Solar transient recognition using deep learning (STRUDL) for heliospheric imager data

Bauer, M. ORCID: https://orcid.org/0000-0002-2507-7616, Le Louëdec, J. ORCID: https://orcid.org/0000-0001-5387-9512, Amerstorfer, T. ORCID: https://orcid.org/0000-0001-9024-6706, Barnard, L. ORCID: https://orcid.org/0000-0001-9876-4612, Barnes, D. ORCID: https://orcid.org/0000-0003-1137-8220 and Lammer, H. (2025) Solar transient recognition using deep learning (STRUDL) for heliospheric imager data. Space Weather, 23 (9). e2025SW004561. ISSN 1542-7390 doi: 10.1029/2025sw004561 Available at https://centaur.reading.ac.uk/124518/

To link to this article DOI: http://dx.doi.org/10.1029/2025sw004561

Publisher: American Geophysical Union

Space Weather®

RESEARCH ARTICLE
10.1029/2025SW004561

Key Points:

- We developed a machine learning model to automatically detect and track coronal mass ejections in heliospheric images
- Automatic detection provides internally consistent results, helping to address some of the challenges of manual tracking
- The model performs well on clear events but struggles with faint or overlapping structures, highlighting areas for future work

Correspondence to:
M. Bauer,
maike.bauer@geosphere.at

Citation:
Bauer, M., Le Louëdec, J., Amerstorfer, T., Barnard, L., Barnes, D., & Lammer, H. (2025). Solar Transient Recognition Using Deep Learning (STRUDL) for heliospheric imager data. *Space Weather*, *23*, e2025SW004561. https://doi.org/10.1029/2025SW004561

Received 1 JUN 2025
Accepted 22 AUG 2025

Author Contributions:
Conceptualization: M. Bauer, J. Le Louëdec, T. Amerstorfer
Data curation: M. Bauer, J. Le Louëdec, L. Barnard
Formal analysis: M. Bauer
Funding acquisition: T. Amerstorfer
Investigation: M. Bauer, J. Le Louëdec
Methodology: M. Bauer, J. Le Louëdec, T. Amerstorfer
Project administration: T. Amerstorfer
Software: M. Bauer, J. Le Louëdec
Supervision: J. Le Louëdec, T. Amerstorfer, H. Lammer
Visualization: M. Bauer
Writing – original draft: M. Bauer, L. Barnard, D. Barnes
Writing – review & editing: M. Bauer, J. Le Louëdec, L. Barnard, D. Barnes, H. Lammer

Solar Transient Recognition Using Deep Learning (STRUDL) for Heliospheric Imager Data

M. Bauer[1,2], J. Le Louëdec[1], T. Amerstorfer[1], L. Barnard[3], D. Barnes[4], and H. Lammer[5]

[1]Austrian Space Weather Office, GeoSphere Austria, Graz, Austria, [2]Institute of Physics, University of Graz, Graz, Austria, [3]Department of Meteorology, University of Reading, Reading, UK, [4]RAL Space, STFC Rutherford Appleton Laboratory, Didcot, UK, [5]Austrian Academy of Sciences, Space Research Institute, Graz, Austria

Abstract Coronal Mass Ejections (CMEs) are space weather phenomena capable of causing significant disruptions to both space- and ground-based infrastructure. The timely and accurate detection and prediction of CMEs is a crucial steps toward implementing strategies to minimize the impacts of such events. CMEs are commonly observed using coronagraphs and heliospheric imagers (HIs), with some forecasting methods relying on manually tracking CMEs across successive images in order to provide an estimate of their arrival time and speed. This process is time-consuming and results may exhibiting considerable interpersonal variation. We investigate the application of machine learning (ML) techniques to the problem of automated CME detection, focusing on data from the HI instruments aboard the STEREO spacecraft. HI data facilitates the tracking of CMEs through interplanetary space, providing valuable information on their evolution. Building on advances in image segmentation, we present the Solar Transient Recognition Using Deep Learning (STRUDL) model. STRUDL is designed to automatically detect and segment CME fronts in HI data. We address the challenges inherent to this task and evaluate the model's performance across a range of solar activity conditions. To complement segmentation, we implement a basic tracking algorithm that links CME detections across successive frames, thus allowing us to automatically generate time-distance profiles. Our results demonstrate the feasibility of applying ML-based segmentation techniques to HI data, while highlighting areas for future improvement, particularly regarding the accurate segmentation and tracking of faint and interacting CMEs.

Plain Language Summary Coronal Mass Ejections (CMEs) are large bursts of solar material that can disrupt satellites, power grids, and communication systems on Earth. To prepare for these events, CMEs need to be detected and assessed quickly. This is often done by hand, but the interpretation of data is not always straightforward and results can vary between forecasters. In this study, we present a ML model called Solar Transient Recognition Using Deep Learning (STRUDL) to automatically detect and track CMEs in images from NASA's STEREO spacecraft. Our model can follow CMEs as they move away from the Sun, which helps us understand how they evolve over time. While our model works well for clear events, it struggles with faint or overlapping CMEs. This work demonstrates that ML can be a valuable tool for monitoring CMEs, although further improvements are necessary to effectively handle complex events.

1. Introduction

Coronal mass ejections (CMEs) are explosive eruptions of plasma and magnetic field from the solar corona. Their speed can range from 100 km/s at the lower end to over 2,500 km/s for fast CMEs (Manoharan & Mujiber Rahman, 2011; Yashiro et al., 2004). Depending on the CME's speed and magnetic field orientation, it may have a significant impact on planets and their magnetospheres. CMEs pose a potential risk to astronauts in space, satellites, and power grids on Earth, making identification and arrival forecasting of Earth-directed CMEs crucial for risk mitigation.

The study of CMEs dates back to the early 1970s, when space-based coronagraphs aboard the OSO-7 (Koomen et al., 1975) and Skylab (MacQueen et al., 1975) missions captured the first images of CMEs. The Solar Mass Ejection Imager (SMEI; Jackson et al., 1997), launched in 2003 aboard the Coriolis spacecraft, was designed to investigate CMEs as drivers of space weather, specifically to assess the viability of using image data to track CMEs and forecast their arrival at Earth (Webb et al., 2002). A major advancement in our understanding of CMEs came with the launch of the twin Solar TErrestrial RElations Observatory (STEREO; Kaiser et al., 2008) spacecraft in 2006. Consisting of STEREO-A and STEREO-B, the mission provided a stereoscopic view of the

Sun, facilitated by two coronagraphs and two wide-field heliospheric imager (HI; Eyles et al., 2009) instruments on board each spacecraft. The combined field-of-view (FOV) of the HI instruments extends from 4 to 88° of elongation from Sun-center, allowing for continuous observation of CMEs as they move away from the Sun. Since the start of the STEREO mission, more spacecraft equipped with HI instruments have been launched (e.g., PSP/WISPR, SoloHI, PUNCH; DeForest et al., 2022; Howard et al., 2020; Vourlidas et al., 2016). These missions further aid the analysis of CME structure and kinematics by providing complementary observations from different viewpoints.

To enable the study of CMEs and their properties, a number of catalogs utilizing different identification techniques and instruments have been published over the years. L. Barnard et al. (2014) introduced a catalog based on the Solar Stormwatch I citizen science initiative, where volunteers were asked to analyze data from the STEREO/HI instruments to identify and track CMEs. By aggregating multiple user observations, time-elongation profiles could be generated for each event. The HICAT catalog, first presented by Harrison et al. (2018) and developed under the EU FP7 HELCATS project, is another, more recent example. It comprises over 3,000 CMEs identified through manual inspection of STEREO/HI data. To account for ambiguity, each event is categorized as either good, fair, or poor in quality, depending on how certain the identification is. An event is classified as good if any person experienced in working with STEREO/HI data would consider it a CME. Conversely, an event is classified as poor if it is likely to be a CME, but there is considerable uncertainty due to factors such as brightness variation, topology, or the presence of data gaps. A fair quality event lies between the other two definitions, meaning that the CME is relatively clearly visible, but not completely unambiguous. The catalog was updated by Barnes et al. (2018) to include geometric parameters for a subset of the CMEs in HICAT. Furthermore, the updated catalog, termed HIGeoCAT, provides time-elongation profiles along the position angle (PA) that corresponds to the center of the angular extent of each CME. Both HICAT and HIGeoCAT start in April 2007, the beginning of the official science-phase for the SECCHI suite of instruments, and are continuously updated. All of these catalogs, while valuable, are time-consuming in their creation and not updated in real-time.

In recent years, several automated CME detection algorithms have been developed to address some of the issues that manual methods face. Many of these algorithms are meant for use with coronagraph data (e.g., CACTus, CAMEL; Robbrecht & Berghmans, 2004; Wang et al., 2019). Automated detection in HI data has received somewhat less widespread attention, likely due to the fact that CMEs are more difficult to identify in HI compared to coronagraph data. A CME catalog based on SMEI data was published by Tappin et al. (2012), though they reported a high number of false positive detections due to the noisy nature of the data. Barnard et al. (2015) developed the J-Tracker method, which relies on Canny edge detection to extract transients from STEREO/HI time-elongation maps, in which the CME is viewed along one particular PA. More recently, CACTus, which was originally developed to automatically detect CMEs in images from the C2 and C3 Large Angle Spectrometric Coronagraph (LASCO; Brueckner et al., 1995) instruments aboard the Solar and Heliospheric Observatory (SOHO; Domingo et al., 1995), has been adapted to work on time-elongation maps of STEREO/HI data (Pant et al., 2016). It utilizes image processing techniques to identify CME fronts based on predefined criteria, aiming to minimize the impact of human subjectivity on detection. Rodriguez et al. (2022) applied the modified CACTus algorithm to a range of years, spanning from April 2007 to August 2020, and found that, compared to the manually created HICAT catalog, CACTus tends to overestimate the number of CMEs, likely due to the algorithm frequently categorizing parts of the same front as separate CMEs.

Moving away from efforts focused on tracking CMEs in time-elongation maps, Kirnosov et al. (2016) developed an automated tracking algorithm that utilizes images from both the COR2 and HI1 instruments aboard STEREO-A. Their algorithm yielded satisfactory estimates of speed and PA for the 15 events under study, but has not been tested on a larger data set. An effort to manually track CME fronts across multiple PAs in a large number of STEREO/HI images was made by the Solar Stormwatch II project (SSW-II; Barnard et al., 2017). Designed as a follow-up study to Solar Stormwatch I, SSW-II was a citizen science project seeking to improve space weather forecasting by recruiting volunteers to track CMEs in STEREO/HI images. Citizen scientists analyzed a number of events that occurred in the 2011–2012 time period, thus generating a valuable source of data cataloging solar storms in STEREO/HI data. Barnard et al. (2017) posit that tracking in HI data directly may have several advantages compared to tracking done in time-elongation maps. These include a more complete picture of the complex structure of the front, as well as eliminating uncertainty in the time coordinate resulting from the averaging and interpolation conventionally applied to time-elongation maps.

The detection of CMEs in image data shares many similarities to problems encountered in other disciplines, such as biomedical imaging. Recent advancements in ML, particularly deep learning, offer promising alternatives for automating the detection of solar phenomena in observational data. In particular, convolutional neural networks (CNNs; LeCun et al., 2010) have demonstrated success in various image recognition tasks. Among these, the U-Net (Ronneberger et al., 2015) and its variants, such as the ResUNet++ (Jha et al., 2019), have proven effective in a variety of contexts (Cui et al., 2019; Fang et al., 2019; Wu et al., 2019). In recent years, ML techniques have started gaining increasing significance in the space sciences, yielding promising results across different tasks covering domains ranging from the solar surface (Jarolim et al., 2024), to interplanetary space (Rüdisser et al., 2022) all the way to the terrestrial atmosphere (Malik et al., 2023) and beyond (Johnson et al., 2020).

In this work, we present an automatic detection pipeline for CMEs based on a U-Net model, specifically tailored to segment CME fronts in STEREO/HI data, which we term Solar Transient Recognition Using Deep Learning (STRUDL). We also introduce an automatic tracking algorithm designed to work with the output that STRUDL provides. In Section 2 of this article, we describe the data products used to train, validate, and test STRUDL with emphasis on how we incorporated data from the SSW-II project into our model. We also give an overview of the methods to which we compare our model. In Section 3, we provide an in-depth description of our model architecture, as well as the post-processing of results and the techniques used to evaluate the model's performance. In Section 4, we summarize the results.

2. Data

We conducted our experiments utilizing data from the HI1 instrument aboard the STEREO-A spacecraft. The data set comprises a total of 13,876 1024 × 1024 images from the time period between January and August 2010, as well as January–May 2012. We identify a total of 258 CMEs in the given time frames. During these periods, the 11-year solar cycle emerged from a minimum and started moving toward its peak in early 2014. Choosing these time periods ensures that the data set covers non-interacting (more common around solar minimum) as well as interacting (more common around solar maximum) events (Rodríguez Gómez et al., 2020) while also avoiding the latter part of the cycle in which STEREO-A moved toward the back of the Sun, placing it in an increasingly unfavorable viewing position.

The STEREO/HI data are preprocessed to enhance the visibility of CMEs and suppress noise. The IDL SolarSoft routine `secchi_prep.pro` is commonly used for this purpose. In this work, we rely on the Python HI-processing suite (see Section 5), which applies the same methods as `secchi_prep.pro`. These include the masking of saturated columns, often caused by planets passing through the spacecraft's FOV, correction of image-smearing occurring due to the shutter-less nature of the imagers' cameras, and the application of flat-field, distortion, and vignetting calibrations. To enhance CME visibility, we make use of running difference images. These data products are created by subtracting the previous frame from the current one, yielding a final image in which the dynamic structure of the CME are amplified compared to the static background.

The lack of a ML-ready data set containing STEREO/HI data required manual annotation of images from the chosen time period. For the 2010 time period, the CMEs appearance in HI was identified according to the HICAT catalog where applicable, and the perceived front was annotated in each image by expert annotators. A 30-pixel wide region was chosen to mark the area of interest, as shown in Figure 1. The marked region later undergoes two iterations of morphological dilation with a disk-shaped kernel of size 2 to better capture the uncertainty inherent to the task. The combination of linewidth and post-processing approach was selected based on visual inspection across the data set and was found to consistently encompass the area of interest along the CME front.

For the 2011 time period, the data collected by the SSW-II project served as a starting point for the annotation. The objective of SSW-II was to track the leading edge of CMEs propagating through STEREO-A and STEREO-B HI1 images. For the period of 2011–2012, running difference images of CMEs, referred to as "assets," were produced to create the data set that the citizen scientists would use to characterize the CMEs. The HICAT catalog was used to identify time windows when a CME would likely be in either the HI1A or HI1B field of view. For the 2011–2012 period, there are 307 entries in the HICAT catalog. The assets were presented as "subjects," where a

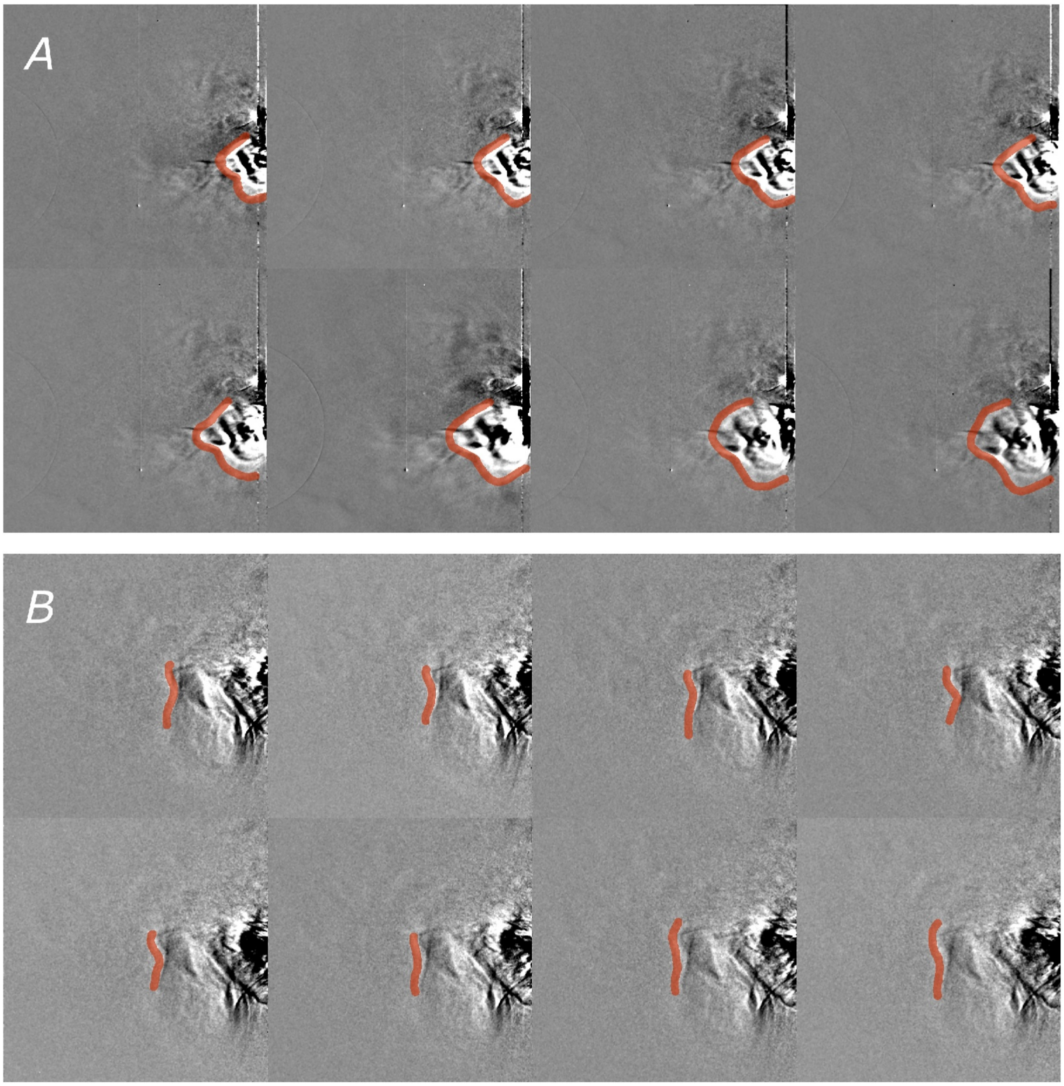

Figure 1. Sequences of STEREO-A HI1 images from January 2010 (a) and May 2010 (b). The data have been post-processed and subsequently reprocessed into running difference images. The manually annotated Coronal Mass Ejection front is overlaid in red.

subject refers to a grouping of three consecutive images. A training exercise was provided to demonstrate to the citizen scientists a range of appearances that CME fronts can take in HI1 data. Participants would then draw up to seven polygons on the three assets in each subject, tracking any feature they identify as a CME front. Analyzing three images at a time allowed participants to gauge the evolution of the CME between frames and was found to aid CME identification. Subjects were constructed so that they overlapped the previous/next subject by one asset, which enabled the continuous tracking of CME fronts. In total, this required 6,672 subjects, which were analyzed by 9,107 participants, resulting in 207218 classifications of CME fronts.

To incorporate SSW-II data into our data set, the various annotations by different participants must be combined into a single mask for each image. To reach a consensus, a series of processing steps is applied to the annotations of each image. As a first step, all annotations are summed up, followed by the application of a Gaussian filter. A Sato filter is used to extract curves from the summed masks, and the final binary mask is obtained by applying a threshold. Regions of interest are extracted from the binary mask and subsequently separated into distinct CMEs with separate labels. The processing steps were determined through iterative experimentation guided by visual inspection; examples of the masks before and after undergoing processing can be seen in Figure 2.

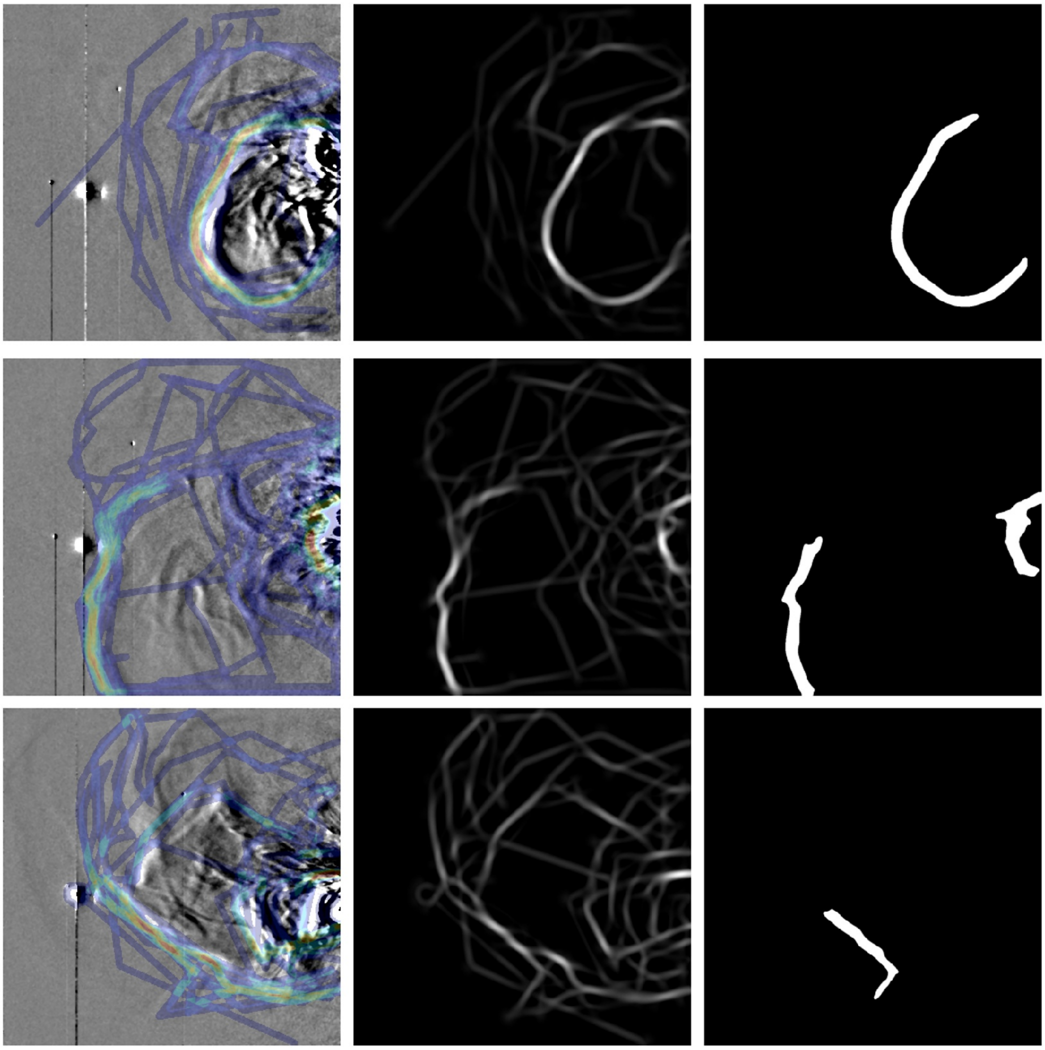

Figure 2. Example of post-processing steps to obtain consensus masks from the Solar Stormwatch II citizen science project, which we use as a source of additional data to train, validate, and test Solar Transient Recognition Using Deep Learning. The leftmost column shows the summed annotations plotted over the corresponding STEREO/heliospheric imager running difference image. The middle column shows the same annotations after application of a Gaussian filter with $\sigma = 8$, followed by a Sato filter with $\sigma \in [1, 10]$. In the rightmost column, the final binary consensus masks, obtained after normalizing the image and applying a threshold $t = 0.3$, are shown.

3. Methods

3.1. Semantic Segmentation

The model described in the following section was implemented using the PyTorch framework, an open-source Python library. For access to the code used in this paper, as well as to the pre-trained models, please refer to Section 5.

We compile our data into non-overlapping sequences, each containing a consecutive series of 16 images. We aim to maintain a consistent ratio of CME to non-CME images in each sequence, although the ratio may be slightly higher or lower at times due to the constraint that sequences must not overlap. We use a stride of two when generating the input sequences, resulting in each frame appearing in multiple sequences. For the segmentation task, we employ a simple U-Net, adapted to work on three-dimensional data, the third dimension being time.

An overview of our model architecture is given in Figure 3. Our model consists of two separate paths, referred to as encoder and decoder. Given that our model is based on a U-Net architecture, these paths are symmetrical. The

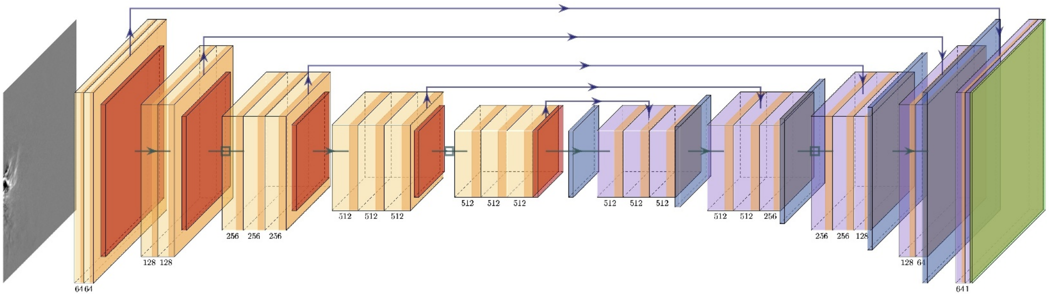

Figure 3. Schematic illustration of our 3D UNet. The diagram illustrates the sequence of operations applied to the input data as it propagates through the network. The left-hand side of the network corresponds to the encoder path, the right-hand side to the decoder path. Arrows indicate the direction of the flow of the data. Some arrows are marked with a square, showing the presence of a dropout layer. Light orange blocks represent convolutions, while darker orange bands represent the rectified linear unit activation function. The numbers below the convolutions indicate the size of the feature space after the convolution operation. Red and blue blocks indicate 3D max pooling and unpooling operations, respectively. Purple blocks indicate transpose convolutions. The terminal layer (green) normalizes the network's output between 0 and 1 using a sigmoid function. The purple arrows between the encoder and decoder paths of the network show the skip connections.

model's encoder path consists of five separate stages, each containing either two or three three-dimensional convolutional layers. Convolutional layers are the basic building blocks of a CNN and are designed to extract relevant features from the data. Each convolutional layer is followed by a so-called activation function. These functions transform their input non-linearly to allow the model to learn complex patterns. We chose the commonly used rectified linear unit (ReLU) activation function, given by $f(x) = \max(0, x)$, for this purpose.

After passing through a group of convolutional layers along the encoder path, the output is spatially downsampled using three-dimensional max pooling layers, preserving important features while reducing the amount of time needed to process the data in subsequent layers. Conversely, our model's decoder reconstructs the data's original spatial resolution by progressively upsampling the feature maps using three-dimensional max unpooling layers. To obtain output data with the same shape as the input data, we apply transpose three-dimensional convolutions along the decoder path, effectively performing the inverse operations of the convolutional layers in the encoder path. Finally, we apply a sigmoid layer to our output. The sigmoid layer ensures that every pixel in our segmentation mask has a value between 0 and 1, giving us a probability that the pixel is part of a CME. The encoder and decoder stages of the model are linked by so-called skip-connections. These connections allow the output of a particular encoder level to be fed directly to the corresponding decoder level, instead of having to traverse the entirety of the network. This helps information to propagate more freely throughout the network and can improve segmentation results.

The model outputs segmentation masks for each individual frame within a sequence, providing pixel-wise labeling of CME fronts. We post-process the segmentation masks to fill small holes within continuous fronts and remove small objects likely to represent spurious detections. To improve the robustness of the model and reduce the risk of overfitting to the training data, we apply simple data augmentation techniques. Specifically, we randomly flip input images horizontally and vertically during training. This encourages the model to learn features that are invariant to orientation, thereby improving its ability to generalize to new, unseen data. Such augmentation techniques are widely used in computer vision tasks, and their value for learning invariance has been demonstrated numerous times (e.g., Mikołajczyk & Grochowski, 2018). Furthermore, we include dropout layers in our model. These layers are designed to randomly ignore, or drop, parts of the input during training, effectively making the model more robust and further reducing the risk of overfitting to the training data.

To update our model's parameters during training and guide it toward convergence, an optimization algorithm must be chosen. We use the Adam optimizer with an initial learning rate of 1×10^{-5}. Adam is a widely used, efficient method that adaptively adjusts the learning rate for each parameter individually, rather than relying on a single global learning rate. Training runs for 150 epochs, but only the model with the highest Intersection-Over-Union (IoU) score on the validation set is saved to prevent overfitting. The IoU score will be explained in detail in Section 3.3.

We have to consider that pixels belonging to a CME represent only a small fraction of pixels in each image, if a CME is present at all, while background pixels dominate. This imbalance could cause the model to prioritize correctly identifying the background pixels over identifying the CME itself. To mitigate the impact of this issue, we select a loss function for training our model that is specifically designed for imbalanced data sets such as ours. In ML, the loss function measures how well a prediction matches the ground truth data, guiding the model's learning process. Binary Cross Entropy (BCE) is a loss function that's commonly used for classification tasks. It computes the loss for every pixel based on how different its predicted probability is from its true label. The larger the difference between the prediction and the true label, the larger the BCE loss for that pixel becomes. The Dice loss, on the other hand, focuses on the overlap between predicted and ground truth regions, making it useful for segmentation tasks. During training, we employ the asymmetric unified focal loss proposed by Yeung et al. (2022), which builds on both the well-established BCE and Dice losses while introducing additional terms to emphasize the importance of CME pixels. The model is trained on an NVIDIA GTX 4090 (24 GB) GPU with a batch size of 4, and training batches are randomly shuffled at the start of each epoch.

3.2. Tracking

To extract individual CME regions from the final segmentation images and track them across frames, we use algorithms implemented in Python's scikit-image library. First, we segment the image into distinct regions using the connected component labeling algorithm, following the methods outlined in Fiorio and Gustedt (1996) and Wu et al. (2005). For tracking purposes, we extract the CME front by applying the skeletonize algorithm (Zhang & Suen, 1984) to the segmented area, yielding a line of pixel coordinates. We transform the pixel coordinates into world coordinates, which are the heliocentric latitude and longitude in the case of STEREO/HI observations. We use the equations found in Thompson (2006) to transform these coordinates into elongation ϵ and PA, as these are commonly used when tracking CMEs in HI data.

We then employ a multi-step algorithm designed to associate CME fronts across successive images. An overview of the tracking algorithm, presented in pseudo-code, is provided in Algorithm 1. First, all CME fronts detected within a single image are compared to each other. If their mean elongation values differ by less than 3°, they are assumed to belong to the same front and are merged. This threshold was chosen based on the effects of our segmentation pre-processing. Specifically, we apply morphological dilation to the masks to ensure that CME fronts are fully captured and not too narrowly defined. As a result, segmented regions can span several degrees in elongation, even when the actual CME front is more localized. The 3° threshold accounts for this spatial spread, and ensures that fragmented detections of the same structure within a frame are grouped before further analysis. Next, each front is evaluated to determine whether it could represent the start of a new CME. This is based on its mean ϵ. If a front appears at an elongation less than 7.3°, it is considered a potential new CME. If it appears at a greater distance, it is not considered a potential starting point, but is kept for further analysis. Once candidate CME fronts are identified in the current image, we attempt to associate them with CMEs in the previous frame, if there are any. Each front in the current image is compared to all fronts in the previous image using the difference in mean elongation $\Delta\epsilon$ at each PA. If $\Delta\epsilon$ lies within the threshold of $-3.2°$ to $+6.7°$, the fronts are considered to belong to the same CME.

If a front in the current image is not associated with any front from the previous frame but was flagged as a potential new CME, it is treated as the beginning of a new CME. Otherwise, it is discarded. In cases where the current image contains no CME candidates but the previous one did, we interpret this as the end of the tracked CME. If neither image contains any CME candidates, no action is necessary. To avoid large temporal gaps between images, the tracking algorithm is only applied when the time interval between two consecutive frames is less than 140 min. We discard CME tracks with maximum elongations below 12.9°, as such detections are unlikely to represent true CME events. The thresholds for identifying new fronts, associating them across successive images, and determining whether a tracked structure qualifies as a CME were established empirically. To define reasonable baseline values, we analyzed 255 CMEs listed in the HIGeoCAT catalog for the years 2021 and 2022. These years were chosen because they represent a range of solar conditions, and don't intersect with any of the data in our training, test, or validation sets.

We determined the 1st percentile of the maximum elongation reached to set the minimum elongation criterion for a CME, yielding 12.9°. To avoid misidentifying background features as CME origins, we used the 99th percentile of starting elongation values to define an upper bound of 7.3° for new CME fronts. To derive the thresholds used for associating fronts between images, we evaluated the range of elongation differences between time steps for each of the 255 events. The 99th percentile of maximum elongation change was 3.7°, and the 1st percentile of minimum change was −0.2°. Because our fronts represent broad structures rather than sharply defined edges, and we rely on mean elongation differences to compare them across images, we added a buffer of 3° to these percentiles. This adjustment results in final association thresholds of −3.2° to +6.7°.

Algorithm 1. Pseudo-code outlining the basic principles of our CME tracking algorithm.

```
Require: $t_i - t_{i-1} < 140$
function AGGREGATE FRONTS ($front_i$)
    for all $front_i^n$ in $image_i$ do
        for all $front_i^m$ in $image_i$ do
            if |mean($\epsilon_i^n$)–mean($\epsilon_i^m$)| < 3 then
                $front_i^n$ and $front_i^m$ belong to same CME
            end if
        end for
    end for
end function
function IDENTIFY NEW CME ($front_i^n$)
    if mean($\epsilon_i^n$) < 7.3 then
        $front_i^n$ is potential start of new CME
    else
        $front_i^n$ is not start of new CME
    end if
end function
AGGREGATE FRONTS($front_i$)                      ▷ Combine fronts in same image
for all $front_i^n$ in $image_i$ do              ▷ Check which front could be start of new CME
    IDENTIFY NEW CME($front_i^n$)
end for
if count($front_i$) > 0 and count($front_{i-1}$) > 0 then ▷ Check association with CMEs in previous image
    for all $front_i^n$ in $image_i$ do
        for all $front_{i-1}^k$ in $image_{i-1}$ do
            $\Delta\epsilon$ = mean($\epsilon_i^n$)–mean($\epsilon_{i-1}^k$)
            if $-3.2 < = \Delta\epsilon < = 6.7$ then
                $front_i^n$ and $front_{i-1}^k$ belong to same CME     ▷ Min. $\Delta\epsilon$ and $\Delta PA$ are associated
            end if
        end for
        if $front_i^n$ not linked to previous CME, and flagged as new CME then
            $front_i^n$ is new CME
        else
            $front_i^n$ is discarded
        end if
    end for
else if count($front_i$) = 0 and count($front_{i-1}$) > 0 then
    CME has ended
else
    No CME present
end if
```

Table 1
Distribution of STEREO/Heliospheric Imager Images Containing/Not Containing Coronal Mass Ejections Across the Training, Test, and Validation Sets for All Five Models Evaluated as Part of the 5-Fold Cross-Validation

Model no.	Class	Training	Test	Validation
Model_1	CME	6,068	1,858	537
	No CME	2,718	787	435
Model_2	CME	6,039	1,465	959
	No CME	2,944	766	230
Model_3	CME	5,996	1,503	964
	No CME	2,728	635	577
Model_4	CME	5,740	1,711	1,012
	No CME	2,505	961	474
Model_5	CME	5,501	1,926	1,036
	No CME	2,871	791	278

Note. We attempt to maintain a similar ratio of CME to non-CME images within each set for all models.

3.3. Evaluation

Accurately evaluating ML models is crucial for determining their effectiveness and utility. Furthermore, it is important to select metrics that are both representative and suitable for the specific task at hand. We aim to evaluate our model's capabilities for CME segmentation and tracking.

For the segmentation task, the STEREO/HI image sequences are randomly distributed between data sets, allocating 70% of the data to the training, 20% to the test, and 10% to the validation set. All images belonging to the same CME are always part of the same set to prevent leakage between them. To provide a more reliable estimate of the model's generalization ability, we perform 5-fold cross-validation. For cross-validation, the data is divided into five equally sized chunks, so-called folds. The model is then trained on four folds, with a percentage of the training data withheld for validation, while one fold is reserved for testing. This process is repeated five times, using different folds for the training and test sets, yielding five trained models. The final performance metrics are calculated by averaging the results on the test data set across all five models. On average, our training set contains 5,869 images with CMEs, while the validation and test sets consist of 902 and 1693, respectively. The distributions of CME and non-CME images for each of the five models can be found in Table 1.

To evaluate our model's performance on the segmentation task, we compare the predicted masks to the ground truth on a per-pixel basis, making the definitions of true positives (TPs), false positives (FPs), true negatives (TNs), and false negatives (FNs) straightforward. A pixel classified as part of a CME by the model is a true positive if it also belongs to a CME in the ground truth. If the pixel is instead classified as a CME when it is part of the background, it is a false positive. Conversely, a true negative occurs when a pixel is correctly identified as background, while a false negative arises when a pixel belonging to a CME is misclassified as background. Since the model sees each image multiple times, and thus produces multiple predictions for the same image, the resulting masks must be aggregated to obtain a single final prediction for each image. To derive the final mask, we take the mean, median, and maximum across all predictions for each image. We compare these three aggregation methods to determine the most suitable one for the task. To evaluate segmentation performance, we compute the IoU, precision, recall, and Dice score, which are defined as follows:

1. Recall: measures how many of the true positives in the data set the model managed to identify. It ranges from 0 to 1, with a score of 1 indicating that the model has correctly classified all positive samples in the data set. Recall is defined as $\text{Recall} = \frac{\text{TP}}{\text{TP}+\text{FN}}$.
2. Precision: measures the model's ability to correctly identify positive instances; in other words, the quality of a positive prediction. Precision also ranges from 0 to 1, and is usually anti-correlated with recall. A precision of 1 would indicate that every positive prediction corresponds to a true positive sample. Precision is defined as $\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$.
3. Dice coefficient: also known as F1 score. Measures the overlap of the predicted mask and the ground truth. The Dice coefficient is defined as $\text{Dice} = \frac{2\text{TP}}{2\text{TP}+\text{FP}+\text{FN}}$.
4. Intersection-over-Union (IoU): also known as Jaccard index. It measures the overlap between a predicted mask and its ground truth, and is thus very similar to the Dice coefficient. The IoU penalizes over- and under-segmentation more than the Dice coefficient. The IoU score is defined as $\text{IoU} = \frac{\text{TP}}{\text{TP}+\text{FP}+\text{FN}}$.

To evaluate tracking performance, we employ two different approaches: event-based evaluation and continuous evaluation. For both methods of tracking evaluation, we compute precision and recall, as well as the mean of the absolute error of the tracks' start and end times, given by:

$$\Delta t_{\text{start}} = \left| t_{\text{start}}^{\text{groundtruth}} - t_{\text{start}}^{\text{predicted}} \right|$$

$$\Delta t_{\text{end}} = \left| t_{\text{end}}^{\text{groundtruth}} - t_{\text{end}}^{\text{predicted}} \right|$$

For event-based evaluation, we assess the model's ability to track CMEs by comparing the predicted tracks to the tracks from the ground truth annotations. Event-based evaluation utilizes the same models as the segmentation task and follows the same cross-validation procedure, meaning that the final scores are averaged across all five models.

For this mode of evaluation, CMEs are counted as true positives if they can be matched with a CME in the ground truth. To be considered a potential match, the CME must exhibit both a sufficient temporal and spatial overlap.

Sufficient temporal overlap is defined as:

$$t_{\text{overlap}} = \frac{\text{overlap}}{\text{duration}} > 0.25$$

where "overlap" is the shared time between both CMEs, and "duration" is the total duration of both CMEs combined. Sufficient spatial overlap is given if $\text{IoU} > 0.1$.

Each predicted CME can only be matched with at most one CME from the ground truth, and vice versa. It is possible for a CME from either data set to have no match in the other. To enforce this, only the predicted CME with the smallest absolute time error is associated with a given ground truth CME. The absolute time error is given by:

$$\Delta t_{\text{total}} = \Delta t_{\text{start}} + \Delta t_{\text{end}}$$

A CME is considered a false positive if it cannot be associated with any CME in the ground truth. Conversely, a false negative occurs when a ground truth CME cannot be associated with any predicted CME.

For continuous tracking evaluation, we aim to assess the model's performance in a more realistic scenario, where it processes a continuous data stream rather than isolated event sequences. To facilitate this, we train an additional model identical to the previous ones in architecture, but with a different data split. Specifically, this model is trained on 90% of the available data and validated on the remaining 10%, without setting aside a dedicated test data set. This model is then applied to a continuous 2-year time series of images and compared against CME tracks from the HIGeoCAT catalog.

To capture different solar activity conditions, we select the years 2009 and 2011 for continuous tracking evaluation, corresponding to solar minimum and the rising phase of solar cycle 24, respectively. While the official solar maximum of cycle 24 occurred in late 2014, STEREO-A was approaching superior solar conjunction at that time, making data interpretation more challenging due to increasing observational gaps and reduced data quality. As a compromise between solar activity levels and spacecraft viewing geometry, 2011 was chosen to represent an active period with favorable observational conditions.

The definitions of TPs, FPs, and FNs closely follow that of event-based evaluation, with HELCATS tracks being used as the ground truth. Some modifications are necessary in the definition of a potential match, as there are no ground truth segmentation masks to compare to (and thus no IoU score to compute) and HELCATS tracks are only defined along a single PA, meaning that we can't rely on spatial overlap of fronts as a criterion. Temporal overlap is not affected by this, but the definition of spatial overlap must be changed to reflect these factors. We define the spatial overlap as the mean difference in ϵ along each PA for the predicted and ground truth CME fronts:

$$\Delta\epsilon = \text{mean}\left(\left|\epsilon_{\text{pa}}^{\text{groundtruth}} - \epsilon_{\text{pa}}^{\text{predicted}}\right|\right).$$

To ensure that no CME in the predicted data set is matched to more than one event in the ground truth data set, and vice versa, we assign the TP label for a given ground truth CME to the predicted CME with the smallest combined Δt_{total} and $\Delta\epsilon$. The definition of FPs and FNs remains analogous to that introduced for the event-based tracking evaluation.

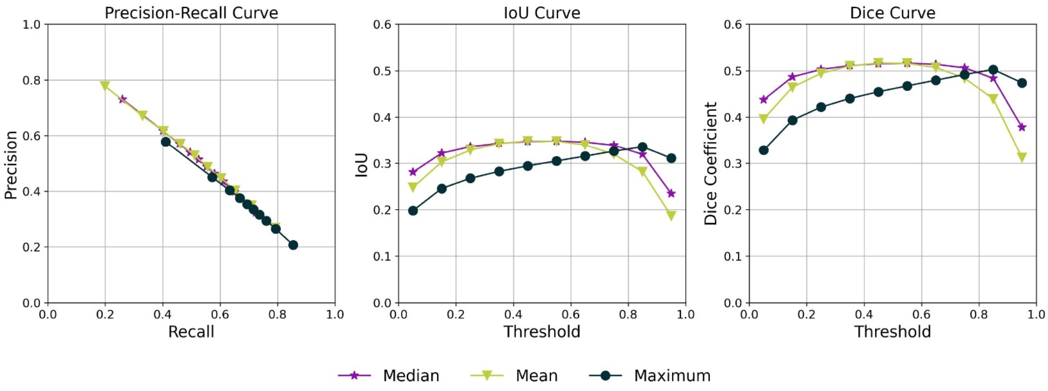

Figure 4. Key metrics for segmentation performance achieved using different aggregation methods (colored), averaged for all 5 models. The leftmost panel shows precision plotted against recall. The middle panel and rightmost panel display the IoU and Dice Score, respectively, at various binarization thresholds ranging from 0.05 to 0.95.

4. Results

4.1. Segmentation

Key segmentation performance metrics are presented in Figure 4 for different aggregation methods, evaluated across binarization thresholds ranging from 0.05 to 0.95. The results indicate a very similar overall performance across aggregation methods and thresholds. As expected, the maximum method performs better in terms of Dice score and IoU at higher binarization thresholds compared to the mean and median methods. Visually, the precision-recall curves presented in the leftmost panel of Figure 4 look alike for the mean and median methods, while the range values for both precision and recall across different thresholds are considerably smaller for the maximum method.

We identify the optimal threshold, as defined by the highest IoU, for each aggregation method and summarize the metrics at this threshold in Table 2. While the optimal threshold varies between methods, with threshold (median) = 0.55, threshold (mean) = 0.45, and threshold (max) = 0.85, the resulting metrics don't exhibit significant variations. The best IoU achieved varies between 0.34 for the maximum method and 0.35 for the median method and mean methods. The Dice score exhibits similar variation, with both the mean and median methods scoring at 0.52, while the maximum method achieves a score of 0.50. The only notable differences lie with the precision and recall scores. In terms of precision, the maximum method performs worse than the mean and median methods with a score of 0.45, while the other two methods achieve scores of 0.49 and 0.51, respectively. Since precision and recall are usually anti-correlated, the opposite is true for the recall metric, with the maximum method scoring 0.57, while the mean and median methods achieve scores of 0.56 and 0.53, respectively. We find that our model is largely insensitive to the choice of aggregation method and threshold, and select the mean method at threshold (mean) = 0.45 as the optimal method-threshold combination since it reaches the highest combined metrics out of all method-threshold combinations.

Figure 5 shows example segmentation results for post-processing using threshold(mean) = 0.45. The upper panel showcases a well-segmented CME, located at the beginning of the HI1 field of view. CMEs close to the Sun tend to be brighter and more easily distinguishable from the background. The model captures most of the front with only minor discrepancies at the edges compared to the ground truth. This example also highlights that an inherent challenge in CME segmentation lies in defining the precise boundary of a CME front, which is subjective. Even minor deviations of the predicted from the ground truth mask negatively impact performance metrics despite the segmentation being visually accurate. The lower panel of Figure 5 presents a more challenging case. There are two CMEs in the image, with one being further away and thus slightly fainter than the other CME. Our model accurately segments the front of the CME closest to the Sun, but struggles to capture the complete front of the preceding CME as it moves outwards.

Table 2
Key Metrics for Different Aggregation Methods, Averaged Across All Five Models, Given at the Binarization Threshold Value That Results in the Highest IoU for the Respective Method

Method	Threshold	Precision	Recall	IoU	Dice
Median	0.55	0.51	0.53	0.35	0.52
Mean	0.45	0.49	0.56	0.35	0.52
Maximum	0.85	0.45	0.57	0.34	0.50

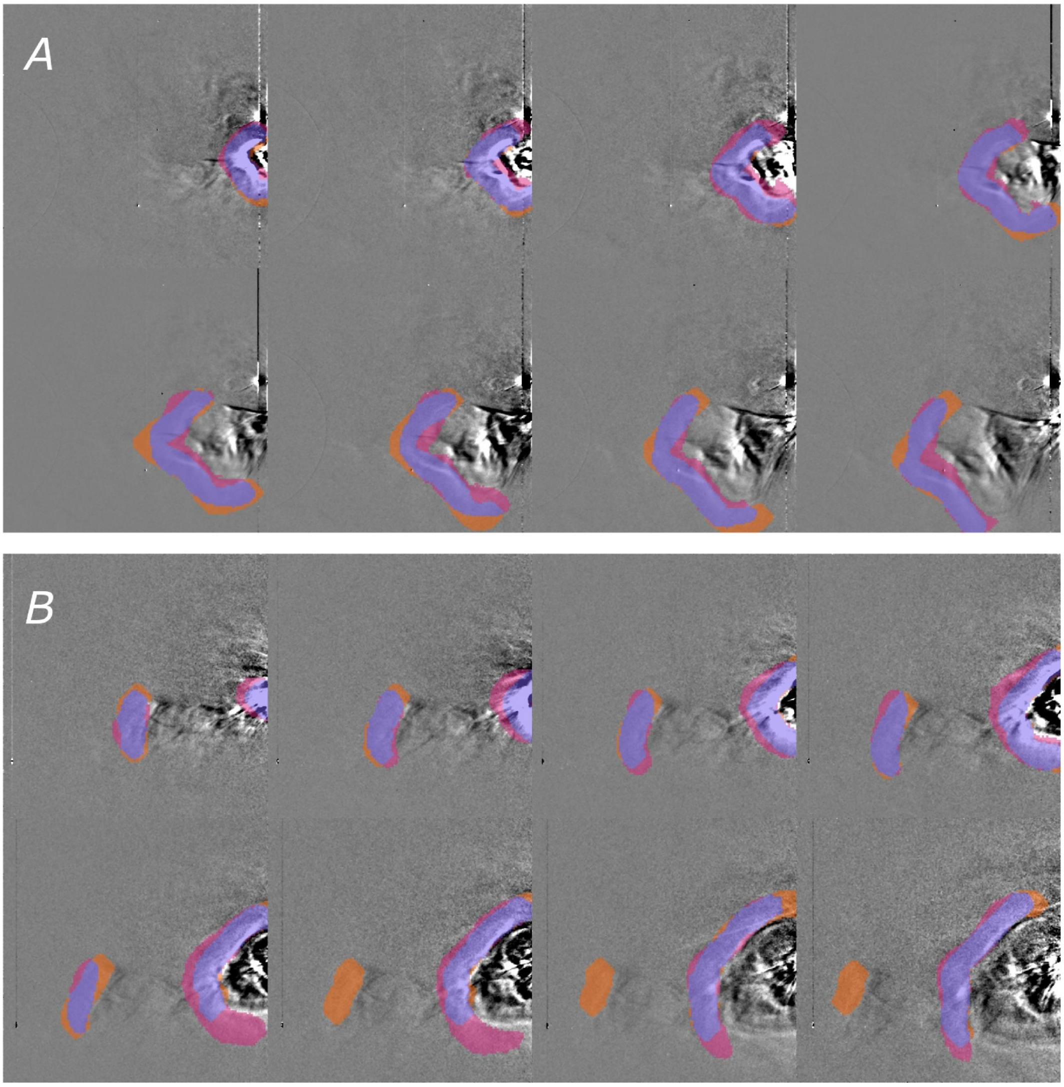

Figure 5. Sequences of images showcasing the model's segmentation performance using the best method-threshold combination threshold(mean) = 0.45. The overlap between the predicted front and the ground truth mask is shown. Each pixel belonging to the front is classified as a TP (purple), FP (pink), or FN (orange). Panel (a) shows a single Coronal Mass Ejection (CME) front close to the Sun, while panel (b) shows two CME fronts, one being further away from the Sun and thus fainter than the other one.

4.2. Event-Based Tracking

Table 3 summarizes the tracking performance for event-based evaluation, analyzed using the optimal method-threshold combination of threshold(mean) = 0.45. Compared to the ground truth tracks, the model achieves an average Δt of 1.42 hr for CME start times and 4.71 hr for CME end times. This difference in results between the average Δt_{start} and Δt_{end} reflects the model's difficulty in tracking faint CME fronts, which often causes the track to be cut short. The model successfully recovers 145 of the 257 ground truth tracks, leaving 112 unmatched, and additionally identifies 23 tracks not present in the ground truth catalog. This corresponds to a precision of 0.87 and a recall of 0.56. The results are consistent across all five models, with standard deviations of 0.41 and 0.76 hr for the Δt_{start} and Δt_{end}, respectively. Precision and recall vary by 0.06, and 0.05 respectively, indicating stable performance and good generalizability across different data splits.

Figure 6 provides qualitative examples of the tracking algorithm's performance. The upper panel shows a case where the model successfully detects and tracks a bright CME front relatively close to the Sun, maintaining good performance throughout the sequence. In contrast, the lower panel depicts a CME that is slightly fainter, with

Table 3
Summary of Key Metrics Used to Gauge Event-Based Tracking Performance

Metric	Result
Δt_{start}	1.42 hr
Δt_{end}	4.71 hr
Precision	0.87
Recall	0.56
Total CMEs	257
True Positives	145
False Positives	23
False Negatives	112

Note. The best method-threshold combination threshold(mean) = 0.45 was used to generate the results. Results for average absolute start (Δt_{start}) and end (Δt_{end}) time error, as well as precision, and recall, were averaged across all 5 models. Results for true positives, false positives, and false negatives were summed up instead to allow for easier comparison to the total number of ground truth CMEs in the given time frames.

tracking performance gradually degrading as the structure propagates outward. These examples highlight both the model's strengths in identifying clear, well-defined CME fronts and its limitations when dealing with weaker, more ambiguous signals.

4.3. Continuous Tracking

Table 4 summarizes the results of continuous tracking for both years under study. As in the case of event-based tracking, the average Δt_{end} is notably higher than the average Δt_{start}: 6.84 hr (2009) and 6.97 hr (2011) for end times, compared to 3.20 hr (2009) and 3.30 hr (2011) for start times. Analyzing the remaining metrics reveals substantial differences between the 2 years. Precision dropped from 0.76 in 2009 to 0.56 in 2011. Recall is higher for 2011 (0.85) than for 2009 (0.64).

These differences are closely linked to the solar cycle. In 2011, closer to the solar maximum, the HIGeoCAT catalog contains a significantly higher number of CMEs (116) than in 2009 (55). While the number of false negatives remains almost unchanged between the two years, the number of false positives increases sharply in 2011, with 78 FP tracks compared to just 11 in 2009. This suggests that the model struggles more in periods of heightened solar activity, likely due to the increased complexity and frequency of overlapping CME events.

Figure 7 provides a visualization of the number of true positive, false positive, and false negative detections. The overall increase in CME activity from 2009 to 2011 is immediately apparent, as is the substantial rise in false positive detections during the more active year. Closer examination reveals that several CMEs listed in the HIGeoCAT catalog are missed by the model, resulting in false negative cases. However, in many of these instances, the model does detect a CME in the relevant time frame but fails to track it consistently along the HELCATS-defined PA. This indicates that while the front is often identified, incomplete segmentation of the front can lead to fragmented or misaligned tracks that do not meet the matching criteria.

It is important to note that only CMEs classified as fair or good according to the HELCATS criteria are included in HIGeoCAT, while poor-quality events are excluded. Furthermore, CMEs at very high or low position angles are often not tracked due to the limited field of view of the HI instruments. Events at both very high or low latitudes become more common near solar maximum. This affects the completeness of HIGeoCAT, and thus the model evaluation based on it. Therefore, any apparent decrease in model performance during solar maximum should be interpreted in the context of these inherent limitations in the reference data.

HICAT includes additional events not present in HIGeoCAT, providing their starting times within the HI1 FOV, but no time-elongation tracks. This adds 20 CMEs for 2009, all of which are classified as poor, and 63 for 2011. Of the 63 additional CMEs in 2011, 11 are classified as good, 32 as fair, and 20 as poor. To account for these CMEs, we attempt to associate unmatched predictions from our model with the HICAT CMEs using a simple time-based matching criterion. Since HICAT only provides a starting time, we rely on temporal proximity alone. As a starting point, we consider the mean starting time error Δt_{start} between our model and HIGeoCAT CMEs, which is ±3.20 hours for 2009 and ±3.30 hours for 2011. Rounding these values to the nearest integer multiple of the HI-1 image cadence (40 min) yields a matching window of ±3.33 hours, which we adopt as a our threshold. Applying this criterion yields 2 additional true positives for 2009 and 30 for 2011.

Figure 8 offers a direct visual comparison between the CME fronts detected and tracked by our model and those from the HIGeoCAT catalog. As in previous examples, the model often struggles to consistently identify the correct front when multiple structures are present in the image. Nevertheless, in both cases, at least one of the CMEs could still be matched to a HIGeoCAT track. A closer comparison between the HIGeoCAT annotations and the model outputs reveals that different parts of the same CME are sometimes being tracked. While our model tends to follow the outermost edge of the front, the HELCATS catalog occasionally traces structures located slightly further behind. This discrepancy underscores the inherent challenge of precisely defining the spatial extent of a CME front, especially in complex or faint events.

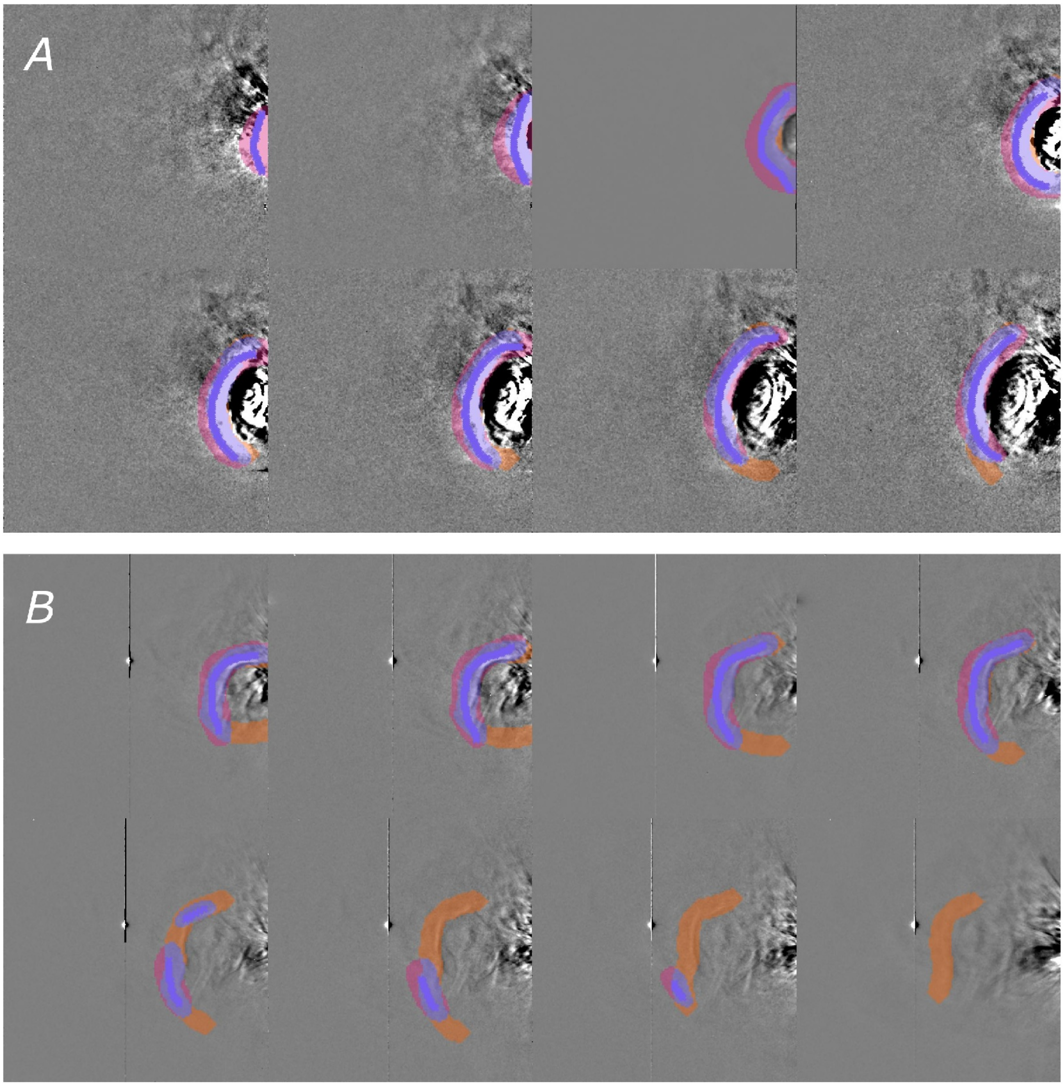

Figure 6. Time-series' of STEREO/heliospheric imager data with results from event-based tracking overlaid. Pixels are classified into true positives, false positives, and false negatives, and subsequently color-coded to show segmentation results as in Figure 5. The fronts identified by the tracking algorithm are overlaid in purple, indicating that they were correctly identified as belonging to a ground truth Coronal Mass Ejection (CME). Panel (a) demonstrates the models' ability to accurately segment and track CMEs close to the Sun, while Panel (b) shows an example of decreasing performance in both tasks as the CME grows fainter.

5. Conclusion and Discussion

The difficulty of consistently interpreting white-light images, as well as their increasing availability, have sparked the need for efficient, automated CME detection and tracking methods. In this work, we present STRUDL, a model capable of detecting CMEs in STEREO/HI data. By using sequences of images instead of individual frames as input for our model, we introduce temporal context to improve the consistency of the segmentation. Our model performs well in identifying bright CMEs but struggles with fainter and more diffuse fronts, especially near the edges of the HI1 FOV. The challenge of CME segmentation is exacerbated by the inherent ambiguity in defining CME fronts. Small mismatches between predicted and ground-truth masks can lead to a significant drop in performance metrics, such as IoU, even when the segmentation is visually accurate.

We also implement a basic tracking algorithm, which works with output from STRUDL and is able to consistently link CMEs detected by our model across images. We evaluate our tracking algorithm by comparing to our own ground-truth tracks (event-based evaluation), as well as to CMEs tracked as part of the HIGeoCAT catalog

Table 4
Summary of Key Metrics Used to Gauge Continuous Tracking Performance

Metric	2009	2011
Δt_{start}	3.20 hr	3.30 hr
Δt_{end}	6.84 hr	6.97 hr
Precision	0.76	0.56
Recall	0.64	0.85
Total CMEs	55	116
True Positives	35	99
False Positives	11	78
False Negatives	20	17

Note. The best method-threshold combination threshold(mean) = 0.45 was used to generate these results. Metrics are listed separately for 2009 and 2011.

(continuous evaluation). In both cases, we observe a higher mean absolute error for CME end times compared to start times, again indicating that the model struggles to fully track events through the HI1 FOV. The model produces several false positive tracks, particularly during solar maximum. Some of these tracks may correspond to real CMEs that were not present in the ground truth data sets. This hypothesis appears to at least partially explain the high number of false positives in the case of continuous tracking evaluation. Not all CMEs during the given periods were tracked in HIGeoCAT, and incorporating additional HICAT detections into our evaluation shows that a portion of our model's apparent false positives detections are associated with real CMEs in the HICAT catalog. Inspecting the quality flags of the unmatched HIGeoCAT CMEs reveals that most of the CMEs not detected by our model are of fair quality (16 out of 20 for 2009, 13 out of 17 for 2011), while most CMEs flagged as good quality were correctly identified. Overall, the model can identify and track most CMEs present in the ground truth data sets, proving its value as a potential early-warning tool in the future.

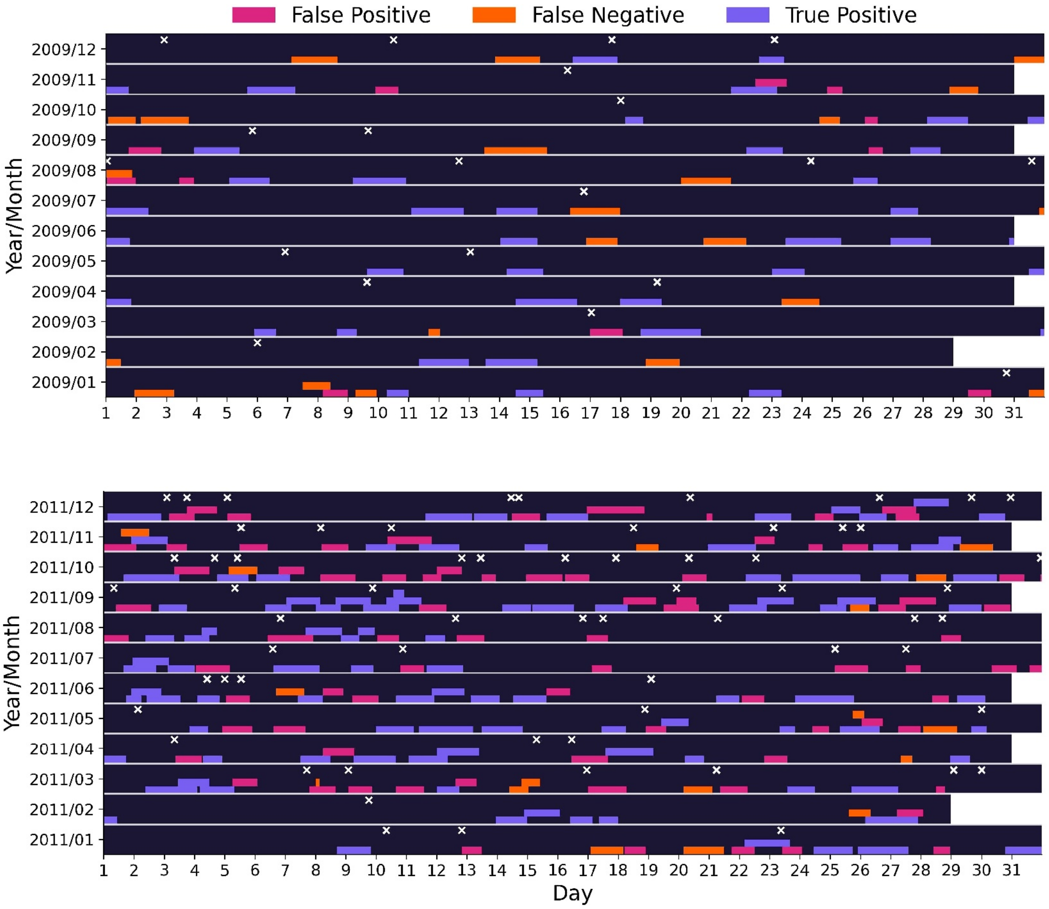

Figure 7. Results of continuous tracking evaluation for the year 2009 (top panel) and 2011 (bottom panel). Tracks obtained using our algorithm are compared to HIGeoCAT tracks of the same time period and assigned either a TP (purple) or an FP (pink) label. The beginning and end times of our track along the position angle given by the corresponding HELCATS track are shown in the case of a TP detection. If a track is labeled as an FP, the minimum beginning time and maximum end times across all PAs are plotted. HIGeoCAT tracks that could not be associated with any of our tracks are labeled as FN (orange), with the beginning and end times as provided in HIGeoCAT. Furthermore, the starting times of Coronal Mass Ejections identified as part of HICAT, but not tracked as part of HIGeoCAT, are marked with white crosses. Each row corresponds to a month within the year. The vertical positions of the tracks within a given month were chosen to prevent overlap during plotting.

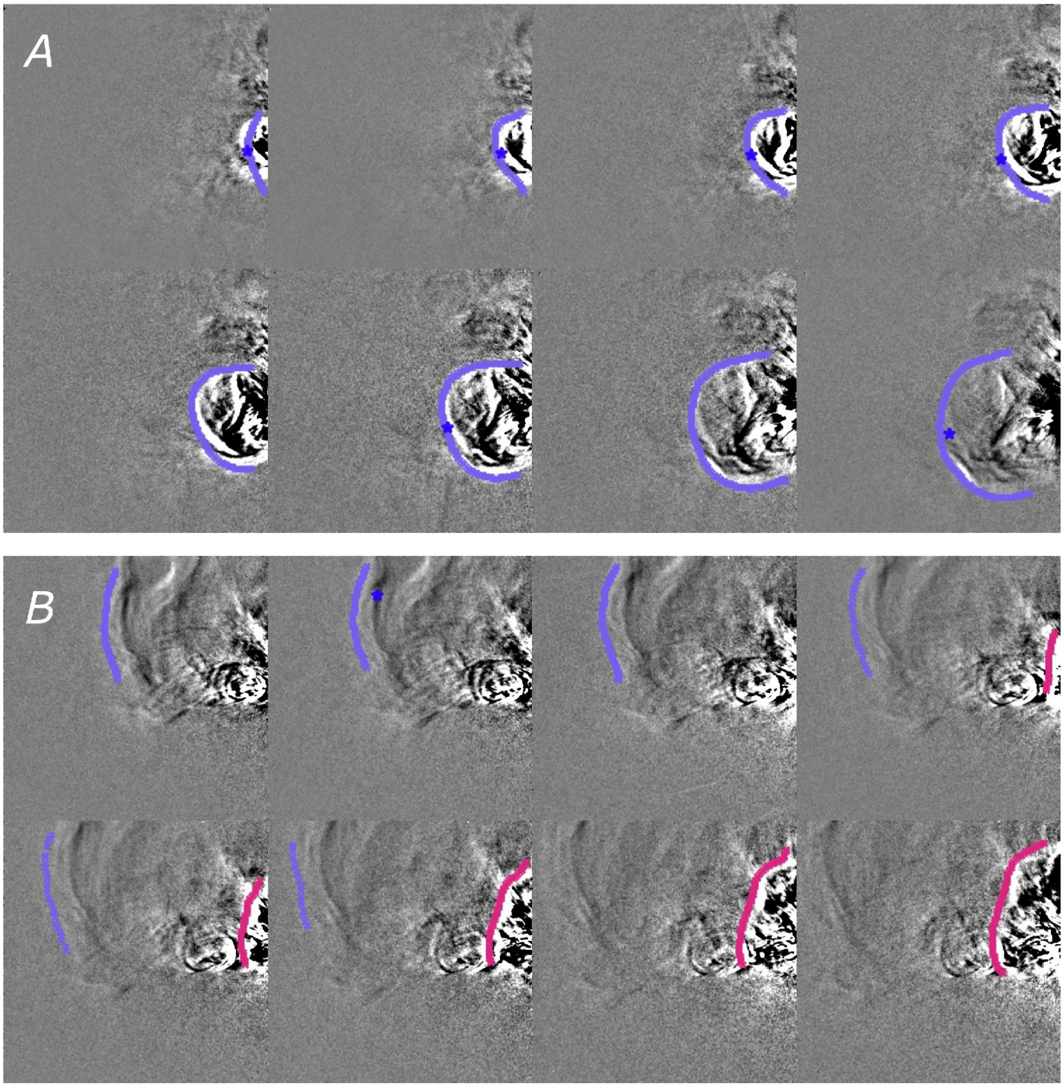

Figure 8. Examples of STEREO/heliospheric imager image sequences showcasing the performance of our algorithm for continuous tracking. Tracks that were successfully associated with a HIGeoCAT track are marked in purple, while ones for which this is not the case are marked in pink. The corresponding HIGeoCAT track is marked with a dark-purple star, where applicable. Note that some frames do not contain any HIGeoCAT data points as tracking is done in Jmaps, not images, resulting in skipped images during tracking. Panel (a) shows an example of a well defined, singular Coronal Mass Ejection (CME), resulting in a detected front that is clean and cohesive. Panel (b) shows an example of a CME immediately followed by a smaller structure, possibly the core, which is then identified as the beginning of a new CME. Despite the fact that this structure is likely to be part of the preceding CME, identifying it as a separate event could potentially be useful from an operational perspective, as both the front and core of a CME can be geoeffective.

Improvements to both segmentation and tracking performance could be made by a different approach to CME annotation. Our current annotation approach marks only the leading edge of each CME, which provides a consistent but limited representation of CME structure. As a result, the model learns to focus on identifying bright fronts, potentially leading it to identify other bright structures (e.g., the core) as new CME fronts. This is particularly apparent in images where multiple CMEs overlap or interact with each other. Distinguishing between successive CMEs and complex internal structures is important from a scientific perspective, and using a more comprehensive annotation strategy, such as using polygons to outline the entirety of a CME, could provide the model with more contextual information. From an operational perspective, however, this distinction is less significant. Both the front and the core of a CME have the potential to cause disruptions at Earth; being alerted to and tracking both separately could thus be a desirable outcome. Furthermore, while the current tracking implementation is based on simple heuristics, more advanced approaches, such as data-driven (e.g., Li et al., 2022) or probabilistic models (e.g., Spilger et al., 2021), could provide more coherent and robust tracking. These techniques may also enable estimation of CME parameters, allowing us to predict CME arrival time and speed, thus opening up more possibilities for assessing performance.

Currently, the model is trained and evaluated on STEREO/HI science data. For real-time applications, however, the model must be able to operate on low-latency, low-resolution beacon data. To facilitate this, enhancements to the input data may be necessary. Le Louëdec et al. (2025) have developed the Beacon2Science model, a machine-learning algorithm trained to convert STEREO/HI beacon data into enhanced beacon data, which has a spatial and temporal resolution closer to that of science data.

To speed up the generation of beacon training data and reduce the time spent on manual annotation, our pre-trained model may be used in a semi-automatic workflow where predicted masks are manually refined and projected onto beacon data. This process could significantly reduce the effort required compared to fully manual annotation, not only for generating beacon training data, but also for increasing the amount of available science training data. This is particularly relevant for preparing for future missions that will carry HI instruments, such as ESA's Vigil mission, scheduled for launch in 2031, or for utilizing data from the PUNCH mission, which is already operational. Further improvements in both segmentation and tracking are needed to ensure reliable performance across the full range of solar conditions. Overall, our model provides a promising foundation for automated CME detection and tracking in HI data.

Data Availability Statement

The STEREO/HI data used in this paper can be downloaded from https://stereo-ssc.nascom.nasa.gov/pub/. We provide the code for downloading and processing the data in Bauer and Le Louëdec (2025a). The code to run STRUDL is available in Bauer and Le Louëdec (2025b). The tracking tool is available as part of the same repository. The HICAT and HIGeoCAT catalogs are available under https://www.helcats-fp7.eu/catalogues/wp2_cat.html and https://www.helcats-fp7.eu/catalogues/wp3_cat.html, respectively. Data from the Solar Stormwatch II citizen science project was used in the creation of this manuscript (Scott et al., 2025). The annotations used to train STRUDL are available on Figshare (Bauer, 2025a). The configuration files needed to reproduce model results, as well as the saved weights of all of our models are available on Figshare (Bauer, 2025b). This research makes use of the AstroPy (Collaboration et al., 2022) and Pytorch (Paszke et al., 2019) libraries for Python.

Acknowledgments
This research was funded in whole or in part by the Austrian Science Fund (FWF) [10.55776/P36093]. For open access purposes, the author has applied a CC BY public copyright license to any author-accepted manuscript version arising from this submission. This work is supported by ERC grant (HELIO4CAST, 10.3030/101042188) funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them. DB recognises the support of the UK Space Agency for funding STEREO/HI operations in the UK. This publication uses data generated via the Zooniverse. org platform, development of which is funded by generous support, including a Global Impact Award from Google, and by a grant from the Alfred P. Sloan Foundation. LB is supported by a UKRI Future Leaders Fellowship (DARES, MR/Y021207/1).

References

Barnard, L., Scott, C., Owens, M., Lockwood, M., Tucker-Hood, K., Thomas, S., et al. (2014). The solar stormwatch CME catalogue: Results from the first space weather citizen science project. *Space Weather*, *12*, 657–674. https://doi.org/10.1002/2014SW001119

Barnard, L., Scott, C. J., Owens, M., Lockwood, M., Crothers, S. R., Davies, J. A., & Harrison, R. A. (2015). Differences between the CME fronts tracked by an expert, an automated algorithm, and the solar stormwatch project. *Space Weather: The International Journal of Research and Applications*, *13*(10), 709–725. https://doi.org/10.1002/2015SW001280

Barnard, L. A., de Koning, C. A., Scott, C. J., Owens, M. J., Wilkinson, J., & Davies, J. A. (2017). Testing the current paradigm for space weather prediction with heliospheric imagers. *Space Weather: The International Journal of Research and Applications*, *15*(6), 782–803. https://doi.org/10.1002/2017SW001609

Barnes, D., Davies, J., & Harrison, R. (2018). HELCATS HCME_WP2_V03. *Figshare*. https://doi.org/10.6084/m9.figshare.5803152.v1

Bauer, M. (2025a). Annotations STEREO-A HI data [Dataset]. *Figshare*. https://doi.org/10.6084/m9.figshare.29196935

Bauer, M. (2025b). STRUDL model weights [Dataset]. *Figshare*. https://doi.org/10.6084/m9.figshare.29197025

Bauer, M., & Le Louëdec, J. (2025a). *STEREO-HI-Data-Processing*. Zenodo. https://doi.org/10.5281/zenodo.16753636

Bauer, M., & Le Louëdec, J. (2025b). *STRUDL*. Zenodo. https://doi.org/10.5281/zenodo.16753563

Brueckner, G. E., Howard, R. A., Koomen, M. J., Korendyke, C. M., Michels, D. J., Moses, J. D., et al. (1995). The Large Angle Spectroscopic Coronagraph (LASCO). *Solar Physics*, *162*(1), 357–402. https://doi.org/10.1007/BF00733434

Collaboration, T. A., Price-Whelan, A. M., Lim, P. L., Earl, N., Starkman, N., Bradley, L., et al. (2022). The astropy project: Sustaining and growing a community-oriented open-source project and the latest major release (v5.0) of the core package. *The Astrophysical Journal*, *935*(2), 167. https://doi.org/10.3847/1538-4357/ac7c74

Cui, H., Liu, X., & Huang, N. (2019). Pulmonary vessel segmentation based on orthogonal fused U-Net++ of chest CT images. In D. Shen, et al. (Eds.), *Medical image computing and computer assisted intervention—MICCAI 2019* (pp. 293–300). Springer International Publishing. https://doi.org/10.1007/978-3-030-32226-7_33

DeForest, C., Killough, R., Gibson, S., Henry, A., Case, T., Beasley, M., et al. (2022). Polarimeter to UNify the Corona and Heliosphere (PUNCH): Science, status, and path to flight. In *2022 IEEE aerospace conference (AERO)* (pp. 1–11). https://doi.org/10.1109/AERO53065.2022.9843340

Domingo, V., Fleck, B., & Poland, A. I. (1995). The SOHO mission: An overview. *Solar Physics*, *162*(1), 1–37. https://doi.org/10.1007/BF00733425

Eyles, C. J., Harrison, R. A., Davis, C. J., Waltham, N. R., Shaughnessy, B. M., Mapson-Menard, H. C. A., et al. (2009). The heliospheric imagers onboard the STEREO mission. *Solar Physics*, *254*(2), 387–445. https://doi.org/10.1007/s11207-008-9299-0

Fang, Z., Chen, Y., Nie, D., Lin, W., & Shen, D. (2019). RCA-U-Net: Residual channel attention U-Net for fast tissue quantification in magnetic resonance fingerprinting. In D. Shen, et al. (Eds.), *Medical image computing and computer assisted intervention—MICCAI 2019* (pp. 101–109). Springer International Publishing. https://doi.org/10.1007/978-3-030-32248-9_12

Fiorio, C., & Gustedt, J. (1996). Two linear time union-find strategies for image processing. *Theoretical Computer Science*, *154*(2), 165–181. https://doi.org/10.1016/0304-3975(94)00262-2

Harrison, R. A., Davies, J. A., Barnes, D., Byrne, J. P., Perry, C. H., Bothmer, V., et al. (2018). CMEs in the heliosphere: I. A statistical analysis of the observational properties of CMEs detected in the heliosphere from 2007 to 2017 by STEREO/HI-1. *Solar Physics*, *293*(5), 77. https://doi.org/10.1007/s11207-018-1297-2

Howard, R. A., Vourlidas, A., Colaninno, R. C., Korendyke, C. M., Plunkett, S. P., Carter, M. T., et al. (2020). The Solar orbiter Heliospheric Imager (SoloHI). *Astronomy and Astrophysics*, *642*, A13. https://doi.org/10.1051/0004-6361/201935202

Jackson, B. V., Buffington, A., Hick, P., Kahler, S. W., Keil, S. L., Altrock, R. C., et al. (1997). The solar mass ejection imager. *Physics and Chemistry of the Earth*, *22*(5), 441–444. https://doi.org/10.1016/S0079-1946(97)00172-9

Jarolim, R., Tremblay, B., Rempel, M., Molnar, M., Veronig, A. M., Thalmann, J. K., & Podladchikova, T. (2024). Advancing solar magnetic field extrapolations through multiheight magnetic field measurements. *The Astrophysical Journal Letters*, *963*(1), L21. https://doi.org/10.3847/2041-8213/ad2450

Jha, D., Smedsrud, P. H., Riegler, M. A., Johansen, D., Lange, T. D., Halvorsen, P., & D. Johansen, H. (2019). ResUNet++: An advanced architecture for medical image segmentation. In *2019 IEEE international symposium on multimedia (ISM)* (pp. 225–2255). https://doi.org/10.1109/ISM46123.2019.00049

Johnson, J. E., Sundaresan, S., Daylan, T., Gavilan, L., Giles, D. K., Silva, S. I., et al. (2020). RotNet: Fast and scalable estimation of stellar rotation periods using convolutional neural networks. In *Machine learning and the physical sciences workshop*.

Kaiser, M. L., Kucera, T. A., Davila, J. M., St. Cyr, O. C., Guhathakurta, M., & Christian, E. (2008). The STEREO mission: An introduction. *Space Science Reviews*, *136*(1–4), 5–16. https://doi.org/10.1007/s11214-007-9277-0

Kirnosov, V., Chang, L.-C., & Pulkkinen, A. (2016). Combining STEREO SECCHI COR2 and HI1 images for automatic CME front edge tracking. *Journal of Space Weather and Space Climate*, *6*, A41. https://doi.org/10.1051/swsc/2016037

Koomen, M. J., Detwiler, C. R., Brueckner, G. E., Cooper, H. W., & Tousey, R. (1975). White light coronagraph in OSO-7. *Applied Optics*, *14*(3), 743–751. https://doi.org/10.1364/AO.14.000743

LeCun, Y., Kavukcuoglu, K., & Farabet, C. (2010). Convolutional networks and applications in vision. In *Proceedings of 2010 IEEE international symposium on circuits and systems* (pp. 253–256). https://doi.org/10.1109/ISCAS.2010.5537907

Le Louëdec, J., Bauer, M., Amerstorfer, T., & Davies, J. A. (2025). Beacon2Science: Enhancing STEREO/HI beacon data with machine learning for efficient CME tracking. *Space Weather*, *23*(7), e2025SW004440. https://doi.org/10.1029/2025SW004440

Li, Y., Bian, C., & Chen, H. (2022). Object tracking in satellite videos: Correlation particle filter tracking method with motion estimation by Kalman Filter. *IEEE Transactions on Geoscience and Remote Sensing*, *60*, 1–12. https://doi.org/10.1109/TGRS.2022.3204105

MacQueen, R. M., Gosling, J. T., Hildner, E., Munro, R. H., Poland, A. I., Ross, C. L., et al. (1975). Photography of comet Kohoutek by Skylab white light coronagraph. *NASA Special Publication*, *355*, 19.

Malik, A., Mirdehghan, P., Nousias, S., Kutulakos, K., & Lindell, D. (2023). Transient neural radiance fields for Lidar view synthesis and 3D reconstruction. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, & S. Levine (Eds.), *Advances in neural information processing systems* (Vol. 36, pp. 71569–71581). Curran Associates, Inc.

Manoharan, P. K., & Mujiber Rahman, A. (2011). Coronal mass ejections—Propagation time and associated internal energy. *Journal of Atmospheric and Solar-Terrestrial Physics*, *73*(5), 671–677. https://doi.org/10.1016/j.jastp.2011.01.017

Mikołajczyk, A., & Grochowski, M. (2018). Data augmentation for improving deep learning in image classification problem. In *2018 International Interdisciplinary PhD Workshop (IIPhDW)* (pp. 117–122). https://doi.org/10.1109/IIPHDW.2018.8388338

Pant, V., Willems, S., Rodriguez, L., Mierla, M., Banerjee, D., & Davies, J. A. (2016). Automated detection of coronal mass ejections in stereo heliospheric imager data. *The Astrophysical Journal*, *833*(1), 80. https://doi.org/10.3847/1538-4357/833/1/80

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., et al. (2019). PyTorch: An imperative style, high-performance deep learning library (no. arXiv:1912.01703). arXiv. https://doi.org/10.48550/arXiv.1912.01703

Robbrecht, E., & Berghmans, D. (2004). Automated recognition of Coronal Mass Ejections (CMEs) in near-real-time data. *Astronomy and Astrophysics*, *425*(3), 1097–1106. https://doi.org/10.1051/0004-6361:20041302

Rodriguez, L., Barnes, D., Hosteaux, S., Davies, J. A., Willems, S., Pant, V., et al. (2022). Comparing the Heliospheric Cataloging, Analysis, and Techniques Service (HELCATS) manual and automatic catalogues of coronal mass ejections using Solar Terrestrial Relations Observatory/Heliospheric Imager (STEREO/HI) data. *Solar Physics*, *297*(2), 23. https://doi.org/10.1007/s11207-022-01959-w

Rodríguez Gómez, J. M., Podladchikova, T., Veronig, A., Ruzmaikin, A., Feynman, J., & Petrukovich, A. (2020). Clustering of fast coronal mass ejections during solar cycles 23 and 24 and the implications for CME–CME interactions. *The Astrophysical Journal*, *899*(1), 47. https://doi.org/10.3847/1538-4357/ab9e72

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. In N. Navab, J. Hornegger, W. M. Wells, & A. F. Frangi (Eds.), *Medical image computing and computer-assisted intervention—MICCAI 2015* (pp. 234–241). Springer International Publishing. https://doi.org/10.1007/978-3-319-24574-4_28

Rüdisser, H. T., Windisch, A., Amerstorfer, U. V., Möstl, C., Amerstorfer, T., Bailey, R. L., & Reiss, M. A. (2022). Automatic detection of interplanetary coronal mass ejections in solar wind in situ data. *Space Weather*, *20*(10), e2022SW003149. https://doi.org/10.1029/2022SW003149

Scott, C., Barnard, L., & Jones, S. (2025). *The solar stormwatch II CME classifications*. University of. https://doi.org/10.17864/1947.001424

Spilger, R., Lee, J.-Y., Chagin, V. O., Schermelleh, L., Cardoso, M. C., Bartenschlager, R., & Rohr, K. (2021). Deep probabilistic tracking of particles in fluorescence microscopy images. *Medical Image Analysis*, *72*, 102128. https://doi.org/10.1016/j.media.2021.102128

Tappin, S. J., Howard, T. A., Hampson, M. M., Thompson, R. N., & Burns, C. E. (2012). On the autonomous detection of coronal mass ejections in heliospheric imager data. *Journal of Geophysical Research*, *117*(A5). https://doi.org/10.1029/2011JA017439

Thompson, W. T. (2006). Coordinate systems for solar image data. *Astronomy and Astrophysics*, *449*(2), 791–803. https://doi.org/10.1051/0004-6361:20054262

Vourlidas, A., Howard, R. A., Plunkett, S. P., Korendyke, C. M., Thernisien, A. F. R., Wang, D., et al. (2016). The Wide-field Imager for Solar Probe Plus (WISPR). *Space Science Reviews*, *204*(1), 83–130. https://doi.org/10.1007/s11214-014-0114-y

Wang, P., Zhang, Y., Feng, L., Yuan, H., Gan, Y., Li, S., et al. (2019). A new automatic tool for CME detection and tracking with machine-learning techniques. *The Astrophysical Journal—Supplement Series*, *244*(1), 9. https://doi.org/10.3847/1538-4365/ab340c

Webb, D. F., Johnston, J. C., Radick, R. R., & Team, S. (2002). The Solar Mass Ejection Imager (SMEI): A new tool for space weather. *Eos, Transactions American Geophysical Union*, *83*(4), 33–39. https://doi.org/10.1029/2002EO000023

Wu, K., Otoo, E., & Shoshani, A. (2005). Optimizing connected component labeling algorithms Medical Imaging 2005: Image Processing. (Vol. *5747*, pp. 1965–1976). https://doi.org/10.1117/12.596105

Wu, S., Wang, Z., Liu, C., Zhu, C., Wu, S., & Xiao, K. (2019). Automatical segmentation of pelvic organs after hysterectomy by using dilated convolution U-Net++. In *2019 IEEE 19th international conference on software Quality, Reliability and Security Companion (QRS-C)* (pp. 362–367). https://doi.org/10.1109/QRS-C.2019.00074

Yashiro, S., Gopalswamy, N., Michalek, G., St. Cyr, O. C., Plunkett, S. P., Rich, N. B., & Howard, R. A. (2004). A catalog of white light coronal mass ejections observed by the SOHO spacecraft. *Journal of Geophysical Research*, *109*(A7). https://doi.org/10.1029/2003JA010282

Yeung, M., Sala, E., Schönlieb, C.-B., & Rundo, L. (2022). Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation. *Computerized Medical Imaging and Graphics*, *95*, 102026. https://doi.org/10.1016/j.compmedimag.2021.102026

Zhang, T. Y., & Suen, C. Y. (1984). A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, *27*(3), 236–239. https://doi.org/10.1145/357994.358023