Deep Learning Object Detection

Jingnian Liu[1,2], Weihong Huang[1,2(✉)], Lijun Xiao[1,2], Yingzi Huo[1,2,3], Huixuan Xiong[1,2], Xiong Li[4], and Weidong Xiao[5]

[1] School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China
{whhuang,ljxiao}@hnust.edu.cn
[2] Hunan Key Laboratory for Service Computing and Novel Software Technology, Xiangtan 411201, China
[3] Guangdong Financial High-Tech Zone "Blockchain +" Fintech Research Institute, Foshan 528253, China
[4] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Xiangtan 411201, China
[5] School of Software Engineering, Xiamen University of Technology, Xiamen 361024, China
xiaoweidong@xmut.edu.cn

Abstract. Object detection techniques are a major part of computer vision research, with large-scale applications in industrial, scientific and other scenarios. Technologies such as face detection, medical image detection, autonomous driving, and traffic detection have played a significant role in people's lives. With the rapid development of deep learning, many application areas, such as image classification, text classification, machine translation, etc., have achieved breakthrough success in combination with deep learning. R-CNN brings object detection into the era of deep learning, and its advantage compared with traditional methods is that the former requires personnel to extract features manually, while the latter uses deep learning to extract features automatically, which greatly improves efficiency, simplifies operation, and opens a new era of object detection research. First, this paper provides an overview of deep learning-based object detection backbone networks, reviews and analyzes milestone object detection algorithms, compares commonly used datasets, summarizes applications, and finally concludes the paper.

Keywords: Computer Vision · Deep Learning · Neural Network · Object-Detection · R-CNN

1 Introduction

The key task of object detection is to correctly identify objects (e.g., humans, animals, vehicles, and logo text) in a picture and to determine the location of the object [1]. By means of a rectangular edge box, to locate the detected object and to distinguish between classes of objects. Object detection has an important role in industrial scenes, scientific research, etc. And similarly, other tasks such as classification, segmentation, motion

M. Qiu et al. (Eds.): SmartCom 2022, LNCS 13828, pp. 300–309, 2023.
https://doi.org/10.1007/978-3-031-28124-2_28

estimation, and scene understanding are also fundamental problems in computer vision [2].

Most traditional object detection algorithms are constructed based on artificially constructed features [3], similar to the Viola-Jones detector [4], *Histogram of Oriented Gradients* (HOG) [5] etc. The structure of these early traditional algorithms is generally divided into three steps: informative region selection, feature extraction and classification [6], such models have obvious drawbacks and shortcomings, such as not fast convergence and poor migration ability on new datasets.

The development of deep learning [7–10] and storage technology [11, 12] has promoted the breakthrough of target detection. DCNN has excellent feature extraction and data migration capabilities, and its emergence has changed the field of object detection. The DCNN network AlexNet [13] was introduced in 2012, which has since opened the era of deep learning research boom.

In this paper, deep learning-based object detection techniques are reviewed and sorted out, and the main part will be organized as described below. In Sect. 2, the main deep convolutional neural network models are reviewed, and their architectures and performances are concisely described. Section 3 summarizes important object detection algorithms from the past to the present, and their structures are carefully analyzed and compared. Section 4 reviews the commonly used datasets and evaluation criteria in object detection. Section 5 summarizes the main current application. Section 6 concludes the paper.

2 Backbone Network for Object Detection

The rise of deep learning [14–16], big data [17, 18], computer capability [19–21], and cloud computing [22, 23], has also led to the development of object detection. In the object detection task, we usually use convolutional neural networks to extract features from images for subsequent recognition and localization of objects, which is a very important part of the object detection field. In the following, we will focus on reviewing landmark networks in deep learning.

2.1 AlexNet

AlexNet is one of the seminal works in the field of deep learning, which opened a new era of modern deep learning. The earliest convolutional neural network was LeNet [24], which perfectly solved the handwritten digit recognition task and achieved an average accuracy of 98% on the MNIST dataset. Alexet uses a deeper and wider network compared to LeNet, consisting of five convolutional layers, three maximum pooling layers, and three fully connected layers. Each convolution layer uses multiple channels to enhance information processing capability. The activation function between the intermediate layers is performed by relu to speed up the model convergence, while a new regularization technique dropout is used on the first two fully connected layers in order to cope with the overfitting problem. The last fully connected layer is passed through softmax, which produces a vector of size 1000 to represent the distribution of categories. AlexNet achieved the best result on the ImageNet LSVRC-2010 dataset at that time, with an accuracy rate of 62.5% and 83% on top-1 and top-5 classifications (Fig. 1).

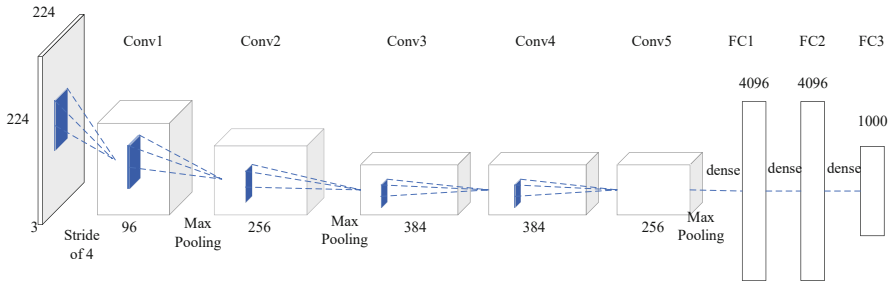

Fig. 1. Schematic diagram of AlexNet structure.

2.2 VGG

VGG has deeper layers and more parameters than AlexNet. VGG uses a convolutional kernel of 3×3 with a step size of 1 to get more information and details about the object from the picture, and uses the ReLu Layer after each convolutional layer. The dominant structures are VGG16 and VGG19, the former consisting of 13 convolutional layers and 3 fully connected layers, and the latter consisting of 16 convolutional layers and 3 fully connected layers [25]. Combining multi-crop and dense evaluation, using scale jittering to resize the images, we achieved second place in the classification task of the ILSVRC-2014 challenge with an error rate of 7.3% and won first place in the localization task. VGG proved at that time that a deeper and wider network would lead to higher accuracy.

2.3 GoogLeNet

Since the start of the deep learning boom, convolutional networks have moved in a deeper and broader direction with more parameters. But larger models also require higher computational costs and are more likely to cause overfitting of data, so how to achieve high accuracy with lower computational costs is the core goal of GoogLeNet [26]. Inception block is an important concept in GoogLeNet. The structure of the Inception model consists of four parallel paths with different sizes of convolutional kernels ($5 \times 5, 3 \times 3, 1 \times 1$) to extract information simultaneously, 1×1 convolutional kernel can change the dimension and reduce the parameters while achieving the purpose of deepening the network and interacting with information across channels [27]. GoogLeNet won the ILSVRC-2014 challenge, outperforming other networks of the same period with an error rate of 6% in the classification task [28].

2.4 Resnet

The deeper layers will cause gradient disappearance, gradient explosion, and degradation problems will occur, and 56 layers will not even perform as well as 26 layers. Resnet introduced Batch Normalization [29] to solve the gradient disappearance and gradient explosion, and proposed residual to solve the degradation problem. The residual module notates the mapping stacked by convolutional layers as H(x), and the first input of these layers is noted as x. By way of a shortcut connection, we can choose to skip some layers.

Rather than expect stacked layers to approximate H(x), we explicitly let these layers approximate a residual function F(x) := H(x) − x [30]. The advantage of this is that at least it does not make the parameters worse, it is able to train deeper models, effectively solving the degradation problem, and the depth of the model can even reach more than 1000 layers. Resnet won first place in all kinds of tasks in the ILSVRC 2015 competition [30]. Some subsequent networks, similar to Densenet [31], and ShuffLeNet [32], were also inspired by the ideas in Resnet and were born.

3 The Architectures of Object Detection

Object detection is not limited to the identification of a particular object, but requires the detection and localization of many objects within an image. Traditional object detection algorithms usually require manual efforts to extract features, which has inherent drawbacks such as poor performance on new datasets and inefficient operation. DCNNs employ deep convolutional networks to automatically extract object features, greatly improving efficiency and speed. In the following, we review the important algorithms and models based on DCNN in the field of object detection (Table 1).

Table 1. Comparison of mainstream algorithms for target detection.

Methods	Backbone	Highlights	Year
R-CNN	AlexNet	The first application of deep neural networks to object detection	2014
Fast R-CNN	VGG16	Put the whole image and its bounding boxes into the neural network	2015
Faster R-CNN	VGG16	Use the neural network to generate proposals to improve efficiency	2016
Yolo	GoogLeNet(Modified)	A neural network is used to complete the work of bounding box generation and feature extraction	2015
SSD	VGG-16	The accuracy is guaranteed while the speed is guaranteed	2016

3.1 Two Stage

R-CNN. R-CNN [33] is the first model that successfully applies deep learning to object detection, and the module is designed as described below. 2000 region proposals are obtained by Selective search, fixing all region proposals to the same size, then applying Alext to each region proposal for feature extraction and outputting a vector of 4096 sizes, and finally classifying them by a trained SVM to remove the candidate bounding boxes with IOU values larger than a threshold by NMS. Finally, a trained regression model is

used to predict the correction of its bounding box by training four parameters, centroid, height and width.

R-CNN obtained the best results in the then VOC2007, VOC2010 and other object detection challenge competitions. However, R-CNN has to perform feature extraction for all 2000 region proposals, and there are many crossovers between the region proposals and redundant feature extraction operations, resulting in slow speed, large space occupation, and the need to train AlexNet, SVM and regressor individually.

Fast R-CNN. In response to a series of problems with R-CNN, Fast R-CNN was born in 2015. Fast R-CNN uses VGG16 as the Backbone of the network, which not only makes a breakthrough in speed but also improves the accuracy rate than before.

The specific operation is region selection first (this step is consistent with R-CNN), unlike R-CNN which puts each region proposal for feature extraction, Fast R-CNN extracts features by putting the whole image and the region proposals on the image into VGG16 at once. The Rol pooling layer [34] is used for selecting the region of interest and feeding the resulting feature vectors into two fully connected layers. One of which is responsible for discriminating the category and one is responsible for locating the anchor box to the correct position. The Fast RCNN uses multi-task loss jointly train classification and bounding-box regression so that two tasks share convolution features.

Faster R-CNN. Although Fast R-CNN has been effective, traditional methods of generating candidate bounding boxes such as selective search still have the problem of taking a long time. Faster R-CNN generates detection boxes directly using RPN [35] based on Fast R-CNN (RPN is a fully connected neural network), which further improves the running speed. This is done as follows: (1) generating a large number of anchors (2) RPN determines whether all the anchors contain objects, but not their categories, and (3) adjusting the positions of the anchors to get more reasonable proposals. The ROI pooling layer is used to adjust the vectors to a uniform size, and then output to the fully connected layer. The Faster R-CNNN is different from the Compared with previous R-CNN series, the region selection task also uses a deep learning approach, which greatly improves the operational efficiency.

3.2 One Stage

Yolo. Traditional two-stage models need two steps to generate a bounding-box and predict object types. In contrast, Yolo is an end-to-end model in which the predicted bounding boxes and object classes are obtained by only one network.

Yolo's Backbone framework is inspired by the GoogLeNet model [36] and consists of 24 convolutional layers, and 2 fully connected layers, but instead of using Inception blocks, a 1×1 convolution is used behind a 3×3 convolution. The full connected layer vector in Yolo is Reshaped into a three-dimensional tensor with a size of $7 \times 7 \times 30$ [36], which is responsible for predicting the object if its center falls in a cell. 7×7 represents the division of the image into 7×7 cells, and 30 represents the generation of two bounding boxes per cell, each predicting five values (c, x, y, w, h), and 20 categories.

Yolo is inferior to Faster R-CNN in terms of accuracy, but faster than Faster R-CNN. The Yolo positioning accuracy is not enough, multiple targets are close together, the

target is too small, and the detection effect is not good. J. Redmon et al. subsequently proposed yolov2 [37], yolov3 [38] (Fig. 2).

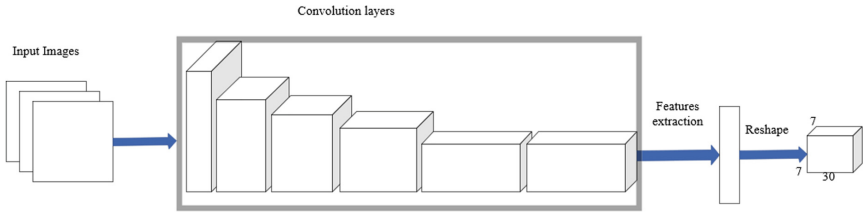

Fig. 2. Schematic diagram of Yolo structure.

SSD. In order to ensure the speed and accuracy at the same time, Liu W et al. proposed SSD, which is the same as the popular detection model nowadays, SSD combines the whole detection process into a single deep neural network, combining the advantages of Faster R-CNN and Yolo, and its speed is faster than the one stage model of the same period, yolo v1 is faster and has higher accuracy, 59 FPS with MAP 74.3% on VOC2007 test, vs Faster R-CNN 7 FPS with MAP 73.2% or YOLO 45 FPS with MAP 63.4% [39].

The Backbone of SSD is VGG16, which uses a base network to extract features, generating multiple anchor frames at each pixel on a feature map at different scales, predicting bounding boxes and categories for each anchor frame. The convolution layer halves the height and width of the input image to arrive at fitting small objects with the bottom layer and large objects with the top layer.

4 Datasets and Evaluation Criteria

4.1 Datasets

Datasets are an important part of the object detection task to train parameters and evaluate models. This subsection will systematically review the classic datasets that have made outstanding contributions and advanced research in the field of object detection.

The creation of the VOC (2005–2012) challenge made an important contribution to the development of the computer vision field. The most commonly used ones are VOC07 and VOC12, which have been extended to 20 classes of objects compared to VOC05. The training set size of VOC07 [40] has been increased to 5k and has more than 12k labeled objects. In contrast, the training set size of VOC12 reached 11k and had 16k labeled objects [41].

The ImageNet Large Scale Visual Recognition Challenge (ILSVRC) (2010–2017) [42], which had made irreplaceable contributions to the development of image classification, target detection and other fields. The ImageNet dataset, created under the auspices of Stanford professor Feifei Li, contains over 14 million tagged images, and 1000 classes of objects, including over 500k images for the target detection class and 200 classes.

MS-COCO is the most challenging dataset at present, with a huge scale and a high status in the industry, mainly used for object detection, instance segmentation and other scenarios.MS-COCO has fewer categories than ImageNet, but more instances per category, with more than 2.5 million tags in 320,000 images, containing 91 common object categories, 82 of which have more than 5,000 tagged instances [43].

Open images is a dataset built by Google that launched its first version in 2016 and includes about 6,000 categories and over 9 million images. In 2018, Google launched Open Images V4, which contains 15.4 million border boxes for 600 categories on 1.9 million images [44].

4.2 Evaluation Criteria

Evaluation criteria are used to measure how good the network is on the dataset. There are many different kinds of evaluation criteria in the object detection task, such as recall, accuracy, mean average precision (MAP), FPS, etc. The following is an analysis of the evaluation criteria in object detection data.

Intersection over union (IOU) is the ratio of intersection and union of predicted and true bounding-box. If the IOU is greater than the threshold value, the prediction is considered as True Positive(TP), and if the IOU is less than the threshold value, the prediction is considered as False Positive(FP). If the object in the bounding box is not detected by the model, it is recorded as False Negative (FN). Precision measures the percentage of correct predictions while recall measures the correct predictions with respect to the ground truth 2.

$$Precision = TP/(TP + FP)\#(1) \tag{1}$$

$$Recall = TP/(TP + FN)\#(2) \tag{2}$$

Based on the above equation, Average Precision is calculated for each class separately. Average Precision of all classes is averaged to obtain *mean Average Precision* (mAP) and MAP is used to compare the performance between detectors.

5 Applications

5.1 Face Detection

Face detection has been an important application scenario in the field of object detection, where the task goal is to find out the face in the image and determine its location, and the traditional face detection is mainly done by manually extracting the face features and then using a sliding window to match out the face in the image, where the representative algorithm is VJ detector [4]. Object detection has achieved great success since it entered the era of deep learning, and face detection algorithms are inextricably linked with general-purpose object detection algorithms such as the RCNN family. A cascaded CNN containing multiple cascaded DCNN classifiers was proposed [45], which improves the speed of face detection and solves the problems caused by illumination and angle in some realistic applications. To improve the problem of multi-pose and face occlusion recognition, [46] was proposed subsequently.

5.2 Text Detection

Text is one of the most important information carriers in human society and is a necessary part of people's lives. Text detection in images has important applications in many aspects, such as intelligent traffic, recognizing road signs and slogans. Used for information extraction, automatic recognition of text in natural scenes can save a lot of resources and protect customer privacy.

In the early days, text detection was usually extracted manually, but in the era of deep learning, features are usually extracted automatically using neural networks. This has greatly improved efficiency and simplified the workflow.

Mainstream text detection is divided into two ways, one is to first detect the text with generic object detection and then identify the text content, including image preprocessing, feature representation, sequence modeling (or character segmentation recognition), and prediction. Among the representative algorithms are [47] and others. And one is the end-to-end recognition approach, in which text detection and text recognition were previously divided into two separate problems, while end-to-end systems unite them into one, and recently, building real-time and efficient end-to-end systems has become a new trend in the community [48].

6 Conclusions

This paper reviewed the evolution of object detection, focusing on the contribution of deep learning-based object detection to the industry and research development, as well as comparing its advantages and where it has advanced compared to traditional approaches. The architecture of landmark backbone networks for target detection, such as AlexNet, VGG, GoogleNet, and ResNet was analyzed. Deep learning-based target detection algorithms such as R-CNN, Fast R-CNN, and Faster R-CNN were summarized and reviewed, and their differences from traditional object detection algorithms are analyzed and their characteristics were compared. The architectures of One Stage algorithms YOLO and SSD were concisely outlined, and their advantages and disadvantages were compared with Two Stage algorithms, and their operational effects were illustrated. Four major datasets in the field of object detection were introduced and the evaluation criteria of the models were parsed. Finally, a summary of the classic applications in target detection.

References

1. Xiao, Y., Tian, Z., Yu, J., et al.: A review of object detection based on deep learning. Multimedia Tools Appl. **79**(33), 23729–23791 (2020)
2. Zaidi, S.S.A, Ansari, M.S., Aslam, A., et al.: A survey of modern deep learning based object detection models. Digital Signal Processing, p. 103514 (2022)
3. Zou, Z., Shi, Z., Guo, Y., et al.: Object detection in 20 years: a survey. arXiv preprint arXiv: 1905.05055 (2019)
4. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features, IEEE CVPR 2001, vol. 1, pp. I-I (2001)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE CVPR, vol. 1, pp. 886–893.3 (2005)

6. Zhao, Z.Q., Zheng, P., Xu, S., et al.: Object detection with deep learning: a review. IEEE Trans. Neural Networks Learn. Syst. **30**(11), 3212–3232 (2019)
7. Liang, W., Long, J., Li, K.C., et al.: A fast defogging image recognition algorithm based on bilateral hybrid filtering. ACM TOMM **17**(2), 1–16 (2021)
8. Diao, C., Zhang, D., Liang, W., et al.: A novel spatial-temporal multi-scale alignment graph neural network security model for vehicles prediction. In: IEEE TITS (2022)
9. Peng, L., Peng, M., Liao, B., et al.: Improved low-rank matrix recovery method for predicting miRNA-disease association. Sci. Rep. **7**(1), 1–10 (2017)
10. Xiao, W., Tang, Z., Yang, C., et al.: ASM-VoFDehaze: a real-time defogging method of zinc froth image. Connect. Sci. **34**(1), 709–731 (2022)
11. Wang, J., Luo, W., Liang, W., et al.: Locally minimum storage regenerating codes in distributed cloud storage systems. China Commun. **14**(11), 82–91 (2017)
12. Liang, W., Huang, Y., Xu, J., et al.: A distributed data secure transmission scheme in wireless sensor network. Int'l J. Distrib. Sensor Netw. **13**(4), 1550147717705552 (2017)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classifification with deep convolutional neural networks. In: NIPS (2012)
14. Qiu, H., Dong, T., et al.: Adversarial attacks against network intrusion detection in IoT systems. IEEE Internet Things J. **8**(13), 10327–10335 (2020)
15. Qiu, H., Zheng, Q., et al.: Topological graph convolutional network-based urban traffic flow and density prediction. IEEE Trans. ITS (2020)
16. Hu, F., Lakdawala, S., et al.: Low-power, intelligent sensor hardware interface for medical data preprocessing. IEEE Trans. Info. Tech. Biome. **13**(4), 656–663 (2009)
17. Gai, K., Qiu, M., Elnagdy, S.: A novel secure big data cyber incident analytics framework for cloud-based cybersecurity insurance. IEEE BigDataSecurity (2016)
18. Li, Y., Gai, K., et al.: Intercrossed access controls for secure financial services on multimedia big data in cloud systems. ACM TMCCA (2016)
19. Qiu, M., Chen, Z., Ming, Z., Qin, X., Niu, J.: Energy-aware data allocation with hybrid memory for mobile cloud systems. IEEE Syst. J. **11**(2), 813–822 (2014)
20. Qiu, M., Xue, C., Shao, Z., Sha, E.: Energy minimization with soft real-time and DVS for uniprocessor and multiprocessor embedded systems. In: IEEE DATE Conference, pp. 1–6 (2007)
21. Qiu, M., Jia, Z., et al.: Voltage assignment with guaranteed probability satisfying timing constraint for real-time multiproceesor DSP, JSPS. Springer (2007)
22. Niu, J., Gao, Y., Qiu, M., Ming, Z.: Selecting proper wireless network interfaces for user experience enhancement with guaranteed probability. JPDC **72**(12), 1565–1575 (2012)
23. Li, J., Ming, Z., et al.: Resource allocation robustness in multi-embedded systems with inaccurate information. J. Syst. Architect. **57**(9), ore e840–849 (2011)
24. LeCun, Y., Bottou, L., Bengio, Y., et al.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
25. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
26. Dhillon, A., Verma, G.K.: Convolutional neural network: a review of models, methodologies and applications to object detection. Progress Artific. Intell. **9**(2), 85–112 (2019). https://doi.org/10.1007/s13748-019-00203-0
27. Liu, L., Ouyang, W., Wang, X., et al.: Deep learning for generic object detection: a survey. Int. J. Comput. Vis. **128**(2), 261–318 (2020)
28. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9 (2015)
29. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. International Conference on Machine Learning. PMLR, pp. 448–456 (2015)

30. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition, IEEE CVPR, pp. 770–778 (2016)
31. Huang, G., Liu, Z., Van Der Maaten, L., et al.: Densely connected convolutional networks, IEEE CVPR, pp. 4700–4708 (2017)
32. Zhang, X., Zhou, X., Lin, M., et al.: Shufflenet: an extremely efficient convolutional neural network for mobile devices. In: IEEE CVPR, pp. 6848–6856 (2018)
33. Girshick, R., Donahue, J., Darrell, T., et al.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: IEEE CVPR, pp. 580–587 (2014)
34. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1440–1448 (2015)
35. Ren, S., He, K., Girshick, R., et al.: Faster r-cnn: towards real-time object detection with region proposal networks. Adv. Neural Info. Process. Syst. **28** (2015)
36. Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: Unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788 (2016)
37. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7263–7271 (2017)
38. Redmon, J., Farhadi, A.: Yolov3: an incremental improvement. arXiv preprint arXiv:1804.02767 (2018)
39. Liu, W., Anguelov, D., Erhan, D., et al.: SSD: single shot multibox detector. In: European Conference on Computer Vision, pp. 21–37. Springer, Cham (2016)
40. Everingham, M., Winn, J.: The PASCAL visual object classes challenge 2007 (VOC2007) development kit. Int. J. Comput. Vis **88**(2), 303–338 (2010)
41. Everingham, M., Winn, J.: The PASCAL visual object classes challenge 2012 (VOC2012) development kit. Pattern Analysis Statistical Modelling and Computational Learning, Technical Report 2012, pp. 1–45 (2007)
42. Russakovsky, O., Deng, J., Su, H., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vis. **115**(3), 211–252 (2015)
43. Lin, T.-Y., Maire, M., Belongie, S., et al.: Microsoft coco: Common objects in context. In: Fleet, David, Pajdla, Tomas, Schiele, Bernt, Tuytelaars, Tinne (eds.) Computer Vision – ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V, pp. 740–755. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10602-1_48
44. Kuznetsova, A., Rom, H., Alldrin, N., et al.: The open images dataset v4. Int. J. Comput. Vis. **128**(7), 1956–1981 (2020)
45. Li, H., Lin, Z., Shen, X., et al.: A convolutional neural network cascade for face detection. In: IEEE CVP, pp. 5325–5334 (2015)
46. Shi, X., Shan, S., Kan, M., et al.: Real-time rotation-invariant face detection with progressive calibration networks. IEEE CVPR, pp. 2295–2303 (2018)
47. Wang, T., Wu, D.J., Coates, A., et al.: End-to-end text recognition with convolutional neural networks. In: 21st IEEE ICPR, pp. 3304–3308 (2012)
48. Chen, X., Jin, L., Zhu, Y., et al.: Text recognition in the wild: a survey. ACM Comput. Surv. **54**(2), 1–35 (2021)