A Comprehensive Review of One-stage Networks for Object Detection

Yifan Zhang, Xu Li, Feiyue Wang, Baoguo Wei, Lixin Li
School of Electronics and Information
Northwestern Polytechnical University
Xi'an, P.R. China
e-mail: lixu@nwpu.edu.cn

2021 IEEE International Conference on Signal Processing, Communications and Computing (ICSPCC) | 978-1-6654-2918-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICSPCC52875.2021.9564613

***Abstract*—Object detection has always been a hot topic in image processing, which is important in a variety of applications. With the advent of the era of big data and the continuous improvement of hardware computing power, deep learning gets more attention in object detection. One popular branch is regression-based (One-stage) model, which uses a single neural network to directly predict bounding boxes and class probabilities from the entire image by one evaluation. One-stage networks can effectively increase the detection speed. This article mainly describes object detection methods based on regression object detectors (One-stage methods), such as You Only Look Once (YOLO) series and Single Shot Multibox Detector (SSD) series. Then, their applications are briefly introduced. The development trend and future development direction of this type of object detection are discussed in the end.**

Keywords—deep learning, object detection, regression, YOLO, SSD

I. INTRODUCTION

The task of object detection is to find out all the interested objects in the image and determine their positions and sizes, which is also one core problem in the field of machine vision. Object detection has been applied in many aspects, such as face recognition [1], pedestrian detection [2], vehicle detection [3] and so on.

Object detection mainly has two tasks: target location and classification. Object detection model mainly has three modules: information area selection, feature extraction and classification. Information area selection is the process of selecting objects which appear in different positions and with different scales and sizes in the image. Feature extraction is to convert arbitrary data into digital features that can be used for machine learning. Classification refers to the process of filtering and separating the target from all other categories.

Deep-learning-based object detection methods are gaining importance in the field of object detection. The main task of deep learning based methods is to build deep convolutional neural networks (DNN), use a large number of sample data as input, and finally obtain a model with strong analysis and recognition capabilities. The model contains the constituent parameters of DNN to be applied to actual tasks.

Deep-learning based object detection usually consists of region proposal (two-stage method) and regression classification object detection (one-stage method). The two methods have their own advantages and disadvantages. The former has higher classification and positioning accuracy, and the latter is faster. This article only focuses on object detection for regression classification (one-stage method).

The YOLO series is one of the masterpieces of one-stage object detection. The first generation of YOLO (YOLOv1) was proposed in 2016 [4]. Compared with the popular two-step object detector at the time, the detection speed of YOLOv1 has a significant advantage which is several times of the two-stage detector. However, its detection accuracy rate is lower than the two-stage detector. In the same year, Single Shot MultiBox Detector (SSD) was proposed [5]. It is also a one-step detector. It improves detection efficiency while retaining the speed advantage, but its ability to detect small objects is insufficient. YOLOv2 and Deconvolutional Single Shot Detector (DSSD) were proposed [6] in 2017, adopting a more effective network structure. The modules added are more suitable for the network and can significantly improve speed and accuracy. From 2018 to 2020, several YOLO versions have been proposed. Compared with the previous versions, there are improvements in various aspects. More details will be introduced in the subsequent sections.

The one-stage detector consumes less time than the two-stage framework, but one-stage detection framework has worse performance when detecting the small objects [4-7]. This is due to the light weight of the backbone network, the avoidance of pre-processing algorithms, the fewer requirements for predicted candidate regions, and the use of FC sub-systems network.

II. REGRESSION OBJECT DETECTION METHOD

Conventional object detecting is a process of using a bounding box to locate an object, then using tags to classify the object and indicate the object confidence on the box. General object detectors are divided into two categories: region-proposal based object detectors and regression-based object detectors.

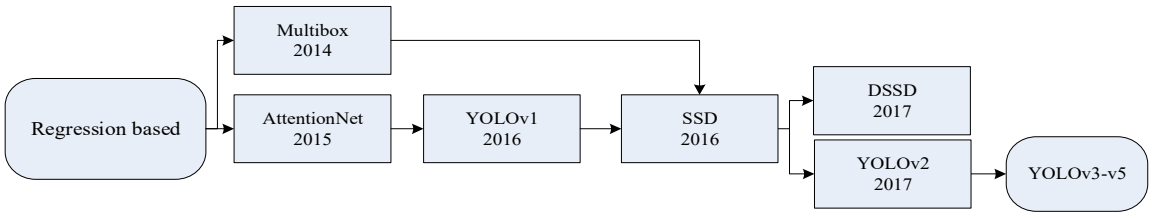

Fig. 1. Schematic diagram of the development of partial regression classification detectors

The methods based on the region proposal (two-stage method) require the algorithms to generate the target candidate frame which is the target position, and then classify and regress the candidate frame. The well-known algorithms are R-CNN [8], Fast R-CNN [9], Faster R- CNN [10], etc. The methods based on regression (one-stage method) only use a convolutional neural network (CNN) to predict the categories and positions of the interested object targets. The representative networks are YOLOv1 [4], SSD [5], etc.

Some one-stage object detection architectures will be discussed in detail, including YOLO [4], SSD [5], and DSSD [6]. Fig. 1 shows the development process of partial regression categorical detectors.

A. YOU ONLY LOOK ONCE

In 2016, Redmon et al. proposed a novel one-stage object detector that uses the highest feature mapping and direct evaluation of category probabilities to predict the bounding box [4]. Fig. 2 shows YOLOv1 object detecting model.

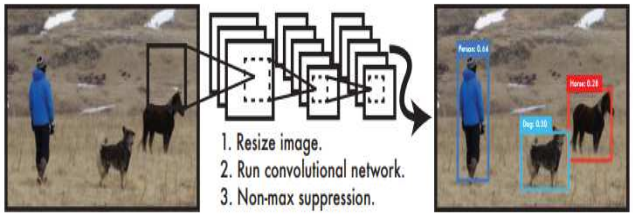

Fig. 2. YOLOv1 object detection model [4]

1) YOLOv1 object detection model can be divided into three steps:

Adjust the input image to a 448×448 image, run the convolutional neural network and determine the threshold of the detection result according to the confidence of the model.

2) The key ideas for the construction of YOLOv1 network are:

a) Prediction problem: YOLOv1 first divides the input image into S×S cells. Bounding boxes and the confidence score of the bounding box will be predicted by each cell. The confidence score includes two points. One is the probability of the existence of the target in this box, the other is the position accuracy of this bounding box. The former is recorded as Pr(object). If there is no target in the box, Pr(object)=0, and if it contains the target, Pr(object)=1. The accuracy of the position of the bounding box is judged by a method called IOU (the area where the predicted box intersects with the real box, and the ratio of the combined area of the predicted box and the real box), which is recorded as IOU (Pred). Use the four values (x, y, w, h) to represent the size and position of the bounding box. Confidence is defined as the multiplication of these two items:

$$Pr(Class_i|Object) * Pr(Object) * IOU_{pred}^{truth}$$

$$= Pr(Class_i) * IOU_{pred}^{truth} \quad (1)$$

b) Classification problem: After each cell predicts the value of (x, y, w, h, c), it is also necessary to give the probability value for the C categories. Then each cell predicts B×5+C values. As Fig. 3 shown, the input is divided into S×S grids, then the final predicted value is a tensor of size of S×S×(B×5+C). B is the number of the bounding boxes, C represents the type of classification and S is the division of cells.

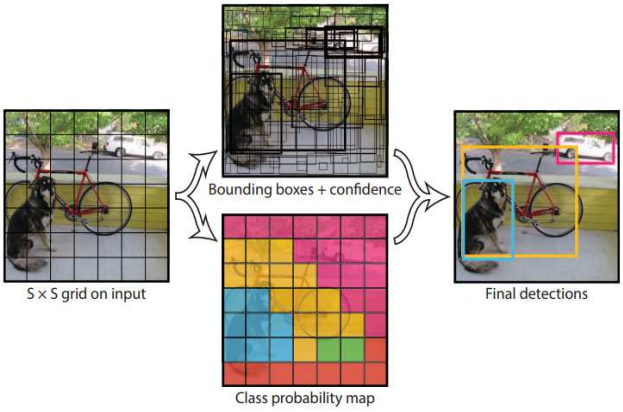

Fig. 3. The steps of YOLOv1 object detecting [4]

In Fig. 4, the network structure of YOLOv1 totally has 24 convolutional layers, and 2 fully connected layers. There are some interspersed 1×1 convolutional layers aiming to reduce previous layer's feature space to reduce operations. Before training, use half of the resolution (the input image with 224×224 pixels) to pre-train the convolutional layer on the ImageNet classification task, and then double the resolution for detection [4].

The class probability and bounding box coordinate of the target are predicted by the last layer of the network. In the final classification detection, the YOLO network uses the non-maximum suppression algorithm (NMS) to solve the problem of multiple detections from one target. All layers in the network structure use leaky rectified linear activation (leaky ReLU):

$$\emptyset(x)=\begin{cases} x, & x>0 \\ 0.1x, & otherwise \end{cases} \quad (2)$$

3) The optimized loss function in the training process is defined as follows:

$$\begin{aligned} &\lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{obj} [(x_i - x`_i)^2 + (y_i - y`_i)^2] \\ &+ \lambda_{coord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{obj} [\left(\sqrt{\omega_i} - \sqrt{\omega`_i}\right)^2 \\ &+ \left(\sqrt{h_i} - \sqrt{h`_i}\right)^2] \\ &+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{obj} (C_i - C`_i)^2 \\ &+ \lambda_{noord} \sum_{i=0}^{S^2} \sum_{j=0}^{B} l_{ij}^{obj} (C_i - C`_i)^2 \\ &+ \sum_{i=0}^{S^2} l_{ij}^{obj} \sum_{c \in class} \left(p_i(c) - p_i`(c)\right)^2 \end{aligned} \quad (3)$$

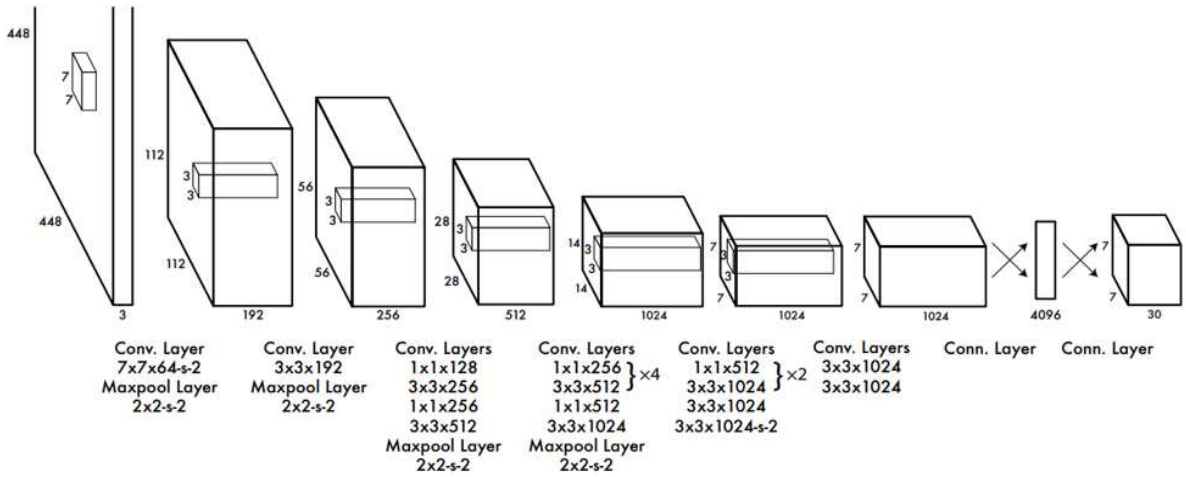

Fig. 4. YOLOv1 network structure [4]

Where l_i^{obj} indicates whether the target appears in cell *i*, and l_{ij}^{obj} denotes the *j-th* bounding box predictor variable in cell *i*. w_i and h_i are the normalized height and width relative to the image, respectively. C_i is a confidence score.

4) Loss function analysis: YOLOv1 uses the mean square error loss function to calculate the loss. Different weights will be used for errors caused by different parts. Compared with the bounding box corresponding to the large target, the coordinate error of the smaller bounding box generated by detecting the small target should be more sensitive. This is achieved by changing the prediction of the width and height of the network bounding box to its square root prediction.

In real-time object detection, YOLOv1 model can process images at 45 frames per second, while the fast version of YOLO can reach 155 frames per second, which is much faster than other real-time target detectors at that time.

5) Advantages of YOLOv1:

a) Fast detection speed, which is suitable for real-time tasks.

b) The prediction target and category are completed by a single network, which can be trained end-to-end to improve its accuracy.

6) Disadvantages of YOLOv1:

a) YOLOv1 does not perform well in detecting adjacent targets and small groups.

b) Weak generalization ability. When objects of the same type have new and abnormal aspect ratios and other situations, the detection effect is not good.

B. YOLOv2

In 2017, the second version (YOLOv2) was designed [11], which has significantly improved speed and accuracy. On the base of YOLOv1, YOLOv2 has made the following improvements:

1) Batch normalization: The Batch Normalization (BN) layer normalizes the input of each layer of the network, so that the network does not need to learn the distribution of data in each layer, which can speed up the convergence [12]. In addition, since BN can standardize the model, YOLOv2 network structure removes the previous dropout after adding BN layer. The experiments prove that mAP increases by 2% after adding the BN layer.

2) Anchor box： YOLOv1 uses data from the fully connected layer to accomplish frame prediction, which leads to more spatial information loss and inaccurate localization making the accuracy low. In YOLOv2, the authors take inspiration from the idea of anchor points in Faster R-CNN (anchor points are the key step in RPN networks) and perform a sliding window operation on the convolutional feature map, where each center predicts 9 different sizes of proposed frames. The result showed that the recall rate increased but the accuracy rate decreased after adding the anchor boxes.

3) High resolution classifier: The classifier increases the input resolution from 224×224 to 448×448 during the detecting process.

4) Darknet-19: YOLOv2 proposes to adopt a new backbone network as the feature extraction part, referring to the advanced experience of predecessors. More 3×3 convolutional kernels in the network structure are used in YOLOv2 , and then the number of channels double after the pooling operation. YOLOv2's network uses a global average pool. The network places 1×1 convolutional kernels between 3×3 convolutional kernels to compress the features. Batch normalization (BN layer) is also used to stabilize model training. The final basic model is Darknet-19, which contains 19 convolutional layers and 5 maxpooling layers. Darknet-19 provides higher accuracy and needs minimal operations to process images, which improves accuracy and speed.

C. YOLOv3

Compared with YOLOv2, YOLOv3 makes some changes [13]: First, it replaces the softmax loss function used in YOLOv2 with the logistic loss function. Each ground truth only matches one priori box and each scale only predicting 3 frames. Compared to predicting five frames per scale in YOLOv2, YOLOv3 reduces the complexity of the network. Secondly, the author used 9 anchors in YOLOv3 rather than 5 in YOLOv2, which increases the IOU. Finally, the backbone network of YOLOv3 changes from Darknet-19 to Darknet-53. YOLOv3 has a great improvement in accuracy without reducing the speed.

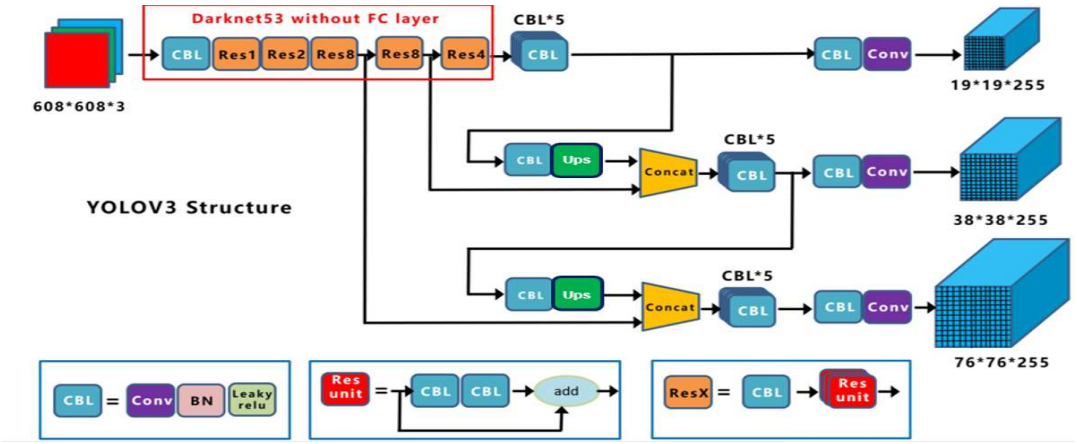

Fig. 5. YOLOv3 network structure [13]

D. YOLOv4 - YOLOv5

1) YOLOv4:

From the network structure, YOLOv4 is similar to YOLOv3, but there are innovations and optimizations in all parts [14]:

a) Input layer: The innovation mentioned here mainly refers to the improvement of the input terminal during training, including Mosaic data enhancement, cmBN, and SAT self-antagonistic training.

b) Backbone network: The backbone network uses CSPDarknet53, introducing Mish activation function, Dropblock, FPN+PAN structure and SPP module.

c) Output layer: Compared with YOLOv3, the main improvements in the output layer are that the loss function during training is adjusted to CIOU_Loss and the NMS of the filter prediction frame is changed to DIOU_nms.

With the improvement of the network, the performance of YOLOv4 has been significantly improved compared with the previous versions.

2) YOLOv5

Whether YOLOv5 should be called the fifth version of the YOLO series or not is still controversial. YOLOv4 and YOLOv5 have their own advantages and disadvantages. YOLOv4 is higher in accuracy than YOLOv5, but YOLOv5 has more advantages in flexibility and network size.

The network structure of YOLOv5 is a bit similar to YOLOv3, but there are some differences in the structure:

a) Input layer: YOLOv5 adopts Mosaic data enhancement, adaptive anchor frame calculation, adaptive image scaling and other innovations at the input.

b) Backbone network: YOLOv5 introduces the Focus structure, CSP structure and FPN+PAN structure in the backbone network.

c) Output layer: Take the GIOU_Loss loss function in prediction.

E. Single Shot Multibox Detector

The first generation of YOLO has difficulty in dealing with the generalization of unusual aspect ratio objects. What's more, multiple down-sampling operations will produce standard features [4]. Moreover, because space constraints have a great influence on the prediction of bounding boxes, it is difficult to detect small targets.

To solve these problems, Liu et al. proposed a model with reference to Multi-Box, which uses anchor points [15], RPN [10] and multi-scale representation [16], called "Single Shot MultiBox Detector (SSD)" [5]. Fig. 6 shows the structure of SSD network. SSD uses a specific feature map for object detection rather than using the default grids in YOLOv1. SSD has different aspect ratios and a default set of anchor boxes. It can be scaled to discrete the output space of the bounding box and therefore achieve better performance. The experimental results show that SSD is faster and more effective thanYOLOv1 [5]. However, the detection effect of SSD on small objects is not ideal.

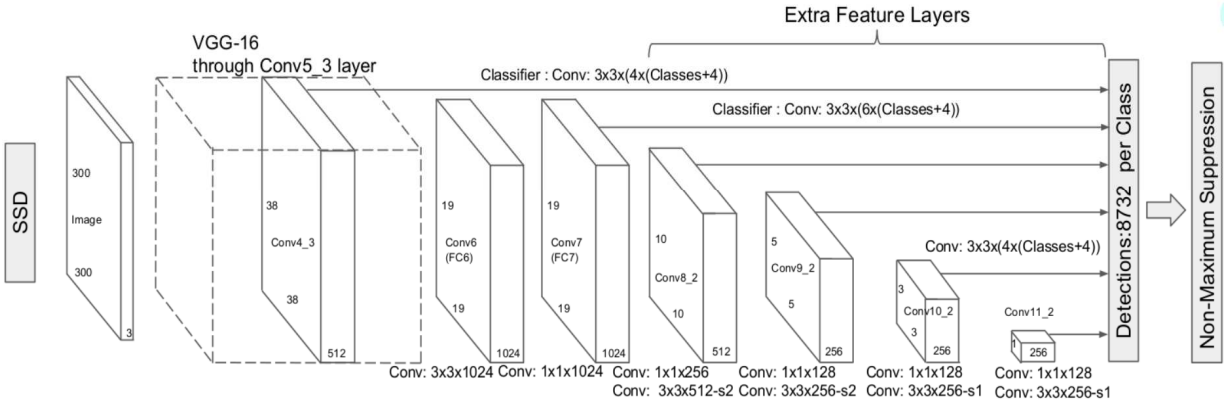

Fig. 6. SSD network structure [5]

F. Deconvolutional Single Shot Detector

To solve the problem of poor detection of small targets in SSD, Deconvolutional Single Shot Detector (DSSD) was proposed. It can effectively solve the problem of small target detection [6].

DSSD uses a better basic network (ResNet). It introduces a deconvolution layer, improves the feature extraction backbone framework (ResNet101) by using some deconvolution layers with jump connections, and connects context information to improve the characterization ability of shallow feature maps [6].

III. Application Areas and Development Trends

One-stage method plays a very important role in security military [17-19], medical [20], transportation and other fields [21-23]. As a summary, Table I lists the performance of each detector under different data sets.

The fast methods such as YOLO and SSD often have good results in real-time target detection. In medicine, Punn et al. proposed a deep learning-based framework that uses the YOLOv3 object detection model to separate people from the background and uses bounding boxes and assigned IDs to track the identified people through the Deepsort method [24]. In agriculture, Tian et al. proposed an improved YOLOv3 model for detecting apples at different growth stages in the orchard with light changes, complex backgrounds, overlapping apples, and branches and leaves [25]. In transportation, Hendry et al. applied the YOLO-darknet deep learning framework to the problem of car license plate detection [26]. This method uses 7 convolutional layers of YOLO to detect a single category. In face detection, considering the difficulty of detection due to different detection target states for face detection, Ranjan et al. proposed to perform multiple tasks (detecting facial landmark positioning and head pose estimation) without affecting the execution of a single task [27]. In addition, one-stage method also has important applications in optical character recognition (OCR) [28, 29].

One-stage method has been playing an increasingly important role in the field of object detection. The authors here briefly talk about the deficiencies and future trends of one-stage method for object detection.

1) Limitations

a) The one-step method often leads to a reduction in accuracy while pursuing speed advantages.

b) For small amounts of data, current frameworks may not yield good results. Most of the current algorithms use migration learning, if the target data is not in a dataset, the training results may be poor or unsatisfied.

c) The interpretability of the framework is poor, especially at a deep level. The lack of clear explanations for the intermediate process makes it more like a black box.

d) The calculation intensity of the framework is too heavy, which leads to high requirements for training equipment in deep learning.

e) Deficiencies in small object target detection.

2) The development direction of one-stage method

Current one-stage methods are still very limited. The authors believe that the future trend can be carried out in the following directions:

a) The object detection performed by the current network model is inseparable from the data set. Therefore, to obtain a higher recognition rate and a more complete recognition type, it is necessary to consider developing a larger data set that contains more types and richer images in the category. Network acceleration, compact design and lightweight networks in target detection build a hot and evolving research field [30-32].

b) The data annotation process is more laborious and becomes more onerous as the volume of the data set increases [33-35]. Unsupervised or weakly supervised network models can be considered to reduce the cost of the training set.

TABLE I. An Overview of the Performance of One-Stage Object Detectors

One-stage Detectors	Backbone	Max FPS	AP	Dataset	Performance analysis
YOLOv1-448 [4]	GoogleNet like	45	63.4	VOC2007+2012	Lightweight, high-speed, poor-accuracy, Insensitive to small goals.
YOLOv2-416 [11]	Darknet-19	67	78.6	VOC2007+2012	Borrowing from R-CNN's anchor box idea, higher-accuracy (compared to YOLOv1).
SSD-300 [5]	VGGNet-16	16.4	31	MSCOCO	Compared to the YOLOv1, SSD can handle objects with abnormal aspect ratio/configuration, but has poor detection capabilities for small objects.
DSSD-321 [6]	ResNet-101	11.8	33.2	MSCOCO	Improved the ability to detect small objects and improved accuracy, but the speed is slightly reduced.
YOLOV3-416 [13]	Darknet-53	34.5	33	MSCOCO	Deepen the depth of the darknet network, and the accuracy is improved again.
YOLOv4 [14]	CSPDarknet-53	125	45.8	MSCOCO	Introduce a series of strengthening measures to greatly improve the accuracy of target detection.
YOLOv5	CSPDarknet-53 like	200	45.5	MSCOCO	There are four models to choose from, which is more flexible than YOLOv4, but the performance is slightly worse.

c) The one-stage network model has the characteristics of the fast speed but poor accuracy, while the two-stage network model has high accuracy but slow speed. Therefore, is there a way to effectively combine the two models to meet the requirements of high accuracy and high speed? This maybe a good research point.

d) FPGA-based object detection networks such as terminal object detection have good prospects in daily life applications, and they are generally equipped with one-stage object detection networks to meet real-time needs.

Deep learning modules based on regression are used more and more in object detection, but there are still many limitations. When these limitations are well addressed, regression-based target detection methods will move toward faster, more accurate, and more efficient aspects that can contribute to the development of human endeavors.

ACKNOWLEDGMENT

This research work has received funding from the Open Fund of Shanghai Key Laboratory of Multidimensional Information Processing, East China Normal University (No. 2019KEY001) and the European Union Horizon 2020-ULTRACEPT (778062).

REFERENCES

[1] K. Sung and T. Poggio, "Example-based learning for view-based human face detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 20, no. 1, pp. 39-51, Jan. 1998.

[2] P. Dollar, C. Wojek, B. Schiele and P. Perona, "Pedestrian Detection: An Evaluation of the State of the Art," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 4, pp. 743-761, April 2012.

[3] K. Liu and G. Máttyus. "Fast Multiclass Vehicle Detection on Aerial Images." IEEE Geoscience and Remote Sensing Letters vol. 12, pp. 1938-1942, 2015.

[4] J. Redmon, S. Divvala, R. B. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection." 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 779-788, 2016.

[5] W. Liu. et al. "SSD: Single Shot MultiBox Detector." 2016 European Conference on Computer Vision, vol. 9905, pp. 21-37, 2016.

[6] C. Fu, W. Liu, A. Ranga, A. Tyagi and A. Berg, "DSSD : Deconvolutional Single Shot Detector." vol. ArXiv abs/1701.06659, 2017.

[7] H. Jonathan. et al. "Speed/Accuracy Trade-Offs for Modern Convolutional Object Detectors." 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3296-3297, 2017.

[8] R. B. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation." 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 580-587, 2014.

[9] R. B. Girshick, "Fast R-CNN." 2015 IEEE International Conference on Computer Vision, pp. 1440-1448, 2015.

[10] S. Ren, K. He, R. B. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, pp. 1137-1149, 2015.

[11] R. Joseph and A. Farhadi, "YOLO9000: Better, Faster, Stronger." 2017 IEEE Conference on Computer Vision and Pattern Recognition, pp. 6517-6525, 2017.

[12] S. Ioffe, C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.", vol. ArXiv abs/1502.03167, 2015.

[13] R. Joseph and A. Farhadi, "YOLOv3: An Incremental Improvement." vol. ArXiv abs/1804.02767, 2018.

[14] Bochkovskiy. Alexey et al. "YOLOv4: Optimal Speed and Accuracy of Object Detection." vol. ArXiv abs/2004.10934, 2020.

[15] D. Erhan, C. Szegedy, A. Toshev and D. Anguelov, "Scalable Object Detection Using Deep Neural Networks." 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2155-2162, 2014.

[16] S. Bell, C. L. Zitnick, K. Bala and R. B. Girshick, "Inside-Outside Net: Detecting Objects in Context with Skip Pooling and Recurrent Neural Networks." 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2874-2883, 2016.

[17] F. Zhang, B. Du, L. Zhang and M. Xu, "Weakly Supervised Learning Based on Coupled Convolutional Neural Networks for Aircraft Detection." IEEE Transactions on Geoscience and Remote Sensing, vol. 54, pp. 5553-5563, 2016.

[18] J. Han, D. Zhang, G. Cheng, L. Guo and J. Ren, "Object Detection in Optical Remote Sensing Images Based on Weakly Supervised Learning and High-Level Feature Learning." IEEE Transactions on Geoscience and Remote Sensing vol.53, pp. 3325-3337, 2015.

[19] Q. Li, Y. Wang, Q. Liu and W. Wang "Hough Transform Guided Deep Feature Extraction for Dense Building Detection in Remote Sensing Images." 2018 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 1872-1876, 2018.

[20] Z. Li. et al., "CLU-CNNs: Object detection for medical images." Neurocomputing, vol. 350, pp. 53-59, 2019.

[21] X. Song. et al., "ApolloCar3D: A Large 3D Car Instance Understanding Benchmark for Autonomous Driving." 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5447-5457, 2019.

[22] M. Liang, B. Yang, S. Wang and R. Urtasun, "Deep Continuous Fusion for Multi-sensor 3D Object Detection." 2018 European Conference on Computer Vision, vol. 11220, pp. 663-678, 2018.

[23] H. Lin, C. Chang, V. L. Tran and J. Shi, "Improved traffic sign recognition for in-car cameras." Journal of the Chinese Institute of Engineers, vol. 43, pp. 300-307, 2020.

[24] N. S. Punn, N. Singh and S. Agarwal, "Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques." vol. ArXiv abs/2005.01385, 2020.

[25] Y. Tian et al., "Apple detection during different growth stages in orchards using the improved YOLO-V3 model." Computers and Electronics in Agriculture. vol. 157, pp. 417-426, 2019.

[26] Hendry and R. Chen, "Automatic License Plate Recognition via sliding-window darknet-YOLO deep learning." Image Vision Comput Image and Vision Computing vol. 87, pp. 47-56, 2019.

[27] R. Ranjan, V. Patel and R. Chellappa, "HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition." IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, pp. 121-135, 2019.

[28] Wei. T, U. U. Sheikh, A. Rahman, "Improved optical character recognition with deep neural network." 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications, pp. 245-249, 2018.

[29] C. Wick, C. Reul and F. Puppe, "Calamari - A High-Performance Tensorflow-based Deep Learning Package for Optical Character Recognition." vol. ArXiv abs/1807.02004, 2020.

[30] T. Yao, Y. Pan, Y. Li and T. Mei, "Exploring Visual Relationship for Image Captioning." 2018 European Conference on Computer Vision, vol. 11218, pp. 711-727, 2018.

[31] A. Howard. et al., "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications." vol. ArXiv abs/1704.04861, 2017.

[32] J. M. Alvarez and M. Salzmann, "Learning the Number of Neurons in Deep Networks." Neural Information Processing Systems, vol. arXiv abs/1611.06321v3, October 2016.

[33] T. Lin. et al., "Microsoft COCO: Common Objects in Context." 2014 European Conference on Computer Vision, vol. 8693, pp. 740-755, 2014.

[34] M. Everingham. et al., "The Pascal Visual Object Classes Challenge: A Retrospective." International Journal of Computer Vision, vol. 111, pp. 98-136, 2014.

[35] O. Russakovsky. et al., "ImageNet Large Scale Visual Recognition Challenge." International Journal of Computer Vision, vol. 115, pp. 211-25, 2015.