Transfer learning for instance segmentation of waste bottles using Mask R-CNN algorithm

Conference or Workshop Item

Accepted Version

www.reading.ac.uk/centaur

CentAUR

Central Archive at the University of Reading

Reading's research outputs online

Transfer Learning for Instance Segmentation of Waste Bottles using Mask R-CNN Algorithm

Punitha Jaikumar[1], Remy Vandaele[1,2], and Varun Ojha[1]

[1] Department of Computer Science, University of Reading, Reading, UK
[2] Department of Meteorology, University of Reading, Reading, UK
{p.jaikumar,r.a.vandaele,v.k.ojha}@reading.ac.uk

Abstract. This paper proposes a methodological approach with a transfer learning scheme for plastic waste bottle detection and instance segmentation using the *mask region proposal convolutional neural network* (Mask R-CNN). Plastic bottles constitute one of the major pollutants posing a serious threat to the environment both in oceans and on land. The automated identification and segregation of bottles can facilitate plastic waste recycling. We prepare a custom-made dataset of 192 bottle images with pixel-by pixel-polygon annotation for the automatic segmentation task. The proposed transfer learning scheme makes use of a Mask R-CNN model pre-trained on Microsoft COCO dataset. We present a comprehensive scheme for fine-tuning the base pre-trained Mask-RCNN model on our custom dataset. Our final fine-tuned model has achieved 59.4 *mean average precision* (mAP), that corresponds to the MS COCO metric. The results indicate promising application of deep leaning for detecting waste bottles.

Keywords: Convolutional neural networks · Object detection · Instance segmentation · Deep learning · Transfer learning

1   Introduction

Over the years, the plastic bottle has evolved from being one of the most useful materials available to a cause for the severe problem of pollution on land and seas [11]. Plastic waste poses a serious threat to the environment. The Ocean Conservancy annual report mentions that plastic bottles ranked among the top 5 pollutants [16]. In 2019, Ocean Conservancy reportedly collected 1.75 million bottles from the beach cleanup initiative [16]. In order to alleviate the problem, organizations around the world have taken various measures to collect and segregate the pollutants for plastic waste recycling and their eco-friendly disposal. Among those measures, drones and video surveillance in urban waste management along with *image processing systems* can be used to alert environmental agencies to take adequate measures.

In this work, we study the performance of a *deep learning* algorithm for detecting the waste bottles from a set of images. We prepared a custom dataset of 192 images for the training and validation of our deep learning bottle detection model developed on *mask region proposal convolutional neural network*

(Mask R-CNN). Since object shapes and forms can vary, a dataset of 192 images may not constitute an adequate training set. Therefore, we developed a methodological framework using transfer learning [17] to exploit a model pre-trained with over 5000 images 'minival' subset of the Microsoft Common Object in Context (COCO) dataset pertaining to 'val2014training' images of 80 object categories [14]. Further, the limitation of the small custom-made dataset is addressed by applying data augmentation techniques during the fine-tuning process to improve the model's performance [20].

Our fine-tuned model was trained on the custom-made dataset and when evaluated on instance segmentation offered a 59.4 *mean average precision* (mAP) following the COCO evaluation metric, *intersection over union* (IoU) of threshold range [0.5:0.05:0.95]. Also, the final fine-tuned Mask R-CNN model was able to detect bottles from test images and videos. The trained models, the outputs (images and videos), and the relevant Mask R-CNN open-source code (adapted from [1]) is available at repository [10].

This paper is organized as follows: Sec. 2 refers to the relevant literature. Sec. 3 describes the instance segmentation model training methods. The obtained results are described in Sec. 4, followed by conclusions in Sec. 5.

2   Related Work

The object segmentation task has evolved over the years in the field of computer vision. Image-based systems have revealed the potential to find an automated solution for the environmental problems and research is initiated for application in waste segregation for recycling purpose. The machine learning techniques such as support vector machine, k-nearest neighbor, decision tree, and logistic regression were used for plastic bottles waste management related work [12,13,23]. The recent advancements in deep neural networks have outperformed traditional machine learning models in the object classification and detection tasks. Deep Learning in the field of computer vision developed from image classification to object localization, object segmentation, and to instance segmentation [7] (cf. Fig. 1). There are two type of object detectors: two stage detectors-Region based R-CNN family and single stage detectors: such as *you look only once* (YOLO) [18] and *single shot detector* (SSD) [15].

Two-Stage Detectors The *region proposed* methods of object detection are from the R-CNN family. The model pipeline detects objects in two stages: (1) generating proposal of *region of interest* (RoI) and (2) classification of objects in the proposed regions. The *region proposal* algorithms have evolved in the following order [24]: R-CNN, Fast R-CNN, Faster R-CNN and Mask R-CNN.

Single-Stage Detectors The single-stage detectors take an approach of a simple regression problem predicting the bounding box coordinates and the class probabilities from images in one evaluation. The YOLO [18] and SSD [15] families belong to this category of detector, and are designed for speed and real-time use while they compromise on accuracy.
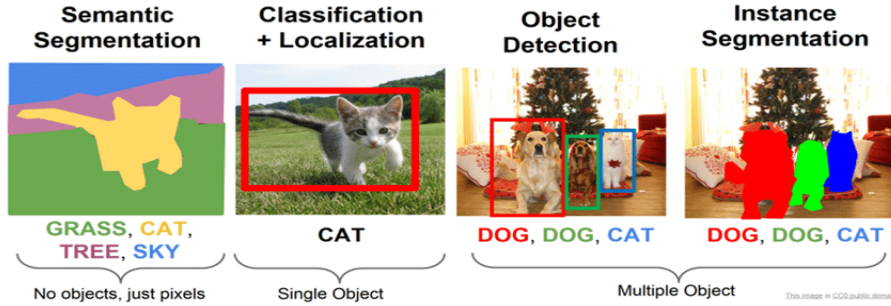
Fig. 1. Comparison of semantic segmentation, classification and localization, object detection and instance segmentation. (Figure adapted from [5])

In [24], authors discussed the approaches adopted in the object segmentation and compares the performance of various algorithms and its benchmark. The training time versus accuracy will be the main practical trade-off for detector selection when compared to other image segmentation and object detection algorithms and methods [9]. Two-stage region proposed detectors performed better over single stage detectors. In [6], the author concluded that Faster R-CNN has significantly outperformed models when applied on the dataset containing plastic bottle images of underwater debris. In [22], authors used Faster R-CNN, *single-shot multi-box detector* (SSD), and YOLO version 2 (YOLOv2) architectures; and compared their performances. They show that the Faster R-CNN with the rotational region proposed network achieved 90.3% PASCAL VOC AP precision [4], followed by 90.1% achieved by SSD and 77.4% achieved by YOLOv2.

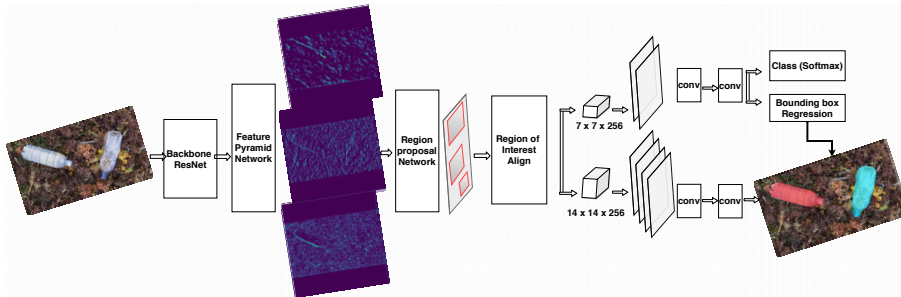**Fig. 2.** Mask R-CNN

Mask R-CNN: This algorithm belongs to the two-stage detectors that uses *feature pyramid network* (FPN), and *region proposal network* (RPN) for the object segmentation. It is an extension of the Faster R-CNN [19] along with a new pipeline for masking the detected objects for instance segmentation [8] (cf. Fig. 2). In Mask R-CNN, the spatial layout of input object encoded by

a mask and predicted by using fully connected network (FCN) pixel to pixel convolutions instead of the fully connected (FC) layers that will flatten into vectors lacking spatial dimensions as in Faster R-CNN. The RoI-align (instead of RoI-pooling) calculated with bi-linear interpolation to improve segmentation accuracy. This technique had gained popularity as it improves object instance segmentation precision [7,9,24]. In this work, for instance segmentation of bottles at pixel-level, we opted a two-stage detector: Mask R-CNN.

3   Bottle Segmentation using Mask R-CNN

The implementation of our bottle segmentation model started with the datasets preparation. The Mask R-CNN model was built with Python 3, TensorFlow, Keras, and OpenCV libraries [2] by adapting the code from open-source Mask R-CNN implementation [1]. The annotated custom dataset along with pre-trained model weights were fed into the training pipeline consisting of several stages.

Model configuration. The ResNet-50 and ResNet-101 were the *feature backbone network* options for Mask R-CNN algorithm. The images were resized to dimension 1024×1024. The initial RoI parameter was 200 for training of the classifier/mask heads. Later, for the incremental training, RoI parameter was 512 and the maximum ground truth instances for one image was updated from 100 to 512. Also, for incremental training, the *detection maximum instance* was updated from 150 to 512 that facilitates the maximum detection of object instances. The *detection minimum confidence* parameter values were 0.7 and 0.5 for tuning model's performance. The parameter setup and training process instructions are provided in [10].

Dataset description. The dataset consists of custom images downloaded from the internet for this work. Though there are many state-of-the-art datasets such as COCO and the PASCAL VOC that have general images for training and research purpose, it was noted that the availability of images pertaining to the waste bottle class was limited. To overcome this problem, a discrete set of images were downloaded from the internet. The set contains single and multiple bottle images of normal and deformed features possessing various backgrounds (cf. Table 1). Detecting objects in multi-bottle images would be a challenge due to occlusion and lighting conditions. This will be a challenge for the generalization of the bottle instance segmentation for the model.

Data pre-processing. Image annotation of each object is a time-consuming and tedious task. However, it is an essential initial step for an instance segmentation task. It is necessary to define the pixel-wise ground truth for the target objects for the models to perform segmentation and mask generation. Unlike other object detectors, the Mask R-CNN model requires pixel-wise annotation for training. There are many publicly available tools for data annotation such as VIA-VGG[3] and LabelMe [21].

For this work, pixel-wise polygon annotation of the images was performed for instance segmentation training as per the COCO dataset format. Custom dataset images were of varied sizes and formats. VIA-VGG annotator tool was

used for the pixel-wise polygon graphical annotation. The tool generates the output in json format for the annotated images. The segmented ground truth mask represents a region-wise *spatial position* and *axis* of each target object. The annotated dataset also includes two *no-instance* images and two *partial-annotation* images. The images were resized to dimension $1024 \times 1024 \times 3$ and to preserve the *aspect ratio*, each image was padded with zero to match the one-size training requirements (square format of same dimension). The dimensions of the images range from small images of 150×255 to 2448×3264 in the dataset. All the images stored in the system in two folders with the training images and the pixel annotation and labeling information as arrays for each image in the json file. These image files are read one-by-one from the system drive for resizing. Video files input for testing is pre-processed with OpenCV libraries in python to extract the images frame-by-frame to be fed into the detection pipeline for segmentation. The segmented output image frames were saved into the set directory path with the OpenCV video.

Table 1. Custom dataset of bottles downloaded from Internet

Type	No. of Waste Images	Bottle Specif Images	Total Bottle Instance Count	Bulk Bottle Instance Count	Image Range- min pixel	Image Range- max pixel	Images Re-sized (RGB)
Train	177	102	712	561	33750	7990272	$1024 \times 1024 \times 3$
Val	15	3	113	104	535824	2250000	$1024 \times 1024 \times 3$

Transfer learning scheme. The custom dataset training was initialized with the Mask R-CNN model pre-trained on MS-COCO dataset with 80 classes. The model weights in the initial layers representing low-level features will be useful in many classification tasks. While deeper layers learn the high-level features, this can be altered and retrained according to the problem definition. In the initial phase, the models were trained with backbone ResNet-50 and ResNet-101 for the waste bottle instance segmentation using *stochastic gradient descent* (SGD) and *Adam* optimizer. The deeper network of ResNet-101 and SGD optimizer performed better comparatively and was opted for further incremental training experiments on the custom dataset. Our transfer learning scheme has the following stages for pre-trained model fine-tuning:

- Stage 1: Head-layer training with all other layers frozen
- Stage 2: Fine-tuning selected layer or all layers
- Stage 3: Extended training (optional)

Table 2 shows the detailed experiment for the transfer learning scheme adopted to train for bottle segmentation. **Stage 1:** *Head layers* trained with Data Augmentation for 30 epochs initiated for ResNet101 Backbone. **Stage 2:** As part of model fine-tuning, the stage 2 models trained with the learned weights of models in the previous stage. The *4+ layers* and *ALL layers* models trained for 30

epoch with selected data augmentation. Additionally, the *4+ layers* models M4 and M10 trained up to 100 epochs. And, as part of **Stage 3**, models trained for an additional 20 epochs to assess whether it improves the performance. With respect to the *ALL layers* training at stage 3, the *4+ layers* trained models M4 and M10 tuned up to 30 epochs initially and then up to 100 epochs. Incrementally, in stage 3, models were trained up to 150 and 160 epochs for *4+ layers* and *ALL layers* respectively. Each model after training were evaluated by using COCO evaluation metric and compared against the model loss.

Model evaluation and optimization. Models performances were compared against loss metrics and were evaluated by using COCO evaluation metric mAP [8,14]. Image segmentation models being far too expensive for the cross-validation method, Mask R-CNN hyper-parameter tuning was based on the configuration parameters.

4   Results and Discussion

Table 3 shows the models *mean average precision* (mAP) results as per the MS COCO evaluation metrics- AP^{50}, AP^{75}, AP^{95} and mAP[0.5:0.95:05]. The stage 2 model M11 performed well compared to all other models trained using the transfer learning scheme. The *4+ layers* with augmentation model in stage 2 at the 100-th epoch achieved 53.0 mAP and at stage 3 tuning of 20 epochs shows gradual improvement achieving 55.3 mAP for both *ALL layers* and *4+ layers*. Tuned model for *ALL layers* without augmentation performed better in stage 2, achieving mAP for model M3 and in stage 3, achieving 56.2 and 56.4 mAP for models M12 and M14 respectively. AP^{90} increased from 13.90 precision in stage 1 to 49.20 precision in stage 3 fine tuning.

Models trained without data augmentation at stage 2 and stage 3 performed better comparatively. This could be due to the model learning with fine-tuned feature with augmentation having less diversity in the training data as it was already trained with augmentation in *Head layer* training. In this task, only horizontal left-right flip was used, more varied augmentation if applied could have shown improvement in the model performance at later stages of tuning.

Further, model behavior requires to be monitored at each incremental training as the false-positives in-addition to true positives may affect the quality of the output. The model's performance did not improve much in later stages of tuning, this could be attributed to an inadequate dataset. While training *ALL layers*, it is important to decide the training based on the dataset as most of the low-level features will be changed drastically; and it can impact the output on extended training leading to over-fitting.

The model M11 achieved 59.4 mAP and 74.6 precision for AP^{50} at stage 2 fine tuning for instance segmentation of bottle images including dense overlapped instances of varied shapes and features compared to the study done with Faster R-CNN models that shows detection of objects (including other class objects) with no overlap in [6] with 60.6 mAP and in [22] with 86.4 precision for AP^{50} and 90.3 for rotational RPN. Figs. 3 shows the sample test image segmentation for

the model M11 and Fig. 4 shows the instance segmentation of dense agglomerate of bottles. The number of instances segmented by the model increases with the decrease in the detection minimum confidence level.

Table 2. Transfer learning(TL) scheme: incremental step by step fine-tuning.

Model Ref No.	Starting Weights^a	TL-Stage	Training Layers	Augmentation	epochs	Total epochs
M1	COCO*	1	HEADS	Y	30	30
M2	M1 (30.h5)	2	ALL	Y	30	60
M3	M1 (30.h5)	2	ALL	N	30	60
M4	M1 (30.h5)	2	4+	Y	30	60
M5	**M4** (30.h5)	2	4+	Y	**70**	130
M6	M5 (100.h5)	**3**	ALL	Y	20	150
M7	M5 (100.h5)	**3**	4+	Y	20	150
M8	**M4** (30.h5)	3	ALL	Y	30	90
M9	M8 (30.h5)	3	ALL	Y	70	160
M10	M1 (30.h5)	2	4+	N	30	60
M11	**M10** (30.h5)	2	4+	N	**70**	130
M12	M11 (100.h5)	**3**	ALL	N	20	150
M13	M11 (100.h5)	**3**	4+	N	20	150
M14	**M10** (30.h5)	3	ALL	N	30	90
M15	M14 (30.h5)	3	ALL	N	70	160

*COCO dataset-Pre-trained model. Fine-tuning with ResNet-101 as the backbone architecture; SGD as the optimizer; steps as 1000; and learning rate as 0.001. ^{a} Starting weights mentions the pre-trained model weights for incremental training.

Table 3. Performance measure of the models shown in Table 2

Model Ref No.	AP^{50}	AP^{75}	AP^{90}	mAP^{50} Train	mAP^a Val
M1	75.60	72.89	13.90	96.60	57.00
M2	71.19	63.01	24.93	98.70	54.53
M3	70.29	62.69	35.77	98.96	55.36
M4	68.86	63.24	31.28	98.86	54.32
M5	66.66	61.46	37.18	99.08	53.02
M6	70.14	64.46	35.08	99.12	55.29
M7	69.93	64.22	35.04	99.11	54.87
M8	68.87	63.46	30.37	99.06	54.37
M9	68.62	63.36	30.90	98.91	54.16
M10	70.72	64.36	31.29	98.91	56.08
M11	74.59	65.31	41.24	99.13	**59.36**
M12	70.26	64.29	47.38	99.09	56.21
M13	70.61	66.50	49.20	99.11	57.83
M14	70.57	64.08	40.66	98.96	56.35
M15	70.40	63.00	40.77	99.11	55.54

^{a}mAP[0.5:0.95:0.05]-is mean of AP of IoU threshold range from 0.5 to 0.95 with 0.05 step size

Table 4. Final model - M11 fine-tuning

Detection Minimum Confidence	50%	70%	90%
mAP_val	76.13	75.14	**74.59**
mAP_val [0.5:0.5:0.95]: (after tuning)	**59.80**	**59.52**	59.36
mAP Increase (in %)	+0.44%	+0.16%	–

Fig. 3. Test image segmentation results

(a) 90% Detection minimum_confidence

(b) 70% Detection minimum Confidence

(c) 50% Detection minimum Confidence
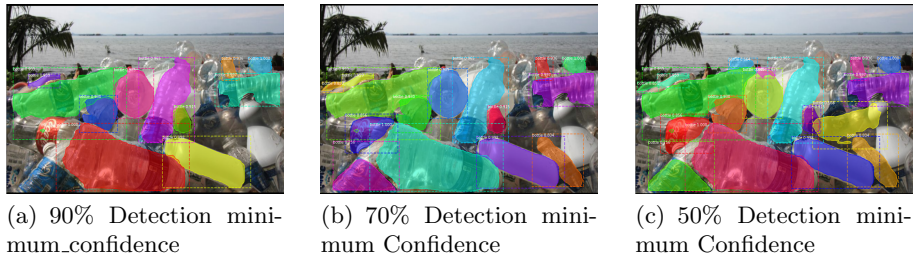
Fig. 4. Dense instance segmentation

Table 4 shows the tuning model M11's hyper-parameter *detection minimum confidence* to 0.5 (50%) and 0.7 (70%). This tuning achieves a marginal improvement in the mAP_val [0.5:0.05:0.95] of 0.44% and 0.16% over the *detection minimum confidence* of 0.9 (90%). Overall, it is evident from the results that with

a well-planned fine-tuning approach, the model performance can be leveraged to achieve better results with a limited dataset.

5   Conclusions

Mask R-CNN, one of the seminal architectures for instance segmentation studied for its application in segmentation of plastic waste bottles. The challenges such as the non-availability of a comprehensive dataset and resources for training the model were addressed by adopting a transfer learning scheme and data augmentation technique. The experiments conducted in different phases by fine-tuning the pre-trained model with a varied parameter setting showed a noticeable improvement in the performance. The models trained initially with *head layers* on the pre-trained model and later fine-tuned with *select layers* training with and without augmentation.

In the transfer learning scheme using Mask R-CNN, applied on waste bottle instance segmentation of images, we observe that the initial *head layers* training with ResNet-101 as backbone network with incremental fine-tuning achieved an AP^{50} of 74.6 precision. Further, the model evaluated with mean average precision (mAP) for instance segmentation of IoU threshold range [0.5:0.95:0.5] had achieved a 59.4 mAP. Test of images and video data produced a qualitatively noticeable good performance, in instance segmentation.

In our future work, we aim to improve training dataset availability using generative adversarial networks and investigate the dense agglomerate of bottle instances, where the model ignores the object segmentation because of the non-max suppression and other configuration thresholds.

References

1. Abdulla, W.: Mask R-CNN for object detection and instance segmentation on keras and tensorflow. https://github.com/matterport/Mask_RCNN (2017), accessed 07-Oct-2020
2. Chollet, F.: Keras: Deep learning library for theano and tensorflow. https://keras.io/ (2015), accessed 07-Oct-2020
3. Dutta, A., Zisserman, A.: The VGG image annotator (VIA). CoRR abs/1904.10699 (2019), http://arxiv.org/abs/1904.10699
4. Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The pascal visual object classes challenge: A retrospective. International journal of computer vision 111(1), 98–136 (2015)
5. Fei-Fei Li, Justin Johnson, S.Y.: Detection and segmentation, cs231n: Convolutional neural networks for visual recognition,stanford university. http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf (2017), accessed 19-sept-2020
6. Fulton, M., Hong, J., Islam, M.J., Sattar, J.: Robotic detection of marine litter using deep visual detection models. In: 2019 International Conference on Robotics and Automation (ICRA). pp. 5752–5758. IEEE (2019)

7. Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Martinez-Gonzalez, P., Garcia-Rodriguez, J.: A survey on deep learning techniques for image and video semantic segmentation. Applied Soft Computing **70**, 41–65 (2018)
8. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 2961–2969 (2017)
9. Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., Fischer, I., Wojna, Z., Song, Y., Guadarrama, S., et al.: Speed/accuracy trade-offs for modern convolutional object detectors. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 7310–7311 (2017)
10. Jaikumar, P.: Fine-tuned Mask R-CNN for plastic bottle instance segmentation. https://github.com/PJ1920/Mask-R-CNN (2020), accessed 09-Oct-2020
11. Jambeck, J.R., Geyer, R., Wilcox, C., Siegler, T.R., Perryman, M., Andrady, A., Narayan, R., Law, K.L.: Plastic waste inputs from land into the ocean. Science **347**(6223), 768–771 (2015)
12. Kambam, L.R., Aarthi, R.: Classification of plastic bottles based on visual and physical features for waste management. In: 2019 IEEE International Conference on Electrical, Computer and Communication Technologies. pp. 1–6 (2019)
13. Kokoulin, A.N., Tur, A.I., Yuzhakov, A.A.: Convolutional neural networks application in plastic waste recognition and sorting. In: 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus). pp. 1094–1098. IEEE (2018)
14. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European Conference on Computer Vision. pp. 740–755. Springer (2014)
15. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. Lecture Notes in Computer Science p. 21–37 (2016). https://doi.org/10.1007/978-3-319-46448-0_2, `http://dx.doi.org/10.1007/978-3-319-46448-0_2`
16. Oceanconservancy.org: The beach and beyond: Ocean conservancy annual report. https://oceanconservancy.org/trash-free-seas/international-coastal-cleanup/annual-data-release/ (2019), accessed 07-Oct-2020
17. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering **22**(10), 1345–1359 (2009)
18. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
19. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
20. Shorten, C., Khoshgoftaar, T.M.: A survey on image data augmentation for deep learning. Journal of Big Data **6**(1), 60 (2019)
21. Torralba, A., Russell, B.C., Yuen, J.: Labelme: Online image annotation and applications. Proceedings of the IEEE **98**(8), 1467–1484 (2010)
22. Wang, J., Guo, W., Pan, T., Yu, H., Duan, L., Yang, W.: Bottle detection in the wild using low-altitude unmanned aerial vehicles. In: 2018 21st International Conference on Information Fusion. pp. 439–444. IEEE (2018)
23. Yang, M., Thung, G.: Classification of trash for recyclability status. CS229 Project Report **2016** (2016)
24. Zhao, Z.Q., Zheng, P., Xu, S.t., Wu, X.: Object detection with deep learning: A review. IEEE Transactions on Neural Networks and Learning Systems **30**(11), 3212–3232 (2019)