

Research on safety helmet wear detection algorithm based on YOLOv5s

Yingjian Xu

University of Wales Trinity Saint David

ABSTRACT In recent years, target detection has been applied in many fields, but target detection algorithms have problems such as large model, slow detection speed, and poor detection effect on small targets. In view of the existing problems, because the existing helmet wearing detection has strong interference and low detection accuracy, the helmet data set is tested, and a YOLOv5s-based modified helmet detection algorithm is proposed. First, use K-Means ++ to readjust the selected box to match the corresponding features; second, add the coordination attention mechanism (coordination attention) to the YOLOv5 backbone network to embed the location information into the channel attention, giving the network a larger regional attention capability. Images of different sizes were randomly entered during the training iteration process. In order to improve the generalization ability of the algorithm, the original feature pyramid module replaced with weighted bidirectional feature pyramid (BiFPN) network structure, achieve efficient bidirectional cross-scale connection and weighted features, finally, add a small target detection head, using four detection head to detect the target, effectively improve the feature extraction ability of small target. The effectiveness of the improvement is proved experimentally.

INDEX TERMS Deep learning; YOLOv5 algorithm; BiFPN; Kmeans++; Coordinate attention mechanism

I. INTRODUCTION

Target detection is one of the most challenging research topics in the field of computer vision. In recent years, thanks to the rapid progress of deep learning, object detection has made great progress, forming the basis for image processing tasks such as object tracking concave, image captioning^[1], and instance segmentation^[2-5].

Deep learning has been developed from artificial neural networks and is an important branch of machine learning. Neural networks originated in the 1940s, and researchers do something by simulating the human brain nervous system and making computers learn automatically. In the 1980s and 1990s, Rumelhart et al. proposed the back-propagation algorithm^[6], and neural networks began to become popular. However, the lack of large-scale data sets, of limited computer resources, and imperfect performance of machine learning algorithms. In 2006, Hinton et al. proposed that neural networks with multiple hidden layers can learn more effective feature information^[7]. Since then, deep learning has attracted wide attention in both academia and industry, and researchers have accelerated their research. The emergence of large-scale data set ImageNet^[8], the rapid development of Graphics Processing Unit (GPU), and the proposal of various excellent network structures such as AlexNet^[9], Overfeat^[10], GoogleNet^[11], VGG^[12], ResNet^[13], let researchers see the development potential of deep learning, and target detection has become a key research topic. Reviewing the history of the past two decades, the development of the target detection field is divided into two main periods. The pre-to-first period is the traditional target detection, representing the VJ detector^[14], the direction gradient histogram detector^[15], and the Deformable Part-based Model (DPM)^[16]. Since Convolutional Neural Networks (CNN) extraction features are more abundant and comprehensive than manual extraction, the object detection based on deep learning has developed rapidly. Usually, the detector design structure is different, and the object detection based on deep learning is divided

into two types: "two-stage detection" and "one-stage detection". In 2014, R. Girshick et al. proposed the R-CNN^[17], which pioneered the "two-stage detection". In the following years, the researchers have successively proposed the SpatialPyramid Pooling (SPP)^[18], Fast R-CNN^[19], Faster R-CNN^[20], FeaturePyramid Networks (FPN)^[21], etc. Since 2015, one-stage phase-based target detectors for deep learning have been proposed, typically for You Only Look Once (YOLO)^[22], Single Shot MultiBox Detector(SSD)^[23], DSSD^[24], RetinaNet^[25], M2Det^[26], RefineDet^[27], EfficientDet^[28], and some subsequent versions of YOLO,^[29-31]. Pedestrian detection, as one of the most important applications of target detection, has now played an important role in autonomous driving, pedestrian analysis, criminal investigation, intelligent monitoring; face recognition is one of the early computer vision applications, has been applied to "face swipe card", "smile" detection, beauty camera and other fields; text is the main carrier of information, so text detection can help people to read, travel, life more convenient; In recent years, remote sensing target detection has also become a research hotspot, widely used in military, disaster situation, urban management and other aspects.

Wearing safety helmets is an important guarantee for the safety of personnel in high-risk industries such as coal mines, steel and construction. Wearing safety helmets can effectively protect the head of construction personnel in time and prevent or reduce head injuries caused by the collapse or falling of objects on the construction site. In various complex construction sites, for some reasons, construction personnel often do not wear safety helmets according to relevant regulations, which not only endangers the safety of workers, but also affects the smooth progress of the construction process. For traditional helmet workers, most of them are based on manual supervision, but with the extension of supervision time and the expansion of supervision scope, the traditional supervision mode is easy to cause visual fatigue, leading to misjudgment and consuming a large amount of human resources. In terms of human resources, it cannot meet the current high-risk industry safety management requirements. Therefore, the research of safety helmet detection algorithm based on deep learning is of great significance.

II. RELATED WORK

2.1 Current status of research at home and abroad

Target detection has been developing for about two decades, which has also witnessed the development of computer vision. Driven by deep learning, target detection has gradually developed from the difficult embryonic period 20 years ago to the hot issues in the current era. Figure 2-1 shows the development roadmap of target detection in the past two decades. Traditional target detection algorithm was roughly proposed before AlexNet^[32] in 2012, and then deep learning-based target detection algorithm was followed this. As can be seen in Figure 2-1, the target detection is gradually moving closer to a stage, due to its unified detection structure, easy training, and saving a lot of memory resources and time overhead.

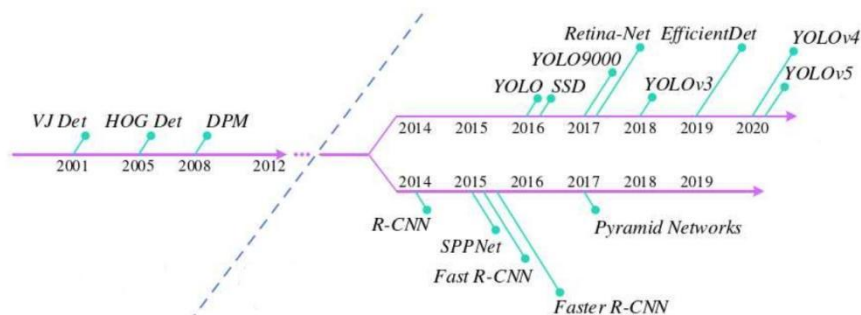


Figure 2-1 Development Roadmap for target detection

2.1.1 Traditional object detection algorithm

Traditional object detection consists of information region selection, feature extraction, and classification. Objects can be distributed in any position of the image, and have different shapes, categories, sizes, usually choose a multi-scale sliding window to scan the image to find out the possible location of the object, but selecting a large number of Windows caused a serious waste of computer resources, and will produce a lot of redundant Windows. In order to accurately classify and locate objects, need rich image feature information, such as Scale-Invariant Feature Transform (SIFT)^[33], direction gradient histogram, Haar^[34], but due to the diversity of background, brightness, appearance and so on, artificial features are difficult to express the information of the object to be detected. Finally, objects are classified according to feature information. Commonused classifiers include support vector machines ^[35], AdaBoost^[36], DPM^[37] and so on.

In 2001, P. Viola and M. Jones proposed the VJ detector, which is a human real-time face detector. In order to improve the detection speed, VJ detector adopts three techniques: using integral image, a calculation method to speed up the border filtering or convolution processing speed, make the calculation complexity of the window and window size; Adaboost algorithm to filter face feature information, the speed up the selection of effective features; in order to reduce the total calculation amount, introduce the detection cascade mechanism, saving the computer resources.

N. Dalal and B. Triggs in 2005 proposed the directional gradient histogram feature representation method, mainly improving on the SIFT and shape contexts^[38]. The principle of the feature extraction method is to calculate the gradient of pixels on the normalized image, then divide the image into cells and build the histogram, then combine the cells into blocks and normalize the gradient histogram, and finally collect the overlapping blocks for feature collection, combined to form the final feature vector. Direction gradient histograms lay an important foundation for the development of numerous target detector^[39].

In 2008, P. Felzenszwalb proposed DPM^[39], which has been improved by R. Girshick et al. for several years. DPM is one of the most successful detectors of traditional target detectors, which is mainly used in pedestrian detection. The DPM algorithm combines improved directional gradient histogram features, support vector machines, and sliding window detection ideas. Context startup, border regression, and difficult to negative case mining used in DPM still have a profound impact on the field of target detection.

On the one hand, the traditional object detection era lacks effective image representation, and can only rely on manually designed low-level feature representation and some shallow models to classify and locate objects; on the other hand, computer resources are scarce, which cannot

support the huge memory consumption and computation of the detection algorithm. Therefore, only improving some existing traditional algorithms can not bring a qualitative leap to the field of target detection. It was not until deep learning developed rapidly that object detection became a research hotspot.

2.1.2 Target detection algorithm based on deep learning

Since deep learning entered the second revolution since 2006, convolutional neural networks have developed rapidly. Compared with traditional methods, convolutional neural network has great advantages, such as convolution layer and pooling layer interweaving hierarchy, facilitating feature extraction, which contains both rich high-level semantic information and low-level geometric information; deep model has stronger information expression ability; it can be integrated into the same network structure. Therefore, the excellent convolutional neural network structure has an important influence on the development of object detection.

In 2012, at ImageNet LSVRC-2012, AlexNet was the first convolutional neural network model to revolution in computer vision. AlexNet has an eight-layer network structure that can learn more comprehensive features, using key technologies: abstention and data enhancement to slow overfitting, change the sigmoid activation function to Rectified Linear Unit (ReLU), and use multiple GPU to accelerate training. The excellent performance of AlexNet has brought the convolutional neural network to much attention from researchers in the field of deep learning. Following AlexNet, the VGG model was proposed by Oxford Oxford Visual Geometry Group in 2014. The VGG inherits the AlexNet framework, increasing the model depth to 16-19 layers, replacing the original 5x5 and 7x7-based convolution kernels, while using small convolution kernels of 3> 3, so that the depth is relatively easy to expand and the structure becomes simpler. At that time, VGG performed on the ImageNet dataset beyond other network structures. In the same year, Christian Szegedy proposed the GoogLeNet^[40]. Unlike AlexNet and VGG, GoogLeNet not only increases the depth and width of the network structure, but also designed the loss function in different layers, proposed the decomposition of convolution and batch normalization, which to a certain extent reduce the risk of overfitting, reduce the gradient disappearance, gradient explosion, reduce the number of parameters, efficient use of computer resources, improve the training effect.

In 2015, K.He et al. proposed ResNet, which is one of the most widely used convolutional neural networks. The previous networks improve the feature extraction ability by expanding the network depth and width of the network, and achieve some effects, but the network depth cannot be blindly deepened. ResNet applies the concept of residual expression to the CNN model to make the network automatically. The residual representation between input and output is learned to make model convergence faster and higher classification accuracy. G. Huang and Z.Liu, inspired by ResNet, proposed DenseNet^[41] in 2017. In DenseNet, the input of each layer comes from the output of all the previous layers, forming a tightly connected block, which enables the sufficient communication of the information between the layers, and is greatly helpful to the gradient information transfer. DenseNet This network structure is easier to train than other networks. In 2018, J.Hu and L.Shen proposed SENet^[42], et al. SENet notes that the feature channels in convolutional neural networks can be improved while convolution. The network structure that can be automatically learned to obtain the weight of each feature channel, and then strengthens useful features according to this weight or suppresses useless features. It is precisely because of these classic convolutional neural network models, which can extract more

comprehensive feature information than the traditional methods, so that researchers can build a better target detection model on this basis.

In 2014, R. Girshick et al. proposed the first two-stage target detection model, R-CNN, which broke the deadlock of the slow development of traditional target detection algorithms. The main idea of R-CNN is: first, use the selective search algorithm^[43] to generate about 2,000 area suggestion boxes for each image. Each candidate region is then scaled to a fixed resolution and input into the CNN model to extract high-level, semantic, and robust feature representations. Finally, the support vector machine is used to predict the existence of an object, and if so, it has to judge its category and use the frame regression algorithm to adjust the frame coordinates, and finally get the predicted position of the object to be detected. R-CNN also has some disadvantages: the selective search algorithm proposes many repetition boxes for each image, and excessive redundant calculation leads to extremely slow detection speed; the need to candidate. The area box scales to the same size, increasing the calculation tation.

In the same year, K. He et al. borrowed some ideas from Spatial Pyramid Matching (SPM)^[44], introduced the spatial pyramid pooling layer, and proposed SPPNet. The candidate region box of any size is pooled into different steps to convert to the feature vector of the same length. Therefore, there is no need to scale the region suggestion box, but only needs to convolve the whole image to extract the features, thus reducing a lot of redundancy calculation and improving the efficiency. The SPPNet is still a two-stage detector, and its training process is not fast enough. The FastR-CNN was proposed in the following year by R.Girshick et al. The main improvement ideas are as follows: not changing the relevant configuration of the neural network, the parallel training of the classifier and the frame regressor as multiple tasks. In addition, instead of extracting features for each region, but for the whole image, thus greatly reducing the redundant calculation and reducing the detection time. Shortly after the Fast R-CNN presentation, the Faster R-CNN was followed by S. Ren et al. Faster R-CNN is the first two-stage target detector close to real-time detection. Its main contribution is in proposing Region Proposal Network (RPN). The difference from previous target detectors is that using a sliding window to find regional suggestion boxes on the feature graph no longer relies on a selective search algorithm. Therefore, the time for generating candidate boxes is saved based on FastR-CNN. Although the detection speed of FasterR-CNN is almost real-time detection, the detector part is still not thin enough, and there is computational redundancy. Various improved network structures have emerged later, including the R-FCN^[45], Light-head R-CNN^[46], and Mask R-CNN, etc.

In 2017, the FPN was presented by T.-Y.Lin et al. Considering that the top-level feature map often is rich in high-level semantic information, but lacks more edge contour information, so it is not conducive to the object frame positioning. Unlike most target detectors, which detect only at the item layer, the feature pyramid network constructs a top-down model structure that integrates low-level information with high-level information to form a pyramid shape that detects objects on multiple scales and takes advantage of more fine-grained features. The FPN has now become the basic module for many advanced target detectors.

Compared with the two-stage target detector, which first proposes the candidate suggestion box, and then classifies and positions the object for the candidate suggestion box, the one-stage target detector adopts the end-to-end idea, input the whole image, and output the object category, confidence and coordinates. The whole process only needs- a convolutional neural network. This direct regression method eliminates many intermediate steps, greatly improves the

detection speed, and realizes real-time detection. Before the one-stage target detector was proposed, there were many researchers trying to model the target detection as a regression or classification task. Szegedy et al. expressed the target detection problem as a regression problem^[47] based on deep neural networks. Pinheiro et al. proposes a binary-branch convolutional neural network model, one that generates a segmentation mask for an agnostic class, and the other that predicts a class-likelihood^[48] at the center of a given object. Erhan et al and Szegedy et al proposed regression-based MultiBox that produce confidence^[49] for class-agnostic candidate proposal regions. Yoo et al. applied an iterative classification method to object detection and proposed the convolutional neural network structure AttentionNet^[50]. Najibi et al proposed G-CNN, the main contribution is to find the path^[51] from the fixed grid to tightly around the object border.

In 2016, R. Joseph et al. proposed the YOLO, the first one-stage target detector based on deep learning. YOLO completely abandoned the two-stage target detector "candidate suggestion box + prediction" mode. It only designed a convolutional neural network, input the whole image, and then divided the image into multiple regions, predicting the bounding box and category confidence for each region respectively. The YOLO model is a convolutional neural network, in which 24 convolutional layers and 2 fully connected layers can detect images in real time at 45 framespersecond (framespersecond, fps). The detection accuracy is lower than the most advanced two-stage detectors at that time, but it is better than other target detectors in the same period. In the same year, the SSD was proposed by W.Liu et al. Unlike previous detectors which only detect objects in the top feature maps, the SSD detects objects of different sizes in different layers. To discretize the output space of the bounding box, the SSD chooses to use anchor boxes with different length and width ratios. The network integrates multiple feature maps of different sizes to detect objects of different sizes. Not only could SSD achieve higher accuracy, but also three times faster than the then advanced two-stage detector. When the input image size is 300x300, the SSD can detect at 59 fps, which is more accurate and efficient than the previous stage detector YOLO. The following year, R. Joseph et al. adopted the -series strategy to improve YOLO and proposed a second version of YOLO9000. YOLO9000 not only makes some improvements to the detection algorithm, such as batch normalization, improved detector resolution, multidimensional clustering generation of anchor box, etc., but also proposes a new joint training method. Furthermore, YOLO9000 proposed the new backbone network, Darknet-19. Eventually, the YOLOv2 achieves higher detection accuracy and detection speed.

In 2017, T. -Y. Lin et al. proposed RetinaNet, introducing the new loss function of focal. The authors argue that the first stage in the two-stage detector filters out most of the negative positions, while the one-stage detector relies on the proposed dense candidate positions, facing new problems of extreme foreground and background class imbalance, resulting in decreased detection accuracy during training. The new loss function focal loss is designed to reduce the loss assigned to well-classified or simple samples, so the detector increases the proportion of the sample loss of difficult or wrong classification during training, which facilitates the model to pay more attention here. The use of focal loss can not only help the one-stage detector to reach a comparable level as the two-stage detector in terms of detection accuracy, but also without losing any detection speed.

In 2018, version 3 of YOLO, YOLOv3, was presented in R.Joseph et al. First, a multi-label classification with an independent logical classifier was selected. Second, multiscale feature maps

were used to predict bounding boxes, and the final convolutional layer predicts 3 D tensors to represent prediction categories, confidence, and bounding box coordinates. Then, drawing on ResNet, a better performing feature extraction network Darknet-53. Thanks to the design of multi-scale feature detection, YOLOv3 is more friendly to small object detection. Although the detection performance of small objects has been improved, the YOLOv3 detection effect for large or medium size objects is not very ideal.

In 2019, Mingxing Tan et al. proposed EfficientDet, a scalable model architecture for target detection based on EfficientNet^[52]. The main contributions of EfficientDet are divided into two aspects: one is to propose a new feature extraction method, BiFPN, and the other is to integrate a hybrid scaling method to mixed scale the backbone network depth and width of the detection network to adjust the network structure under different resource constraints.

The following year, the YOLOv4 was presented by Alexey Bochkovskiy et al. YOLOv4 was originally designed to be able to apply real-time object detection systems in work or life areas. The main contributions include three points: first, continue to study simpler and more efficient algorithms on the basis of the existing target detection algorithms; Then train to verify the influence of some recent skills and strategies on target detection; Finally, some advanced algorithms are simplified and optimized, For example, cross-iteration batch normalization (Cross-Iteration Batch Normalization, CBN), path aggregation network (Path Aggregation Network, PAN), the spatial attention module (Spatial Attention Module, SAM), Finally, it makes the model training more convenient, A excellent target detector can be trained with just a piece of GPU. Only more than 60 days, YOLOv5, its design Mosaic data enhancement, adaptive anchor frame calculation, Focus structure, etc., in the backbone network and neck part designed the two cross-stage local (Cross-Stage Partial Connections, CSP) structure, the design makes the model can lightweight, but will not reduce the accuracy of the model, at the same time improve the learning ability of the CNN. The YOLOv5 training speed is very fast and the model occupies very small memory, which reduces the deployment cost and facilitates the rapid deployment of the model.

In recent years, more and more methods related to target detection have emerged- -some excellent methods, algorithms, structures, etc., except, As described above, There is also RSDet^[53], R3Det^[54], for a dynamic anchor frame learning ^[55], Co-Mining^[56], YOL Obile^[57], StarNet^[58] ReXNet^[59], UP-DETR^[60], OPANAS^[61], Instant-Teaching^[62], Generalized Focal Loss V2^[63], Points as for arbitrary directional object detection Queries^[64], open-world-oriented target detection ^[65], YOLOF ^[66], attention mechanism ^[67] of mobile network, distillation method based on general instances ^[68], initial convolution ^[69] with effective void search, etc.

Nowadays, target detection is developing faster and faster, and both the accuracy and speed have reached a high level. As one of the most basic and important applications of target detection, pedestrian detection has played an important role in autonomous driving, pedestrian analysis, criminal investigation, intelligent monitoring and other aspects. Some traditional pedestrian detectors, such as direction gradient histogram detector, Integral Channel Features (ICF) detector [70], have opened up new ideas for the development of universal target detection. Classifier design, detection acceleration, and feature representation in traditional detection algorithms. Classical algorithms still have a profound impact on the field of target detection. Pedestrian detection is now integrated into some general target detection algorithms, and the detection effect has been greatly improved. Face detection is one of the predecessors of target detection applications. Early face detection effectively promoted the development of target detection, such

as VJ detector. Later, researchers proposed more and more advanced target detectors, and some of their design concepts were inspired by VJ detectors. Nowadays, people can directly feel many facial detection applications, such as beauty camera selfies, mobile phone application unlocking, and "face" recognition through the floodgate. Humans have been communicating and conveying information through text for thousands of years. The application scope of text detection is also particularly wide, for example, some people with visual impairment, can obtain some currency, road signs and other information through text detection; in the process of digital map creation, text detection can be used to obtain door numbers, road signs and so on. Testing makes human life more convenient. Deep learning promotes the development of autonomous driving. In the past, most people have only paid attention to universal object detection and ignored traffic-related detection. It is believed that this research will face more and more challenges. Remote sensing imaging technology can help people to more intuitively and conveniently understand the earth and understand the living environment. In recent years, with the improvement of remote sensing image resolution, remote sensing target detection has been more and more widely used, such as building, ship, fire detection, etc.

2.2 Main research content and chapter arrangement

In this paper, we propose an improvement scheme based on the safety helmet dataset on the model of the YOLOv5 target detection algorithm, and conduct the experiments. The specific chapters are arranged as follows. The traditional object detection algorithm has not a high accuracy for small objects, thus proposing several optimization methods.

- (1) The kmeans ++ algorithm was used to improve the prior boxes and improve the matching degree of the target box and the prior boxes.
- (2) The attention mechanism is added to improve the ability of the network results to extract the features of the target objects.
- (3) The original feature pyramid module of the feature fusion module is replaced with a weighted bidirectional feature pyramid (BiFPN) network structure to realize the efficient bidirectional cross-scale connection and weighted feature fusion.
- (4) Adding a small target detection head, and using 4 detection heads to detect targets, effectively improves the feature extraction ability for small targets.

2.3 convolutional neural network

2.3.1 Concepts of convolutional neural networks

Convolutional neural network is an image recognition algorithm based on deep learning, which has later been widely used in speech recognition, image and video classification. In the 1860s, Hubel and other scientists were inspired by the mechanism of biological visual perception, and proposed the concept of receptive field. By the 1980s, the scientist Fukushima proposed a neurocognitive machine model based on the sensory field, which was the first network model to successfully complete the convolutional neural network. The neurocognitive machine decomposes the transmission process of visual information into several sub-modes (features), extracts the visual information at all levels, and constantly presses it, and finally forms the visual information with a high degree of abstraction, and transmits it to the brain to complete the recognition of ^[71]. Since traditional machine learning algorithms rely on artificial design, they perform less than convolutional neural networks under deep convolution when dealing with complex tasks.

2.3.2 Structure of the convolutional neural networks

Typically, convolutional neural networks include the input, hidden, and output layers. The hidden layer also includes the convolution layer, the pooling layer and the full connection layer and other components. Figure 2-2 is a common convolutional neural network architecture, and the functionality of these underlying network structures is presented below.

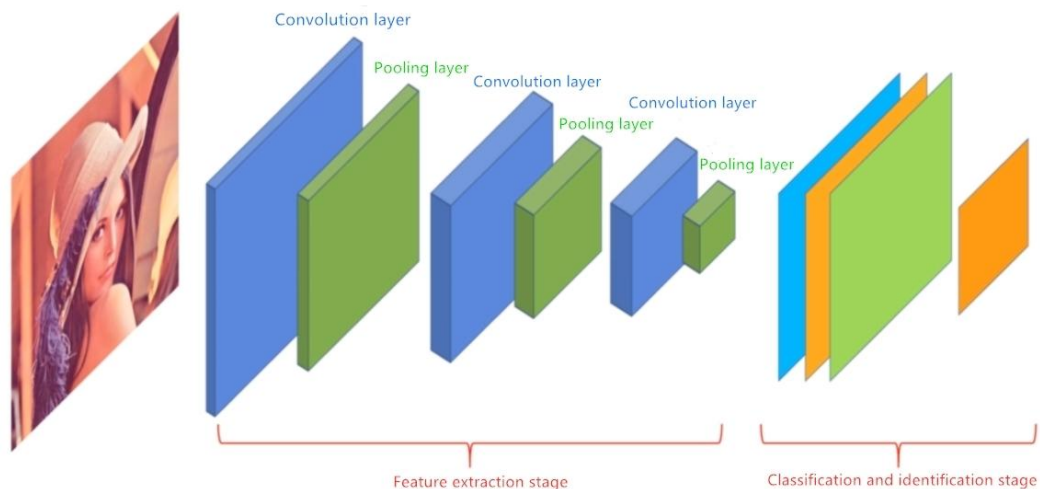


Figure 2-2 Convolutional neural network architecture diagram

(1) Convolutional layer

A convolutional network is a convolutional calculation of the input features and passing the results to the next convolutional layer, pooling layer, or fully connected layer, performing most of the operations. Convolution layer mainly contains two components, convolution kernel and activation function. Convolutional kernels are mainly used to extract the features of the input. In general, the convolution layer contains multiple convolutional kernels, and their receptive fields are very small. When the convolution kernel is used for convolution calculation, the 2 D activation map can be obtained by moving along the wide and high direction of the picture.

Figure 2-3 is an example:

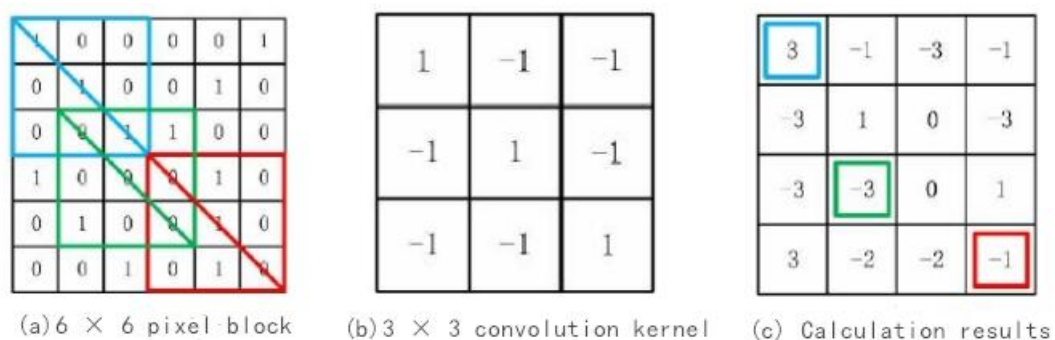


Figure 2-3 Example of the convolution operation

FIGS. 2-2 (a) is a 6X6 pixel block, while FIGS. 2-2 (b) is a 3X3 convolution kernel, performed with a 3X3 convolution kernel. The sliding step is 1 to convolution the input data and finally obtain the convolution result of 4X4.

After the image data is loaded in the input layer, the pixel matrix of the three channels (R, G, B) is sent into the convolution layer for convolution operation, and then add a bias on the corresponding results. Finally, the activation function is solved and mapped into a feature graph.

$$s(x) = g(\omega x + b) \quad (2-1)$$

In equation (2-1), x is the input data, it is the convolution kernel, b is the bias, and g (x) is the

activation function.

(2) pooling layer

After completing the convolution layer operation, the feature data obtained is usually very large. If these features are input directly to the next layer, it not only requires a large number of operations, but also produces a large number of repeated features, which affects the accuracy of identification. Therefore, the downsampling operation of the input feature map can reduce the resolution of the input image, and some information of the picture can be discarded, and then the remaining picture information can be sent to the neural network for the next step of training, in place of the whole image. Thus reducing the dimension and complexity of the feature, simply put, the pooling layer can essentially be regarded as a downsampling method of a nonlinear form. The two common function types are: maximum pooling and average pooling. The maximum pooling function is usually used for pooling, that is, taking the maximum value as the eigenvalue of the pooling layer, so as to achieve the purpose of dimension sampling. Figure 2-4 is an example of a pooling operation:

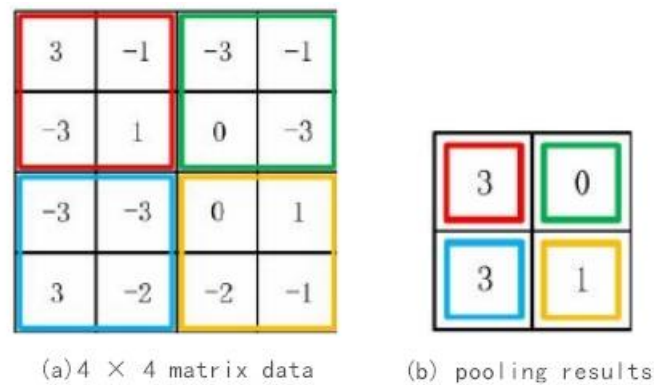


Figure 2-4 Schematic diagram of the pooling operation flow

In Figure 2-3, enter the feature diagram of 4×4 , select the 2×2 pooling block, select the maximum pooling function for pooling, and finally get the 2×2 matrix.

(3) fully connected layer

In convolutional neural networks, a fully connected layer is a hidden layer. The fully connected layer is generally behind the convolutional and pooling layers, the bottom of the entire network. Its main function is to integrate the feature information extracted by the convolutional neural network, and to integrate the local feature information into the overall feature information. The core work of the fully connected layer is to perform the linear transformation of the local feature information in the feature space, so as to classify the image by point-multiplying the feature matrix of the upper and lower layers. The data are then passed to the output layer for final computational identification, and the results are normalized at the output, so that the network can determine the probability of the various class characteristics of the input as 1. Figure 2-5 shows the full connection layer:

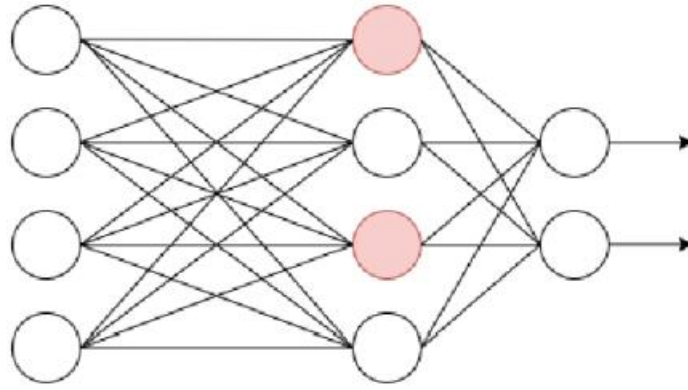


Figure 2-5, the full connection layer

2.4 Five target detection algorithms

2.4.1 SSD algorithm

In 2016, Liu et al. proposed a target recognition algorithm based on a single neural network, namely SSD (Single Shot MultiBox Detector) algorithm, which is the most commonly used detection method, which can locate and classify targets at a time. SSD mainly includes two aspects: VGG-16 convolutional neural network, and multi-scale feature object recognition network. The convolutional layer is the rectangular square and its type of information at each point in the prediction feature map; the features are obtained from different layers and predicted. With the increase of the hierarchy structure, the semantics becomes more complex. The SSD algorithm uses direct convolution to predict the feature map and obtain the results. Because of the unique network structure of SSD, the convolution kernel of 3×3 is used to detect the feature map of $m \times n \times q$, so that its computational amount is greatly reduced. SSD is detected by multi-scale feature maps, where large objects are detected by deep networks, while small objects are detected by ^[72] from shallow networks. Figure 2-6 shows the flow chart of the algorithm:

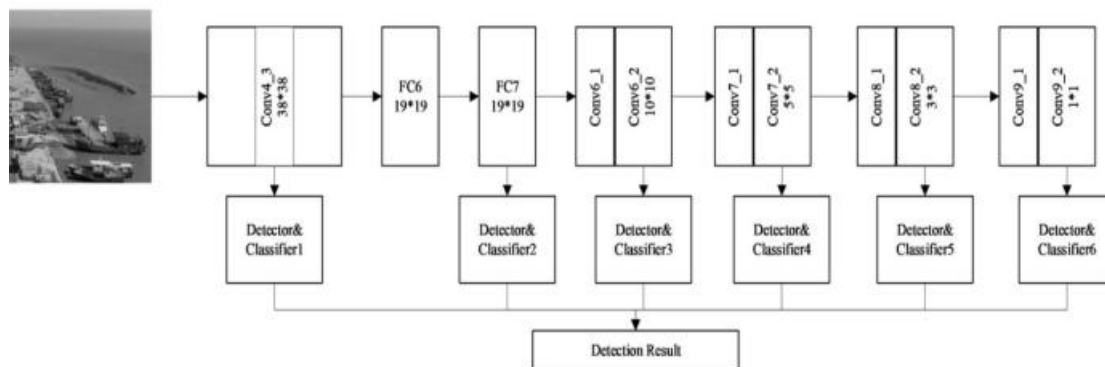


Figure 2-6 The S S D flow chart

The SSD algorithm has the following advantages: (1) the convolution layer in VGG-16 architecture replaces the convolution layer to increase the operation speed; (2) the void convolution to get more score mapping; (3) to adapt to predict multiscale targets; (4) the random tailoring method, so the method is very robust.

Lack of SSD algorithm: SSD refers to the candidate box in Faster R-CNN, To reduce the difficulty of the training session, Set fewer 4 prior boxes than Faster R-CNN, Although effectively increasing the speed of target detection, But as the hierarchy deepens, Shape-prone target semantics can become more abstract, Increase the difficulty of feature extraction; in addition, SSD algorithm

because to set the size of the anchor box manually, Do not match the true target size, Failure to effectively combine the global features, This leads to insufficient feature extraction of small targets, leading to missing detection.

2.4.2 Fast R-CNN

Fast R-CNN adopted the SPP Net method on the basis of R CNN, and made the improved target detection method that can be detected quickly later. The proposed method can better improve the detection accuracy and reduce the detection time. The Fast R-CNN includes four different structures: convolutional layer, pooling layer, full connection layer, and output layer.(1) Convolutional layer: input the picture in the convolutional layer, and then go through multiple convolution operations, and finally get the convolution feature graph. Because all fully connected layers require input data of the same size and have different feature size and shape for each candidate region, they are introduced into the ROI pooling layer for pooling.(2) ROI pooling layer: in each candidate region, fixed-size features are extracted from the mapping region of the feature layer respectively, and ROI Pooling s with different sizes are used for multi-scale feature extraction. In short, you can set a fixed size to determine the grid size of each sample, and then sample it to the maximum value. The ROI pooling layer pools each feature box into a fixed size, and outputs the feature vector of the corresponding number of candidate boxes.(3) Full connection layer: input the previously extracted feature vector into the full connection layer, non-combine the extracted features nonlinearly, and finally get the classification results.(4) Output layer: the feature vector obtained through the full connection layer mapping is decomposed into two output vectors: Softmax probability and the regression offset of each type of detection box, and then get the final identification result. As shown in Figure 2-7:

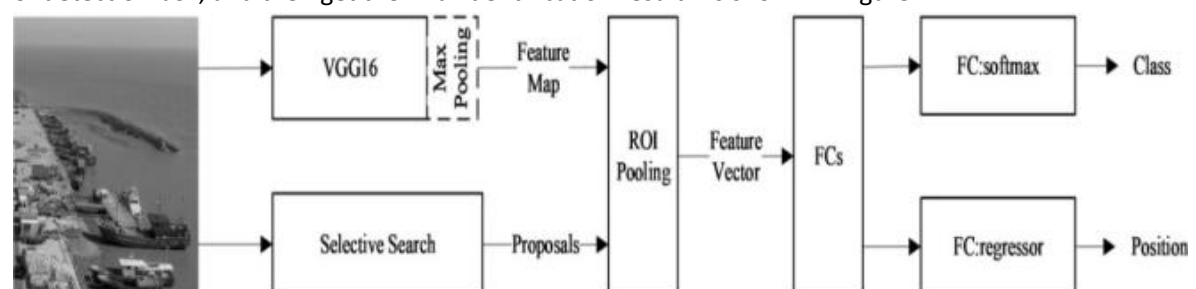


Figure 2-7 Fast R- -CNN flow chart

Advantages of Fast R-CNN algorithm: (1) using pooling layer improves accuracy; (2) avoids repeated feature operation and reduces network training and test time; (3) by introducing multi-task loss function (Multi-task Loss) to the CNN network, realize the unification of candidate region classification task and candidate box regression task, thus occupying no additional storage space.

Disadvantages of the Fast R-CNN algorithm: (1) because the selective search method is used when selecting regions. The input pictures need to be divided according to the color, shape, texture and other characteristics. After statistical analysis, a series of candidate regions are obtained. The process of selecting this area takes a lot of time, thus reducing the speed of detection.(2) The Fast R-CNN still has room for improvement in the detection speed, and it still does not achieve end-to-end target detection, such as the inability to synchronously acquire candidate regions.

2.4.3 Faster R-CNN

In 2016, Ross B.Girshick proposed the new Faster R-CNN. The algorithm uses Region Proposal

Networks (RPN) to replace the selective search algorithm, which effectively reduces the calculation time on the basis of the accuracy guarantee. Faster R-CNN mainly includes four aspects: convolutional layer, candidate box recommendation network, feature aggregation layer and classification layer. Where (1) Convolutional layer: input the whole image to the convolutional layer to extract the features of the picture, and the output extracted feature is called the feature graph.(2) Candidate box recommendation network: the generated feature map is transmitted to the RPN network for recommending candidate regions, and finally output as the top 300 scoring candidate regions.(3) Feature aggregation layer: the candidate regions of the top 300 score are feature extracted, and after the feature map is obtained, it is sent to the ROI layer, and finally the input of different sizes is converted into a fixed length output.(4) Classification layer: input the feature vector of the same size in the classification and bounding box regression network and classify the measured objects, and finally output which category the candidate region belongs to and the specific position in the image. As shown in Figure 2-8:

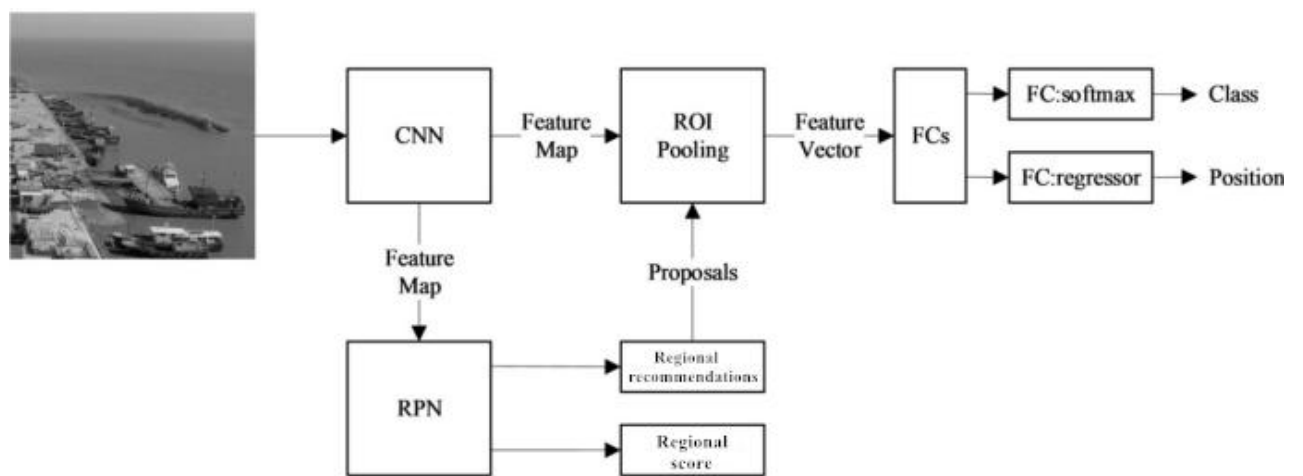


Figure 2-8 Faster R-CNN flow chart

The Faster R-CNN algorithm has the following advantages: (1) integrating feature extraction, candidate box extraction, frame regression, classification and other functions to improve its performance and accelerate the detection speed.(2) Compared with the traditional sliding window algorithm, the RPN network of extracting candidate regions has greatly improved the speed of generating detection boxes. This method can not only effectively solve serious time-consuming problems, but also effectively solve multi-scale problems.

The advantages of Faster R-CNN algorithm: the two stages based on the candidate area detection algorithm in detection accuracy and performance, but need to classify each candidate box, so need a lot of operations, and there is also the original image and feature map registration problem, thus affecting the accuracy of the model, and for small target detection task, its accuracy also has defects.

2.4.4 Cascade R-CNN

Cascade R-CNN is a multi-stage objective recognition algorithm proposed by Cai and Vasconcelos et al. There are two main improvements: first, the detection network of Faster R-CNN is improved. On this basis, we propose a cascade detection network [48] that raises the IOU threshold layer by layer; secondly, we replace the ROI pooling with ROI Align to keep the consistent size of the feature map. Cascade R-CNN mainly includes: feature extraction network, RPN network, ROI Align layer and cascade detection network.(1) Feature extraction network: the feature map is obtained

by inputting the acquired image into the network for feature extraction; (2) RPN network: the RPN network produces candidate regions that may contain targets; (3) Cascading detection network: classifying and locating three different thresholds for the input. As shown in Figure 2-9:

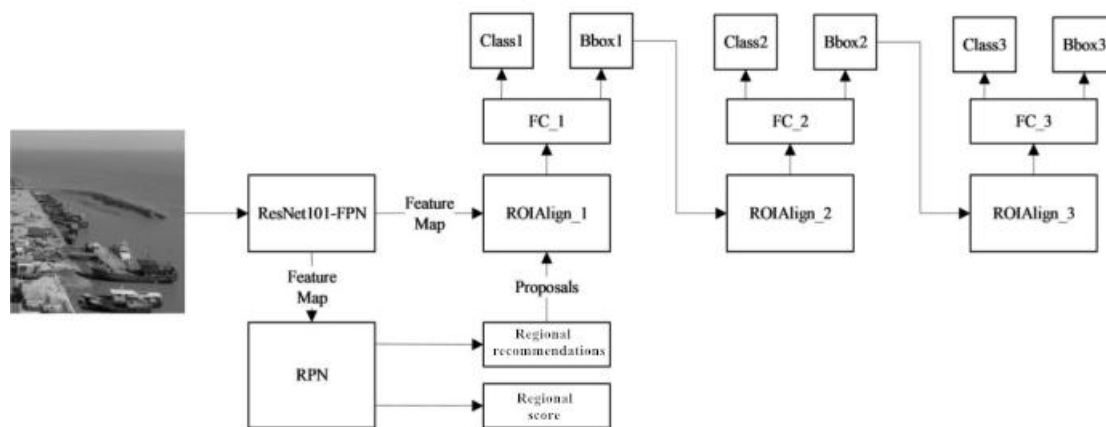


Figure 2-9 Cascade R-CNN flow chart

The Cascade R-CNN series three detectors with different IOU thresholds, corresponding to the respective IOU thresholds, thus avoiding the data overfitting problem caused by simply increasing the threshold of the IOU, and thus improving the detection accuracy. At the same time, the training and calculation time are greatly improved.

2.4.5 The YOLO series

The YOLO (You Only Look Once) algorithm is a regression-based target detection algorithm, whose core idea is to treat the object detection task as a single regression task. At its core is taking object detection as a simple regression task. The author then improved it, proposing YOLOv2 (and YOLO9000) ^[73] in 2017, through the introduction of BN layer and pooling mechanism. Therefore, information loss is avoided, and the positioning accuracy and recall rate of target recognition are improved. In 2020, someone proposed the YOLOv4 algorithm ^[74]. YOLOv4 adopted many optimization strategies, such as using Mosaic data enhancement on the input side, and adding PAN (Path Aggregation Network) to the original FPN (Feature Pyramid Networks). Later, Ultralytics proposed the YOLOv5 algorithm, which further improves the performance of the YOLOv4, and the model size of the YOLOv5 is only 1 / 9 the size of the YOLOv4, that is, when the structure is smaller, it can meet the real-time detection requirements. YOLOv5 is a strong real-time object detection algorithm, which includes four components: input terminal, backbone network (Backbone), Neck network, and detection head (Prediction). Taking the YOLOv5s network structure diagram 2-10 as an example, the image is first input into the Focus structure, then perform the slice operation, and then undergo a convolution operation of 32 convolution cores, and finally get the feature graph of $304 * 304 * 32$. Based on this data, the number of network layers was further increased by using the backbone network consisting of the Focus module, the CBL convolution layer, and the CSP1_X module. Therefore, rich semantic information feature maps are obtained, thus avoiding the problems of gradient disappearance and explosion. The module Focus is the ReOrg + Conv operation, which is also the reverse operation of the subpixel convolution. The biggest advantage of this module is the downsampling operation on the premise of minimizing information loss. In Neck network, CSP2_X and FPN + PAN models are used to make better use of the shallow detailed features ^[75]. Finally, the fusion of multi-scale receptive fields is realized, and the ability of feature fusion is enhanced, so that the

problem of small target feature loss is improved. For the detection head, a regression-based + classification method was used to divide the input images into three different size grids: 7676,3838, and 1919, realizing the detection of large, medium, and small targets, respectively. Compared with the traditional two-stage detection method, the YOLOv5s algorithm saves a lot of training time and improves the detection speed while improving the detection accuracy.

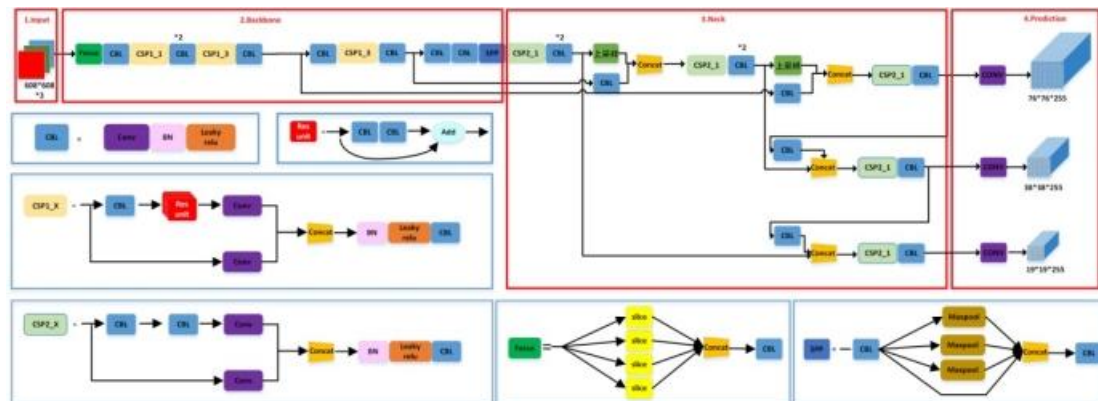


Figure 2-10 The YOLOv5s network structure diagram

Compared to previous target detection algorithms, all components in the YOLOv5 network are improved and improved. For example, mosaic data enhancement, adaptive anchor frame calculation, adaptive image scaling, focus structure and CSP structure are added to the backbone network on the input page, FPN and PAN structure is added to the neck network; The loss function of the target detection box on the output side is GIOU_Loss, and NMS non-maximum suppression method is proposed.

(1) The Input

The YOLOv5 input terminal mainly consists of three parts: mosaic data enhancement, adaptive compute anchor box and adaptive scaling image. Mosaic data is used to improve the model training phase. Combining four images through random scaling, random cropping, and random arrangement can increase the number of small targets in the dataset, improving the model's ability to detect small targets. The results are shown in Fig. 2-11 displayed. During the model training, yolov5 starts the adaptive anchor box calculation function according to the parameters to adaptively calculate the best anchor box value of different types of training sets. The network outputs the prediction box according to the original anchor box, then compares the difference between the prediction box and the real box, and then vice versa updates the iterative network parameters to display the prediction result diagram shown in Figure 2-12. Adaptive zoom image means zooming the original image to a uniform standard size and then sending it to the network for detection. The grey border around the image in Fig. 2-11 is the result of adaptive scaling of the image.



Figure 2-11 Mosaic data enhancement renderings



Figure 2-12 Initial box gets the effect diagram of the prediction box

(2) Trunk Network (Backbone)

The backbone of the model consists mainly of the Focus, C3 (improved Bottleneck CSP) and Spatial Pyramid Pooling (SPP) modules. Focus is a slice operation of the feature map that integrates the information about the width w and height h of the image into the c dimension, in particular stacking four positions with a distance of 2 onto a channel, so that both height h and width w are reduced to $1/2$ and the number of channels c is increased $4\times$. The specific process is shown in Figure 2-13. The focus module is designed to reduce FLOPS (number of floating point operations per second) and improve detection speed. The innovation of the YOLOv5-P5 model is that the newly supported SiLU unlock function in PyTorch1.7 replaces the LeakyReLU() and Hardswish() activation functions used in the previous version and cuts out part of the Conv module of BottleneckCSP in the previous version. The comparison diagram of the improved BottleneckCSP module and the improved BottleneckCSP module (C3) is shown in Figure 2-14. The SPP module uses three combined, multi-scale maximum pooling layers that can lift the receptive field almost without speed reduction to solve the alignment problem of the anchor frame and feature layers.

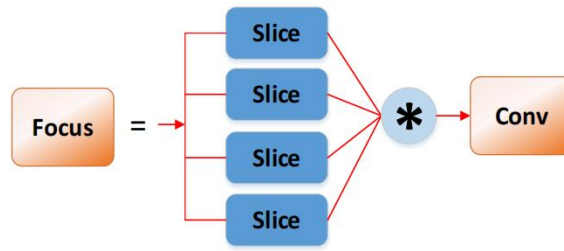


Figure 2-13 Focus slice manipulation

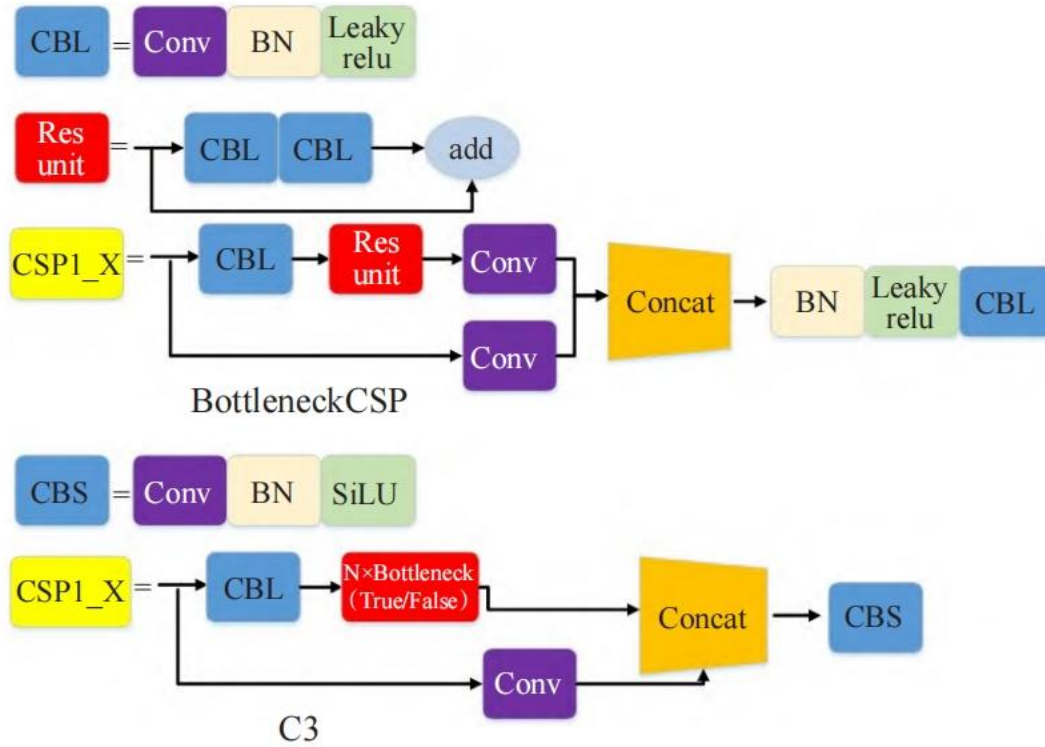


Figure 2-14 Structural diagram of BottleneckCSP and C3 modules

(3) The Neck

The module adopts Feature Pyramid Networks (FPN) + Pathaggregation Network (PAN) ^[18] structure, which can strengthen the network's ability to integrate the characteristics of objects with different scales of scale. FPN mainly improves the detection effect of small target objects by using the properties of high and low layer features. PAN adds a bottom-up path to the FPN so that the underlying position information is better transferred to the top level. The combination of the two can aggregate different detection layers from different backbone layers, and this structure can further improve the detection performance of dense targets.

(4) The Output

The output uses the GIoU function as the loss function of the bounding box. In the post-processing of target detection, non-maximum inhibition (NMS) is used to screen multi-target boxes, which improves the detection capability of multi-target and blocked targets.

2.5 Experimental results and analysis

2.5.1 Hard hat dataset

During the classification training, the convolutional neural network will input the preset category-labeled images into the convolutional network. After training, the output result is error compared with the set class tag. The classification power of the convolutional neural networks is

improved by the continuous optimization of the algorithm. Therefore, a data set containing hard helmet objects and their labels needs to be established before training on flame recognition. For the correct prediction, the neural network needs a lot of data to conduct systematic training. This paper is based on the SHWD safety helmet wear and human head detection data set, and uses the network to crawl the images for data enhancement, as shown in Figure 2-15. The number of images was increased by random cutting, rotation, cutting, and finally annotated using labeling, and divided into two major categories: hat and person. The total number of pictures in the data set was 8,478 pictures, which were divided into training and validation data sets, respectively, with a ratio of 8:2. The dataset is divided into no helmet image and unworn images, no helmet image is known as positive sample, and no helmet image is known as negative sample.



Figure 2-15 Example of the data augmentation

2.5.2 Experimental environment and evaluation index

(1) Experimental environment

The operating system for this experiment was Windows10 with 32GB of running memory. The GPU model is NVIDIA RTX3070, 8 GB and 24GB. The YOLOv5 model is based on the Pytorch deep learning framework, the programming language is Python, accelerated GPU using CUDA11.3.1.1 and CUDNN8.2.1, batch size set to 16, learning rate set to 0.01, and training at 300epoch. The specific experimental parameters table is shown in the table below.

Table 2 Experimental parameters

parameter	numerical value
Lr0	0.01
lrf	0.2
momentum	0.937

warmup_epochs	3
Batchsize	16
Weight decay	0.0005

(2) Performance evaluation index

In this experiment, the model evaluation indicators used were as follows:

1. Combine it with IOU

The intersection has an important reference value than IOU. IOU represents the coincidence rate of the candidate block and the original marker box, when IOU is 1, and the candidate box is completely consistent. The calculation formula is shown in (2-2):

$$IOU = \frac{A \cap B}{A \cup B} \quad (2-2)$$

The intersection ratio presents a positive proportion to the detection results, that is, as the intersection ratio increases, the distance between the prediction box and the real marker box of the target will get closer and closer. When the degree of coincidence between the actual marking box and the prediction box increases, it shows that the method can accurately frame the tested object, and the higher the accuracy of the annotation, the better the effect of the method is. In general, when the algorithm iterates repeatedly, its intersection ratio is gradually close to 1.

2. Precision rate (Precision, P) and recall rate (Recall, R):

$$P = \frac{TP}{TP + FP} \quad (2-3)$$

$$R = \frac{TP}{(TP + FN)} \quad (2-4)$$

In the formula, TP (True Positive, TP) is the positive detection rate, indicating the number of correctly detected targets; FP (False Positive, FP) is the false detection rate, indicating the number of wrongly detected targets; FN (False Negative, FN) is the missed detection rate, indicating the number of missed detected targets.

3. Mean mean accuracy mAP (mean Average Precision, mAP)

Average accuracy AP (Average Precision) refers to the mean value of the accuracy rate at different recall rates, with the formula: (2-5):

$$AP = \int_0^1 P(R) dR \quad (2-5)$$

The mAP value is the mean of the AP value, which is the average accuracy of all categories. The calculation formula is: (2-6):

$$mAP = \frac{\sum_{i=1}^N AP_i}{N} \quad (2-6)$$

In the formula, n indicates the category, and N indicates the total number of categories. Since the fire detection is a single target detection task, and the values of the average accuracy (AP) and the mean average accuracy (mAP) are equal, the mAP is used as the index to evaluate the model performance in the experiment.

4. Detection speed (Frames Per Second, FPS):

The detection speed FPS represents the number of images detected per second, namely the number of frames.

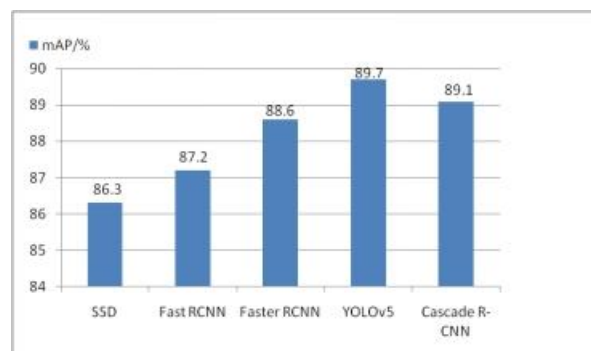
2.5.3 Results of the five detection algorithms

To validate the performance of the five different methods, we tested five different methods, including SSD, Fast R-CNN, Faster R-CNN, YOLOv5, and Cascade R-CNN, in the safety helmet dataset, and compared the experimental results. The analysis of the average detection and average detection accuracy of mAP in Figure 2-11 and the detection speed of the five algorithms in Figure 2-12 is available:

1) In terms of the average accuracy, the Fast R-CNN, Faster R-CNN, and Cascade R-CNN algorithms all have a higher average accuracy than the one-stage target detection SSD algorithm. In the safety helmet target detection, the highest average accuracy mAP among the five algorithms is the YOLOv5 algorithm, with values of up to 89.7%.

2) In terms of the detection speed, the detection speed of the two-stage and multi-stage detection algorithms Fast R-CNN, Faster R-CNN and Cascade R-CNN are much lower than those of the single-stage detection SSD algorithm. In fire target detection, the fastest detection speed among the five algorithms is the YOLOv5 algorithm with a detection speed of 48.4FP.

3) Combined with the performance of the average accuracy and the detection speed, we can get the conclusion that the YOLOv5 algorithm is the strongest comprehensive performance. It can be seen that the YOLOv5 algorithm's detection speed is also much faster than the other algorithms, which is suitable for safety helmet detection scenarios.



Figures 2-11 Five algorithms detect mean mean accuracy mAP

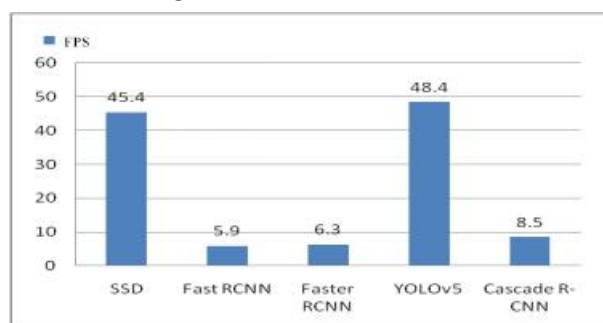


Figure 2-12 Comparison diagram of the detection speed of the five algorithms

2.6 Subsection of this chapter

This chapter expounds the basic principles and structure of convolutional neural networks and summarizes five commonly used target detection models. Firstly, the main structure of the network is introduced; then, the network structure of YOLO series, SSD, Fast R-CNN, Faster R-CNN,

Cascade R-CNN models and their advantages and disadvantages; finally, the experimental environment, the helmet data set used and the used algorithm evaluation index are briefly introduced, the five algorithms are applied to helmet detection, and the training results are analyzed. The results show that the YOLOv5 algorithm has the best detection results and the highest detection speed, but its detection accuracy for small targets still needs to be further improved.

III. PROPOSED MODEL

In the previous chapter, safety helmet detection was performed using five target detection algorithms, drawing the conclusion that the YOLOv5 algorithm detects the best. However, the YOLOv5 algorithm still has the disadvantage of low recognition rate of multi-target for small targets and complex environments. Therefore, this chapter proposes a new fire detection improvement algorithm based on the YOLOv5 algorithm. First, the K-means algorithm in the original YOLOv5 replaces the K-means ++ algorithm for clustering, which reduces the effect of the initial value on the clustering effect, improves the average IOU between the prior box and the real box, and thus improves the recall rate of the algorithm. Secondly, the CA (coordinate attention) attention mechanism is used to join the backbone network to improve the feature extraction ability of the network structure; replace the original feature pyramid module with a weighted bidirectional feature pyramid (BiFPN) network structure to realize efficient bidirectional cross-scale connection and weighted feature fusion. Finally, a small target detection layer is added, and four detection headers are used to obtain multi-scale information, which further improves the accuracy of multi-scale and small targets, and further improves the identification accuracy in complex scenarios.

3.1 The K-means ++ algorithm improves the anchor frame mechanism in YOLOv5

Target recognition is to use the boundary box to locate the measured object, and to automatically modify the coordinate information of the positioning box accordingly. In the early stage of training, by selecting the appropriate prior boundary box, the network model can be easier to obtain better prediction results.

The YOLOv5 model introduces the candidate region box (anchor box) into the process of target identification. Among them, anchor box is a set of initial regions with fixed size and aspect ratio. During training, the closer the initial candidate box is to the real bounding box, the easier the model will train, and the more accurate the prediction bounding box will be. Therefore, the anchor parameters in the YOLOv5 model should also be adjusted appropriately according to the dataset. For the YOLOv5 model network structure, the wide and high dimensions of 9 cluster centers need to be found as the value of the anchor parameter, so the parameter setting of 9 sets of initial candidate boxes can be implemented through the K-Means clustering algorithm. Traditional K-means clustering algorithm adopts Euclidean distance as the metric function, but when the wide and high gaps of different samples are relatively large, using Euclidean distance leads to large errors in clustering results. While the anchor box clustering aims to get a prior box with higher IOU values between the actual box and the prediction box, so the YOLOv5 instead uses the IOU distance as a metric function in the clustering process.

IOU is the quantity to measure the similarity of two boxes (including position, width, and height similarity). It is the ratio of the area of two boxes to the area of the phase, so it is also called intersection ratio. As shown in Figure 3-1.

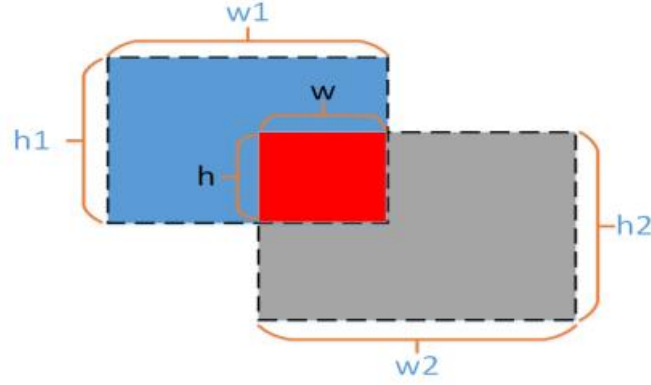


Figure 3-1: I O U intersection and union ratio

As shown in the figure above, the width and height of the two boxes are $(w1, h1)$ and $(w2, h2)$, the red area is the intersection area of the two boxes, its width and height are (w, h) , the "blue + red + gray" area is the phase junction area of the two boxes, then the area of the intersection area is formula (3-1):

$$S_j = w \times h \quad (3-1)$$

The area formula (3-2) is:

$$S_b = w1 \times h1 + w2 \times h2 - w \times h \quad (3-2)$$

Then you get the IOU:

$$IOU = \frac{S_j}{S_b} \quad (3-3)$$

The value range of IOU is $[0,1]$, IOU takes 0 when the two boxes have no intersection area, and IOU takes 1 when the two boxes are completely overlapping. So the larger the IOU value, the more similar the shape and position of the two boxes are. In order to negatively correlate the measurement value and the similarity, that is, the smaller the measurement value, the greater the similarity, we take a negative value of the IOU value plus 1 to obtain the formula (3-4):

$$d_{iou} = 1 - IOU \quad (3-4)$$

The K-means algorithm resulted in 9 prior boxes, respectively corresponding to the feature plots of different scales, and the prior boxes are shown in Tables 3 – 1, respectively.

Table 3-1 Prior box distribution after clustering using the K-means algorithm

Feature map size		Anchor frame size	
80×80	(8, 10)	(11, 13)	(15, 18)
40×40	(24, 27)	(36, 42)	(54, 64)
20×20	(87, 101)	(155, 183)	(267, 336)

The K-means clustering algorithm is characterized by fast convergence and linearly increasing computational complexity. However, there are also two disadvantages: the first disadvantage is that the number of K of cluster centers must be set in advance, but the selection of this K value is very difficult to predict in practice, but in most cases, a specific data set can be divided in advance; the second disadvantage is that the initial clustering center of K-means must be manually determined, and different initial clustering centers may lead to completely different clustering effects.

For the second drawback of K-means algorithm, K-means ++ algorithm is used. Compared with the traditional K-means clustering methods, the K-means ++ algorithm can effectively optimize when selecting the initial points, and can get Anchor box more adapted to small targets, thus improving the identification accuracy of small targets.

The central idea of the K-means ++ algorithm for the initial cluster center selection is: the mutual distance between the initial cluster centers should be far enough. The initial mean vector selection steps are as follows:

K-means ++ cluster anchor frame step

- (1) randomly select a sample as the first cluster center;
- (2) Calculate the shortest distance between each sample and the current existing cluster center (the distance of the nearest cluster center), as expressed by $D(x)$;

Then, the probability that each sample is selected as the next cluster center is calculated:

$$\frac{D(x)^2}{\sum_{x \in X} D(x)^2} \quad (3-5)$$

A greater this value indicates a greater chance that the sample is selected as a cluster center;

- (3) Select the next cluster center by the wheel wheel method; repeat step (2) until the k cluster centers are selected

The incoming prior box distribution of the cluster anchor box using K-means ++ is shown in Tables 3 – 2.

Table 3-2 Prior box distribution after clustering using the K-means algorithm ++

Feature map size		Anchor frame size	
80×80	(8, 10)	(10, 12)	(12, 14)
40×40	(15, 18)	(21, 24)	(29, 33)
20×20	(42, 48)	(67, 77)	(120, 143)

The improved matching degree between the prior box and the dataset target was increased from 0.8553 to 0.8689, reducing the impact of the initial value on the clustering effect, and improving the average IOU between the prior box and the real box.

3.2 Improvement of the backbone network

For small targets in the image, there are fewer pixels, and they are easy to be influenced by background factors. For example, the original yolov5 model algorithm is easy to lose small target feature information in folding sampling, and the small target detection effect is not very good. Therefore, the coordinate attention mechanism is introduced in this paper to tell the model what content and position to pay more attention to, what is suitable for small eyes. For standard and density targets, feature information can be extracted effectively to further improve detection accuracy.

At present, the attention mechanism of lightweight networks mostly adopts SE module, which only considers the information between channels and ignores the location information. Although later BAM and CBAM tried to extract positional attention information by convolution after reducing the number of channels, convolution can only extract local relations and lacks the ability to extract long-distance relations. To this end, HOU Q B, ZHOU D Q, FENG J S proposes a new efficient attention mechanism, coordinate attention ^[76], which can encode the horizontal and vertical location information into channel attention, so that mobile networks can pay attention to

a large range of location information without bringing too much computation.

The advantages of coordinated attention mainly include the following:

(1) Not only the inter channel information is obtained, but also the direction related position information is considered, which helps the model to better locate and identify the target.

(2) It is flexible and lightweight enough to be easily inserted into the core structure of the mobile network.

(3) It can be used as a pre training model in a variety of tasks, such as detection and segmentation, and has a good performance improvement.

Attention coordination (CA) ^[76] adds location information to the attention channel so that the network can focus on a larger area, in order to reduce the information loss caused by the two-dimensional global location clustering proposed by the attention mechanism. Attention channel decomposes attention into two parallel one-dimensional feature coding processes and two directional feature clustering processes, one direction obtains distance dependence and the other direction obtains accurate information of distance dependence. The generated feature map is encoded to form a pair of direction sensors and position sensitive features.

As shown in Figure 3-2, CA encodes the channel relationship and the long-distance dependence relationship through the accurate position information, which mainly includes two steps: coordinate information embedding (coordinate information embedding) and coordinate attention generation (coordinate attention generation). A CA module can be viewed as a computational unit used to enhance the feature representation power, which can take any intermediate tensor $X = [x_1, x_2, \dots, x_c] \in \mathbb{R}^{C \times H \times W}$ as input and output an output of the same size with an enhanced representation capacity $Y = [y_1, y_2, \dots, y_c]$. Where C is the number of channels, h and w are the height and width of the input pictures, respectively ^[76].

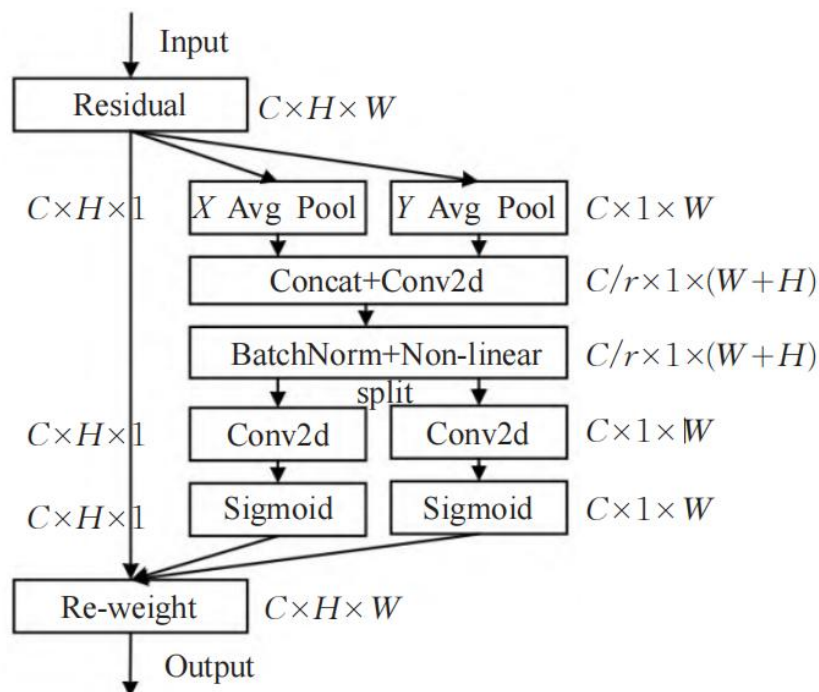


Figure 3-2 The CA module structure

To embed coordinate information, the CA module divides the global pool into one- to one-dimensional feature encoding operations to promote spatial dependencies over long

distances where the attention module can obtain accurate location information. For the input feature map x , the dimension $c \times h \times w$ is. First, the merged cores with sizes $(h, 1)$ and $(1, w)$ are used along the horizontal and vertical coordinates, i.e. the C channel with height h and the C channel with width w . The output formula is displayed in equation (1) and equation (2) [76]:

$$Z_c^h(h) = \frac{1}{w} \sum_{0 \leq i < w} X_c(h, i) \quad (1)$$

$$Z_c^w(w) = \frac{1}{H} \sum_{0 \leq j < H} X_c(j, w) \quad (2)$$

These formulas realize bidirectional clustering of features, and return the features of interest Z^h and Z^w to a pair of directions, which is helpful for more accurate target recognition.

In order to coordinate the generation of interest, the cascaded mr module is formed before the two feature layers, and then a total of 11 convolutional transformations $F1$ are used. Formula (3) is as follows [76]:

$$f = \delta(F_1([Z^h, Z^w])) \quad (3)$$

Where $f \in \mathbb{R}^{C/r \times (H+W)}$ is the intermediate feature map of spatial information in horizontal and vertical directions, R represents the subsampling rate and is set to 16 to control the size of the module. δ Represents a nonlinear activation function. Then the spatial dimension is divided into two separate tensors $f^h \in \mathbb{R}^{C/r \times H}$ and $f^w \in \mathbb{R}^{C/r \times W}$ repeatedly uses the two 11-folds F^h and F^w to convert the feature maps f^h and f^w into the same number of channels as the input x and gets the results in equations (4) and (5) [76]:

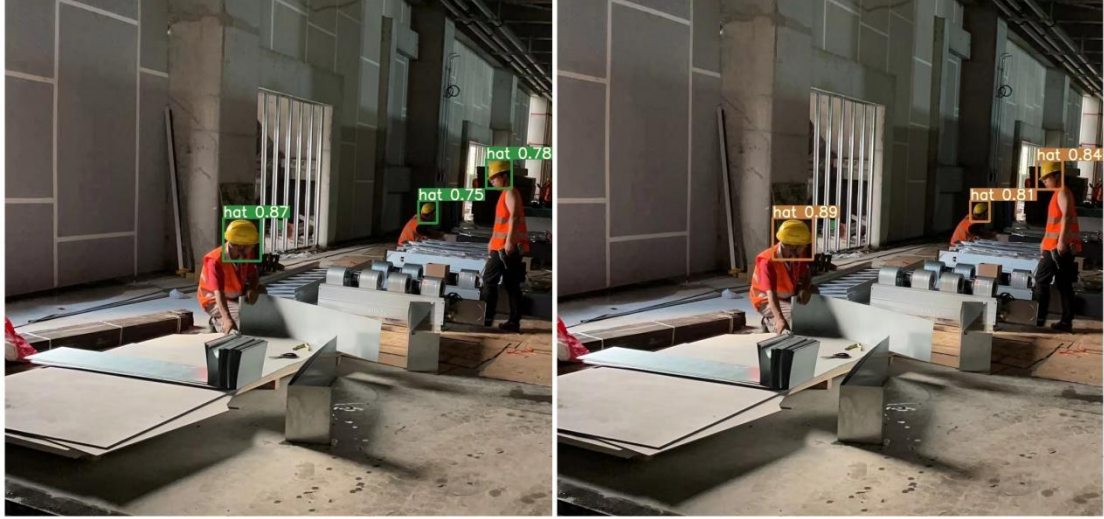
$$g^h = \sigma(F_h(f^h)) \quad (4)$$

$$g^w = \sigma(F_w(f^w)) \quad (5)$$

Finally, g^h and g^w are extended. For weights, the final output of CA module can be expressed as (6) showed [76]:

$$y_c(i, j) = x_c(i, j) \times g_c^h(i) \times g_c^w(j) \quad (6)$$

Coordinated attention is a simple and flexible plug and play module. It is a new mobile network attention mechanism, which can improve the accuracy of the network even without increasing any cost [76]. Because of these advantages, this paper adds the CA module to the yolov5 backbone model and inserts it behind the CSP module. The actual detection results of the network structure after the improvement and before the improvement are shown in Figure 3-3: (a) the test results of the original yolov5 model, and the detection confidence obtained in the figure is 0.87, 0.75, and 0.78 respectively. (b) the actual detection results of the CA attention module inserted into the yolov5 model are shown in Figure 3-3, The detection confidence obtained in the figure is 0.89, 0.81 and 0.84 respectively. It can be seen from the figure that the attention mechanism can make the network model pay more attention to the target and improve the feature extraction ability of the network.



(a)Original YOLOv5s model detection results (b) the detection results after the addition of the attention mechanism

Figure 3-3 Comparison of the model detection effect before and after the addition of the attention mechanism

3.3 Improvement of the characteristic pyramid structure

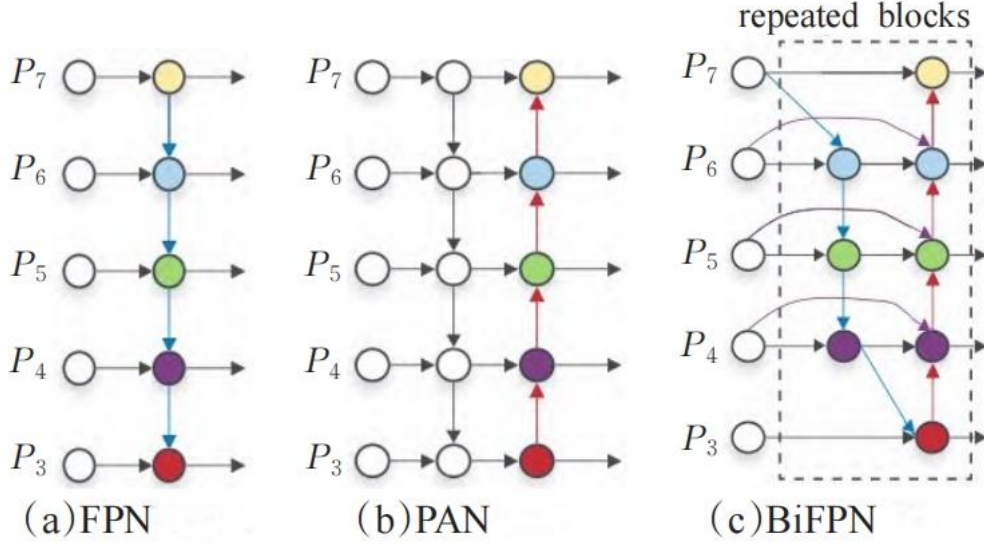
The BiFPN (bidirectional feature pyramid network) -weighted bidirectional (top-down + bottom-up) feature pyramid network structure^[28] is a brand new feature fusion method proposed by the Google Brain team.

The idea of BiFPN is an efficient bidirectional cross-scale connection and weighted feature fusion, that is, based on the idea of path enhancement, see Figure 3-4, first, top-down feature fusion, and then bottom-up feature fusion. FIGS. 3-4 (a) show the FPN structure that establishes a top-down pathway for feature fusion, which can then be predicted by a fused feature layer with higher semantic information, but this structure is limited by one-way information flow.

In order to solve this problem, Liu et al. proposed figure 3-4 (b) of the PAN structure, the structure is on the basis of FPN to establish a bottom-up path, the underlying position information to the prediction feature layer, make the prediction feature layer has both the top semantic information and the underlying position information, which can greatly improve the accuracy of target detection.

BiFPN is improved on the basis of the PAN. The structure is shown in Fig. 3-4 (c). Two-way cross-scale connection: First, delete a node with only one input, no feature fusion, little contribution, no significant impact, and simplify the network; Then an edge is added between the original input and output nodes to merge more functions without increasing the cost; Finally, the top-down and bottom-up paths are merged into a module to stack a parent Feature Fusion. Weighted Feature Fusion: BiFPN uses fast normalization fusion, which divides and sums the weight directly by the possession value. At the same time, it normalizes the weight to [0,1], which improves the calculation speed. As shown in formula (7)^[28]:

$$O = \sum_i \frac{w_i}{\epsilon + \sum_j w_j} \cdot I_i \quad (7)$$



Figures 3-4 The F P N, PANet, and BiFPN structures

Where, the activation function ReLU is used to ensure the weight $w_i \geq 0$, and the weight is obtained by the network training, and l_i represents the characteristics of the input. Since the scalar weights are unbounded, softmax is used for normalization, which causes training instability. The improved network takes the three different-scale features P_3 , P_4 and P_7 extracted from the backbone network as input to BiFPN to integrate cross-scale connections and weighted features, and finally sets the predicted branches with three different scales of 2020, 4040 and 8080. Some of the weights are $([0.499\ 98, 0.499\ 98])$, $([0.333\ 32, 0.333\ 32, 0.333\ 32])$, $([0.499\ 98, 0.499\ 98])$; $([0.500\ 00, 0.500\ 00])$, $([0.333\ 25, 0.333\ 25, 0.333\ 25])$, $([0.500\ 00, 0.500\ 00])$. Taking node P_6 as an example, the two fusion feature processes formed are formed as follows^[28]:

$$P_6^{td} = \text{Conv} \left(\frac{w_1 \cdot P_6^{in} + w_2 \cdot \text{Resize}(P_7^{in})}{w_1 + w_2 + \epsilon} \right) \quad (8)$$

$$P_6^{out} = \text{Conv} \left(\frac{w'_1 \cdot P_6^{in} + w'_2 \cdot P_6^{td} + w'_3 \cdot \text{Resize}(P_5^{out})}{w'_1 + w'_2 + w'_3 + \epsilon} \right) \quad (9)$$

Where, P_6^{td} is the top-down intermediate feature (Figure 3-4 (c) middle blue circle), P_6^{out} is the bottom-up output feature (Figure 3-4 (c) right-most blue circle), Resize is the upsampling or downsampling operation, and Conv is the convolution operation.

Based on the above advantages, the PAN module in the YOLOv5 structure is replaced with BiFPN to strengthen the feature fusion and improve the detection speed.

3.4 Add the detection layer

The original YOLOv5s (version 5.0) model has only three detection layers. When the input image size is 640, 640, the neck network is subsampled by 8, 16, and 32 times, respectively, and the corresponding detection layer feature maps are 80×80 , 40×40 , and 20×20 , respectively, to detect small, medium, and large target. In the actual construction scenario, the staff is all over the site at all levels and corners, so the helmet mapped to the image also has many different sizes, especially the dense images containing multiple helmet targets. A dense map may contain many helmet targets, including large scale, mesoscale, small scale, and very small scale. Therefore, in order to cope with complex and dense scenes, this paper adds a detection layer to the three

detection layers of the original YOLOv5s network. Relative to the original YOLOv5s neck network, it is only subsampled twice (stitched together with layers 6 and 4 of the backbone network, respectively), and then subsamples the operation. This paper is through three upsampling, and the third upsampling is first fused with the backbone network layer 2, and then the downsampling operation. That is, after 8 x downsampling, the downsampling is conducted first, change the feature map size to 160×160 , and then continue the downsampling. At the same time, the original 8 x downsampling detection layer is changed into a feature fusion layer. The detection layer after upsampling and downsampling detection layer is taken as the new 8 x downsampling detection layer. After 32 times downsampling, but a 32 times downsampling detection layer is added, that is, the 20×20 scale detection layer feature map. In this way, there are four detection layers, the network depth is further deepened, and the feature information can be extracted from a deeper network, so as to enhance the ability of the model to multi-scale learning under crowded targets, so as to better learn the multi-level feature information of dense targets, and improve the detection performance of the model in dense scenarios.

As shown in Figures 3 – 5, a detection layer was added in the neck of the original model. For the detection effect in the dense helmet scene, such as Figure 3-6, (a) the detection effect diagram of the original yolov5s model, and (b) the actual detection effect diagram after adding the detection layer. The detection confidence obtained in the figure is hat0.87, hat0.91; hat0.90, person0.56, hat0.92. Misdetecion for a remote target before adding the detection layer, and the target detection confidence is significantly improved, indicating that the model after increasing the detection layer has better generalization ability in dense scenarios.

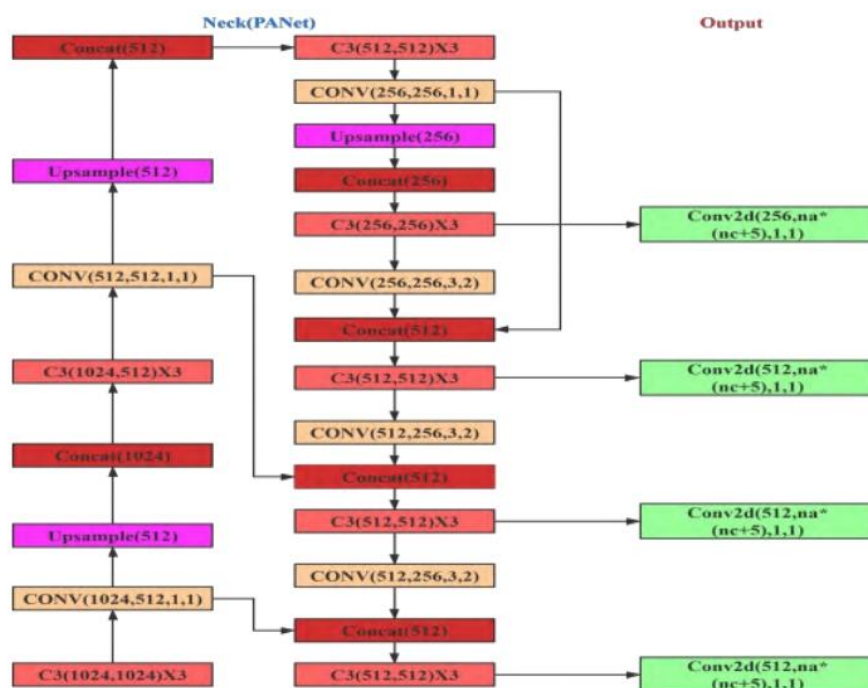


Figure 3-5 Schematic diagram of the added detection layer



(a)Original yolov5s model detection effect chart (b) The yolov5s model detection effect chart after adding the detection layer

FIG. 3-6 compares the actual detection effect map after adding the detection layer

3.5 Improved network structure after this process

The improved network structure is shown in Table 3-3 below, and the overall number of parameters is 7.76MB, which greatly improves the detection accuracy while maintaining the lightweight level.

Table 3-3 Table of improved model network structures and parameters

Order number	From	Params	Module	Arguments
0	-1	3520	Focus	[3, 32, 3]
1	-1	18560	Conv	[32, 64, 3, 2]
2	-1	18816	C3	[64, 64, 1]
3	-1	1688	CoordAtt	[64, 64]
4	-1	73984	Conv	[64, 128, 3, 2]
5	-1	156928	C3	[128, 128, 3]
6	-1	3352	CoordAtt	[128, 128]
7	-1	295424	Conv	[128, 256, 3, 2]
8	-1	625152	C3	[256, 256, 3]
9	-1	6680	CoordAtt	[256, 256]
10	-1	1180672	Conv	[256, 512, 3, 2]
11	-1	656896	SPP	[512, 512, [5, 9, 13]]
12	-1	1182720	C3	[512, 512, 1, False]
13	-1	25648	CoordAtt	[512, 512]
14	-1	131584	Conv	[512, 256, 1, 1]
15	-1	0	Upsample	[None, 2, 'nearest']
16	[-1, 8]	2	BifpnConcat	[1]
17	-1	361984	C3	[512, 256, 1, False]
18	-1	66048	Conv	[256, 256, 1, 1]
19	-1	0	Upsample	[None, 2, 'nearest']
20	[-1, 5]	2	BifpnConcat	[1]
21	-1	329216	C3	[384, 256, 1, False]

22	-1	33024	Conv	[256, 128, 1, 1]
23	-1	0	Upsample	[None, 2, 'nearest']
24	[-1, 3]	2	BifpnConcat	[1]
25	-1	82688	C3	[192, 128, 1, False]
26	-1	147712	Conv	[128, 128, 3, 2]
27	[-1, 22]	2	BifpnConcat	[1]
28	-1	90880	C3	[256, 128, 1, False]
29	-1	147712	Conv	[128, 128, 3, 2]
30	[-1, 18]	2	BifpnConcat	[1]
31	-1	329216	C3	[384, 256, 1, False]
32	-1	590336	Conv	[256, 256, 3, 2]
33	[-1, 14]	2	BifpnConcat	[1]
34	-1	1182720	C3	[512, 512, 1, False]

IV. RESULT AND DISSCUSION

To further demonstrate the impact of different improvements on the network model, the ablation experiments were done, and the ablation experiments are shown in Table 4-1 below.

Table 4-1 Comparison table of the ablation experiments

Model	K-means+ +	CA	Bifpn	Add detection layer	Precision /%	Recall/%	mAP0.5 /%	mAP0.5:0.95 / %
YOLOv5s	×	×	×	×	90.3	86.2	89.8	57.1
YOLOv5s-K	✓	×	×	×	89.7	86.9	90.5	58.5
YOLOv5s-KC	✓	✓	×	×	89.9	87.2	91.2	58.5
YOLOv5s-KCB	✓	✓	✓	×	90.1	87.3	91.8	58.7
YOLOv5s-KCBA	✓	✓	✓	✓	90.8	87.5	92.7	59.7

As can be seen from Table 4-1, After adjusting the a priori box by using K-means + +, Increased the matching degree between the primary box and the target box, The Recall was increased by 0.7%, The mAP0.5 was increased by 0.7%, The m A P of 0.5:0.95 was increased by 1.4%; After the addition of the CA attention mechanism to the backbone network, The Precision was improved by 0.2%, The Recall was increased by 0.3%, The mAP0.5 was increased by 0.7%, No change in mAP0.5:0.95; After replacing the original feature pyramid module of the feature fusion module with a weighted two-way feature pyramid (BiFPN) network structure, Realizing efficient bidirectional cross-scale connectivity and weighted feature fusion, The Precision was improved by 0.2%, The Recall was improved by 0.1%, The mAP0.5 was increased by 0.6%, The m A P of 0.5:0.95 was increased by 0.2%; last, Adding a small target detection layer, Using the 4 detection heads, The Precision was increased by 0.7%, The mAP0.5 was increased by 0.9%, The m A P of 0.5:0.95 was increased by 1%. According to the results before and after improvement, the Precision increased by 0.5%, 1.3%, mAP0.5 by 2.9%, and mAP0.5:0.95 by 2.6%.

From Figure 4-1 and Figure 4-2, the training results of the model are normal, the training process of the model is normal, and there are no bad phenomena such as overfitting. After the improvement, the evaluation indicators obtained by the model training have been improved.

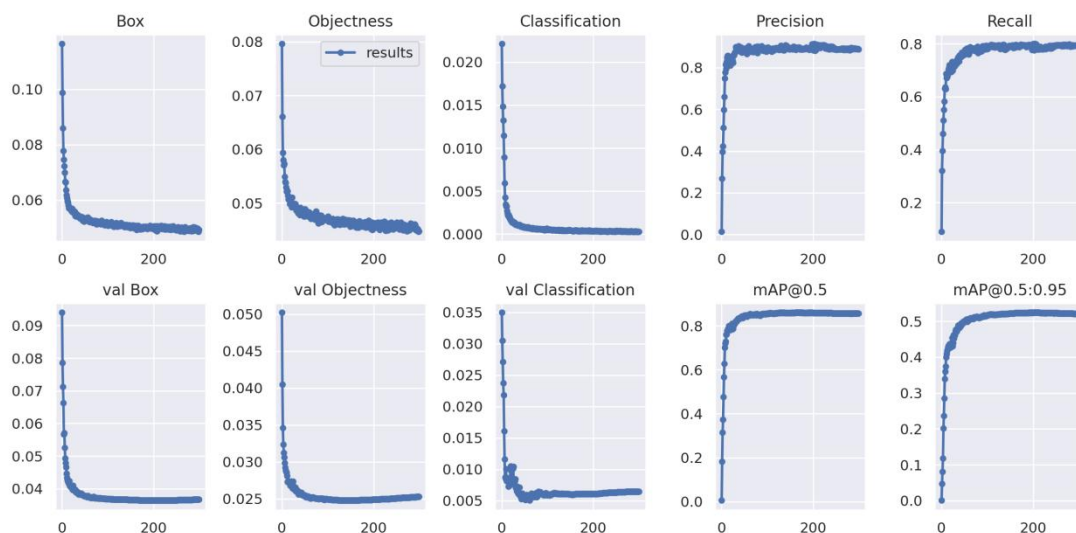


Figure 4-1 Improved previous training results in Fig

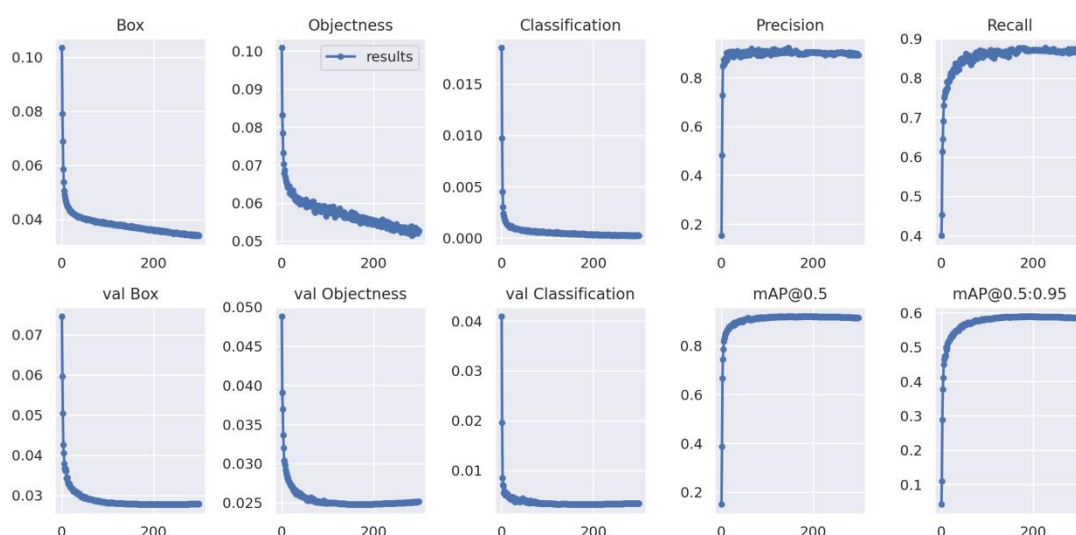


Figure 4-2 Figure of training results after improvement

The Total_Loss comparison obtained before and after the improvement is shown in Figure 4-3. Figure 4-3 shows that the Loss value of the improved model is lower than the Loss value of the improved model. The YOLOv5-KCBA model converges faster and has smaller loss value compared with the YOLOv5s model, indicating that the network improves the original loss function.

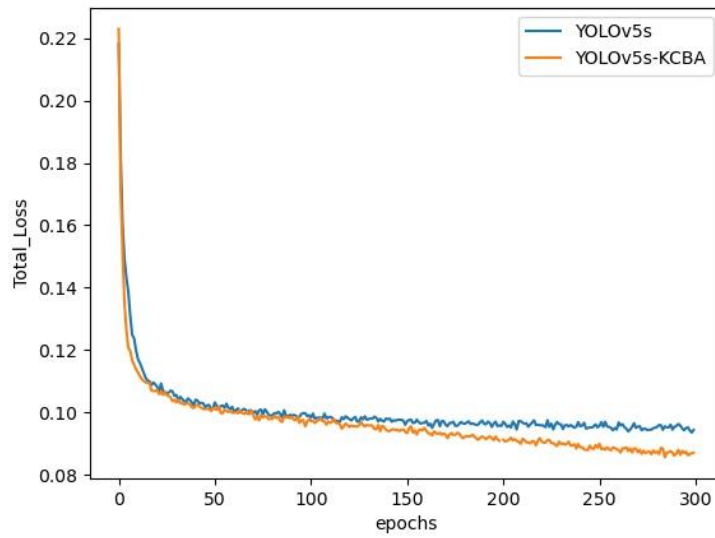


Figure 4-3 Comparison of the trained Loss values of the original YOLOv5s model and the later trained Loss values of the modified YOLOv5s-KCBA model

The comparison diagram of the actual detection effect of the model before and after the improvement is shown below in Figure 4-4 and Figure 4-5 below. The confidence obtained in Figure 4-4 is 0.90,0.83,0.83,0.79,0.84,0.86, and those detected in Figure 4-4 are 0.91,0.87,0.89,0.87,0.87. The actual detection of the YOLOv5s-KCBA model was higher than that of the results of the original YOLOv5s model. The confidence obtained in Figures 4-5 is 0.85 and 0.82, respectively, and that obtained in Figure 4-4 is 0.87 and 0.86, respectively, which were also higher than the results of the original YOLOv5s model.



(a)Effect of YOLOv5s model detection (b) Effect of YOLOv5s-KCBA model detection

Figure 4-4 Comparison of model pre-and post-improvement detection effects



(a) YOLOv5s model test effect diagram (b) YOLOv5s-KCBA model test effect diagram
Figure 4-5 Comparison of model pre-and post-improvement test effects

V. The GUI Interface Design Based on PyQt

5.1 Introduction to PyQt

PyQt is a toolkit that uses a successful fusion of the Python programming language and the Qt library for creating GUI applications. It has a cross-platform advantage, and can be easily run on all major operating systems, including UNIX, Windows, and Mac. Generally, a GUI interface is created as shown in Table 5-1.

Table 5-1 Establishing the GUI interface process

step	The main steps
(1) Create the main window	The main window provides a GUI application architecture with a preset layout, where users can add controls to the layout. Users can add different controls to the main window, such as toolbar, menu bar, status bar, etc., to meet the interface effect required by users.
(2) Add control	PyQt can provide many types of controls, including a text-box class, a push-button class, a window-drawing class, etc. Developers can choose the corresponding control to develop according to the actual needs.
(3) Set signal and slot	Mutual communication between the PyQt controls is realized by the signal and the slot. Therefore, the most critical step is to set the slot and signal to the corresponding control. When triggering the corresponding operation to the control, the signal is triggered, and the function program in the slot bound to the signal is executed, and the result of the processing is returned.

5.2 Establish a GUI interface

There are two ways to make a GUI interface in PyQt: choose a UI making tool and choose to write

code. Qt Designer is the most commonly used interface design tool in UI tools. It is powerful and flexible, enabling developers to quickly design and layout user interfaces and drag, and visually.

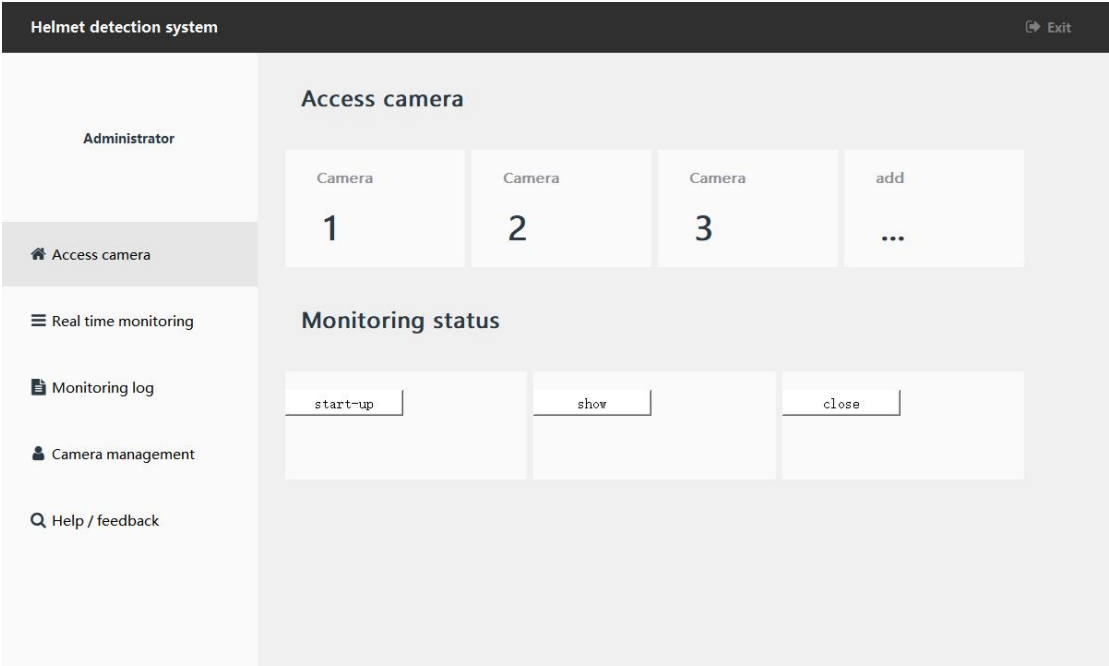


Figure 5-1 Safety helmet detection system interface

(1) Design of safety helmet detection interface

Because the interface design with written code is very time-consuming and laborious, the UI design tools can be used to assist users in developing the interface. The interface is shown in Figure 5-1.

Next, we can make the interface of the helmet identification and detection system, link the background camera background, can display the monitoring picture of different cameras, real-time monitoring of multiple directions, automatically generate monitoring logs to view historical monitoring information, followed by help and feedback.

(2) Safety helmet detection effect of the safety helmet detection system

In the safety helmet identification and detection system, its main functions have two parts: the monitoring and recording of the people who do not wear the safety helmet, and the feedback to the system has a dangerous behavior.

VI. Conclusion

The occurrence of not wearing a helmet is easy to cause unnecessary injuries and economic losses. Therefore, it is of great research value to quickly identify and eliminate it in the bud. With the gradual development of intelligent monitoring system, the use of the monitoring system to monitor whether the helmet for real-time efficient detection is of positive significance to timely avoid greater casualties and property losses. Although there are various methods of safety helmet detection, but the traditional manual Angle inspection exists, can not be completely covered in real time, low efficiency, poor timeliness, vulnerable to human factors interference and other factors. Therefore, this paper proposes the deep learning-based safety helmet detection algorithm research. In order to overcome the shortcomings of the current target detection technology, this paper proposes its own improvement method for the YOLOv5s model, and improves the safety helmet recognition model. The main work and research results are

summarized as follows:

(1) After consulting a large number of literature and materials, the author analyzed the current situation of domestic and foreign target detection methods and deep learning-based detection methods. The disadvantages of the traditional detection methods are summarized, such as the limited detection range, the vulnerable to external environment interference, the low accuracy, and the slow response time, and so on. In order to solve the above problems, after analyzing the advantages of safety helmet recognition based on deep learning, the convolutional neural network is used to detect the actual research background and the defects existing in the existing safety helmet detection algorithm.

(2) Collect the images from the network, intercept the public network data set, and establish the image data set, and use the data enhancement technology to preprocess the images in the data set, to realize the effect of sample expansion and feature enhancement. After experimental experiments, YOLOv5 models with good detection performance were selected among five target detection models.

(3) To improve the accuracy of multiple targets and small targets, Separutilizing the K-Mean + + algorithm in the deep convolution domain, Improving the prior box using the kmeans + + algorithm, Improve the matching degree of the target box and the prior box; Adding attention mechanisms, Improve the ability of the network results to extract the target object features; Replace the original feature pyramid module of the feature fusion module with a weighted bidirectional feature pyramid (BiFPN) network structure, To achieve efficient bidirectional cross-scale connection and weighted feature fusion; Add a small target detection head, Targets were tested using 4 detection heads, It effectively improves the feature extraction ability for small targets. It can be seen from the experimental results that the improved YOLOv5 model has significantly improved the accuracy of fire target detection.

REFERENCE

- [1] Lin T-Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 2117-2125.
- [2] Dai J, He K, Sun J. Instance-aware semantic segmentation via multi-task network cascades[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016: 3150-3158.
- [3] Hariharan B, Arbelaez P, Girshick R, et al. Simultaneous detection and segmentation[C]. European Conference on Computer Vision. Springer, Cham, 2014: 297-312.
- [4] Hariharan B, Arbelaez P, Girshick R, et al. Hypercolumns for object segmentation and fine-grained localization[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015: 447-456.
- [5] He K, Gkioxari G, Dollar P, et al. Mask r-cnn[C]. Proceedings of the IEEE international conference on computer vision, 2017: 2961-2969.
- [6] Rumelhart D E, Hinton G E, Williams R J. Learning representations by back propagating errors[J]. nature, 1986, 323 (6088): 533-536.
- [7] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. science, 2006, 313 (5786): 504-507.
- [8] Deng J, Dong W, Socher R, et al. Imagenet: A large- scale hierarchical image database[C]. 2009

- IEEE conference on computer vision and pattern recognition, 2009: 248-255.
- [9] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60 (6): 84-90.
- [10] Sermanet P, Eigen D, Zhang X, et al. Overfeat: Integrated recognition, localization and detection using convolutional networks[J]. arXiv preprint arXiv:1312.6229, 2013.
- [11] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2015: 1-9.
- [12] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv: 1409.1556, 2014.
- [13] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C] Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 770-778.
- [14] Viola P, Jones M. Rapid object detection using a boosted cascade of simple features[C]. Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR2001, 2001: I-I.
- [15] Dalal N, Triggs B. Histograms of oriented gradients for human detection[C]. 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05), 2005: 886-893.
- [16] Felzenszwalb P F, Girshick R B, Mcallester D. Cascade object detection with deformable part models[C]. 2010 IEEE Computer society conference on computer vision and pattern recognition, 2010: 2241-2248.
- [17] Girshick R, Donahue J, Darrell T, et al. Region-based convolutional networks for accurate object detection and segmentation[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 38 (1): 142-158.
- [18] He K, Zhang X, Ren S, et al. Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37 (9): 1904-1916.
- [19] Girshick R. Fast r-cnn[C]. Proceedings of the IEEE international conference on computer vision, 2015: 1440-1448.
- [20] Ren S, He K, Girshick R, et al. Faster R-CNN: towards real-time object detection with region proposal networks[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(6): 1137-1149.
- [21] Lin T-Y, Dollar P, Girshick R, et al. Feature pyramid networks for object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 2117-2125.
- [22] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified, real time object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 779-788.
- [23] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]. European conference on computer vision, 2016: 21-37.
- [24] Fu C-Y, Liu W, Ranga A, et al. Dssd: Deconvolutional single shot detector[J]. arXiv preprint arXiv:1701.06659, 2017.
- [25] Lin T-Y, Goyal P, Girshick R, et al. Focal loss for dense object detection[C]. Proceedings of the IEEE international conference on computer vision, 2017: 2980-2988.
- [26] Zhao Q, Sheng T, Wang Y, et al. M2det: A single-shot object detector based on m

ulti-level feature pyramid network[C]. Proceedings of the AAAI conference on artificial intelligence, 2019: 9259-9266.

[27] Zhang S, Wen L, Bian X, et al. Single-shot refinement neural network for object detection[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018: 4203-4212.

[28] Tan M, Pang R, Le Q V. Efficientdet: Scalable and efficient object detection[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020: 10781-10790.

[29] Redmon J, Farhadi A. YOLO9000: better, faster, stronger[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 7263-7271.

[30] Redmon J, Farhadi A. Yolov3: An incremental improvement[J]. arXiv preprint arXiv: 1804.02767, 2018.

[31] Bochkovskiy A, Wang C-Y, Liao H-Y M. Yolov4: Optimal speed and accuracy of object detection[J]. arXiv preprint arXiv:2004.10934, 2020.

[32] Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.

[33] Lowe D G. Distinctive image features from scale-invariant keypoints[J]. International journal of computer vision, 2004, 60(2): 91-110.

[34] Lienhart R, Maydt J. An extended set of haar like features for rapid object detection[C]. Proceedings. international conference on image processing, 2002: I-I.

[35] Joachims T. SvmLight: Support vector machine[J]. SVM-Light Support Vector Machine <http://svmlight.joachims.org/>, University of Dortmund, 1999, 19(4).

[36] Freund Y, Schapire R E. A decision-theoretic generalization of on-line learning and an application to boosting[J]. Journal of computer and system sciences, 1997, 55(1): 119-139.

[37] Felzenszwalb P F, Girshick R B, Mcallester D, et al. Object detection with discriminatively trained part-based models[J]. IEEE transactions on pattern analysis and machine intelligence, 2009, 32(9): 1627-1645.

[38] Belongie S, Malik J, Puzicha J. Shape matching and object recognition using shape contexts[J]. IEEE transactions on pattern analysis and machine intelligence, 2002, 24 (4): 509-522.

[39] Felzenszwalb P, Mcallester D, Ramanan D. A discriminatively trained, multiscale, deformable part model[C]. 2008 IEEE conference on computer vision and pattern recognition, 2008: 1-8.

[40] Szegedy C, Ioffe S, Vanhoucke V, et al. Inception-v4, inception-resnet and the impact of residual connections on learning[C]. Proceedings of the AAAI Conference on Artificial Intelligence. 2017, 31(1).

[41] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2017: 4700-4708.

[42] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2018: 7132-7141.

[43] Uijlings J R, Van De Sande K E, Gevers T, et al. Selective search for object recognition[J]. International journal of computer vision, 2013, 104(2): 154-171.

- [44] Lazebnik S, Schmid C, Ponce J. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories[C]. 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06), 2006:2169-2178.
- [45] Dai J, Li Y, He K, et al. R-fcn: Object detection via region based fully convolutional networks[J]. arXiv preprint arXiv: 1605.06409, 2016.
- [46] Li Z, Peng C, Yu G, et al. Light-head r cnn: In defense of two-stage object detector[J]. arXiv preprint arXiv:1711.07264, 2017.
- [47] Szegedy C, Toshev A, Erhan D. Deep neural networks for object detection[J]. 2013.
- [48] Pinheiro P O, Collobert R, Dollar P. Learning to segment object candidates[J]. arXiv preprint arXiv: 1506.06204, 2015.
- [49] Erhan D, Szegedy C, Toshev A, et al. Scalable object detection using deep neural networks[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2014: 2147-2154.
- [50] Yoo D, Park S, Lee J-Y, et al. Attentionnet: Aggregating weak directions for accurate object detection[C]. Proceedings of the IEEE International Conference on Computer Vision, 2015: 2659-2667.
- [51] Najibi M, Rastegari M, Davis L S. G-cnn: an iterative grid based object detector[C]. Proceedings of the IEEE conference on computer vision and pattern recognition, 2016: 2369-2377.
- [52] Tan M, Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks[C]. International Conference on Machine Learning, 2019: 6105-6114.
- [53] Qian W, Yang X, Peng S, et al. Learning modulated loss for rotated object detection[J]. arXiv preprint arXiv:1911.08299, 2019.
- [54] Yang X, Liu Q, Yan J, et al. R3det: Refined single-stage detector with feature refinement for rotating object[J]. arXiv preprint arXiv: 1908.05612, 2019.
- [55] Ming Q, Zhou Z, Miao L, et al. Dynamic Anchor Learning for Arbitrary-Oriented Object Detection[J]. arXiv preprint arXiv:2012.04150, 2020.
- [56] Wang T, Yang T, Cao J, et al. Co-mining: Self-Supervised Learning for Sparsely Annotated Object Detection[J]. arXiv preprint arXiv:2012.01950, 2020.
- [57] Cai Y, Li H, Yuan G, et al. YOLOmobile: Real-Time Object Detection on Mobile Devices via Compression-Compilation Co-Design[J]. arXiv preprint arXiv:2009.05697, 2020.
- [58] Karlinsky L, Shtok J, Alfassy A, et al. StarNet: towards weakly supervised few-shot detection and explainable few-shot classification[J]. arXiv preprint arXiv:2003.06798, 2020.
- [59] Han D, Yun S, Heo B, et al. Rexnet: Diminishing representational bottleneck on convolutional neural network[J]. arXiv preprint arXiv:2007.00992, 2020.
- [60] Dai Z, Cai B, Lin Y, et al. UP-DETR: Unsupervised Pre-training for Object Detection with Transformers[J]. arXiv preprint arXiv:2011.09094, 2020.
- [61] Liang T, Wang Y, Tang Z, et al. OP ANAS: One-Shot Path Aggregation Network Architecture Search for Object Detection[J]. arXiv preprint arXiv:2103.04507, 2021.
- [62] Zhou Q, Yu C, Wang Z, et al. Instant-Teaching: An End-to-End Semi-Supervised Object Detection Framework[J]. arXiv preprint arXiv:2103.11402, 2021.
- [63] Li X, Wang W, Hu X, et al. Generalized Focal Loss V2: Learning Reliable Localization Quality Estimation for Dense Object Detection[J]. arXiv preprint arXiv:2011.12885, 2020.
- [64] Chen L, Yang T, Zhang X, et al. Points as Queries: Weakly Semi-supervised Object De

- tection by Points[J]. arXiv preprint arXiv:2104.07434, 2021.
- [65] Joseph K, Khan S, Khan F S, et al. Towards Open World Object Detection[J]. arXiv preprint arXiv:2103.02603, 2021.
- [66] Chen Q, Wang Y, Yang T, et al. You only look one-level feature[J]. arXiv preprint arXiv:2103.09460, 2021.
- [67] Hou Q, Zhou D, Feng J. Coordinate attention for efficient mobile network design[J]. arXiv preprint arXiv:2103.02907, 2021.
- [68] Dai X, Jiang Z, Wu Z, et al. General Instance Distillation for Object Detection[J]. arXiv preprint arXiv:2103.02340, 2021.
- [69] Liu J, Li C, Liang F, et al. Inception Convolution with Efficient Dilation Search[J]. arXiv preprint arXiv:2012.13587, 2020.
- [70] DollarP, Tu Z, Perona P, et al. Integral channel features[J]. 2009.
- [71] Fukushima K, Miyake S, Ito T. Neocognitron: A neural network model for a mechanism of visual pattern recognition[J]. IEEE Transactions on Systems, Man, and Cybernetics, 1983, SMC-13; SMC-13(5; 5): 826-834.
- [72] Wang Zhenhua, Li Jing, Zhang Xinyue, et al. Review of Deep Learning Target Recognition Algorithms for video data [J / OL]. Computer Engineering: 1-19 [2022-03-26].
- [73] REDMON J, FARHADI A. YOLO9000: better, faster, stronger[J]. CoRR, 2016, abs/1612.08242
- [74] BOCHKOVSKIY A, WANG C Y, LIAO H. YOLOv4: optimal speed and accuracy of object detection [EB/OL]. [2020-04-23]. <https://arxiv.org/abs/2004.10934.pdf>.
- [75] Jiang Rongqi, Peng Yueping, Xie Wenxuan, Xie Guorong. Improved YOLOv4 small object detection algorithm embedded in the scSE module [J]. Journal of Atlas, 2021, 42 (04): 546-555.
- [76] HOU Q B, ZHOU D Q, FENG J S. Coordinate attention for efficient mobile network design[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021: 13713-13722.