Pedestrian Detection Based on Deep Learning Under the Background of University Epidemic Prevention

Ruiyan Du, Jia Zhao(✉), Jiangfan Xie, and Tian Wen

Hebei Normal University, Shijiazhuang 050010, China
zhaojia2021@hebtu.edu.cn

Abstract. In the context of the current normalization of epidemic prevention, the nucleic acid detection process in colleges and universities is limited in human and material resources. Teachers and students who perform nucleic acid detection often cannot maintain a distance of more than one meter from others, and there is a pedestrian group behavior that has a large cross-infection safety hazard. This article uses Depthwise Separable Convolution to improve the YOLOv3 algorithm, and the improved network structure constructs a pedestrian detection, pedestrian tracking, pedestrian counting and pedestrian cluster system based on Deep Learning under the TensorFlow framework. The training parameters and training time of the improved network model are reduced to a certain extent, improved the operation efficiency of the network model. The advantage is that it realizes the function of monitoring centralized nucleic acid detection scenes in colleges and universities and assisting volunteers to maintain a reasonable order, which can effectively prevent cross-infection problems caused by cluster effects.

Keywords: YOLOv3 · TensorFlow · Pedestrian detection · Pedestrian tracking · Pedestrian counting

1 Introduction

1.1 The Development and Research Status of Pedestrian Detection Technology

The prevention and control of the epidemic situation in colleges and universities has long been a key link in the local prevention and control work. College students come from all over the country, and there is greater mobility within the school. In the process of regularly organizing and centralized nucleic acid testing for all relevant personnel, it was discovered that due to limited manpower and material resources, there are often group behaviors that pose safety hazards such as personnel gathering and trajectory overlap. Therefore, a pedestrian distance, trajectory, and number detection system based on deep learning has been constructed in colleges and universities. And a standardized nucleic acid detection execution mechanism is very necessary.

Pedestrian detection faces various challenges such as diverse human postures, complex detection scenarios, complex model building, and serious occlusion problems.

H. Jin et al. (Eds.): CROWNCOM 2021/WiCON 2021, LNICST 427, pp. 192–202, 2022.
https://doi.org/10.1007/978-3-030-98002-3_14

Therefore, there is a huge room for optimization and progress. Pedestrian detection essentially belongs to the category of target detection, and the effect of image feature extraction is the key factor affecting the detection quality. Pedestrian detection can be divided into the following two categories according to the extraction method of video features: one is traditional pedestrian detection algorithms that integrate machine learning, image processing and artificial design features, and the other is deep learning [1, 2] pedestrian detection method based on CNN feature extraction.

1.2 Traditional Pedestrian Detection Based on Machine Learning

In the 1990s, traditional pedestrian detection methods that integrated machine learning, image processing, and artificially designed features began to rise and gradually developed. Machine learning in the traditional mode uses the human body's own appearance characteristics for manual design to train classifiers. The target classification feature is obtained by pattern classification, and the extraction method of its main feature has gone through the following development process: In 1999, the SIFT scale invariant feature transformation can extract the scale information in the extreme points of the spatial scale [3]; in 2005, the edge orientation and intensity based on information research, a gradient histogram of all pixels in the grid was proposed [4, 5]; in 2008, a DPM variability component model using HOG features appeared, which can be independently modeled according to different parts of pedestrians. Generally speaking, the framework of traditional pedestrian detection can be divided into the following main modules, as shown in Fig. 1.

Fig. 1. The general sequence of the main modules in the traditional pedestrian detection

1.3 Pedestrian Detection Based on Deep Convolutional Neural Network

Traditional pedestrian detection based on artificial features and machine learning can achieve high accuracy under certain conditions. However, since Deep Learning has been applied to large-scale image classification, academia and industry have realized that the features learned by deep learning have excellent robustness. The CNN feature extraction technology can be traced back to the 1960s. Hubel and Wiesel [6] called the area sensed by neurons as the "receptive field" and discovered that the working mechanism of the nerve-center-brain is an iterative and abstract process. In 1980, Kunihiko Fukushima proposed a neurocognitive machine model that received two-dimensional analog signals to form a multilayer network with simple cognitive capabilities, that is, the earliest network form of CNN; Le Net-5 [7] came out in 1998, the training of the handwritten digit data set used back propagation to modify the network parameters. However, due to the hardware conditions of the computer at that time, the deep convolutional network entered the winter of research.

It was not until 2012 that Alex Net, an eight-layer neural network constructed with CNN technology and accelerated by GPU, was proposed for the first time, and more and more indepth neural networks came out in the upsurge. In the same year, R-CNN, a target detection algorithm appeared. The convolutional network in the algorithm is only used for feature extraction. The average accuracy of detection in standard databases is about 20% higher than that of traditional algorithms. Fast R-CNN was proposed in 2015, and the use of the SoftMax function greatly reduced the time consumed in the entire detection process.

Since the detection process is divided into candidate region generation and regression classification, the detection speed of the target detection algorithm based on region generation at that time was relatively slow. Later, the end-to-end skip region generation step YOLO [8] and SSD [9] algorithms appeared. Among them, the YOLO algorithm can achieve end-to-end target detection and divide the input image into $S \times S$ grids, and this grid will predict the bounding box and its confidence. If no object falls into the network cell, the confidence score should be zero.

In summary, in the current process of nucleic acid detection in colleges and universities due to limited human and material resources, there is a problem of pedestrian cluster behavior of the teachers and students who perform nucleic acid detection often cannot maintain a distance of more than 1 m from others. This article uses deep separable convolution to improve the YOLOv3 algorithm. Construct a pedestrian detection, pedestrian tracking, pedestrian counting and pedestrian cluster system based on Deep Learning under the TensorFlow framework to monitor centralized nucleic acid detection scenarios in colleges and universities to maintain a reasonable order, which can effectively prevent cross-infection problems.

2 Overview of Deep Neural Networks

As we all know, the ability of humans to think comes from the human brain, a complex network composed of billions of highly interconnected neurons. Therefore, the realization of brain-like intelligence [10] is inseparable from the study of the working mechanism of the brain. The computer can use “linear weighted sum” and “function mapping” to simulate the process of nerve cells receiving stimuli and outputting signals. Interpretation, labeling, and clustering of data features through the weight link and activation function algorithm of neural nodes between layers, class data features, the sequence data, the sequence data that we directly touch every day must be converted into numerical values before being sent to the neural network for deep learning before it can be operated on. A deep neural network consists of a network composed of multiple layers and several nodes that are connected and crossed to realize the specified algorithm. With the rapid development of computer hardware and performance, the three models of shallow fully connected network model, convolution network model and deep residual convolution network model can roughly represent the three change stages of neural network structure.

2.1 Shallow Fully Connected Network Model

The shallow fully connected neural network model has strong modeling capabilities for networks with few input features, and its structure is clear, which is easy to understand and operate. As shown in Fig. 2 below, the structure contains 10 input feature nodes. Calculation of the first layer (hidden layer): $z^{[1]} = W^{[1]} \cdot x + b^{[1]}$, $a^{[1]} = \sigma(z^{[1]})$ the calculation process of the second layer (output layer) network is similar to that of the first layer. There is a causal relationship between the network outputs y_1, y_2 and each input node.

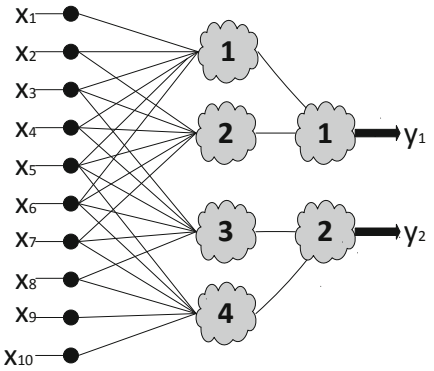

Fig. 2. Schematic diagram of shallow fully connected structure

Although the shallow neural network is relatively convenient to use, it is possible to build models for cases with more input features such as images, but the excessive number of weight matrices and offset vector parameters require higher computing capabilities for the computer, and the parameters to be trained too much, it is impossible to take into account the position information of the closer pixels between the pictures. In a network with a large number of layers, the loss function gradient transmission of this structure is more difficult, and the abstraction of the network expression will be affected, which limits the application scenario.

2.2 Convolutional Network Mechanism

With the introduction of deep learning and the improvement of hardware equipment, convolutional neural networks perform feature extraction and pattern recognition of image objects through an end-to-end learning method, and the local areas and functions of each layer of node features are weighted and activated by the convolution kernel function. After that, the node features of the next layer are obtained.

The convolution kernel of a convolutional neural network can share parameters. One of its advantages, which is different from ordinary neural networks, is its sparsity feature. This unique property can not only reduce the amount of model calculations, but also effectively limit its fitting ability; due to the maximum pooling operation of the

feature map and its own computational characteristics, for the input features, the convolutional layer has the ability to move and not deform. The above features make the convolutional layer can still perform image feature extraction well under the premise of much fewer learning parameters work.

2.3 Deep Residual Network Model

Experiments have shown that after the number of layers of the neural network increases to a certain level, the convolution neural network carries out feature extraction and pattern recognition on the image object through an end-to-end learning method. After the local region and function of the node feature of each layer are weighted and activated by the convolution kernel function, the node feature of the next layer is obtained.

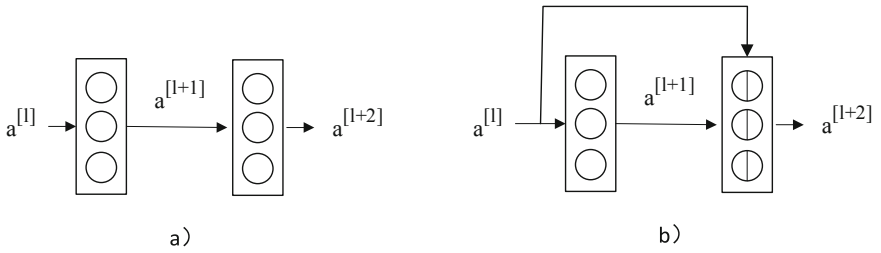

Fig. 3. Schematic diagram of ordinary network and residual unit

In Fig. 3, a) is a fragment of a plain network. The front and back layers of the Res Net network with the same feature map size. The necessary jump connections are added to each two layers to form a residual block. As shown in b), a shortcut is added. The network layer is between the l and l + 2 layers. This structure skips the feature matrix of the l + 1 layer and directly enters the l + 2 layer from the l layer for subsequent operations such as feature extraction. The residual network module learns $a^{[l+2]} = a^{[l]}$ is not difficult, so adding two jump layers to the neural network will not affect its performance.

3 Improved YOLOv3 Network Model

3.1 Brief Overview of YOLOv3 Algorithm

With the development of the concept of deep learning, the field of computer vision research and GPU computing performance continue to improve, allowing it to complete image analysis and processing tasks. Automatically marking the pedestrian bounding box position from the input video is the pedestrian target detection. This algorithm is the basis for intelligent analysis of pedestrian distance measurement, counting, and tracking in the nucleic acid detection video content of colleges and universities.

The YOLO algorithm proposed on the basis of deep learning in 2016 is an end-to-end learning method that inherits the algorithm idea of R-CNN, and performs classification and target regression in a convolutional network to greatly improve the detection speed and achieve for real-time target detection and multi-target detection tasks in the video field, the main idea of this algorithm is to divide the input image into S × S grid areas and detect the image whose center point falls into the grid. When YOLO is performing algorithm training, each cell with a target that is divided into S × S will select the prediction box with the largest IOU after the labeled data frame is compared to predict the target pedestrian object, making the other prediction bounding boxes of the grid indicates that the object is not included. However, the YOLO algorithm cannot achieve a very accurate prediction when the objects overlap. For example, when two objects fall into the same cell, only one object can be randomly selected for prediction.

3.2 Improved YOLOv3 Network Model

In the actual environment where the nucleic acid detection is concentrated in colleges and universities, it is found that YOLOv3 has high requirements for the GPU performance of the machine during training and prediction, and the actual detection effect is easily affected by external environments such as weather factors and scene layout. In order to further improve the universality of the algorithm, proposed to use depth separable convolution [11] to replace the convolution operation in the Res Net module, which greatly reduces the amount of model parameters.

Darknet-53 is a backbone network designed by YOLOv3 with 53 deeper neural network convolutional layers. It uses feature pyramid networks to design a multi-scale feature extraction structure. Its most notable feature is that it can perform target classification and location regression on three different scales. The network is composed of 3 × 3 and 1 × 1 convolutional layers, Res Net skip connection layer, upsampling layer for bilinear interpolation, feature fusion route layer and detection map output layer, etc. The three object detection processes are carried out by the 82nd, 94th, and 106th layers of the network. The input image sequence is divided into $S \times S$ grids, and a 52 × 52 × 255 detection feature map is finally generated for classification and the location is back. On the basis of the backbone network Darknet-53, it is proposed to use deep separable convolution to replace the convolution operation in the Res Net module. Under the condition of reducing model training parameters, the function of improving the YOLOv3 algorithm to accelerate the network training speed is realized. The improved Darknet- 53 network structure is shown in Fig. 4.

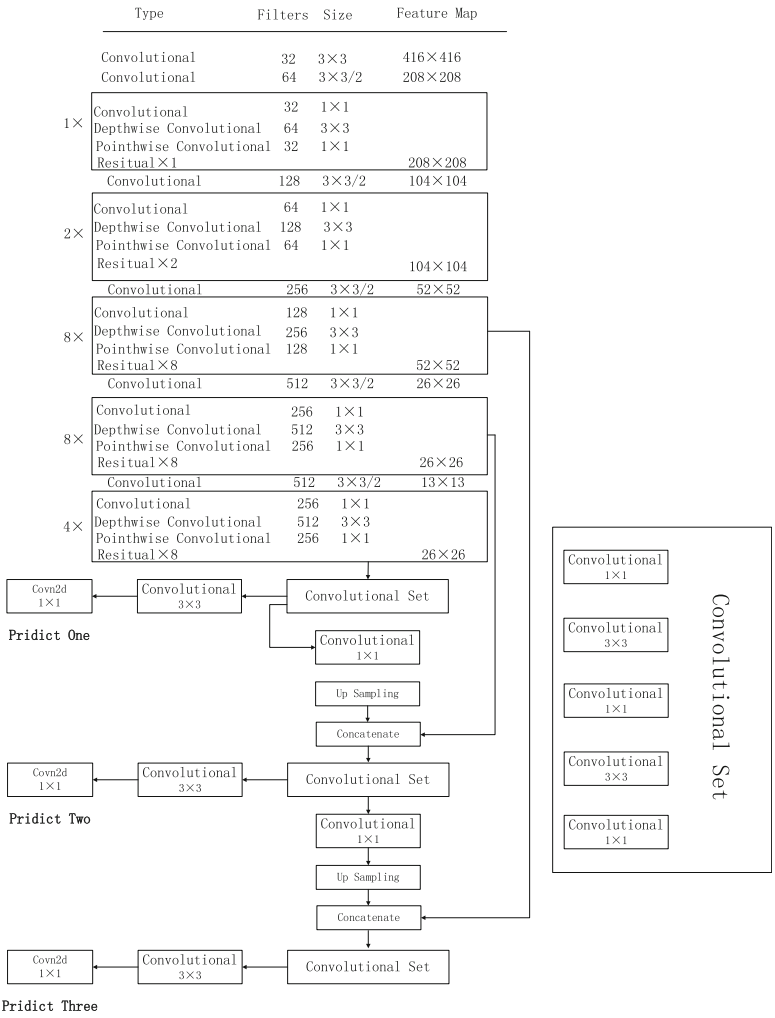

Fig. 4. Improved Darknet-53 network structure

Depthwise Separable Convolution extracts features by combining Depthwise (DW) and Pointwise (PW), reducing the amount of parameters and computing costs, and making the network more lightweight [12]. Channel-by-channel convolution performs independent convolution operations on each channel of the input layer, combined with point-by-point convolution, can further effectively use the feature information of different channels in the same spatial position, which is beneficial to the lightweight of the network. Improved YOLOv3 model the comparison before and after the feature extraction network is shown in Fig. 5.

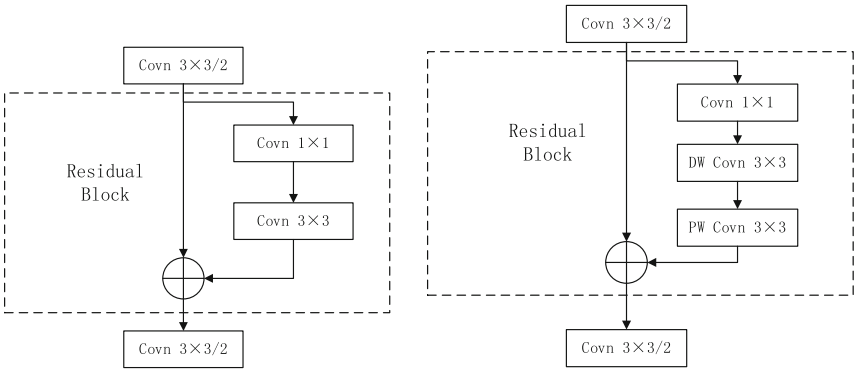

Fig. 5. Improved yolov3 model feature extraction network model structure

In order to realize the function of detecting the pedestrian detection, pedestrian tracking, pedestrian counting and pedestrian cluster system of collective nucleic acid detection in colleges and universities and maintain a reasonable order, a module for identifying and analyzing whether pedestrians have "cluster phenomenon" is added to the improved yolov3 network model. The specific output effect is shown in Fig. 6.

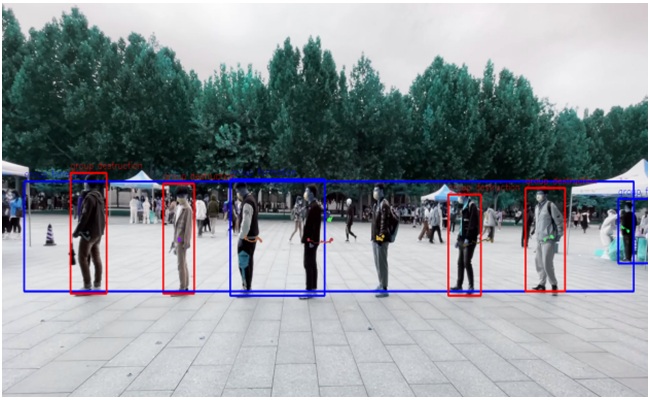

Fig. 6. Schematic diagram of "cluster phenomeno" output

This module mainly uses d(x, y): $=\sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$ to calculate the Euclidean distance between the person and the center point of the person, when the distance is less than twice its own width, it is judged as walking together, and the detection frames with cluster phenomenon are combined into a blue detection frame and output, marking group formation; people who are judged to be walking alone output a red detection frame, marking it as group destruction.

4 Experimental Environment Construction

4.1 Experimental Equipment and Environment

The experimental environment configuration is as follows: the motherboard is MSI B360M, the CPU processor is Intel(R) Core (TM)i9-10900K CPU, and the GPU is NVIDIA GeForce GTX 2080Ti. The computer software environment is: win10 operating system, anaconda-navigator1.9.12, python 3.7.6, TensorFlow 1.14, Keras 2.3.1, IDE is PyCharm 2019.3.2 (Community Edition).

4.2 System Specific Function Realization and Operation

First, perform data processing on the VOC2007 data set, filter out images containing pedestrians from the data set, modify the storage path of the experimental training data, run annotation.py to get 2007_train.txt and 2007_val.txt for training and verification, and then run check_data.py detects whether the data label is correct, and finally runs yolov_detector/trian.py to train the network to detect the location of pedestrians.

Train the deep residual network offline on the pedestrian re-recognition data set to train the DEEP SORT model, obtain pedestrian trajectories and count them through the Hungarian algorithm and Kalman filter, which are used to extract the features of the bounding box, and the appearance model is used to add rules for box matching. The appearance model adds rules for box matching, which can alleviate the occlusion problem and reduce the number of pedestrian ID switching.

Deep Sort uses a cascading matching algorithm. Each detector and each tracker that sets the time since update parameter can achieve one-to-one matching. The detection is constructed through the frame regression obtained by the bounding box learning, and the local maximum is filtered out by NMS. Finally, the probability of classifying the rectangular boxes that may be objects is screened out. After all the trackers of the previous frame are traversed and predicted, the results are stored and the effects of visualization, tracker update module, and feature set update module are achieved.

Pedestrian detection based on deep learning in the context of epidemic prevention and control in colleges and universities use PyQt to create a GUI application. Run main_ui.py under yolov3_ped_test. One-click to start the User Interface, click “Select Video” to retrieve the advance for the test video stored in the yolov3_ped_test folder, click “Start” to initialize the configuration of the running environment.

4.3 Experimental Results

After the configuration is complete, the detection system will start to detect, track, count and distinguish the cluster phenomenon of the input video. The total number of pedestrians displayed on the interface is the number of all pedestrians in the input video as shown in Fig. 7. The current number of pedestrians is the number of pedestrians in the video at this time. The number of pedestrians, each detected pedestrian will correspondingly output a detection frame and a corresponding color tracking line. The blue box indicates the cluster behavior of the detected object, that is, in the context of centralized nucleic acid detection in colleges and universities, there is a potential risk of

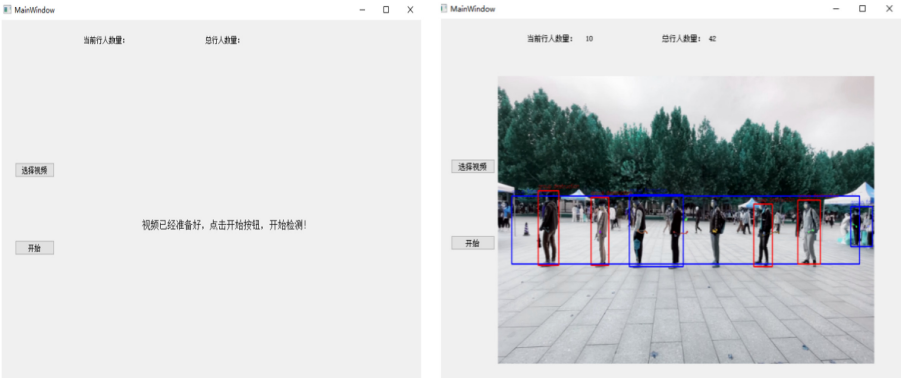

Fig. 7. User interface

cross infection in this cluster behavior. Relevant teachers and students should be notified to conduct nucleic acid detection at least at an interval of one meter or more according to the regulations.

The number of samples selected for one training of YOLOv3 is set to 16, and the Adam optimizer is selected. The initial learning rate is 1×10^{-3}. If the loss of the verification set does not drop for three consecutive times, the learning rate will be reduced to 0.1 times the original. Training for 100 epochs, after the training is completed, the pedestrian detection model training obtained by using tensorboard according to the training log is shown in Fig. 8. From the convergence curve, it can be considered that the network model training has reached the expected effect at this time.

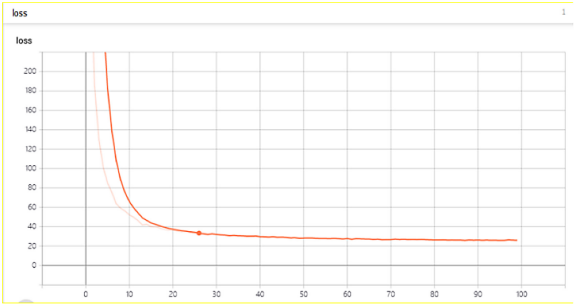

Fig. 8. Loss function curve

5 Conclusion

The pedestrian detection system based on deep learning in the context of epidemic prevention and control in colleges and universities implemented in this paper uses Depthwise Separable Convolution to improve yolov3 algorithm to effectively improve the accuracy and efficiency in a certain nucleic acid detection scene. Among them, the

parameter scale and test time are further lightweight, but in the actual scene test process, it will still be affected by some factors. For example, the actual detection background of college collective nucleic acid detection is complex and diverse; light intensity varies; pedestrian clothing is different; partial or total occlusion of pedestrians; the pedestrians to be detected cannot be accurately selected, and the detection of cluster effect will be affected by small target passers-by; pedestrian posture diversification and other influencing factors, which are also the main problems faced in this field at present. Therefore, in the future study and work, we will continue to actively learn relevant knowledge and skills in the field, improve the pedestrian detection, tracking, counting and cluster discrimination system, and improve the accuracy of pedestrian detection.

References

1. Du, P., Chen, M., Su, T.: Deep Learning and Target Detection. Electronic Industry Press, Beijing, pp. 2–25 (2019)
2. Pooja, G., Varsha, S., Sunita, V.: People detection and counting using YOLOv3 and SSD models. Mater. Today. Proc. **44**, 2069–2079 (2021)
3. Li, L., Guo, B., Shao, K.: Geometrically robust image watermarking using scale-invariant feature transform and Zernike moments. Chin. Optics Lett. **06**, 332–335 (2007)
4. Zhu, Q., Yeh, M.C., Cheng, K.T.: Fast human detection using a cascade of histograms of oriented gradients. In: 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1491–149. IEEE (2006)
5. Girshick, R.: Fast R-CNN. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 1440–1448 (2015)
6. Hubel, D.H., Wiesel, T.N.: Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. J. Physiol. **160**(1), 106–154 (1962)
7. 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 (NIPS), pp. 91–99 (2015)
8. Redmon, J., et al.: You only look once: unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, pp. 779–788 (2016)
9. Cao, S., Zhao, D., Liud, X.: Real-time robust detector for underwater live crabs based on deep learning. Comput. Electron. Agric. **172**, 105339 (2020)
10. 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 (CVPR), pp. 779–788 (2016)
11. Redmon, J., Farhadi, A.: YOLOv3: an incremental improvement. arXiv preprint: arXiv: 1804.02767 (2018)
12. Liu, W., et al.: SSD: single shot multibox detector. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9905, pp. 21–37. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46448-0_2