VALENCYXJOSH
Bellefair Regular	6			handglovestucz VALENCYXJOSH
Gentium Plus Regular	9			handglovestucz VALENCYXJOSH
Gentium Plus Italic	2			*handglovestucz VALENCYXJOSH*
IM Fell Double Pica Regular	5			handglovestucz VALENCYXJOSH
IM Fell Double Pica Italic	11			*handglovestucz VALENCYXJOSH*
Literata Regular optical size 10	11			handglovestucz VALENCYXJOSH
Literata Regular optical size 18	7			handglovestucz VALENCYXJOSH
Neuton Regular	4			handglovestucz VALENCYXJOSH
PT Serif Regular	6			handglovestucz VALENCYXJOSH
PT Serif Italic	10			*handglovestucz VALENCYXJOSH*
Sorts Mill Goudy Regular	8			handglovestucz VALENCYXJOSH
Sorts Mill Goudy Italic	7			*handglovestucz VALENCYXJOSH*
Source Sans Pro Light	4			handglovestucz VALENCYXJOSH
Source Sans Pro SemiBold	5			handglovestucz VALENCYXJOSH
Tinos Regular	7			handglovestucz VALENCYXJOSH
Tinos Italic	5			*handglovestucz VALENCYXJOSH*
Yrsa Medium	10			handglovestucz VALENCYXJOSH
Alegreya Sans Regular	21	32	34	handglovestucz VALENCYXJOSH
Source Sans Pro Regular	18		25	handglovestucz VALENCYXJOSH
Slabo 13px Regular	7	35	29	handglovestucz VALENCYXJOSH
Literata Regular optical size 14	20	38	26	handglovestucz VALENCYXJOSH
Yrsa Regular	20	26	27	handglovestucz VALENCYXJOSH
Source Serif 4 Regular	29	24	24	handglovestucz VALENCYXJOSH
STIX Two Text Regular	25	26	21[1]	handglovestucz VALENCYXJOSH
Slabo 27px Regular	18	17	18	handglovestucz VALENCYXJOSH
Fira Sans Condensed	16	20	10	handglovestucz VALENCYXJOSH
Yrsa Bold	8	14	11	handglovestucz VALENCYXJOSH

1. Early in the testing, a version of this typeface that had been refitted using a flawed version of the *kf* algorithm, which did not correctly set the minimum sidebearings for diagonal profiles, was accidentally exposed on the test site and received 16 exposures. Marks from those exposures were included in the general response statistics, such as the breakdowns by demographic group, but the marks were removed and not included in the head-to-head comparisons of algorithms.

In total, nine of the typefaces had exposures recorded in all three of the test conditions (control group and both fitting algorithms). One additional typeface, Source Sans Pro Regular, received exposures in the version refitted by the rival *kf* algorithm, but none in the version refitted by the composite algorithm. Although regrettable, it is believed that this is an effect of the randomized selection of test fonts shown to each respondent.

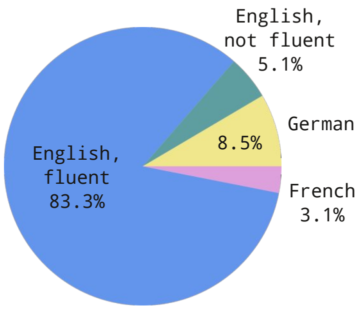


Figure 6.1
Language and fluency reported by respondents.

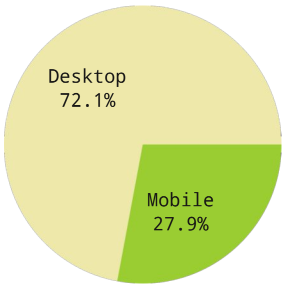


Figure 6.2
Device classes of respondents.

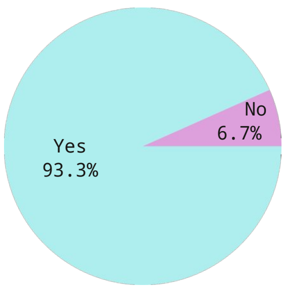


Figure 6.3
Vision status reported of respondents. 'Yes' indicates normal or corrected-to normal vision.

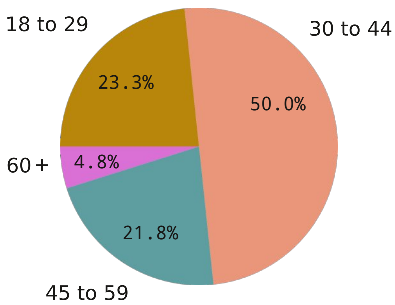


Figure 6.4
Age ranges reported of respondents.

6.1.1 General response statistics and demographics

Across all of the tests, there were 390 respondents who viewed at least one text sample. 1035 exposures were viewed in total, out of which 611 exposures received at least one mark on a sequence of characters. This resulted in 8320 marked sequences (a mean of 8 marks per exposure). Among the exposures receiving at least one mark, the mean was 13.6 marks per exposure.

Beginning with battery two, the testing site began by asking the respondents to select the language they wished to view samples in and whether they wished to view a desktop/laptop (multi-column) layout identical to the layout used in the first battery and pilot test, or a mobile device layout that used a single column and shorter overall sample paragraphs. Counting all responses before battery two as English-language and desktop-layout (which was the only configuration available), 345 respondents (88.5%) who viewed a sample exposure viewed samples in English, 33 (8%) viewed samples in German, and 12 (3%) viewed samples in French. *(See figure 6.1)* In total, 281 of the respondents (72%) used the desktop/laptop multi-column layout. *(See figure 6.2)*

The basic demographic questions asked each respondent about their language fluency with the language selected for sample text exposures, normal or corrected eyesight, and general age range. Of the 345 English-sample respondents, 20 said they considered themselves not fluent readers; all respondents who viewed German and French samples said they considered themselves fluent in those languages. 364 of the respondents (93%) reported that they either had normal vision or their vision was corrected to normal. *(See figure 6.3)*

The age-range question was limited by the requirement that participants in the tests be 18 or older. Of the respondents, 91 (23%) said they were 18 and 29 years of age, 195 (49%) said they were from 30 and 44 years of age, 85 (22%) said they were from 45 to 59 years of age, and 19 (5%) said they were age 60 or older. *(See figure 6.4)*

6.1.2 Experience with type and typography in the response set

An early concern that arose during the design of the testing framework was whether it would be possible to recruit volunteer participants of varied backgrounds while sufficiently representing relevant typographic experience levels, and how that factor might in turn impact the resulting data. In particular, a potential critique of the methodology was that an anonymous, public-facing web site would attract a disproportionately large number of respondents with little or no typographic or typeface-design experience, although how the proportion of experienced respondents could affect the results is not known. To measure that factor, the questionnaire section asked about three conditions:

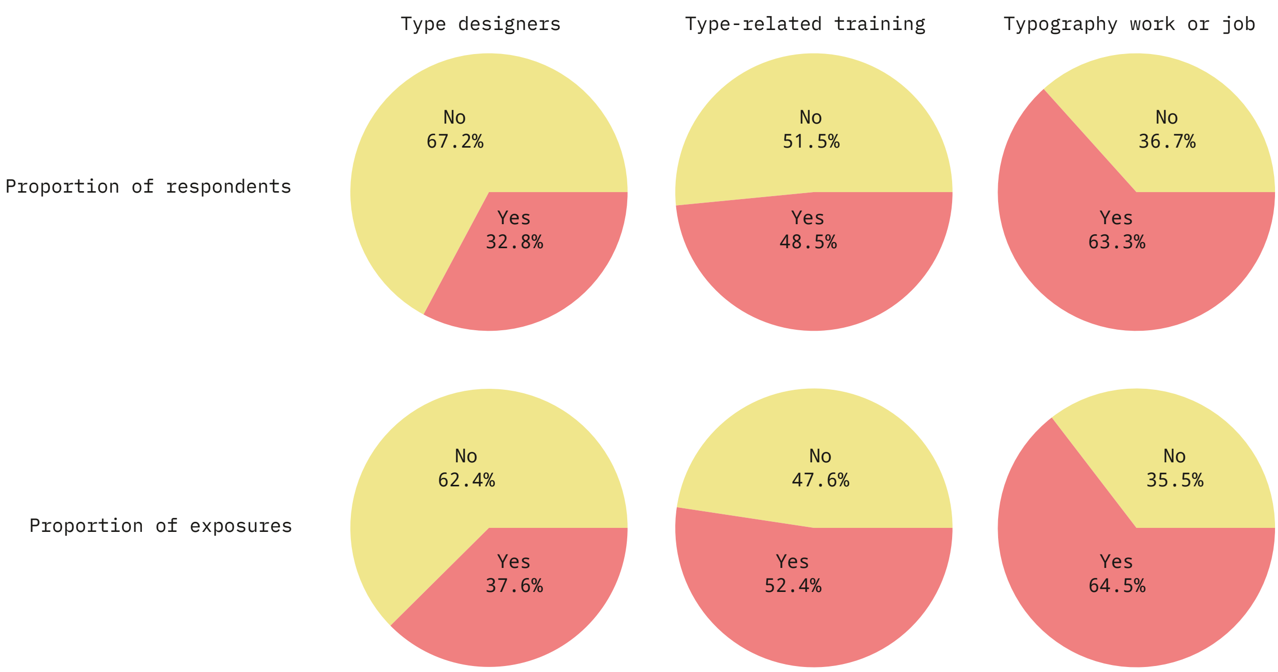
- Whether the respondent self-identified as type designer
- Whether the respondent's work involved type or typography
- Whether the respondent had ever received formal training in type, lettering, calligraphy, or a related subject.

The questions were worded to examine various dimensions in which a respondent might be considered experienced. On these questions, 128 respondents (33%) described themself as a type designer, 247 (63%) said their work involved type or typography, and 189 (49%) said that they had received formal training in type or one of the listed related subjects. *(See figure 6.5)*

Respondents who reported having received formal training were asked to describe it in their own words. 163 provided a description. Of those:

- 72 mentioned a specific college or post-graduate degree
- 38 mentioned college or school courses
- 6 mentioned non-academic study courses or workshops
- 6 mentioned internships or formal on-the-job training
- 29 mentioned either being self-taught or referenced their own work experience (other than on-the-job training)

Figure 6.5
Typographic experience reported by respondents, by respondent and by exposures viewed. Respondents were each asked three questions related to their experience level with type and typography.



Some respondents mentioned items in multiple categories; some were difficult to classify or interpret. The above classifications are meant to be descriptive only. Across the three questions, it is believed that the sample contained a satisfactory mix of respondents with different degrees and varieties of typographic experience.

6.2 Exposure and mark data

The preceding look at the responses can provide perspective into the sample of respondents, but the core data collected is the marks of 'Too much space' and 'Not enough space' made by the respondents on specific letterforms seen in the exposures. Here, too, some general observations can be made about the overall characteristics of the responses that assist the analysis.

6.2.1 General characteristics of the exposure set

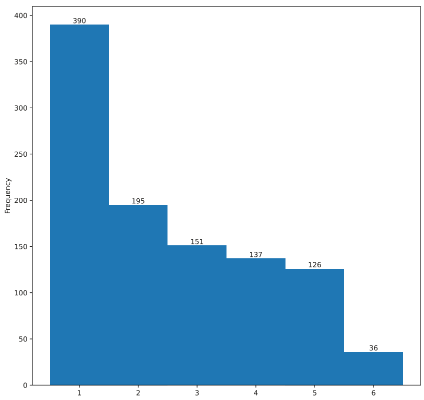


Figure 6.6
Respondent drop-out rate as visible in the number of exposures viewed. The sixth exposures reported in the figure are from the pilot-testing phase, which permitted six total exposures.

Two high-level patterns of potential interest were observed in the set of exposures. The first is the dropout rate. Of 390 respondents that viewed one exposure, 216 (55%) proceeded to view additional exposures, with 147 eventually viewing all five exposures available during a session. *(See figure 6.6)* What percentage of the respondents who viewed an exposure but stopped the survey without attempting to assess the sample cannot be reliably determined.

The second pattern observed in the exposure set relates to the question of typographic experience level raised in the previous section. All three experience-level groupings accounted for a larger proportion of exposures viewed than their corresponding proportion in the respondent set. Although respondents who identified themselves as type designers constituted 33% of the respondents, they viewed 389 (38%) of the total exposures. Respondents who said they had received training viewed 542 (52%) of the exposures (from 49% of the respondents), and respondents who said their work involved type or typography viewed 668 (65%) of the exposures, (from 63% of the respondents). This may reflect a higher degree of motivation among the typographic-experience group to complete the task. *(See again figure 6.5)*

6.2.2 General characteristics of the mark set

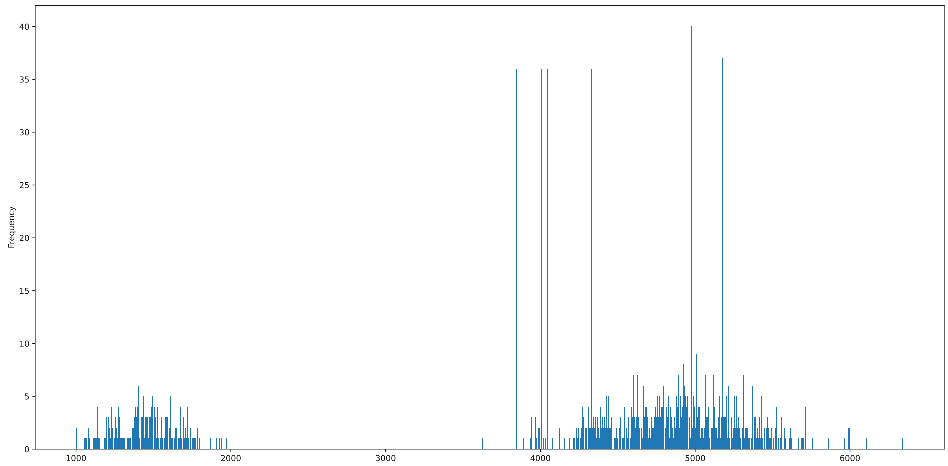
Regarding the marks recorded on the exposures, the text contents of each exposure was randomized to reduce the chances of marks being skewed towards certain letterforms based on their position in the sample page and to avoid confusion caused by showing respondents the same text multiple times.[2] As a result, the exposures varied in length and in the set of letter-pairs they contained. The desktop/laptop multi-column layout was

2. Randomization of sample texts: as was discussed in chapter five, § 5.3.3.

Figure 6.7

Lengths of exposures viewed by respondents in the collected data set, in characters. Because the method involved randomizing the sample pages, the exposures varied in length. The two clusters clearly distinguish the longer, mutli-column layout shown to desktop/laptop devices from the shorter single-column layout shown to mobile devices.

The six spikes correspond to the pilot-test response sessions, which used a non-randomized set of sample texts and thus did not vary in length.



distinctly longer, with an average of 4757 characters (including all typeforms and word spaces) among the exposures viewed by respondents. The mobile, single-column layout was abbreviated, with an average of 1447 characters (including all typeforms and word spaces). *(See figure 6.7)*

The variation in lengths between exposures is perhaps most important when considering how the marks made are used to estimate respondents' level of satisfaction with the fitting in the exposure: 'marks per exposure' alone is not a sufficient metric on which to compare algorithms, due to the varying lengths of the exposures. A 'marks per thousand characters' rate might be a better first approximation although, as will be seen in § 6.3, an elementary mark count (even when normalized by the number of characters) is not sufficiently detailed to be the basis for evaluating fitting algorithms.

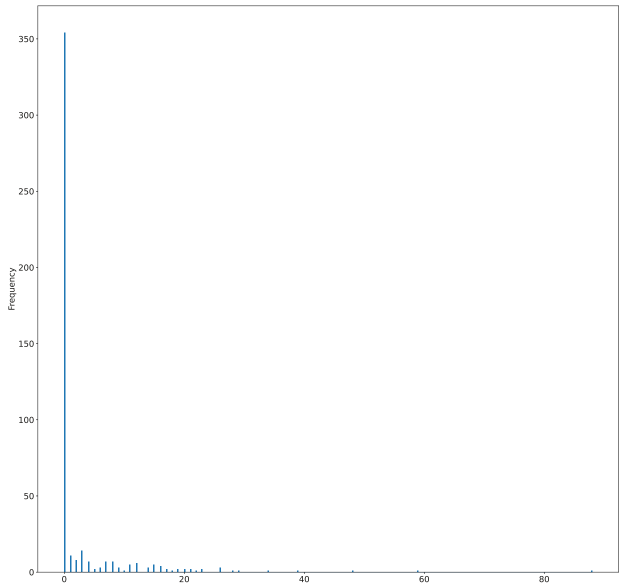
Nevertheless, it is useful to observe the overall rates at which exposures were marked, simply to note the level at which any signals in the data may be found. Across all exposures, there were an average of 4.2 marks per thousand characters (SD = 27.7); 7.1 marks per thousand characters (SD = 35) if exposures with zero marks are excluded. *(See figure 6.8, over page)*

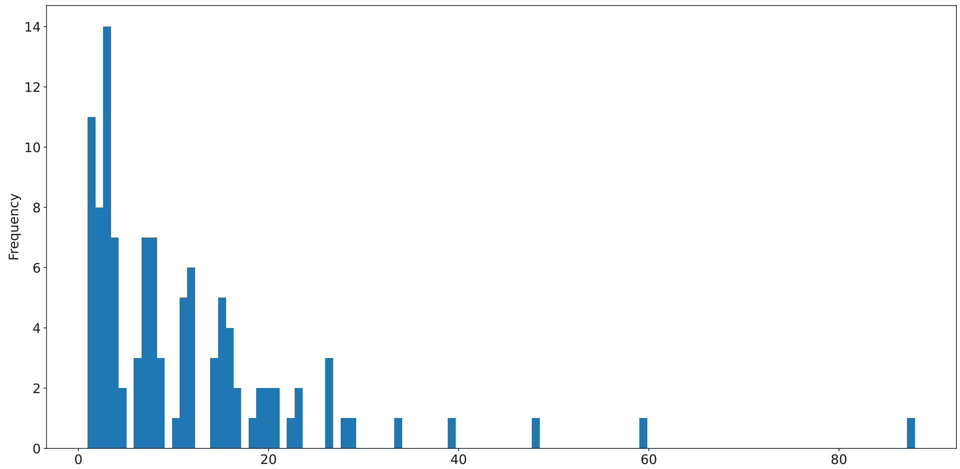
These unadjusted rates do not take into account whether the respondents marked pairs of forms on the samples, as the instructions asked them to do. Even in the pilot test, some respondents were observed to highlight and mark entire words or phrases, and the free-form nature of the test could not prevent this. Across all of the test batteries, the majority of the 8320 marks made were, as instructed, pairs of exactly two characters ('characters' here including word spaces in addition to typeforms). Of the remaining marks, 126 were less than two characters long,[3] 926 were three-character marks, 363 were four-character marks, and so on, tapering off with mark-length, with a small number of marks persisting in the count up to 33 and a few outliers beyond that range. *(See figure 6.9, over page)* Four marks (all made in the pilot test) were more than 300 characters in length; whether this is due to error, to the respondent intentionally registering

3. 121 of these marks were one-character long. 5 of the marks were zero characters in length. The situation causing zero-length marks is unclear, but is most likely the result of users attempting to select text outside the sample area.

Figure 6.8
Right: The number of marked sequences made per thousand characters in each exposure. A large number of exposures received zero marks, perhaps in some cases due to drop-outs.

Below right: The second chart removes the zero-mark column to more easily see the other columns.





dissatisfaction with an entire paragraph, or to some other cause could not be determined.

It was decided to include the greater-than-length-two marks in the data analysis on the grounds that they represent a respondent's intentional reaction to the fitting of the marked section of the sample and each such mark could be unambiguously be split into constituent pairs (e.g., an abc sequence was split into two pairs: ab and bc). Marks of less than length two were dropped from the analysis because they could not be unambiguously associated with a pair. When these subdivided marks were added to the length-two mark set, there were a total of 18583 marked pairs. 17721 of these marked pairs were on the desktop/laptop layout (95%).

Each mark made was tagged by the respondent according to whether it showed 'Too much space' or 'Not enough space'. Across all three test conditions, the 'Not enough space' marks accounted for 14028 of the marked pairs (75%). A similar ratio was observed when looking only at the control group of typefaces using their original, unaltered fitting: 8541 out of 11298 marked pairs (75%) made on control-group exposures were 'Not enough space' marks. Each exposure also showed sample text in three sizes.

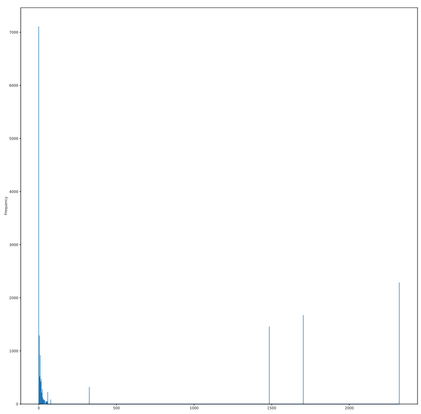


Figure 6.9
Length in characters of the sequences marked by all respondents. Note that this chart plots the original length of each sequence as it was marked by the respondent; marks of longer than two characters were split into pairs before the data set was analysed.

The 16-point text blocks received the most marks (of both mark categories) at 13662 (74%), followed by the 12 point at 3247 (17%), and 10 point at 1674 (9%).

6.3 Defining metrics to evaluate fitting from the mark data

Fundamentally, the testing framework used in this research was developed to measure each respondent's level of dissatisfaction of the fitting seen in each exposure. The expectation is that analysing many such results from many respondents for a particular algorithm provides a valid estimate for how successful the algorithm would be received by readers outside of the testing framework.

The first step in this process was determining how the marks made on pairs of typeforms in the exposures are best integrated into a meaningful evaluator of an algorithm. It is simple enough to split the data collected into three sets of exposures, one for each of the test conditions (the control group with the original fitting in each font, the group of fonts refitted by the composite algorithm, and the group of fonts refitted by the rival *kf* algorithm). Within each of those test conditions, each of the raw marks comprise a pair of typeforms from one of the exposures plus the respondent's tag indicating whether the pair exhibited 'Too much space' or 'Not enough space' (from here, the terms 'loose' and 'tight', respectively, will be used for the sake of brevity). Developing a meaningful evaluator began with analysing those marks.

6.3.1 Per-pair metrics: exposure mark rates

As discussed in chapter 2, the Latin fitting process focuses on determining left and right sidebearings for each typeform. It has been assumed in this analysis that when a respondent marks a pair ab in an exposure as having unacceptable fitting, that response correlates to a potential problem with either the right sidebearing of a, the left sidebearing of b, or with both. In accordance with this principle, the marked ab pairs are interpreted as marks on the interior, facing profiles in between the forms (the right profile of a and the left profile of b), and not as marks on the external profiles. Counting these marks across all of the pair permutations in the mark set, if there are more marks on the right profile of a than marks on the left profile of b, then the right profile of a has scored worse under that test condition, allowing the right profile of a to be identified as the more likely location of the problem (although the marks on both profiles must still be counted; both profiles in a pair could exhibit a problem, or the problem could be unique to the pair).

This basic relationship is complicated by the fact that each mark is tagged with the 'loose' or 'tight' designator, which indicate opposing problems. In aggregate, an equal number of 'loose' and 'tight' marks for a particular profile were treated as balancing each other out. This would certainly be the natural interpretation for a marked pair in a single exposure: if the same respondent marked ab as overly tight twice and overly loose twice, then perhaps *some* visual problem does exists with one or both profiles in the pair, but it cannot be decisively attributed to a tight-or-loose fitting issue. Applying this same logic across multiple respondents and exposures generalizes to a degree, but it remains in line with the overall assessment methodology: if equal numbers of respondents independently mark the pair ab as being overly loose and as being overly tight in a particular test condition, then no conclusion should be drawn about whether the pair is fitted too loose or too close by that test condition's fitting algorithm.

To aggregate the marks across a given subset of exposures (for example, across all exposures of a typeface within a given test condition), the number of exposures where a mark was made for each pair was counted, and each such count was divided by the number of exposures in which the pair occurred in the sample text. This was done to normalize the counts for more frequently-occurring pairs of forms receiving more marks. This results in a fractional value between 0 and 1 representing what proportion of exposures containing a given pair received a mark for that pair.[4]

This ratio was calculated separately for the 'loose' and 'tight' marks. A heatmap matrix can be constructed to visualize either ratio for a given subset of exposures and potentially identify patterns of interest. *(See an example for one typeface at figure 6.10, over page)* Even at this early stage, some patterns are discernible from visual inspection, by structuring the heatmap matrices to group the typeforms by class (namely, lowercase letterforms, capital letterforms, numerals, and punctuation & other symbols). With both test algorithms, there are noticeably more occurrences of high 'tight' mark ratios in the capital-to-capital segments of the heatmaps. This pattern was expected based on the construction of the test algorithms, because both test algorithms fitted the capitals for capital-to-lowercase text usage and did not implement a separate set of capital-to-capital fittings via a kerning feature.

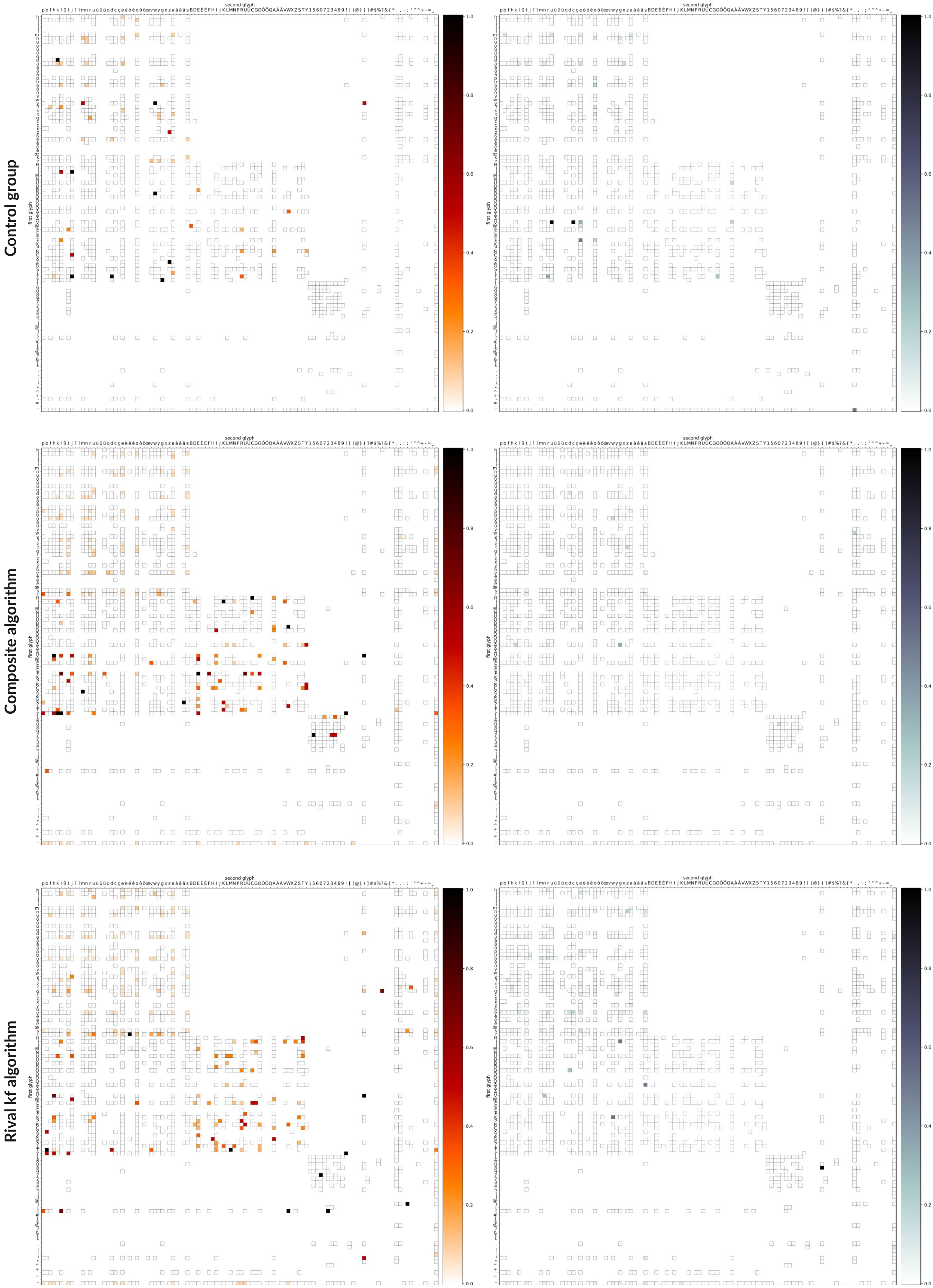
It was also observed that the numeral-to-numeral segments of the heatmaps exhibited more of high 'tight' mark ratios in both test algorithms. The construction and the practical implementations of the test algorithms did not specifically predict this pattern, but it is perhaps interesting to observe in reference to the theoretical discussions about how numerals are fitted.[5] The test algorithms fitted the numerals for setting with the lowercase letterforms, using the same method; as Noordzij and others have noted, however, numerals serve a different function than

4. Other methods of aggregating the marks across exposures were explored. Of note, the method chosen, counting all *exposures* in which the pair was marked, rather than the number of separate marks made in those exposures, guards against undue influence by respondents who marked longer-than-two-character sequences. For example, if one respondent marked every occurrence of an exceptionally common pair (such as te), then that single response could outweigh multiple other respondents who chose to mark each pair only once.

5. See chapter 2.

Figure 6.10

Heatmap matrices of mark ratios for Yrsa Regular in the three test conditions. Left column is ‘tight’ mark ratios; right column is ‘loose’ mark ratios. The rows of each matrix index the first character in each mark pair; the columns index the second character. Forms are sorted by case and profile shape (right-side profiles on row indices; left-side profiles on column indices).



do individual letterforms. The rules of fitting numerals perhaps warrants investigation.

The next step consolidated the 'tight' mark ratio and 'loose' mark ratio for each of the pairs into a single signal representing the balance of the two. The result is a measurement of whether the overall perception of the pair skewed towards 'tight' or 'loose'. As was the case when deciding how to reconcile instances of a single respondent marking the same pair of forms in conflicting fashion, a small amount of information is lost, but it preserves the overall response to the pair across all of the exposures in the set. This consolidation subtracted the 'tight' mark ratio from 'loose' mark ratio, which results in a value between -1 and 1, with negative numbers representing overly 'tight' and positive numbers representing overly 'loose'. *(See figure 6.11)*

This metric will be referred to as the *exposure mark rate* and forms the basis for the following analysis. Here, and in the following analysis, it should be noted that the exposure mark rate measures the aggregate response as to whether a pair is too 'tight' or too 'loose', but it does not

Figure 6.11

When a profile receives some 'tight' and some 'loose' marks (left chart), subtracting the ratios reveals the balance between the two (right chart). Shown here are the left-profile rates for lowercase letterforms in Fira Sans Condensed, from the control-group exposures.

It should be noted that these ratios are computed separately for the left and right profile of each form.

The order of the forms visible in the index is an automatic sort on the basis of the left profile shape, which has the benefit of grouping similarly-shaped forms together in the chart.

measure how 'tight' or how 'loose' the pair is. That is, if ab features an exposure mark rate of 0.2 and bc an exposure mark rate of 0.4, this indicates that there was more agreement that bc was fitted too loosely in the samples, but it does not mean that the sidebearings of bc in the samples are twice as far from correct as the sidebearings of ab.

6.3.2 Assessing results across typeforms and profiles

As defined, the exposure mark rate for a pair of forms represents the aggregate response across the given subset of exposures for that pair of forms, and it can be calculated for every pair occurring in the exposure subset. Across all of the sample texts used in the test (and seen in the heatmap matrices), there were a total of 110 different characters, giving 12100 possible pairs and 12100 possible exposure mark rates — although not every pair occurred in the sample text.

Several approaches were considered for how to convert those individual exposure mark rates into an assessment of the overall response to the given subset of exposures. In a blunt approach, one could simply take the mean or median of all of the exposure mark rates for every exposure in a particular test condition and arrive at a number, but that number would reveal little in the way of useful insight. Furthermore, because the typefaces tested were chosen to represent a number of distinct typographic design styles, consolidating them into a single group would risk obscuring differences by typographic variables. For this research, it was decided that the most informative approach was to analyse each of the tested typefaces independently. Within each, it was decided to cluster the letterforms by their profile shapes, and test for statistical differences between algorithms on a per-profile-shape basis. There are two benefits to analysing forms by profile-shape clusters.

First, as was discussed in chapter 2, the fitting process for Latin text fonts incorporates a number of distinct axioms that are applied to groups of similarly-shaped profiles. Consequently, similarly shaped profiles are expected to have similar fitting — both by typeface designers when performing fitting (as per axioms L–1 and L–2) and by readers when encountering text. Thus, although an analysis at the per-letterform level might discover atypical cases (such as the left sidebearing of c, e, and o differing substantially), those incidents would be expected to be rare and likely indicate errors rather than revealing a meaningful pattern.

Second, clustering the letterforms by profile shapes leverages knowledge about how the test algorithms affect similar profiles. Both of the test algorithms that were implemented deliberately apply similar fitting to similar profile shapes, and the composite algorithm developed in chapter 4 incorporated a distinct method to fitting open-counter profiles as well as several tunable parameters, such as the ratio between the internal space of n and the standard inter-letter area, or minimum-distance parameter for diagonal profiles. Thus, clustering the mark error rates by profile shape could provide insight into whether or not the distinct method employed for open-counter profiles or any of the tunable parameters may correlate to a different acceptance response from those of other profile shapes, perhaps informing future development. In a case where the details of the fitting algorithm were not known, this profile-shape clustering step might be less applicable.

Using this approach, it was possible to analyse each of the nine typefaces that received exposures in all three test conditions via a substantially smaller batch of computations and statistical tests (and to increase the reliability of the metrics by analysing larger data sets), while still retaining connections to the methods used to develop the test algorithms and, ultimately, to the fitting processes employed manually by type designers working with Latin text typefaces.

The final consideration was determining what comparisons between the three test conditions were of interest to the research questions of this project. It was decided that pairwise comparisons would provide the most useful conclusions. Specifically, it was of interest whether there were profile shapes in any of the test typefaces for which the composite algorithm performed better, worse, or similarly to the original fitting from the control group, or (separately) for which the composite algorithm performed better, worse, or similarly to the rival *kf* algorithm. Comparisons between the original fitting and the rival *kf* algorithm could also prove useful, as an assessment of the *kf* algorithm conducted independently of URW's internal tests, apart from the interest in the composite algorithm as a product of this research project. As mentioned above, it was of special interest how the composite algorithm performed with open-counter profiles, and investigating that question necessitates looking at all of the profile shapes.

For these comparisons, it was decided to focus on the lowercase letterforms for a number of practical reasons. First, the lowercase-to-lowercase fitting overwhelmingly dominates Latin text set for continuous reading — even in German, which utilises more capital-to-lowercase pairings than French and English. Second, with nine typefaces receiving exposures in three test conditions, it was deemed important to limit the total number of comparisons tested. Third, but related to the second reason, there are unresolved questions encountered when determining how to categorise the profile shapes of capital forms fitted to lowercase forms. For example, in pairings such as Co, it could be argued that clipping the C at the x-height, as would be done for measuring inter-letter area between the baseline and x-height, makes its right-side profile unbounded, like L. But this depends on the openness or closedness of the aperture on C as well as on the relative x-height of the o. Ideally, all permutations could be tested, but here again, the practical problem of controlling the number of tests makes this difficult. It was decided to focus on the well-defined lowercase-to-lowercase comparisons for this project, to establish initial results before extending into other areas.

6.4 Evaluation of algorithms by typeface and letterform profile shape

For the nine typefaces that received exposures in all three test conditions, the left and right profiles of the letterforms were categorised into six profile-shape groups: straight, round, diagonal, full open, half open, and unbounded, as determined by the shape of each profile between the baseline and the x-height. The ‘unbounded’ category referred to forms where there was a horizontal main stroke at either the baseline or the x-height, but not at the other, in particular r. This categorisation step was performed by hand early in the test-font preparation process; in practice, only the a, g, and J letterforms were found to vary in profile shape among the upright typeface styles; the lowercase diagonal-profile letterforms (v, w, and y) were categorised as round for some of the italic typefaces, although ultimately there were no italic typefaces tested in all three test conditions.

The analysis took each of the nine typefaces in turn, subsetting the exposures for that typeface from the data set and, within the subset, computing the exposure mark rates for the lowercase-to-lowercase pairs in each test condition. For each profile shape, the exposure mark rates were extracted, both for the left-side profiles and the right-side profiles. This resulted in 54 subsets of data to analyse (9 typefaces × 6 profile shapes), testing to see if statistically significant differences could be identified between the three test conditions in any of the subsets.

A series of preliminary one-way ANOVA tests with $\alpha=0.05$ was conducted, using the null hypothesis (H_0) that the means of the exposure mark rates were equal between the three test conditions. In 15 of the 54 tests, a significant difference was reported at the chosen significance level (0.05), which should provide grounds to reject the null hypothesis for those typeface/profile-shape combinations. However, the one-way ANOVA tests cannot report which of the test-conditions resulted in a mean exposure mark rate significantly different from the others.

As a result, the analysis next conducted a series of Tukey Honestly Significant Difference (HSD) tests, also at $\alpha=0.05$, which performs pairwise comparisons between the three test conditions for each of the 54 typeface / profile-shape combinations. This provided three head-to-head comparisons in each combination:

- The composite algorithm vs the control group
- The composite algorithm vs the rival *kf* algorithm
- The control group vs the rival *kf* algorithm

for a total of 162 pairwise comparisons. The Tukey HSD tests reported 19 pairwise combinations where a significant difference was found (the p value for each comparison varies, and is reported in the tables that follow).

The pairings where the mean exposure mark rates were significantly different, grouped by profile shape, were:

Straight profiles:

- Slabo 13px Regular — composite algorithm vs control group
- Slabo 13px Regular — control group vs *kf* algorithm
- STIX Two Text Regular — composite algorithm vs *kf* algorithm
- Yrsa Bold — composite algorithm vs control group
- Yrsa Bold — composite algorithm vs *kf* algorithm

Round profiles:

- Alegreya Sans Regular — control group vs *kf* algorithm
- Literata Regular, opsz 14 — composite algorithm vs control group
- Literata Regular, opsz 14 — composite algorithm vs *kf* algorithm
- Slabo 13px Regular — composite algorithm vs control group
- Slabo 13px Regular — control group vs *kf* algorithm
- STIX Two Text Regular — composite algorithm vs *kf* algorithm
- Yrsa Bold — composite algorithm vs *kf* algorithm

Full-open profiles:

- Fira Sans Condensed — composite algorithm vs control group
- Literata Regular, opsz 14 — composite algorithm vs control group
- Slabo 13px Regular — composite algorithm vs *kf* algorithm
- Yrsa Regular — composite algorithm vs control group
- Yrsa Regular — control group vs *kf* algorithm

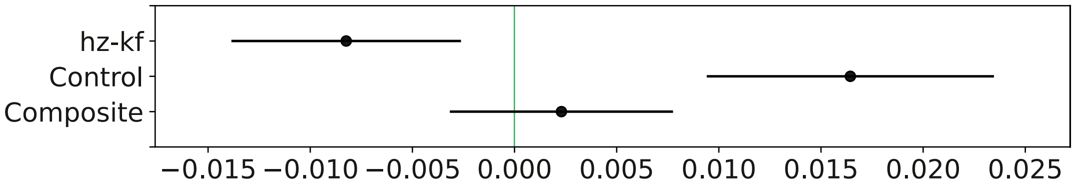
Half-open profiles:

- Literata Regular, opsz 14 — composite algorithm vs control group
- Yrsa Regular — composite algorithm vs *kf* algorithm

These pairings are designated with a Yes in the ‘reject H_0’ column of the Tukey HSD tables that follow.

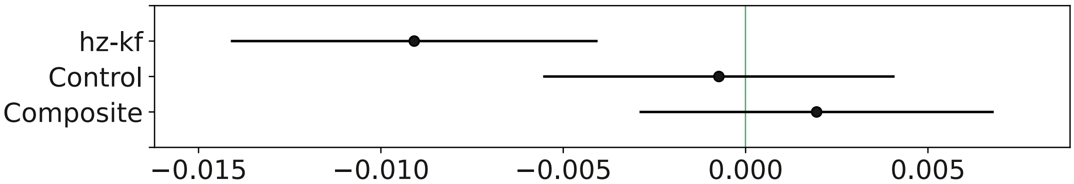
Each table is accompanied by a graph showing the means and 95% confidence interval for all three test conditions. This is important to the interpretation of the results. An exposure mark rate closer to zero corresponds to better acceptance of the fitting among respondents who viewed the exposures in the subset, but the Tukey HSD’s result indicates only whether there were significantly fewer marks or significantly more marks between the mean exposure mark rates of the test conditions at the tested error rate; it does not report which mean was closer to zero. For each pairing of significant difference in the tables that follow, the test condition closer to zero is highlighted in gold — indicating that the highlighted test condition showed a significantly lower reader-dissatisfaction rate.

Tukey Honestly Significant Difference test, FWER=0.05:
Slabo 13px Regular, profile shape "straight"



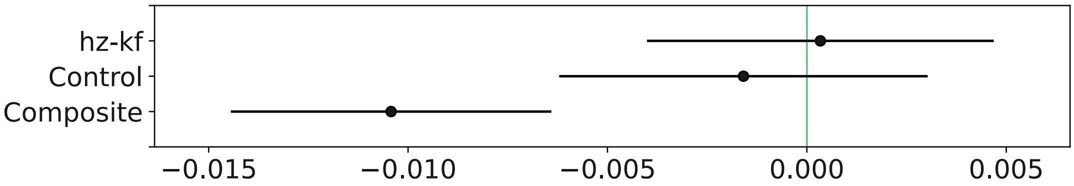
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	0.0141	0.0218	0.0017	0.0266	**Yes**
Composite	kf	-0.0105	0.0665	-0.0216	0.0005	*No*
Control	kf	-0.0247	0.0	-0.0373	-0.012	**Yes**

Tukey Honestly Significant Difference test, FWER=0.05:
STIX Two Text Regular, profile shape "straight"



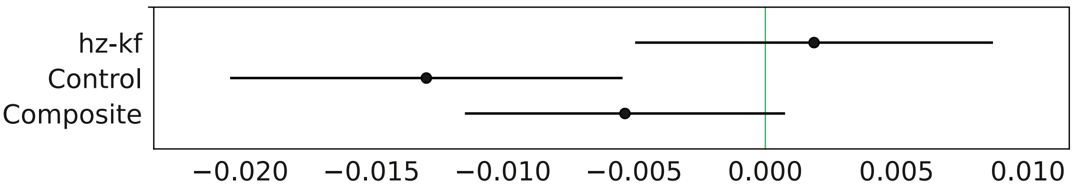
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0027	0.7924	-0.0124	0.007	*No*
Composite	kf	-0.011	0.0242	-0.0209	-0.0011	**Yes**
Control	kf	-0.0084	0.115	-0.0182	0.0015	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Yrsa Bold, profile shape "straight"



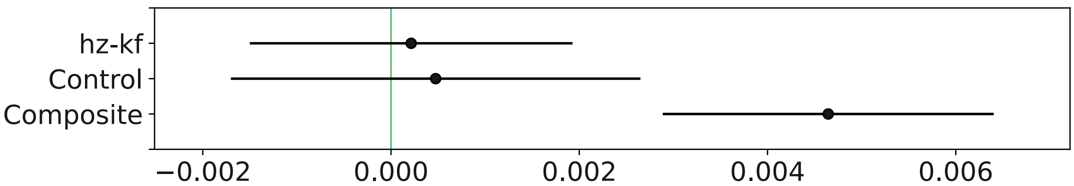
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	0.0088	0.0436	0.0002	0.0175	**Yes**
Composite	kf	0.0108	0.0073	0.0024	0.0191	**Yes**
Control	kf	0.0019	0.869	-0.007	0.0109	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Alegreya Sans Regular, profile shape "round"



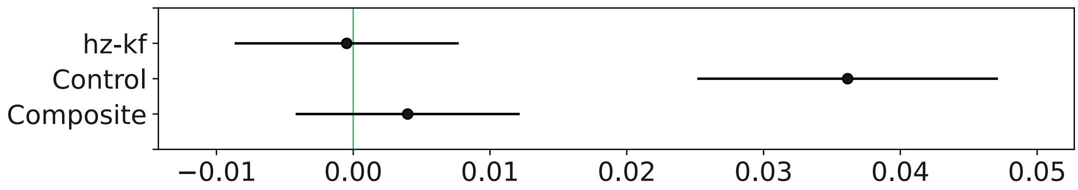
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0076	0.39	-0.0211	0.006	*No*
Composite	kf	0.0072	0.3897	-0.0057	0.0201	*No*
Control	kf	0.0148	0.0409	0.0005	0.0291	**Yes**

Tukey Honestly Significant Difference test, FWER=0.05:
Literata Regular at opsz 14, profile shape "round"



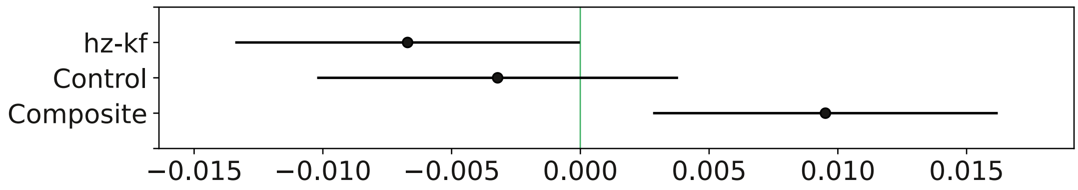
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0042	0.0347	-0.0081	-0.0002	**Yes**
Composite	kf	-0.0044	0.008	-0.0079	-0.001	**Yes**
Control	kf	-0.0003	0.9865	-0.0042	0.0036	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Slabo 13px Regular, profile shape "round"



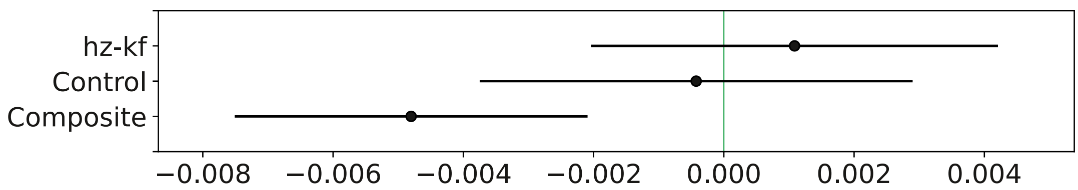
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	0.0322	0.0003	0.013	0.0513	**Yes**
Composite	kf	-0.0045	0.7987	-0.0208	0.0119	*No*
Control	kf	-0.0366	0.0	-0.0558	-0.0174	**Yes**

Tukey Honestly Significant Difference test, FWER=0.05:
STIX Two Text Regular, profile shape "round"



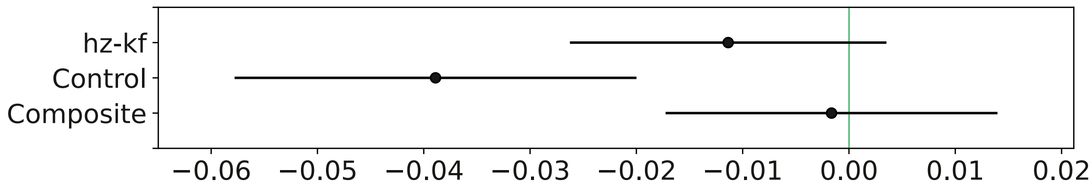
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0127	0.0749	-0.0264	0.001	*No*
Composite	kf	-0.0162	0.0126	-0.0296	-0.0028	**Yes**
Control	kf	-0.0035	0.8203	-0.0172	0.0102	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Yrsa Bold, profile shape "round"



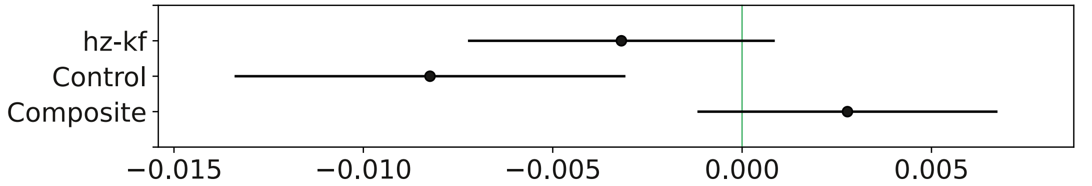
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	0.0044	0.2039	-0.0017	0.0104	*No*
Composite	kf	0.0059	0.0473	0.0001	0.0117	**Yes**
Control	kf	0.0015	0.8459	-0.0049	0.008	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Fira Sans Condensed, profile shape "full-open"



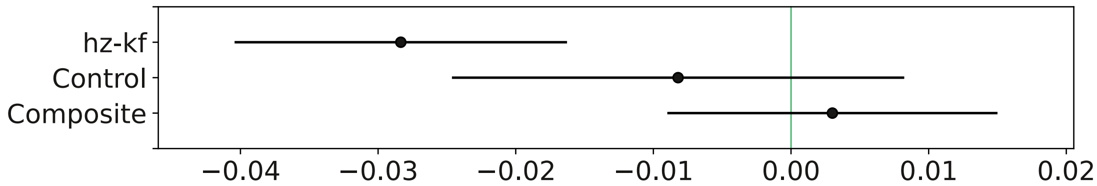
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0373	0.0308	-0.0718	-0.0027	**Yes**
Composite	kf	-0.0097	0.7335	-0.0402	0.0208	*No*
Control	kf	0.0275	0.1351	-0.0063	0.0613	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Literata Regular at opsz 14, profile shape "full-open"



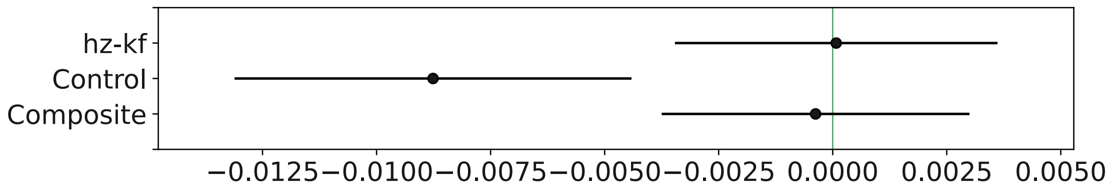
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.011	0.013	-0.0201	-0.0019	**Yes**
Composite	kf	-0.006	0.1872	-0.014	0.002	*No*
Control	kf	0.0051	0.4014	-0.0042	0.0143	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Slabo 13px Regular, profile shape "full-open"



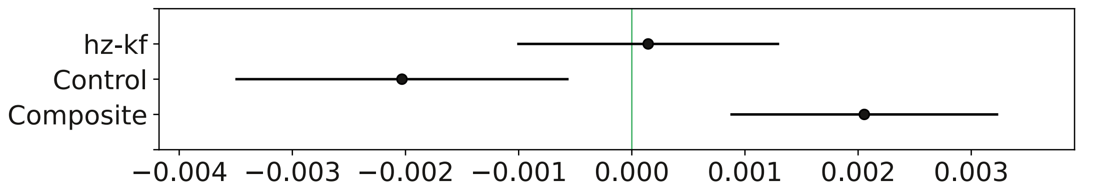
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0112	0.6225	-0.0396	0.0172	*No*
Composite	kf	-0.0314	0.0066	-0.0554	-0.0073	**Yes**
Control	kf	-0.0201	0.2207	-0.0487	0.0084	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Yrsa Regular, profile shape "full-open"



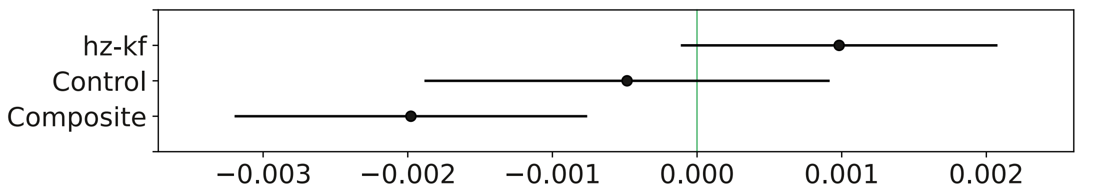
		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0084	0.0296	-0.0161	-0.0007	**Yes**
Composite	kf	0.0005	0.987	-0.0065	0.0074	*No*
Control	kf	0.0088	0.0236	0.001	0.0167	**Yes**

Tukey Honestly Significant Difference test, FWER=0.05:
Literata Regular at opsz 14, profile shape "half-open"



		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	-0.0041	0.001	-0.0067	-0.0014	**Yes**
Composite	kf	-0.0019	0.1346	-0.0043	0.0004	*No*
Control	kf	0.0022	0.1272	-0.0005	0.0048	*No*

Tukey Honestly Significant Difference test, FWER=0.05:
Yrsa Regular, profile shape "half-open"



		meandiff	p-adj	lower	upper	reject H_0
Composite	Control	0.0015	0.3725	-0.0011	0.0041	*No*
Composite	kf	0.003	0.0078	0.0006	0.0053	**Yes**
Control	kf	0.0015	0.3509	-0.001	0.004	*No*

Collectively, these test results present a mixed picture; each of the three test conditions (the two algorithms and the control group) fared better, with statistical significance, in more than one of the pairwise comparisons. Furthermore, no significant difference was found in 143 of the 162 pairwise comparisons. Thus, there is not a clear 'winner' to declare across the full set of typefaces tested on that basis.

Nevertheless, the fact that the two algorithms tested were only shown to be *less preferable* than the control group at the α=0.05 significance level in 3 of the comparisons was a more successful outcome than had been anticipated. Among the pairwise comparisons, and within the individual profile-shape groups, there are more interesting distinctions to note.

6.4.1 Examining pairwise results involving the composite algorithm

In the pairwise comparisons between the composite algorithm and the control group, there were 8 typeface + profile-group tests in which a significant difference in the mean exposure mark rate was observed. Out of those 8, the composite algorithm exhibited a lower reader-dissatisfaction rate in 5 comparisons, which is the result regarded as a success. A closer-to-zero mean suggests that fewer respondents considered the fitting to look incorrect for those typeforms when the composite algorithm had been used to generate the letter fitting. Conversely, in the 3 comparisons for which the control group had a lower reader-dissatisfaction rate, more respondents considered the fitting of the typeforms incorrect when the composite algorithm had been used to generate the letter fitting.

In the pairwise comparisons between the composite algorithm and the rival *kf* algorithm, there were 7 typeface + profile-group tests in which a significant difference in the mean exposure mark rate was observed. Out of those 7, the composite algorithm exhibited a lower reader-dissatisfaction rate in just 2 comparisons, and the rival *kf* algorithm exhibited a lower reader-dissatisfaction rate in 5. This suggests that the *kf* algorithm produced more acceptable fitting for the tested exposures than did the composite algorithm.

As discussed in section 6.3.2, though, attempts to reduce the data down to a single, overall success signal discards too much information to be of practical applicability. In particular, although it is intriguing to look at the the head-to-head pairwise comparisons between algorithms as a group, the question of greater interest was whether the tests would identify any significant differences between the algorithms that can be reliably linked to the design and implementation of the algorithms themselves. Any such differences could shed new light on algorithm implementation or on the axiomatic model upon which the composite algorithm was designed.

6.4.2 Examining pairwise results by profile shape

To that end, it is arguably more revealing to consider the results of the Tukey HSD tests by looking at the profile-shape groups. On that question, in four of the profile shapes, statistically significant differences were identified in one or more of the pairwise comparisons: straight profiles, round profiles, full-open profiles, and half-open profiles.

In the straight profiles, the composite algorithm was identified as having a preferable mean exposure mark rate for Slabo 13px Regular (vs the control group) and STIX Two Text Regular (vs the rival *kf* algorithm), but a less preferable mean exposure mark rate for Yrsa Bold (vs the control group). In the round profiles, the composite algorithm was identified as having a preferable mean exposure mark rate only for Slabo 13px Regular (vs the control group), and a less preferable mean exposure mark rate for Literata Regular at opsz 14 (vs the control group and vs the rival *kf* algorithm) and for STIX Two Text and Yrsa Bold (vs the rival *kf* algorithm). In the half-open profile group, though, the composite algorithm was identified as having a less preferable mean exposure mark rate for both Literata Regular at opsz 14 (vs the control group) and Yrsa Regular (vs rival *kf* algorithm).

In the full-open profile group, though, the composite algorithm was identified as having a preferable mean exposure mark rate for Fira Sans Condensed, Literata Regular at opsz 14, and Yrsa Regular (vs the control group) and for Slabo 13px (vs the rival *kf* algorithm).

Because the composite algorithm employed novel techniques for fitting open-counter profiles, the lower reader-dissatisfaction rate indicated by the tests may suggest that the techniques have merit and are worth further exploration. The results of these pairwise comparisons are far from an overwhelming success for the techniques, most notably because the only statistically significant differences found for the half-open profiles showed the composite algorithm to exhibit a higher reader-dissatisfaction rate.

By the same token, the composite algorithm fared less successfully than the control group and the rival *kf* algorithm in the pairwise comparisons for straight and round profiles, a result which could inform further refinement. Recall that in the composite algorithm, the straight and round profiles were both fitted strictly on the basis of the standard inter-letter area rule, which computed a standard inter-letter area by calculating the interior space of the key letterform n and scaling that by a tunable parameter. Thus, if further testing were conducted, an interesting avenue for future research might be to adjust the tunable parameter and assess whether that produces more preferable outcomes.

6.4.3 Size and scope of the effects observed

Drawing inferences directly from the data or the statistical tests must only be done cautiously. The overall signal in the data is small; on average, most respondents marked only a few sequences as looking incorrect. That overall threshold for marks made by respondents corresponds to the low exposure mark rates calculated, but it is also built into the test:

respondents were not asked to exhaustively evaluate pairs of letterforms or sequences, but instead only to mark what they noticed.

This was by design, of course: the testing methodology sought to measure where differences in letter fitting were perceived as noticeable by readers in a Latin text setting. The overall small signal observed in the test may, therefore, be explained by concluding that the two test algorithms produced fitting that was close to the acceptable tolerances of readers in the real world, but both of the test algorithms may still benefit from further refinement. More sensitive measurement might necessitate testing with greater numbers of readers or relying on other testing methods.

To assess the size of the effect observable between test conditions and whether it constituted a meaningful effect for real-world typeface design, the mean exposure mark rates were examined for each typeface + profile-shape group. There are, of course, separate ranges for each of the pairwise comparisons. Overall, the 95% confidence intervals for the mean exposure mark rates in those pairwise comparisons where a significant difference was reported ranged from -0.08 to +0.06. Those extremes correspond to 6–8% of occurrences of a particular profile shape being identified as exhibiting a fitting problem (either too 'tight' or too 'close') by the respondents. Some profile shapes, however, exhibited exposure mark rates one or two orders of magnitude smaller — which may signify that all of the test conditions performed equally well.

Because this measurement methodology is new, there is no pre-defined standard against which this size of effect can be judged. However, it did seem reasonable to hypothesize that a 6–8% rate of readers regarding a letterform as being poorly fit would constitute a significant issue for typeface designers or typographers.

6.4.4 Interpreting the results of the tests

Pure numbers aside, in order to interpret the meaning of the test results it should also be recalled that a *more noticeable* problem with the fitting of a particular pair of letterforms is not the same as saying that the fitting is *more wrong*. In real-world text settings, some pairs may be more noticeable for semantic reasons (such as pairs occurring at the beginnings of sentences) or because of the layout (such as the first or last lines in a paragraph). The randomization of the texts in the tests was intended to temper this effect, but there may also be other cases in which a fitting problem is more noticeable.

In addition, not all letterform profiles are of equal importance because letters are used at different frequencies; that is, the letterforms and profiles in real texts — regardless of the language — are not evenly distributed (Grigas and Juškevičienė, 2018). Yet that fact by itself does not mean that the most frequently occurring pairings matter more. It was observed in the study of Latin letter-fitting practice in chapter 2 that some of the fitting axioms are tied to legibility, such as the prohibition against collisions in axiom L–11. For text settings, then, ranking the letterforms by the frequency at which they occur in the language could be misleading; it may be instead that deciding the importance of pairs should also involve considering the fitting axioms.

Weighting the prohibition of collisions as more serious, even if they occur only in infrequent combinations, is just one example. It was noted in chapter 3 that some of the Latin fitting axioms, such as the vertical stroke-rhythm axiom (L–5), apply to text at a larger scale than pairs of adjacent forms. If the importance of vertical stroke rhythm lies in how it dominates the patterns of Latin text for continuous reading, then perhaps straight-profile letterforms should be considered more important when interpreting the results of quantitative tests such as those in this project.

6.4.5 Algorithm design

Some final interpretations drawn from the quantitative test results are observations about the test algorithms themselves. It was reassuring to see that the novel method developed for open-counter profiles in the composite algorithm showed some positive results, not simply because it uses a different technique from the rival *kf* algorithm, but because it suggests that improvements could be made by considering the various fitting axioms independently of each other. That has implications for the development of letter-fitting algorithms for other scripts and for continuing to improve on Latin text fitting: identifying and understanding the rules that govern fitting in a script, even incrementally, can improve how algorithms fit text, without necessarily requiring the development of a new algorithm from scratch.

It was interesting to note that the rival *kf* algorithm was found to have statistically significant advantages over the composite algorithm and the control group in several of the tested font and profile-shape groups. However, in the head-to-head comparison, it should be remembered that the tested version of the kf algorithm was a new reimplementation based on the documentation available in the original patent, rather than being the original URW software. Attempts were made not to deviate from the details available, but several parameters that were described only as options in the patent filing, and values for those parameters had to be chosen.

Notably, that includes two parameters which were also used in the composite algorithm: the minimum sidebearing distance and the choice of scaling factor used in computing the standard inter-letter area. As described in the published material, the *kf* algorithm used o as the key letterform from which the standard inter-letter area for lowercase forms was calculated, but no multiplier or adjustment factor was described to scale the interior counter-area of o. Instead, the unadjusted interior counter area was used. In contrast, the composite algorithm always scaled down the interior area of its key letter (n), to follow the advice of the letter-fitting literature. The practical result was that the *kf* algorithm consistently used a larger inter-letter area than the composite algorithm.

That unscaled value for *kf*'s inter-letter area perhaps reflects the intent of the algorithm accurately, but it should be recalled that in the quantitative tests, a significantly larger proportion of all marks made (75%) were of the 'tight' variety. That could mean that a consistently larger inter-letter area will consistently skew towards better results, even with all other factors being equal; it may be that tight spacing errors are more noticeable to readers in this test methodology. There are sources in type-design literature that advise the use of uniformly looser fitting for smaller point sizes (Unger 2007, p. 115; Hochuli 2015, p. 26), in addition to some readability studies that suggest uniformly looser typographic spacing improves readability for some readers (Beier et al. 2021; Łuniewska et al. 2022).

The other tunable parameter chosen for the reimplementation of the *kf* algorithm was the minimum distance; zero was used in the tested reimplementation for the sake of simplicity. That choice by itself is perhaps defensible, but when used in combination with the consistently looser fitting of the *kf* reimplementation's inter-letter counter area, it is unknown whether it affected the *kf* test condition for better or worse.

Ultimately, of course, speculating on how changes to any of the algorithms or parameters could impact quantitative test results is a theoretical topic. In an ideal world, it would be possible to stage tests with every permutation of parameters and in a wide variety of typeface styles. But time and volunteer respondents are both finite, relatively scare resources. The quantitative testing described in this chapter provided some insights into how successfully the test algorithms might be received if employed to fit other typefaces, but on its own, it did not definitively answer the project's research questions about the overall viability of a letter-fitting algorithm to generate fitting that cannot be distinguished from manually-determined fitting and about the modelling of manual-fitting processes. As will be discussed in the next chapter, there are potentially useful insights to be gleaned from other portions of the project, such as the historical survey, axiomatic modelling, and analysis, that may inform other useful research into algorithmic letter fitting.

7. Conclusions

This research has sought to investigate how and to what extent an algorithm can perform the task of letter fitting a well-designed text typeface, in a manner that is consistent with the proven manual fitting processes of type practitioners, as type is experienced and is assessed by readers when they engage with text in the real world. By its very nature, these questions are multi-faceted; to begin with, they invoke the independent perspectives and concerns of distinct groups of people — type designers and readers — and they furthermore require probing into the processes of design and reading. Exploring the research questions has thus required a multi-disciplinary approach. First, by seeking to establish reliable connections between the designer's experience and the reader's, via the avenue of the typeface itself. Second, by deriving a method to empirically assess the results reported by the reader in a manner that can likewise be meaningfully linked back to the designer's craft.

Maintaining this link has been a conscious exercise throughout the research, specifically because it holds the possibility for uncovering new insights. Where prior legibility and readability research has looked at the spacing of letters, it has tended to do so at the typographic level: adding or subtracting the same distances between all of the typeforms in a text. While such research is useful in the typesetting context, those studies have not examined the fitting of particular forms as is done during typeface design. Conversely, there have been investigations into employing statistical analysis to describe and characterize the fitting of existing typefaces, either purely formulaically or by loading sets of typefaces into machine-learning training systems. But those approaches can only capture the results of fitting already done. Thus, neither approach offers much to typeface designers to increase their understanding of the process, or speaks to how the task of fitting can and should be approached for typefaces that designers will create or refine tomorrow.

Chapter 2 began by examining the task of letter-fitting as it is practised, as it is taught, and as it is captured in the historical literature of typemaking and related disciplines and in typeface design tools. From that examination, it derived an axiomatic model focused on capturing how type designers perform the fitting task when designing Latin text typefaces. The model that it derived consists of 16 axioms that reflect the first principles of fitting Latin text typefaces, at least so far as a consensus approach can be established. The axioms themselves are not an algorithm that performs letter fitting, but a set of interrelated principles, each of which individually provides answers to some — but not all — of the letter-fitting decisions required for a given set of typeforms.

Each of the axioms provides a more formal expression of a principle likely known to practitioners of letter fitting, including where the principle can be applied and where it cannot, as well as what sort of answer the axiom can provide. There are some axioms concerned with the relative size of one space compared to another, but other axioms that provide absolute minimums or other limits. The final portion of chapter 2 demonstrates that this more formalized expression has practical utility, by discussing how various axioms interact with one another in a network of relationships that type designers must navigate. Although the focus of this research project has been the fitting of Latin typefaces for text settings, the specificity and the historical footing for the axioms in the model show that the model functions within a particular typographic context. Consequently, as noted in the discussion of the axiomatic model, the same research-based derivation process could be used to explore fitting in other scripts and writing systems.

Chapter 3 examined the Latin text fitting model as a construct in its own right, with a particular focus on finding and understanding which facets of the model can be clearly mapped to procedures in an algorithm, which are less formally described and thus pose practical challenges for an algorithmic implementation, and which are well-understood theoretically but still require further investigation. It identified two axioms as opportune for more detailed investigation: the optical centring of forms within a triplet and the handling of forms with concave or 'open-counter' side-profiles.

Chapter 4 reported on investigations into these two axioms and the identification of a potential connection to link them for practical application. On one hand, the fundamental question of fitting typeforms with an open-counter profile is where to define the boundary between the interior and exterior space of the open-counter profile. Prior work at automating letter fitting had acknowledged the need to determine a boundary, but none had established an approach grounded in theory. On the other hand, finding the optical centres of typeforms is a problem that was explored at great length by the LOGOS project of David Kindersley and Neil Wiseman — though the LOGOS method, as a whole, utilized its optical centrepoints in concert with other innovative techniques and sought to produce fitting for every typeform in the same fashion.

Through the investigation of these two axioms, it was theorized that the problem of defining the interior-exterior boundary for open-counter profiles might be addressed by finding the optical centre of the typeform in question. This possibility was suitably well-defined to be put to the test, given that the LOGOS centrepoint method is documented and could be implemented. But it also held value as a potential test of the axiomatic model for Latin text fitting; the model states that open-counter profiles are governed by principles that do not apply to closed profiles.

The majority of the prior work implementing letter-fitting automation tools has not addressed different profile-shapes with different techniques, and where the prior work has treated different profile shapes with distinct techniques, the different treatments have generally been coincidental (such as the differing effects seen with the *kf* algorithm's fixed cut-in angles) or opaque (such as LS Cadencer's pre-determined measurement tables). Chapter 4 then concluded by exploring how to incorporate the testable new method for handling of open-counter profiles into a composite fitting algorithm that combined several techniques to address a complete set of Latin text typefaces.

Chapter 5 explored the practical problem of putting any letter-fitting algorithm to the test with readers. It looked at the prior testing regimens and comparison methods found in scholarly research, commercial product marketing, and in publicly available records of software development. It established that testing refitted typefaces with readers could reliably record responses to fitting algorithms with the general public, removed from the risk of experimenter bias, while capturing information detailed enough to study the results of fitting algorithms at the per-typeface and per-profile levels.

It then described the construction of a framework that can be used to survey readers about what they perceive as unacceptable letter-fitting between individual pairs of typeforms, for fonts viewed in randomized sample text documents, by asking the readers to mark pairs of forms on each sample. By surveying a broad sample of the reading public, using fonts randomly chosen from the original versions and versions refitted by test algorithms, the framework can collect data that captures potential differences in the readers' responses to the algorithms.

The test framework was deployed on a publicly available web site, testing various Latin text fonts in stages. A set of nine fonts chosen to cover a variety of letterform constructions, styles, and typographic weights and widths was tested in three conditions: the original, manual letter fitting; a refitted version modified by the composite algorithm developed in chapter 4; and a refitted version modified by a reimplementation of the *kf* algorithm from URW's *hz-program* suite.

Chapter 6 reported on the results of those tests. The number of marks made on forms in test exposures were consolidated into metrics that represent the overall level of dissatisfaction of readers for letterforms of different profile shapes. There were statistically significant differences found between the composite algorithm developed in chapter 4, the rival *kf* algorithm, and the original fittings of the same fonts. For letterforms with open-counter profiles, the composite algorithm resulted in a lower rate of dissatisfaction among the test respondents, although with other letterform profile shapes, the novel algorithm resulted in a higher rate of dissatisfaction. This mixed result suggests that the method used to fit open-counter forms has merit, but that additional refinement of techniques for other profile shapes would likely be required to devise an algorithm that consistently tests well with readers.

While the results of the tests showed an advantage for the composite algorithm in some profile shapes and a disadvantage in other profile shapes, that divergence can be perhaps explained by the choice made in chapters 3 and 4 to focus attention on developing a new technique for fitting open-counter profiles. The novel techniques developed in chapter 3 were specific to open-profile shapes and, thus, were expected to result in an observable effect on those shapes. The fitting of other profile shapes is governed by other axioms, and without the benefit of tuning for those profile shapes, the composite algorithm was not assumed to result in significantly better results for those other shapes. Consequently, I believe that the project as a whole points to an affirmative answer to the primary research question, which asked if an algorithm can be constructed that will generate letter fitting for a well-designed typeface which cannot be distinguished from letter fitting determined manually. The testing methodology reframed the original goal of the research question from the somewhat imprecise phrasing of 'cannot be distinguished' into a more quantifiable notion of evaluating success with readers in text for continuous reading. The composite algorithm constructed and tested in the project tested similarly with readers for many typeforms across several typefaces, but its success with open-counter profiles fitted by the focused techniques shows that there is promise in further exploration.

Re-examining the central research question, the course of the research project showed that the original question was too general in its framing of letter fitting as a singular, perhaps monolithic, task. Fitting is often described as a single stage in the design process in the historical sources, but that is surely a rhetorical device to accentuate its importance: all of the forms in a typeface must be addressed before the fitting process can be considered complete enough for the typeface to be used. The literature of type design and the practices taught to type designers, however, are clear that the task of fitting involves multiple considerations, multiple evaluations, and multiple choices and trade-offs.

It is true that many letter-fitting automation tools in years past approached fitting with a singular approach, such as the *kf* algorithm's application of an equal inter-letter area to all forms or the LOGOS algorithm's application of triplet centring to all forms. But this research has shown that there are realizable gains to be made by constructing fitting algorithms with more nuance and complexity than was perhaps feasible on the computing systems that were prevalent when those algorithms were offered on the commercial market. It has also shown that systematic exploration of fitting can still reveal new useful insights that can potentially improve the performance of algorithmic letter fitting.

Although the primary research question was formulated in terms of the creation of an algorithm, when the project is considered as a whole, the course of the research contained several distinct contributions to the field that warrant discussion in turn.

7.1 Modelling the manual practice of letter fitting

The analysis of the manual practices of Latin text fitting described in chapter 2 and the axiomatic model based on that analysis offer a new perspective on a task that is shared by all typeface design projects. Historical research in type design has often investigated the practices of type designers and manufacturers, but it has rarely explored the problem of constructing a conceptual model based on those practices and giving it formal expression.

At the pragmatic level, of course, the formal model provides a structural substrate for developing fitting algorithms. But there are additional benefits to the model in its own right. A well-defined model permits systematic discussion and detailed debate about the task of fitting, by providing a common set of terminology and precise definitions. Furthermore, the examination in the workings of the model in the abstract, rather than while fitting a specific typeface, can enhance the understanding of how fitting is performed in typeface design or of how 'successful fitting' is seen in text. The discussion of domains, ranges, interactions, and interdependencies of Latin text fitting axioms that concluded chapter 2 illustrates both.

Deliberate attention was also paid, during the analysis of historical Latin fitting and the development of the Latin text fitting model, to consciously maintain a separation between the analysis and the particulars of the Latin script. Where possible, it was noted when axioms in the Latin text fitting model were unique to the script or could be known to function differently in other scripts. That analysis shows that building a script-specific model, consistent with the practice of fitting text in a particular script, is a repeatable process with general application beyond Latin alone.

The utility of the model also reinforces the value of systematically analysing historical narratives that frame type design (and potentially other design processes) as a craft that is performed manually and must, therefore, be taught only within a manual context. Analysis of the manual craft can deepen the understanding of the problems that the craftsperson solves and of the tooling available for solving design problems; probing those topics is not always comfortable, but — far from discounting the historical narrative — re-examining these historical processes can move the discourse forward by uncovering additional useful information.

Similarly, it is widely accepted — within type design as well as within other design studies — that there is always value in revisiting prior art. The novel techniques and the connections between them pointed to by this research demonstrate that rewards can be yielded by revisiting prior efforts at modelling, automation, and tool development. As noted in § 2.2.5, models describing the process of letter fitting can often be more complete complete than the technology of the day could implement conveniently. By revisiting prior analysis of design processes, design discourse can mitigate the loss of potentially valuable insights that are easy to overlook.

7.2 Analysis and implementation of algorithms

The practical component of this project involved, perhaps most notably, the design and software implementation of a technique for establishing sidebearings for typeforms with concave or open-counter side profiles, based on combining concepts known from prior letter-fitting implementations in a new way. It is undeniable that this specific technique, even if refined further, would at best provide useful answers for a subset of the typeforms that need to be fitted in a Latin text typeface. But, more generally, the results indicate that it is possible to incrementally improve a fitting algorithm in a form-by-form fashion. Furthermore, the practical investigations and experimentation that led to the technique afforded other opportunities to gain insight into the implementation of typeform-fitting software and the overall design of algorithms for fitting.

Based on the historical survey, it appears that the LOGOS component developed for this project may be the first independently written reimplementation of the core LOGOS methods since the original project. Similarly, although there are implementations recorded for other components in the *hz-program* suite (Thế Thành 2000), the reimplementation of the *kf* algorithm for this project may also be the first. In both cases, independently developing and putting the algorithms to the test was instructive in ways apart from their direct use in the fitting algorithms.

The LOGOS reimplementation revealed new avenues for exploring how letterforms are classified. The centrepoint-finding component was developed in order to address the problem of defining the boundary between interior and exterior space, but in practice it may provide an objective test for whether a form should be classified as having an open-counter profile, which can be a nebulous question for forms such as f, t, and r, in some typefaces.

In the higher-level problem of devising a multi-part algorithm that determines letter-fitting for a set of typeforms, implementing the *kf* algorithm helped highlight issues about how the Latin text fitting axioms should be mapped into a tool that is practical for type designers. For example, although type-design literature dating back to Fournier has posited that the interior width of the counter in the key letterform n should form the basis for the standard inter-letter area of all of the typeforms (Axiom L–6: Interior-Exterior Balance), implementing that rule during development revealed that there is considerable disagreement about the exact ratio between that interior width and the standard inter-letter area (see chapter 2, p. 49–50). The published *kf* patent does not address this ratio, nor do contemporary software tools dealing with fitting by equal inter-letter areas, such as HT Letterspacer. The only way to make adjustments to this ratio when working with HT Letterspacer is to edit the Python source code and, even then, applying a different ratio requires altering several hard-coded values.

In the course of this project's practical implementation, engaging with these issues raised new questions about whether the relationship between the counter in the key letterform n and the standard inter-letter area is well-understood, particularly as it varies across weight, width, and style, and it revealed many opportunities to re-evaluate how tunable settings and decisions that should be available to the type designer might be presented as easily understood parameters either in discussions of letter fitting or in letter-fitting software tools.

More generally, this project's findings about the interconnectivity and interactions between the oft-cited principles of letter fitting show that a systematic discussion — and even dissection — of aspects of design work holds practical value for the craft. At times, design work can be too easily categorized as dominated by the need for intuitive judgment. Type design, with its scrupulous attention to detail and its concern with practical outcomes like readability, is more resistant to this temptation that some other fields. The practical findings of this work reinforce that systematic investigation can generate usable insights, and has a place within the discourse.

7.3 Quantitative test methodology

The testing framework described in chapter 5 and the analytical methods used in chapter 6 to evaluate fitting algorithms are both contributions with general usage for conducting type research. As was noted at the start of chapter 5, there has been little structured research into letter fitting, and the evaluation methods that were historically employed were deemed not appropriate for assessing the success of letter-fitting algorithms with readers. The assessment methods in prior research were deemed not to be appropriate often because they did not test with readers, and relied instead on the evaluations of the researcher (which are subject to bias). But other issues were identified, such as reliance on fixed sample texts, or by ultimately defining the metric for success as whether or not an algorithm reproduced letter fitting identical to the original fitting of the typeface.

The methodology developed in this project provides a new, general framework for testing fitting with readers that addresses those concerns directly. Tests can be conducted in large sample sizes, recording basic technical and demographic variables about the participants, while preserving their anonymity. The tests conducted for this research show that it is possible to recruit participants in significant numbers both from the general reading public and from within the more narrow confines of people who possess experience with type and typography. Where the data itself is concerned, the framework was deployed in tests that focused on the most fundamental fitting question for Latin text: whether two adjacent typeforms are too close or too far apart in the horizontal direction. But it is capable of more general tests; it would be simple to add more options to the choices offered when a respondent highlights characters and thus investigate other questions, such as whether diacritics, marks, or secondary forms are considered to be positioned too high or too low in the vertical direction. Potentially, additional enhancements to the same basic mark-making technique would also enable the testing of scripts that feature other letter-fitting questions.

In the analysis stage, the data collected by the framework was detailed enough that the typeforms of each tested typeface could be collated by profile shape, but the same data format would permit analysis of each typeform individually or across the entire typeface. For this project, metrics were defined that captured the patterns of marks made by profile shape, but that decision was based on the functionality of the algorithms being tested. The same test framework and data format could alternatively be used to analyse the effect of a fitting change on a single form, on classes of forms defined in some other fashion, or perhaps even to look for previously unknown relationships between forms and classes.

It is also notable that the test framework permits direct-comparison analysis of fitting algorithms when those algorithms are used to refit the same typeface. For this project, such head-to-head comparisons were deemed important, because they permitted the comparison of a new fitting algorithm against the original, unmodified fitting of each font, and did so without assuming that reproducing the original fitting of the font was the universal measure of success. Nevertheless, the framework itself and the data model are general enough that this style of algorithm-versus-algorithm test is not the only possible experiment. The testing framework could be used without modification to make comparisons across demographic groups, across languages, on changes made to the design of typeforms, or on incremental changes to the parameters of any particular fitting algorithm.

Naturally, there are limits to what the framework and data model can test. Perhaps the most notable limitations are that the framework is a web-only, client-server testing environment that does not test printed samples and — in remote tests — cannot record all of the system or environmental factors (such as the brightness and dot-pitch of the display or room lighting) that might be of interest to researchers. However, some of these technical limitations may be surmountable by further development of the software, or by performing tests in a controlled environment.

It is also not essential to the testing methodology that letter-fitting be the sole task given to test respondents. The core functionality of the framework could potentially be incorporated into broader tests with readers or user-experience research, with letter-fitting issues being just one among several forms of feedback obtained. To consider the testing methodology more generally, the quantitative component of this project illustrates that empirical testing and analysis is capable of addressing design questions beyond the legibility and readability studies that have steadily gained popularity in recent years.

7.4 Discussion

The new technique developed for fitting typeforms with open-counter profiles showed promise in the public tests. But it must be remembered that addressing those forms makes up only one part of the overall task of fitting a typeface, and that the tests attempted to measure success with readers by providing a specific form of feedback: marking letterforms where the fitting appeared incorrect. To progress further towards the development of a fully comprehensive fitting algorithm, work remains to be done.

As discussed in chapter 4, the scope of the composite algorithm was restricted so that it could be tested. The restrictions put in place for the tests are areas where the composite algorithm can clearly be extended, including implementing kerning, implementing the exception rules for single-stroke forms or adjacent extenders, implementing capital-to-capital fitting, and the further refinement of the tunable parameters. Likewise, the testing methodology can be extended. There are facets of the test framework that could be further refined, such as the layout and design of the text samples. Additionally, there are technical improvements to consider, like developing methods for monitoring aspects of the response session (such as detecting the zoom level of the browser or custom tweaks to font settings which may interfere with the test) that could not be reliably monitored during the public tests.

In addition to specifics such as these, there are other, more architectural aspects to designing a fitting algorithm that warrant further consideration. For example, it was noted in chapter 4 that a decision had been made to design an algorithm capable of addressing the basic Latin letterforms in the simplest fashion. That meant that each sidebearing was determined once, based on a rule chosen according to the shape of the profile. But chapter 4 also noted that this was not the only possible approach. There may be improvements to be seen by finding other ways to traverse the entire set of letterforms, perhaps invoking several axioms on each profile and finding a technique to balance the results. There is also the possibility that further exploration of the two axioms deemed incomplete in chapter 3 (L–5: Vertical Stroke Rhythm and L–3: Shells of Space) can bear fruit and that they will prove useful for algorithmic fitting, and that more detailed explorations of the interactions between the axioms can yield further insight.

Then again, perhaps it goes without saying that more complex or more nuanced fitting algorithms could be developed with further study, for that is likely always the case. Computer scientist Donald Knuth, in his foreword to Robert Sedgewick and Philippe Flajolet's *An Introduction to the Analysis of Algorithms*, wrote:

> People who analyze algorithms have double happiness. First of all they experience the sheer beauty of elegant mathematical patterns that surround elegant computational procedures. They receive a practical payoff when their theories make it possible to get other jobs done more quickly and more economically. (Knuth in Sedgewick and Flajolet 2013)

Although this project, like many investigations into algorithmic letter fitting before it, initially looked at fitting algorithms as utilities for tackling the job of letter fitting more quickly and economically, perhaps the most far-reaching outcome of the research is not the techniques or algorithms themselves, but rather the more thorough understanding of the procedures used in letter fitting and the underlying patterns that govern how typeforms in Latin are fitted. That understanding was developed through the systematic examination of the Latin letter-fitting task as it is practised, and by translating optical judgements and intuitive rules into more concrete and definite expressions.

This systematic understanding of fitting, as a task composed of known rules and relationships, can yield dividends over a patchwork of disparate techniques, even if fitting continues to be done by a designer manually. The research findings confirm that algorithmic approaches to letter fitting have a role to play in the future of type design that extends beyond the promise of any one-size-fits-all 'generate the fitting' button. Thinking about the model that governs successful fitting — in any script — can lead to practical improvements in visualization and tooling for typeface design, for experimentation with fitting and space in general, and to richer conversations about the function that space plays in the design and reading of letters.

7.5 Prospects for further research

Additional research could explore the task of fitting in other writing systems and in Latin fitting beyond the setting of text for continuous reading. A great deal of the literature and historical record for letter fitting within Latin text has focused on 'regular' weights and proportions, with less scrutiny applied to lighter and heavier weights, width variations, and optical sizing, which is a sensible point from which to start, but leaves considerable typographic design-space unexplored — as well as the fitting of numerals and punctuation. Furthermore, the task of fitting is intertwined with design questions, such as the classification of letterforms and the optical alignment of diacritics and marks, that have seldom been the object of scholarly research.

Perhaps it also goes without saying that no historical survey can ever be one hundred percent comprehensive. In addition to the Latin fitting axioms exhibiting unresolved questions discussed in § 3.3, there are numerous historical investigations into letter fitting as a process or letter-fitting automation worthy of detailed scrutiny. There are also potentially interesting lines of inquiry to be found in more formally analysing the historical trends of letter fitting across the history of type manufacturing, as well as in analysing and characterizing the fitting styles unique to particular type designers.

On the analytical front, there are potential avenues for further research in establishing other metrics by which to assess readers' satisfaction, not just dissatisfaction, with fitting, as well as to explore the role of letter-fitting in reading speed and comprehension — which have typically studied only typographic spacing. It is also evident that other testing methodologies could be used in parallel, including tests on printed samples or side-by-side comparison tests. There are open questions regarding how fine a change in letter fitting can be before readers can no longer distinguish between small adjustments. It is not clear that data granular enough to tell a type designer 'five more units of space are needed here' can ever be collected. That is to say, it may be that clearing the 'dissatisfaction' hurdle with readers is all that can be asked of an algorithm, and that goal may answer the practical question of algorithmic fitting that cannot be distinguished from manual fitting. But even if that were shown to be the case, that deeper understanding of the accuracy and noticability of letter fitting could open the door to new approaches in spacing lines and paragraphs.

There are also many questions left to explore regarding how a letter-fitting algorithm can and should be implemented as a tool for the type designer. The unattended algorithm that calculates sidebearings for all the forms in a typeface might be welcomed by enough users to survive as a viable utility for quite some time. But, as was discussed in chapter 2, the process of designing a typeface is iterative, passing from design to fitting to testing and back again numerous times with even the most experienced type designer. Ultimately, a letter-fitting algorithm must make itself useful to that process, as employed by type designers in their practice. The approaches, models, and frameworks explored in this research project form a launching point for researchers to continue to pursue these questions and develop a more fully realized understanding of the relationships between letterforms and the spaces that surround them when type is set.

Glossary

i. Conventions used in this work

As noted in chapter one, § 1.3, this work has adopted some conventions in terminology for the purpose of clarity, where more variation may be found in the historical sources or online discourse. The term *fitting* is used in place of 'spacing' in order to distinguish the task of letter fitting in typeface design more clearly from typographic spacing or tracking.

Along those same lines, when discussing fitting, this work makes an effort to use the term *letterform* to refer to the shapes of letters being fitted, as distinguished from the conceptual components of the alphabet, or the term *typeform* to refer to the broader category of shapes of letters, ligatures, numerals, punctuation marks and other symbols. This facilitates a clearer discussion in some cases, such as distinguishing between the one-storey and two-storey forms of a and g. Nevertheless, the literature of fitting commonly refers to the task itself as "letter fitting", which is preserved for clarity; it is of course clear that this term is one that reflects the overwhelming importance of letterforms versus the other typeforms that make up a text.

Wherever possible, this work has also adhered to the convention of using *typeface* when referring to the design of a family of letters, and *font* when referring to the final product — in contemporary usage, the digital file produced, installed on a computer, and used in the various experiments and tests of chapters 3 through 5. Nevertheless, the distinction between the typeface and the font is nebulous in some discussions of practice.

This work has also standardized on using the term *legibility* to refer to the ease with which a letterform can be recognized and distinguished from other forms, and the term *readability* to refer to a reader's ease or comfort in comprehending a text.

ii. Definitions

For terminology used in this work when referring to typography or the anatomy of typeforms, effort was made to adhere to standardized definitions as found in historical sources (Tracy 2003; Cheng 2005; Baines and Haslam 2005; Rosendorf 2016). For terms from web specifications, TrueType, and OpenType, effort was made to adhere to standardized definitions from their respective vendors (World Wide Web Consortium 2023; Apple Inc 2023; Microsoft 2022). A brief reference is included for convenience.

The **advance width** of a typeform is the total horizontal distance that the form contributes when it is added to a word or to a line of text, including the width of the form itself and its sidebearings.

Aperture refers to an opening that is partially bounded by the contours of a letterform. Some sources equate aperture with the distance from one side of the opening to the other, and not with any white area enclosed within the letterform, but the distinction is nebulous.

The **baseline** in Latin type is the invisible line that runs along the bottom of virtually all letterforms, not including any descenders, and not including any undershoots. For digital fonts, the baselines corresponds to *y=0* in the internal coordinate system.

Black, **foreground**, and **ink** are used more-or-less interchangeably to refer to the positive image of a typeform.

Body size refers to the total height, typically expressed in font units, of a typeform, including all of the empty space above and below. In metal type, the body size would be the physical top-to-bottom size of the sort. Consequently, most forms in a typeface have the same body size, and some usage of the term refers to the body size of the entire typeface.

Bowl refers to a round or elliptical component of a letterform.

Bézier curves are the quadratic or cubic function segments that make up the contours of a typeform in contemporary digital vector fonts. Béziers can be curved or straight.

CSS or Cascading Style Sheets is the W3C specification for stylistic markup in HTML documents.

CSS Weight is a numeric font property defined in the CSS specification meant to represent the typical range of typefaces' weights. CSS Weight is defined to be a number from 1 to 1000, and regular text weight is defined to be CSS Weight 400. These numbers, however, are conventions and not measured quantities, and thus do not map consistently to the stroke thicknesses or density of fonts.

Capital height is the height of capital letters in a typeface. The capital height is typically stored as a font-wide property in digital fonts files, and is measured on a straight-sided capital form like H. Letterforms like A or O may overshoot the value and diacritics on capitals may exceed the capital height, but are still regarded as being at the capital height.

Contour or curve refers to any of the lines that define the shapes of a typeform.

Contrast refers to the ratio seen between the thickest and thinnest strokes of a letterform or typeface. In Latin, vertical strokes tend to be the thickest and horizontal strokes tend to be the thinnest, but the thickest and thinnest points may be found at any angle, depending on the style. Regardless, only the main strokes of letterforms are generally considered when discussing contrast; thin serifs or thick terminals do not factor into contrast.

Counter refers to a region of white space that in bounded, enclosed, or partially enclosed by the contours of a letterform. A **closed counter** refers to such a region that is bounded on all sides, such as the interior of `o`, `p`, `b`, or `d`. Some sources will also consider a counter closed if it is bounded on all sides except at the baseline, such as the interior of `n` or `h`. Regardless, it is widely accepted that an **open counter** refers to such a region that is *not* enclosed on one side or not enclosed at the top.

Design space refers to the set of possible variations within which a typeface family might include individual member fonts: the full range of weights from thin to heavy, the full set of widths from condensed to extended, the various optical sizes from caption to headline, and perhaps even other variants.

Diacritic refers to any mark or sign added to a basic letterform, which in combination results in a new form. Most Latin diacritics are positioned above or below the letterform, although there are exceptions.

Em is a unit that refers to the maximum body size of the forms in a typeface. When text is rendered with CSS, the em is scaled to be the point size declared for the font (e.g., 10 point or 16 point). Internally, the grid system in which the contours of the forms are defined covers one em in width and one em in height. Historically, body size was closely related to the em, but in casual usage, the em may also be used to refer to width, while body size is generally only a term used for height.

An **extender** is a stroke in a Latin letterform that either rises above the x-height (termed an ascender) or drops below the baseline (termed a descender).

A **family** of typefaces or fonts is a set of typefaces or fonts that are designed and intended to work together, often sharing construction, stylistic touches, and proportions, but varying in weight, width, slant, or optical size.

Font units are the numeric coordinates that are used internally to define the points and contours of a typeform. They have no physical size, and start at *(0, 0)* at the leftmost point on the baseline.

A **glyph** is the commonly used technical term for how a typeform is stored in a digital font file: the contours, metrics, and various metadata needed to render it or print it. Most glyphs represent typeforms, but digital font formats can include other elements (such as diacritics and reusable components), so there can be a distinction. For example, the letterform `j` may be stored as two component glyphs, one for the dot and one for the base stroke, to simplify the inclusion of related forms like `ĵ`.

`GPOS` and **`GSUB`** are the tables in OpenType or TrueType font files that contain smart-font features such as ligature substitution rules (`GSUB`) and contextual positioning rules (`GPOS`). The `GPOS` table is where most kerning information is stored in contemporary digital fonts.

Green's Theorem is a relationship in calculus that allows the computation of integrals on a two-dimensional shape (such as the shape's area) by converting them into distinct but related integrals that operate entirely on the boundary curves of the shape.

A **kern**, in contemporary digital fonts, is an adjustment made between two adjacent typeforms. In metal and wood type, the term may have referred instead to the actual cuts made into a sort or the overhanging part of the sort after the cut was made.

A **matrix**, in metal typemaking, was a hardened metal block containing the negative cavity of a typeform. When a matrix was fitted into a mould, the molten metal was poured to cast a sort. The negative of the typeform was struck into the blank matrix with a punch, after which the matrix had to be *justified* to ensure its sides and faces were straight and that the correct space was allocated on each side.

A **mould**, in metal typemaking, was the receptacle holding the matrix and in which a metal type sort was cast.

A **numeral** refers, in this work, to the decimal digits 0 through 9.

Oblique and **italic** refer to slanted typeface styles in Latin. Various sources debate the nuanced meaning of italic, which can involve changes to the skeletons and proportions of forms, but in this work the term is used strictly in contrast with upright or roman styles.

OpenType is a digital font-file specification, managed jointly by Microsoft, Adobe, and other groups. Much of its contents overlaps with the TrueType specification.

Optical size refers to the intended rendered or printed size for which a typeface was designed. Many of the proportions and construction details of the typeforms may vary between a small optical size and a large optical size. Optical sizes are sometimes described by type designers in semantic terms (such as caption or headline), but some designers describe them in point size.

Overshoots and **undershoots** are regions of a typeform that go beyond the standard vertical measurement lines (e.g., above the x-height or below the baseline). Round forms and pointed vertices often undershoot or overshoot deliberately in order to achieve optical harmony with straight forms. It is generally understood that a form with undershoots or overshoots is still considered to be "on the baseline" or "at x-height".

Phototypesetting refers to printing made by exposing the images of typeforms onto light-sensitive media, either by flashing light through a film negative or by drawing the form with a cathode-ray tube (CRT) emitter. Either way, the use of photographic exposure meant that positioning of typeforms (including fitting and kerning) did not have the physical limitations of metal typesetting.

Point size is the measurement unit used to describe the physical size at which a typeface is printed or rendered. In CSS, a point is nominally defined as $^{1}/_{72}$ of one inch. In theory, the em square of a digital font is scaled so that it equals the point size defined for the text, but software and display devices make this relationship somewhat unreliable.

Profile refers to the outer shape of one side of a typeform (either the left or right side).

A **punch** was the steel tool into which a typeform was carved in metal typemaking. The punch was driven into a blank matrix to create the cavity into which the molten metal was eventually cast.

The term **script** and **writing system** are used in roughly the same way in this work, referring to the system of representation for a written or printed language. In other works and historical sources, the distinction between the two can become quite important; here both refer in general to an alphabet or other set of typeforms, regardless of language.

A **serif** in Latin type is the small, ancillary stroke attached to the ends of many main strokes in letterforms. Certain typeface styles can be difficult to classify as serif or sans-serif designs, because there is ambiguity as to when flared stroke endings begin to be called serifs.

SFNT is the underlying structure of contemporary digital fonts, including all TrueType, OpenType, and WOFF font formats.

The **sidebearing** of a form is distance from the outermost edge on one side of the form itself to the boundary width of the form on that side.

The **skeleton** or **construction** of a form is the arrangement of its main strokes, regardless of whether or not it includes serifs.

A **sort**, as used here, is a single unit of metal type, as cast in a mould. A sort may contain a single letterform, a ligature, or more.

Stem refers to a straight, vertical main stroke in Latin type. Stems are distinguished from bowls, diagonal strokes, and various connecting strokes in discussions of visual stem rhythm.

A **stroke** is any shape in a typeform that corresponds to linear gesture or mark made by a writing implement. Strokes can be large or small, straight or curved, but are typically distinguished from dots.

Style refers broadly to any design characteristics or motifs expressed in the typeforms and character of a typeface. In this work, the term is used to distinguish such creative facets of a typeface from facets that are more concretely assessed, such as letterform proportions, weight and width.

A **terminal** is the ending of a stroke; in some forms with open counters, such as `a` or `c`, the size and shape of the terminal has a major effect on how open or how closed the aperture is.

TrueType is a digital font-file specification maintained by Apple. Most, but not all, of its contents overlap with the OpenType specification.

Type 1 is a retired digital font-file specification maintained by Adobe.

UPEM or units per em, is the number of font units in the em of a particular font. The UPEM value is stored as a font-wide property in digital font files.

The terms **upright** and **roman** refer to typeface styles in Latin for which the main vertical strokes of the forms are perpendicular to the baseline. As with italic and oblique, some historical sources are concerned with more specific definitions, but in this work only the broad distinction is intended.

Variable fonts are digital font files that use recent enhancements to the OpenType and TrueType formats to effectively combine several members of a font family into a single file. Typically a variable font will incorporate several weights, widths, or optical sizes of a typeface, and settings allow users to choose between them when typesetting, or to adjust the parameters to values in between.

WOFF and **WOFF2** are compressed file formats encoding OpenType or TrueType fonts. The formats are optimized for web browser usage, but retain all of the core information about the typeforms.

The **weight** of a typeface is a descriptive term communicating the heaviness of the strokes. The typical stroke widths of typefaces meant for continuous reading are often referred to as 'regular', with other variants named in comparison: thinner variants called 'light' and thicker variants called 'bold', etc., but such terms by themselves do not have formal definitions.

The **width** of a typeface is a descriptive term communicating the relative horizontal and vertical proportions of the letterforms. Typical proportions in typefaces meant for continuous reading are often referred to as 'normal' or a similar term, and other proportions are described relative to that: 'narrow', 'wide', 'condensed', 'extended', etc. As with weight, however, such terms do not have formal definitions.

The **x-height** of a Latin typeface is the height of the tops of lowercase letterforms above the baseline, not including any ascenders. The letterform `x` is the standard reference because it has no ascender and its top typically does not overshoot. However, the measurement is generally understood to refer to a standard, font-wide value; forms with overshoots are considered to still be at the same x-height as the others.

Bibliography

1. Works cited

Adobe Type. 'AFDKO', 28 September, 2010. http://www.adobe.com/content/dam/Adobe/en/devnet/opentype/afdko/ Accessed via https://web.archive.org/web/20100928011722/http://www.adobe.com/content/dam/Adobe/en/devnet/opentype/afdko/

Apple Inc. 'Fonts - TrueType Reference Manual'. *Apple Developer*. Accessed 27 September, 2023. https://developer.apple.com/fonts/TrueType-Reference-Manual/

Ahrens, Tim, and Shoko Mugikura. *Size-Specific Adjustments to Type Designs: An Investigation of the Principles Guiding the Design of Optical Sizes*, Just Another Foundry, 2014.

Arrighi, Ludovico Vicentino degli. *La Operina*, 1522.

Baines, Phil, and Andrew Haslam. *Type and typography*. Second edition. London: Laurence King, 2005.

Banjanin, Bojan, and Uroš Nedeljković. 'Comparing different letter spacing methods in sans-serif typeface design'. In *Proceedings of the 7th Internationsl Symposium on Graphic Engineering and Design (GRID14)*. GRID. Novi Sad, Serbia: University of Novi Sad, Faculty of Technical Sciences, Department of Graphic Engineering and Design, 2014.

———. 'Sidebearings Analysis of Alphabet Letters with Complex Shape'. University of Novi Sad, Faculty of Technical Sciences, Department of Graphic Engineering and Design, 20 November, 2014.

Baruchel, Thomas. 'KERNDICT'. *CTAN: Package T1infos,* 2002. https://ctan.org/pkg/t1infos

Beier, Sofie, Chiron A. T. Oderkerk, Birte Bay, and Michael Larsen. 'Increased Letter Spacing and Greater Letter Width Improve Reading Acuity in Low Vision Readers'. *Information Design Journal* 26, no. 1 (19 July, 2021): 73–88. https://doi.org/10.1075/idj.19033.bei

Beier, Sofie. *Type Tricks: Your Personal Guide to Type Design*. Amsterdam: BIS Publishers, 2017.

Bergsland, David. *Practical Font Design with FontLab 5*. Mankato, MN: Radiqx Press, 2016.

Berkson, William. 'Reviving Caslon'. *I Love Typography*, 26 July, 2010. https://ilovetypography.com/2010/07/26/reviving-caslon-the-snare-of-authenticity/

Bigelow, Charles, and Jonathan Seybold. 'Aesthetics vs Technology: Does Digital Typesetting Mean Degraded Type Design?' *The Seybold Report* 10, no. 24 (24 August, 1981).

———. 'The Font Wars, Part 1'. *IEEE Annals of the History of Computing* 42, no. 1 (1 January, 2020): 7–24. https://doi.org/10.1109/MAHC.2020.2971202

———. ‘The Font Wars, Part 2’. *IEEE Annals of the History of Computing* 42, no. 1 (1 January, 2020): 25–40. https://doi.org/10.1109/MAHC.2020.2971745

Blacker, Elwyn, and Michael Blacker. ‘Spoiled for choice’. In Sassoon, Rosemary, ed. *Computers and Typography*. Oxford, England: Intellect, 1993:69–74.

Blokland, Frank E, and Simon Cozens. ‘Understanding the LeMo Cadence Grid’. *TypeDrawers*. Accessed 11 March, 2019. http://typedrawers.com/discussion/1140/understanding-the-lemo-cadence-grid.

Blokland, Frank E. *LetterModeller Manual*. Dutch Type Library, Autumn 2016.

———. ‘On the origin of patterning in movable Latin type: Renaissance standardisation, systematisation, and unitisation of textura and roman type’. Thesis. Leiden University, 2016.

Blumenthal, Joseph. ‘The Fitting of Type’. In *The Dolphin*, 2:71–81. New York, N.Y.: Limited Editions Club, 1935.

Březina, David. ‘Coherence in Typeface Design: Visual Similarity of Characters in Cyrillic, Devanagari, and Latin’. Thesis. University of Reading, 2018.

Briem, Gunnlaugur SE. ‘Notes on Type Design’. *FontLab 8*, 1998. https://help.fontlab.com/fontlab/8/tutorials/briem/

Browne, Cameron Bolitho, Michael Richard Arnold, and Paul Quentin Scott. *Automatic kerning of text*. United States US6829748B1, filed 26 November, 1999, and issued 7 December, 2004. https://patents.google.com/patent/US6829748B1/en

Buerkle, Brandon. ‘Spacing a Font, Part 1’. *Society of Fonts* (blog), 19 September, 2018. https://www.societyoffonts.com/2018/09/19/spacing-a-font-part-1/

———. ‘Spacing a Font, Part 2’. *Society of Fonts* (blog), 26 September, 2018. https://www.societyoffonts.com/2018/09/26/spacing-a-font-part-2/

Burkholder, Dave. ‘Device-Detector: Python3 Port of Matomo’s Device Detector’. Accessed 1 June, 2023. https://github.com/thinkwelltwd/device_detector

Burnhill, Peter. *Type Spaces: In-House Norms in the Typography of Aldus Manutius*. London: Hyphen Press, 2003.

Campe, Chris, and Ulrike Rausch. *Making Fonts: A Comprehensive Guide to Professional Type-Design*. Berkeley, California: Gingko Press Inc., 2022.

Carter, Harry Graham. ‘Optical Scale in Type Founding’. *Printing Historical Society Bulletin* 13 (September 1984).

———. *A View of Early Typography up to about 1600. Reprinted with an introduction by James Mosley*. London: Hyphen Press, 2002.

Celso, Alejandro Lo. ‘Rhythm in Type Design’. MATD dissertation, University of Reading, 2000.

Cheng, Karen. *Designing Type*. New Haven: Yale University Press, 2005.

Chung, Susana T L. ‘The Effect of Letter Spacing on Reading Speed in Central and Peripheral Vision’. *Investigative Ophthalmology & Visual Science* 43, no. 4 (1 April, 2002): 1270–76.

Coates, Daniel Robert. ‘Quantifying Crowded and Uncrowded Letter Recognition’. PhD dissertation, University of California at Berkeley, 2015.
Cost, Patricia. ‘The Contributions of Linn Boyd Benton and Morris Fuller Benton to the Technology of Typesetting and Typeface Design’. Thesis, Rochester Institute of Technology, 1986.
Cozens, Simon. ‘Spaceman: Just Space, Man’. *Simoncozens*, 2016. https://github.com/simoncozens/Spaceman
———. ‘Atokern’. *Simoncozens*, 12 December, 2018. https://github.com/simoncozens/atokern
———. ‘Fontmetrics Library’. *Simoncozens*, 24 March, 2019. http://simoncozens.github.io/fontmetrics-library/
———. ‘Countershape-Based Spacing and Kerning Library and Plugin’. *Simoncozens/CounterSpace*, 8 November, 2019. https://github.com/simoncozens/CounterSpace
———. ‘HT LetterKerner’. *GitHub gist*, 1 February 2019. https://gist.github.com/simoncozens/4f506dacb99d8e4b1f96f5fbfc102fa5
———. ‘Hands, Face, Space’. *GitHub gist*, 18 October, 2021. https://gist.github.com/simoncozens/03da7e5ad7f52af711948ed52a797e23
———. ‘Kern Determiner’. *Simoncozens*, 29 October, 2022. https://github.com/simoncozens/kerndeterminer
Cree, Edward. ‘Monokern’. *Ec429*, 2013. https://github.com/ec429/monokern
Descroix, Lucas. *Nostra v 0.3*, typeface specimen. 9 December, 2018. https://www.futurefonts.xyz/lucas-descroix/nostra?v=0.3
Dowzall, Martin E. *Indicia alignment device*. United States US4364184A, filed 11 June, 1980, and issued 21 December, 1982. https://patents.google.com/patent/US4364184A/
Dowzall, Martin Edward, and Vazgen John Houssian. *Apparatus for use in spacing letters*. European Union EP0180316A2, filed 23 September, 1985, and issued 7 May, 1986. https://patents.google.com/patent/EP0180316A2/
Dreyfus, John. ‘Watch this space’. In Clayton, Ewan, John Dreyfus, Gerald Fleuss, David M. Levy, Tom Perkins, and Sumner Stone. *Font: Sumner Stone, Calligraphy, and Type Design in a Digital Age*. Ditchling, Sussex: Edward Johnston Foundation and Ditchling Museum, 2000:31–34.
Dwiggins, William. *WAD to RR: A Letter about Designing Type*. Cambridge, Massachusetts: Harvard College Library, Department of Printing and Graphic Arts, 1940.
———. ‘Causerie On Fitting’, Letter to Chauncy H. Griffith, 72m31-02. Chauncey Hawley Griffith papers, 1903-1969. 10 May 1940, University of Kentucky Special Collections Research Center.
———. Letter to Chauncy H. Griffith, 72m31-02. Chauncey Hawley Griffith papers, 1903-1969. 18 August 1942, University of Kentucky Special Collections Research Center.

Ecker, Heinz-Dieter. 'HZ (Hermann Zapf)-Algorithm', 8 May, 1999. http://www.pvv.ntnu.no/~aslakr/hz.html

Egorov, Sergei. 'Aldine Italic in Fourier Plane', 16 March, 2005. http://www.malgil.com/esl/aldus-fft/

Eng, Torbjørn. 'InDesign, the Hz-Program and Gutenberg's Secret', 20 September, 2009. http://www.typografi.org/justering/gut_hz/gutenberg_hz_english.html

Espinoza, Ramiro et al. 'In Praise of Kernus 3.0' *TypeDrawers*, 24 January, 2016. http://typedrawers.com/discussion/1337/in-praise-of-kernus-3-0

FontForge. 'Auto Width and Auto Kern'. Accessed 11 March, 2019. http://fontforge.github.io/autowidth.html

FontTools. 'FontTools'. *FontTools*, 24 July, 2013. https://github.com/fonttools/fonttools.

Fournier, Pierre Simon. *Fournier on Typefounding. The Text of the Manuel Typographique, 1764-1766, Translated and Edited with Notes by Harry Carter*. Translated by Harry Graham Carter. London: Soncino Press, 1930.

Frömberg, Mark. 'Kernkraft'. March 8, 2019. https://github.com/BBoxType/Kernkraft

Frutiger, Adrian, Heidrun Osterer, and Philipp Stamm. *Adrian Frutiger - Typefaces: Complete Works*. Third edition. Basel: Birkhäuser, 2021.

Galliussi, Jessica, Luciano Perondi, Giuseppe Chia, Walter Gerbino, and Paolo Bernardis. 'Inter-Letter Spacing, Inter-Word Spacing, and Font with Dyslexia-Friendly Features: Testing Text Readability in People with and without Dyslexia'. *Annals of Dyslexia* 70 (14 March, 2020). https://doi.org/10.1007/s11881-020-00194-x

Gaultney, James Victor. 'Designing italics: Approaches to the design of contemporary secondary text typefaces'. Thesis, University of Reading, 2020.

Google Fonts. 'Analytics'. *Google Fonts*. Accessed 8 November, 2021. https://fonts.google.com/analytics

Grable, Joey. 'Joeygrable94/KernBot '. 2018. https://github.com/joeygrable94/KernBot

Grigas, Gintautas, and Anita Juškevičienė. 'Letter Frequency Analysis of Languages Using Latin Alphabet'. *International Linguistics Research* 1, no. 1 (26 March, 2018): p18–p18. https://doi.org/10.30560/ilr.v1n1p18

HT Letterspacer. 'HTLetterspacer on GitHub'. *Huertatipografica/HTLetterspacer*, 4 August, 2016. https://github.com/huertatipografica/HTLetterspacer

Haralambous, Yannis. *Fonts & Encodings*. First ed. Sebastopol, Calif: O'Reilly Media, 2007.

Harvey, Michael. *Creative Lettering Today*. New York: Published by Design Books, 1996.

Henestrosa, Cristobal, Laura Meseguer, and José Scaglione. *How to Create Typefaces: From Sketch to Screen*. Madrid: Tipo E Editorial, 2017.

Highsmith, Cyrus. *Inside Paragraphs: Typographic Fundamentals*. Second edition. New York: Princeton Architectural Press, 2020.

Hochuli, Jost. *Detail in Typography*. Translated by Charles Whitehouse. Paris: Éditions B42, 2015.

Hornus, Jeremie. 'Counter Shape https://t.co/ULZhXiiE4W'. Tweet. Twitter, 24 September, 2016. https://twitter.com/JeremieHornus/status/779754639403212801

Impallari, Pablo. 'Spacing Macro'. *Impallari Type*, 12 July, 2012. http://www.impallari.com/projects/overview/spacing-macro, Accessed 10 March, 2018 via https://web.archive.org/web/20180310055208/http://www.impallari.com/projects/overview/spacing-macro

Jamra, Mark. 'Basic Character Spacing in Type Design', 2004. https://typeculture.com/academic-resource/articles-essays/basic-character-spacing-in-type-design/

Karow, Peter et al. *Font Technology: Description and Tools*. Hamburg (Harksheider Strasse 102): URW Software & Type GmbH, 1994.

Karow, Peter, and Frank E Blokland. *Digital Typography & Artificial Intelligence*. The Hague: Dutch Type Library, 2013.

Karow, Peter, Bodo Kämmle, and Margret Albrecht. *Process for spacing of typefaces*. European Union EP0465704A1, filed 11 July, 1990, and issued 15 January, 1992. https://patents.google.com/patent/EP0465704A1/en

Karow, Peter, and John R. MacMillan. *Pointsize-variable character spacing*. United States US5937420A, filed 23 July, 1996, and issued 10 August, 1999. https://patents.google.com/patent/US5937420/en.

Karow, Peter. 'URW's Hz-Program'. *The Seybold Report on Publishing Systems* 21, no. 13 (30 March, 1992): 47.

———. *Typeface Statistics*. URW Edition No. 1. Hamburg: URW Verlag, 1993.

———. D*igital Typefaces: Description and Formats*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994. https://doi.org/10.1007/978-3-642-78105-6

———. 'Two Decades of Typographic Research at URW: A Retrospective'. In *Electronic Publishing, Artistic Imaging, and Digital Typography*, edited by Roger D. Hersch, Jacques André, and Heather Brown, 1375:265–80. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998. https://doi.org/10.1007/BFb0053276

———. 'Digital Typography with Hermann Zapf'. *TUGboat* 36, no. 2 (2015).

Kascenas, Antanas. 'Machine Learning of Fonts'. Master of Infomatics report. School of Informatics, University of Edinburgh, 2017.

Kindersley, David, and Neil Wiseman. *Spacing of characters in or in preparation for printing*. GB2004502, filed 20 September, 1978, and issued 24 March, 1982.

Kindersley, David. 'Towards Perfect Spacing'. *Book Design and Production* 5, no. 3 (1962): 178–83.

———. *An Essay in Optical Letter Spacing and Its Mechanical Application*. London: Wynkyn de Worde Society, 1966.

———. *Space Craft*. The Wynkyn de Worde Society Stationers Hall Meeting. Great Sutton St. London: Moore & Matthes, 1973.

———. *Optical Letter Spacing for New Printing Systems*. Second edition. Great Britain: The Wynkyn de Worde Society, 1976.

———. Letter to Neil Wiseman, 4 August 1976. Cambridge University Library, Department of Manuscripts and University Archives, Neil Wiseman papers, MS Wiseman.

———. Letter to Neil Wiseman, 16 September 1976. Cambridge University Library, Department of Manuscripts and University Archives, Neil Wiseman papers, MS Wiseman.

———. Letter to Colin Campbell, 6 September 1977. Cambridge University Library, Department of Manuscripts and University Archives, Neil Wiseman papers, MS Wiseman.

———. 'How Close?' *The Seybold Report on Publishing Systems* 16, no. 15 (13 April, 1987): 34.

Kindersley, David, and Lida Lopes Cardozo. *David Kindersley's Workshop: Letters in Stone & Glass and on Paper*. Rijksmuseum Meermanno-Westreenianum/Museum van Het Boek. 'S-Gravenhage: Staatsuitgeverij, 1987.

Kindersley, David, Lida Lopes Cardozo, and Francis Cave. *Optical Letter Spacing for New Printing Systems*. Third edition. Cambridge: Cardozo Kindersley, 2001.

Knuth, Donald Ervin. *The Art of Computer Programming. Volume 1: Fundamental Algorithms*. Addison-Wesley Series in Computer Science and Information Processing. Reading, Mass: Addison-Wesley, 1968.

Knuth, Donald Ervin. 'Foreword'. In Sedgewick, Robert, and Philippe Flajolet. *An Introduction to the Analysis of Algorithms*. Second edition. Upper Saddle River, NJ: Addison-Wesley, 2013.

Kolås, Øyvind. 'Kernagic'. *Hodefoting*, 26 April 2013. https://github.com/hodefoting/kernagic/

Kosch, Sebastian. 'Spacing Research', *Aldusleaf.Org*, 2010. http://aldusleaf.org/rhea.php, accessed via http://web.archive.org/web/20130326013609/http://aldusleaf.org/rhea.php

———. 'Fittingroom on GitHub' *Skosch/Fittingroom* 3 December, 2015. https://github.com/skosch/fittingroom

———. 'Towards an Attention-Based Letterfitting Model', 2020. https://skosch.github.io/electricbubble/

Käch, Walter. *Rhythm and Proportion in Lettering*. Translated by Elizabeth Friedlander. Olten und Freiburg im Breisgau: Walter-Verlag, 1956.

Kämmle, Bodo. 'Letter Spacing'. In Karow, Peter et al. *Font Technology: Description and Tools*. Hamburg (Harksheider Strasse 102): URW Software & Type GmbH, 1994:173–192.

Lee, Micah F, William Budington, and Dan Auerbach. 'Cryptolog'. *Electronic Frontier Foundation*, 2015. https://github.com/EFForg/cryptolog

Leonidas, Gerry. 'Designing Greek'. In Wittner, Ben, Sascha Thoma, and Timm Hartmann, eds. *Bi-Scriptual: Typography and Graphic Design with Multiple Script Systems*. 1st edition. Salenstein: Niggli, 2018.

Levien, Raph et al. 'David Kindersley on Spacing'. *Typophile*, 18 May, 2006. http://typophile.com/node/19789#comment-123868, accessed via https://web.archive.org/web/20120827024823/http://typophile.com/node/19789#comment-123868

Logan, David J. *Method and apparatus for automatically spacing characters during composition*. United States US4591999, issued 27 May, 1986.

Luckhurst, Raymond. 'Psoptkern'. *ScriptIt*, 2018. https://github.com/scriptituk/psoptkern

MacKay, Mark. 'Kern Type', 10 November, 2011. https://type.method.ac.

Mathews, M. *Apparatus and method for spacing or kerning typeset characters*. United States US3712443A, filed 19 August, 1970, and issued 23 January, 1973. https://patents.google.com/patent/US3712443A/en?oq=US3712443

Mello Vargas, Fernando de. 'Approaches to Applying Spacing Methods in Seriffed and Sans-Serif Typeface Designs'. MATD dissertation, University of Reading, 2007.

———. 'Journey into Spacing'. *TypeNotes* 2, 2018: 30–35.

Mengelt, Christian, and Roger D. Hersch, ed. 'Visual Aspects of Type'. In *Visual and Technical Aspects of Type*. Cambridge: Cambridge University Press, 1993:34–46.

Microsoft. 'OpenType Specification (OpenType 1.9)'. *Microsoft Typography*, 16 June, 2022. https://learn.microsoft.com/en-us/typography/opentype/spec/

Middendorp, Jan. *Dutch Type*. Reprint of 2004 original edition. Berlin: Druk Editions, 2018.

Moye, Stephen. *Fontographer: Type by Design*. 1st ed. New York: MIS:Press, 1995.

Naiman, Avi C. 'High-Quality Text For Raster Displays'. MSc thesis, University of Toronto, 1985.

Neville, Paul H., and William J. Fox. *Method and apparatus for automatic, shape-based character spacing*. United States US5803629A, filed 14 March, 1997, and issued 8 September, 1998. https://patents.google.com/patent/US5803629/en.

Nicholson, Ward. 'The Sector Kerning Debate'. *The Seybold Report on Publishing Systems* 19, no. 11 (26 February, 1990): 44–48.

Noordzij, Gerrit. 'A programme for teaching letterforms' In Fernand Baudin, and John Dreyfus. *ATypI Dossier A–Z 73*. Association Typographique Internationale, 1973.

———. *The Stroke of the Pen: Fundamental Aspects of Western Writing*. The Hague: Koninklijke Academie van Beeldende Kunsten, 1982.

———. *De handen van de zeven zusters*. Amsterdam: Van Oorschot, 2000.

———. *Letterletter: An Inconsistent Collection of Tentative Theories That Do Not Claim Any Other Authority than Common Sense*. Point Roberts, Calif: Hartley and Marks, 2000.

———. *The stroke: theory of writing*. English edition, translated by Peter Enneson. London: Hyphen Press, 2005.

Omagari, Toshi. *BubbleKern*. 1 September, 2015. http://tosche.net/2015/09/bubblekern.html

———. 'BubbleKern'. Presented at ATypI, 2016. Video published online 24 October 2016. https://www.atypi.org/type-typography/bubblekern

OpenAnnotation. 'Annotator v1.2.10'. *Openannotation/Annotator*, 26 February, 2015. https://github.com/openannotation/annotator/releases/tag/v1.2.10

Pallets Project. 'Flask Documentation (2.3.x)', 2010. https://flask.palletsprojects.com/en/2.3.x/

Perea, Manuel, Carmen Moret-Tatay, and Pablo Gómez. 'The Effects of Interletter Spacing in Visual-Word Recognition'. *Acta Psychologica* 137, no. 3 (July 2011): 345–51. https://doi.org/10.1016/j.actpsy.2011.04.003.

Perea, Manuel, Victoria Panadero, Carmen Moret-Tatay, and Pablo Gómez. 'The Effects of Inter-Letter Spacing in Visual-Word Recognition: Evidence with Young Normal Readers and Developmental Dyslexics'. *Learning and Instruction* 22, no. 6 (1 December, 2012): 420–30. https://doi.org/10.1016/j.learninstruc.2012.04.001

Perea, Manuel, and Pablo Gomez. 'Increasing Interletter Spacing Facilitates Encoding of Words'. *Psychonomic Bulletin & Review* 19, no. 2 (1 April, 2012): 332–38. https://doi.org/10.3758/s13423-011-0214-6

———. 'Subtle Increases in Interletter Spacing Facilitate the Encoding of Words during Normal Reading'. Edited by Kevin Paterson. *PLoS ONE* 7, no. 10 (17 October, 2012): e47568. https://doi.org/10.1371/journal.pone.0047568

Perktold, Josef, Skipper Seabold, Kevin Sheppard, ChadFulton, Kerby Shedden, Jbrockmendel, J-Grana6, et al. 'Statsmodels/Statsmodels: Release 0.14.0'. Zenodo, 5 May, 2023. https://doi.org/10.5281/ZENODO.593847

Petuchowski, Ethan. 'Font-Prediction_mahout'. *Ethanp*, 2014. https://github.com/ethanp/font-prediction_mahout

Ponting, Bob. 'Font Editor Now Creates Postscript Hints'. *InfoWorld*, 21 August, 1989.

Renckens, Maarten. 'RhythmInfluencer'. *Artengar/Glyphs_Plugins*, 3 December, 2020. https://github.com/Artengar/Glyphs_Plugins

Revolver Type Foundry. 'Tools'. *Revolver Type*, Accessed 11 March, 2019. https://www.revolvertype.com/tools/cadencer.html/

Reynolds, Linda, and Sue Walker. ''You Can't See What the Words Say': Word Spacing and Letter Spacing in Children's Reading Books'. *Journal of Research in Reading* 27, no. 1 (2004): 87–98. https://doi.org/10.1111/j.1467-9817.2004.00216.x

Rosendorf, Theodore. *The Typographic Desk Reference*. Second edition. New Castle, Delaware: Oak Knoll Press, 2016.

Ross, David Jonathan. *Fit*, typeface specimen. 2017. https://djr.com/fit

Rubinstein, Richard. *Digital Typography: An Introduction to Type and Composition for Computer System Design*. Reading, Mass.: Addison-Wesley Pub. Co., 1988.

SIL International. 'What Is Special About Awami Nastaliq?' *Awami Nastaliq*. 17 July, 2017. https://software.sil.org/awami/what-is-special/

Sassoon, Rosemary, ed. *Computers and Typography*. Oxford, England: Intellect, 1993.

Sassoon, Rosemary. *The Practical Guide to Lettering & Applied Calligraphy*. London: Thames and Hudson, 1987.

Sawada, Akira, Hiroshi Kumamoto, and Takashi Suzuki. *Apparatus and Method For Adjusting Character Spacing*. US Patent 5,501,538, issued 26 March, 1996.

Schaffzin, Gabi. 'OpticalLetterSpacing.Js'. *GabiSchaffzin*, 2012. https://github.com/GabiSchaffzin/OpticalLetterSpacing.js

Scheichelbauer, Rainer Erich. 'Kerning'. *Glyphs*, 28 September, 2021. https://glyphsapp.com/learn/kerning

———. 'Spacing'. *Glyphs*, 25 July, 2022. https://glyphsapp.com/learn/spacing

Schneider, Lukas. *LS Cadencer & LS Cadenculator*. Revolver Type Foundry, 2016.

Schwartz, Barry. 'Spacing by anchors'. *Sortsmill*, 14 October, 2021. https://github.com/chemoelectric/sortsmill/blob/90b97a9296582211a133970bb577013c9c86ed81/tools/spacing_by_anchors.py

Seybold, Jonathan, and Arlene E. Karsh. 'Composition Quality: Can URW 'One-Up' Gutenberg with Hz-Program?' *The Seybold Report on Publishing Systems* 22, no. 11 (22 February, 1993): 3–9.

Seybold, Jonathan. 'Kingsley/ATF Now Shipping Fonts'. *The Seybold Report on Publishing Systems* 4, no. 2 (16 October, 1989): 23.

———. 'Kerning on the Desktop'. *The Seybold Report on Publishing Systems* 18, no. 16 (22 May, 1989): 27–28.

———. 'URW Reoganizes; Karow, Rubow Depart'. *The Seybold Report on Publishing Systems* 24, no. 13 (13 March, 1995): 2.

———. 'URW Software for Better Justification'. *The Seybold Report on Publishing Systems* 21, no. 7 (6 December, 1991): 27.

Smeijers, Fred. *Counterpunch*. London: Hyphen Press, 1996.

———. 'Putting Letters next to Each Other: A Historical Overview'. *Vom Buch Auf Die Straße: Große Schrift Im Öffentlichen Raum, Journal der HGB*, no. 3 (2014): 227–250.

Sorkin, Eben, Jason Pagura, Vernon Adams, Ben Martin, Nathan Willis, and Molly Sharp. *Start Designing with FontForge: A Guide to Making Type*. Book Sprints, FontForge Press, 2012.

Southall, Richard. *Printer's Type in the Twentieth Century: Manufacturing and Design Methods*. London : New Castle, DE: British Library ; Oak Knoll Press, 2005.

Spells, Henry D. *System and method for automatically spacing characters*. US Patent 5,623.593, issued 22 April, 1997.

Spiekermann, Erik. *Rhyme & Reason: A Typographic Novel*. Translated by Paul Stiff. Berlin: H Berthold AG, 1987.

Thế Thành, Hàn. 'Micro-Typographic Extensions to the TEX Typesetting System'. Thesis. Masaryk University Brno, 2000.

Tracy, Walter. *Letters of Credit: A View of Type Design*. Second edition. Boston: D.R. Godine, 2003.

Type.tools. 'Machine Learning Kerning – The Kids Are Kerned Out'. *TypeLab 2020*. Typographics, 19 June, 2020. https://vimeo.com/431803660

TypeMyType. 'Introduction to Spacing'. *Robofont*, 2021. https://robofont.com/documentation/tutorials/spacing-intro/

URW Software & Type GmbH. *Hz-Program: Micro-Typography for Advanced Typesetting*. Hamburg, 1993.

Unger, Gerard. *While You're Reading*. New York: Mark Batty, 2007.

———. *Theory of Type Design*. Rotterdam: nai010 publishers, 2018.

Van Blokland, Petr. *Cambridge Super Vision report*. 29 July, 1986. Cardozo-Kinderlsey Workshop archives.

Watanabe, Kiyoshi. *Character processing apparatus capable of automatic kerning*. United States US5432890A, filed 7 October, 1994, and issued July 11, 1995. https://patents.google.com/patent/US5432890A/

Waxweiler, Nikolaus, and Andrés Torresi. 'Idea: Per-Glyph Parameters?', 2 May, 2018. https://github.com/huertatipografica/HTLetterspacer/issues/37

Wiegel, Peter. 'Peter Wiegel's Auto Spacing Idea'. *Ft-Devel*, 14 June, 2012. https://lists.nongnu.org/archive/html/freetype-devel/2012-06/msg00084.html

Wilkins, A J, Smith, J, Willison, C K, Beare, T, Boyd, A, Hardy, G, Mell, L, Peach, C, and Harper, S. (2007). Stripes within words affect reading. *Perception* 36, no. 12, p. 1788-1803.

The World Wide Web Consortium. 'W3C Standards and Drafts'. *W3C*, 26 September, 2023. https://www.w3.org/TR/

Yu, Deyue, Sing-Hang Cheung, Gordon E. Legge, and Susana T. L. Chung. 'Effect of Letter Spacing on Visual Span and Reading Speed'. *Journal of Vision* 7, no. 2 (February 1, 2007): 2. https://doi.org/10.1167/7.2.2

Yue, Xin. 'Yuex/Cjk-Auto-Spacing'. 2013. https://github.com/yuex/cjk-auto-spacing

Zapf, Hermann. 'About Micro-Typography and the Hz-Program'. *Electronic Publishing* 6, no. 3 (September 1993): 283–88.

'Zhspacing on Google Code'. Accessed 11 March, 2019. https://code.google.com/archive/p/zhspacing/

Łuniewska, Magdalena, Marta Wójcik, and Katarzyna Jednoróg. 'The Effect of Inter-Letter Spacing on Reading Performance and Eye Movements in Typically Reading and Dyslexic Children'. *Learning and Instruction* 80 (August 2022): 101576. https://doi.org/10.1016/j.learninstruc.2021.101576

2. Additional sources consulted in historical study

The historical study of letter-fitting literature, practice, and fitting-automation projects, described in chapter 2, § 2.2, also reviewed the following sources.

Ahrens, Tim. 'Kern On'. Accessed 18 June, 2021. https://kern-on.com/

Andersson, Rasmus. 'Working through Some Ideas for Machine Learning-Based Automatic Kerning'. Tweet. Twitter, 25 April, 2020. https://twitter.com/rsms/status/1254181634569416704.

Baltus, Viktor. 'A Revised Spacing Chart'. *Type Design Class*. Accessed 14 September, 2022. https://www.typedesignclass.com/resources/revised-spacing-chart

———. 'Spacing and Kerning Your Font'. *Type Design Class*. Accessed 14 September, 2022. https://www.typedesignclass.com/resources/spacing-and-kerning-basics

Bartels, Samuel A. *The Art of Spacing: A Treatise on the Proper Distribution of White Space in Typography*. Chicago: Inland Printer, 1926.

Bauermeister, Benjamin. *A Manual of Comparative Typography: The PANOSE System*. New York: Van Nostrand Reinhold, 1988.

Beier, Sofie. *Reading Letters: Designing for Legibility*. Amsterdam: BIS, 2012.

Berlow, David. Interview. Video, 25 May, 2023.

Bil'ak, Peter. 'Size-Specific Spacing of Fonts'. *Typotheque*, Accessed 4 June, 2021. https://www.typotheque.com/articles/size-specific_spacing_of_fonts

Blokland, Frank E. et al. 'The LeMo method', *Typedrawers*. October 2014. http://typedrawers.com/discussion/749/the-lemo-method

Blokland, Frank E. *LetterModeller 5.0*, 19 December 2014. http://www.lettermodel.org/wordpress/?page_id=13#LeMo

Bohm, Thomas. 'Micro-Typography: How To Space And Kern Punctuation Marks And Other Symbols'. *Smashing Magazine*, May 2020. https://www.smashingmagazine.com/2020/05/micro-typography-space-kern-punctuation-marks-symbols/.

Chen, Charles M. *Typefacet Autokern*, 21 October 2012. http://charlesmchen.github.io/typefacet/topics/autokern/index.html

Cozens, Simon et al. 'A New Approach to Spacing?' *TypeDrawers*. Accessed 11 March, 2019. http://typedrawers.com/discussion/1736/a-new-approach-to-spacing

Daines, Mike. 'Letters — Slate Cut'. *Baseline* 8, 1986, p.16–17.

Dutch Type Library. *KernMaster Manual*. Hamburg: 's-Hertogenbosch, 2004.

Dutch Type Library. 'History'. *DTL Font Tools*. Accessed 18 July, 2021. https://www.fontmaster.nl/history.html/.

FontLab, Ltd. *FontLab Studio 5.0.2 for Mac User Manual*. 2006.

Fontsmith. 'How to Space a Typeface'. 5 February, 2018. https://www.fontsmith.com/blog/2018/02/05/how-to-space-a-typeface.

Frutiger, Adrian, Maurice Besset, Emil Ruder, and Hans Rudolf Schneebeli. *Type, Sign, Symbol*. Zurich: ABC Edition, 1980.

Gates, David. *Lettering for Reproduction*. New York: Watson Guptill, 1976.

Gill, Eric. *An Essay on Typography*. London: Penguin Books, 2013.

Goudy, Frederic W. *The Alphabet; and Elements of Lettering: Revised and Enlarged with Many Full-Page Plates and Other Illustrations Drawn & Arranged by the Author*. New York: Dover, 1963.

Goudy, Frederic W. *Typologia: Studies in Type Design and Type Making - with Comments on the Invention of Typography : The First Types, Legibility and Fine Printing*. London: University of California Press, 1977.

Harradine, John. *On the Nature of the Fendragon*. Wellingborough: Skelton's Press, 1977.

Harvey, Michael. *Lettering Design Form & Skill in the Design & Use of Letters*. Barre, Mass: Barre, 1976.

Hess, Stanley. *The Modification of Letterforms*. New York: Art Direction Book Co., 1981.

Hudson, David. 'TypeButter'. *Hudsonfoo*, 13 July, 2012. https://github.com/hudsonfoo/typebutter

Hudson, John, Laurence Penney, et al. 'Laurence Penney's Kern-font proposal', *Comp.fonts*. 18–25 May 1996. https://groups.google.com/d/topic/comp.fonts/GEjTE9_H52M/discussion

Hudson, John. *Scholarly types*, 18 November 2009. Video published online 20 March 2012. https://www.youtube.com/watch?v=DT5Qn_Dhiqk

Invers Software. *TypeArt 2.0: Der mächtigste Fonteditor für Calamus-Fonts im CFN-Format*. Ulf Dunkel, 2007. Accessed via https://www.calamus.net/les/dl.php?File=TA2deMan.pdf

Jacoby, Henry Sylvester. *A Text-Book on Plain Lettering*. New York, The Engineering News Publishing Company, 1901. http://archive.org/details/textbookonplainl00jacorich

Jamra, Mark. 'Form and Proportion in a Text Typeface: A Few Guidelines'. *TypeCulture*, 2003. https://typeculture.com/wp-content/uploads/2016/02/tc_article_7.pdf

Johnston, Edward. *Writing & Illuminating & Lettering*. New York: Dover Publications, 1995.

Joss, Molly W, Jonathan Seybold, and Andrew Tribute. 'Harlequin's Automatic Kerning'. *The Seybold Report on Publishing Systems* 20, no. 8 (7 January, 1991): 16.
Kalen, Dean. 'Building a Spacing Calculator'. *TypeWknd*, 2021. Video published at https://www.youtube.com/watch?v=ZS8WqPxGoAk
Karow, Peter. 'Extending control of digital typography', *Visible language* XXXII, no. 2 (1998), p. 100–127.
Kindel, Eric. 'A Reconstruction of Stencilling Based on the Description by Gilles Filleaudes Billettes'. In *Typography Papers* 9. Hyphen Press, 2014, p. 28–65.
Kindersley, David. 'Optical letter spacing', *The Penrose annual*, Volume 62 (1969), p. 167–176.
———.'Eye spacing: are letters black or white?', *ATypI summary report of the second working seminar on the teaching of letterforms*. Department of Typography & Graphic Communication, University of Reading, 1976, p. 16–20.
Kindersley, David, and Lida Lopes Cardozo. *Letters Slate Cut*. Cambridge: Cardozo Kindersley, 1990.
Kindersley, David and Neil Wiseman. 'Computer-aided letter design', *Printing World* 202, no. 44 (31 October 1979): 13–17.
Kosch, Sebastian. 'Fitting letters with nonnegative matrix factorization', 28 November 2015. https://aldusleaf.org/fitting-letters-with-nonnegative-matrix-factorization/
Leonidas, Gerry. 'A few things about typeface design'. *Letter Exchange Forum* 18, September 2009.
———. Interview. Video, 24 May, 2023.
Levien, Raph et al. 'The Bouma of Space Craft'. *Typophile*, 7 April, 2005. http://www.typophile.com/node/10191
Lucas, Alex John. 'The Fundamentals of Spacing a Typeface a Typeface'. *Alex John Lucas a Typeface Designer*. 14 April, 2019. https://alexjohnlucas.com/type/spacing-a-typeface
Lücke, Karsten, URW++, and Dutch Type Library. *DTL OTMaster Manual*. Hamburg: 's-Hertogenbosch, 2016.
Macrakis, Michael S, and Greek Font Society, eds. *Greek Letters: From Tablets to Pixels*. New Castle, Del: Oak Knoll Press, 1996.
Macromedia. 'Using Fontographer 4.1: Metrics — Spacing and Kerning'. In *Using Fontographer*, 4th ed., 143–52. San Francisco, CA, 1996.
Marini, Igino. *iKern*, 2008. http://ikern.com
——. *The iKern theory*, 8 January 2011. http://ikern.com/k1/ikern/the-ikern-theory/
——. *Introduction to iKern*, 7 February 2016. http://ikern.com/k1/wp-content/uploads/iKern_Intro_2016_02_07.pdf
Marini, Igino, Jérémie Hornus, Nathan Willis, Tim Ahrens, and Toshi Omagari. 'Automated Kerning and Spacing: Present and Future'. Presented at *ATypI Tech Talks*, 29 October 2022.

McQueen, Clyde D, and R G Beausoleil. 'Infinifont: A Parametric Font Generation System'. *Electronic Publishing* 6, no. 3 (September 1993): 117–32.

Mills, Ross, and John Hudson. *Mathematical typesetting*. Canada: Microsoft Corporation, 2007.

Mitchell, Terena Frederick. *Writing Arabic: A Practical Introduction to Ruq'ah Script*. London; New York: Oxford University Press, 1970.

Mulvey, Frank. *Graphic Perception of Space*. New York; London: Reinhold Book Corp. ; Studio Vista, 1969.

Nemeth, Titus. 'On Arabic Justification, Part 1 – a Brief History'. *TypoArabic*, 15 November, 2019. https://research.reading.ac.uk/typoarabic/on-arabic-justification-part-1/.

———. 'On Arabic Justification, Part 2 – Software Implementations'. *TypoArabic*, 18 December, 2019. https://research.reading.ac.uk/typoarabic/on-arabic-justification-part-2-software-implementations/.

———. 'On Arabic Justification, Part 3 – Historical Models'. *TypoArabic*, 11 March, 2020. https://research.reading.ac.uk/typoarabic/on-arabic-justification-part-3-historical-models/.

Nemeth, Titus, ed. *Arabic Typography: History and Practice*. 1st edition. Salenstein: Niggli, 2023.

Nicholson, Ward. 'URW's Kernus; auto kern-pair generators; sector kerning', *Comp.fonts*. 1 January 1996. https://groups.google.com/d/msg/comp.fonts/6AWQ1JEGzRE/4VzJ7G_-_HUJ

Omagari, Toshi. *BubbleKern manual*, 5 September 2015. https://github.com/Tosche/BubbleKern/blob/master/BubbleKern%20Manual.pdf

——. 'BubbleKern', *Tosche*. Initial commit 1 September 2015. https://github.com/Tosche/BubbleKern/

Peters, Yves. 'Adventures in Space: Kerning'. *FontShop*, 23 February, 2016. https://www.fontshop.com/content/adventures-in-space-kerning.

———. 'Adventures in Space: Spacing'. *FontShop*, 4 February, 2016. https://www.fontshop.com/content/adventures-in-space_spacing.

———. 'Adventures in Space: Special Cases'. *FontShop*, 6 April, 2016. https://www.fontshop.com/content/adventures-in-space-special-cases.

———. 'Adventures in Space: Tracking'. *FontShop*, 25 April, 2016. https://www.fontshop.com/content/adventures-in-space_tracking.

———. 'Adventures in Spacing: Spaces'. *FontShop*, 14 June, 2016. https://www.fontshop.com/content/adventures-in-space_spaces.

Phillips, Arthur. 'Guide to Mathematical Composition: Spacing'. *The Monotype Recorder* 40, no. 4 (Winter 1956): 12–15.

Phinney, Thomas. *About Adobe's optical kerning*, 2010. http://blog.extensis.com/adobe/about-adobe%E2%80%99s-optical-kerning.php

Phinney, Thomas. 'On Kerning (and Spacing) Fonts'. *Phinney on Fonts*, 14 January, 2014. https://www.thomasphinney.com/2014/01/kerning-and-spacing-fonts/.

Quin, Liam et al. ‘Re: f-ligature frustrations’, *Comp.fonts*. 14 May 1996. Accessed via https://groups.google.com/d/msg/comp.fonts/GGPtw0gROY0/jSkHZljDPvMJ

Re, Margaret, et al., *Typographically speaking: the art of Matthew Carter*. Baltimore, Maryland: Princeton Architectural Press, 2003.

Salkic, Zlatan. ‘DecoType’s Advanced Composition Engine’. *Medium* (blog), 22 February, 2023. https://zlatansalkic.medium.com/decotypes-advanced-composition-engine-dc22a9cbd912

Samara, Timothy. *Letterforms: Type Design from Past to Future*. Beverly, MA: Rockport, 2018.

Scheichelbauer, Rainer Erich. Interview. Video, 10 August, 2023.

Schenk, Walter. *Die Schriften des Malers*. Gießen: Fachbuchverlag Dr Pfannenberg & Co, 1958.

Shahn, Ben. *Love and Joy about Letters*. London: Cory, Adams & Mackay, 1964.

Shaw, Paul. ‘The Tricky Art of Letter-Spacing’. *Salon*, 14 November, 2011. https://www.salon.com/2011/11/14/letter_spacing_imprint/

Shaw, Paul, ed. *The Eternal Letter: Two Millennia of the Classical Roman Capital*. Cambridge, Massachusetts: The MIT Press, 2015. p.40–47.

Skala, Matthew. *Electric kerning*, 9 December 2010. http://ansuz.sooke.bc.ca/entry/133

———. *Tsukurimashou project*, last updated 11 December 2015. http://tsukurimashou.osdn.jp/

———. *Tsukurimashou user’s manual*, version 0.9, 17 August 2014. http://tsukurimashou.osdn.jp/tsukurimashou.pdf

Slinn, Judy, Sebastian Carter, Richard Southall, Andrew Boag, and Monotype Corporation, eds. *History of the Monotype Corporation*. London: Printing Historical Society [u.a.], 2014, p. 308–315.

Sousa, Miguel. ‘Spacing method’, *Typophile*. 22 October 2005. http://www.typophile.com/node/15794

Thiemich, Thomas. ‘Die Andere Seite Der Schrift’. Unpublished research paper, Hochschule für Grafik und Buchkunst, 2018.

Thompson, Tommy. *How to Render Roman Letter Forms: A Pattern to Understanding and Drawing Roman Letters and Other Styles of Lettering and Type Faces Related to Them*. New York: American Studio Books, 1946.

TypeNetwork. ‘Spacing Primer | Variable Fonts’. Accessed 14 August, 2019. https://variablefonts.typenetwork.com/topics/spacing/

Vane, Olivia. “Timeline Design for Visualising Cultural Heritage Data.” Thesis, Royal College of Art, 2019.

Wardle, Tiffany. ‘Experimental typefaces of William Addison Dwiggins: Falcon, Charter, Arcadia, and Stuyvesant’. MATD dissertation, University of Reading, 2000.

Weber, Hendrik. *Italic: What Gives Typography Its Emphasis*. Translated by Dan Reynolds. 1st English edition. Salenstein: Niggli, 2021.

Wichary, Marcin. ‘Space Yourself’. *Smashing Magazine*, 21 October, 2015. https://www.smashingmagazine.com/2015/10/space-yourself/

Wilme, B.P. *A hand-book for mapping, engineering, and architectural drawing*. London: John Weale, 1846.

Willis, Nathan. ‘Conceptual approaches to automated letter fitting’. MATD dissertation, University of Reading, 2017.

Willis, Nathan. ‘Four Takes on the Problem of Spacing Automation’. Presented at *TypeCon 2019*, 1 September 2019.

Willis, Nathan. ‘Thinking About Talking About Spacing’. *Herb Lubalin Lecture Series*, Cooper Union, 20 September 2020.

Zapf, Hermann. *Hermann Zapf and His Design Philosophy*. Chicago: Society of Typographic Arts, 1987.

Zhukov, Maxim. ‘The Peculiarities of Cyrillic Letterforms’. *Typography Papers*, no. 1 (March 1996).

Appendix A: mathematical and statistical notes

i. Calculation of moments in LOGOS

As discussed in chapter 4, the LOGOS software determined the centrepoint of a letterform by:

1. Splitting the letterform into two halves
2. Finding the centroid of each half
3. Calculating the fourth polar moment of each half, with respect to the centroid of that half
4. Comparing the two moments
5. Repeating the above steps 1–4, splitting the letterform in a different spot
6. Stopping the process when the two moments compared are equal

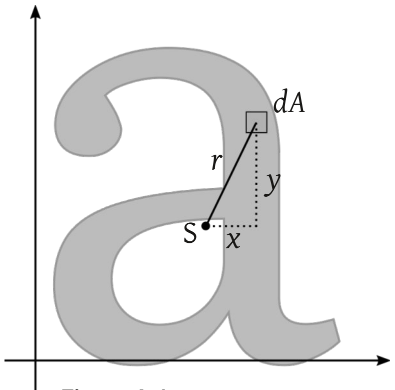


Figure A.1
A duplication of figure 4.3. The polar moments of area are integrals over the entire form, based on the distance *r* measured from the centroid of the form, *S*. In the fourth polar moment, the quantity integrated is r^4. The centroid of the form must be found first, adding computational complexity (illustration by the author).

The generic formula for the fourth polar moment, converted to *x* and *y* by the Pythagorean theorem *(see figure A-1)*, is:

$$\iint_A r^4\, dA = \iint_A (x^2 + y^2)^2\, dA$$

which can be expanded to

$$\begin{aligned} \iint_A r^4\, dA &= \iint_A (x^2 + y^2)^2\, dA \\ &= \iint_A (x^2 + y^2)(x^2 + y^2)\, dA \\ &= \iint_A (x^4 + 2x^2 y^2 + y^4)\, dA \\ &= \iint_A x^4\, dA + 2 \cdot \iint_A x^2 y^2\, dA + \iint_A y^4\, dA \end{aligned}$$

Crucially, each of these three integrals is a planar moment:

$$\begin{aligned} I_{yyyy} &= \iint_A x^4\, dA \\ I_{xxyy} &= \iint_A x^2 y^2\, dA \\ I_{xxxx} &= \iint_A y^4\, dA \end{aligned}$$

allowing the fourth polar moment to be expressed more compactly in those terms:

$$\iint_A r^4\, dA = I_{xxxx} + 2 \cdot I_{xxyy} + I_{yyyy}$$

Practically speaking, this is a useful result because the FontTools Python library provides a `MomentsPen` module designed for calculating various moments, directly on the Bézier curves of the glyphs in a TrueType or OpenType digital font. With modification, the `MomentsPen` module was used to calculate fourth polar moments for the composite algorithm developed in chapter 4. The modifications used are listed in appendix B.

ii. Computation of canonical rectangles in LOGOS

Naturally, the fact that the approach above is workable for the fourth polar moments required by LOGOS suggests that the other components in the original LOGOS method should also be explored. Specifically, the canonical-rectangle component of the original LOGOS also utilized fourth polar moments.

As a reminder, the canonical rectangle component determined a new width for each letterform in a typeface by calculating the fourth polar moments of a set of rectangular shapes, then matching each letterform to a rectangle that produced the same calculated result.

The rectangles used in this technique were bespoke for each typeface: they had the same height as the letterforms, and had vertical and horizontal strokes the same thickness as the vertical and horizontal strokes of the letterforms. The simpler construction of the rectangles makes the moments computations far more straightforward than for the arbitrarily complicated Bézier curves of a letterform: each is a rectangle of height h and width w, with a smaller rectangle of height h' and width w' removed from its centre. *(See figure A.2)* The moment for the canonical rectangle is thus the moment of the outer rectangle minus the moment of the inner rectangle.

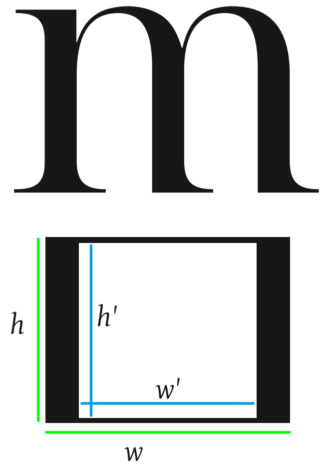


Figure A.2
A hypothetical canonical rectangle in the LOGOS method. The thicknesses of the vertical and horizontal sides of the rectangle must be chosen to match the vertical and horizontal stroke thicknesses of the letterform. The goal of the technique is to find a width *w* such that the rectangle's moment matches that of the letterform (illustration by the author).

Recall from the previous section that the fourth polar moment is found with the formula:

$$\iint_A r^4\, dA = I_{xxxx} + 2 \cdot I_{xxyy} + I_{yyyy}$$

For the outer and inner rectangles, these integrals can be found by plugging in h, w, h', and w'. For the outer rectangle:

$$\begin{aligned} I_{xxxx} &= \iint_A y^4\, dA \\ &= \int_{-\frac{h}{2}}^{\frac{h}{2}} y^4 \cdot w\, dy \\ &= \frac{w}{5}\left(\left(\frac{h}{2}^5\right) - \left(\frac{-h}{2}^5\right)\right) \\ &= \frac{w}{5}\left(\left(\frac{h^5}{32}\right) + \left(\frac{h^5}{32}\right)\right) \\ &= \frac{w h^5}{80} \end{aligned}$$

The integral for I_{yyyy} is similar to that of I_{xxxx}, but with *w* and *h* trading places:

$$I_{yyyy} = \iint_A x^4 \, dA$$
$$= \frac{w^5 h}{80}$$

The integral for I_{xxyy} is more involved, but also solvable as an equation:

$$I_{xxyy} = \iint_A x^2 y^2 \, dA$$
$$= \int_{-\frac{h}{2}}^{\frac{h}{2}} \int_{-\frac{w}{2}}^{\frac{w}{2}} x^2 y^2 \, dx \, dy$$
$$= \int_{-\frac{h}{2}}^{\frac{h}{2}} \frac{1}{3}\left(\left(\frac{w}{2}\right)^3 - \left(\frac{-w}{2}\right)^3\right) \cdot y^2 \, dy$$
$$= \int_{-\frac{h}{2}}^{\frac{h}{2}} \frac{1}{3}\left(\frac{w^3}{8} + \frac{w^3}{8}\right) \cdot y^2 \, dy$$
$$= \frac{w^3}{12} \int_{-\frac{h}{2}}^{\frac{h}{2}} y^2 \, dy$$
$$= \frac{w^3}{12} \cdot \frac{1}{3}\left(\left(\frac{h}{2}\right)^3 - \left(\frac{-h}{2}\right)^3\right)$$
$$= \frac{w^3}{12} \cdot \frac{1}{3}\left(\frac{h^3}{8} + \frac{h^3}{8}\right)$$
$$= \frac{w^3 h^3}{144}$$

As a result, calculating the value of the expression for the outer rectangle requires only plugging in *w* and *h*:

$$I_{xxxx} + I_{yyyy} + 2 \cdot I_{xxyy} = \frac{w h^5}{80} + \frac{w^5 h}{80} + \frac{w^3 h^3}{72}$$

The inner rectangle works the same way for *w'* and *h'*, and the moment for the entire canonical rectangle is found by subtracting them:

$$M(rect) = \frac{w h^5}{80} + \frac{w^5 h}{80} + \frac{w^3 h^3}{72} - \left(\frac{w' h'^5}{80} + \frac{w'^5 h'}{80} + \frac{w'^3 h'^3}{72}\right)$$

The difficulty of the method arises from the need to solve this equation for the canonical width, *w*. Most of the other variables, including *M*, *h*, *h'*, and *w'* are known from the letterform. The height of the canonical rectangle *h* is the height of the letterform. The thicknessess of the vertical and horizontal strokes in the letterform are both known, and those give *h'* and give the *difference* between *w* and *w'*:

$$h' = (h - 2s_h)$$
$$w' = (w - 2s_v)$$

and the value of *M* is meant to be the same as that of the moment calculated separately for the letterform. The final equation is algebraic, but of the fifth order in the variable of interest, *w*:

$$M(rect) = \frac{wh^5}{80} + \frac{w^5 h}{80} + \frac{w^3 h^3}{72} - \frac{(w-2s_v)(h-25s_h)^5}{80} - \frac{(w-2s_v)^5(h-25s_h)}{80} - \frac{(w-2s_v)^3(h-25s_h)^3}{72}$$

Thus, it cannot be solved for *w*, even with the other values inserted.

If one could compute *w* directly during the fitting process, then the width for each letterform could be calculated in a single function call. The fact that the equation does not permit this explains why the original LOGOS method chose, instead, to pre-compute moments for a range of different widths and find a match by using a table look-up.

iii. Linear regression model for standard inter-letter-area default values in composite algorithm

As discussed in chapter 4, § 4.4.2, a linear regression analysis was conducted on a sample of the top 100 most-used Latin text fonts from the Google Fonts library, with the goal of establishing a neutral default setting for the standard inter-letter-area parameter for the composite fitting algorithm. The intent of the analysis was to determine a default value for the ratio between the internal counter area of the key letter `n` and the standard inter-letter area, derived from each font's measurements.

For the analysis, six measurements were made on each font in the sample:

- `counter`: the width of internal counter of `n`
- `stroke`: the width of the vertical stroke of `i`
- `contrast`: the ratio of the thinnest and thickest strokes of `o`
- `xh`: the x-height
- `serif`: the length of the serif of `n`
- `bearing`: the left sidebearing of `n` + the right sidebearing of `n`

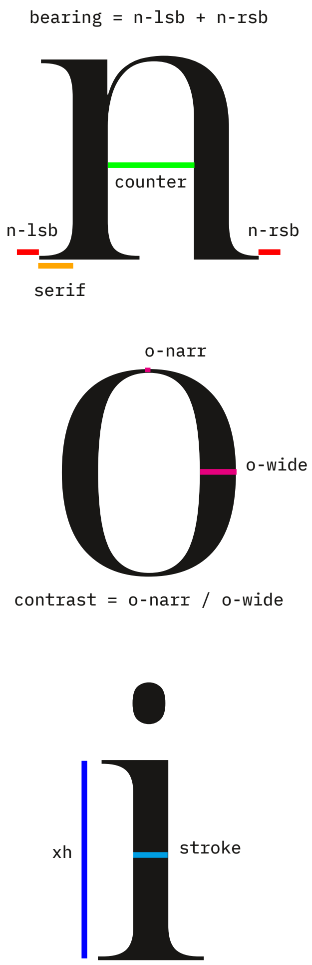


Figure A.3
Measurements made on glyphs from the top 100 most-used Latin text fonts in the Google Fonts library (illustration by the author).

Each of the above raw measurements was divided by the `UPEM` value of the font, in order to normalize the data across the sample set. The choice of measurements was based on the measurement techniques used in the PANOSE system (Bauermeister 1988) and in Karow's *Schriftstatistick* project (Karow 1993). Illustrations of these measurements are shown in figure A.3. Some of the specifics (such as measuring the stroke-width on i rather than on n) were chosen in order to automate the process for batch measurement of the set.

An ordinary least squares multiple regression analysis was then performed using the statsmodels Python library (Perktold et al., 2023), with `bearing` as the dependent variable. The output of that analysis is reproduced below.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:                bearing   R-squared:                       0.709
Model:                            OLS   Adj. R-squared:                  0.708
Method:                 Least Squares   F-statistic:                     480.9
Date:                Wed, 03 May 2023   Prob (F-statistic):         2.37e-261
Time:                        08:59:21   Log-Likelihood:                 2197.1
No. Observations:                 991   AIC:                            -4382.
Df Residuals:                     985   BIC:                            -4353.
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const          0.0293      0.011      2.642      0.008       0.008       0.051
counter        0.3292      0.018     18.175      0.000       0.294       0.365
stroke        -0.1130      0.026     -4.300      0.000      -0.165      -0.061
contrast      -0.0006   8.29e-05     -7.069      0.000      -0.001      -0.000
xh             0.1119      0.020      5.706      0.000       0.073       0.150
serif         -1.0223      0.032    -31.510      0.000      -1.086      -0.959
==============================================================================
Omnibus:                      106.804   Durbin-Watson:                   0.876
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              466.693
Skew:                          -0.406   Prob(JB):                    4.56e-102
Kurtosis:                       6.262   Cond. No.                         420.
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
```

Appendix B: software source code

i. Moments for LOGOS reimplmentation

This code extends the FontTools `MomentsPen` module to add polar moments as required by the LOGOS reimplementation.

```
# Imports
from fontTools.pens.statisticsPen import StatisticsPen
from fontTools.misc.symfont import x, y, printGreenPen
from fontTools.pens.basePen import BasePen
from fontTools.pens.momentsPen import MomentsPen

# This function auto-generates an extension of the built-in MomentsPen,
# using the symfont symbolic algebra to build numerical approximations
# of the integrals via Green's Theorem. It *must* be run and the resulting
# NewMomentsPen class definition *must* declared before the
# statisticsPen can be defined in the next stage.
#
# The auto-generated NewMomentsPen is extremely long, consisting of
# polynomial expansions for the numerical approximations for different
# Bezier curve types. This is as expected.
# Note that the function-naming convention differs from the default
# MomentsPen.
printGreenPen('NewMomentsPen', [
      ('area', 1),
      ('Planar1stMomentWrtX', y),
      ('Planar1stMomentWrtY', x),
      ('Planar2ndMomentWrtX', y**2),
      ('Planar2ndMomentWrtY', x**2),
      ('Planar3rdMomentWrtX', y**3),
      ('Planar3rdMomentWrtY', x**3),
      ('Planar4thMomentWrtX', y**4),
      ('Planar4thMomentWrtY', x**4),
      ('ProductMomentXY', x*y),
      ('ProductMomentXXY', x*x*y),
      ('ProductMomentXYY', x*y*y),
      ('ProductMomentXXYY', x*x*y*y),
     ])

# This class creates a NewStatisticsPen that can return the polar moments
# of TTGlyphs as required by the LOGOS reimplementation
class NewStatisticsPen(NewMomentsPen):
        """Pen calculating area, center of mass, variance and
        standard-deviation, covariance and correlation, 0th to 4th
        planar moments, four product moments, 2nd and 4th polar
        moments, and slant, of glyph shapes.
        Note that all the calculated values are 'signed'. Ie. if the
        glyph shape is self-intersecting, the values are not correct
        (but well-defined). As such, area will be negative if contour
        directions are clockwise.  Moreover, variance might be negative
        if the shapes are self-intersecting in certain ways."""

        def __init__(self, glyphset=None):
                NewMomentsPen.__init__(self, glyphset=glyphset)
                self.__zero()
```

```
def _closePath(self):
        NewMomentsPen._closePath(self)
        self.__update()

def __zero(self):
        self.meanX = 0
        self.meanY = 0
        self.varianceX = 0
        self.varianceY = 0
        self.stddevX = 0
        self.stddevY = 0
        self.covariance = 0
        self.correlation = 0
        self.slant = 0
        # Backward-compatibility properties
        # Do not rely on these in new code.
        self.momentX = 0
        self.momentY = 0
        self.momentXX = 0
        self.momentYY = 0
        self.momentXY = 0
        # Polar 2nd and 4th moments
        self.Polar2ndMoment = 0
        self.Polar4thMoment = 0

def __update(self):

        area = self.area
        if not area:
                self.__zero()
                return

        # Center of mass
        # https://en.wikipedia.org/wiki/Center_of_mass#A_continuous_volume
        self.meanX = meanX = self.Planar1stMomentWrtY / area
        self.meanY = meanY = self.Planar1stMomentWrtX / area

        #  Var(X) = E[X^2] - E[X]^2
        self.varianceX = varianceX = self.Planar2ndMomentWrtY / area - meanX**2
        self.varianceY = varianceY = self.Planar2ndMomentWrtX / area - meanY**2

        self.stddevX = stddevX = math.copysign(abs(varianceX)**.5, varianceX)
        self.stddevY = stddevY = math.copysign(abs(varianceY)**.5, varianceY)

        #  Covariance(X,Y) = ( E[X.Y] - E[X]E[Y] )
        self.covariance = covariance = self.ProductMomentXY / area - meanX*meanY

        #  Correlation(X,Y) = Covariance(X,Y) / ( stddev(X) * stddev(Y) )
        # https://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient
        correlation = covariance / (stddevX * stddevY)
        self.correlation = correlation if abs(correlation) > 1e-3 else 0

        slant = covariance / varianceY
        self.slant = slant if abs(slant) > 1e-3 else 0

        # Backward-compatibility properties
        # Do not rely on these in new code.
        self.momentX = self.Planar1stMomentWrtY
        self.momentY = self.Planar1stMomentWrtX
        self.momentXX = self.Planar2ndMomentWrtY
```

```
            self.momentYY = self.Planar2ndMomentWrtX
            self.momentXY = self.ProductMomentXY

            # Polar 2nd and 4th moments
            #
            # (Polar 1st and 3rd moments don't have polynomial solutions easily
            #  plugged into GreenPen form at present.)
            self.Polar2ndMoment = self.Planar1stMomentWrtX + self.Planar1stMomentWrtY
            self.Polar4thMoment = self.Planar4thMomentWrtX + self.Planar4thMomentWrtY + \
            2*self.ProductMomentXXYY
```

ii. LOGOS centrepoint reimplementation

This code implements a basic LOGOS centrepoint-finding technique. It requires the use of the `NewMomentsPen` and `NewStatisticsPen` classes from the previous section.

```
##############################################
#
# Originally implemented as a module named logosTools.py
#
##############################################

# Imports
from fontTools.ttLib import TTFont, TTCollection, removeOverlaps
import math
import os
import brotli
from beziers.line import Line
from beziers.point import Point
from beziers.path import BezierPath
from beziers.boundingbox import BoundingBox
import matplotlib.pyplot as plt
from fontTools.pens.boundsPen import BoundsPen
from fontTools.pens.statisticsPen import StatisticsPen
from fontTools.pens.transformPen import TransformPen
from fontTools.pens.ttGlyphPen import TTGlyphPen
from fontTools.pens.areaPen import AreaPen
from fontTools.pens.recordingPen import RecordingPen, DecomposingRecordingPen
from fontTools.misc.transform import Offset, Scale
import pathops

# Utility functions
def clip_glyph(glyphName, font, xMin, yMin, xMax, yMax):
```

```
    """Clip the given glyph at the X and Y bounds provided and
       return the resulting tGlyph object.
       """
    gpath = removeOverlaps.skPathFromGlyph(glyphName, font.getGlyphSet())
    #gpath = pathFromGlyph(glyph)
    # make a Path from the box coordinates
    box = pathops.Path()
    box.moveTo(xMin, yMin)
    box.lineTo(xMax, yMin)
    box.lineTo(xMax, yMax)
    box.lineTo(xMin, yMax)
    box.close()
    # intersect the Paths
    clipped = pathops.op(gpath, box, pathops.PathOp.INTERSECTION)
    return removeOverlaps.ttfGlyphFromSkPath(clipped)

def bsearch_mid_h(glyphName, font, metric):
    """High-level binary search wrapper, left-right."""

    #bpen = BoundsPen(glyphSet)
    #glyphSet[glyphName].draw(bpen)
    #xMin, yMin, xMax, yMax = bpen.bounds # values have been offset; unneeded

    glyph = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph[0].bounds()
    for b in glyph:
        bbox.extend(b.bounds())
    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

    # debug
    #print(xMin, yMin, xMax, yMax)

    start = xMin
    end = xMax

    mid = (start + end) / 2

    # debug
    #print("start: ", start, "mid: ", mid, "end: ", end)

    while (start <= end):

        left = metric(glyphName, font, xMin, yMin, mid, yMax)
        right = metric(glyphName, font, mid, yMin, xMax, yMax)
        #debug
        #print(left, right)

        if math.isclose(left, right):
            # debug
            #print("left/right converge - start: ", start, "mid: ", mid, "end: ", end)
            return mid

        if (start == mid) or (mid == end):
            # debug
            #print("middle converge - start: ", start, "mid: ", mid, "end: ", end)
            return mid

        if left < right: # if metric(left) is smaller than metric(right), true midpoint is further
right
            start = mid
            mid = (end + mid) / 2
```

```
            # debug
            #print("start: ", start, "mid: ", mid, "end: ", end)
        else: # if metric(right) is smaller than metric(left), true midpoint is further left
            end = mid
            mid = (start + mid) / 2
            # debug
            #print("start: ", start, "mid: ", mid, "end: ", end)

    return -1


def bsearch_mid_v(glyphName, font, metric):
    """High-level binary search wrapper, top-bottom."""

    glyph = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph[0].bounds()
    for b in glyph:
        bbox.extend(b.bounds())
    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

    # debug
    #print(xMin, yMin, xMax, yMax)

    start = yMin
    end = yMax

    mid = (start + end) / 2

    # debug
    #print("start: ", start, "mid: ", mid, "end: ", end)

    while (start <= end):

        bottom = metric(glyphName, font, xMin, yMin, xMax, mid)
        top = metric(glyphName, font, xMin, mid, xMax, yMax)
        #debug
        #print(bottom, top)

        if math.isclose(bottom, top):
            # debug
            #print("top/bottom converge - start: ", start, "mid: ", mid, "end: ", end)
            return mid

        if (start == mid) or (mid == end):
            # debug
            #print("middle converge - start: ", start, "mid: ", mid, "end: ", end)
            return mid

        if bottom < top: # if metric(bottom) is smaller than metric(top), true midpoint is further
top
            start = mid
            mid = (end + mid) / 2
            # debug
            #print("start: ", start, "mid: ", mid, "end: ", end)
        else: # if metric(top) is smaller than metric(bottom), true midpoint is further bottom
            end = mid
            mid = (start + mid) / 2
            # debug
            #print("start: ", start, "mid: ", mid, "end: ", end)

    return -1
```

```
def plot_metric_x(glyphName, font, metricval, metriclabel, plt, color=None):
    """Add a result to the matplotlib plot plt"""
    #bp = BoundsPen(font.getGlyphSet())
    #glyphSet[glyphName].draw(bp)
    #xMin, yMin, xMax, yMax = bp.bounds
    glyph = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph[0].bounds()
    for b in glyph:
        bbox.extend(b.bounds())
    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

    point1 = [metricval, yMin - 10]
    point2 = [metricval, yMax + 10]
    x_values = [point1[0], point2[0]]
    y_values = [point1[1], point2[1]]

    if color:
        plt.plot(x_values, y_values,label=metriclabel, color=color)
    else:
        plt.plot(x_values, y_values,label=metriclabel)
    plt.legend()

    return 0


def plot_metric_y(glyphName, font, metricval, metriclabel, plt, color=None):
    """Add a result to the matplotlib plot plt"""
    #bp = BoundsPen(glyphSet)
    #glyphSet[glyphName].draw(bp)
    #xMin, yMin, xMax, yMax = bp.bounds
    glyph = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph[0].bounds()
    for b in glyph:
        bbox.extend(b.bounds())
    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

    point1 = [xMin - 10, metricval]
    point2 = [xMax + 10, metricval]
    x_values = [point1[0], point2[0]]
    y_values = [point1[1], point2[1]]

    if color:
        plt.plot(x_values, y_values,label=metriclabel, color=color)
    else:
        plt.plot(x_values, y_values,label=metriclabel)

    plt.legend()

    return 0



def plot_metric_pt(glyphName, font, metricval_x, metricval_y, metriclabel, plt, color=None):
    """Add an (x,y) point result to the matplotlib plot plt"""
    if color:
        plt.plot(metricval_x, metricval_y, marker="o", markersize=5, label=metriclabel,
markerfacecolor=color, markeredgecolor=color)
    else:
        plt.plot(metricval_x, metricval_y, marker="o", markersize=5, label=metriclabel)
```

```
    plt.legend()

    return 0


def area_metric(glyphName, font, xMin, yMin, xMax, yMax):
    """"Simple metric function to compute the area of the glyph when
        clipped by the given bounds.
    """
    spen = NewStatisticsPen(font.getGlyphSet())

    clipped_glyph = clip_glyph(glyphName, font, xMin, yMin, xMax, yMax)

    clipped_glyph.draw(spen, font.getGlyphSet())
    #debug
    #print(apen.value)
    return spen.area


def secondpolar_metric(glyphName, font, xMin, yMin, xMax, yMax):
    """Compute second polar moment of the glyph when clipped by
       the given bounds.
    """

    glyphset = font.getGlyphSet()

    # back up the original glyph
    backuppen = DecomposingRecordingPen(glyphset)
    glyphset[glyphName].draw(backuppen)

    # Clip the active glyph
    clipped_glyph = clip_glyph(glyphName, font, xMin, yMin, xMax, yMax)

    # Insert the clipped glyph into the font
    dcpen = DecomposingRecordingPen(glyphset)
    clipped_glyph.draw(dcpen, font["glyf"])
    path = pathops.Path()
    pathPen = path.getPen()
    dcpen.replay(pathPen)
    ttPen = TTGlyphPen(None)
    path.draw(ttPen)
    font["glyf"][glyphName] = ttPen.glyph()

    # Find the centroid of the clipped glyph
    # x coordinate
    cx = bsearch_mid_h(glyphName, font, area_metric)
    # y coordinate
    cy = bsearch_mid_v(glyphName, font, area_metric)

    # Set up a statistics pen
    spen = NewStatisticsPen(glyphset)

    # Re-center the clipped glyph to the centroid
    # and get its stats
    pen = TransformPen(spen, Offset(x=-cx, y=-cy))
    glyphset[glyphName].draw(pen)

    # Restore the original glyph so it is ready for the next iteration
    backuppath = pathops.Path()
    backuppathPen = backuppath.getPen()
```

```
    backuppen.replay(backuppathPen)
    backupttPen = TTGlyphPen(None)
    backuppath.draw(backupttPen)
    font["glyf"][glyphName] = backupttPen.glyph()


    return spen.Polar2ndMoment


def fourthpolar_metric(glyphName, font, xMin, yMin, xMax, yMax):
    """Compute fourth polar moment of the glyph when clipped by
       the given bounds.
    """

    glyphset = font.getGlyphSet()

    # back up the original glyph
    backuppen = DecomposingRecordingPen(glyphset)
    glyphset[glyphName].draw(backuppen)

    # Clip the active glyph
    clipped_glyph = clip_glyph(glyphName, font, xMin, yMin, xMax, yMax)

    # Insert the clipped glyph into the font
    dcpen = DecomposingRecordingPen(glyphset)
    clipped_glyph.draw(dcpen, font["glyf"])
    path = pathops.Path()
    pathPen = path.getPen()
    dcpen.replay(pathPen)
    ttPen = TTGlyphPen(None)
    path.draw(ttPen)
    font["glyf"][glyphName] = ttPen.glyph()

    # Find the centroid of the clipped glyph
    # x coordinate
    cx = bsearch_mid_h(glyphName, font, area_metric)
    # y coordinate
    cy = bsearch_mid_v(glyphName, font, area_metric)

    # Set up a statistics pen
    spen = NewStatisticsPen(glyphset)

    # Re-center the clipped glyph to the centroid
    # and get its stats
    pen = TransformPen(spen, Offset(x=-cx, y=-cy))
    glyphset[glyphName].draw(pen)

    # Restore the original glyph so it is ready for the next iteration
    backuppath = pathops.Path()
    backuppathPen = backuppath.getPen()
    backuppen.replay(backuppathPen)
    backupttPen = TTGlyphPen(None)
    backuppath.draw(backupttPen)
    font["glyf"][glyphName] = backupttPen.glyph()


    return spen.Polar4thMoment


def plot_logos_stats(glyphName, font, plt):
    glyphset = font.getGlyphSet()
```

```
    xheight = f['OS/2'].sxHeight

    glyph_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph_image[0].bounds()
    for b in glyph_image:
        b.plot(ax)
        bbox.extend(b.bounds())

    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top


    # back up the original glyph
    backuppen = DecomposingRecordingPen(glyphset)
    glyphset[glyphName].draw(backuppen)

    # Clip the active glyph
    clipped_glyph = clip_glyph(glyphName, font, xMin, 0, xMax, xheight)

    # Insert the clipped glyph into the font
    dcpen = DecomposingRecordingPen(glyphset)
    clipped_glyph.draw(dcpen, font["glyf"])
    path = pathops.Path()
    pathPen = path.getPen()
    dcpen.replay(pathPen)
    ttPen = TTGlyphPen(None)
    path.draw(ttPen)
    font["glyf"][glyphName] = ttPen.glyph()

    glyph_cropped_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    for b in glyph_cropped_image:
        b.plot(ax)

    cx = bsearch_mid_h(glyphName, font, area_metric)
    cy = bsearch_mid_v(glyphName, font, area_metric)
    print("Centroid: (", cx, ",", cy, ")")

    plt.plot(cx, cy, marker="$\u22c8$", markersize=10, label="Centroid", markerfacecolor="blue",
markeredgecolor="blue")

    m2x = bsearch_mid_h(glyphName, font, secondpolar_metric)
    m2y = bsearch_mid_v(glyphName, font, secondpolar_metric)

    print("Polar2M: (", m2x, ",", m2y, ")")
    plt.plot(m2x, m2y, marker="$\u25ce$", markersize=10, label="Polar2M", markerfacecolor="orange",
markeredgecolor="orange")

    m4x = bsearch_mid_h(glyphName, font, fourthpolar_metric)
    m4y = bsearch_mid_v(glyphName, font, fourthpolar_metric)

    print("Polar4M: (", m4x, ",", m4y, ")")
    plt.plot(m4x, m4y, marker="$\u27d0$", markersize=10, label="Polar4M", markerfacecolor="red",
markeredgecolor="red")

     # Restore the original glyph so it is ready for the next iteration
    backuppath = pathops.Path()
    backuppathPen = backuppath.getPen()
    backuppen.replay(backuppathPen)
    backupttPen = TTGlyphPen(None)
    backuppath.draw(backupttPen)
    font["glyf"][glyphName] = backupttPen.glyph()
```

```
    return cx, cy, m2x, m2y, m4x, m4y


def x_logos_stats(glyphName, font):
    glyphset = font.getGlyphSet()
    xheight = font['OS/2'].sxHeight

    glyph_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph_image[0].bounds()
    for b in glyph_image:
        #b.plot(ax)
        bbox.extend(b.bounds())

    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top


    # back up the original glyph
    backuppen = DecomposingRecordingPen(glyphset)
    glyphset[glyphName].draw(backuppen)

    # Clip the active glyph
    clipped_glyph = clip_glyph(glyphName, font, xMin, 0, xMax, xheight)

    # Insert the clipped glyph into the font
    dcpen = DecomposingRecordingPen(glyphset)
    clipped_glyph.draw(dcpen, font["glyf"])
    path = pathops.Path()
    pathPen = path.getPen()
    dcpen.replay(pathPen)
    ttPen = TTGlyphPen(None)
    path.draw(ttPen)
    font["glyf"][glyphName] = ttPen.glyph()

    #glyph_cropped_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    #for b in glyph_cropped_image:
    #    b.plot(ax)

    cx = bsearch_mid_h(glyphName, font, area_metric)
    cy = bsearch_mid_v(glyphName, font, area_metric)
    #print("Centroid: (", cx, ",", cy, ")")

    #plt.plot(cx, cy, marker="$\u22c8$", markersize=10, label="Centroid", markerfacecolor="blue",
markeredgecolor="blue")

    m2x = bsearch_mid_h(glyphName, font, secondpolar_metric)
    m2y = bsearch_mid_v(glyphName, font, secondpolar_metric)

    #print("Polar2M: (", m2x, ",", m2y, ")")
    #plt.plot(m2x, m2y, marker="$\u25ce$", markersize=10, label="Polar2M",
markerfacecolor="orange", markeredgecolor="orange")

    m4x = bsearch_mid_h(glyphName, font, fourthpolar_metric)
    m4y = bsearch_mid_v(glyphName, font, fourthpolar_metric)

    #print("Polar4M: (", m4x, ",", m4y, ")")
    #plt.plot(m4x, m4y, marker="$\u27d0$", markersize=10, label="Polar4M", markerfacecolor="red",
markeredgecolor="red")

     # Restore the original glyph so it is ready for the next iteration
    backuppath = pathops.Path()
```

```
    backuppathPen = backuppath.getPen()
    backuppen.replay(backuppathPen)
    backupttPen = TTGlyphPen(None)
    backuppath.draw(backupttPen)
    font["glyf"][glyphName] = backupttPen.glyph()

    return cx, cy, m2x, m2y, m4x, m4y


def cap_logos_stats(glyphName, font):
    glyphset = font.getGlyphSet()
    capheight = font['OS/2'].sCapHeight

    glyph_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    bbox = glyph_image[0].bounds()
    for b in glyph_image:
        #b.plot(ax)
        bbox.extend(b.bounds())

    xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top


    # back up the original glyph
    backuppen = DecomposingRecordingPen(glyphset)
    glyphset[glyphName].draw(backuppen)

    # Clip the active glyph
    clipped_glyph = clip_glyph(glyphName, font, xMin, 0, xMax, capheight)

    # Insert the clipped glyph into the font
    dcpen = DecomposingRecordingPen(glyphset)
    clipped_glyph.draw(dcpen, font["glyf"])
    path = pathops.Path()
    pathPen = path.getPen()
    dcpen.replay(pathPen)
    ttPen = TTGlyphPen(None)
    path.draw(ttPen)
    font["glyf"][glyphName] = ttPen.glyph()

    #glyph_cropped_image = BezierPath.fromFonttoolsGlyph(font, glyphName)
    #for b in glyph_cropped_image:
    #    b.plot(ax)

    cx = bsearch_mid_h(glyphName, font, area_metric)
    cy = bsearch_mid_v(glyphName, font, area_metric)
    #print("Centroid: (", cx, ",", cy, ")")

    #plt.plot(cx, cy, marker="$\u22c8$", markersize=10, label="Centroid", markerfacecolor="blue",
markeredgecolor="blue")

    m2x = bsearch_mid_h(glyphName, font, secondpolar_metric)
    m2y = bsearch_mid_v(glyphName, font, secondpolar_metric)

    #print("Polar2M: (", m2x, ",", m2y, ")")
    #plt.plot(m2x, m2y, marker="$\u25ce$", markersize=10, label="Polar2M",
markerfacecolor="orange", markeredgecolor="orange")

    m4x = bsearch_mid_h(glyphName, font, fourthpolar_metric)
    m4y = bsearch_mid_v(glyphName, font, fourthpolar_metric)

    #print("Polar4M: (", m4x, ",", m4y, ")")
```

```
    #plt.plot(m4x, m4y, marker="$\u27d0$", markersize=10, label="Polar4M", markerfacecolor="red",
markeredgecolor="red")

     # Restore the original glyph so it is ready for the next iteration
    backuppath = pathops.Path()
    backuppathPen = backuppath.getPen()
    backuppen.replay(backuppathPen)
    backupttPen = TTGlyphPen(None)
    backuppath.draw(backupttPen)
    font["glyf"][glyphName] = backupttPen.glyph()

    return cx, cy, m2x, m2y, m4x, m4y


##############################################
#
# Batch-computation of M2 and M4 points at x-height
#
##############################################
import logosTools as lt
import os, argparse, csv
from beziers.path import BezierPath
from beziers.boundingbox import BoundingBox
import matplotlib.pyplot as plt
from fontTools.ttLib import TTFont, TTCollection
from fontTools.misc.cliTools import makeOutputFileName

import glob, sys

# Example glyphlist; basic Latin
glyphlist = ["a", "b", "c", "d", "e", "f", "g", "h", "i", "j", "k", "l", "m", "n", "o", "p", "q",
"r", "s", "t", "u", "v", "w", "x", "y", "z", "A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K",
"L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", '0', '1', '2', '3', '4',
'5', '6', '7', '8', '9', '@', '(', '[', '!', '{', '?', '#', '$', '%', '&', ')', '}', ']', '\'',
"\"", '.', ',', ':', ';', '*', '^', '+', '-', '=', '_']

#glyphlist = ["”", '.', ',', ':', ';', '*', '^', '+', '-', '=', '_']
#glyphlist = ["’"]
filepath = sys.argv[1]
d, infile = os.path.split(filepath)
fontfile, ext = os.path.splitext(infile)

# We'll use fontname as the name rather than extracting it from the binary
outfile = makeOutputFileName(input=fontfile, extension=".csv")

f = TTFont(filepath)

cm = f.getBestCmap()


with open(outfile, 'w', encoding='utf8') as csvfile:
    writer=csv.writer(csvfile)
    writer.writerow(['glyph', 'cx', 'cy', 'm2x', 'm2y', 'm4x', 'm4y'])

    for g in glyphlist:
        stats = []
        stats.append(g)

        cx, cy, m2x, m2y, m4x, m4y = lt.x_logos_stats(cm[ord(g)], f)

        stats.append(cx)  # This method of saving results clearly is not efficient, but works
```

```
        stats.append(cy)  #
        stats.append(m2x) #
        stats.append(m2y) #
        stats.append(m4x)
        stats.append(m4y)

        print(stats)
        fig, ax = plt.subplots()
        plt.axis('scaled')

        glyph_image = BezierPath.fromFonttoolsGlyph(f, cm[ord(g)])
        bbox = glyph_image[0].bounds()

        # draw the glyph image
        for b in glyph_image:
            b.plot(ax)
            bbox.extend(b.bounds())

        xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

        # draw the cropping lines
        for y in (0, f['OS/2'].sxHeight):
            point1 = [xMin - 10, y]
            point2 = [xMax + 10, y]
            x_values = [point1[0], point2[0]]
            y_values = [point1[1], point2[1]]
            plt.plot(x_values, y_values, color="gray")

        # draw the metrics
        plt.plot(cx, cy, marker="$\u22c8$", markersize=10, label="Centroid",
markerfacecolor="blue", markeredgecolor="blue")
        plt.plot(m2x, m2y, marker="$\u25ce$", markersize=10, label="Polar2M",
markerfacecolor="orange", markeredgecolor="orange")
        plt.plot(m4x, m4y, marker="$\u27d0$", markersize=10, label="Polar4M",
markerfacecolor="red", markeredgecolor="red")
        plt.legend()
        ax.set_title(fontfile + " \'" + g + "\'")

        svgfilename = fontfile + "_" + cm[ord(g)] + "_x_logos.svg"
        plt.savefig(svgfilename, dpi=300, bbox_inches="tight")


        # write the list as a row to the CSV
        writer.writerow(stats)



##############################################################
#
# Batch-computation of M2 and M4 points at cap-height
#
##############################################################
import logosTools as lt
import os, argparse, csv
from beziers.path import BezierPath
from beziers.boundingbox import BoundingBox
import matplotlib.pyplot as plt
from fontTools.ttLib import TTFont, TTCollection
from fontTools.misc.cliTools import makeOutputFileName

import glob, sys
```

```
# Example glyphlist; would need to be checked against actual heights
glyphlist = ["b", "d", "f", "h", "i", "j", "k", "l", "t", "A", "B", "C", "D", "E", "F", "G", "H",
"I", "J", "K", "L", "M", "N", "O", "P", "Q", "R", "S", "T", "U", "V", "W", "X", "Y", "Z", '0', '1',
'2', '3', '4', '5', '6', '7', '8', '9', '@', '(', '[', '!', '{', '?', '#', '$', '%', '&', ')', '}',
']', '\'', "\"", '*', '^']

#glyphlist = [""", '.', ',', ':', ';', '*', '^', '+', '-', '=', '_']
#glyphlist = ["’"]
filepath = sys.argv[1]
d, infile = os.path.split(filepath)
fontfile, ext = os.path.splitext(infile)

# We'll use fontname as the name rather than extracting it from the binary
outfile = makeOutputFileName(input=fontfile, extension="_caps.csv")

f = TTFont(filepath)

cm = f.getBestCmap()


with open(outfile, 'w', encoding='utf8') as csvfile:
    writer=csv.writer(csvfile)
    writer.writerow(['glyph', 'cx', 'cy', 'm2x', 'm2y', 'm4x', 'm4y'])

    for g in glyphlist:
        stats = []
        stats.append(g)

        cx, cy, m2x, m2y, m4x, m4y = lt.cap_logos_stats(cm[ord(g)], f)

        stats.append(cx)  # Still not efficient
        stats.append(cy)  #
        stats.append(m2x) #
        stats.append(m2y) #
        stats.append(m4x)
        stats.append(m4y)

        print(stats)
        fig, ax = plt.subplots()
        plt.axis('scaled')

        glyph_image = BezierPath.fromFonttoolsGlyph(f, cm[ord(g)])
        bbox = glyph_image[0].bounds()

        # draw the glyph image
        for b in glyph_image:
            b.plot(ax)
            bbox.extend(b.bounds())

        xMin, yMin, xMax, yMax = bbox.left, bbox.bottom, bbox.right, bbox.top

        # draw the cropping lines
        for y in (0, f['OS/2'].sxHeight):
            point1 = [xMin - 10, y]
            point2 = [xMax + 10, y]
            x_values = [point1[0], point2[0]]
            y_values = [point1[1], point2[1]]
            plt.plot(x_values, y_values, color="gray")

        # draw the metrics
```

```
        plt.plot(cx, cy, marker="$\u22c8$", markersize=10, label="Centroid",
markerfacecolor="blue", markeredgecolor="blue")
        plt.plot(m2x, m2y, marker="$\u25ce$", markersize=10, label="Polar2M",
markerfacecolor="orange", markeredgecolor="orange")
        plt.plot(m4x, m4y, marker="$\u27d0$", markersize=10, label="Polar4M",
markerfacecolor="red", markeredgecolor="red")
        plt.legend()
        ax.set_title(fontfile + " \'" + g + "\'")

        svgfilename = fontfile + "_" + cm[ord(g)] + "_cap_logos.svg"
        plt.savefig(svgfilename, dpi=300, bbox_inches="tight")


        # write the list as a row to the CSV
        writer.writerow(stats)
```

Appendix C: fonts tested

i. Pilot test

Cantarell Regular, designed by Dave Crossland, the Cantarell Project: 2009. Version 0.30. https://cantarell.gnome.org/

Fira Sans Extra Condensed, designed by Carrois Apostrophe: 2014. Version 4.203. https://github.com/mozilla/Fira

Alfa Slab One Regular, designed by José M. Solė: 2016. Version 2.00. https://github.com/google/fonts/tree/main/ofl/alfaslabone

Libre Caslon Text Italic, designed by Pablo Impallari, Impallari Type: 2018. Version 1.10. https://github.com/thundernixon/Libre-Caslon

Rajdhani Light, Latin designed by Shiva Nalleperumal, Indian Type Foundry: 2014. Version 1.20. https://github.com/itfoundry/rajdhani

Tenor Sans Regular, designed by Denis Masharov: 2010. Version 1.00. https://github.com/google/fonts/tree/main/ofl/tenorsans

ii. Public test batteries

Abril Fatface Regular, designed by TypeTogether: 2011. Version 1.001. https://www.type-together.com/abril-fatface-free

Alegreya Regular, designed by Juan Pablo del Peral, Huerta Tipográfica: 2011. Version 2.003. https://www.huertatipografica.com/en/fonts/alegreya-ht-pro

Alegreya Italic, designed by Juan Pablo del Peral, Huerta Tipográfica: 2011. Version 2.003. https://www.huertatipografica.com/en/fonts/alegreya-ht-pro

Alegreya Sans Italic, designed by Juan Pablo del Peral, Huerta Tipográfica: 2013. Version 2.004. https://www.huertatipografica.com/en/fonts/alegreya-sans-ht

Amiri Regular, Latin designed by Sebastian Kosch, Amiri Font Project: 2010. Version 0.113. https://www.amirifont.org/

Amiri Italic, Latin designed by Sebastian Kosch, Amiri Font Project: 2010. Version 0.113. https://www.amirifont.org/

Andika Regular, designed by SIL International: 2004. Version 5.000. https://software.sil.org/andika/

Bellefair Regular, designed by Nick Shinn and Liron Lavi Turkenic: 2015. Version 1.003. https://github.com/shinntype/bellefair

Gentium Plus Regular, designed by Victor Gaultney, SIL International: 2003. Version 5.000. https://software.sil.org/gentium/

Gentium Plus Italic, designed by Victor Gaultney, SIL International: 2003. Version 5.000. https://software.sil.org/gentium/

IM Fell Double Pica Regular, designed by Igino Marini: 2010. Version 3.00. https://iginomarini.com/fell/the-revival-fonts/

IM Fell Double Pica Italic, designed by Igino Marini: 2010. Version 3.00. https://iginomarini.com/fell/the-revival-fonts/

Literata Regular optical size 10, designed by Veronika Burian and José Scaglione: TypeTogether, 2017. Version 3.002. https://www.type-together.com/literata-font

Literata Regular optical size 18, designed by Veronika Burian and José Scaglione: TypeTogether, 2017. Version 3.002. https://www.type-together.com/literata-font

Neuton Regular, designed by Brian Zick: 2010. Version 1.560. https://github.com/anoxic/neuton

PT Serif Regular, designed by ParaType, 2010. Version 1.000W OFL. https://fonts.google.com/specimen/PT+Serif

PT Serif Italic, designed by ParaType, 2010. Version 1.000W OFL. https://fonts.google.com/specimen/PT+Serif

Sorts Mill Goudy Regular, designed by Barry Schwartz: 2010. Version 003.101. https://www.theleagueofmoveabletype.com/sorts-mill-goudy

Sorts Mill Goudy Italic, designed by Barry Schwartz: 2010. Version 003.101. https://www.theleagueofmoveabletype.com/sorts-mill-goudy

Source Sans Pro Light, designed by Paul D. Hunt: Adobe, 2010. Version 2.021. https://fonts.adobe.com/fonts/source-sans

Source Sans Pro SemiBold, designed by Paul D. Hunt: Adobe, 2010. Version 2.021. https://fonts.adobe.com/fonts/source-sans

Tinos Regular, designed by Steve Matteson: Ascender, 2010. Version 1.23. https://github.com/googlefonts/tinos

Tinos Italic, designed by Steve Matteson, Ascender, 2010. Version 1.23. https://github.com/googlefonts/tinos

Yrsa Medium, designed by Anna Giedrys, David Brezina, the Yrsa-Rasa Project: Rosetta Type, 2015. Version 1.001. https://rosettatype.com/custom-services/Yrsa-and-Rasa-for-Google

Alegreya Sans Regular, designed by Juan Pablo del Peral: Huerta Tipográfica, 2013. Verison 2.004. https://www.huertatipografica.com/en/fonts/alegreya-sans-ht

Source Sans Pro Regular, designed by Paul D. Hunt: Adobe, 2010. Version 2.021. https://fonts.adobe.com/fonts/source-sans

Slabo 13px Regular, designed by John Hudson: Tiro Typeworks, 2013. Version 1.02 Build 005a. https://github.com/TiroTypeworks/Slabo

Literata Regular optical size 14, designed by Veronika Burian and José Scaglione: TypeTogether, 2017. Version 3.002. https://www.type-together.com/literata-font

Yrsa Regular, designed by Anna Giedrys, David Brezina, the Yrsa-Rasa Project: Rosetta Type, 2015. Version 1.001. https://rosettatype.com/custom-services/Yrsa-and-Rasa-for-Google

Source Serif 4 Regular, designed by Frank Grießhammer: Adobe, 2014. Version 4.004. https://fonts.adobe.com/fonts/source-serif-4

STIX Two Text Regular, designed by Ross Mills, John Hudson, and Paul Hanslow: Tiro Typeworks, 2021. Version 2.13 b171. https://www.stixfonts.org/

Slabo 27px Regular, designed by John Hudson: Tiro Typeworks, 2013. Version 1.02 Build 0003a. https://github.com/TiroTypeworks/Slabo

Fira Sans Condensed, designed by Carrois Apostrophe: 2015. Version 4.203. https://github.com/mozilla/Fira

Yrsa Bold, designed by Anna Giedrys, David Brezina, the Yrsa-Rasa Project: Rosetta Type, 2015. Version 1.001. https://rosettatype.com/custom-services/Yrsa-and-Rasa-for-Google

Appendix D: refitting data

i. Samples of fonts tested

The following pages show samples of each font tested in each test condition. Sample text blocks are pulled from the public-testing set.

Alegreya Sans Regular, control (original fitting)

Vaudeville is a theatrical genre of variety entertainment born in France at the end of the 18th century. A vaudeville was originally a comedy without psychological or moral intentions, based on a comical situation: a kind of dramatic composition or light poetry, interspersed with songs or ballets. It became popular in the United States and Canada from the early 1880s until the early 1930s, but the idea of vaudeville's theatre changed radically from its French antecedent.

Square Awesome YMCA. Fharo Rquest Gnarly. Chico Xmen Eva. Illegal Ghata Skank. NASA. JORTS.

Vaudeville developed from many sources, also including the concert saloon, minstrelsy, freak shows, dime museums, and literary American burlesque. Called "the heart of American show business", vaudeville was one of the most popular types of entertainment in North America for several decades.

Flax (Linum usitatissimum), also known as common flax or linseed, is a member of the genus Linum in the family Linaceae. It is a food and fiber crop cultivated in cooler regions of the world. Textiles made from flax are known in the Western countries as linen, and traditionally used for bed sheets, underclothes, and table linen. Its oil is known as linseed oil. In addition to referring to the plant itself, the word "flax" may refer to the unspun fibers of the flax plant.

The plant species is known only as a cultivated plant, and appears to have been domesticated just once from the wild species Linum bienne, called pale flax.

Jwala Ibiza, Vmware. KFC. TRENCHCOAT. Yuletide, Lorping, Particulate. Ingrate. SOFFIT. AYES. HAVE. IT. Nine. Fletching.

A craft or trade is a pastime or a profession that requires particular skills and knowledge of skilled work. In a historical sense, particularly the Middle Ages and earlier, the term is usually applied to people occupied in small-scale production of goods, or their maintenance, for example by tinkers. The traditional term craftsman is nowadays often replaced by artisan and rarely by craftsperson (craftspeople).

Historically, the more specialized crafts with high value products tended to concentrate in urban centers and formed guilds. The skill required by their professions and the need to be permanently involved in the exchange of goods often demanded a generally higher level of education, and craftsmen were usually in a more privileged position than the peasantry in societal hierarchy.

Cotton is a soft, fluffy staple fiber that grows in a boll, or protective case, around the seeds of the cotton plants of the genus Gossypium in the mallow family Malvaceae. The fiber is almost pure cellulose.

Ekistic Termite. HOODWINKED. Equestre Ian Biology. Tgre, Juliana, Xkon, Ihaan, Hannah. Konstantin Brave. VANAHEIM.

The plant is a shrub native to tropical and subtropical regions around the world, including the Americas, Africa, Egypt and India. The greatest diversity of wild cotton species is found in Mexico, followed by Australia and Africa. Cotton was independently domesticated in the Old and New Worlds. The fiber is most often spun into yarn or thread and used to make a soft, breathable textile.

Duarte Kajole. Emma Qwertee Epilogo. RECOMBINER. After Jkrowling. Siri Bulgaria Drive. BOVINATE. Iota Voice.

Tonnage is a measure of the cargo-carrying capacity of a ship. The term derives from the taxation paid on tuns or casks of wine. In modern maritime usage, "tonnage" specifically refers to a calculation of the volume or cargo volume of a ship. Tonnage should not be confused with displacement, which refers to the actual weight of the vessel. Tonnage is commonly used to assess fees on commercial shipping.

The modern casting process is subdivided into two main categories: expendable and non-expendable casting. It is further broken down by the mold material, such as sand or metal, and pouring method, such as gravity, vacuum, or low pressure.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3" (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3" (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Alegreya Sans Regular, composite algorithm

Vaudeville is a theatrical genre of variety entertainment born in France at the end of the 18th century. A vaudeville was originally a comedy without psychological or moral intentions, based on a comical situation: a kind of dramatic composition or light poetry, interspersed with songs or ballets. It became popular in the United States and Canada from the early 1880s until the early 1930s, but the idea of vaudeville's theatre changed radically from its French antecedent.

Square Awesome YMCA. Fharo Rquest Gnarly. Chico Xmen Eva. Illegal Ghata Skank. NASA. JORTS.

Vaudeville developed from many sources, also including the concert saloon, minstrelsy, freak shows, dime museums, and literary American burlesque. Called "the heart of American show business", vaudeville was one of the most popular types of entertainment in North America for several decades.

Flax (Linum usitatissimum), also known as common flax or linseed, is a member of the genus Linum in the family Linaceae. It is a food and fiber crop cultivated in cooler regions of the world. Textiles made from flax are known in the Western countries as linen, and traditionally used for bed sheets, underclothes, and table linen. Its oil is known as linseed oil. In addition to referring to the plant itself, the word "flax" may refer to the unspun fibers of the flax plant.

The plant species is known only as a cultivated plant, and appears to have been domesticated just once from the wild species Linum bienne, called pale flax.

Jwala Ibiza, Vmware. KFC. TRENCHCOAT. Yuletide, Lorping, Particulate. Ingrate. SOFFIT. AYES. HAVE. IT. Nine. Fletching.

A craft or trade is a pastime or a profession that requires particular skills and knowledge of skilled work. In a historical sense, particularly the Middle Ages and earlier, the term is usually applied to people occupied in small-scale production of goods, or their maintenance, for example by tinkers. The traditional term craftsman is nowadays often replaced by artisan and rarely by craftsperson (craftspeople).

Historically, the more specialized crafts with high value products tended to concentrate in urban centers and formed guilds. The skill required by their professions and the need to be permanently involved in the exchange of goods often demanded a generally higher level of education, and craftsmen were usually in a more privileged position than the peasantry in societal hierarchy.

Cotton is a soft, fluffy staple fiber that grows in a boll, or protective case, around the seeds of the cotton plants of the genus Gossypium in the mallow family Malvaceae. The fiber is almost pure cellulose.

Ekistic Termite. HOODWINKED. Equestre Ian Biology. Tgre, Juliana, Xkon, Ihaan, Hannah. Konstantin Brave. VANAHEIM.

The plant is a shrub native to tropical and subtropical regions around the world, including the Americas, Africa, Egypt and India. The greatest diversity of wild cotton species is found in Mexico, followed by Australia and Africa. Cotton was independently domesticated in the Old and New Worlds. The fiber is most often spun into yarn or thread and used to make a soft, breathable textile.

Duarte Kajole. Emma Qwertee Epilogo. RECOMBINER. After Jkrowling. Siri Bulgaria Drive. BOVINATE. Iota Voice.

Tonnage is a measure of the cargo-carrying capacity of a ship. The term derives from the taxation paid on tuns or casks of wine. In modern maritime usage, "tonnage" specifically refers to a calculation of the volume or cargo volume of a ship. Tonnage should not be confused with displacement, which refers to the actual weight of the vessel. Tonnage is commonly used to assess fees on commercial shipping.

The modern casting process is subdivided into two main categories: expendable and non-expendable casting. It is further broken down by the mold material, such as sand or metal, and pouring method, such as gravity, vacuum, or low pressure.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3″ (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3″ (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Alegreya Sans Regular, rival kf algorithm

Vaudeville is a theatrical genre of variety entertainment born in France at the end of the 18th century. A vaudeville was originally a comedy without psychological or moral intentions, based on a comical situation: a kind of dramatic composition or light poetry, interspersed with songs or ballets. It became popular in the United States and Canada from the early 1880s until the early 1930s, but the idea of vaudeville's theatre changed radically from its French antecedent.

Square Awesome YMCA. Fharo Rquest Gnarly. Chico Xmen Eva. Illegal Ghata Skank. NASA. JORTS.

Vaudeville developed from many sources, also including the concert saloon, minstrelsy, freak shows, dime museums, and literary American burlesque. Called "the heart of American show business", vaudeville was one of the most popular types of entertainment in North America for several decades.

Flax (Linum usitatissimum), also known as common flax or linseed, is a member of the genus Linum in the family Linaceae. It is a food and fiber crop cultivated in cooler regions of the world. Textiles made from flax are known in the Western countries as linen, and traditionally used for bed sheets, underclothes, and table linen. Its oil is known as linseed oil. In addition to referring to the plant itself, the word "flax" may refer to the unspun fibers of the flax plant.

The plant species is known only as a cultivated plant, and appears to have been domesticated just once from the wild species Linum bienne, called pale flax.

Jwala Ibiza, Vmware. KFC. TRENCHCOAT. Yuletide, Lorping, Particulate. Ingrate. SOFFIT. AYES. HAVE. IT. Nine. Fletching.

A craft or trade is a pastime or a profession that requires particular skills and knowledge of skilled work. In a historical sense, particularly the Middle Ages and earlier, the term is usually applied to people occupied in small-scale production of goods, or their maintenance, for example by tinkers. The traditional term craftsman is nowadays often replaced by artisan and rarely by craftsperson (craftspeople).

Historically, the more specialized crafts with high value products tended to concentrate in urban centers and formed guilds. The skill required by their professions and the need to be permanently involved in the exchange of goods often demanded a generally higher level of education, and craftsmen were usually in a more privileged position than the peasantry in societal hierarchy.

Cotton is a soft, fluffy staple fiber that grows in a boll, or protective case, around the seeds of the cotton plants of the genus Gossypium in the mallow family Malvaceae. The fiber is almost pure cellulose.

Ekistic Termite. HOODWINKED. Equestre Ian Biology. Tgre, Juliana, Xkon, Ihaan, Hannah. Konstantin Brave. VANAHEIM.

The plant is a shrub native to tropical and subtropical regions around the world, including the Americas, Africa, Egypt and India. The greatest diversity of wild cotton species is found in Mexico, followed by Australia and Africa. Cotton was independently domesticated in the Old and New Worlds. The fiber is most often spun into yarn or thread and used to make a soft, breathable textile.

Duarte Kajole. Emma Qwertee Epilogo. RECOMBINER. After Jkrowling. Siri Bulgaria Drive. BOVINATE. Iota Voice.

Tonnage is a measure of the cargo-carrying capacity of a ship. The term derives from the taxation paid on tuns or casks of wine. In modern maritime usage, "tonnage" specifically refers to a calculation of the volume or cargo volume of a ship. Tonnage should not be confused with displacement, which refers to the actual weight of the vessel. Tonnage is commonly used to assess fees on commercial shipping.

The modern casting process is subdivided into two main categories: expendable and non-expendable casting. It is further broken down by the mold material, such as sand or metal, and pouring method, such as gravity, vacuum, or low pressure.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3″ (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3″ (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Fira Sans Condensed, control (original fitting)

Sachverhalt ist ein interdisziplinärer Begriff, unter dem die inhaltliche Gesamtheit aller Aussagen zu einem abgegrenzten Themenbereich verstanden wird. Entgegen der Bedeutung in der Umgangssprache sind Sachverhalte nicht mit Tatsachen gleichzusetzen. Ein Sachverhalt ist in der Wissenschaftstheorie lediglich der Gegenstand einer Aussage. Ein Sachverhalt ist solange fiktiv, bis er einer Überprüfung unterzogen und verifiziert werden konnte. Damit erst wandelt er sich zur wahren Aussage, zur Tatsache. Wurde er hingegen für falsch eingestuft (falsifiziert), handelt es sich um eine „negative Tatsache“, die keinen Sachverhalt darstellt.

Fehlzeiten, Quone, Far Side, Fast. STANDORT. Passiv, Xerxes, Macht, KRANKHEIT. Seil, Dichtung, Gelblich.

Die Aussagenlogik hat eher mit Sachverhalten zu tun als mit Tatsachen. Sie ist eine Theorie über die Existenz von Sachverhalten. Werden Aussagen miteinander verknüpft, ergibt sich die Aussage über einen Sachverhalt. Eine Aussage ist ein atomares sprachliches Gebilde, das einen Sachverhalt intendiert und dadurch die Eigenschaft erhält, wahr oder falsch zu sein. Demnach ist eine Aussage nur dann wahr, wenn der intendierte Sachverhalt tatsächlich besteht. Eine positive Tatsache ist ein bestehender Sachverhalt; eine negative Tatsache ist ein nichtbestehender Sachverhalt.

Sachverhalt im Recht ist der zweckhafte Sinnzusammenhang von rechtserheblichem menschlichen Verhalten, von Rechtsverhältnissen, Rechtstatsachen oder von Daten und Fakten der kulturell-sozialen Wirklichkeit im Hinblick auf rechtlich geregeltes menschliches Verhalten. Während der Tatbestand meist generell-abstrakt ist, ist der Sachverhalt individuell-konkret.

Bei der Erarbeitung eines Sachverhalts, etwa anhand von aktenkundigen Zeugenaussagen oder Polizeiberichten dürfen keine wesentlichen Umstände weggelassen oder hinzugefügt werden. Der Fehler von Jurastudenten, in einer Sachverhaltsdarstellung Aussagen zu machen, die sich nicht aus mitgeteilten Tatsachen ergeben, sondern aus eigener Interpretation, heißt unter Prüfern „Sachverhaltsquetsche“. Der zu bearbeitende Sachverhalt versteht sich als konkreter Lebenssachverhalt, also die sprachlich umgesetzte Wiedergabe eines – wenn auch kleinen – Ausschnitts aus der gesellschaftlichen Lebenswirklichkeit.

Die Kolibris sind eine Familie kleiner, Nektar fressender Vögel aus Süd-, Mittel- und Nordamerika. Sie sind sowohl mit den Seglern als auch mit den Baumseglern so nahe verwandt, dass sie mit diesen in der Ordnung der Seglervögel zusammengefasst werden. Einige Autoren stellen sie auch in eine eigenständige Ordnung, die Trochiliformes.

Luisa, Samantha, Fluss. Vkontakte, Rjava, DJANGO. Eira Scrabble, Helio Ahimsa. PROTEIN PRIMAL. Kzar Amora.

Die Familie der Kolibris umfasst mehr als 100 Gattungen mit zusammen 371 Arten. Der Name Kolibri wurde im 18. Jahrhundert aus dem Französischen entlehnt (frz. colibri) und stammt wohl aus einer karibischen Sprache.

Kolibris führen ihren Schwirrflug mit einer sehr hohen Frequenz von 40 bis 50 Flügelschlägen pro Sekunde aus. Während des Flugs beschreiben die Bewegungen ihrer Flügelspitzen eine liegende Lemniskate. Mit ihren beweglichen Flügeln können sie auf der Stelle fliegen, um zum Beispiel Nektar zu trinken. Sie können auch seitwärts und sogar rückwärts fliegen. Damit ist der Kolibri der einzige Vogel auf der Welt, der diese Fähigkeit besitzt.

Bezogen auf ihre Körpergröße sind Kolibris die wohl schnellsten Wirbeltiere der Welt.[4] So erreichen die etwa zehn Zentimeter großen Annakolibris bei ihren Balzflügen Geschwindigkeiten von 385 Körperlängen pro Sekunde (27,3 m/s bzw. 98 km/h), bei Beschleunigungswerten von etwa dem Zehnfachen der Erdbeschleunigung.

Untertitel Gustavo Derry. Emporkömmling Oberyn Serena. Jlaw Zar, Himalaya. Unheimlicher Zorn. Bent Bkent. ONLINE ANGEWENDETE ASSUAGE.

Das Herz der Kolibris ist im Verhältnis zum Körper sehr groß und schlägt 400- bis 500-mal pro Minute, ihre Atemfrequenz liegt bei bis zu 250 Zügen pro Minute. Während des Schlafes senken viele Kolibris ihre Herzfrequenz stark ab, um Energie zu sparen.

Archipel ist ein allgemeiner Begriff für verschiedene Inselgruppen der Weltmeere, beziehungsweise deren gesamte Regionen, einschließlich der Gewässer zwischen den Inseln.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

Im Seerecht haben auf Archipele begrenzte Staaten die besondere Möglichkeit, ihre Souveränität auf die gesamten Gewässer zwischen denjenigen ihrer Inseln auszudehnen, die nicht weiter als rund 100 Seemeilen auseinanderliegen.[2] Ihre Souveränität erstreckt sich dann auf die Inseln und die Archipelgewässer. Der Archipelstaat mit den meisten Inseln, nämlich 17.508, ist Indonesien.

Im 13. bis 16. Jahrhundert waren die Kykladen vor Athen ein Herzogtum italienischer Fürsten, das Herzogtum Archipelagos.

Der ATKIS-Objektartenkatalog DLM250 definiert Inselgruppe als „eine Gruppe mehrerer nahe beieinander liegender Inseln geologisch gleicher Entstehung“. Die Inseln einer Inselgruppe können unter Umständen auch zu kleineren Inselgruppen gruppiert werden, wie das etwa bei den Hebriden der Fall ist, die sich in die Inneren und Äußeren Hebriden untergliedern.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

Oftmals spielen jedoch für die Zusammenfassung mehrerer Inseln zu einer Inselgruppe politische, historische oder kulturelle Kriterien eine entscheidende Rolle. Dasselbe gilt auch für die Unterteilung einer geographisch zusammengehörigen Inselgruppe in Untergruppen. Je nach der topografischen Anordnung der Inseln spricht man von einer Inselkette, einem Inselbogen oder auch von einer Doppelinsel.

FAKTUELL, Isometrischer Parameter, Pauke, Euter, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Fira Sans Condensed, composite algorithm

Sachverhalt ist ein interdisziplinärer Begriff, unter dem die inhaltliche Gesamtheit aller Aussagen zu einem abgegrenzten Themenbereich verstanden wird. Entgegen der Bedeutung in der Umgangssprache sind Sachverhalte nicht mit Tatsachen gleichzusetzen. Ein Sachverhalt ist in der Wissenschaftstheorie lediglich der Gegenstand einer Aussage. Ein Sachverhalt ist solange fiktiv, bis er einer Überprüfung unterzogen und verifiziert werden konnte. Damit erst wandelt er sich zur wahren Aussage, zur Tatsache. Wurde er hingegen für falsch eingestuft (falsifiziert), handelt es sich um eine „negative Tatsache", die keinen Sachverhalt darstellt.

Fehlzeiten, Quone, Far Side, Fast. STANDORT. Passiv, Xerxes, Macht, KRANKHEIT. Seil, Dichtung, Gelblich.

Die Aussagenlogik hat eher mit Sachverhalten zu tun als mit Tatsachen. Sie ist eine Theorie über die Existenz von Sachverhalten. Werden Aussagen miteinander verknüpft, ergibt sich die Aussage über einen Sachverhalt. Eine Aussage ist ein atomares sprachliches Gebilde, das einen Sachverhalt intendiert und dadurch die Eigenschaft erhält, wahr oder falsch zu sein. Demnach ist eine Aussage nur dann wahr, wenn der intendierte Sachverhalt tatsächlich besteht. Eine positive Tatsache ist ein bestehender Sachverhalt; eine negative Tatsache ist ein nichtbestehender Sachverhalt.

Sachverhalt im Recht ist der zweckhafte Sinnzusammenhang von rechtserheblichem menschlichen Verhalten, von Rechtsverhältnissen, Rechtstatsachen oder von Daten und Fakten der kulturell-sozialen Wirklichkeit im Hinblick auf rechtlich geregeltes menschliches Verhalten. Während der Tatbestand meist generell-abstrakt ist, ist der Sachverhalt individuell-konkret.

Bei der Erarbeitung eines Sachverhalts, etwa anhand von aktenkundigen Zeugenaussagen oder Polizeiberichten dürfen keine wesentlichen Umstände weggelassen oder hinzugefügt werden. Der Fehler von Jurastudenten, in einer Sachverhaltsdarstellung Aussagen zu machen, die sich nicht aus mitgeteilten Tatsachen ergeben, sondern aus eigener Interpretation, heißt unter Prüfern „Sachverhaltsquetsche". Der zu bearbeitende Sachverhalt versteht sich als konkreter Lebenssachverhalt, also die sprachlich umgesetzte Wiedergabe eines – wenn auch kleinen – Ausschnitts aus der gesellschaftlichen Lebenswirklichkeit.

Die Kolibris sind eine Familie kleiner, Nektar fressender Vögel aus Süd-, Mittel- und Nordamerika. Sie sind sowohl mit den Seglern als auch mit den Baumseglern so nahe verwandt, dass sie mit diesen in der Ordnung der Seglervögel zusammengefasst werden. Einige Autoren stellen sie auch in eine eigenständige Ordnung, die Trochiliformes.

Luisa, Samantha, Fluss. Vkontakte, Rjava, DJANGO. Eira Scrabble, Helio Ahimsa. PROTEIN PRIMAL. Kzar Amora.

Die Familie der Kolibris umfasst mehr als 100 Gattungen mit zusammen 371 Arten. Der Name Kolibri wurde im 18. Jahrhundert aus dem Französischen entlehnt (frz. colibri) und stammt wohl aus einer karibischen Sprache.

Kolibris führen ihren Schwirrflug mit einer sehr hohen Frequenz von 40 bis 50 Flügelschlägen pro Sekunde aus. Während des Flugs beschreiben die Bewegungen ihrer Flügelspitzen eine liegende Lemniskate. Mit ihren beweglichen Flügeln können sie auf der Stelle fliegen, um zum Beispiel Nektar zu trinken. Sie können auch seitwärts und sogar rückwärts fliegen. Damit ist der Kolibri der einzige Vogel auf der Welt, der diese Fähigkeit besitzt.

Bezogen auf ihre Körpergröße sind Kolibris die wohl schnellsten Wirbeltiere der Welt.[4] So erreichen die etwa zehn Zentimeter großen Annakolibris bei ihren Balzflügen Geschwindigkeiten von 385 Körperlängen pro Sekunde (27,3 m/s bzw. 98 km/h), bei Beschleunigungswerten von etwa dem Zehnfachen der Erdbeschleunigung.

Untertitel Gustavo Derry. Emporkömmling Oberyn Serena. Jlaw Zar, Himalaya. Unheimlicher Zorn. Bent Bkent. ONLINE ANGEWENDETE ASSUAGE.

Das Herz der Kolibris ist im Verhältnis zum Körper sehr groß und schlägt 400- bis 500-mal pro Minute, ihre Atemfrequenz liegt bei bis zu 250 Zügen pro Minute. Während des Schlafes senken viele Kolibris ihre Herzfrequenz stark ab, um Energie zu sparen.

Archipel ist ein allgemeiner Begriff für verschiedene Inselgruppen der Weltmeere, beziehungsweise deren gesamte Regionen, einschließlich der Gewässer zwischen den Inseln.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

Im Seerecht haben auf Archipele begrenzte Staaten die besondere Möglichkeit, ihre Souveränität auf die gesamten Gewässer zwischen denjenigen ihrer Inseln auszudehnen, die nicht weiter als rund 100 Seemeilen auseinanderliegen.[2] Ihre Souveränität erstreckt sich dann auf die Inseln und die Archipelgewässer. Der Archipelstaat mit den meisten Inseln, nämlich 17.508, ist Indonesien.

Im 13. bis 16. Jahrhundert waren die Kykladen vor Athen ein Herzogtum italienischer Fürsten, das Herzogtum Archipelagos.

Der ATKIS-Objektartenkatalog DLM250 definiert Inselgruppe als „eine Gruppe mehrerer nahe beieinander liegender Inseln geologisch gleicher Entstehung". Die Inseln einer Inselgruppe können unter Umständen auch zu kleineren Inselgruppen gruppiert werden, wie das etwa bei den Hebriden der Fall ist, die sich in die Inneren und Äußeren Hebriden untergliedern.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

Oftmals spielen jedoch für die Zusammenfassung mehrerer Inseln zu einer Inselgruppe politische, historische oder kulturelle Kriterien eine entscheidende Rolle. Dasselbe gilt auch für die Unterteilung einer geographisch zusammengehörigen Inselgruppe in Untergruppen. Je nach der topografischen Anordnung der Inseln spricht man von einer Inselkette, einem Inselbogen oder auch von einer Doppelinsel.

FAKTUELL, Isometrischer Parameter, Pauke, Euter, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Fira Sans Condensed, rival kf algorithm

Sachverhalt ist ein interdisziplinärer Begriff, unter dem die inhaltliche Gesamtheit aller Aussagen zu einem abgegrenzten Themenbereich verstanden wird. Entgegen der Bedeutung in der Umgangssprache sind Sachverhalte nicht mit Tatsachen gleichzusetzen. Ein Sachverhalt ist in der Wissenschaftstheorie lediglich der Gegenstand einer Aussage. Ein Sachverhalt ist solange fiktiv, bis er einer Überprüfung unterzogen und verifiziert werden konnte. Damit erst wandelt er sich zur wahren Aussage, zur Tatsache. Wurde er hingegen für falsch eingestuft (falsifiziert), handelt es sich um eine „negative Tatsache", die keinen Sachverhalt darstellt.

Fehlzeiten, Quone, Far Side, Fast. STANDORT. Passiv, Xerxes, Macht, KRANKHEIT. Seil, Dichtung, Gelblich.

Die Aussagenlogik hat eher mit Sachverhalten zu tun als mit Tatsachen. Sie ist eine Theorie über die Existenz von Sachverhalten. Werden Aussagen miteinander verknüpft, ergibt sich die Aussage über einen Sachverhalt. Eine Aussage ist ein atomares sprachliches Gebilde, das einen Sachverhalt intendiert und dadurch die Eigenschaft erhält, wahr oder falsch zu sein. Demnach ist eine Aussage nur dann wahr, wenn der intendierte Sachverhalt tatsächlich besteht. Eine positive Tatsache ist ein bestehender Sachverhalt; eine negative Tatsache ist ein nichtbestehender Sachverhalt.

Sachverhalt im Recht ist der zweckhafte Sinnzusammenhang von rechtserheblichem menschlichen Verhalten, von Rechtsverhältnissen, Rechtstatsachen oder von Daten und Fakten der kulturell-sozialen Wirklichkeit im Hinblick auf rechtlich geregeltes menschliches Verhalten. Während der Tatbestand meist generell-abstrakt ist, ist der Sachverhalt individuell-konkret.

Bei der Erarbeitung eines Sachverhalts, etwa anhand von aktenkundigen Zeugenaussagen oder Polizeiberichten dürfen keine wesentlichen Umstände weggelassen oder hinzugefügt werden. Der Fehler von Jurastudenten, in einer Sachverhaltsdarstellung Aussagen zu machen, die sich nicht aus mitgeteilten Tatsachen ergeben, sondern aus eigener Interpretation, heißt unter Prüfern „Sachverhaltsquetsche". Der zu bearbeitende Sachverhalt versteht sich als konkreter Lebenssachverhalt, also die sprachlich umgesetzte Wiedergabe eines – wenn auch kleinen – Ausschnitts aus der gesellschaftlichen Lebenswirklichkeit.

Die Kolibris sind eine Familie kleiner, Nektar fressender Vögel aus Süd-, Mittel- und Nordamerika. Sie sind sowohl mit den Seglern als auch mit den Baumseglern so nahe verwandt, dass sie mit diesen in der Ordnung der Seglervögel zusammengefasst werden. Einige Autoren stellen sie auch in eine eigenständige Ordnung, die Trochiliformes.

Luisa, Samantha, Fluss. Vkontakte, Rjava, DJANGO. Eira Scrabble, Helio Ahimsa. PROTEIN PRIMAL. Kzar Amora.

Die Familie der Kolibris umfasst mehr als 100 Gattungen mit zusammen 371 Arten. Der Name Kolibri wurde im 18. Jahrhundert aus dem Französischen entlehnt (frz. colibri) und stammt wohl aus einer karibischen Sprache.

Kolibris führen ihren Schwirrflug mit einer sehr hohen Frequenz von 40 bis 50 Flügelschlägen pro Sekunde aus. Während des Flugs beschreiben die Bewegungen ihrer Flügelspitzen eine liegende Lemniskate. Mit ihren beweglichen Flügeln können sie auf der Stelle fliegen, um zum Beispiel Nektar zu trinken. Sie können auch seitwärts und sogar rückwärts fliegen. Damit ist der Kolibri der einzige Vogel auf der Welt, der diese Fähigkeit besitzt.

Bezogen auf ihre Körpergröße sind Kolibris die wohl schnellsten Wirbeltiere der Welt.[4] So erreichen die etwa zehn Zentimeter großen Annakolibris bei ihren Balzflügen Geschwindigkeiten von 385 Körperlängen pro Sekunde (27,3 m/s bzw. 98 km/h), bei Beschleunigungswerten von etwa dem Zehnfachen der Erdbeschleunigung.

Untertitel Gustavo Derry. Emporkömmling Oberyn Serena. Jlaw Zar, Himalaya. Unheimlicher Zorn. Bent Bkent. ONLINE ANGEWENDETE ASSUAGE.

Das Herz der Kolibris ist im Verhältnis zum Körper sehr groß und schlägt 400- bis 500-mal pro Minute, ihre Atemfrequenz liegt bei bis zu 250 Zügen pro Minute. Während des Schlafes senken viele Kolibris ihre Herzfrequenz stark ab, um Energie zu sparen.

Archipel ist ein allgemeiner Begriff für verschiedene Inselgruppen der Weltmeere, beziehungsweise deren gesamte Regionen, einschließlich der Gewässer zwischen den Inseln.

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

Im Seerecht haben auf Archipele begrenzte Staaten die besondere Möglichkeit, ihre Souveränität auf die gesamten Gewässer zwischen denjenigen ihrer Inseln auszudehnen, die nicht weiter als rund 100 Seemeilen auseinanderliegen.[2] Ihre Souveränität erstreckt sich dann auf die Inseln und die Archipelgewässer. Der Archipelstaat mit den meisten Inseln, nämlich 17.508, ist Indonesien.

Im 13. bis 16. Jahrhundert waren die Kykladen vor Athen ein Herzogtum italienischer Fürsten, das Herzogtum Archipelagos.

Der ATKIS-Objektartenkatalog DLM250 definiert Inselgruppe als „eine Gruppe mehrerer nahe beieinander liegender Inseln geologisch gleicher Entstehung". Die Inseln einer Inselgruppe können unter Umständen auch zu kleineren Inselgruppen gruppiert werden, wie das etwa bei den Hebriden der Fall ist, die sich in die Inneren und Äußeren Hebriden untergliedern.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

Oftmals spielen jedoch für die Zusammenfassung mehrerer Inseln zu einer Inselgruppe politische, historische oder kulturelle Kriterien eine entscheidende Rolle. Dasselbe gilt auch für die Unterteilung einer geographisch zusammengehörigen Inselgruppe in Untergruppen. Je nach der topografischen Anordnung der Inseln spricht man von einer Inselkette, einem Inselbogen oder auch von einer Doppelinsel.

FAKTUELL, Isometrischer Parameter, Pauke, Euter, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Literata Regular opsz 14, control (original fitting)

An unmanned aerial vehicle (UAV) (or uncrewed aerial vehicle, commonly known as a 'drone') is an aircraft without a human pilot on board and a type of unmanned vehicle. UAVs are a component of an unmanned aircraft system (UAS); which include a UAV, a ground-based controller, and a system of communications between the two. The flight of UAVs may operate with various degrees of autonomy: either under remote control by a human operator or autonomously by onboard computers.

Klaus Iiesh Bjork. Agora Aorta Uva. STRAWBERRY! Tbilisi Cedric Rlocums Rudolfo Hmm.... Yperite. REINING.

An agricultural drone is an unmanned aerial vehicle applied to farming in order to help increase crop production and monitor crop growth. Sensors and digital imaging capabilities can give farmers a richer picture of their fields. This information may prove useful in improving crop yields and farm efficiency. Agricultural drones let farmers see their fields from the sky. This bird's-eye view can reveal many issues such as irrigation problems, soil variation, and pest and fungal infestations.

Ykee Nero, Diana Gaya, Turkey. RNA SEQUENCE. Maria, Sdds, Xscape. Sjogren Barriga, Miguel Twang. CALLING NANCY.

In general, objects emit infrared radiation across a spectrum of wavelengths, but sometimes only a limited region of the spectrum is of interest because sensors usually collect radiation only within a specific bandwidth. Thermal infrared radiation also has a maximum emission wavelength, which is inversely proportional to the absolute temperature of object, in accordance with Wien's displacement law. Therefore, the infrared band is often subdivided into smaller sections.

Wien's displacement law states that the black-body radiation curve for different temperatures will peak at different wavelengths that are inversely proportional to the temperature. The shift of that peak is a direct consequence of the Planck radiation law, which describes the spectral brightness of black-body radiation as a function of wavelength at any given temperature.

A territory is an administrative division, usually an area that is under the jurisdiction of a state. In most countries, a territory is an organized land controlled division of an area that is controlled by a country but is not formally developed into, or incorporated into, a political unit of the country that is of equal status to other political units that may often be referred to by words such as "provinces" or "states". In international politics, a territory is usually a non-sovereign geographic area which has come under the authority of another government; which has not been granted the powers of self-government normally devolved to secondary territorial divisions; or both.

Territory made its debut as a word in Middle English during the 14th century. At this point the suffix -orium, which denotes place, was replaced with -ory which also expresses place.

Sledge, Eh? McDonald's Mhari, Tlumacz. VERSE. Tsareena Wuthering. EKE. Iuri Utero, Wario. Florence Xfight. FOOTER.

Etymology is the study of the history of words. By extension, the term "the etymology (of a word)" means the origin of the particular word. For place names, there is a specific term, toponymy.

For Greek—with a long written history—etymologists make use of texts, and texts about the language, to gather knowledge about how words were used during earlier periods and when they entered the language . Etymologists also apply the methods of comparative linguistics to reconstruct information about languages that are too old for any direct information to be available.

Xjet Undo Octavio Pilar. DNS LOOKUP. Lt. Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. SMOOTHNESS.

By analyzing related languages with a technique known as the comparative method, linguists can make inferences about their shared parent language and its vocabulary. In this way, word roots have been found that can be traced all the way back to the origin of, for instance, the Indo-European language family.

Literata Regular opsz 14, composite algorithm

An unmanned aerial vehicle (UAV) (or uncrewed aerial vehicle, commonly known as a 'drone') is an aircraft without a human pilot on board and a type of unmanned vehicle. UAVs are a component of an unmanned aircraft system (UAS); which include a UAV, a ground-based controller, and a system of communications between the two. The flight of UAVs may operate with various degrees of autonomy: either under remote control by a human operator or autonomously by onboard computers.

Klaus Iiesh Bjork. Agora Aorta Uva. STRAWBERRY! Tbilisi Cedric Rlocums Rudolfo Hmm.... Yperite. REINING.

An agricultural drone is an unmanned aerial vehicle applied to farming in order to help increase crop production and monitor crop growth. Sensors and digital imaging capabilities can give farmers a richer picture of their fields. This information may prove useful in improving crop yields and farm efficiency. Agricultural drones let farmers see their fields from the sky. This bird's-eye view can reveal many issues such as irrigation problems, soil variation, and pest and fungal infestations.

Ykee Nero, Diana Gaya, Turkey. RNA SEQUENCE. Maria, Sdds, Xscape. Sjogren Barriga, Miguel Twang. CALLING NANCY.

In general, objects emit infrared radiation across a spectrum of wavelengths, but sometimes only a limited region of the spectrum is of interest because sensors usually collect radiation only within a specific bandwidth. Thermal infrared radiation also has a maximum emission wavelength, which is inversely proportional to the absolute temperature of object, in accordance with Wien's displacement law. Therefore, the infrared band is often subdivided into smaller sections.

Wien's displacement law states that the black-body radiation curve for different temperatures will peak at different wavelengths that are inversely proportional to the temperature. The shift of that peak is a direct consequence of the Planck radiation law, which describes the spectral brightness of black-body radiation as a function of wavelength at any given temperature.

A territory is an administrative division, usually an area that is under the jurisdiction of a state. In most countries, a territory is an organized land controlled division of an area that is controlled by a country but is not formally developed into, or incorporated into, a political unit of the country that is of equal status to other political units that may often be referred to by words such as "provinces" or "states". In international politics, a territory is usually a non-sovereign geographic area which has come under the authority of another government; which has not been granted the powers of self-government normally devolved to secondary territorial divisions; or both.

Territory made its debut as a word in Middle English during the 14th century. At this point the suffix -orium, which denotes place, was replaced with -ory which also expresses place.

Sledge, Eh? McDonald's Mhari, Tlumacz. VERSE. Tsareena Wuthering. EKE. Iuri Utero, Wario. Florence Xfight. FOOTER.

Etymology is the study of the history of words. By extension, the term "the etymology (of a word)" means the origin of the particular word. For place names, there is a specific term, toponymy.

For Greek—with a long written history—etymologists make use of texts, and texts about the language, to gather knowledge about how words were used during earlier periods and when they entered the language . Etymologists also apply the methods of comparative linguistics to reconstruct information about languages that are too old for any direct information to be available.

Xjet Undo Octavio Pilar. DNS LOOKUP. Lt. Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. SMOOTHNESS.

By analyzing related languages with a technique known as the comparative method, linguists can make inferences about their shared parent language and its vocabulary. In this way, word roots have been found that can be traced all the way back to the origin of, for instance, the Indo-European language family.

Literata Regular opsz 14, rival kf algorithm

An unmanned aerial vehicle (UAV) (or uncrewed aerial vehicle, commonly known as a 'drone') is an aircraft without a human pilot on board and a type of unmanned vehicle. UAVs are a component of an unmanned aircraft system (UAS); which include a UAV, a ground-based controller, and a system of communications between the two. The flight of UAVs may operate with various degrees of autonomy: either under remote control by a human operator or autonomously by onboard computers.

Klaus Iiesh Bjork. Agora Aorta Uva. STRAWBERRY! Tbilisi Cedric Rlocums Rudolfo Hmm.... Yperite. REINING.

An agricultural drone is an unmanned aerial vehicle applied to farming in order to help increase crop production and monitor crop growth. Sensors and digital imaging capabilities can give farmers a richer picture of their fields. This information may prove useful in improving crop yields and farm efficiency. Agricultural drones let farmers see their fields from the sky. This bird's-eye view can reveal many issues such as irrigation problems, soil variation, and pest and fungal infestations.

Ykee Nero, Diana Gaya, Turkey. RNA SEQUENCE. Maria, Sdds, Xscape. Sjogren Barriga, Miguel Twang. CALLING NANCY.

In general, objects emit infrared radiation across a spectrum of wavelengths, but sometimes only a limited region of the spectrum is of interest because sensors usually collect radiation only within a specific bandwidth. Thermal infrared radiation also has a maximum emission wavelength, which is inversely proportional to the absolute temperature of object, in accordance with Wien's displacement law. Therefore, the infrared band is often subdivided into smaller sections.

Wien's displacement law states that the black-body radiation curve for different temperatures will peak at different wavelengths that are inversely proportional to the temperature. The shift of that peak is a direct consequence of the Planck radiation law, which describes the spectral brightness of black-body radiation as a function of wavelength at any given temperature.

A territory is an administrative division, usually an area that is under the jurisdiction of a state. In most countries, a territory is an organized land controlled division of an area that is controlled by a country but is not formally developed into, or incorporated into, a political unit of the country that is of equal status to other political units that may often be referred to by words such as "provinces" or "states". In international politics, a territory is usually a non-sovereign geographic area which has come under the authority of another government; which has not been granted the powers of self-government normally devolved to secondary territorial divisions; or both.

Territory made its debut as a word in Middle English during the 14th century. At this point the suffix -orium, which denotes place, was replaced with -ory which also expresses place.

Sledge, Eh? McDonald's Mhari, Tlumacz. VERSE. Tsareena Wuthering. EKE. Iuri Utero, Wario. Florence Xfight. FOOTER.

Etymology is the study of the history of words. By extension, the term "the etymology (of a word)" means the origin of the particular word. For place names, there is a specific term, toponymy.

For Greek—with a long written history—etymologists make use of texts, and texts about the language, to gather knowledge about how words were used during earlier periods and when they entered the language . Etymologists also apply the methods of comparative linguistics to reconstruct information about languages that are too old for any direct information to be available.

Xjet Undo Octavio Pilar. DNS LOOKUP. Lt. Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. SMOOTHNESS.

By analyzing related languages with a technique known as the comparative method, linguists can make inferences about their shared parent language and its vocabulary. In this way, word roots have been found that can be traced all the way back to the origin of, for instance, the Indo-European language family.

Slabo 13px, control (original fitting)

Blue is one of the three primary colours of pigments in painting and traditional colour theory, as well as in the RGB colour model. It lies between violet and green on the spectrum of visible light. The eye perceives blue when observing light with a dominant wavelength between approximately 450 and 495 nanometres. Most blues contain a slight mixture of other colours; azure contains some green, while ultramarine contains some violet. The clear daytime sky and the deep sea appear blue because of an optical effect known as Rayleigh scattering. An optical effect called Tyndall scattering explains blue eyes. Distant objects appear more blue because of another optical effect called aerial perspective.

Cummin Atmo Yclept Xray. GADFLY. Keratin Yves Fokionas Ptolomeu. PATRICIAN. Udon Ybelote Oona Gracinda.

Baseball is a bat-and-ball game played between two opposing teams who take turns batting and fielding. The game proceeds when a player on the fielding team, called the pitcher, throws a ball which a player on the batting team tries to hit with a bat. The objectives of the offensive team (batting team) are to hit the ball into the field of play, and to run the bases—having its runners advance counter-clockwise around four bases to score what are called "runs".

Null Level INJUNCTION. Sooner, Rather, Later. HALLWAY AVOIDANCE. Porches, Swans, Lyrics, Sanity. DRAWN. Copper, Cartwheel. Ivan, Zwingli.

Crasis is a type of contraction in which two vowels or diphthongs merge into one new vowel or diphthong, making one word out of two. Crasis occurs in Portuguese, French and Arabic as well as in Ancient Greek, for which it was first described.

A language family is a group of languages related through descent from a common ancestral language or parental language, called the proto-language of that family. The term "family" reflects the tree model of language origination in historical linguistics, which makes use of a metaphor comparing languages to people in a biological family tree, or in a subsequent modification, to species in a phylogenetic tree of evolutionary taxonomy.

Linguists therefore describe the daughter languages within a language family as being genetically related.

The Silk Road derives its name from the lucrative trade in silk carried out along its length, beginning in the Han dynasty in China (207 BCE–220 CE). The Han dynasty expanded the Central Asian section of the trade routes around 114 BCE through the missions and explorations of the Chinese imperial envoy Zhang Qian, as well as several military conquests.

The Silk Road trade played a significant role in the development of the civilizations of China, Korea, Japan, the Indian subcontinent, Iran, Europe, the Horn of Africa and Arabia, opening long-distance political and economic relations between the civilizations.

Zoological Timpani Flexbox, TOPCOAT. ILL EXPLORATION, DEFT CAVALIER. Retread, SSL, Hushed Bobcat Petersburg, Klaxons.

For at least a portion of its life, a star shines due to thermonuclear fusion of hydrogen into helium in its core, releasing energy that traverses the star's interior and then radiates into outer space.

Binary and multi-star systems consist of two or more stars that are gravitationally bound and generally move around each other in stable orbits. When two such stars have a relatively close orbit, their gravitational interaction can have a significant impact on their evolution. Stars can form part of a much larger gravitationally bound structure, such as a star cluster or a galaxy.

Andes Usted Pedro; Ovo! Szondi Owen, Treadmill! RAPACIOUS ELDERBERRY. Swat Vjeran Phoenix. EVENTUAL 100 Km, Hungary.

The concept of a global city (or world city) is of a city that has a direct and tangible effect on global affairs through socioeconomic means. The term has become increasingly familiar, because of the rise of globalization.

Finger food is food meant to be eaten directly using the hands, in contrast to food eaten with a knife and fork, spoon, chopsticks, or other utensils. In most cultures, food is almost always eaten with the hands; for example, Ethiopian cuisine is eaten by rolling various dishes up in 'injera' bread. Foods considered street foods are frequently, though not exclusively, finger foods.

Won Ljubljana Ciara. FORESIGHT REGARDING. Nora Uggs Kkora Ijyalsila Lvisa. ELEPHANTINE. Zwieback Kumar Wjec Tjena.

In many western countries there are catering businesses that supply finger foods for events such as weddings, engagements, birthdays and other milestone celebrations. For weddings, in particular, finger foods are becoming more popular because they are less expensive and offer more flexibility with menu choices.

Tdawson Indra. Okra Ssamjang Pfizer. WINSLOW OVERCAST. NECKBONE. Asma Igor Ouro Oasis Tamara. Tflon Zlotys.

Paper is a thin material produced by pressing together moist fibres of cellulose pulp derived from wood, rags or grasses, and drying them into flexible sheets. It is a versatile material with many uses, including writing, printing, packaging, cleaning, decorating, and a number of industrial and construction processes. Papers are essential in legal or non-legal documentation.

In the 13th century, the knowledge and uses of paper spread from China through the Middle East to medieval Europe, where the first water powered paper mills were built. Because paper was introduced to the West through the city of Baghdad, it was first called bagdatikos.

Slabo 13px, composite algorithm

Blue is one of the three primary colours of pigments in painting and traditional colour theory, as well as in the RGB colour model. It lies between violet and green on the spectrum of visible light. The eye perceives blue when observing light with a dominant wavelength between approximately 450 and 495 nanometres. Most blues contain a slight mixture of other colours; azure contains some green, while ultramarine contains some violet. The clear daytime sky and the deep sea appear blue because of an optical effect known as Rayleigh scattering. An optical effect called Tyndall scattering explains blue eyes. Distant objects appear more blue because of another optical effect called aerial perspective.

Cummin Atmo Yclept Xray. GADFLY. Keratin Yves Fokionas Ptolomeu. PATRICIAN. Udon Ybelote Oona Gracinda.

Baseball is a bat-and-ball game played between two opposing teams who take turns batting and fielding. The game proceeds when a player on the fielding team, called the pitcher, throws a ball which a player on the batting team tries to hit with a bat. The objectives of the offensive team (batting team) are to hit the ball into the field of play, and to run the bases—having its runners advance counter-clockwise around four bases to score what are called "runs".

Null Level INJUNCTION. Sooner, Rather, Later. HALLWAY AVOIDANCE. Porches, Swans, Lyrics, Sanity. DRAWN. Copper, Cartwheel. Ivan, Zwingli.

Crasis is a type of contraction in which two vowels or diphthongs merge into one new vowel or diphthong, making one word out of two. Crasis occurs in Portuguese, French and Arabic as well as in Ancient Greek, for which it was first described.

A language family is a group of languages related through descent from a common ancestral language or parental language, called the proto-language of that family. The term "family" reflects the tree model of language origination in historical linguistics, which makes use of a metaphor comparing languages to people in a biological family tree, or in a subsequent modification, to species in a phylogenetic tree of evolutionary taxonomy.

Linguists therefore describe the daughter languages within a language family as being genetically related.

The Silk Road derives its name from the lucrative trade in silk carried out along its length, beginning in the Han dynasty in China (207 BCE–220 CE). The Han dynasty expanded the Central Asian section of the trade routes around 114 BCE through the missions and explorations of the Chinese imperial envoy Zhang Qian, as well as several military conquests.

The Silk Road trade played a significant role in the development of the civilizations of China, Korea, Japan, the Indian subcontinent, Iran, Europe, the Horn of Africa and Arabia, opening long-distance political and economic relations between the civilizations.

Zoological Timpani Flexbox, TOPCOAT. ILL EXPLORATION, DEFT CAVALIER. Retread, SSL, Hushed Bobcat Petersburg, Klaxons.

For at least a portion of its life, a star shines due to thermonuclear fusion of hydrogen into helium in its core, releasing energy that traverses the star's interior and then radiates into outer space.

Binary and multi-star systems consist of two or more stars that are gravitationally bound and generally move around each other in stable orbits. When two such stars have a relatively close orbit, their gravitational interaction can have a significant impact on their evolution. Stars can form part of a much larger gravitationally bound structure, such as a star cluster or a galaxy.

Andes Usted Pedro; Ovo! Szondi Owen, Treadmill! RAPACIOUS ELDERBERRY. Swat Vjeran Phoenix. EVENTUAL 100 Km, Hungary.

The concept of a global city (or world city) is of a city that has a direct and tangible effect on global affairs through socioeconomic means. The term has become increasingly familiar, because of the rise of globalization.

Finger food is food meant to be eaten directly using the hands, in contrast to food eaten with a knife and fork, spoon, chopsticks, or other utensils. In most cultures, food is almost always eaten with the hands; for example, Ethiopian cuisine is eaten by rolling various dishes up in 'injera' bread. Foods considered street foods are frequently, though not exclusively, finger foods.

Won Ljubljana Ciara. FORESIGHT REGARDING. Nora Uggs Kkora Ijyalsila Lvisa. ELEPHANTINE. Zwieback Kumar Wjec Tjena.

In many western countries there are catering businesses that supply finger foods for events such as weddings, engagements, birthdays and other milestone celebrations. For weddings, in particular, finger foods are becoming more popular because they are less expensive and offer more flexibility with menu choices.

Tdawson Indra. Okra Ssamjang Pfizer. WINSLOW OVERCAST. NECKBONE. Asma Igor Ouro Oasis Tamara. Tflon Zlotys.

Paper is a thin material produced by pressing together moist fibres of cellulose pulp derived from wood, rags or grasses, and drying them into flexible sheets. It is a versatile material with many uses, including writing, printing, packaging, cleaning, decorating, and a number of industrial and construction processes. Papers are essential in legal or non-legal documentation.

In the 13th century, the knowledge and uses of paper spread from China through the Middle East to medieval Europe, where the first water powered paper mills were built. Because paper was introduced to the West through the city of Baghdad, it was first called bagdatikos.

Slabo 13px, rival kf algorithm

Blue is one of the three primary colours of pigments in painting and traditional colour theory, as well as in the RGB colour model. It lies between violet and green on the spectrum of visible light. The eye perceives blue when observing light with a dominant wavelength between approximately 450 and 495 nanometres. Most blues contain a slight mixture of other colours; azure contains some green, while ultramarine contains some violet. The clear daytime sky and the deep sea appear blue because of an optical effect known as Rayleigh scattering. An optical effect called Tyndall scattering explains blue eyes. Distant objects appear more blue because of another optical effect called aerial perspective.

Cummin Atmo Yclept Xray. GADFLY. Keratin Yves Fokionas Ptolomeu. PATRICIAN. Udon Ybelote Oona Gracinda.

Baseball is a bat-and-ball game played between two opposing teams who take turns batting and fielding. The game proceeds when a player on the fielding team, called the pitcher, throws a ball which a player on the batting team tries to hit with a bat. The objectives of the offensive team (batting team) are to hit the ball into the field of play, and to run the bases—having its runners advance counter-clockwise around four bases to score what are called "runs".

Null Level INJUNCTION. Sooner, Rather, Later. HALLWAY AVOIDANCE. Porches, Swans, Lyrics, Sanity. DRAWN. Copper, Cartwheel. Ivan, Zwingli.

Crasis is a type of contraction in which two vowels or diphthongs merge into one new vowel or diphthong, making one word out of two. Crasis occurs in Portuguese, French and Arabic as well as in Ancient Greek, for which it was first described.

A language family is a group of languages related through descent from a common ancestral language or parental language, called the proto-language of that family. The term "family" reflects the tree model of language origination in historical linguistics, which makes use of a metaphor comparing languages to people in a biological family tree, or in a subsequent modification, to species in a phylogenetic tree of evolutionary taxonomy.

Linguists therefore describe the daughter languages within a language family as being genetically related.

The Silk Road derives its name from the lucrative trade in silk carried out along its length, beginning in the Han dynasty in China (207 BCE–220 CE). The Han dynasty expanded the Central Asian section of the trade routes around 114 BCE through the missions and explorations of the Chinese imperial envoy Zhang Qian, as well as several military conquests.

The Silk Road trade played a significant role in the development of the civilizations of China, Korea, Japan, the Indian subcontinent, Iran, Europe, the Horn of Africa and Arabia, opening long-distance political and economic relations between the civilizations.

Zoological Timpani Flexbox, TOPCOAT. ILL EXPLORATION, DEFT CAVALIER. Retread, SSL, Hushed Bobcat Petersburg, Klaxons.

For at least a portion of its life, a star shines due to thermonuclear fusion of hydrogen into helium in its core, releasing energy that traverses the star's interior and then radiates into outer space.

Binary and multi-star systems consist of two or more stars that are gravitationally bound and generally move around each other in stable orbits. When two such stars have a relatively close orbit, their gravitational interaction can have a significant impact on their evolution. Stars can form part of a much larger gravitationally bound structure, such as a star cluster or a galaxy.

Andes Usted Pedro; Ovo! Szondi Owen, Treadmill! RAPACIOUS ELDERBERRY. Swat Vjeran Phoenix. EVENTUAL 100 Km, Hungary.

The concept of a global city (or world city) is of a city that has a direct and tangible effect on global affairs through socioeconomic means. The term has become increasingly familiar, because of the rise of globalization.

Finger food is food meant to be eaten directly using the hands, in contrast to food eaten with a knife and fork, spoon, chopsticks, or other utensils. In most cultures, food is almost always eaten with the hands; for example, Ethiopian cuisine is eaten by rolling various dishes up in 'injera' bread. Foods considered street foods are frequently, though not exclusively, finger foods.

Won Ljubljana Ciara. FORESIGHT REGARDING. Nora Uggs Kkora Ijyalsila Lvisa. ELEPHANTINE. Zwieback Kumar Wjec Tjena.

In many western countries there are catering businesses that supply finger foods for events such as weddings, engagements, birthdays and other milestone celebrations. For weddings, in particular, finger foods are becoming more popular because they are less expensive and offer more flexibility with menu choices.

Tdawson Indra. Okra Ssamjang Pfizer. WINSLOW OVERCAST. NECKBONE. Asma Igor Ouro Oasis Tamara. Tflon Zlotys.

Paper is a thin material produced by pressing together moist fibres of cellulose pulp derived from wood, rags or grasses, and drying them into flexible sheets. It is a versatile material with many uses, including writing, printing, packaging, cleaning, decorating, and a number of industrial and construction processes. Papers are essential in legal or non-legal documentation.

In the 13th century, the knowledge and uses of paper spread from China through the Middle East to medieval Europe, where the first water powered paper mills were built. Because paper was introduced to the West through the city of Baghdad, it was first called bagdatikos.

Slabo 27px, control (original fitting)

An island or isle is any piece of sub-continental land that is surrounded by water. Very small islands such as emergent land features on atolls can be called islets, skerries, cays or keys. An island in a river or a lake island may be called an eyot or ait, and a small island off the coast may be called a holm. A grouping of geographically or geologically related islands is called an archipelago, such as the Philippines.

Tutors Of Tennis. Volvo. NON-AUTOMATIC. Jalapeno. Illicit. Affluent. Capital. LOCALIZED. Repurpose. Foil. War. Pond. Dresser. Cedric. Korn.

False cognates are pairs of words that seem to be cognates because of similar sounds and meaning, but have different etymologies; they can be within the same language or from different languages, even within the same family. For example, the English word 'dog' and the Mbabaram word 'dog' have exactly the same meaning and very similar pronunciations, but by complete coincidence. Likewise, English much and Spanish mucho which came by their similar meanings via completely different Proto-Indo-European roots. Other examples include "island" and "isle".

Donated Heptagons Only Forever. RIVERWALKS HOPEFULNESS WASHBOARD. Judicial Starlight Seersucker Vexed Affluence.

In etymology, back-formation is the process of creating a new lexeme by removing actual or supposed affixes. The resulting neologism is called a back-formation, a term coined by James Murray in 1889. (OED online preserves its first use of 'back-formation' from 1889 in the definition of to burgle; from burglar.) Back-formation may be particularly common in English given that many English words are borrowed from Latin, French and Greek, which together provide English a large range of common affixes.

Many words with affixes have entered English, such as dismantle and dishevelled, so it may be easy to believe that these are formed from roots such as mantle (meaning to put something together) and shevelled (meaning well-dressed) although these words have no history of existing in English. Many words came into English by this route: 'pease' was once a mass noun but was reinterpreted as a plural, leading to the back-formation 'pea'.

Dog sports are sports in which dogs participate. There are a broad range of activities that may be included under the term. Many of the sports have human interacting with dogs to give direction (e.g., obedience trials, herding trials, and sled dog racing). Other events, such as greyhound racing, have minimal involvement from humans during the competition. Some of the sports are viewed by a wide audience and may be televised, while others are practiced mainly by enthusiasts.

Yuccas Oeric Uehiro Erica. RYE. Xhaka Qinetiq Gwen Ifran Ylang. TANK-TOP. Json Opal Zircon.

The short ton is a unit of mass equal to 2,000 pounds-mass (907.18474 kg). The unit is most commonly used in the United States where it is known simply as the ton. The short ton sometimes describes force. One short-ton contains 2,000 pounds-mass, which converted into slugs and multiplied by one standard gravity applies a weight of 2,000 pounds-force as per Newton's second law of motion.

Both the long and short ton are defined as 20 hundredweights, but a hundredweight is 100 pounds (45.359237 kg) in the U.S. system (short or net hundredweight) and 112 pounds (50.802345 kg) in the imperial system (long or gross hundredweight).

Bquiet Cravo Qrcode Otoman. QARTH. Sven Jquery Qtar, YAVANNA. Leonor Bpolar Uber. WARLIKE. Enter Ethan.

The United Kingdom used to use the imperial ton, long ton in the USA. The use of the imperial or short ton ceased under the Weights and Measures Act 1985. Most Commonwealth countries followed British practice with the exception of Canada, which continued to use short tons as well as long tons but now predominantly uses metric tons (tonnes).

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3″ (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3″ (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Slabo 27px, composite algorithm

An island or isle is any piece of sub-continental land that is surrounded by water. Very small islands such as emergent land features on atolls can be called islets, skerries, cays or keys. An island in a river or a lake island may be called an eyot or ait, and a small island off the coast may be called a holm. A grouping of geographically or geologically related islands is called an archipelago, such as the Philippines.

Tutors Of Tennis. Volvo. NON-AUTOMATIC. Jalapeno. Illicit. Affluent. Capital. LOCALIZED. Repurpose. Foil. War. Pond. Dresser. Cedric. Korn.

False cognates are pairs of words that seem to be cognates because of similar sounds and meaning, but have different etymologies; they can be within the same language or from different languages, even within the same family. For example, the English word 'dog' and the Mbabaram word 'dog' have exactly the same meaning and very similar pronunciations, but by complete coincidence. Likewise, English much and Spanish mucho which came by their similar meanings via completely different Proto-Indo-European roots. Other examples include "island" and "isle".

Donated Heptagons Only Forever. RIVERWALKS HOPEFULNESS WASHBOARD. Judicial Starlight Seersucker Vexed Affluence.

In etymology, back-formation is the process of creating a new lexeme by removing actual or supposed affixes. The resulting neologism is called a back-formation, a term coined by James Murray in 1889. (OED online preserves its first use of 'back-formation' from 1889 in the definition of to burgle; from burglar.) Back-formation may be particularly common in English given that many English words are borrowed from Latin, French and Greek, which together provide English a large range of common affixes.

Many words with affixes have entered English, such as dismantle and dishevelled, so it may be easy to believe that these are formed from roots such as mantle (meaning to put something together) and shevelled (meaning well-dressed) although these words have no history of existing in English. Many words came into English by this route: 'pease' was once a mass noun but was reinterpreted as a plural, leading to the back-formation 'pea'.

Dog sports are sports in which dogs participate. There are a broad range of activities that may be included under the term. Many of the sports have human interacting with dogs to give direction (e.g., obedience trials, herding trials, and sled dog racing). Other events, such as greyhound racing, have minimal involvement from humans during the competition. Some of the sports are viewed by a wide audience and may be televised, while others are practiced mainly by enthusiasts.

Yuccas Oeric Uehiro Erica. RYE. Xhaka Qinetiq Gwen Ifran Ylang. TANK-TOP. Json Opal Zircon.

The short ton is a unit of mass equal to 2,000 pounds-mass (907.18474 kg). The unit is most commonly used in the United States where it is known simply as the ton. The short ton sometimes describes force. One short-ton contains 2,000 pounds-mass, which converted into slugs and multiplied by one standard gravity applies a weight of 2,000 pounds-force as per Newton's second law of motion.

Both the long and short ton are defined as 20 hundredweights, but a hundredweight is 100 pounds (45.359237 kg) in the U.S. system (short or net hundredweight) and 112 pounds (50.802345 kg) in the imperial system (long or gross hundredweight).

Bquiet Cravo Qrcode Otoman. QARTH. Sven Jquery Qtar, YAVANNA. Leonor Bpolar Uber. WARLIKE. Enter Ethan.

The United Kingdom used to use the imperial ton, long ton in the USA. The use of the imperial or short ton ceased under the Weights and Measures Act 1985. Most Commonwealth countries followed British practice with the exception of Canada, which continued to use short tons as well as long tons but now predominantly uses metric tons (tonnes).

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3″ (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3″ (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Slabo 27px, rival kf algorithm

An island or isle is any piece of sub-continental land that is surrounded by water. Very small islands such as emergent land features on atolls can be called islets, skerries, cays or keys. An island in a river or a lake island may be called an eyot or ait, and a small island off the coast may be called a holm. A grouping of geographically or geologically related islands is called an archipelago, such as the Philippines.

Tutors Of Tennis. Volvo. NON-AUTOMATIC. Jalapeno. Illicit. Affluent. Capital. LOCALIZED. Repurpose. Foil. War. Pond. Dresser. Cedric. Korn.

False cognates are pairs of words that seem to be cognates because of similar sounds and meaning, but have different etymologies; they can be within the same language or from different languages, even within the same family. For example, the English word 'dog' and the Mbabaram word 'dog' have exactly the same meaning and very similar pronunciations, but by complete coincidence. Likewise, English much and Spanish mucho which came by their similar meanings via completely different Proto-Indo-European roots. Other examples include "island" and "isle".

Donated Heptagons Only Forever. RIVERWALKS HOPEFULNESS WASHBOARD. Judicial Starlight Seersucker Vexed Affluence.

In etymology, back-formation is the process of creating a new lexeme by removing actual or supposed affixes. The resulting neologism is called a back-formation, a term coined by James Murray in 1889. (OED online preserves its first use of 'back-formation' from 1889 in the definition of to burgle; from burglar.) Back-formation may be particularly common in English given that many English words are borrowed from Latin, French and Greek, which together provide English a large range of common affixes.

Many words with affixes have entered English, such as dismantle and dishevelled, so it may be easy to believe that these are formed from roots such as mantle (meaning to put something together) and shevelled (meaning well-dressed) although these words have no history of existing in English. Many words came into English by this route: 'pease' was once a mass noun but was reinterpreted as a plural, leading to the back-formation 'pea'.

Dog sports are sports in which dogs participate. There are a broad range of activities that may be included under the term. Many of the sports have human interacting with dogs to give direction (e.g., obedience trials, herding trials, and sled dog racing). Other events, such as greyhound racing, have minimal involvement from humans during the competition. Some of the sports are viewed by a wide audience and may be televised, while others are practiced mainly by enthusiasts.

Yuccas Oeric Uehiro Erica. RYE. Xhaka Qinetiq Gwen Ifran Ylang. TANK-TOP. Json Opal Zircon.

The short ton is a unit of mass equal to 2,000 pounds-mass (907.18474 kg). The unit is most commonly used in the United States where it is known simply as the ton. The short ton sometimes describes force. One short-ton contains 2,000 pounds-mass, which converted into slugs and multiplied by one standard gravity applies a weight of 2,000 pounds-force as per Newton's second law of motion.

Both the long and short ton are defined as 20 hundredweights, but a hundredweight is 100 pounds (45.359237 kg) in the U.S. system (short or net hundredweight) and 112 pounds (50.802345 kg) in the imperial system (long or gross hundredweight).

Bquiet Cravo Qrcode Otoman. QARTH. Sven Jquery Qtar, YAVANNA. Leonor Bpolar Uber. WARLIKE. Enter Ethan.

The United Kingdom used to use the imperial ton, long ton in the USA. The use of the imperial or short ton ceased under the Weights and Measures Act 1985. Most Commonwealth countries followed British practice with the exception of Canada, which continued to use short tons as well as long tons but now predominantly uses metric tons (tonnes).

Ivan, Yttria, Qadreh, Bored, Akinesia, Knee, Ikarus, Eugenio, Fjord, Thor, Vera, Qonnect.

In metalworking and jewellery making, casting is a process in which a liquid metal is somehow delivered into a mold (it is usually delivered by a crucible) that contains a hollow shape (i.e., a 3-dimensional negative image) of the intended shape. The metal is poured into the mold through a hollow channel called a 'sprue'. The metal and mold are then cooled, and the metal part (the casting) is extracted. Casting is most often used for making complex shapes that would be difficult or uneconomical to make by other methods.

Kjara, Uummannaq, Francisca, ATLAS V, Liliana, Kg, Lhaam, Kshir, Tkmax, Icaro, Uqasha, Njck, Elmo.

The tropics are the region of the Earth surrounding the Equator. They are delimited in latitude by the Tropic of Cancer in the Northern Hemisphere at 23°26′12.3″ (or 23.43674°) N and the Tropic of Capricorn in the Southern Hemisphere at 23°26′12.3″ (or 23.43674°) S; these latitudes correspond to the axial tilt of the Earth. The tropics are also referred to as the tropical zone and the torrid zone (see geographical zone). The tropics include all the areas on the Earth where the Sun contacts a point directly overhead at least once during the solar year (which is a subsolar point) - thus the latitude of the tropics is roughly equal to the angle of the Earth's axial tilt.

Source Sans Pro Regular, control (original fitting)

Blues is a music genre and musical form which was originated in the Deep South of the United States around the 1870s by African Americans from roots in African musical traditions, African-American work songs, and spirituals. Blues incorporated spirituals, work songs, field hollers, shouts, chants, and rhymed simple narrative ballads. The blues form, ubiquitous in jazz, rhythm and blues and rock and roll, is characterized by the call-and-response pattern, the blues scale and specific chord progressions, of which the twelve-bar blues is the most common. Blue notes (or "worried notes"), usually thirds, fifths or sevenths flattened in pitch are also an essential part of the sound. Blues shuffles or walking bass reinforce the trance-like rhythm and form a repetitive effect known as the groove.

Butterflies have the typical four-stage insect life cycle. Winged adults lay eggs on the food plant on which their larvae, known as caterpillars, will feed. The caterpillars grow, sometimes very rapidly, and when fully developed, pupate in a chrysalis. When metamorphosis is complete, the pupal skin splits, the adult insect climbs out, and after its wings have expanded and dried, it flies off. Some butterflies, especially in the tropics, have several generations in a year, while others have a single generation, and a few in cold locations may take several years to pass through their entire life cycle.

Zaragosa, Isaac, Osso, Carlos, Rita, Gjetost, Vladmir, Ipil, Xavier, Giacometti, Vremea, Ondine.

Leaf vegetables, also called leafy greens, salad greens, pot herbs, vegetable greens, or simply greens, are plant leaves eaten as a vegetable, sometimes accompanied by tender petioles and shoots. Although they come from a very wide variety of plants, most share a great deal with other leaf vegetables in nutrition and cooking methods.

REVEAL, Unguent, Ghana, Westworld. Inundate. INUNDATE. Century, Ball Lightning. Samosa, Declarative Actions, Estimate.

Winter sports or winter activities are competitive sports or non-competitive recreational activities which are played on snow or ice. Most are variations of skiing, ice skating and sledding. Traditionally, such games were only played in cold areas during winter, but artificial snow and artificial ice allow more flexibility. Artificial ice can be used to provide ice rinks for ice skating, ice hockey, and bandy in a milder climate.

Pajamas or pyjamas (UK), often shortened to PJs or jammies, can refer to several related types of clothing originating from the Indian subcontinent. In the Western world, pajamas are loose-fitting garments derived from the original garment and worn chiefly for sleeping, but sometimes for lounging as well by both the sexes. More generally, pajamas may refer to several garments, for both daywear and nightwear, derived from traditional pajamas and involving variations of style and material.

Clothing (also known as clothes, apparel and attire) is a collective term for items worn on the body. Clothing is typically made of fabrics or textiles but over time has included garments made from animal skin or other thin sheets of materials put together. The wearing of clothing is mostly restricted to human beings and is a feature of all human societies. The amount and type of clothing worn depends on gender, body type, social, and geographic considerations.

Bwana, Mr., Jaguar, Natalia, Gonçalo, Yemen. Joan, Apple, Iqlas. Itan, Ezra, Smell. BLOCK, FOREFOOT, AVATAR.

A textile is a flexible material consisting of a network of natural or artificial fibers (yarn or thread). Yarn is produced by spinning raw fibres of wool, flax, cotton, hemp, or other materials to produce long strands. Textiles are formed by weaving, knitting, crocheting, knotting or tatting, felting, or braiding.

The related words "fabric" and "cloth" and "material" are often used in textile assembly trades (such as tailoring and dressmaking) as synonyms for textile. However, there are subtle differences in these terms in specialized usage.

Kraken, Zucchini, Porto. LOCKBOX, HINGED, PAINTER. Fulki, Dobby, Moria, Aztecs. Rss, Ajar, ATOM. Llorax, Action, Uzo.

In printing, a stereotype, also known as a cliché, stereoplate or simply a stereo, was a "solid plate of type metal, cast from a papier-mâché or plaster mould (called a flong) taken from the surface of a forme of type" and used for printing instead of the original.

Winston Aqua, Adore Zombie. GRAPH AGE. Easter Ofelia, Ying. Ferreira Rwanda. THOUSAND FACES. Tmobile Sbavatura. Purity.

In the days of set movable type, printing involved placing individual letters (called types) plus other elements (including leading and furniture) into a block called a chase. Cumulatively, this full setup for printing a single page was called a forme. Ink was then applied to the forme, pressed against paper and a printed page was made. This process of creating formes was labor-intensive, costly and prevented the printer from using their types, leading, furniture and chases for other work. Furthermore, printers who underestimated demand would be forced to reset the type for subsequent print runs.

OBSOLETE. Qechua Aumento, Oman Iman. Mt Xchange, Aaron. Algarve, Prato, Vcelka. Aided In Ufology. CORDUROY.

Phototypesetting is a method of setting type, rendered obsolete with the popularity of the personal computer and desktop publishing software, that uses a photographic process to generate columns of type on a scroll of photographic paper.

The first phototypesetters quickly project light through a film negative image of an individual character in a font, then through a lens that magnifies or reduces the size of the character onto photographic paper, which is collected on a spool in a light-proof canister.

Source Sans Pro Regular, rival kf algorithm

Blues is a music genre and musical form which was originated in the Deep South of the United States around the 1870s by African Americans from roots in African musical traditions, African-American work songs, and spirituals. Blues incorporated spirituals, work songs, field hollers, shouts, chants, and rhymed simple narrative ballads. The blues form, ubiquitous in jazz, rhythm and blues and rock and roll, is characterized by the call-and-response pattern, the blues scale and specific chord progressions, of which the twelve-bar blues is the most common. Blue notes (or "worried notes"), usually thirds, fifths or sevenths flattened in pitch are also an essential part of the sound. Blues shuffles or walking bass reinforce the trance-like rhythm and form a repetitive effect known as the groove.

Butterflies have the typical four-stage insect life cycle. Winged adults lay eggs on the food plant on which their larvae, known as caterpillars, will feed. The caterpillars grow, sometimes very rapidly, and when fully developed, pupate in a chrysalis. When metamorphosis is complete, the pupal skin splits, the adult insect climbs out, and after its wings have expanded and dried, it flies off. Some butterflies, especially in the tropics, have several generations in a year, while others have a single generation, and a few in cold locations may take several years to pass through their entire life cycle.

Zaragosa, Isaac, Osso, Carlos, Rita, Gjetost, Vladmir, Ipil, Xavier, Giacometti, Vremea, Ondine.

Leaf vegetables, also called leafy greens, salad greens, pot herbs, vegetable greens, or simply greens, are plant leaves eaten as a vegetable, sometimes accompanied by tender petioles and shoots. Although they come from a very wide variety of plants, most share a great deal with other leaf vegetables in nutrition and cooking methods.

REVEAL, Unguent, Ghana, Westworld. Inundate. INUNDATE. Century, Ball Lightning. Samosa, Declarative Actions, Estimate.

Winter sports or winter activities are competitive sports or non-competitive recreational activities which are played on snow or ice. Most are variations of skiing, ice skating and sledding. Traditionally, such games were only played in cold areas during winter, but artificial snow and artificial ice allow more flexibility. Artificial ice can be used to provide ice rinks for ice skating, ice hockey, and bandy in a milder climate.

Pajamas or pyjamas (UK), often shortened to PJs or jammies, can refer to several related types of clothing originating from the Indian subcontinent. In the Western world, pajamas are loose-fitting garments derived from the original garment and worn chiefly for sleeping, but sometimes for lounging as well by both the sexes. More generally, pajamas may refer to several garments, for both daywear and nightwear, derived from traditional pajamas and involving variations of style and material.

Clothing (also known as clothes, apparel and attire) is a collective term for items worn on the body. Clothing is typically made of fabrics or textiles but over time has included garments made from animal skin or other thin sheets of materials put together. The wearing of clothing is mostly restricted to human beings and is a feature of all human societies. The amount and type of clothing worn depends on gender, body type, social, and geographic considerations.

Bwana, Mr., Jaguar, Natalia, Gonçalo, Yemen. Joan, Apple, Iqlas. Itan, Ezra, Smell. BLOCK, FOREFOOT, AVATAR.

A textile is a flexible material consisting of a network of natural or artificial fibers (yarn or thread). Yarn is produced by spinning raw fibres of wool, flax, cotton, hemp, or other materials to produce long strands. Textiles are formed by weaving, knitting, crocheting, knotting or tatting, felting, or braiding.

The related words "fabric" and "cloth" and "material" are often used in textile assembly trades (such as tailoring and dressmaking) as synonyms for textile. However, there are subtle differences in these terms in specialized usage.

Kraken, Zucchini, Porto. LOCKBOX, HINGED, PAINTER. Fulki, Dobby, Moria, Aztecs. Rss, Ajar, ATOM. Llorax, Action, Uzo.

In printing, a stereotype, also known as a cliché, stereoplate or simply a stereo, was a "solid plate of type metal, cast from a papier-mâché or plaster mould (called a flong) taken from the surface of a forme of type" and used for printing instead of the original.

Winston Aqua, Adore Zombie. GRAPH AGE. Easter Ofelia, Ying. Ferreira Rwanda. THOUSAND FACES. Tmobile Sbavatura. Purity.

In the days of set movable type, printing involved placing individual letters (called types) plus other elements (including leading and furniture) into a block called a chase. Cumulatively, this full setup for printing a single page was called a forme. Ink was then applied to the forme, pressed against paper and a printed page was made. This process of creating formes was labor-intensive, costly and prevented the printer from using their types, leading, furniture and chases for other work. Furthermore, printers who underestimated demand would be forced to reset the type for subsequent print runs.

OBSOLETE. Qechua Aumento, Oman Iman. Mt Xchange, Aaron. Algarve, Prato, Vcelka. Aided In Ufology. CORDUROY.

Phototypesetting is a method of setting type, rendered obsolete with the popularity of the personal computer and desktop publishing software, that uses a photographic process to generate columns of type on a scroll of photographic paper.

The first phototypesetters quickly project light through a film negative image of an individual character in a font, then through a lens that magnifies or reduces the size of the character onto photographic paper, which is collected on a spool in a light-proof canister.

Source Serif 4 Regular, control (original fitting)

Printing is a process for reproducing text and images using a master form or template. The earliest non-paper products involving printing include cylinder seals and objects such as the Cyrus Cylinder and the Cylinders of Nabonidus. The earliest known form of printing as applied to paper was woodblock printing, which appeared in China before 220 AD. Later developments in printing technology include the movable type invented by Bi Sheng around 1040 AD and the printing press invented by Johannes Gutenberg in the 15th century. The technology of printing played a key role in the development of the Renaissance and the scientific revolution, and laid the material basis for the modern knowledge-based economy and the spread of learning to the masses.

Today, Tomorrow, Refinery. VALUES. Minimum, Portent, Tyres. Czar. Most Ushers Ionize. HIJINKS. Food Bookkeeping Last.

Gutenberg was a German blacksmith, goldsmith, inventor, printer, and publisher who introduced printing to Europe with the printing press. His introduction of mechanical movable type printing to Europe started the Printing Revolution and is regarded as a milestone of the second millennium, ushering in the modern period of human history.

Kent Gotham's Voroni Diagram. Radial Robots. West SOHO. Zones Accept Fast Youths. OWLS. Cooling Xanax Even In-flight.

Type metal (sometimes called hot metal) refers to the metal alloys used in traditional typefounding and hot metal typesetting. Historically, type metal was an alloy of lead, tin and antimony in different proportions depending on the application, be it individual character mechanical casting for hand setting, mechanical line casting or individual character mechanical typesetting and stereo plate casting. The proportions used are in the range: lead 50–86%, antimony 11–30% and tin 3–20%. Antimony and tin are added to lead for durability while reducing the difference between the coefficients of expansion of the matrix and the alloy. Apart from durability, the general requirements for type-metal are that it should produce a true and sharp cast, and retain correct dimensions and form after cooling down.

Gravity (from Latin gravitas, meaning 'weight'), or gravitation, is a natural phenomenon by which all things with mass or energy —including planets, stars, galaxies, and even light—are brought toward (or gravitate toward) one another. On Earth, gravity gives weight to physical objects, and the Moon's gravity causes the ocean tides. The gravitational attraction of the original gaseous matter present in the Universe caused it to begin coalescing, forming stars—and for the stars to group together into galaxies—so gravity is responsible for many of the large-scale structures in the Universe. Gravity has an infinite range, although its effects become increasingly weaker on farther objects.

Gravity is most accurately described by the general theory of relativity (proposed by Albert Einstein in 1915) which describes gravity not as a force, but as a consequence of the curvature of spacetime caused by the uneven distribution of mass.

Aviator, Xgood, Nuno, Bhangras, Xone, Odor, Snot, Quando, Vnom, Kwanza, Izzy, Nimrod.

The most extreme example of this curvature of spacetime is a black hole, from which nothing—not even light—can escape once past the black hole's event horizon. However, for most applications, gravity is well approximated by Newton's law of universal gravitation, which describes gravity as a force which causes any two bodies to be attracted to each other, with the force proportional to the product of their masses and inversely proportional to the square of the distance between them.

Gravity is the weakest of the four fundamental interactions of physics, approximately 10^38 times weaker than the strong interaction, 10^36 times weaker than the electromagnetic force and 10^29 times weaker than the weak interaction.

For example, gravity causes the Earth and the other planets to orbit the Sun, it also causes the Moon to orbit the Earth, and causes the formation of tides, the formation and evolution of the Solar System, stars and galaxies.

Archimedes discovered the center of gravity of a triangle. The triangle, however, does not generalize on its own.

Cooking or cookery is the art, technology, science and craft of preparing food for consumption. Cooking techniques and ingredients vary widely across the world, from grilling food over an open fire to using electric stoves, to baking in various types of ovens, reflecting unique environmental, economic, and cultural traditions and trends. The ways or types of cooking also depend on the skill and type of training an individual cook has. Cooking is done both by people in their own dwellings and by professional cooks and chefs in restaurants and other food establishments. Cooking can also occur through chemical reactions without the presence of heat, such as in ceviche, a traditional South American dish where fish is cooked with the acids in lemon or lime juice.

VARNISH. Cjay Writing. Abacus, Ultimate Soul. Xuma, Time. PANTHERIZE. Gerardo Pneumatic. Jvlivs, Uakari Valeria.

The expansion of agriculture, commerce, trade, and transportation between civilizations in different regions offered cooks many new ingredients. New inventions and technologies, such as the invention of pottery for holding and boiling water, expanded cooking techniques. Some modern cooks apply advanced scientific techniques to food preparation to further enhance the flavor of the dish served.

Source Serif 4 Regular, composite algorithm

Printing is a process for reproducing text and images using a master form or template. The earliest non-paper products involving printing include cylinder seals and objects such as the Cyrus Cylinder and the Cylinders of Nabonidus. The earliest known form of printing as applied to paper was woodblock printing, which appeared in China before 220 AD. Later developments in printing technology include the movable type invented by Bi Sheng around 1040 AD and the printing press invented by Johannes Gutenberg in the 15th century. The technology of printing played a key role in the development of the Renaissance and the scientific revolution, and laid the material basis for the modern knowledge-based economy and the spread of learning to the masses.

Today, Tomorrow, Refinery. VALUES. Minimum, Portent, Tyres. Czar. Most Ushers Ionize. HIJINKS. Food Bookkeeping Last.

Gutenberg was a German blacksmith, goldsmith, inventor, printer, and publisher who introduced printing to Europe with the printing press. His introduction of mechanical movable type printing to Europe started the Printing Revolution and is regarded as a milestone of the second millennium, ushering in the modern period of human history.

Kent Gotham's Voroni Diagram. Radial Robots. West SOHO. Zones Accept Fast Youths. OWLS. Cooling Xanax Even In-flight.

Type metal (sometimes called hot metal) refers to the metal alloys used in traditional typefounding and hot metal typesetting. Historically, type metal was an alloy of lead, tin and antimony in different proportions depending on the application, be it individual character mechanical casting for hand setting, mechanical line casting or individual character mechanical typesetting and stereo plate casting. The proportions used are in the range: lead 50–86%, antimony 11–30% and tin 3–20%. Antimony and tin are added to lead for durability while reducing the difference between the coefficients of expansion of the matrix and the alloy. Apart from durability, the general requirements for type-metal are that it should produce a true and sharp cast, and retain correct dimensions and form after cooling down.

Gravity (from Latin gravitas, meaning 'weight'), or gravitation, is a natural phenomenon by which all things with mass or energy—including planets, stars, galaxies, and even light—are brought toward (or gravitate toward) one another. On Earth, gravity gives weight to physical objects, and the Moon's gravity causes the ocean tides. The gravitational attraction of the original gaseous matter present in the Universe caused it to begin coalescing, forming stars—and for the stars to group together into galaxies—so gravity is responsible for many of the large-scale structures in the Universe. Gravity has an infinite range, although its effects become increasingly weaker on farther objects.

Gravity is most accurately described by the general theory of relativity (proposed by Albert Einstein in 1915) which describes gravity not as a force, but as a consequence of the curvature of spacetime caused by the uneven distribution of mass.

Aviator, Xgood, Nuno, Bhangras, Xone, Odor, Snot, Quando, Vnom, Kwanza, Izzy, Nimrod.

The most extreme example of this curvature of spacetime is a black hole, from which nothing—not even light—can escape once past the black hole's event horizon. However, for most applications, gravity is well approximated by Newton's law of universal gravitation, which describes gravity as a force which causes any two bodies to be attracted to each other, with the force proportional to the product of their masses and inversely proportional to the square of the distance between them.

Gravity is the weakest of the four fundamental interactions of physics, approximately 10^38 times weaker than the strong interaction, 10^36 times weaker than the electromagnetic force and 10^29 times weaker than the weak interaction.

For example, gravity causes the Earth and the other planets to orbit the Sun, it also causes the Moon to orbit the Earth, and causes the formation of tides, the formation and evolution of the Solar System, stars and galaxies.

Archimedes discovered the center of gravity of a triangle. The triangle, however, does not generalize on its own.

Cooking or cookery is the art, technology, science and craft of preparing food for consumption. Cooking techniques and ingredients vary widely across the world, from grilling food over an open fire to using electric stoves, to baking in various types of ovens, reflecting unique environmental, economic, and cultural traditions and trends. The ways or types of cooking also depend on the skill and type of training an individual cook has. Cooking is done both by people in their own dwellings and by professional cooks and chefs in restaurants and other food establishments. Cooking can also occur through chemical reactions without the presence of heat, such as in ceviche, a traditional South American dish where fish is cooked with the acids in lemon or lime juice.

VARNISH. Cjay Writing. Abacus, Ultimate Soul. Xuma, Time. PANTHERIZE. Gerardo Pneumatic. Jvlivs, Uakari Valeria.

The expansion of agriculture, commerce, trade, and transportation between civilizations in different regions offered cooks many new ingredients. New inventions and technologies, such as the invention of pottery for holding and boiling water, expanded cooking techniques. Some modern cooks apply advanced scientific techniques to food preparation to further enhance the flavor of the dish served.

Source Serif 4 Regular, rival kf algorithm

Printing is a process for reproducing text and images using a master form or template. The earliest non-paper products involving printing include cylinder seals and objects such as the Cyrus Cylinder and the Cylinders of Nabonidus. The earliest known form of printing as applied to paper was woodblock printing, which appeared in China before 220 AD. Later developments in printing technology include the movable type invented by Bi Sheng around 1040 AD and the printing press invented by Johannes Gutenberg in the 15th century. The technology of printing played a key role in the development of the Renaissance and the scientific revolution, and laid the material basis for the modern knowledge-based economy and the spread of learning to the masses.

Today, Tomorrow, Refinery. VALUES. Minimum, Portent, Tyres. Czar. Most Ushers Ionize. HIJINKS. Food Bookkeeping Last.

Gutenberg was a German blacksmith, goldsmith, inventor, printer, and publisher who introduced printing to Europe with the printing press. His introduction of mechanical movable type printing to Europe started the Printing Revolution and is regarded as a milestone of the second millennium, ushering in the modern period of human history.

Kent Gotham's Voroni Diagram. Radial Robots. West SOHO. Zones Accept Fast Youths. OWLS. Cooling Xanax Even In-flight.

Type metal (sometimes called hot metal) refers to the metal alloys used in traditional typefounding and hot metal typesetting. Historically, type metal was an alloy of lead, tin and antimony in different proportions depending on the application, be it individual character mechanical casting for hand setting, mechanical line casting or individual character mechanical typesetting and stereo plate casting. The proportions used are in the range: lead 50–86%, antimony 11–30% and tin 3–20%. Antimony and tin are added to lead for durability while reducing the difference between the coefficients of expansion of the matrix and the alloy. Apart from durability, the general requirements for type-metal are that it should produce a true and sharp cast, and retain correct dimensions and form after cooling down.

Gravity (from Latin gravitas, meaning 'weight'), or gravitation, is a natural phenomenon by which all things with mass or energy —including planets, stars, galaxies, and even light—are brought toward (or gravitate toward) one another. On Earth, gravity gives weight to physical objects, and the Moon's gravity causes the ocean tides. The gravitational attraction of the original gaseous matter present in the Universe caused it to begin coalescing, forming stars—and for the stars to group together into galaxies—so gravity is responsible for many of the large-scale structures in the Universe. Gravity has an infinite range, although its effects become increasingly weaker on farther objects.

Gravity is most accurately described by the general theory of relativity (proposed by Albert Einstein in 1915) which describes gravity not as a force, but as a consequence of the curvature of spacetime caused by the uneven distribution of mass.

Aviator, Xgood, Nuno, Bhangras, Xone, Odor, Snot, Quando, Vnom, Kwanza, Izzy, Nimrod.

The most extreme example of this curvature of spacetime is a black hole, from which nothing—not even light—can escape once past the black hole's event horizon. However, for most applications, gravity is well approximated by Newton's law of universal gravitation, which describes gravity as a force which causes any two bodies to be attracted to each other, with the force proportional to the product of their masses and inversely proportional to the square of the distance between them.

Gravity is the weakest of the four fundamental interactions of physics, approximately 10^38 times weaker than the strong interaction, 10^36 times weaker than the electromagnetic force and 10^29 times weaker than the weak interaction.

For example, gravity causes the Earth and the other planets to orbit the Sun, it also causes the Moon to orbit the Earth, and causes the formation of tides, the formation and evolution of the Solar System, stars and galaxies.

Archimedes discovered the center of gravity of a triangle. The triangle, however, does not generalize on its own.

Cooking or cookery is the art, technology, science and craft of preparing food for consumption. Cooking techniques and ingredients vary widely across the world, from grilling food over an open fire to using electric stoves, to baking in various types of ovens, reflecting unique environmental, economic, and cultural traditions and trends. The ways or types of cooking also depend on the skill and type of training an individual cook has. Cooking is done both by people in their own dwellings and by professional cooks and chefs in restaurants and other food establishments. Cooking can also occur through chemical reactions without the presence of heat, such as in ceviche, a traditional South American dish where fish is cooked with the acids in lemon or lime juice.

VARNISH. Cjay Writing. Abacus, Ultimate Soul. Xuma, Time. PANTHERIZE. Gerardo Pneumatic. Jvlivs, Uakari Valeria.

The expansion of agriculture, commerce, trade, and transportation between civilizations in different regions offered cooks many new ingredients. New inventions and technologies, such as the invention of pottery for holding and boiling water, expanded cooking techniques. Some modern cooks apply advanced scientific techniques to food preparation to further enhance the flavor of the dish served.

STIX Two Text Regular, control (original fitting)

En géographie physique, un territoire désigne un espace à métrique topographique (métrique caractérisée par la continuité et la contiguïté). L'une des dimensions du territoire est le milieu physique. En raison de l'existence de différents phénomènes géomorphologiques en action sur la planète, le territoire physique est en perpétuel mouvement.

En géographie politique, le territoire est défini en se concentrant sur les rapports de pouvoir et leurs transcriptions dans l'espace. Une partie de la définition de Marie-Christine Jaillet peut servir à illustrer l'acception de la notion en géographie politique : « le territoire désigne à la fois une circonscription politique et l'espace de vie d'un groupe cherche à en maîtriser l'usage à son seul avantage ». Cette définition est utile pour la compréhension du terme en géographie politique mais il est important de relever que Marie-Christine Jaillet est critique par rapport à cette définition qui n'est, selon elle, plus d'actualité, notamment en ce qui concerne l'idée qu'à chaque groupe correspond un territoire.

Luge, Hein ? McDonald's Mhari, Tlumacz. VERSET. Tsareena Wuthering. EKÉ. Iuri Utero, Wario. Florence Xfight. BAS DE PAGE.

Riemann introduisit sa théorie de l'intégration, puis Karl Weierstrass sa définition des limites. Richard Dedekind construisit les nombres réels avec ses coupures. En même temps, on commença à étudier la « taille » des ensembles de réels. En outre, des « monstres mathématiques » commencèrent à être créés. Dans ce contexte, Camille Jordan développa sa théorie sur la mesure et Georg Cantor, ce qu'on appelle aujourd'hui la théorie naïve des ensembles.

L'analyse complexe est un domaine des mathématiques traitant des fonctions à valeurs complexes (ou, plus généralement, à valeurs dans un C-espace vectoriel) et qui sont dérivables par rapport à une ou plusieurs variables complexes.

Les fonctions dérivables sur un ouvert du plan complexe sont appelées holomorphes et satisfont de nombreuses propriétés plus fortes que celles vérifiées par les fonctions dérivables en analyse réelle. Entre autres, toute fonction holomorphe est analytique et vérifie le principe du maximum.

Clara Ogre, Dhan. Jrashesh Jjval, Lara. Filet De Souris Xhosa. Yaourt Ydalir. FIL Professeur. Jhana Urzé Totoro. Noix. SERPENTS SNACK.

Zéro est un chiffre et un nombre. Son nom a été emprunté en 1485 à l'italien zero, contraction de zefiro. En tant que chiffre, il est utilisé pour « garder le rang »3 et marquer une position vide dans l'écriture des nombres en notation positionnelle.

En tant que nombre, zéro est un objet mathématique permettant d'exprimer une absence comme une quantité nulle : c'est le nombre d'éléments de l'ensemble vide. Il est le plus petit des entiers positifs ou nuls. Ses propriétés arithmétiques particulières, notamment l'impossibilité de la division par zéro, impliquent parfois de traiter son cas à part. Il sépare les nombres réels en positifs et négatifs et tient lieu d'origine pour repérer des points sur la droite réelle.

Dans la base dix que l'on utilise, le chiffre le plus à droite indique les unités, le deuxième chiffre indique les dizaines, le troisième les centaines, le quatrième les milliers...

INJONCTION De Niveau Nul. Plus Tôt, Plutôt, Plus Tard. COULOIR À ÉVITER. Porches, Cygnes, Paroles, Santé Mentale. DESSINÉ. Cuivre, Roue De Charrette. Ivana, Zwingli.

Le zéro joue donc un rôle particulier dans le système arithmétique positionnel, quel qu'il soit du reste. Rappelons que l'usage de la base 10, en provenance de l'Inde, s'est imposé en France par rapport à d'autres bases, par exemple 12 et 60 qui étaient utilisées dans certaines civilisations, le système vicésimal ayant laissé des traces dans la langue française, et le système duodécimal des modes de calcul chez les Britanniques.

REVEAL, Onguent, Ghana, Westworld. Inonde. INONDER. Siècle, Boule De Foudre. Samosa, Actions Déclaratives, Estimation.

Le vitrail est une composition de verre formée de pièces de verre. Celles-ci peuvent être blanches ou colorées et peuvent recevoir un décor. Le mot vitrail désigne une technique, alors que la fermeture d'une baie fixe avec du verre s'appelle une verrière.

Un vitrail est appelé vitrerie lorsque son dessin est géométrique et répétitif (par exemple des losanges ou des bornes). La vitrerie est généralement claire et sans peinture.

Un vitrail et une vitrerie au plomb se réalisent suivant le même procédé : en associant le plomb et le verre. Le verre utilisé est plat, d'une épaisseur variant entre 1,5 mm et 5 mm, et le plomb se présente sous forme de baguettes, en forme de H couché. Les pièces de verre sont serties dans les plombs puis l'ensemble est maintenu définitivement grâce aux soudures réalisées à chaque intersection des plombs.

Cette étape détermine l'harmonie des différents verres colorés. On choisit les verres suivant les couleurs indiquées sur la maquette et en tenant compte de leur nature : antique, plaqué, imprimé... Vers 1655 Pierre Boulanger inventa de nombreuses formes différentes de vitraux. C'est le « dessin technique » du panneau, réalisé à l'échelle 1/1 sur du papier bulle. Le réseau des plombs est tracé avec précision en utilisant la maquette comme modèle. Il est ensuite calibré pour que l'on coupe les verres. Dans le cas d'un vitrail figuratif, on procède à un agrandissement de la maquette. Si l'on possède un carton, on peut décalquer le dessin des plombs sur le papier. Dans le cas d'une vitrerie, on utilise un compas à balustre pour reporter les écartements identiques.

Diagramme De Voronoi De Kent Gotham. Robots Radiaux. SOHO Ouest. Les Zones Acceptent Les Jeunes Rapides. CHOUETTES. Refroidir Xanax Même En Vol.

STIX Two Text Regular, composite algorithm

En géographie physique, un territoire désigne un espace à métrique topographique (métrique caractérisée par la continuité et la contiguïté). L'une des dimensions du territoire est le milieu physique. En raison de l'existence de différents phénomènes géomorphologiques en action sur la planète, le territoire physique est en perpétuel mouvement.

En géographie politique, le territoire est défini en se concentrant sur les rapports de pouvoir et leurs transcriptions dans l'espace. Une partie de la définition de Marie-Christine Jaillet peut servir à illustrer l'acception de la notion en géographie politique : « le territoire désigne à la fois une circonscription politique et l'espace de vie d'un groupe cherche à en maîtriser l'usage à son seul avantage ». Cette définition est utile pour la compréhension du terme en géographie politique mais il est important de relever que Marie-Christine Jaillet est critique par rapport à cette définition qui n'est, selon elle, plus d'actualité, notamment en ce qui concerne l'idée qu'à chaque groupe correspond un territoire.

Luge, Hein ? McDonald's Mhari, Tlumacz. VERSET. Tsareena Wuthering. EKÉ. Iuri Utero, Wario. Florence Xfight. BAS DE PAGE.

Riemann introduisit sa théorie de l'intégration, puis Karl Weierstrass sa définition des limites. Richard Dedekind construisit les nombres réels avec ses coupures. En même temps, on commença à étudier la « taille » des ensembles de réels. En outre, des « monstres mathématiques » commencèrent à être créés. Dans ce contexte, Camille Jordan développa sa théorie sur la mesure et Georg Cantor, ce qu'on appelle aujourd'hui la théorie naïve des ensembles.

L'analyse complexe est un domaine des mathématiques traitant des fonctions à valeurs complexes (ou, plus généralement, à valeurs dans un C-espace vectoriel) et qui sont dérivables par rapport à une ou plusieurs variables complexes.

Les fonctions dérivables sur un ouvert du plan complexe sont appelées holomorphes et satisfont de nombreuses propriétés plus fortes que celles vérifiées par les fonctions dérivables en analyse réelle. Entre autres, toute fonction holomorphe est analytique et vérifie le principe du maximum.

Clara Ogre, Dhan. Jrashesh Jjval, Lara. Filet De Souris Xhosa. Yaourt Ydalir. FIL Professeur. Jhana Urzé Totoro. Noix. SERPENIS SNACK.

Zéro est un chiffre et un nombre. Son nom a été emprunté en 1485 à l'italien zero, contraction de zefiro. En tant que chiffre, il est utilisé pour « garder le rang »3 et marquer une position vide dans l'écriture des nombres en notation positionnelle.

En tant que nombre, zéro est un objet mathématique permettant d'exprimer une absence comme une quantité nulle : c'est le nombre d'éléments de l'ensemble vide. Il est le plus petit des entiers positifs ou nuls. Ses propriétés arithmétiques particulières, notamment l'impossibilité de la division par zéro, impliquent parfois de traiter son cas à part. Il sépare les nombres réels en positifs et négatifs et tient lieu d'origine pour repérer des points sur la droite réelle.

Dans la base dix que l'on utilise, le chiffre le plus à droite indique les unités, le deuxième chiffre indique les dizaines, le troisième les centaines, le quatrième les milliers...

INJONCTION De Niveau Nul. Plus Tôt, Plutôt, Plus Tard. COULOIR À ÉVIER. Porches, Cygnes, Paroles, Santé Mentale DESSINÉ. Cuivre, Roue De Charrette. Ivana, Zwingli.

Le zéro joue donc un rôle particulier dans le système arithmétique positionnel, quel qu'il soit du reste. Rappelons que l'usage de la base 10, en provenance de l'Inde, s'est imposé en France par rapport à d'autres bases, par exemple 1: et 60 qui étaient utilisées dans certaines civilisations, le système vicésimal ayant laissé des traces dans la langue française, et le système duodécimal des modes de calcul chez les Britanniques.

REVEAL, Onguent, Ghana, Westworld. Inonde. INONDER. Siècle, Boule De Foudre. Samosa, Actions Déclaratives, Estimation.

Le vitrail est une composition de verre formée de pièces de verre. Celles-ci peuvent être blanches ou colorées et peuvent recevoir un décor. Le mot vitrail désigne une technique, alors que la fermeture d'une baie fixe avec du verre s'appelle une verrière.

Un vitrail est appelé vitrerie lorsque son dessin est géométrique et répétitif (par exemple des losanges ou des bornes). La vitrerie est généralement claire et sans peinture.

Un vitrail et une vitrerie au plomb se réalisent suivant le même procédé : en associant le plomb et le verre. Le verre utilisé est plat, d'une épaisseur variant entre 1,5 mm et 5 mm, et le plomb se présente sous forme de baguettes, en forme de H couché. Les pièces de verre sont serties dans les plombs puis l'ensemble est maintenu définitivement grâce aux soudures réalisées à chaque intersection des plombs.

Cette étape détermine l'harmonie des différents verres colorés. On choisit les verres suivant les couleurs indiquées sur la maquette et en tenant compte de leur nature : antique, plaqué, imprimé... Vers 1655 Pierre Boulanger inventa de nombreuses formes différentes de vitraux. C'est le « dessin technique » du panneau, réalisé à l'échelle 1/1 sur du papier bulle. Le réseau des plombs est tracé avec précision en utilisant la maquette comme modèle. Il est ensuite calibré pour que l'on coupe les verres. Dans le cas d'un vitrail figuratif, on procède à un agrandissement de la maquette. Si l'on possède un carton, on peut décalquer le dessin des plombs sur le papier. Dans le cas d'une vitrerie, on utilise un compas à balustre pour reporter les écartements identiques.

Diagramme De Voronoi De Kent Gotham. Robots Radiaux. SOHO Ouest. Les Zones Acceptent Les Jeunes Rapides. CHOUETTES. Refroidir Xanax Même En Vol.

STIX Two Text Regular, rival kf algorithm

En géographie physique, un territoire désigne un espace à métrique topographique (métrique caractérisée par la continuité et la contiguïté). L'une des dimensions du territoire est le milieu physique. En raison de l'existence de différents phénomènes géomorphologiques en action sur la planète, le territoire physique est en perpétuel mouvement.

En géographie politique, le territoire est défini en se concentrant sur les rapports de pouvoir et leurs transcriptions dans l'espace. Une partie de la définition de Marie-Christine Jaillet peut servir à illustrer l'acception de la notion en géographie politique : « le territoire désigne à la fois une circonscription politique et l'espace de vie d'un groupe cherche à en maîtriser l'usage à son seul avantage ». Cette définition est utile pour la compréhension du terme en géographie politique mais il est important de relever que Marie-Christine Jaillet est critique par rapport à cette définition qui n'est, selon elle, plus d'actualité, notamment en ce qui concerne l'idée qu'à chaque groupe correspond un territoire.

Luge, Hein ? McDonald's Mhari, Tlumacz. VERSET. Tsareena Wuthering. EKÉ. Iuri Utero, Wario. Florence Xfight. BAS DE PAGE.

Riemann introduisit sa théorie de l'intégration, puis Karl Weierstrass sa définition des limites. Richard Dedekind construisit les nombres réels avec ses coupures. En même temps, on commença à étudier la « taille » des ensembles de réels. En outre, des « monstres mathématiques » commencèrent à être créés. Dans ce contexte, Camille Jordan développa sa théorie sur la mesure et Georg Cantor, ce qu'on appelle aujourd'hui la théorie naïve des ensembles.

L'analyse complexe est un domaine des mathématiques traitant des fonctions à valeurs complexes (ou, plus généralement, à valeurs dans un C-espace vectoriel) et qui sont dérivables par rapport à une ou plusieurs variables complexes.

Les fonctions dérivables sur un ouvert du plan complexe sont appelées holomorphes et satisfont de nombreuses propriétés plus fortes que celles vérifiées par les fonctions dérivables en analyse réelle. Entre autres, toute fonction holomorphe est analytique et vérifie le principe du maximum.

Clara Ogre, Dhan. Jrashesh Jjval, Lara. Filet De Souris Xhosa. Yaourt Ydalir. FIL Professeur. Jhana Urzé Totoro. Noix. SERPENTS SNACK.

Zéro est un chiffre et un nombre. Son nom a été emprunté en 1485 à l'italien zero, contraction de zefiro. En tant que chiffre, il est utilisé pour « garder le rang »3 et marquer une position vide dans l'écriture des nombres en notation positionnelle.

En tant que nombre, zéro est un objet mathématique permettant d'exprimer une absence comme une quantité nulle : c'est le nombre d'éléments de l'ensemble vide. Il est le plus petit des entiers positifs ou nuls. Ses propriétés arithmétiques particulières, notamment l'impossibilité de la division par zéro, impliquent parfois de traiter son cas à part. Il sépare les nombres réels en positifs et négatifs et tient lieu d'origine pour repérer des points sur la droite réelle.

Dans la base dix que l'on utilise, le chiffre le plus à droite indique les unités, le deuxième chiffre indique les dizaines, le troisième les centaines, le quatrième les milliers...

INJONCTION De Niveau Nul. Plus Tôt, Plutôt, Plus Tard. COULOIR À ÉVITER. Porches, Cygnes, Paroles, Santé Mentale. DESSINÉ. Cuivre, Roue De Charrette. Ivana, Zwingli.

Le zéro joue donc un rôle particulier dans le système arithmétique positionnel, quel qu'il soit du reste. Rappelons que l'usage de la base 10, en provenance de l'Inde, s'est imposé en France par rapport à d'autres bases, par exemple 12 et 60 qui étaient utilisées dans certaines civilisations, le système vicésimal ayant laissé des traces dans la langue française, et le système duodécimal des modes de calcul chez les Britanniques.

REVEAL, Onguent, Ghana, Westworld. Inonde. INONDER. Siècle, Boule De Foudre. Samosa, Actions Déclaratives, Estimation.

Le vitrail est une composition de verre formée de pièces de verre. Celles-ci peuvent être blanches ou colorées et peuvent recevoir un décor. Le mot vitrail désigne une technique, alors que la fermeture d'une baie fixe avec du verre s'appelle une verrière.

Un vitrail est appelé vitrerie lorsque son dessin est géométrique et répétitif (par exemple des losanges ou des bornes). La vitrerie est généralement claire et sans peinture.

Un vitrail et une vitrerie au plomb se réalisent suivant le même procédé : en associant le plomb et le verre. Le verre utilisé est plat, d'une épaisseur variant entre 1,5 mm et 5 mm, et le plomb se présente sous forme de baguettes, en forme de H couché. Les pièces de verre sont serties dans les plombs puis l'ensemble est maintenu définitivement grâce aux soudures réalisées à chaque intersection des plombs.

Cette étape détermine l'harmonie des différents verres colorés. On choisit les verres suivant les couleurs indiquées sur la maquette et en tenant compte de leur nature : antique, plaqué, imprimé... Vers 1655 Pierre Boulanger inventa de nombreuses formes différentes de vitraux. C'est le « dessin technique » du panneau, réalisé à l'échelle 1/1 sur du papier bulle. Le réseau des plombs est tracé avec précision en utilisant la maquette comme modèle. Il est ensuite calibré pour que l'on coupe les verres. Dans le cas d'un vitrail figuratif, on procède à un agrandissement de la maquette. Si l'on possède un carton, on peut décalquer le dessin des plombs sur le papier. Dans le cas d'une vitrerie, on utilise un compas à balustre pour reporter les écartements identiques.

Diagramme De Voronoi De Kent Gotham. Robots Radiaux. SOHO Ouest. Les Zones Acceptent Les Jeunes Rapides. CHOUETTES. Refroidir Xanax Même En Vol.

Yrsa Regular, control (original fitting)

Because seasons and astronomical events do not repeat in a whole number of days, calendars that have the same number of days in each year drift over time with respect to the event that the year is supposed to track. By inserting (also called intercalating) an additional day or month into the year, the drift can be corrected. A year that is not a leap year is called a common year. For example, in the Gregorian calendar, each leap year has 366 days instead of 365, by extending February to 29 days rather than the common 28. These extra days occur in years which are multiples of four (with the exception of centennial years not divisible by 400).

Absenteeism, Quone, Far Side, Nearly. LOCATION. Passive, Xerxes, Might, MALADY. Rope, Gasket, Yellowish.

Key historical trends of the High Middle Ages include the rapidly increasing population of Europe, which brought about great social and political change from the preceding era, and the Renaissance of the 12th century, including the first developments of rural exodus and of urbanization. By 1250, the robust population increase had greatly benefited the European economy, which reached levels that would not be seen again in some areas until the 19th century. That trend faltered during the Late Middle Ages because of a series of calamities, most notably the Black Death, but also numerous wars as well as economic stagnation.

FACTUAL, Isometric Parameter, Timpani, Udder, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Scholasticism is not so much a philosophy or a theology as a method of learning, as it places a strong emphasis on dialectical reasoning to extend knowledge by inference and to resolve contradictions. Scholastic thought is also known for rigorous conceptual analysis and the careful drawing of distinctions. In the classroom and in writing, it often takes the form of explicit disputation; a topic drawn from the tradition is broached in the form of a question, opponents' responses are given, a counterproposal is argued and opponents' arguments rebutted. Because of its emphasis on rigorous dialectical method, scholasticism was eventually applied to many other fields of study.

Invisibility is the state of an object that cannot be seen. An object in this state is said to be invisible (literally, "not visible"). The term is often used in fantasy/science fiction, where objects cannot be seen by magical or technological means; however, its effects can also be demonstrated in the real world, particularly in physics and perceptual psychology classes.

Coral Qhotels, Xena. Aeroporto Egg? RECRUITMENT PROCESSING. Khavia Ukraine, Iraq. Iwan Mwara, Uitlander. SPECTRUM Tv.

Since objects can be seen by light in the visible spectrum from a source reflecting off their surfaces and hitting the viewer's eye, the most natural form of invisibility (whether real or fictional) is an object that neither reflects nor absorbs light (that is, it allows light to pass through it). This is known as transparency, and is seen in many naturally occurring materials (although no naturally occurring material is 100% transparent).

Camouflage is the use of any combination of materials, coloration, or illumination for concealment, either by making animals or objects hard to see (crypsis), or by disguising them as something else (mimesis). Examples include the leopard's spotted coat, the battledress of a modern soldier, and the leaf-mimic katydid's wings.

Eolithic Ujena. Muriatic ACID. Esters Rtweet. Vhan Tzatziki Weston, Eduardo! DOUBLE Sharon. Uwatch, Blood.

In the open ocean, where there is no background, the principal methods of camouflage are transparency, silvering, and countershading, while the ability to produce light is among other things used for counter-illumination on the undersides of cephalopods such as squid.

Magic, which encompasses the subgenres of illusion, stage magic, and close up magic, is a performing art in which audiences are entertained by staged tricks or illusions of seemingly impossible feats using natural means. It is to be distinguished from paranormal magic which are effects claimed to be created through supernatural means. It is one of the oldest performing arts in the world.

Modern entertainment magic, as pioneered by 19th-century magician Jean Eugène Robert-Houdin, has become a popular theatrical art form. In the late 19th and early 20th centuries, magicians such as Maskelyne and Devant, Howard Thurston, Harry Kellar, and Harry Houdini achieved widespread commercial success during what has become known as "The Golden Age of Magic".

Qlikview Sport Artist Star. Sriracha Order Vincent. Dwindle Jerome Vulcanologist. RICHTER SCALE ALABASTER. Nha Yaris.

Yrsa Regular, composite algorithm

Because seasons and astronomical events do not repeat in a whole number of days, calendars that have the same number of days in each year drift over time with respect to the event that the year is supposed to track. By inserting (also called intercalating) an additional day or month into the year, the drift can be corrected. A year that is not a leap year is called a common year. For example, in the Gregorian calendar, each leap year has 366 days instead of 365, by extending February to 29 days rather than the common 28. These extra days occur in years which are multiples of four (with the exception of centennial years not divisible by 400).

Absenteeism, Quone, Far Side, Nearly. LOCATION. Passive, Xerxes, Might, MALADY. Rope, Gasket, Yellowish.

Key historical trends of the High Middle Ages include the rapidly increasing population of Europe, which brought about great social and political change from the preceding era, and the Renaissance of the 12th century, including the first developments of rural exodus and of urbanization. By 1250, the robust population increase had greatly benefited the European economy, which reached levels that would not be seen again in some areas until the 19th century. That trend faltered during the Late Middle Ages because of a series of calamities, most notably the Black Death, but also numerous wars as well as economic stagnation.

FACTUAL, Isometric Parameter, Timpani, Udder, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Scholasticism is not so much a philosophy or a theology as a method of learning, as it places a strong emphasis on dialectical reasoning to extend knowledge by inference and to resolve contradictions. Scholastic thought is also known for rigorous conceptual analysis and the careful drawing of distinctions. In the classroom and in writing, it often takes the form of explicit disputation; a topic drawn from the tradition is broached in the form of a question, opponents' responses are given, a counterproposal is argued and opponents' arguments rebutted. Because of its emphasis on rigorous dialectical method, scholasticism was eventually applied to many other fields of study.

Invisibility is the state of an object that cannot be seen. An object in this state is said to be invisible (literally, "not visible"). The term is often used in fantasy/science fiction, where objects cannot be seen by magical or technological means; however, its effects can also be demonstrated in the real world, particularly in physics and perceptual psychology classes.

Coral Qhotels, Xena. Aeroporto Egg? RECRUITMENT PROCESSING. Khavia Ukraine, Iraq. Iwan Mwara, Uitlander. SPECTRUM Tv.

Since objects can be seen by light in the visible spectrum from a source reflecting off their surfaces and hitting the viewer's eye, the most natural form of invisibility (whether real or fictional) is an object that neither reflects nor absorbs light (that is, it allows light to pass through it). This is known as transparency, and is seen in many naturally occurring materials (although no naturally occurring material is 100% transparent).

Camouflage is the use of any combination of materials, coloration, or illumination for concealment, either by making animals or objects hard to see (crypsis), or by disguising them as something else (mimesis). Examples include the leopard's spotted coat, the battledress of a modern soldier, and the leaf-mimic katydid's wings.

Eolithic Ujena. Muriatic ACID. Esters Rtweet. Vhan Tzatziki Weston, Eduardo! DOUBLE Sharon. Uwatch, Blood.

In the open ocean, where there is no background, the principal methods of camouflage are transparency, silvering, and countershading, while the ability to produce light is among other things used for counter-illumination on the undersides of cephalopods such as squid.

Magic, which encompasses the subgenres of illusion, stage magic, and close up magic, is a performing art in which audiences are entertained by staged tricks or illusions of seemingly impossible feats using natural means. It is to be distinguished from paranormal magic which are effects claimed to be created through supernatural means. It is one of the oldest performing arts in the world.

Modern entertainment magic, as pioneered by 19th-century magician Jean Eugène Robert-Houdin, has become a popular theatrical art form. In the late 19th and early 20th centuries, magicians such as Maskelyne and Devant, Howard Thurston, Harry Kellar, and Harry Houdini achieved widespread commercial success during what has become known as "The Golden Age of Magic".

Qlikview Sport Artist Star. Sriracha Order Vincent. Dwindle Jerome Vulcanologist. RICHTER SCALE ALABASTER. Nha Yaris.

Yrsa Regular, rival kf algorithm

Because seasons and astronomical events do not repeat in a whole number of days, calendars that have the same number of days in each year drift over time with respect to the event that the year is supposed to track. By inserting (also called intercalating) an additional day or month into the year, the drift can be corrected. A year that is not a leap year is called a common year. For example, in the Gregorian calendar, each leap year has 366 days instead of 365, by extending February to 29 days rather than the common 28. These extra days occur in years which are multiples of four (with the exception of centennial years not divisible by 400).

Absenteeism, Quone, Far Side, Nearly. LOCATION. Passive, Xerxes, Might, MALADY. Rope, Gasket, Yellowish.

Key historical trends of the High Middle Ages include the rapidly increasing population of Europe, which brought about great social and political change from the preceding era, and the Renaissance of the 12th century, including the first developments of rural exodus and of urbanization. By 1250, the robust population increase had greatly benefited the European economy, which reached levels that would not be seen again in some areas until the 19th century. That trend faltered during the Late Middle Ages because of a series of calamities, most notably the Black Death, but also numerous wars as well as economic stagnation.

FACTUAL, Isometric Parameter, Timpani, Udder, Zeno, Hub, Yards, WAITER, Pollen Estate, Quine, QUOTE, NINES, Liquid.

Scholasticism is not so much a philosophy or a theology as a method of learning, as it places a strong emphasis on dialectical reasoning to extend knowledge by inference and to resolve contradictions. Scholastic thought is also known for rigorous conceptual analysis and the careful drawing of distinctions. In the classroom and in writing, it often takes the form of explicit disputation; a topic drawn from the tradition is broached in the form of a question, opponents' responses are given, a counterproposal is argued and opponents' arguments rebutted. Because of its emphasis on rigorous dialectical method, scholasticism was eventually applied to many other fields of study.

Invisibility is the state of an object that cannot be seen. An object in this state is said to be invisible (literally, "not visible"). The term is often used in fantasy/science fiction, where objects cannot be seen by magical or technological means; however, its effects can also be demonstrated in the real world, particularly in physics and perceptual psychology classes.

Coral Qhotels, Xena. Aeroporto Egg? RECRUITMENT PROCESSING. Khavia Ukraine, Iraq. Iwan Mwara, Uitlander. SPECTRUM Tv.

Since objects can be seen by light in the visible spectrum from a source reflecting off their surfaces and hitting the viewer's eye, the most natural form of invisibility (whether real or fictional) is an object that neither reflects nor absorbs light (that is, it allows light to pass through it). This is known as transparency, and is seen in many naturally occurring materials (although no naturally occurring material is 100% transparent).

Camouflage is the use of any combination of materials, coloration, or illumination for concealment, either by making animals or objects hard to see (crypsis), or by disguising them as something else (mimesis). Examples include the leopard's spotted coat, the battledress of a modern soldier, and the leaf-mimic katydid's wings.

Eolithic Ujena. Muriatic ACID. Esters Rtweet. Vhan Tzatziki Weston, Eduardo! DOUBLE Sharon. Uwatch, Blood.

In the open ocean, where there is no background, the principal methods of camouflage are transparency, silvering, and countershading, while the ability to produce light is among other things used for counter-illumination on the undersides of cephalopods such as squid.

Magic, which encompasses the subgenres of illusion, stage magic, and close up magic, is a performing art in which audiences are entertained by staged tricks or illusions of seemingly impossible feats using natural means. It is to be distinguished from paranormal magic which are effects claimed to be created through supernatural means. It is one of the oldest performing arts in the world.

Modern entertainment magic, as pioneered by 19th-century magician Jean Eugène Robert-Houdin, has become a popular theatrical art form. In the late 19th and early 20th centuries, magicians such as Maskelyne and Devant, Howard Thurston, Harry Kellar, and Harry Houdini achieved widespread commercial success during what has become known as "The Golden Age of Magic".

Qlikview Sport Artist Star. Sriracha Order Vincent. Dwindle Jerome Vulcanologist. RICHTER SCALE ALABASTER. Nha Yaris.

Yrsa Bold, control (original fitting)

Eine Camera obscura (lat. camera „Kammer“; obscura „dunkel“) ist ein dunkler Raum mit einem Loch in der Wand, die als Metapher für die menschliche Wahrnehmung und für die Herstellung von Bildern verwendet wird. Hat der dunkle Raum die Größe einer Schachtel, spricht man auch von einer Lochkamera.

Während die technischen Prinzipien der Lochkamera bereits in der Antike bekannt waren, wurde die Nutzung des technischen Konzepts zur Herstellung von Bildern mit einer linearen Perspektive in Gemälden, Zeichnungen, Karten, architektonischen Umsetzungen und später auch Fotografien erst in der (bzw., vgl. Erwin Panofsky, den) Renaissance(n) der europäischen Kunst und der wissenschaftlichen Revolution der Neuzeit angewendet. Unter anderem nutzte Leonardo da Vinci die Camera obscura als Ebenbild des Auges, René Descartes für das Zusammenspiel von Auge und Bewusstsein und John Locke begann das Prinzip als Metapher des menschlichen Bewusstseins an sich zu benutzen.

Qlikview Sport Artist Star. Sriracha-Orden Vincent. Dwindle Jerome Vulkanologe. RICHTER SKALA ALABASTER. Nha Yaris.

Fällt Licht durch ein kleines Loch in einen ansonsten lichtdichten Hohlkörper, so wird in ihm ein seitenverkehrtes und auf dem Kopf stehendes Bild, eine Projektion des Außenraumes erzeugt. Die Schemazeichnung rechts oben zeigt exemplarisch zwei Strahlenbündel, die von zwei Punkten eines Gegenstands in das Loch eintreten. Der kleine Durchmesser der Blende beschränkt die Bündel auf einen kleinen Öffnungswinkel und verhindert die vollständige Überlappung der Lichtstrahlen. Strahlen vom oberen Bereich eines Gegenstands fallen auf den unteren Rand der Projektionsfläche, Strahlen vom unteren Bereich werden nach oben weitergeleitet.

Clara Oger, Dhan. Jrashesh Jjval, Lara. NETTING MÄUSE Xhosa. Joghurt Ydalir. THREAD Lehrer. Jhana Urze Totoro. Nüsse. SNACK SCHLANGEN.

Im Jahre 1686 konstruierte Johann Zahn eine transportable Camera obscura. Ein Spiegel, der im Winkel von 45 Grad zur optischen Achse der Linse im Inneren der Kamera angebracht war, reflektierte das Bild nach oben auf eine Mattscheibe, die beim Transport durch einen aufklappbaren Deckel geschützt werden konnte.

Als Seidenstraße bezeichnet man ein altes Netz von Karawanenstraßen, dessen Hauptroute den Mittelmeerraum auf dem Landweg über Zentralasien mit Ostasien verband. Die Bezeichnung geht auf den deutschen Geografen Ferdinand von Richthofen zurück, der den Begriff erstmals 1877 verwendete.

Auf der antiken Seidenstraße wurde in westliche Richtung hauptsächlich Seide, gen Osten vor allem Wolle, Gold und Silber gehandelt (siehe auch Indienhandel). Seide war für den Westen wohl das außergewöhnlichste Handelsgut, das die Seidenstraße passierte.

Zoologische Pauke Flexbox, TOPCOAT. KRANKE ERKUNDUNG, GESCHICKTER KAVALIER. Runderneuert, SSL, Hushed Bobcat Petersburg, Klaxons.

Ein großer Anteil der Sterne ist im Frühstadium des Universums vor über 10 Milliarden Jahren entstanden. Aber auch heute bilden sich noch Sterne. Massereiche Sterne entstehen seltener als massearme. Dies wird beschrieben durch die Ursprüngliche Massenfunktion. Aus einer Globule kann sowohl ein Doppel- oder Mehrfachsternsystem als auch ein einzelner Stern entstehen.

Der weitere Verlauf der Sternentwicklung wird im Wesentlichen durch die Masse bestimmt. Je größer die Masse eines Sternes ist, desto kürzer ist seine Brenndauer. Die massereichsten Sterne verbrauchen in nur wenigen hunderttausend Jahren ihren gesamten Brennstoff. Ihre Strahlungsleistung übertrifft dabei die der Sonne um das Hunderttausendfache oder mehr.

Anden Usted Pedro; Ovo! Szondi Owen, Laufband! RAUBIGE HOLUNDERBEERE. Swat Vjeran Phönix. Eventuell 100 Km, Ungarn.

Als Weltstadt werden Städte von überragender weltweiter Wichtigkeit bezeichnet. Eine Weltstadt besitzt zentrale Bedeutung auf politischem, wirtschaftlichem oder kulturellem Gebiet. Im Gegensatz zum teilweise synonym angewandten Begriff Metropole.

Ein Quad oder ATV („Geländefahrzeug“) ist ein kleines Kraftfahrzeug für ein bis drei Personen mit vier Rädern oder seltener mit vier Gleisketten, häufig mit dicken Ballonreifen als Geländefahrzeug. In Deutschland werden Sport- und Freizeitfahrzeuge häufig als Quad bezeichnet, Arbeitsfahrzeuge für den Geländeeinsatz, beispielsweise Bergrettungsfahrzeuge mit Allradantrieb, eher als ATV. In Kanada oder den USA gibt es diese Unterscheidung nicht.

Xjet Rückgängig Machen Octavio Pilar. DNS-SUCHE. Leutnant Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. GLÄTTE Tv.

Silber (in der Pharmazie auch lateinisch Argentum) ist ein chemisches Element mit dem Elementsymbol Ag und der Ordnungszahl 47. Es zählt zu den Übergangsmetallen. Im Periodensystem steht es in der 5. Periode und der 1. Nebengruppe (Gruppe 11) oder Kupfergruppe. Das Elementsymbol Ag leitet sich vom lateinischen Wort argentum für „Silber“ ab. Silber gehört zu den Edelmetallen.

Es ist ein weiches, gut verformbares (duktiles) Schwermetall mit der höchsten elektrischen Leitfähigkeit aller Elemente im unmodifizierten Zustand (Kohlenstoff in der Form von Graphen besitzt nochmals eine höhere Leitfähigkeit) und der höchsten thermischen Leitfähigkeit aller Metalle.

Eolithisches Ujena. Salzsäure. Ester Rtweet. Vhan Tzatziki Weston, Eduardo! DOPPELTE Sharon. Uwatch, Blut.

Ein Passepartout bezeichnet in der Kunst eine Papier- oder Kartonumrahmung für Grafiken, Fotos und Gemälde. Passepartouts werden einerseits verwendet, um die Betrachtung auf das Kunstwerk zu richten, indem vom Bilderrahmen abgelenkt wird, andererseits gelingt es dadurch, ein Kunstwerk in einen Rahmen einzupassen, dessen Maße die Abmessungen des Kunstwerks übertreffen.

Yrsa Bold, composite algorithm

Eine Camera obscura (lat. camera „Kammer“; obscura „dunkel“) ist ein dunkler Raum mit einem Loch in der Wand, die als Metapher für die menschliche Wahrnehmung und für die Herstellung von Bildern verwendet wird. Hat der dunkle Raum die Größe einer Schachtel, spricht man auch von einer Lochkamera.

Während die technischen Prinzipien der Lochkamera bereits in der Antike bekannt waren, wurde die Nutzung des technischen Konzepts zur Herstellung von Bildern mit einer linearen Perspektive in Gemälden, Zeichnungen, Karten, architektonischen Umsetzungen und später auch Fotografien erst in der (bzw., vgl. Erwin Panofsky, den) Renaissance(n) der europäischen Kunst und der wissenschaftlichen Revolution der Neuzeit angewendet. Unter anderem nutzte Leonardo da Vinci die Camera obscura als Ebenbild des Auges, René Descartes für das Zusammenspiel von Auge und Bewusstsein und John Locke begann das Prinzip als Metapher des menschlichen Bewusstseins an sich zu benutzen.

Qlikview Sport Artist Star. Sriracha-Orden Vincent. Dwindle Jerome Vulkanologe. RICHIER SKALA ALABASIER. Nha Yaris.

Fällt Licht durch ein kleines Loch in einen ansonsten lichtdichten Hohlkörper, so wird in ihm ein seitenverkehrtes und auf dem Kopf stehendes Bild, eine Projektion des Außenraumes erzeugt. Die Schemazeichnung rechts oben zeigt exemplarisch zwei Strahlenbündel, die von zwei Punkten eines Gegenstands in das Loch eintreten. Der kleine Durchmesser der Blende beschränkt die Bündel auf einen kleinen Öffnungswinkel und verhindert die vollständige Überlappung der Lichtstrahlen. Strahlen vom oberen Bereich eines Gegenstands fallen auf den unteren Rand der Projektionsfläche, Strahlen vom unteren Bereich werden nach oben weitergeleitet.

Clara Oger, Dhan. Jrashesh Jjval, Lara. NETTING MÄUSE Xhosa. Joghurt Ydalir. THREAD Lehrer. Jhana Urze Totoro. Nüsse. SNACK SCHLANGEN.

Im Jahre 1686 konstruierte Johann Zahn eine transportable Camera obscura. Ein Spiegel, der im Winkel von 45 Grad zur optischen Achse der Linse im Inneren der Kamera angebracht war, reflektierte das Bild nach oben auf eine Mattscheibe, die beim Transport durch einen aufklappbaren Deckel geschützt werden konnte.

Als Seidenstraße bezeichnet man ein altes Netz von Karawanenstraßen, dessen Hauptroute den Mittelmeerraum auf dem Landweg über Zentralasien mit Ostasien verband. Die Bezeichnung geht auf den deutschen Geografen Ferdinand von Richthofen zurück, der den Begriff erstmals 1877 verwendete.

Auf der antiken Seidenstraße wurde in westliche Richtung hauptsächlich Seide, gen Osten vor allem Wolle, Gold und Silber gehandelt (siehe auch Indienhandel). Seide war für den Westen wohl das außergewöhnlichste Handelsgut, das die Seidenstraße passierte.

Zoologische Pauke Flexbox, TOPCOAT. KRANKE ERKUNDUNG, GESCHICKTER KAVALIER. Runderneuert, SSL, Hushed Bobcat Petersburg, Klaxons.

Ein großer Anteil der Sterne ist im Frühstadium des Universums vor über 10 Milliarden Jahren entstanden. Aber auch heute bilden sich noch Sterne. Massereiche Sterne entstehen seltener als massearme. Dies wird beschrieben durch die Ursprüngliche Massenfunktion. Aus einer Globule kann sowohl ein Doppel- oder Mehrfachsternsystem als auch ein einzelner Stern entstehen.

Der weitere Verlauf der Sternentwicklung wird im Wesentlichen durch die Masse bestimmt. Je größer die Masse eines Sternes ist, desto kürzer ist seine Brenndauer. Die massereichsten Sterne verbrauchen in nur wenigen hunderttausend Jahren ihren gesamten Brennstoff. Ihre Strahlungsleistung übertrifft dabei die der Sonne um das Hunderttausendfache oder mehr.

Anden Usted Pedro; Ovo! Szondi Owen, Laufband! RAUBIGE HOLUNDERBEERE. Swat Vjeran Phönix. Eventuell 100 Km, Ungarn.

Als Weltstadt werden Städte von überragender weltweiter Wichtigkeit bezeichnet. Eine Weltstadt besitzt zentrale Bedeutung auf politischem, wirtschaftlichem oder kulturellem Gebiet. Im Gegensatz zum teilweise synonym angewandten Begriff Metropole.

Ein Quad oder ATV („Geländefahrzeug“) ist ein kleines Kraftfahrzeug für ein bis drei Personen mit vier Rädern oder seltener mit vier Gleisketten, häufig mit dicken Ballonreifen als Geländefahrzeug. In Deutschland werden Sport- und Freizeitfahrzeuge häufig als Quad bezeichnet, Arbeitsfahrzeuge für den Geländeeinsatz, beispielsweise Bergrettungsfahrzeuge mit Allradantrieb, eher als ATV. In Kanada oder den USA gibt es diese Unterscheidung nicht.

Xjet Rückgängig Machen Octavio Pilar. DNS-SUCHE. Leutnant Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. GLÄTTE Tv.

Silber (in der Pharmazie auch lateinisch Argentum) ist ein chemisches Element mit dem Elementsymbol Ag und der Ordnungszahl 47. Es zählt zu den Übergangsmetallen. Im Periodensystem steht es in der 5. Periode und der 1. Nebengruppe (Gruppe 11) oder Kupfergruppe. Das Elementsymbol Ag leitet sich vom lateinischen Wort argentum für „Silber“ ab. Silber gehört zu den Edelmetallen.

Es ist ein weiches, gut verformbares (duktiles) Schwermetall mit der höchsten elektrischen Leitfähigkeit aller Elemente im unmodifizierten Zustand (Kohlenstoff in der Form von Graphen besitzt nochmals eine höhere Leitfähigkeit) und der höchsten thermischen Leitfähigkeit aller Metalle.

Eolithisches Ujena. Salzsäure. Ester Rtweet. Vhan Tzatziki Weston, Eduardo! DOPPELTE Sharon. Uwatch, Blut.

Ein Passepartout bezeichnet in der Kunst eine Papier- oder Kartonumrahmung für Grafiken, Fotos und Gemälde. Passepartouts werden einerseits verwendet, um die Betrachtung auf das Kunstwerk zu richten, indem vom Bilderrahmen abgelenkt wird, andererseits gelingt es dadurch, ein Kunstwerk in einen Rahmen einzupassen, dessen Maße die Abmessungen des Kunstwerks übertreffen.

Yrsa Bold, rival kf algorithm

Eine Camera obscura (lat. camera „Kammer“; obscura „dunkel“) ist ein dunkler Raum mit einem Loch in der Wand, die als Metapher für die menschliche Wahrnehmung und für die Herstellung von Bildern verwendet wird. Hat der dunkle Raum die Größe einer Schachtel, spricht man auch von einer Lochkamera.

Während die technischen Prinzipien der Lochkamera bereits in der Antike bekannt waren, wurde die Nutzung des technischen Konzepts zur Herstellung von Bildern mit einer linearen Perspektive in Gemälden, Zeichnungen, Karten, architektonischen Umsetzungen und später auch Fotografien erst in der (bzw., vgl. Erwin Panofsky, den) Renaissance(n) der europäischen Kunst und der wissenschaftlichen Revolution der Neuzeit angewendet. Unter anderem nutzte Leonardo da Vinci die Camera obscura als Ebenbild des Auges, René Descartes für das Zusammenspiel von Auge und Bewusstsein und John Locke begann das Prinzip als Metapher des menschlichen Bewusstseins an sich zu benutzen.

Qlikview Sport Artist Star. Sriracha-Orden Vincent. Dwindle Jerome Vulkanologe. RICHTER SKALA ALABASTER. Nha Yaris.

Fällt Licht durch ein kleines Loch in einen ansonsten lichtdichten Hohlkörper, so wird in ihm ein seitenverkehrtes und auf dem Kopf stehendes Bild, eine Projektion des Außenraumes erzeugt. Die Schemazeichnung rechts oben zeigt exemplarisch zwei Strahlenbündel, die von zwei Punkten eines Gegenstands in das Loch eintreten. Der kleine Durchmesser der Blende beschränkt die Bündel auf einen kleinen Öffnungswinkel und verhindert die vollständige Überlappung der Lichtstrahlen. Strahlen vom oberen Bereich eines Gegenstands fallen auf den unteren Rand der Projektionsfläche, Strahlen vom unteren Bereich werden nach oben weitergeleitet.

Clara Oger, Dhan. Jrashesh Jjval, Lara. NETTING MÄUSE Xhosa. Joghurt Ydalir. THREAD Lehrer. Jhana Urze Totoro. Nüsse. SNACK SCHLANGEN.

Im Jahre 1686 konstruierte Johann Zahn eine transportable Camera obscura. Ein Spiegel, der im Winkel von 45 Grad zur optischen Achse der Linse im Inneren der Kamera angebracht war, reflektierte das Bild nach oben auf eine Mattscheibe, die beim Transport durch einen aufklappbaren Deckel geschützt werden konnte.

Als Seidenstraße bezeichnet man ein altes Netz von Karawanenstraßen, dessen Hauptroute den Mittelmeerraum auf dem Landweg über Zentralasien mit Ostasien verband. Die Bezeichnung geht auf den deutschen Geografen Ferdinand von Richthofen zurück, der den Begriff erstmals 1877 verwendete.

Auf der antiken Seidenstraße wurde in westliche Richtung hauptsächlich Seide, gen Osten vor allem Wolle, Gold und Silber gehandelt (siehe auch Indienhandel). Seide war für den Westen wohl das außergewöhnlichste Handelsgut, das die Seidenstraße passierte.

Zoologische Pauke Flexbox, TOPCOAT. KRANKE ERKUNDUNG, GESCHICKTER KAVALIER. Runderneuert, SSL, Hushed Bobcat Petersburg, Klaxons.

Ein großer Anteil der Sterne ist im Frühstadium des Universums vor über 10 Milliarden Jahren entstanden. Aber auch heute bilden sich noch Sterne. Massereiche Sterne entstehen seltener als massearme. Dies wird beschrieben durch die Ursprüngliche Massenfunktion. Aus einer Globule kann sowohl ein Doppel- oder Mehrfachsternsystem als auch ein einzelner Stern entstehen.

Der weitere Verlauf der Sternentwicklung wird im Wesentlichen durch die Masse bestimmt. Je größer die Masse eines Sternes ist, desto kürzer ist seine Brenndauer. Die massereichsten Sterne verbrauchen in nur wenigen hunderttausend Jahren ihren gesamten Brennstoff. Ihre Strahlungsleistung übertrifft dabei die der Sonne um das Hunderttausendfache oder mehr.

Anden Usted Pedro; Ovo! Szondi Owen, Laufband! RAUBIGE HOLUNDERBEERE. Swat Vjeran Phönix. Eventuell 100 Km, Ungarn.

Als Weltstadt werden Städte von überragender weltweiter Wichtigkeit bezeichnet. Eine Weltstadt besitzt zentrale Bedeutung auf politischem, wirtschaftlichem oder kulturellem Gebiet. Im Gegensatz zum teilweise synonym angewandten Begriff Metropole.

Ein Quad oder ATV („Geländefahrzeug“) ist ein kleines Kraftfahrzeug für ein bis drei Personen mit vier Rädern oder seltener mit vier Gleisketten, häufig mit dicken Ballonreifen als Geländefahrzeug. In Deutschland werden Sport- und Freizeitfahrzeuge häufig als Quad bezeichnet, Arbeitsfahrzeuge für den Geländeeinsatz, beispielsweise Bergrettungsfahrzeuge mit Allradantrieb, eher als ATV. In Kanada oder den USA gibt es diese Unterscheidung nicht.

Xjet Rückgängig Machen Octavio Pilar. DNS-SUCHE. Leutnant Horace, Rato. Tteokbokki Fionan. Xdream Ewan Tnt. GLÄTTE Tv.

Silber (in der Pharmazie auch lateinisch Argentum) ist ein chemisches Element mit dem Elementsymbol Ag und der Ordnungszahl 47. Es zählt zu den Übergangsmetallen. Im Periodensystem steht es in der 5. Periode und der 1. Nebengruppe (Gruppe 11) oder Kupfergruppe. Das Elementsymbol Ag leitet sich vom lateinischen Wort argentum für „Silber“ ab. Silber gehört zu den Edelmetallen.

Es ist ein weiches, gut verformbares (duktiles) Schwermetall mit der höchsten elektrischen Leitfähigkeit aller Elemente im unmodifizierten Zustand (Kohlenstoff in der Form von Graphen besitzt nochmals eine höhere Leitfähigkeit) und der höchsten thermischen Leitfähigkeit aller Metalle.

Eolithisches Ujena. Salzsäure. Ester Rtweet. Vhan Tzatziki Weston, Eduardo! DOPPELTE Sharon. Uwatch, Blut.

Ein Passepartout bezeichnet in der Kunst eine Papier- oder Kartonumrahmung für Grafiken, Fotos und Gemälde. Passepartouts werden einerseits verwendet, um die Betrachtung auf das Kunstwerk zu richten, indem vom Bilderrahmen abgelenkt wird, andererseits gelingt es dadurch, ein Kunstwerk in einen Rahmen einzupassen, dessen Maße die Abmessungen des Kunstwerks übertreffen.

ii. Sidebearing values

The following tables provide the left and right sidebearing values for each of the fonts in the public tests, in each of the test conditions used.

Alegreya Sans Regular

	LSB		*typeform*		*RSB*	
kf	**composite**	**control**		**control**	**composite**	**kf**
3	10	14	A	22	10	11
3	10	14	Acircumflex	22	10	11
3	10	14	Agrave	22	10	11
83	55	93	B	39	12	40
54	26	38	C	23	10	-176
83	55	93	D	38	19	48
84	56	93	E	23	-23	-103
84	56	93	Eacute	23	-23	-103
84	56	93	Egrave	23	-23	-103
84	56	93	F	-6	-40	-140
54	26	38	G	62	-17	-80
85	57	94	H	94	56	84
84	56	93	I	93	56	84
-43	-71	6	J	93	53	81
84	56	94	K	0	10	-72
84	56	90	L	3	10	-115
63	35	60	M	61	37	65
85	57	94	N	93	55	83
54	26	38	O	37	23	51
54	26	38	Ocircumflex	37	23	51
54	26	38	Odieresis	37	23	51
84	56	93	P	35	10	-87
54	26	38	Q	-94	-112	-84
83	55	93	R	35	10	12
-33	14	42	S	43	9	-2
-100	-128	22	T	21	-126	-98
68	39	74	U	74	36	64
68	39	74	Udieresis	74	36	64
-51	-71	15	V	7	-57	-63
-44	-48	21	W	8	-55	-60
-40	10	29	X	28	10	-40
-86	-80	15	Y	14	-89	-103
-34	-6	31	Z	36	-15	-88
12	14	42	a	30	-3	25
12	14	42	acircumflex	30	-3	25
12	14	42	adieresis	30	-3	25
12	14	42	agrave	30	-3	25
83	55	84	b	43	26	54
53	25	42	c	17	16	-52
53	25	42	ccedilla	17	16	-52
52	23	42	d	21	-3	25
52	23	42	e	44	15	0
52	23	42	eacute	44	15	0
52	23	42	ecircumflex	44	15	0
52	23	42	egrave	44	15	0
23	10	36	f	-47	-35	-67
50	22	55	g	2	-4	-37
83	55	84	h	72	50	78
78	50	75	i	76	51	79
78	50	81	dotlessi	83	51	79
78	50	-13	icircumflex	-12	51	79
23	50	23	idieresis	24	51	79
-1	-29	5	j	83	55	83
83	55	84	k	8	10	-9
83	55	84	l	83	55	83
84	56	80	m	72	50	78
84	56	80	n	72	50	78
54	26	42	o	44	26	54
54	26	42	ocircumflex	44	26	54
54	26	42	odieresis	44	26	54
26	26	37	oe	37	15	15
83	54	80	p	43	26	54
52	24	42	q	80	55	83
84	56	80	r	7	10	-36
19	19	32	s	42	23	22
-5	-5	36	germandbls	9	23	22
24	-4	25	t	13	12	-12
78	50	72	u	20	-1	27
78	50	72	ucircumflex	20	-1	27
78	50	72	udieresis	20	-1	27
78	50	72	ugrave	20	-1	27
17	10	20	v	8	10	4
24	10	20	w	8	10	13
15	10	23	x	18	10	10
-23	-16	-17	y	11	10	4
-4	16	33	z	33	13	-5

Fira Sans Condensed

	LSB		*typeform*		*RSB*	
kf	**composite**	**control**		**control**	**composite**	**kf**
4	9	7	A	7	9	4
4	9	7	Acircumflex	7	9	4
4	9	7	Agrave	7	9	4
80	52	85	B	38	9	37
50	22	50	C	15	9	0
80	52	85	D	50	17	45
80	52	85	E	39	9	0
80	52	85	Eacute	39	9	0
80	52	85	Egrave	39	9	0
80	52	85	F	19	-42	-41
51	24	50	G	55	8	-12
80	52	85	H	85	52	80
80	52	85	I	84	52	80
-17	-45	2	J	80	52	79
80	52	85	K	-3	9	-64
80	52	85	L	14	9	0
60	32	55	M	55	32	60
80	52	85	N	85	52	80
51	23	50	O	50	23	51
51	23	50	Ocircumflex	50	23	51
80	52	50	Odieresis	50	9	0
50	22	85	P	28	-4	20
80	52	50	Q	20	-21	6
-67	-3	85	R	19	-2	-16
-94	-122	22	S	38	-128	-100
68	40	11	T	5	40	68
68	40	75	U	74	40	68
-38	40	75	Udieresis	74	40	-37
5	-35	7	V	8	-35	5
0	-23	23	W	23	-23	0
-74	9	4	X	4	9	-74
-32	-66	4	Y	4	-66	-68
-5	-10	25	Z	35	-3	42
-5	4	40	a	44	14	42
-5	4	40	acircumflex	44	14	42
-5	4	40	adieresis	44	14	42
80	4	40	agrave	44	14	53
47	52	81	b	51	26	0
47	20	49	c	18	9	0
51	20	49	ccedilla	18	9	80
49	24	51	d	81	52	-1
49	21	49	e	44	5	-1
49	21	49	eacute	44	5	-1
49	21	49	ecircumflex	44	5	-1
13	21	49	egrave	44	5	-64
18	-15	13	f	-56	-55	-28
80	-10	20	g	3	1	75
66	38	81	h	77	38	66
66	38	67	i	67	38	66
66	38	81	dotlessi	81	38	66
66	38	-23	icircumflex	-23	38	66
-13	38	-21	idieresis	-19	38	66
80	-41	-12	j	67	38	-21
78	52	81	k	1	-2	10
80	50	76	l	17	-18	75
80	52	81	m	77	48	75
48	52	81	n	77	47	48
48	20	49	o	49	20	48
48	20	49	ocircumflex	49	20	48
49	20	49	odieresis	49	20	44
48	49	49	oe	44	44	-1
80	20	81	p	51	5	54
51	52	51	q	81	26	80
80	51	81	r	7	81	0
-14	52	23	s	34	9	3
81	0	81	germandbls	32	10	32
13	81	11	t	-3	32	3
11	-15	77	u	81	10	-36
75	-17	77	ucircumflex	81	-14	80
75	48	77	udieresis	81	52	80
75	48	77	ugrave	81	52	80
75	48	9	v	9	52	80
0	48	19	w	19	52	2
25	9	4	x	4	9	26
0	9	9	y	8	9	0
3	9	20	z	26	9	2

Literata Regular opsz 14

LSB			*typeform*	*RSB*		
kf	**composite**	**control**		**control**	**composite**	**kf**
-45	8	15	A	15	8	-38
13	10	62	B	43	32	34
50	48	50	C	54	8	-183
13	10	62	D	50	35	37
13	10	62	E	53	33	-4
13	10	62	F	43	-43	-90
52	50	50	G	33	-22	-63
13	10	62	H	62	11	14
9	7	62	I	62	7	9
-93	8	40	J	35	-17	-15
9	7	62	K	18	8	-131
13	10	62	L	60	8	-93
4	2	66	M	59	-3	-1
14	11	62	N	62	1	4
52	50	50	O	50	46	48
13	10	62	P	25	-99	-96
53	51	50	Q	-117	-115	-113
13	10	62	R	26	8	-59
3	48	56	S	58	30	6
-158	-44	42	T	42	-44	-157
-16	-18	49	U	46	-19	-17
-112	-107	21	V	21	-114	-118
-81	-83	21	W	21	-92	-89
-78	8	15	X	14	8	-87
-155	-143	21	Y	21	-150	-159
-50	17	58	Z	60	12	-34
37	56	52	a	5	-7	-5
5	2	17	b	46	58	60
54	51	46	c	29	34	-44
57	55	46	d	24	16	18
53	51	46	e	38	44	2
23	21	41	f	-66	-76	-105
39	37	25	g	1	15	-64
13	11	42	h	31	14	16
23	21	40	i	24	12	14
14	12	34	j	88	82	85
14	11	42	k	-8	8	-59
14	11	42	l	25	12	15
26	24	46	m	31	14	16
26	24	46	n	31	14	16
53	51	46	o	46	51	54
19	17	28	p	46	58	60
57	55	46	q	17	12	15
26	24	46	r	7	8	-45
42	58	49	s	30	58	46
19	16	37	t	5	30	-4
16	14	27	u	30	17	19
-38	8	12	v	11	8	-42
-26	8	7	w	7	8	-29
-21	8	7	x	3	8	-34
-48	4	10	y	2	8	-48
30	55	37	z	27	52	28
53	51	46	oe	38	44	2
23	20	41	germandbls	25	58	46
23	21	40	dotlessi	24	12	24
-45	8	15	Acircumflex	15	8	-38
-45	8	15	Agrave	15	8	-38
13	10	62	Eacute	53	33	-4
13	10	62	Egrave	53	33	-4
52	50	50	Ocircumflex	50	46	48
52	50	50	Odieresis	50	46	48
-16	-18	49	Udieresis	46	-19	-17
37	56	52	acircumflex	5	-7	-5
37	56	52	adieresis	5	-7	-5
37	56	52	agrave	5	-7	-5
54	51	46	ccedilla	29	34	-44
53	51	46	eacute	38	44	2
53	51	46	ecircumflex	38	44	2
53	51	46	egrave	38	44	2
23	21	-13	icircumflex	24	12	14
23	21	-16	idieresis	23	12	14
53	51	46	ocircumflex	46	51	54
53	51	46	odieresis	46	51	54
16	14	27	ucircumflex	30	17	19
16	14	27	udieresis	30	17	19
16	14	27	ugrave	30	17	19

Slabo 13px Regular

	LSB		typeform		RSB	
kf	composite	control		control	composite	kf
-36	6	-60	A	0	6	-34
20	18	0	B	80	35	37
36	34	25	C	80	6	-125
20	18	0	D	85	33	35
20	18	0	E	80	8	10
20	18	0	F	80	6	-97
34	32	25	G	60	-33	-32
20	18	0	H	60	18	20
20	18	0	I	60	18	20
-29	-31	-40	J	60	9	11
20	18	0	K	60	6	-81
20	18	0	L	60	6	-104
20	18	0	M	60	18	20
20	18	0	N	60	9	11
37	35	25	O	85	35	37
20	18	0	P	55	6	-106
37	35	25	Q	40	-11	-10
20	18	0	R	60	6	-18
9	43	-5	S	55	21	-3
-79	6	0	T	60	6	-79
-8	-9	0	U	60	-11	-9
-86	-74	-60	V	0	-75	-90
-85	-73	-60	W	0	-74	-87
-50	6	0	X	60	6	-51
-129	-98	-60	Y	0	-98	-129
-20	20	0	Z	80	28	-15
-36	6	-60	Agrave	0	6	-24
-36	6	-60	Acircumflex	0	6	-24
20	18	0	Egrave	80	8	60
20	18	0	Eacute	80	8	60
37	35	25	Ocircumflex	85	35	73
37	35	25	Odieresis	85	35	73
-8	-9	0	Udieresis	60	-11	68
2	26	-5	a	50	11	13
9	7	0	b	80	34	36
31	30	20	c	60	30	-53
36	34	20	d	60	18	20
35	33	20	e	60	30	-1
28	26	0	f	-60	-74	-86
35	34	20	g	120	67	68
20	18	0	h	50	12	13
28	26	0	i	60	18	20
0	-2	-20	j	110	59	61
20	18	0	k	60	12	-34
20	18	0	l	60	18	20
28	26	0	m	50	12	13
28	26	0	n	50	12	13
34	32	20	o	80	33	35
20	18	0	p	80	34	36
35	34	20	q	60	7	8
28	26	0	r	40	6	-34
22	38	30	s	80	36	20
15	13	0	t	50	36	-17
13	11	-10	u	60	18	20
-28	6	-40	v	40	6	-23
-28	6	-40	w	40	6	-23
-11	6	0	x	60	6	-14
-28	6	-40	y	40	6	-23
25	44	0	z	60	46	25
2	26	-5	agrave	50	11	43
2	26	-5	acircumflex	50	11	43
2	26	-5	adieresis	50	11	43
31	30	20	ccedilla	60	30	49
35	33	20	egrave	60	30	45
35	33	20	eacute	60	30	45
35	33	20	ecircumflex	60	30	45
0	26	0	dotlessi	60	18	60
-30	26	-30	icircumflex	30	18	30
-40	26	-40	idieresis	20	18	20
34	32	20	ocircumflex	80	33	66
34	32	20	odieresis	80	33	66
34	34	20	oe	60	30	30
27	27	1	germandbls	60	36	36
13	11	-10	ugrave	60	18	37
13	11	-10	ucircumflex	60	18	37
13	11	-10	udieresis	60	18	37

Slabo 27px Regular

kf	LSB composite	LSB control	typeform	RSB control	RSB composite	kf
0	12	0	A	0	12	0
9	10	30	B	40	28	27
33	34	40	C	30	12	0
9	10	30	D	47	26	25
9	10	30	E	30	14	0
9	10	30	F	10	-58	-70
32	33	40	G	30	12	0
9	10	30	H	30	10	9
9	10	30	I	30	10	9
-23	-22	6	J	30	2	1
9	10	30	K	0	12	0
9	10	30	L	20	12	0
7	8	20	M	20	7	6
9	10	30	N	30	2	1
36	37	47	O	47	36	35
9	10	30	P	37	12	0
35	37	47	Q	-60	-72	-73
9	10	30	R	0	12	0
0	34	30	S	30	20	0
-93	-91	17	T	17	-91	-93
-12	-11	20	U	20	-12	-13
-81	-70	0	V	0	-72	-83
-80	-68	0	W	0	-70	-82
0	12	0	X	0	12	0
-110	-99	0	Y	0	-99	-112
0	8	30	Z	40	13	0
0	12	0	Agrave	0	-12	0
0	12	0	Acircumflex	0	-12	0
9	10	30	Egrave	30	50	0
9	10	30	Eacute	30	50	0
36	37	47	Ocircumflex	47	57	35
36	29	47	Odieresis	47	14	35
-12	12	20	Udieresis	20	35	-13
10	29	40	a	10	14	34
33	12	25	b	40	35	0
34	34	40	c	40	30	17
34	35	40	d	20	19	0
24	35	35	e	30	28	-62
21	25	30	f	-52	-50	0
17	22	20	g	10	28	15
24	19	20	h	20	16	17
-3	25	30	i	20	19	52
17	-2	0	j	41	53	0
17	19	30	k	0	19	17
24	19	30	l	20	19	15
24	25	30	m	10	16	15
34	25	30	n	10	16	35
17	36	35	o	35	36	35
34	19	25	p	40	36	10
24	35	40	q	-10	11	0
16	25	30	r	10	12	14
15	37	35	s	30	32	0
14	16	20	t	20	31	17
0	15	20	u	20	19	0
0	12	0	v	0	12	0
0	12	0	w	0	12	0
0	12	0	x	0	12	0
12	12	0	y	0	12	6
10	36	30	z	30	32	13
10	29	40	agrave	10	14	34
10	29	40	acircumflex	10	14	34
33	29	40	adieresis	10	14	34
34	34	40	ccedilla	40	30	17
24	35	35	egrave	30	28	-62
24	35	35	eacute	30	28	-62
24	35	35	ecircumflex	30	28	-62
-3	25	30	dotlessi	20	19	52
-3	25	-13	icircumflex	-13	19	52
-3	25	-10	idieresis	-10	19	52
17	36	35	ocircumflex	35	36	35
17	36	35	odieresis	35	36	35
17	36	35	oe	30	28	-62
21	25	30	germandbls	30	32	0
0	15	20	ugrave	20	19	0
0	15	20	ucircumflex	20	19	0
0	15	20	udieresis	20	19	0

Source Sans Pro Regular

LSB kf	*LSB* control	*typeform*	*RSB* control	*RSB* kf
12	3	A	3	12
94	90	B	40	48
59	52	C	32	-186
94	90	D	51	54
94	90	E	49	-109
94	90	F	26	-155
58	52	G	67	-15
94	90	H	90	94
94	90	I	90	94
-112	31	J	87	84
94	90	K	4	-38
94	90	L	26	-143
94	90	M	90	94
94	90	N	90	94
59	52	O	51	59
94	90	P	43	-110
59	52	Q	37	44
94	90	R	25	24
-47	42	S	39	-5
-104	28	T	28	-104
75	87	U	87	76
-39	0	V	0	-37
6	23	W	24	8
-18	15	X	15	-18
-76	-1	Y	-1	-76
-32	45	Z	42	-78
17	52	a	71	85
94	82	b	46	62
56	46	c	25	-74
60	47	d	82	94
56	46	e	38	4
37	30	f	-27	-32
50	45	g	12	-27
94	82	h	73	87
79	67	i	65	77
-29	-40	j	66	78
94	82	k	9	-3
92	82	l	39	48
94	82	m	76	87
94	82	n	76	87
56	46	o	46	56
94	82	p	48	61
60	47	q	82	94
94	82	r	-3	-59
6	28	s	32	17
27	24	t	13	-11
87	75	u	82	94
7	12	v	12	8
27	24	w	24	29
12	14	x	14	12
-3	12	y	12	10
-4	31	z	26	-6
12	3	Agrave	3	-6
12	3	Acircumflex	3	-6
94	90	Egrave	49	45
94	90	Eacute	49	45
59	52	Ocircumflex	51	44
59	52	Odieresis	51	44
75	87	Udieresis	87	99
17	52	agrave	71	106
17	52	acircumflex	71	87
17	52	adieresis	71	96
56	46	ccedilla	25	15
56	46	egrave	38	28
56	46	eacute	38	28
56	46	ecircumflex	38	28
-32	-32	icircumflex	-32	-32
-23	-23	idieresis	-23	-23
82	82	dotlessi	82	82
56	46	ocircumflex	46	36
56	46	odieresis	46	36
46	46	oe	38	38
82	82	germandbls	29	29
87	75	ugrave	82	70
87	75	ucircumflex	82	70
87	75	udieresis	82	70

Source Serif 4 Regular

	LSB		***typeform***		***RSB***	
kf	**composite**	**control**		**control**	**composite**	**kf**
-49	7	13	A	15	7	-49
4	2	38	B	44	29	31
48	46	48	C	42	7	-178
4	2	38	D	47	40	42
4	2	40	E	41	31	-5
4	2	40	F	35	-106	-115
48	46	48	G	23	-30	-96
4	2	38	H	38	2	4
4	2	38	I	38	2	4
-86	-88	-23	J	29	-11	-9
4	2	38	K	13	7	-85
4	2	40	L	38	7	-102
6	4	36	M	38	2	4
9	7	38	N	34	-4	-1
52	49	48	O	48	48	51
4	2	38	P	31	7	-105
52	50	48	Q	48	49	51
4	2	38	R	17	7	-28
3	48	40	S	48	35	-7
-134	-54	20	T	20	-54	-133
-21	-24	32	U	29	-24	-22
-128	-127	14	V	10	-126	-122
-96	-115	14	W	10	-116	-97
-85	7	11	X	10	7	-84
-129	-140	19	Y	11	-149	-137
-41	13	24	Z	24	21	-39
37	48	47	a	9	-3	-1
14	12	33	b	47	57	59
53	51	47	c	37	29	-54
58	56	47	d	32	16	18
54	52	47	e	43	38	7
25	23	37	f	-79	-98	-116
40	37	34	g	29	26	-43
14	11	33	h	27	14	16
26	23	37	i	39	20	22
-90	-92	-73	j	74	68	70
14	11	33	k	10	7	-38
14	11	33	l	36	15	18
26	24	37	m	29	14	17
26	24	37	n	29	14	16
52	49	47	o	47	49	51
18	16	40	p	47	57	59
58	56	47	q	17	6	8
26	24	39	r	14	7	-72
41	55	48	s	45	47	35
11	9	19	t	11	37	-19
15	13	21	u	35	20	22
-42	7	12	v	17	7	-43
-29	7	12	w	17	7	-30
-22	7	17	x	17	7	-27
-53	-3	4	y	12	7	-40
26	51	33	z	30	47	22
-49	7	13	Agrave	15	7	77
-49	7	13	Acircumflex	15	7	77
4	2	40	Egrave	41	31	77
4	2	40	Eacute	41	31	77
52	49	48	Ocircumflex	48	48	44
52	49	48	Odieresis	48	48	44
-21	49	32	Udieresis	29	48	82
37	48	47	agrave	9	-3	19
37	48	47	acircumflex	9	-3	19
37	48	47	adieresis	9	-3	19
53	51	47	ccedilla	37	29	31
54	52	47	egrave	43	38	36
54	52	47	eacute	43	38	36
54	52	47	ecircumflex	43	38	36
12	23	12	icircumflex	12	20	12
8	23	8	idieresis	5	20	5
37	23	37	idotless	39	20	39
52	49	47	ocircumflex	47	49	42
52	49	47	odieresis	47	49	42
49	49	47	oe	43	38	38
16	16	33	germandbls	26	47	47
15	13	21	ugrave	35	20	41
15	13	21	ucircumflex	35	20	41
15	13	21	udieresis	35	20	41

STIX Two Text Regular

	LSB		***typeform***		***RSB***	
kf	**composite**	**control**		**control**	**composite**	**kf**
-60	7	3	A	4	7	-64
-4	7	36	B	35	33	33
47	47	50	C	38	7	-184
-5	7	35	D	50	39	39
-4	7	36	E	28	29	4
-4	7	36	F	14	-119	-119
49	49	50	G	35	7	-59
-4	7	36	H	35	7	-5
-6	7	34	I	35	7	-5
-60	7	0	J	21	-13	-13
-6	7	34	K	6	7	-151
2	7	42	L	17	7	-110
-2	7	28	M	45	7	4
-1	7	29	N	27	-6	-6
47	47	50	O	50	45	45
3	7	44	P	25	7	-119
46	46	50	Q	23	15	15
3	7	43	R	-5	7	-71
8	47	35	S	33	33	7
-147	-147	15	T	16	-146	-146
-30	-30	20	U	18	-28	-28
-147	-124	-10	V	-7	-119	-140
-137	-120	-10	W	1	-106	-110
-95	7	2	X	4	7	-94
-167	-153	-5	Y	-5	-156	-163
-62	7	28	Z	27	18	-43
-60	7	3	Agrave	4	7	-64
-60	7	3	Acircumflex	4	7	-64
-4	7	36	Egrave	28	29	4
-4	7	36	Eacute	28	29	4
47	47	50	Ocircumflex	50	45	45
47	47	50	Odieresis	50	45	45
-30	-30	20	Udieresis	18	-28	-28
24	42	38	a	4	7	1
10	10	8	b	35	51	51
51	51	33	c	19	26	-45
51	51	33	d	25	19	19
51	51	34	e	30	31	-7
14	14	18	f	-72	-65	-89
31	31	21	g	12	32	-30
12	12	22	h	21	11	11
26	26	32	i	28	18	18
-104	-104	-95	j	65	70	70
16	16	26	k	3	7	-70
16	16	26	l	23	14	14
26	26	32	m	23	14	14
26	26	32	n	21	10	10
49	49	34	o	34	50	50
18	18	22	p	35	51	51
51	51	33	q	14	19	19
23	23	29	r	7	7	-57
56	63	39	s	27	48	40
32	32	37	t	3	7	-17
16	16	23	u	30	21	21
-45	7	-13	v	-17	7	-44
-49	7	-17	w	-13	7	-40
-28	7	-2	x	-3	7	-25
-51	7	-12	y	-10	7	-39
15	40	31	z	33	51	25
24	42	38	agrave	4	7	1
24	42	38	acircumflex	4	7	1
24	42	38	adieresis	4	7	1
51	51	33	ccedilla	19	26	-45
51	51	34	egrave	30	31	-7
51	51	34	eacute	30	31	-7
51	51	34	ecircumflex	30	31	-7
26	-10	-6	icircumflex	28	26	18
26	-35	-29	idieresis	5	1	18
26	32	32	dotlessi	28	28	18
49	49	34	ocircumflex	34	50	50
49	49	34	odieresis	34	50	50
49	49	34	oe	30	31	-7
14	14	22	germandbls	31	48	40
16	16	23	ugrave	30	21	21
16	16	23	ucircumflex	30	21	21
16	16	23	udieresis	30	21	21

Yrsa Regular

kf	LSB composite	LSB control	typeform	RSB control	RSB composite	kf
-93	16	2	A	-1	16	-93
17	12	109	B	111	87	92
129	124	125	C	121	16	-319
17	12	109	D	124	106	111
17	12	109	E	92	16	-179
16	11	109	F	67	-178	-359
126	121	124	G	89	-13	-147
17	12	109	H	109	12	17
17	12	109	I	110	13	18
-96	-101	-7	J	97	4	9
17	12	109	K	14	16	-186
17	12	109	L	59	16	-225
-24	16	63	M	44	16	-38
17	12	109	N	97	7	12
133	128	125	O	125	116	121
17	12	109	P	79	16	-228
132	127	125	Q	-30	-40	-35
17	12	109	R	32	16	-66
33	116	133	S	113	64	2
-292	16	77	T	77	16	-292
-1	-6	101	U	98	-13	-8
-273	-248	-9	V	8	-245	-248
-225	-231	-2	W	10	-215	-210
-175	16	25	X	23	16	-168
-332	-314	-7	Y	17	-298	-297
-92	20	67	Z	120	75	-47
-18	76	97	a	56	29	35
0	-5	-15	b	112	120	125
122	117	111	c	102	83	-84
126	121	113	d	71	46	51
124	119	111	e	115	88	-15
47	42	80	f	-119	-160	-234
92	86	92	g	48	63	-90
23	18	46	h	55	27	32
55	50	84	i	70	42	47
-70	-76	-48	j	199	200	206
23	18	46	k	32	16	-73
20	14	46	l	53	22	27
55	50	84	m	55	26	32
55	50	84	n	56	28	33
126	121	113	o	112	121	126
38	33	61	p	113	121	126
126	121	112	q	56	58	63
55	50	84	r	72	16	-101
104	133	122	s	108	117	87
42	37	63	t	30	44	-100
34	29	58	u	71	47	52
-73	16	18	v	28	16	-63
-35	16	27	w	30	16	-32
-43	16	48	x	50	16	-38
-75	7	27	y	21	16	-59
39	94	85	z	114	126	62
-93	16	2	Agrave	-1	16	94
-93	16	2	Acircumflex	-1	16	94
17	12	109	Egrave	92	16	184
17	12	109	Eacute	92	16	184
133	128	125	Ocircumflex	125	116	117
133	128	125	Odieresis	125	116	117
-1	-6	101	Udieresis	96	-13	198
-18	76	97	agrave	56	29	171
-18	76	97	acircumflex	56	29	171
-18	76	97	adieresis	56	29	171
122	117	111	ccedilla	102	83	91
124	119	111	egrave	115	88	102
124	119	111	eacute	115	88	102
124	119	111	ecircumflex	115	88	102
-3	50	-3	icircumflex	54	42	54
84	50	84	dotlessi	70	42	70
-16	50	-16	idieresis	50	42	50
126	121	113	ocircumflex	112	121	99
126	121	113	odieresis	112	121	99
113	121	113	oe	115	88	115
81	45	81	germandbls	73	117	73
34	29	58	ugrave	71	47	95
34	29	58	ucircumflex	71	47	95
34	29	58	udieresis	71	47	95

Yrsa Bold

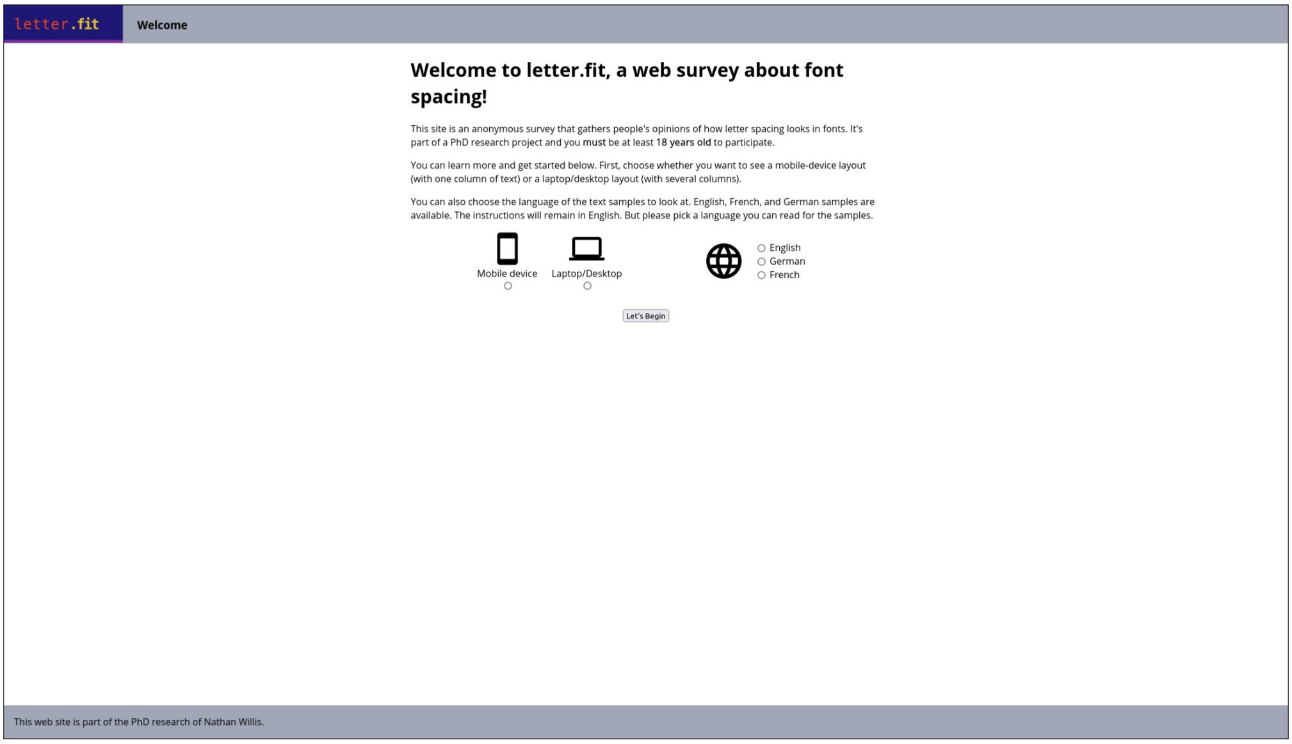
	LSB		*typeform*		*RSB*	
kf	**composite**	**control**		**control**	**composite**	**kf**
0	35	-17	A	-10	35	0
-5	-58	89	B	79	-1	154
43	41	94	C	99	35	-130
-5	-58	89	D	94	20	34
-5	-58	89	E	72	35	0
-5	-58	89	F	43	-171	-197
47	43	95	G	64	-54	-58
-5	-58	89	H	89	-59	-5
-5	-58	89	I	89	-58	-5
-130	-191	-41	J	75	-71	-11
-5	-58	89	K	10	34	0
-5	-58	89	L	46	37	0
-70	-102	40	M	32	-108	-80
-5	-58	89	N	70	-76	-16
49	44	94	O	94	32	49
-5	-58	89	P	46	34	0
47	43	94	Q	-86	-151	-134
-5	-58	89	R	26	35	154
43	53	110	S	86	-2	11
-164	-321	56	T	56	-319	-163
-7	-74	84	U	84	-84	-16
-214	-217	-20	V	-4	-219	-200
-181	-206	-12	W	-5	-211	-173
0	35	16	X	8	35	0
-238	-259	-24	Y	9	-238	-204
-56	-41	49	Z	96	14	-14
-61	0	65	a	39	-35	-27
-57	-66	-26	b	82	32	121
37	28	79	c	76	4	-85
45	36	82	d	52	-14	-5
43	35	82	e	80	4	-66
2	-7	67	f	-142	-137	-171
10	1	64	g	30	-8	118
-31	-39	29	h	42	-33	-25
6	-3	69	i	50	-19	-10
-146	-155	-82	j	164	114	123
-31	-39	29	k	21	35	-106
-30	-39	35	l	44	-29	-20
6	-3	69	m	42	-34	-26
6	-3	69	n	42	-33	-25
41	33	81	o	82	32	42
-15	-24	47	p	81	33	42
44	36	82	q	23	-27	-18
6	-2	69	r	46	35	0
51	63	97	s	74	36	20
-21	-30	47	t	10	-127	-118
-18	-27	50	u	51	-14	-6
0	35	5	v	18	35	0
0	35	11	w	20	35	0
0	35	40	x	27	35	0
0	35	15	y	9	35	0
-15	26	58	z	81	52	5
0	35	-17	Agrave	-10	35	0
0	35	-17	Acircumflex	-10	35	0
-5	-58	89	Egrave	72	35	0
-5	-58	89	Eacute	72	35	0
49	44	94	Ocircumflex	94	32	49
49	44	94	Odieresis	94	32	49
-7	-74	84	Udieresis	78	-84	-16
-61	0	65	agrave	39	-35	-27
-61	0	65	acircumflex	39	-35	-27
-61	0	65	adieresis	39	-35	-27
37	28	79	ccedilla	76	4	-85
43	35	82	egrave	80	4	-66
43	35	82	eacute	80	4	-66
43	35	82	ecircumflex	80	4	-66
6	-3	-11	icircumflex	44	-19	-10
6	-3	69	dotlessi	50	-19	-10
6	-3	-35	idieresis	19	-19	-10
41	33	81	ocircumflex	82	32	42
41	33	81	odieresis	82	32	42
41	33	81	oe	79	4	-66
2	-7	67	germandbls	42	36	20
-18	-27	50	ugrave	51	-14	-6
-18	-27	50	ucircumflex	51	-14	-6
-18	-27	50	udieresis	51	-14	-6

Appendix E: quantitative test materials

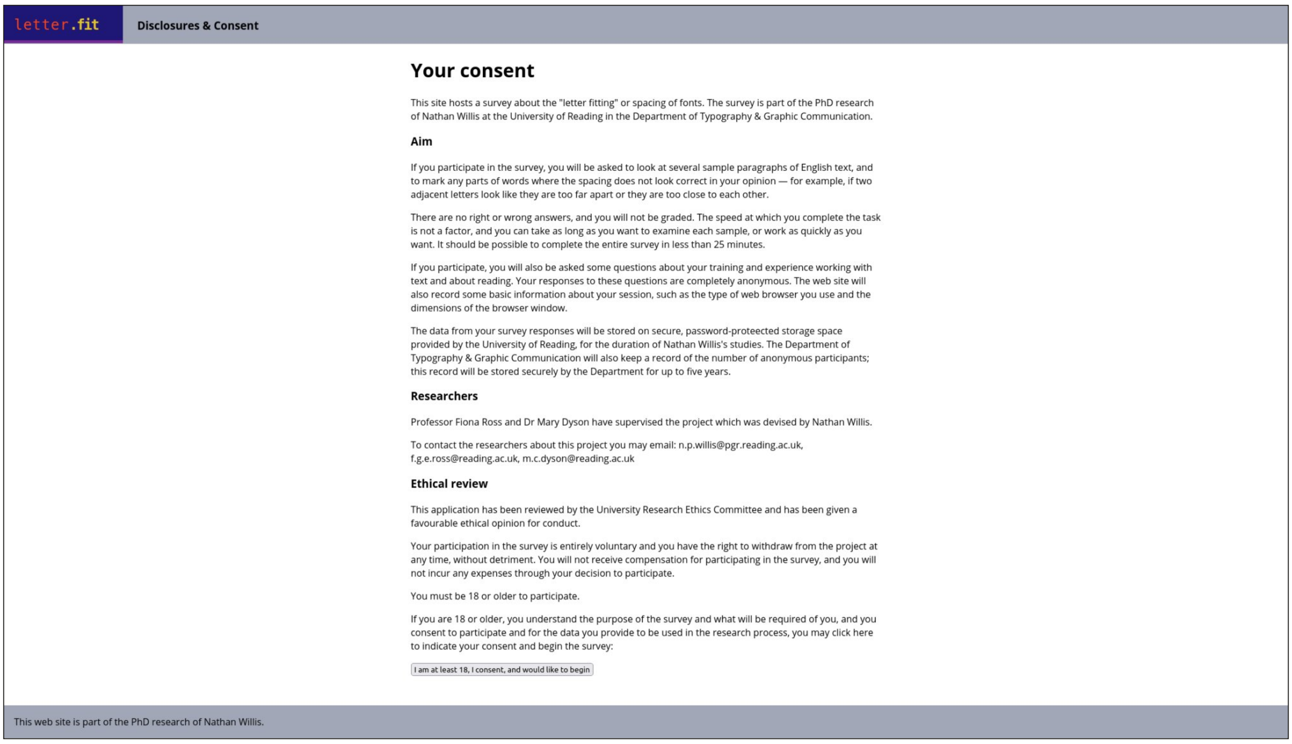
i. Public test web-application screenshots

The following pages provide screenshots of the desktop/laptop and mobile variants of each page used in the public tests. Only one example is included for the text-specimen page; all five specimen pages were identical in formatting and varied by the randomly selected test font and sample text generated by the application server.

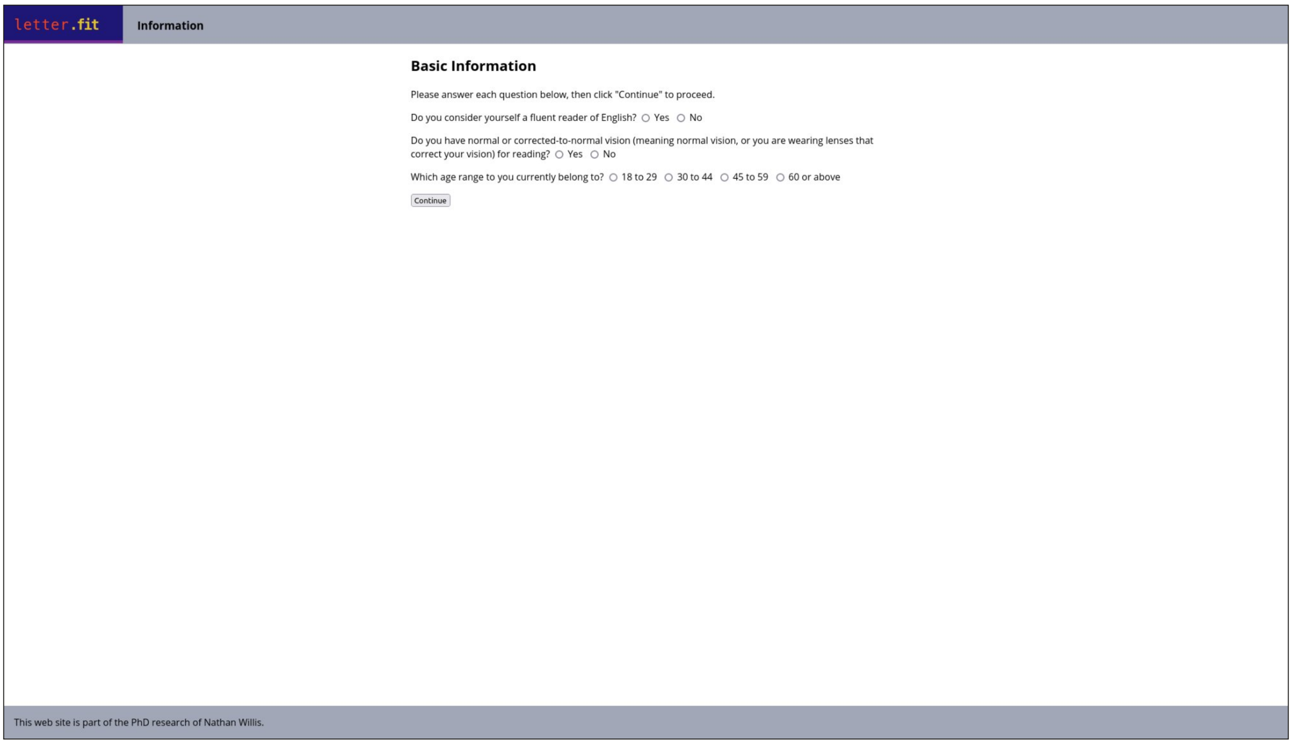
Welcome page, desktop/laptop



Disclosure & consent page, desktop/laptop



Demographic questions page, desktop/laptop


letter.fit Information

Basic Information

Please answer each question below, then click "Continue" to proceed.

Do you consider yourself a fluent reader of English? ○ Yes ○ No

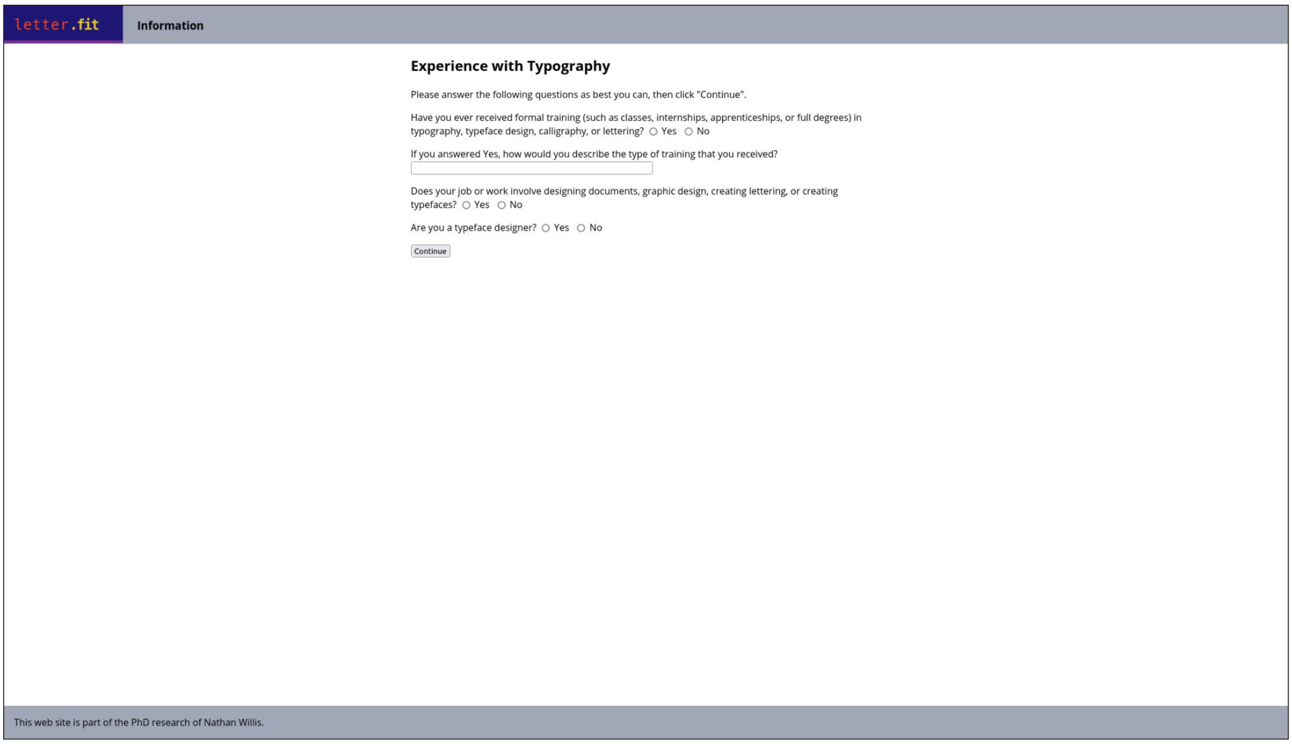
Do you have normal or corrected-to-normal vision (meaning normal vision, or you are wearing lenses that correct your vision) for reading? ○ Yes ○ No

Which age range to you currently belong to? ○ 18 to 29 ○ 30 to 44 ○ 45 to 59 ○ 60 or above

Continue

This web site is part of the PhD research of Nathan Willis.

Typographic experience questions page, desktop/laptop


letter.fit Information

Experience with Typography

Please answer the following questions as best you can, then click "Continue".

Have you ever received formal training (such as classes, internships, apprenticeships, or full degrees) in typography, typeface design, calligraphy, or lettering? ○ Yes ○ No

If you answered Yes, how would you describe the type of training that you received?

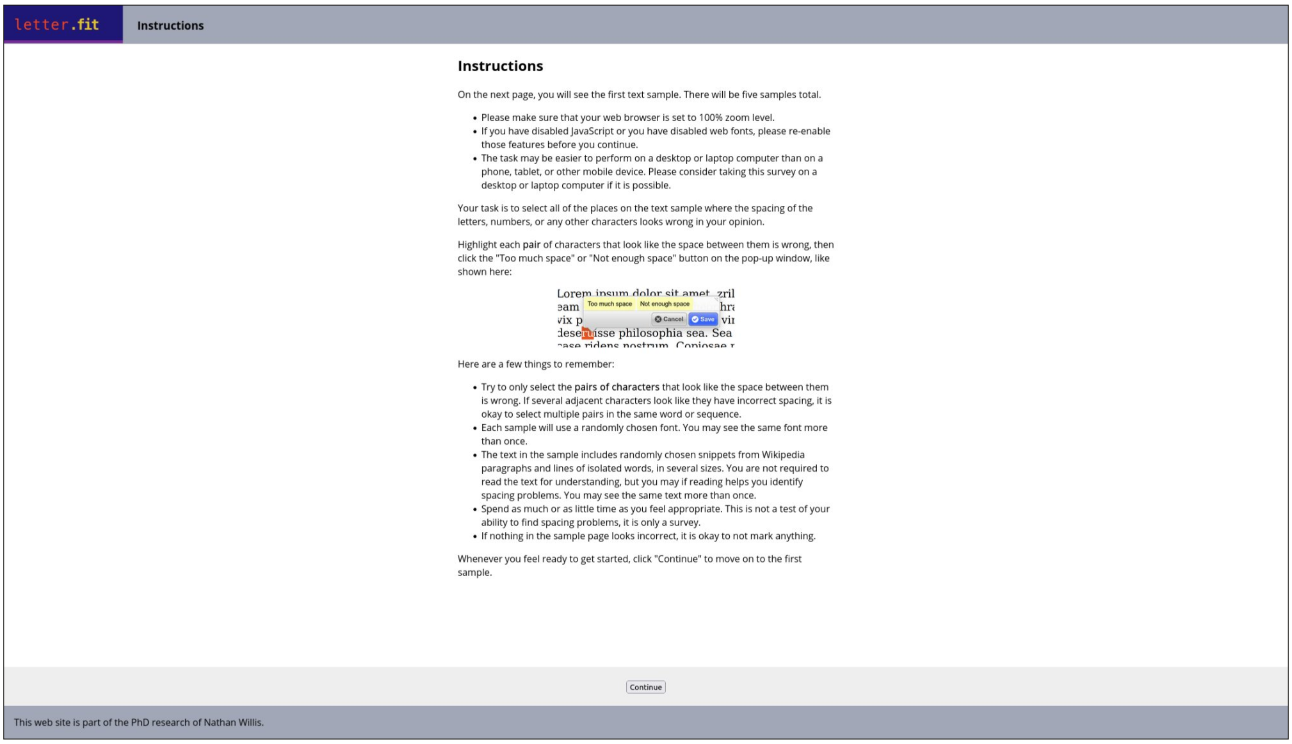
Does your job or work involve designing documents, graphic design, creating lettering, or creating typefaces? ○ Yes ○ No

Are you a typeface designer? ○ Yes ○ No

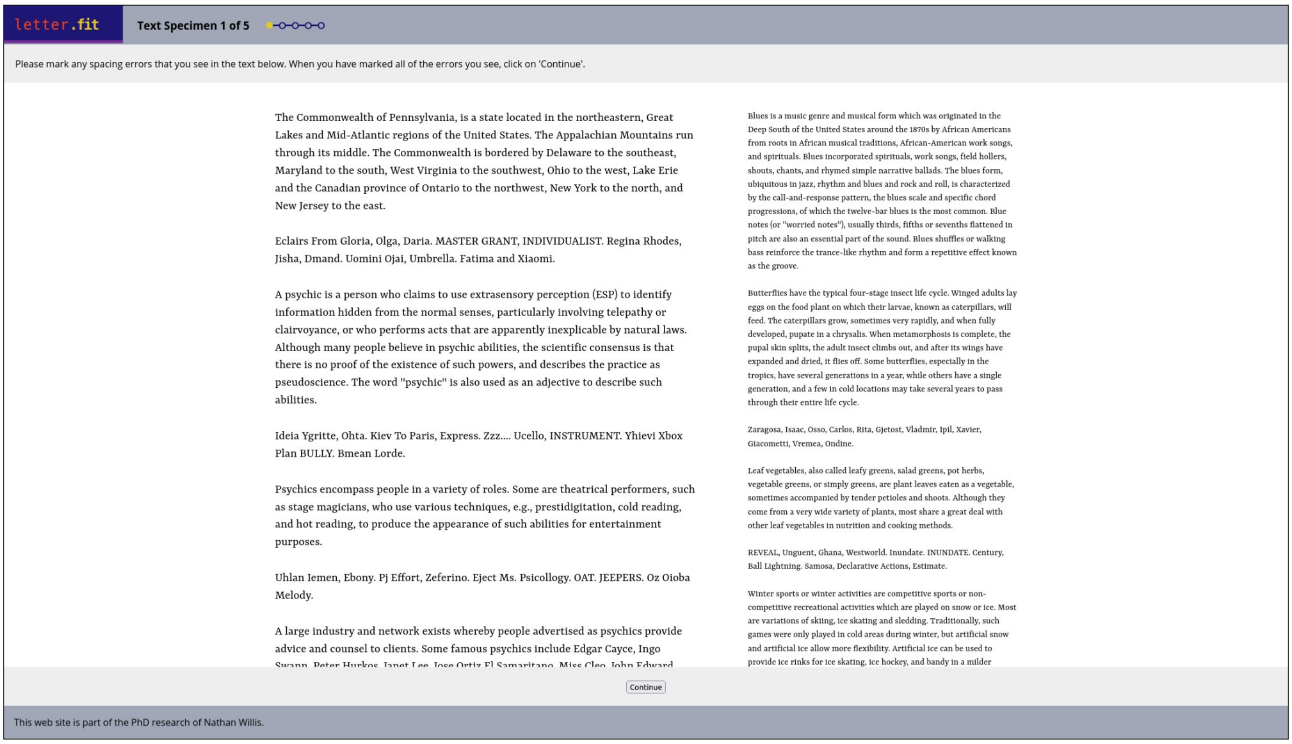
Continue

This web site is part of the PhD research of Nathan Willis.

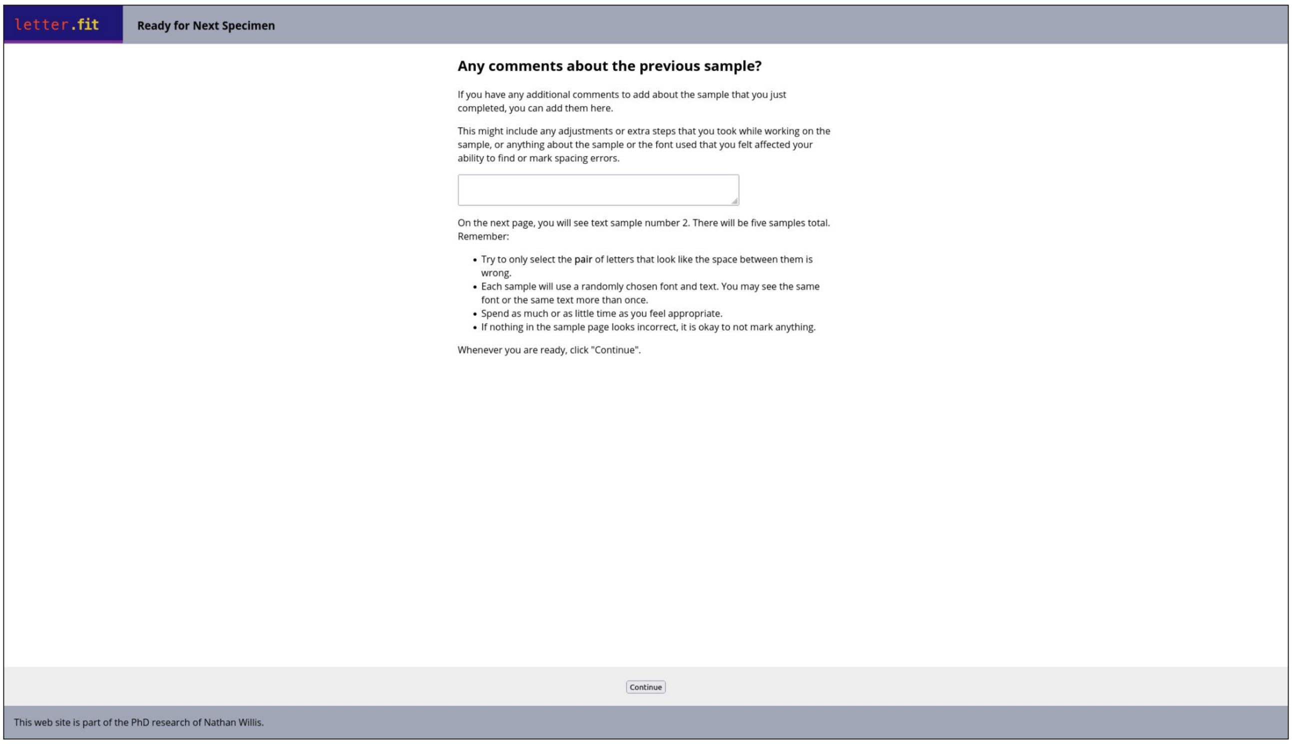
Task instructions page, desktop/laptop



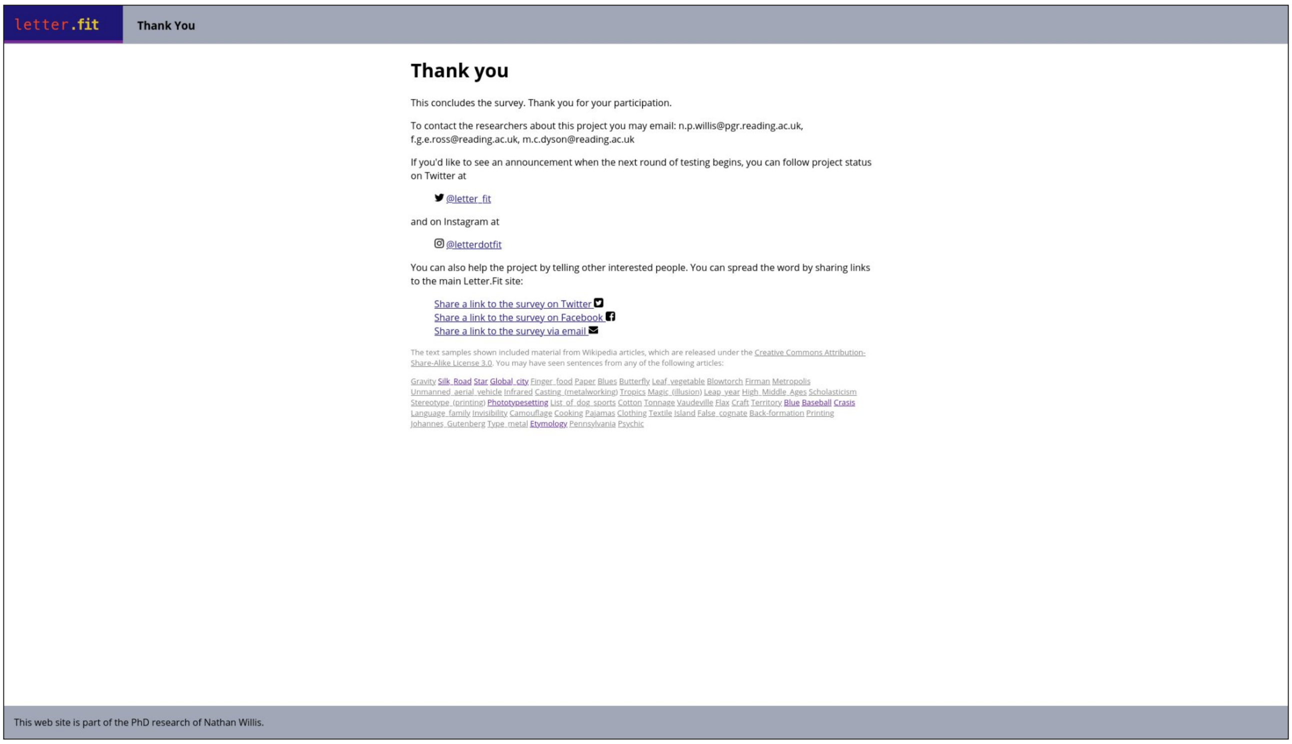
Specimen page, desktop/laptop



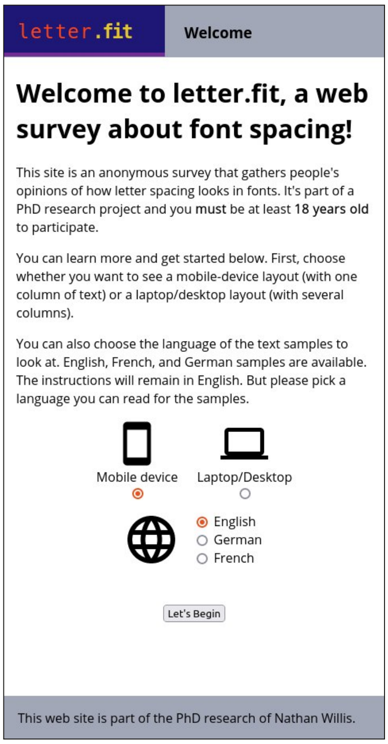
Between-specimens reset page, desktop/laptop



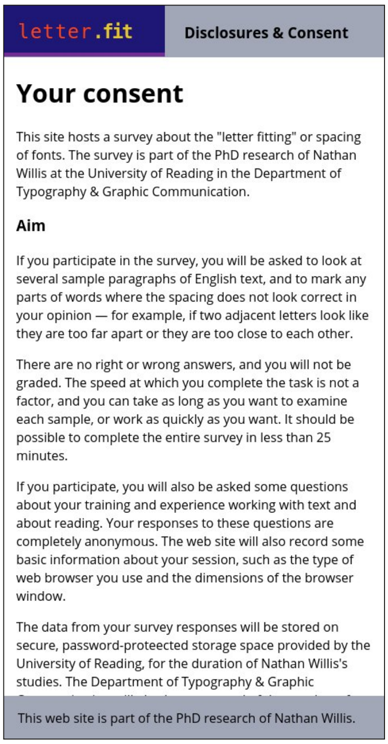
Thank you page, desktop/laptop



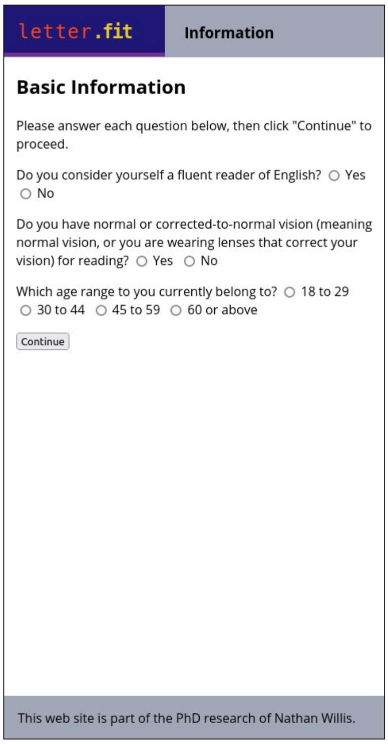
Welcome page, mobile



Disclosure & consent page, mobile



Demographic questions page, mobile



letter.fit Information

Basic Information

Please answer each question below, then click "Continue" to proceed.

Do you consider yourself a fluent reader of English? ○ Yes ○ No

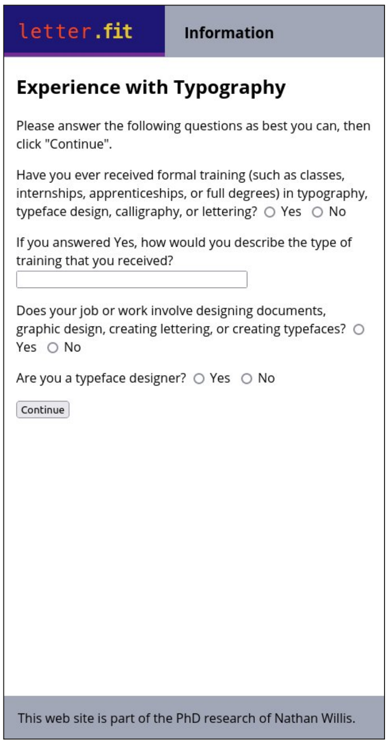
Do you have normal or corrected-to-normal vision (meaning normal vision, or you are wearing lenses that correct your vision) for reading? ○ Yes ○ No

Which age range to you currently belong to? ○ 18 to 29 ○ 30 to 44 ○ 45 to 59 ○ 60 or above

Continue

This web site is part of the PhD research of Nathan Willis.

Typographic experience questions page, mobile



letter.fit Information

Experience with Typography

Please answer the following questions as best you can, then click "Continue".

Have you ever received formal training (such as classes, internships, apprenticeships, or full degrees) in typography, typeface design, calligraphy, or lettering? ○ Yes ○ No

If you answered Yes, how would you describe the type of training that you received?

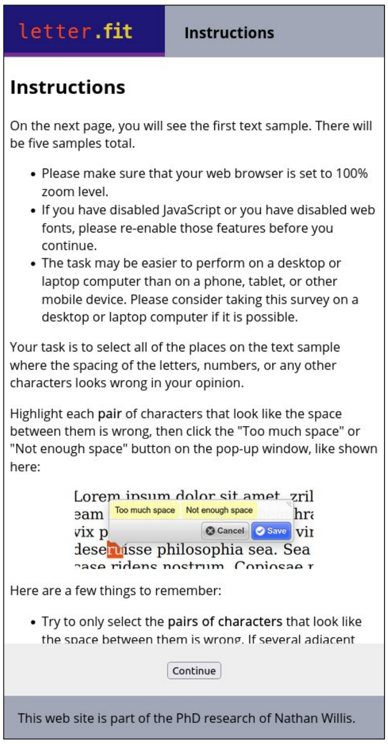
Does your job or work involve designing documents, graphic design, creating lettering, or creating typefaces? ○ Yes ○ No

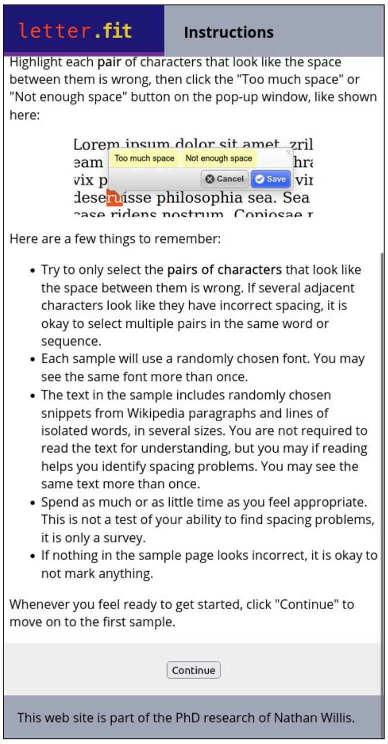
Are you a typeface designer? ○ Yes ○ No

Continue

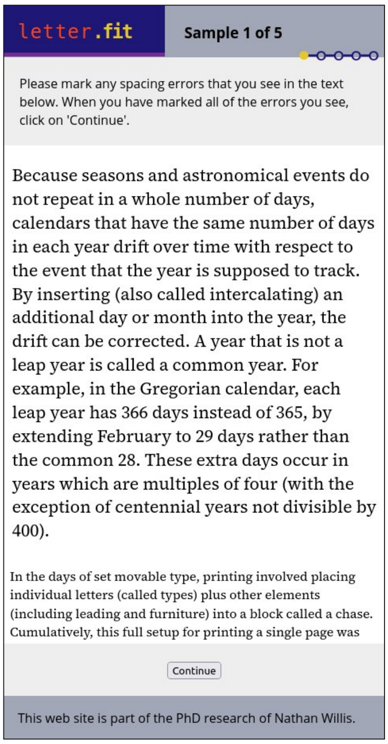
This web site is part of the PhD research of Nathan Willis.

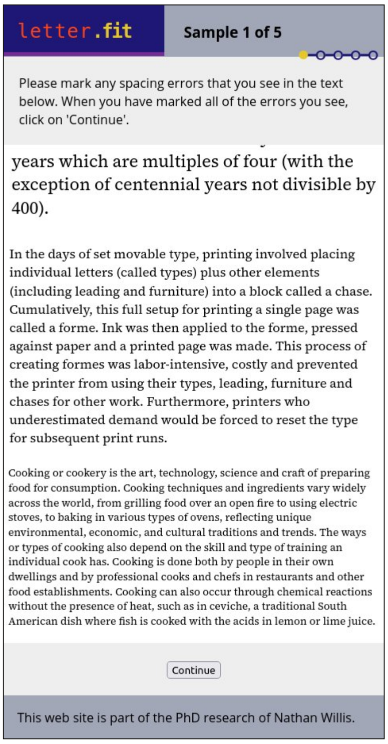
Task instructions page, mobile



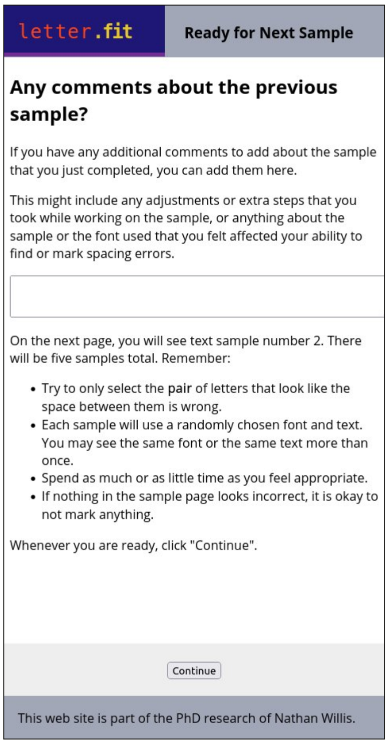


Specimen page, mobile





Between-specimens reset page, mobile


letter.fit | Ready for Next Sample

Any comments about the previous sample?

If you have any additional comments to add about the sample that you just completed, you can add them here.

This might include any adjustments or extra steps that you took while working on the sample, or anything about the sample or the font used that you felt affected your ability to find or mark spacing errors.

On the next page, you will see text sample number 2. There will be five samples total. Remember:

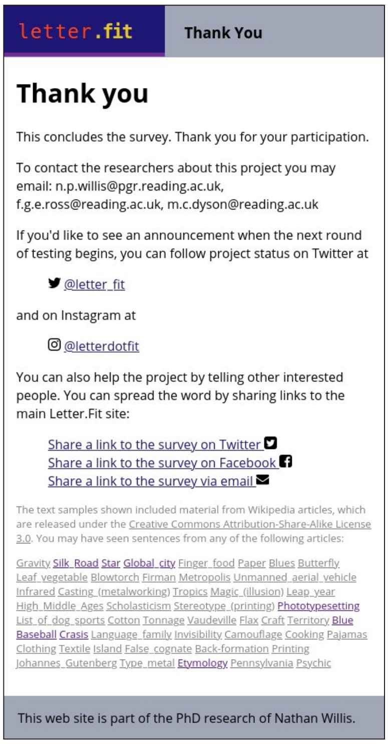
- Try to only select the pair of letters that look like the space between them is wrong.
- Each sample will use a randomly chosen font and text. You may see the same font or the same text more than once.
- Spend as much or as little time as you feel appropriate.
- If nothing in the sample page looks incorrect, it is okay to not mark anything.

Whenever you are ready, click "Continue".

Continue

This web site is part of the PhD research of Nathan Willis.

Thank you page, mobile


letter.fit | Thank You

Thank you

This concludes the survey. Thank you for your participation.

To contact the researchers about this project you may email: n.p.willis@pgr.reading.ac.uk, f.g.e.ross@reading.ac.uk, m.c.dyson@reading.ac.uk

If you'd like to see an announcement when the next round of testing begins, you can follow project status on Twitter at

@letter_fit

and on Instagram at

@letterdotfit

You can also help the project by telling other interested people. You can spread the word by sharing links to the main Letter.Fit site:

Share a link to the survey on Twitter
Share a link to the survey on Facebook
Share a link to the survey via email

The text samples shown included material from Wikipedia articles, which are released under the Creative Commons Attribution-Share-Alike License 3.0. You may have seen sentences from any of the following articles:

Gravity Silk_Road Star Global_city Finger_food Paper Blues Butterfly Leaf_vegetable Blowtorch Firman Metropolis Unmanned_aerial_vehicle Infrared Casting_(metalworking) Tropics Magic_(illusion) Leap_year High_Middle_Ages Scholasticism Stereotype_(printing) Phototypesetting List_of_dog_sports Cotton Tonnage Vaudeville Flax Craft Territory Blue Baseball Crasis Language_family Invisibility Camouflage Cooking Pajamas Clothing Textile Island False_cognate Back-formation Printing Johannes_Gutenberg Type_metal Etymology Pennsylvania Psychic

This web site is part of the PhD research of Nathan Willis.

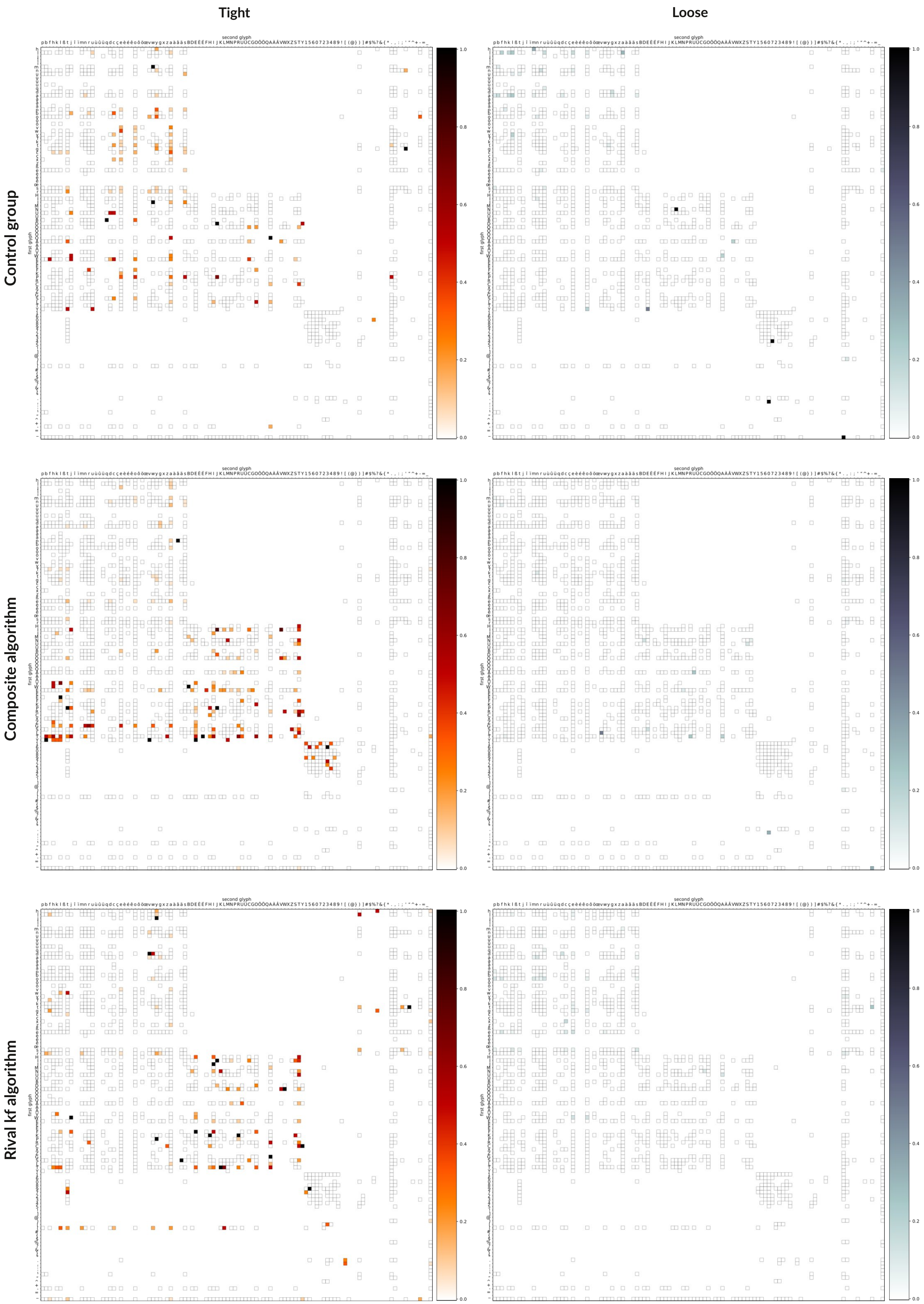
ii. Exposure mark count heatmaps

The following pages provide per–letter-pair heatmap plots of the "tight" and "loose" exposure mark rates for the fonts tested in the quantitative public tests, in each test condition for which data was collected.

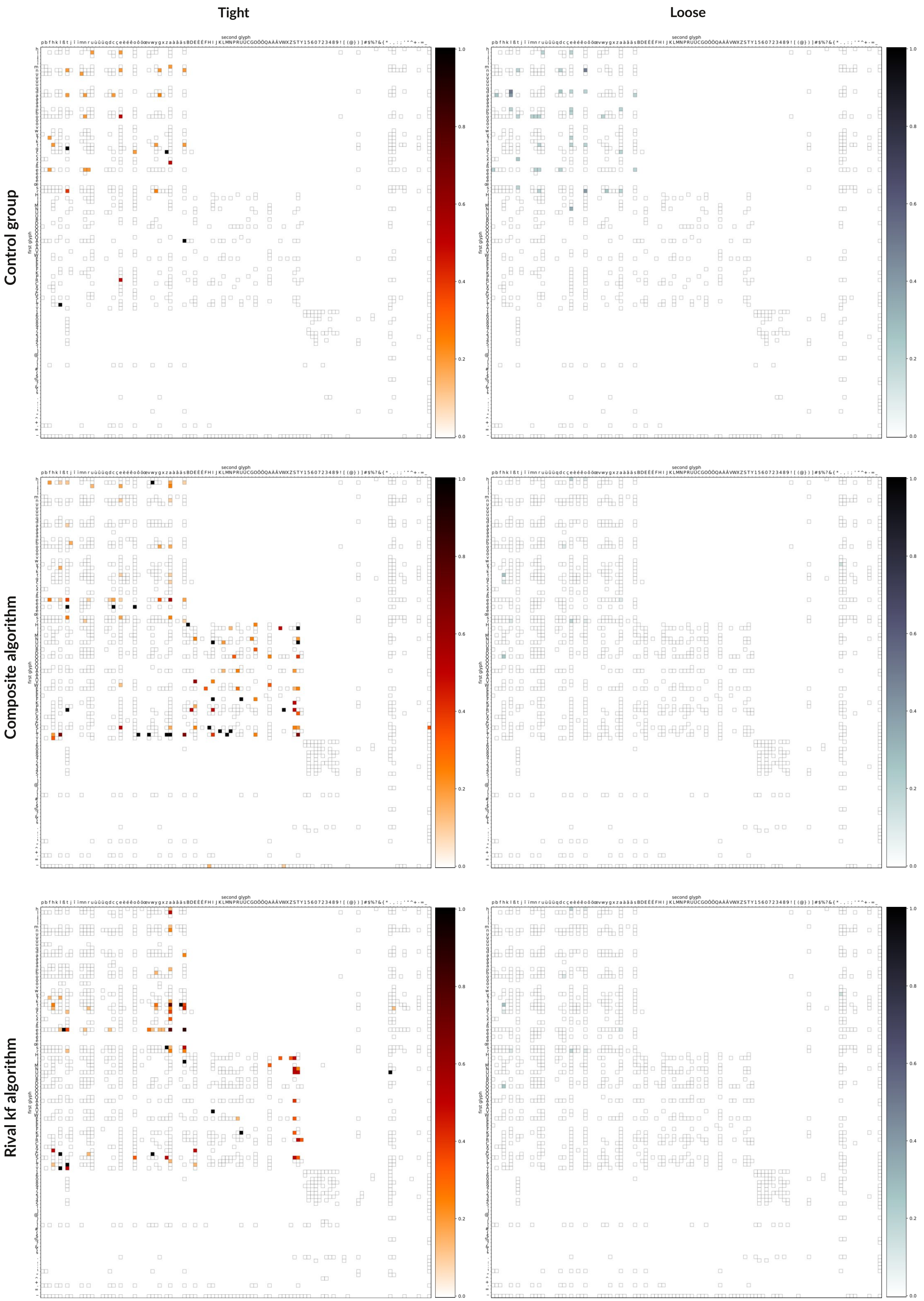
As described in chapter 6, § 6.3.1, the exposure mark rates shown represent the proportion from 0 to 1.0 of exposures in which each letter pair received a mark out of the total number of exposures that included that pair. Each cell is thus shaded according to the proportionate number of marks for the pair, not by how often the pair occurred in the sample texts. Cells with no border indicate the pair did not occur in an exposure.

Vertical axes index the first form of each pair; horizontal axes index the second form of each pair. Within each axis, forms are sorted by class (*lowercase – capital – numeral – punctuation and symbols – word space*); within each class, forms are sorted by profile shape on the interior side (i.e., by the right profile of the first form and the left profile of the second form) in the order *straight – round – diagonal – open – half-open – unbounded*.

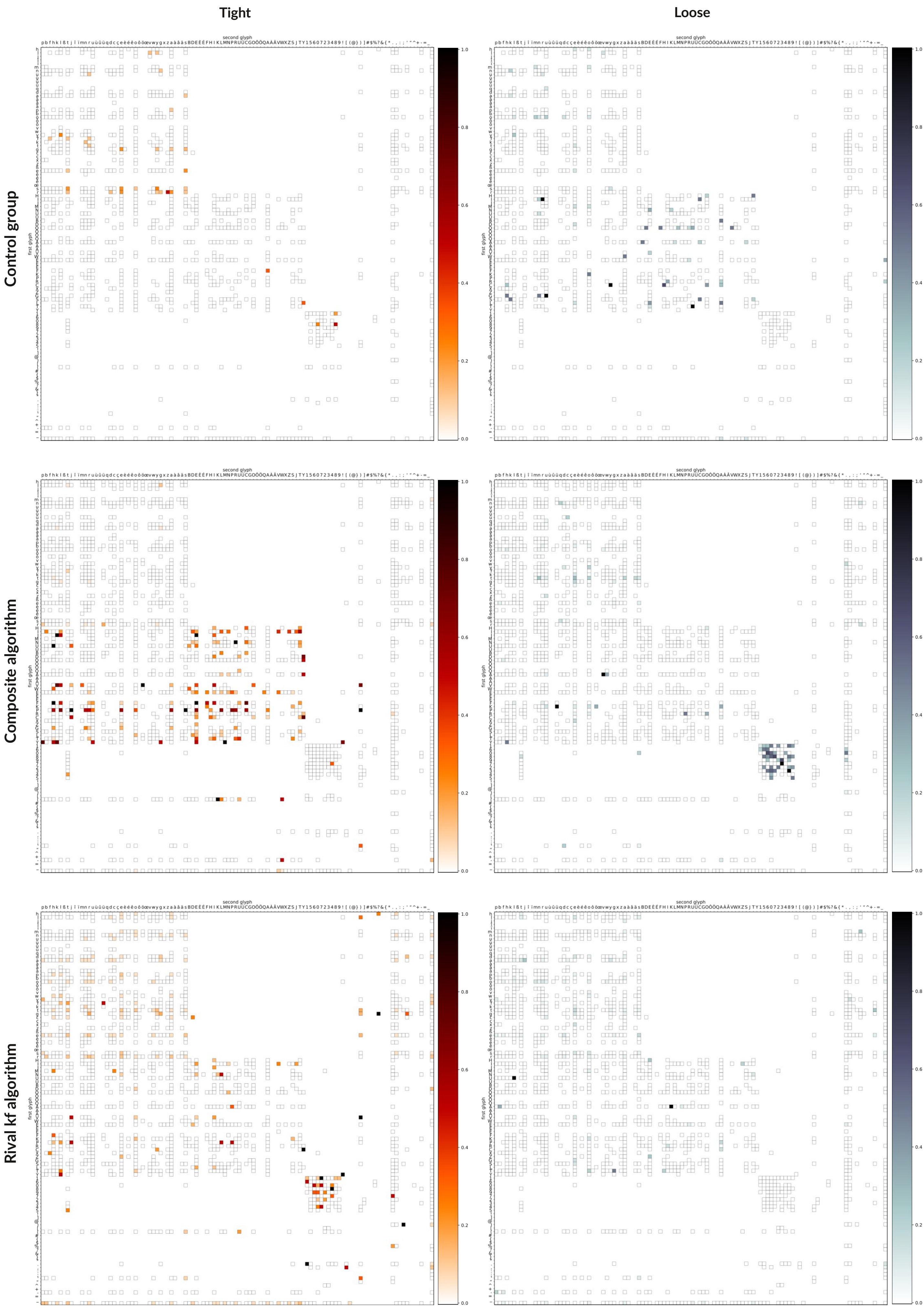
Alegreya Sans Regular: exposure mark rates



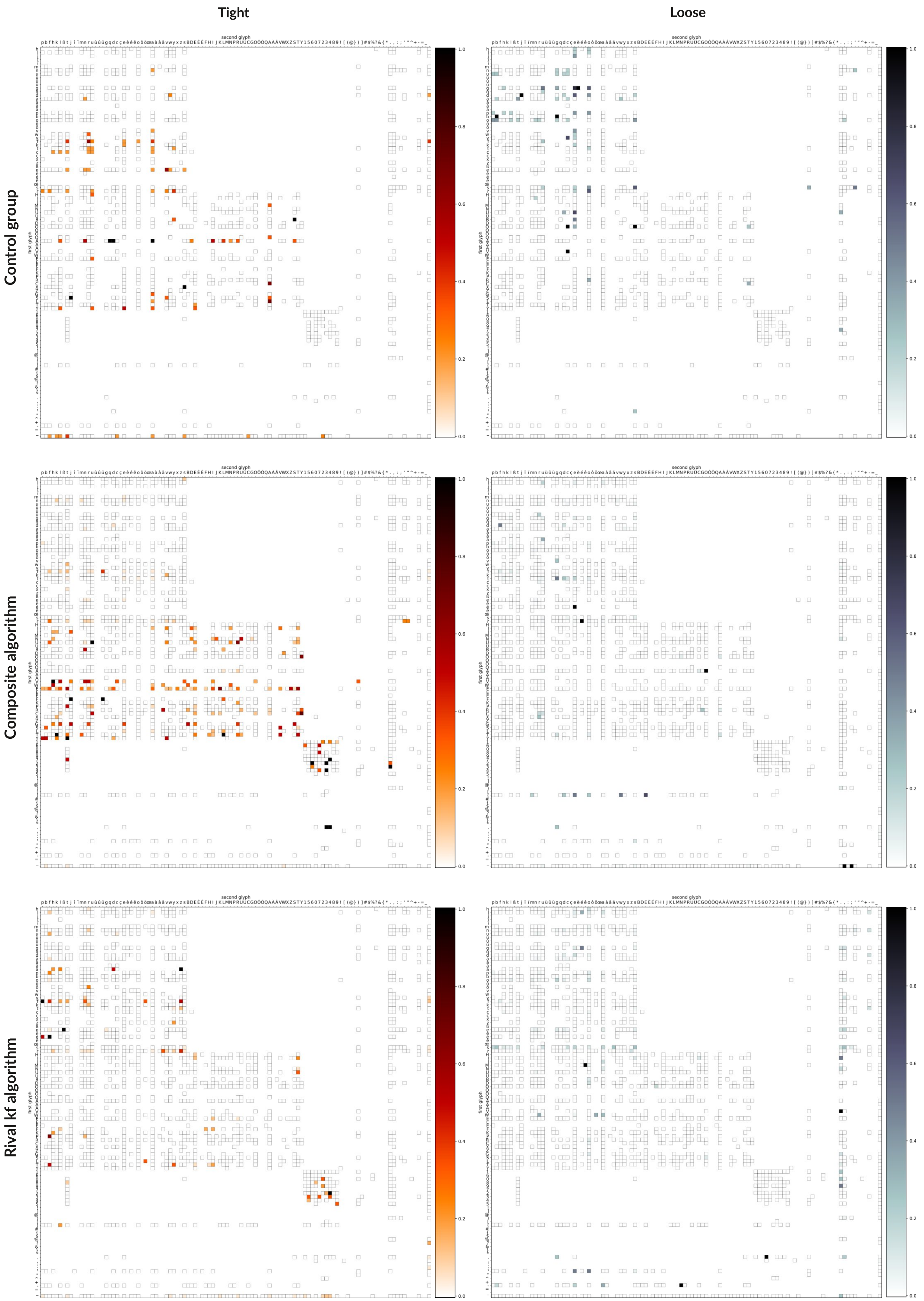
Fira Sans Condensed: exposure mark rates



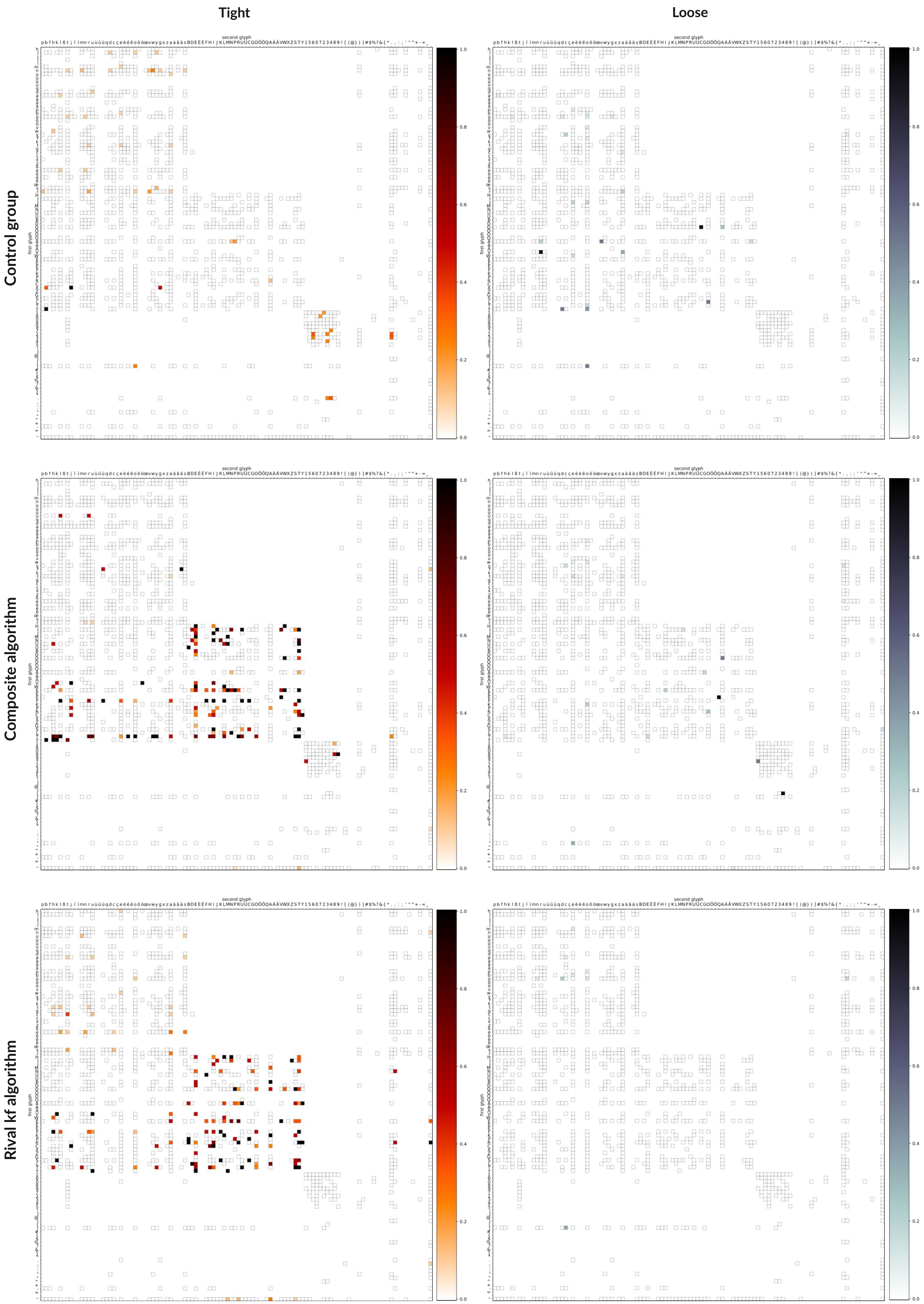
Literata Regulra opsz 14: exposure mark rates



Slabo 13px: exposure mark rates



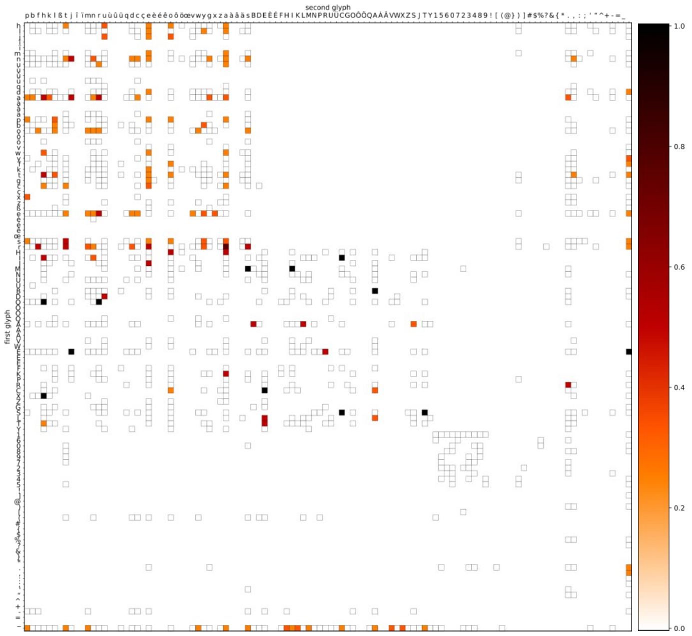
Slabo 27px: exposure mark rates

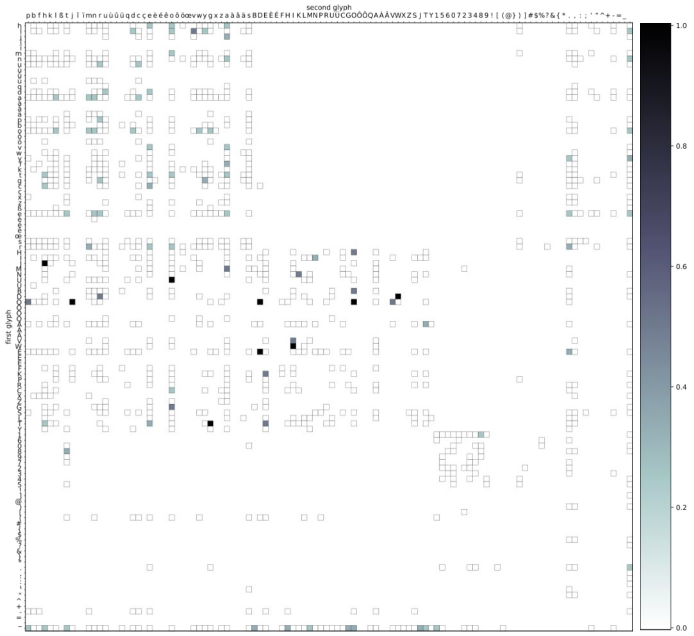


Source Sans Pro Regular: exposure mark rates

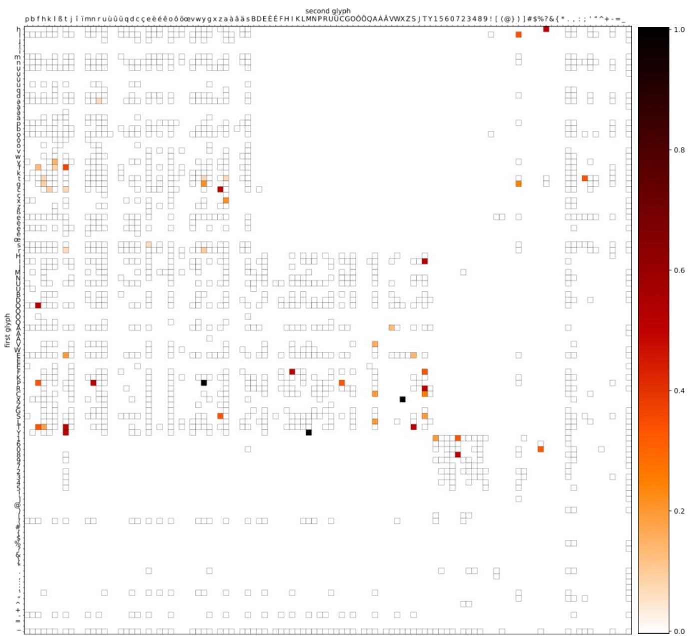
Tight | **Loose**

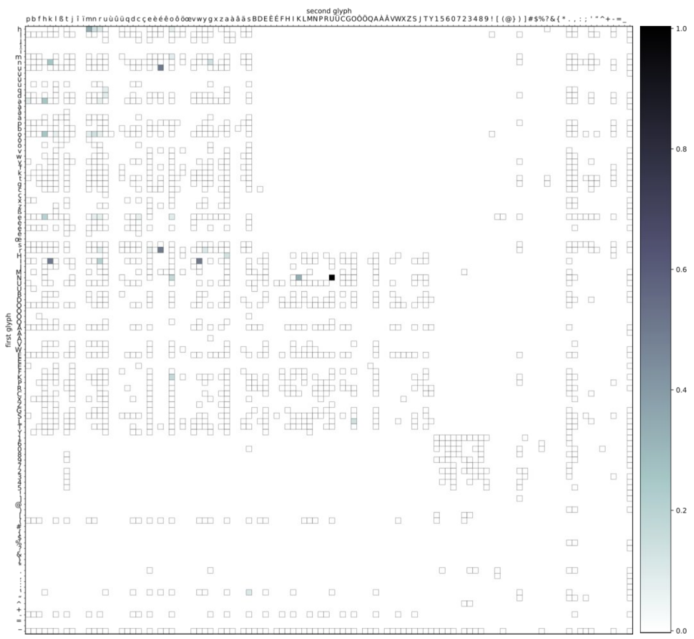
Control group



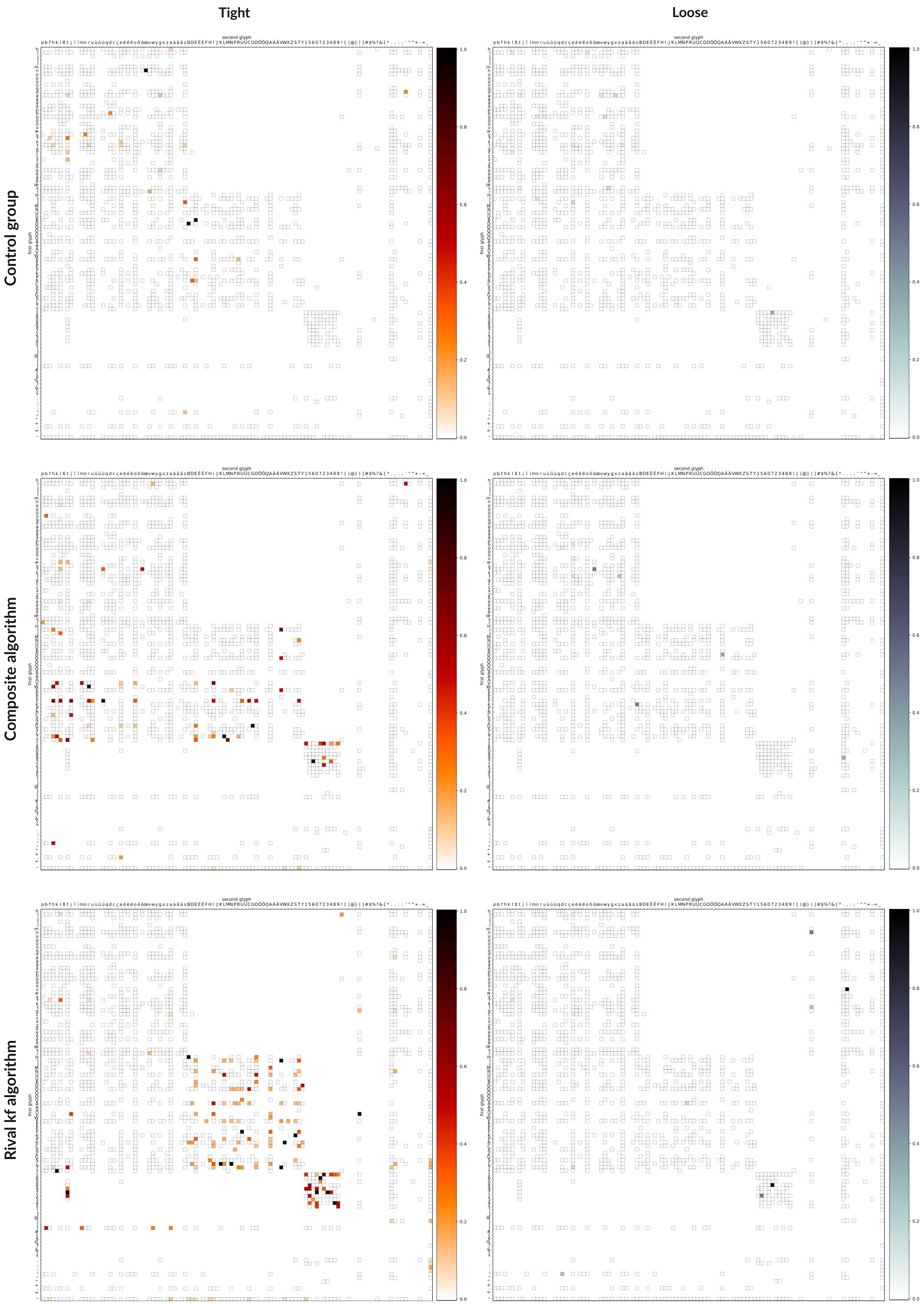


Rival kf algorithm

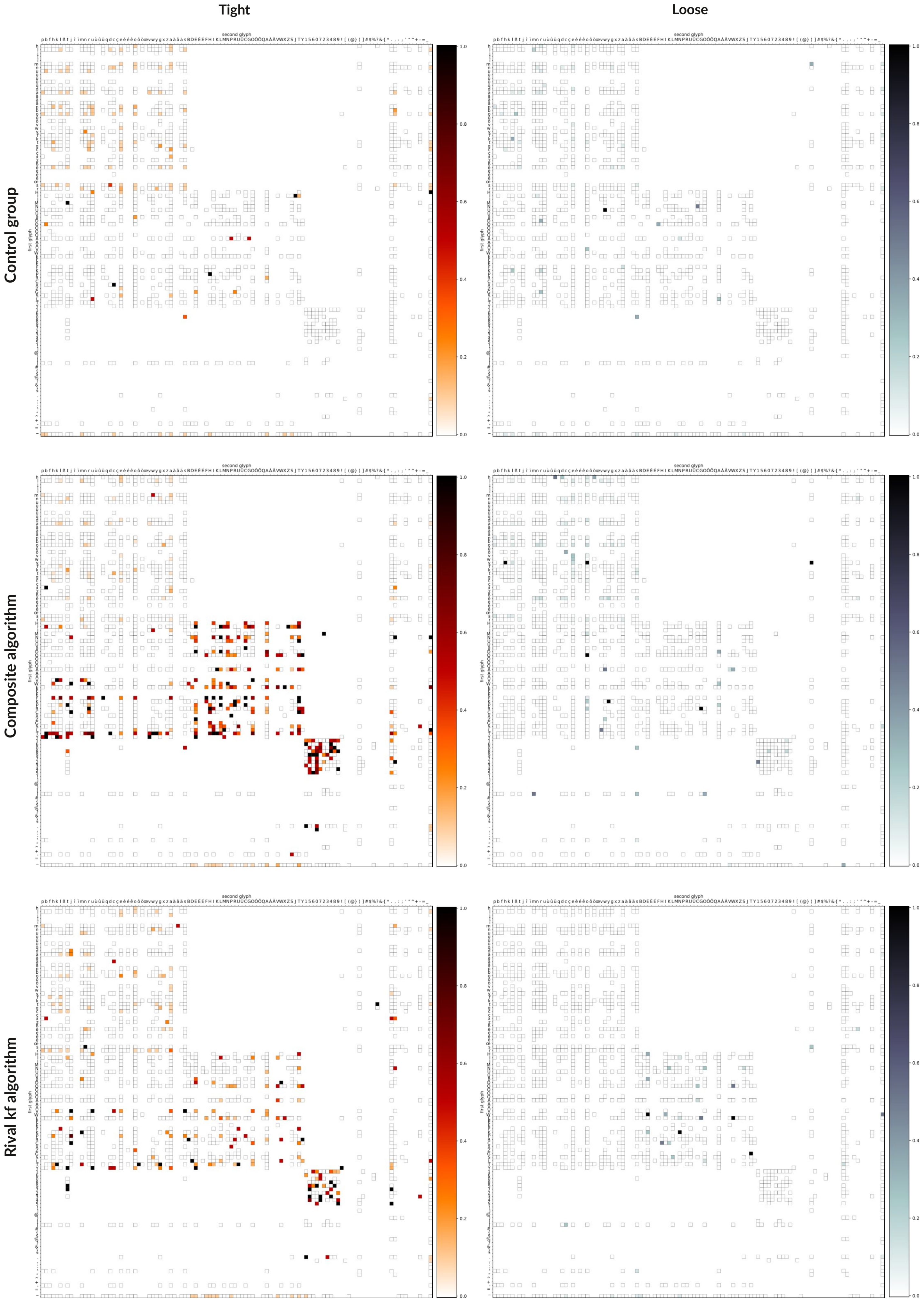




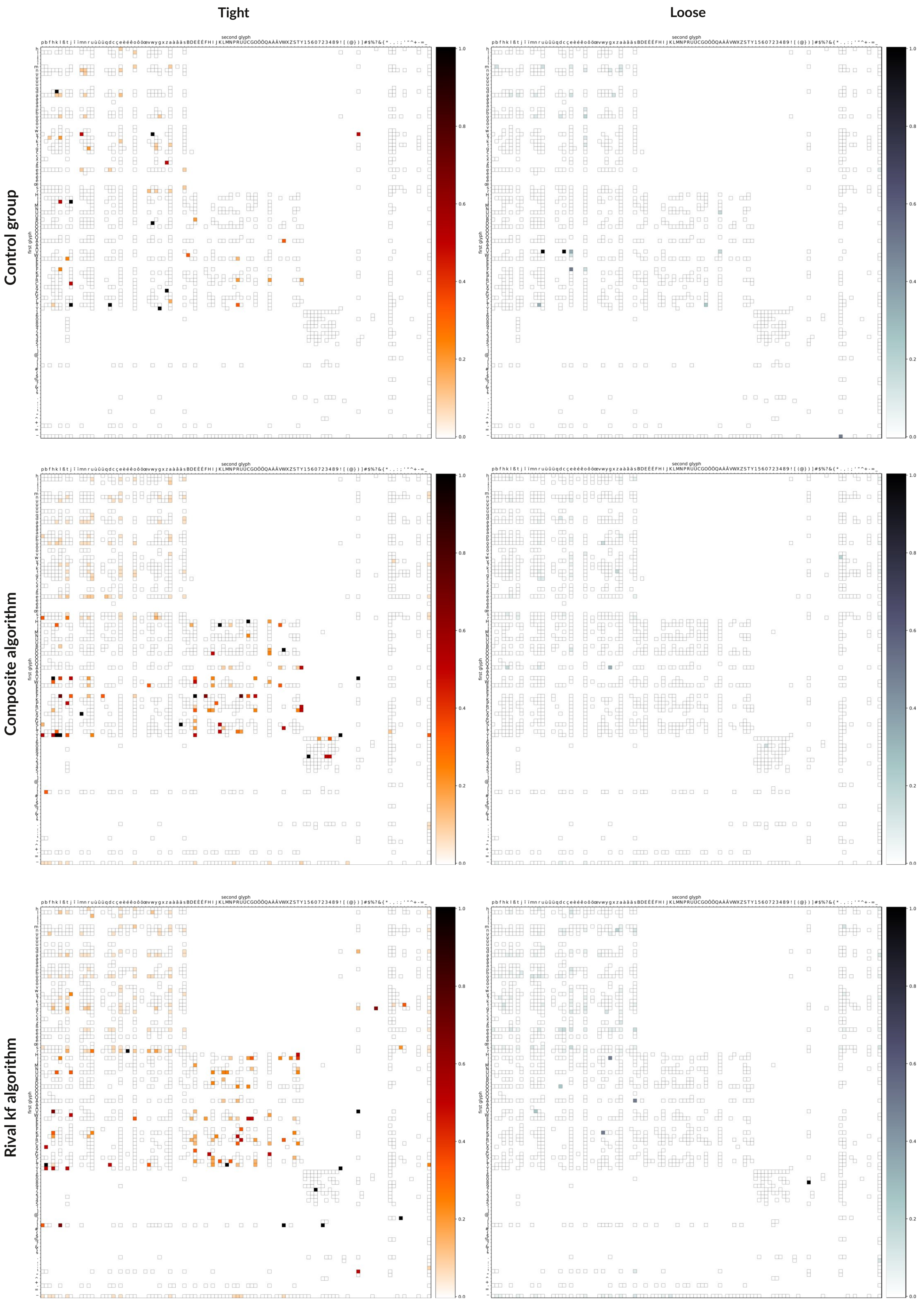
Source Serif 4 Regular exposure mark rates



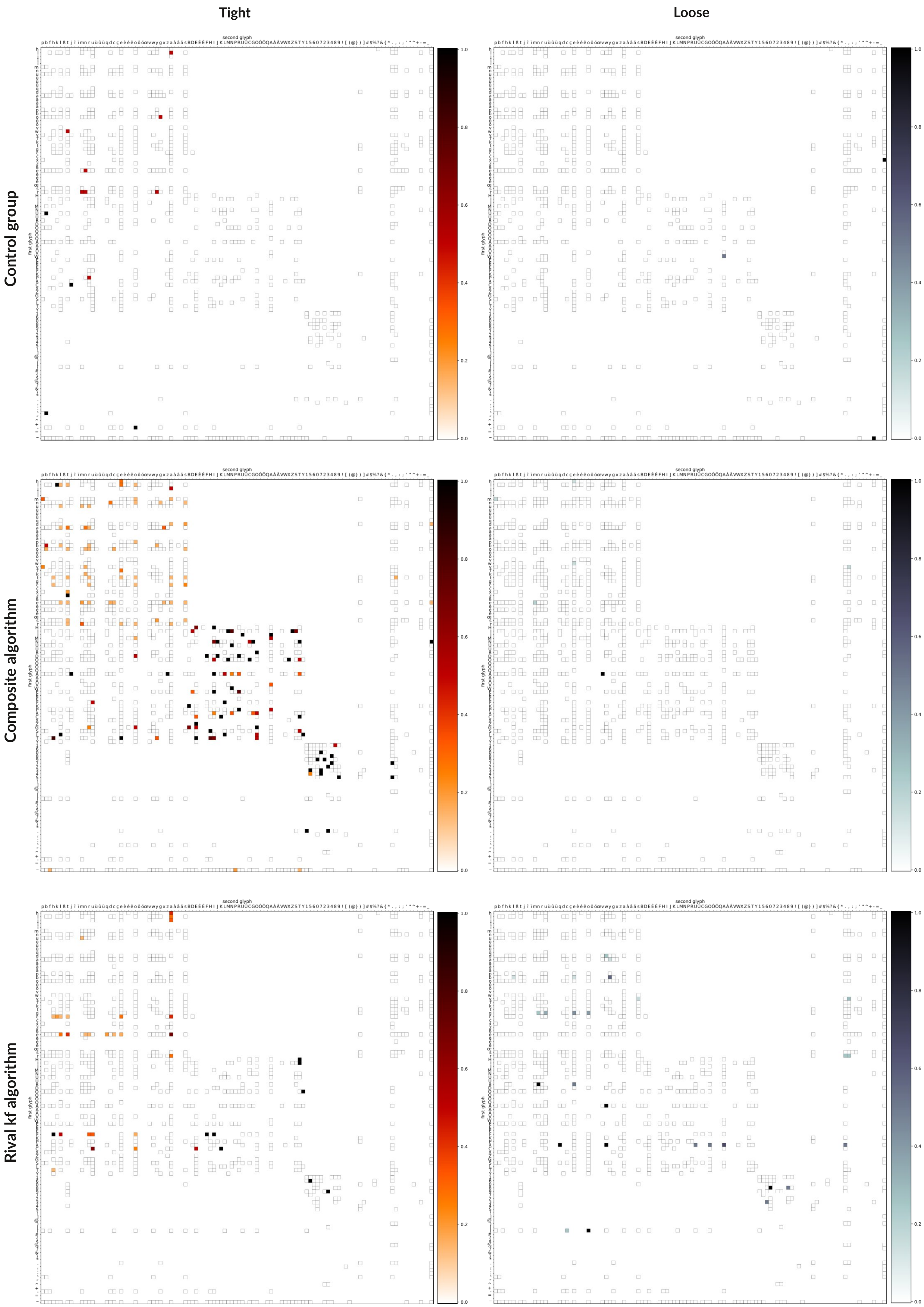
STIX Two Text Regular: exposure mark rates



Yrsa Regular: exposure mark rates



Yrsa Bold: exposure mark rates



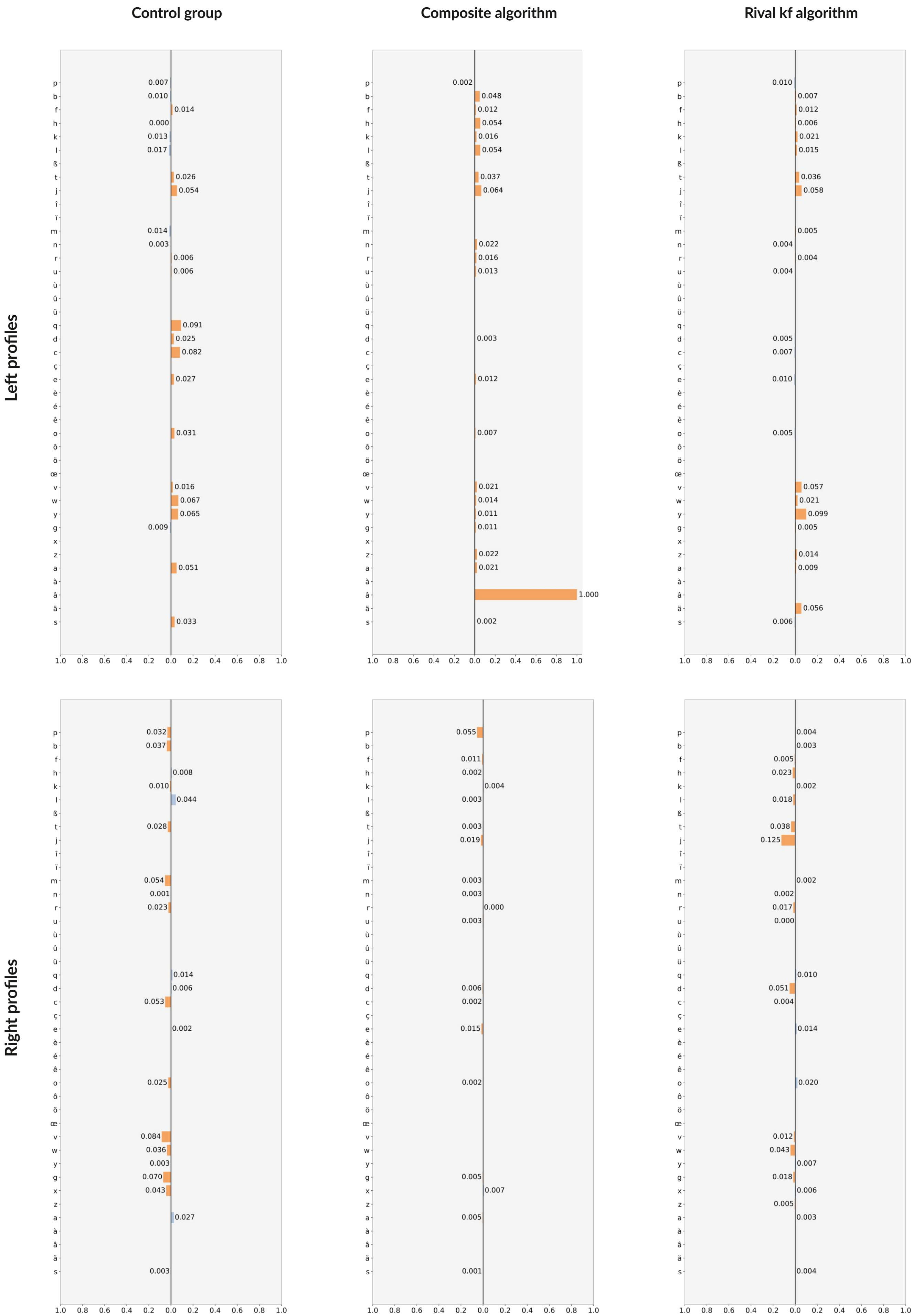
ii. Per-letterform profile mark balances

The following pages provide per-letterform balances for the left and right profiles of the lowercases letters from the quantitative public tests. Balances are shown for all the fonts tested in the quantitative public tests, in each test condition for which data was collected.

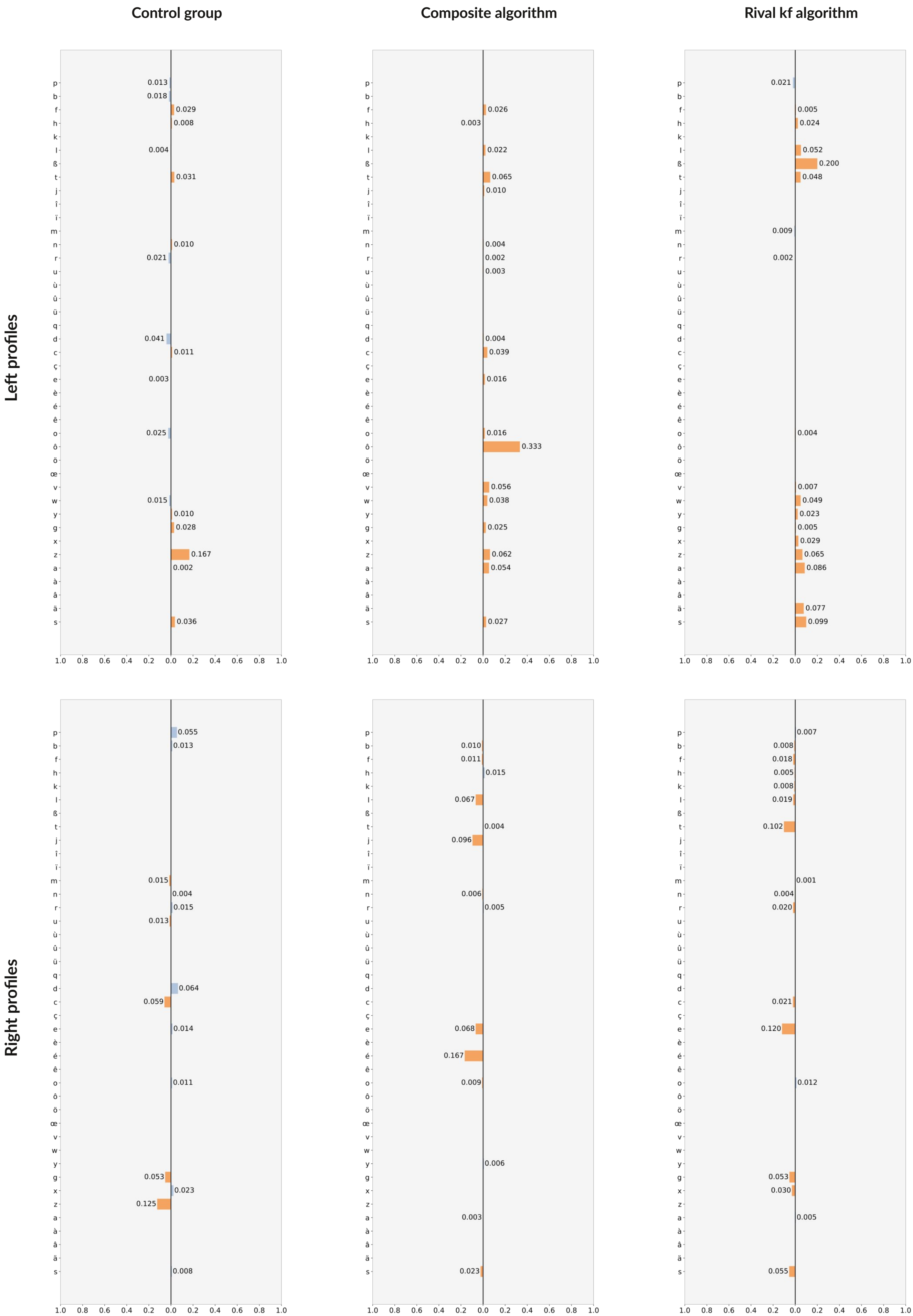
As described in chapter 6, § 6.3.1, each balance represents the difference between the "tight" and "loose" exposure mark rate for that profile, which is interpreted as capturing the overall bias of the rate at which marks were made on the profile in the text exposures.

All of the plots are indexed identically, sorted by left profile shape, for ease of comparing the left and right balances for each form. The orange colour bars represent a balance on the side of the "tight" mark class; blue colour bars represent a balance on the side of the "loose" mark class.

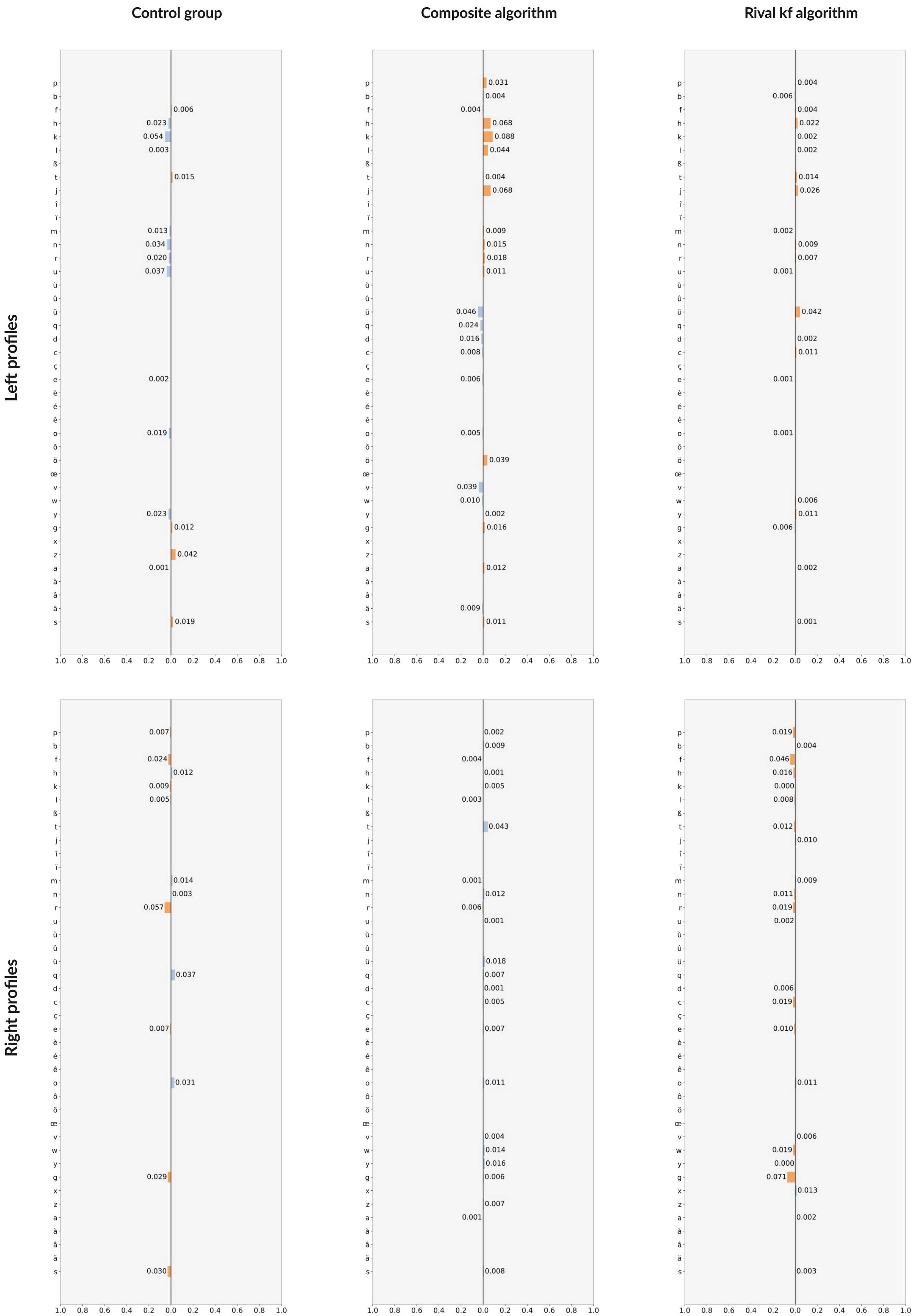
Alegreya Sans Regular: per-letterform profile mark balances, lowercase



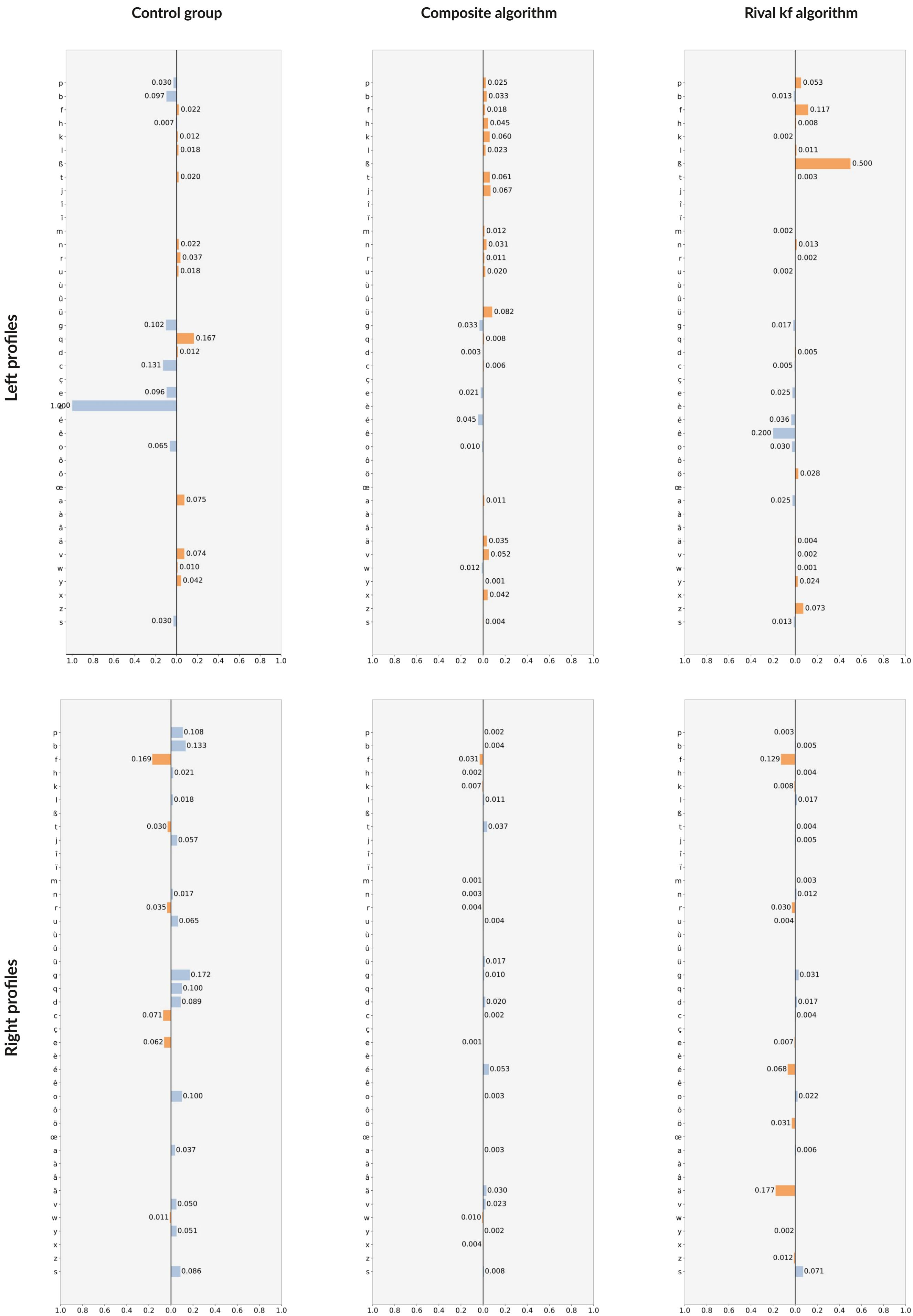
Fira Sans Condensed: per-letterform profile mark balances, lowercase

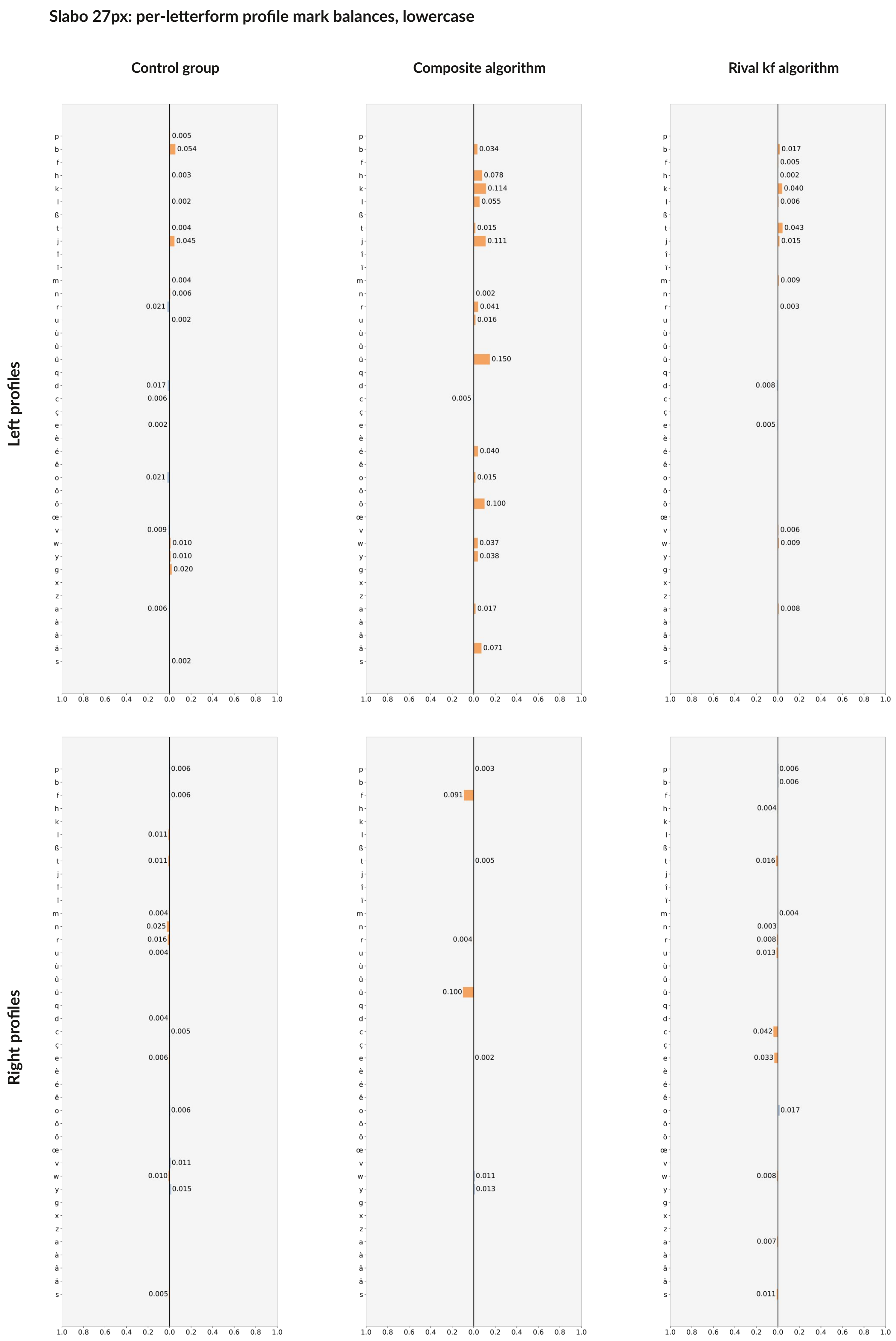


Literata Regular opsz 14: per-letterform profile mark balances, lowercase

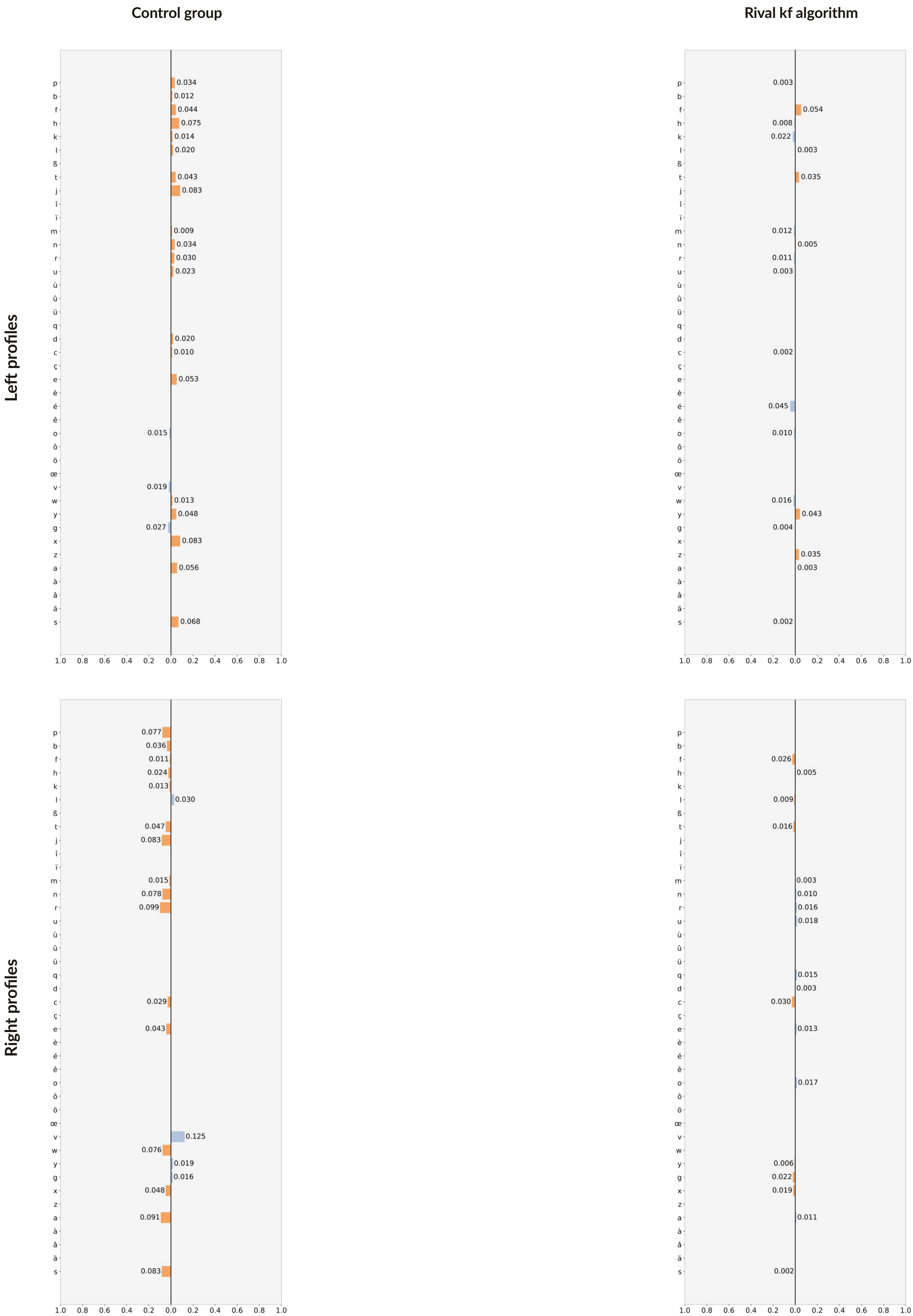


Slabo 13px: per-letterform profile mark balances, lowercase

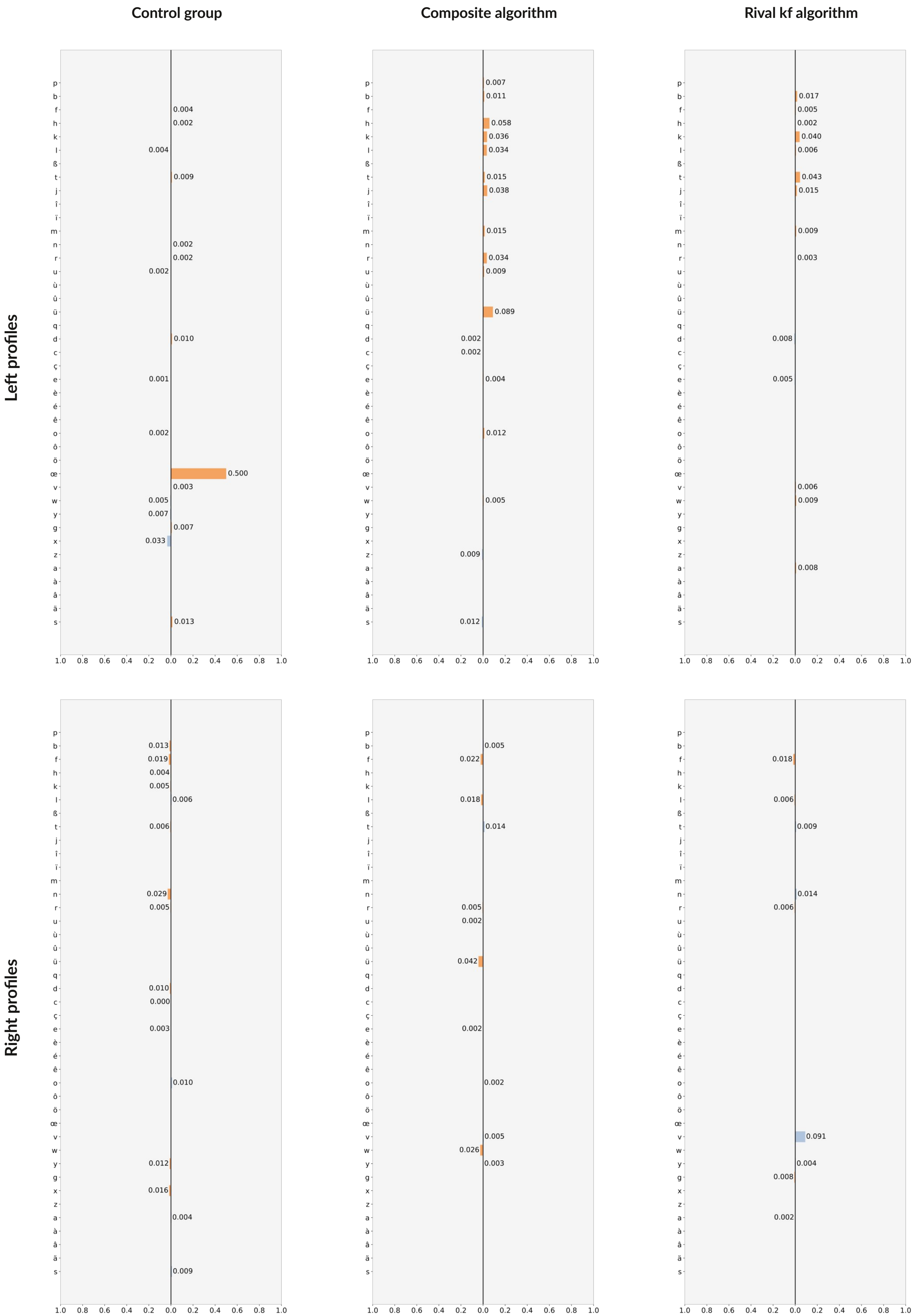



Slabo 27px: per-letterform profile mark balances, lowercase
Control group
Composite algorithm
Rival kf algorithm
Left profiles
Right profiles

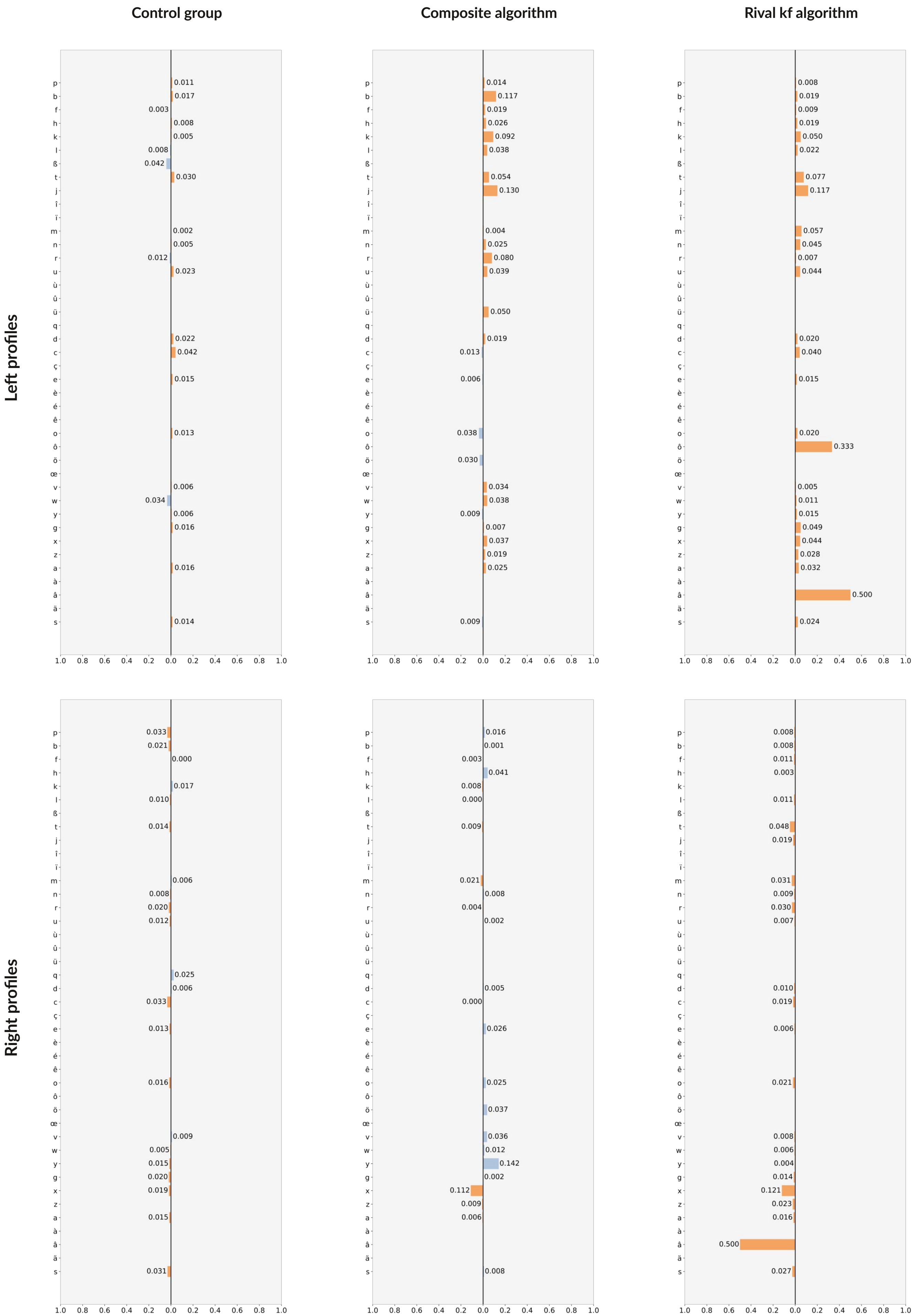
Source Sans Pro Regular: per-letterform profile mark balances, lowercase



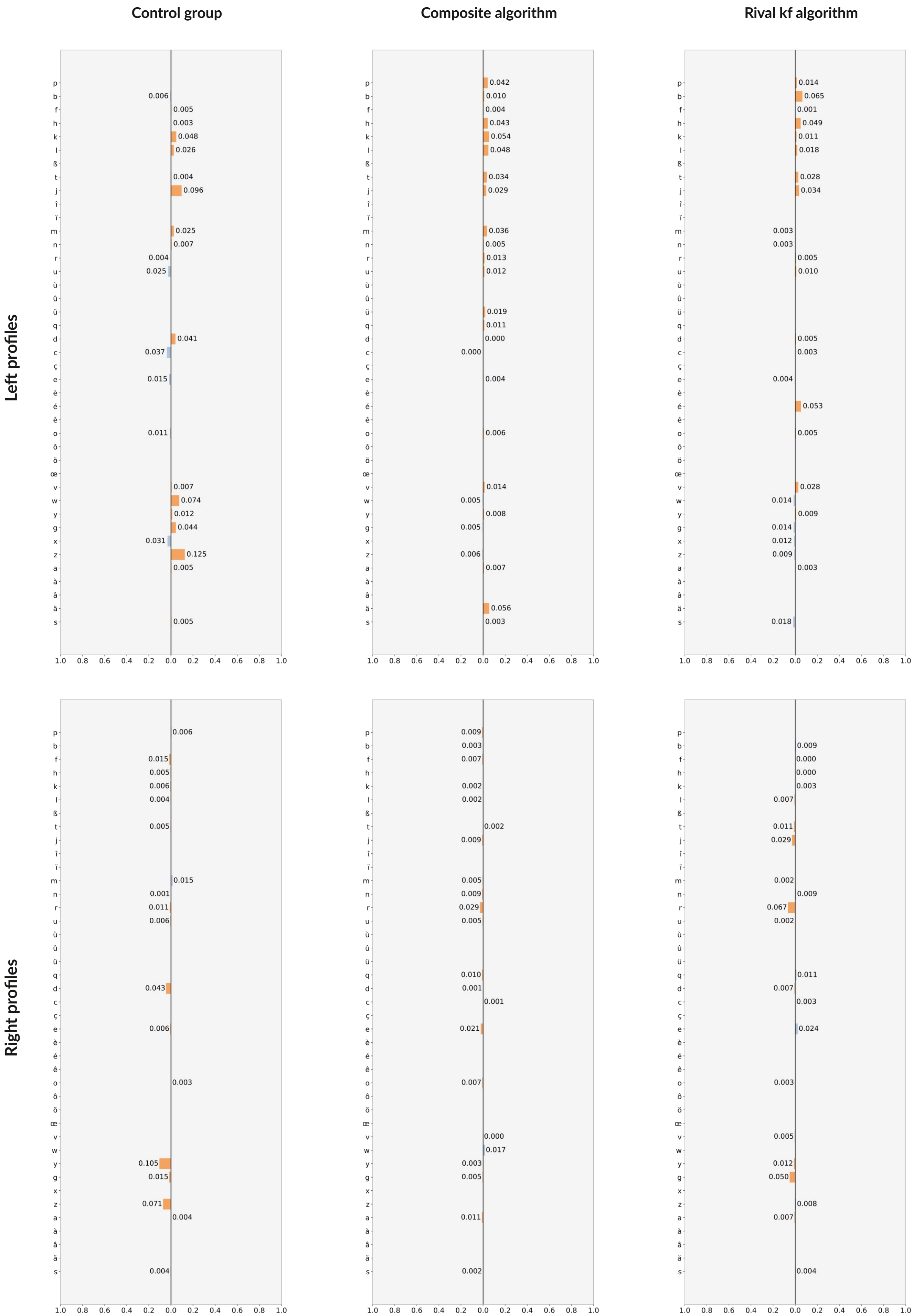
Source Serif 4 Regular: per-letterform profile mark balances, lowercase



STIX Two Text Regular: per-letterform profile mark balances, lowercase



Yrsa Regular: per-letterform profile mark balances, lowercase



Yrsa Bold: per-letterform profile mark balances, lowercase