Optimizing YOLOv5 algorithm for Mask-wearing Detection

Yang Fan, Wu Wang
School of Mathematics and Computer Science, Yunnan Minzu University
2929# Yuehua Street, Kunming, 650500, China
Email:1354368984@qq.com, wwkmyn@139.com

The novel coronavirus has a strong ability to spread and survive. Wearing a mask correctly can effectively reduce the spread of the virus among the crowd. How to intelligently and efficiently detect the wearing of a mask is of great significance. Detecting whether to wear a mask is the target detection content that many researchers are currently studying. YOLOv5 (You Only Look Once) is an excellent algorithm in target detection. Given that detecting whether a mask is worn is different from other target detection tasks, in this paper, we tried to optimize YOLOv5 algorithm to make it more suitable for mask-wearing detection. In words, detection layers, attention mechanism were added, and proper loss function was chosen strictly to the YOLOv5 target detection algorithm. So that optimal YOLOv5 algorithm model was proposed. The accuracy rate (precision), recall rate (recall) and average precision (mAP) of the algorithm on the test set were 83%, 83.3% and 81.7% respectively, higher than YOLOv3, YOLOv4, YOLOv5 detection algorithm.

***Index Terms*—COVID-19, target detection, mask-wearing detection, YOLOv5, attention mechanism.**

I. Introduction

FROM late 2019, a full-blown outbreak and rapid spread of novel coronavirus pneumonia (COVID-19) began in China, while the National Health and Wellness Commission issued prevention guidelines requiring the mandatory wearing of masks in public. The novel coronavirus is highly transmissible and survivable, and respiratory droplets are its' main transmission route [1]. Wearing a mask correctly can effectively prevent the virus droplet transmission, therefore, testing whether people are effectively wearing masks in public has become a central task at present. Because manual detecting takes a huge amount of time and energy, and close contact with the tested person will bring certain security risks. Thus, establishing a monitoring system to detect mask-wearing, enable epidemic prevention automate will be a long-term practical need in the future, which is of great significance to the society and has usage scenarios in many industries [2].

With the rapid development of deep learning in recent years, technologies related to the field of computer vision have been widely put into industrial and daily life. There is an urgent need to use computer vision technology to replace traditional manual detection, improve detection efficiency and save public resources. Target detection refers to the identification, localisation and classification of a target in the field of view. In terms of computer vision, the use of computers for target detection was still a daunting task until a decade or so ago. Research on target detection began in the 1990s, and the need for this technology in a wide range of fields has led to a large number of domestic and international scholars investigating the technology. In recent years, deep learning techniques have been widely used in the field of target recognition [3]. Feng et al. used deep learning to study the automatic recognition of safety helmets [4]. Li et al. used a convolutional neural network model to extract road information from high-resolution remote sensing images [5]. Zhang et al. used the Fast Region-based Convolutional Network (Fast-RCNN) algorithm to study [6]. Xu et al. improved the Single Shot MultiBox Detector (SSD) algorithm and added an improved non-maximum suppression process at the end of the classifier [7]. Deng in [8] proposed a retinal neural network-based method for mask-wearing detection, his work used pre-trained RESNET models to help train new models through migration learning. Finally, the good results were achieved on the validation set. You Only Look Once (YOLO) is the latest target detection model. On the basis of the YOLOv3 (YOLO Version 3, similarly hereinafter) network model, Wang et al. introduced an improved spatial pyramid pool structure, optimized the multi-scale prediction network, replaced the loss function, and improved the accuracy 14.9 % [9]. Based on the YOLOv5 network model, Tan. et al. extended the original data set, obtained 30,000 pictures by flipping and rotating, and then trained these pictures, detection accuracy was greatly improved. Other researchers have also carried out various optimizations, which have achieved great improvements in accuracy and precision [10].

Similar to traditional target detection tasks, mask-wearing detection for practical applications faces a variety of challenges, not only for single but also for multiple person in closed environments, such as buses, subways and waiting rooms; at the same time, it is necessary to solve the problems of target loss, misjudgement and missed detection caused by the large number of people being inspected, the existence of partial coverings, the diversity of mask types and non-mask covering the face. To address mask-wearing detection issue and integrates the accuracy, this paper proposes an optimal model based on YOLOv5, the main improvements including:

- Adding a detection layer. The detection layer was added to the original YOLOv5s algorithm, specifically, we added a small target detection layer to the 31st layer of the network structure. Therefore, the optimal model used four layers [21,24,27,30] to perform detection, which provided

Manuscript received March 25, 2023; revised May 7, 2023. Corresponding author: Wu Wang (email: wwkmyn@139.com).

a good improvement in the detection of small targets.

- Adding the attention mechanism. Specifically, we added the SENet module to original model, which used automatic learning to takes the importance of each channel in the feature map and according to this importance to assign a value to each feature, so that it improves the detection accuracy for small targets.
- The CIoU_Loss is optimized for the mask-wearing detection. So that, optimal CIoU_Loss took into account more scale information like the width-to-height ratio of the bounding box than the GIoU_Loss does, which was measured from three perspectives, namely the overlap area, centroid distance, and aspect ratio. It makes prediction box regression results better.

The rest of this paper organized as follows, Section II provided a series of overviews about two typical target detection models. Then focusing on the structure of the YOLOv5 algorithm, including Input, Backbone, Neck, and Head. Section III introduced the improvement strategies to original YOLOv5 algorithm, including adding a detection layer, adding an attention mechanism, and optimizing the loss function. Section IV presented the results from our optimal algorithms and other detection algorithms. Finally, Section V presented the conclusion of this paper.

II. YOLOv5 Algorithm

The mainstream algorithms of target detection are mainly divided into two types. One is two-stage algorithm, and the other is one-stage algorithm. The difference is that two-stage has a region proposal process, which is similar to an open audition process. In such network, it will generate the location and category based on the proposal area, whereas the one-stage algorithm directly generates the location and category from the image. YOLO is a one-stage algorithm. YOLOv1 [11] creatively uses the first-order structure to complete two tasks of classification and target localization. Subsequently, both speed and accuracy have improved at later version of YOLO, which accelerated the application of target detection in the industry.

From YOLOv1 in 2015 to the latest YOLOv5, YOLOv4 firstly adopts the CSP (cross stage partial network) structure in its the backbone network, Ordinary convolution is operated in its Neck, it can complete the training on an ordinary GPU (1080ti) [12]. Whereas YOLOv5 designs two CSP structures, the CSP1_X structure is applied to the backbone network, and another CSP2_X structure is applied to the component Neck, which can strengthen its feature fusion. According to the depth and width of the network, YOLOv5 is divided into YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The model size and accuracy of these four versions increase sequentially. YOLOv5s has the smallest network structure, and the fastest image reasoning speed can reach 0.007s. Therefore, this article optimal network structure based on YOLOv5s.

The network structure of YOLOv5 is composed of four parts, namely the Input, Backbone, Neck, and Head, as shown in Fig. 1.

1) Input

On the input side, Mosaic data augmentation method is used, which takes 4 training images in different contexts and tiles them into one. It then performs random cropping to produce an augmented image of size equal to the original training image size. Which can enrich the data, increase many small targets, and improve the recognition ability of small targets. Four images can be calculated at the same time, which is equivalent to increasing the Mini-batch size and reducing the consumption of GPU memory. In the YOLOv5 algorithm, different data sets will set anchor boxes with different initial lengths and widths. When training data, the prediction box is obtained on the basis of the initial anchor box, and compared with the real box, the difference between the two is calculated. Reverse update, iterative update of network structure parameters, adaptive anchor box calculation can get the best anchor box value. Then the scaling mode of adaptive image size is used in the prediction, and the prediction speed is improved by reducing the black border.

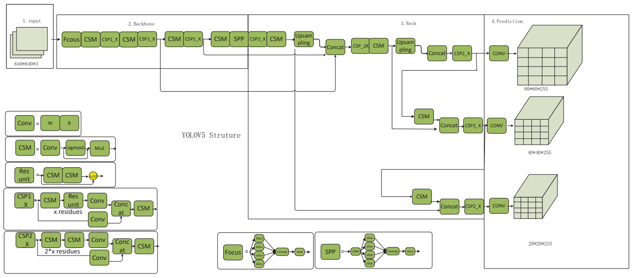

Fig. 1. The structure of YOLOv5

2) Backbone

YOLOv5 adopts the focus and CSPnet modules in backbone. Focus performs slice operation. The original 68 × 608 × 3 image enters focus, it is first cut into a 304 × 304 × 12 feature map, and then conduct convolution operation under 32 convolution kernels, the final output is a feature map of 304 × 304 × 32, as shown in Fig. 2. In addition, regarding the CSPnet module, YOLOv5 designed two different CSP structures, the CSP1_X structure is applied to the Backbone network, and the other CSP2_X structure is applied to the Neck. As shown in Fig. 3.

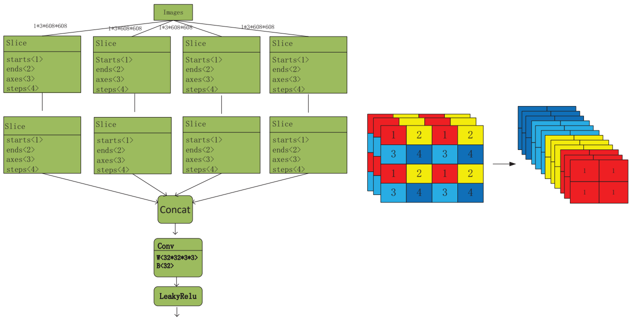

Fig. 2. Slice operation at Focus

3) Neck

When YOLOv5 was first launched, its Neck only used the FPN (feature pyramid network) structure, and later added the PAN (personal area network) structure to form the current FPN+PAN structure. In addition, other parts of the network

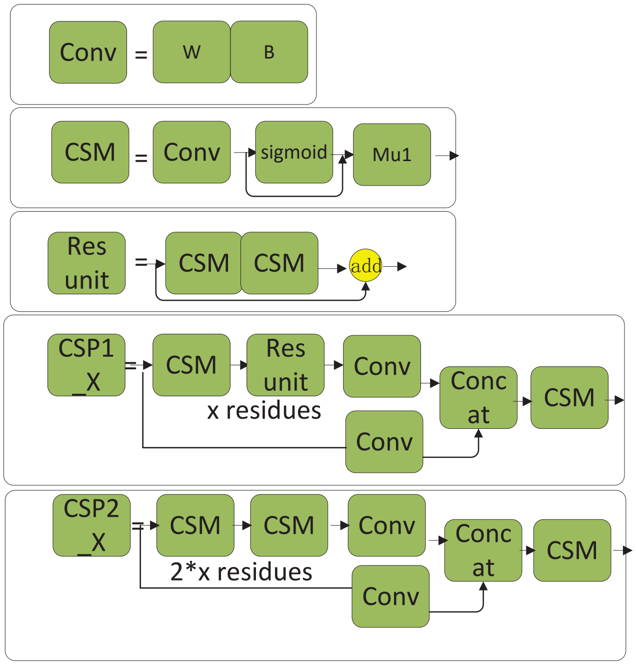

Fig. 3. Two CSP modules in YOLOv5

have also been adjusted, and the high-level feature information is transferred and fused by upsampling to obtain a predictable feature map. The PAN here uses the feature pyramid to convey position information from the bottom up, and the structure is shown in Fig. 4:

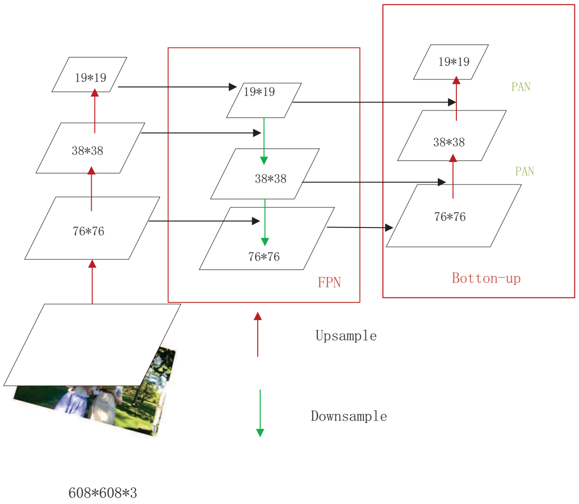

Fig. 4. Illustration of FPN+PAN structure

4) Head

Head contains Boundingbox loss function and NMS (Non-Maximum Suppression). YOLOv5 uses GIoU_Loss as its loss function, which increases the measurement method of intersection scale, solves the problem that the target frame and the prediction frame do not intersect, and can distinguish the overlap between the target frame and the prediction frame when the IoU is the same. The GIoU_loss could be calculate as follows:

$$GIoU\_loss = 1 - GIoU, \tag{1}$$

where, $GIoU = IoU - \frac{|C \ (A \cup B)|}{|C|}$. The classification loss in the training phase uses binary cross-entropy loss (BCEloss). Therefore, the complete loss function is composed of bounding box regression loss, confidence prediction loss and category prediction loss. As shown in the formula (2).

$$\begin{aligned} Loss(obj) = GIoU\_loss \\ &+ \textstyle\sum_{\mathrm{i=0}}^{S \times S} \sum_{\mathrm{j=0}}^{\mathrm{B}} 1_{\mathrm{ij}}^{\mathrm{obj_i}} \left[\mathrm{C_i} \log\left(\mathrm{C_i}\right)\right] \\ &+ \left(1 - \mathrm{C_i}\right) \log\left(1 - \mathrm{C_i}\right) \\ &- \textstyle\sum_{\mathrm{i=0}}^{S \times \mathrm{S}} \sum_{\mathrm{j=0}}^{\mathrm{B}} 1_{\mathrm{ij}}^{\mathrm{noobj}} \left[\mathrm{C_i} \log\left(\mathrm{C_i}\right)\right] \\ &+ \left(1 - \mathrm{C_i}\right) \log\left(1 - \mathrm{C_i}\right) \\ &+ \textstyle\sum_{\mathrm{i=0}}^{\mathrm{S \times S}} \sum_{\mathrm{j=0}}^{\mathrm{B}} 1_{\mathrm{ij}}^{\mathrm{obj}} \sum_{\mathrm{c \in\ classes}} \left[\mathrm{p_i(c)} \log\left(\mathrm{p_i(c)}\right.\right. \\ &+ \left.\left(1 - \mathrm{p_i(c)}\right) \log\left(1 - \mathrm{p_i(c)}\right)\right] \end{aligned} \tag{2}$$

GIoU focuses on not only overlapping areas, but also other non-overlapping areas, which can reflect the degree of overlap between the two better. So, formula (2) could be rewrote as follows:

$$\begin{aligned} GIoU\_Loss &= 1 - CIoU \\ &= 1 - (IoU - \frac{Distance\_2^2}{Distance\_C^2} \\ &- \frac{V^2}{1 - IoU + V}), \end{aligned} \tag{3}$$

where v is a parameter to measure the consistency of aspect ratio, it could be defined in formula (4):

$$\mathrm{v} = \frac{4}{\pi^2} \left(\arctan \frac{\mathrm{w^{gt}}}{\mathrm{h^{gt}}} - \arctan \frac{\mathrm{w^p}}{\mathrm{h^p}} \right)^2. \tag{4}$$

In this way, CIoU_Loss takes into account the three important geometric factors of the target frame regression function: overlapping area, center point distance, and aspect ratio. When selecting the optimal prediction frame, YOLOv5 uses a weighted NMS method to remove redundant prediction frames [13].

III. Proposed Optimal YOLOv5

A. Adding Detection Layers

As shown in Fig. 5, the Backbone network of original YOLOv5s is mainly to extract image features, the head part outputs a total of three scale detection layers. A 640 × 640 image be input, the output size at detection layer are 80 × 80, 40 × 40 and 20 × 20, which are used for small, medium and large goals detection, respectively. In order to improve the detection accuracy, in this paper, we add detection headers and enhance features starting from layer 2, as shown in Fig. 6. The neck mainly adopts a method similar to FPN+PAN to perform multi-scale fusion processing on the features extracted by the backbone network, and then send them to the detection layer. Added several layers in the head module, continue to upsample the feature map after the 17th layer, so that the feature map continues to expand. At the same time, on the 20th layer, we concat the obtained feature map with a size of 160 × 160 in the Backbone network, so as to obtain a larger feature map to detect small targets. In the 31st layer (the detection layer), we add a small target detection layer here, and a total of four layers: 21, 24, 27, 30 are used for detection. After adding the detection layer, there is indeed a good improvement for small targets.

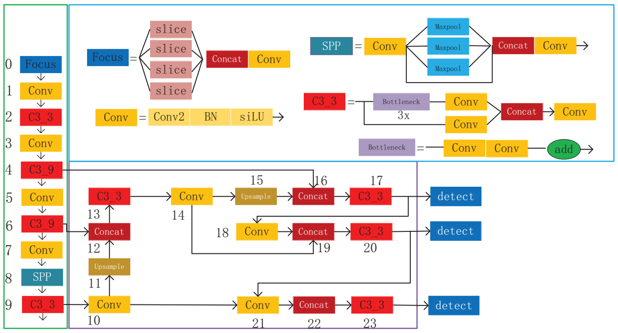

Fig. 5. Original YOLOv5 model structure

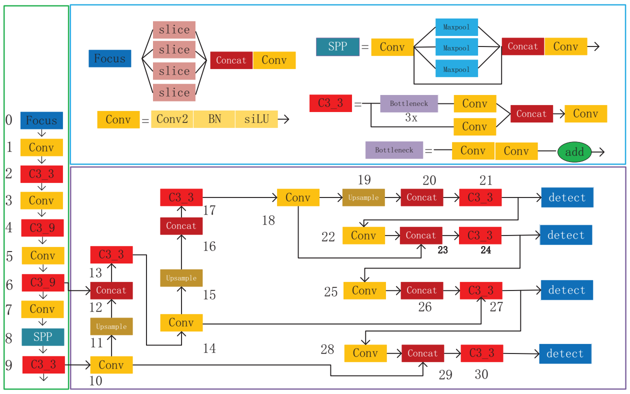

Fig. 6. Optimal YOLOv5 structure

B. SENet Attention Mechanism

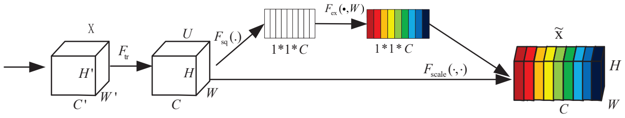

Fig. 7. Squeeze-and-Excitation module

As the YOLOv5s network layer continues to deepen, the information extracted at the output is gradually abstracted, it is more difficult to detect small distant targets in the images, especially, it is difficult to accurately detect facial occlusion in congested road sections. To this end, this paper integrates attention mechanism into the network. SENet (squeeze-and-excitation networks) is a typical channel attention network [14], which won the ImageNet 2017 classification competition champion. The structure of such network is shown in Fig. 7. In deep learning neural networks, not all extracted features are important. The role of SE attention mechanism is to enhance important features and suppress general features, in words, using new neural network to obtain the importance of each channel of the feature map, and then use this importance to assign a weight value to each feature, so that the neural network can focus on certain feature channel. Promote channels of feature maps that are useful for the current task, and suppress feature channels that are not very useful for the current task. It performs Squeeze, Excitation and feature recalibration on the feature map obtained by convolution [15].

Squeeze compresses each channel of the feature map, performs global average pooling on the two-dimensional vector of $H \times W$, and outputs a vector of $1 \times 1 \times C$.

$$z_k = \frac{1}{H \times W} \sum_{i}^{H} \sum_{j}^{W} u_k(i,j). \tag{5}$$

After excitation gets the $1 \times 1 \times C$ vector, it uses the fully connected layer to generate a weight for each channel, which is used to represent the importance of the feature channel.

$$\mathrm{s} = \sigma\left(\mathrm{W}_2 \delta\left(\mathrm{W}_1 \mathrm{z}\right)\right). \tag{6}$$

$$\widetilde{X}_C = s_c u_c. \tag{7}$$

The effectiveness of the SE module has been verified in some applications, but where it is more effective to embed it in the network, there is no complete theoretical explanation [16]. This paper designs two fusion methods at different positions. That is, the SE module is fused with the backbone and head modules respectively. The main function of backbone is to extract the depth features in the image through a relatively deep convolutional network. As the number of network layers deepens, the width of the feature map becomes smaller and deeper, the SE module can be used to reconstruct the channel attention of the feature maps at different positions, and then use the BottleneckCSP structure to aggregate the features of different levels, so put SE after BottleneckCSP, as shown in Fig. 8. Fuse the SE module with the Head, which purpose is to reconstruct the attention of each feature map before prediction, as shown in Fig. 9.

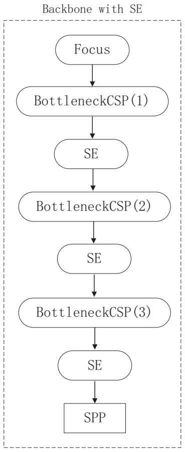

Fig. 8. Backbone fuse with SE module

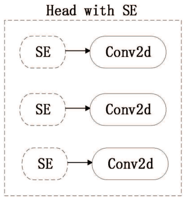

Fig. 9. Head fuse with SE module

C. Optimizing Loss Function CIoU_Loss

IoU is called the intersection and union ratio, which is an indicator for evaluating the performance of the target detection. It calculates the ratio of the intersection and union between the predicted frame and the real frame [17]. GIoU introduces a penalty term based on IoU to more accurately reflect the intersection of the detection frame and the real frame. The formula (8) is the L2 loss used in YOLOv3, the formula (9) is the traditional IoU and its border loss expression, and the formula (10) is the GIoU and its border loss expression. Compared with the IoU_loss, L2 and GIoU have the ability to measure deviation from the trend. As shown on the left side of Fig. 10, when the traditional IoU=0, the final loss is the same regardless of the distance of the border distance, but GIoU behaves as the distance between the two borders increases will close to -1. At the same time, GIoU will drive the predicted frame to be distributed in the up, down, left, and right directions of the real frame, and will impose greater losses on the prediction results in the oblique direction, as shown on the right side of Fig. 10. However, when this special case occurs , namely, $C = A \cup B$ between detection frame and the real frame, then the penalty item in GIoU will become 0, that is, GIoU will degenerate into IoU, and the advantage of GIoU will disappear at this time. Therefore, this paper chooses CIoU_Loss as the bounding box loss function to make the predicted box fit the real box more closely.

$$\lambda_{\text{coord}} \sum_{i}^{S^2} \sum_{j}^{\text{anchors}} 1_{ij}^{obj} \sum_{l \in [x,y,w,h]} \left(l_{ij}^{\text{true}} - l_{ij}^{\text{pred}} \right)^2 \tag{8}$$

$$\text{IoU} = \frac{|\text{A}\cap\text{B}|}{|\text{A}\cup\text{B}|} \quad \mathcal{L}_{\text{IoU}} = 1 - \text{IoU} \tag{9}$$

$$\text{GIoU} = \text{IoU} - \frac{|\text{C}\backslash(\text{AUB})|}{|\text{C}|} \quad \mathcal{L}_{\text{GIoU}} = 1 - \text{GIoU} \tag{10}$$

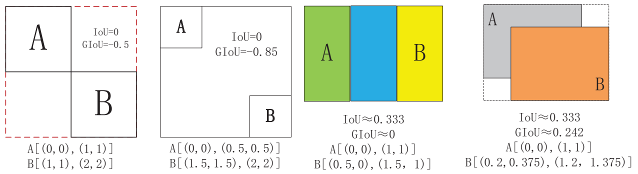

Fig. 10. IoU vs GIoU

Compared with GIoU_Loss, CIoU_Loss considers the scale information of the aspect ratio of the bounding box, and measures it from three aspects of overlapping area, center point distance and aspect ratio, which makes the effect of prediction frame regression better. The CIoU calculation process is (11):

$$\text{CIoU} = \text{IoU} - \frac{\rho^{2\left(b, b^{\text{gt}}\right)}}{\text{c}^2} - \alpha\nu. \tag{11}$$

Where b and b^{gt} represent the center points of the predicted border and the real border, respectively, $\rho^{2(b,b^{gt})}$ represents the Euclidean distance between the center points of the predicted frame and the real frame, that is, d and c in the Fig. 11 represent the diagonal distance of the smallest closure area that can contain both the predicted frame and the real frame.

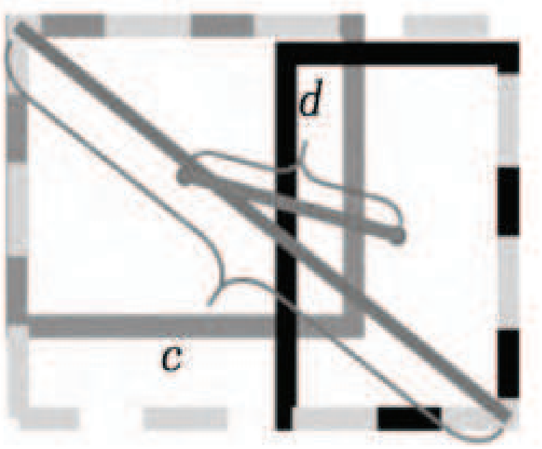

Fig. 11. Diagonal distance of the smallest closure area of predicted frame and the real frame in CIoU

where α is a trade-off parameter to measure the aspect ratio consistency [18], the calculation formula is (12),

$$\alpha = \frac{v}{1 - \text{IoU} + v}$$
$$v = \frac{4}{\pi^2} \left(\arctan \frac{w^{\text{gt}}}{h^{gt}} - \arctan \frac{w}{h} \right)^2 . \tag{12}$$

Where, w and w^{gt} respectively represent the width of the predicted border and the real border, h and h^{gt} respectively represent the height of the predicted border and the real border. The calculation formula of CIoU is (13),

$$\text{CIoU} = 1 - \text{CIoU} \tag{13}$$

IV. Experiment and Analysis

A. Data Set

The experimental data set includes 5,000 images is all collected from Internet, which comprehensively include various scenes images, covering situations where individual wears or does not wear a mask, and multiple people wear or do not wear a mask. According to the ratio of 9:1, the set is divided into training and verification set. An example of the data set is shown in Fig. 12 . The data set is in PASCAL VOC format, and LabelImg [19] is used to label the images, including two categories of mask and face, where mask indicates that the person to be detected has correctly worn a mask; face indicates that the person is not wearing a mask.

Fig. 12. Partial images of the data set

B. Experimental Environment and Parameter Configuration

The experimental environment: the CPU is AMD Ryzen75800H which main frequency is 3.20GHz, the memory is 16GiB, the NVIDIA graphics card which driver is NVIDIAGeForceRTX3050Ti and CUDA10.1, the system operation is Win10, and the deep learning framework uses PyTorch-1.9.0 , using LableMe as the labeling tool, the programming language is Python3.8.

Parameter settings in the training phase: this algorithm uses 90% of 5000 images as the training set, adopts the data parallel method for training, sets the number of generations (epoch) to 100, the optimization method is stochastic gradient descent (SGD), and the size of bachsize is 16, the size of the image is 640 × 640.

Test phase parameter settings: this algorithm uses 10% of 5000 images as the test set. In the test phase, load the trained weight file and set the image size to 640 × 640.

C. Performance Comparison

1) Evaluating Indicators

This paper uses precision (P), recall (R) and average precision (mAP) as an evaluation index for whether wears a mask or not. The precision indicates the proportion of the samples classified as positive samples by the classifier that are actually positive samples, that is, the probability of splitting samples. The (14) is the formula of precision:

$$\text{precision } = \frac{t_p}{t_{p+}f_p}, \tag{14}$$

where, t_p represents the number of samples that are actually positive samples and are classified as positive samples by the classifier, and f_p represents the number of samples that are actually negative samples but are classified as negative samples by the classifier.

The recall means that the number of positive samples classified by the classifier as positive samples accounts for the proportion of positive samples in the full sample. The formula (15) for calculating the recall rate.

$$\text{recall } = \frac{\mathrm{t_p}}{\mathrm{t_p} + \mathrm{f_n}} \tag{15}$$

where, f_n represents the number of positive samples that are actually classified as negative samples by the classifier.

Average Precision Mean (mAP) is the arithmetic mean of the average precision (AP) of each category. This indicator is a comprehensive measure of the detection target. Common mAPs include mAP@.5 and mAP@.5:.95, which represent the average precision when the threshold is set to 0.5 and 0.5 to 0.95. The formula for mAP is (16).

$$\mathrm{mAP} = \frac{1}{\mathrm{N}} \sum_{\mathrm{i=1}}^{\mathrm{N}} \mathrm{AveP(i)} \tag{16}$$

where, N is the number of categories, and AveP(i) is the average precision of the i-th category.

2) Performance Comparison

In order to verify the performance of our optimal YOLOv5 algorithm, we not only compared it with the YOLOv5 algorithm, but also compared it with the YOLOv3 and YOLOv4 detection algorithms. The detection performance comparison results of the four algorithms in the same data set are shown in Table I.

TABLE I
OPTIMAL YOLOV5 COMPARE WITH PREVIOUS YOLOS

Model	P/%	R/%	mAP@.5%	mAP@.5:.95:%
YOLOv5s	0.768	0.813	0.789	0.374
Optimal YOLOv5	0.830	0.833	0.818	0.389
YOLOv4	0.754	0.789	0.784	0.369
YOLOv3	0.751	0.790	0.780	0.364

It can be seen from Table I that the optimal YOLOv5 has a precisian of 83%, recall of 83.3%, and mAP@.5 of 81.8% in the same data set for training and testing. All three indicators are higher than YOLOv5, YOLOv3 and YOLOv4 detection algorithms.

3) Training Process Performance and Loss

The loss function of the YOLOv5 network consists of three parts: Lbox is the positioning loss, which is used to measure the coordinate positioning error of the prediction frame; Lobj is the confidence loss function, which reflects the confidence error of the prediction frame; Lcls is the classification loss function, it reflects the error of the target category by the prediction box. Lbox uses the CIoU_Loss [20] function, Lobj and Lcls use the cross-entropy loss function. The convergence curve of each loss function during the YOLOv5 training loss function process is shown in Fig. 13 and the accuracy curve is shown in Fig. 14. The training process is set to automatically stop iterations when it tends to be stable. It can be seen from these two figures that Lbox, Lcls, and Lobj drop sharply during the 0-30 iterations, and then decrease slowly during the subsequent training process. After 100 iterations, the loss value gradually stabilizes, while the mAP accuracy is stable at around 0.818. It is clear that the network worked well during the training phase.

4) Detection Result Comparison

In order to compare the actual detection effect of the optimal YOLOv5 model and the original YOLOv5 model more intuitively, the defect images of the test set were applied

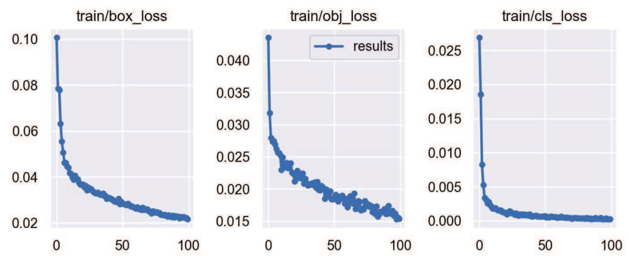

Fig. 13. Loss function curves of Optimal YOLOv5

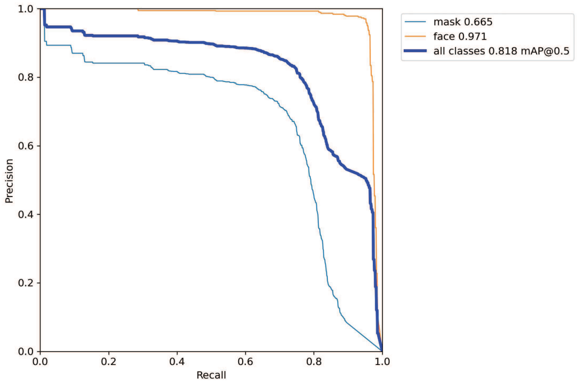

Fig. 14. Precision curve of Optimal YOLOv5

to two models as well, Fig. 15 shows the detection effect of the original YOLOv5 model, and Fig. 16 shows the detection effect of the optimal YOLOv5 model. It can be seen that the original YOLOv5s model has missed detection for some targets, and the confidence level is generally low. Compared with the original YOLOv5s model, the confidence of the optimal YOLOv5 model detection target has been significantly improved. In addition, the localization of defects is more accurate.

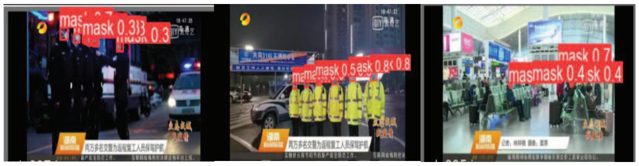

Fig. 15. detection result of original YOLOV5s model

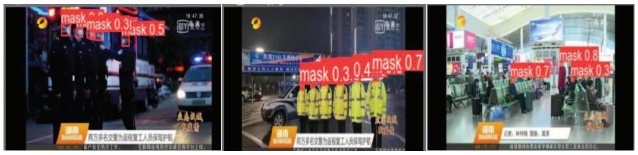

Fig. 16. detection result of OPTIMAL YOLOV5 model

V. Conclusion

In this paper, we added the detection layers and added attention mechanism to original YOLOv5 algorithm, optimizing loss function for original YOLOv5 algorithm as well. Did all of these to optimize the YOLOv5 algorithm for mask-wearing detection. The experimental results of optimal algorithm showed that the precision is 83%, recall is 83.3% and mAP is 81.8%, all of measurements exceed YOLOv3, YOLOv4 and the original YOLOv5 target detection algorithm. Our work has a certain practical significance in fact that it promotes the automation and intelligence for mask-wearing detection.

References

[1] Liu, Shao, Sos S. Agaian. ”COVID-19 face mask detection in a crowd using multi-model based on YOLOv3 and hand-crafted features.” Multimodal Image Exploitation and Learning 2021. Vol. 11734. SPIE, 2021.

[2] Mercaldo, Francesco, Antonella Santone. ”Transfer learning for mobile real-time face mask detection and localization.” Journal of the American Medical Informatics Association 28.7 (2021): 1548-1554.

[3] Arya, Chandrakala, H Pandey,et al. ”Object detection using deep learning: A review.” Journal of Physics: Conference Series. Vol. 1854. No. 1. IOP Publishing, 2021.

[4] Feng, Guochen, Chen, Yanyan, Chen, Ning et al. ”Research on Automatic Helmet Recognition Technology Based on Machine Vision.” Mechanical Design and Manufacturing Engineering 44.10 (2015): 39-42.

[5] Li, Meiling, Wang, Fu, Jing HuiXiao et al. ”Road Extraction of High-Resolution Remote Sensing Imagery.” Remote Sensing Information 31.2 (2016): 64-68.

[6] Zhang, Xiubao, Lin, Ziyuan, Tian, Wangxin et al. ”Face Wearing Mask Recognition Technology in All-weather Natural Scenes.” Science China:Information Science 50.7(2020):11.

[7] Xu, Yunhui, Cao, Yilin, Liu, Yiwei. ”Research on pedestrian detection based on improved SSD algorithm.” 2021 international conference on information science, parallel and distributed systems (ISPDS). IEEE, 2021.

[8] Deng, Huangxiao. ”Method of mask wearing detection based on transfer learning and RetinaNet.” Electronic Technology and Software Engineering 5 (2020): 209-211.

[9] Wang, Yihao, Ding, Hongwei, Li, Bo, Yang, Zhijun et al. ”Mask wearing detection algorithm based on improved YOLOv3 in complex scenes.” Computer engineering 46.11 (2020): 12-22.

[10] Tan, Shilei, Lu, Gonglin, Jiang, Ziqiang, Huang,Li. ”Improved YOLOv5 network model and application in safety helmet detection.” 2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR). IEEE, 2021.

[11] Redmon J, Divvala S, Girshick R et al. ”You only look once: Unified, real-time object detection.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[12] Bochkovskiy, Alexey, Wang, Chienyao, Liao, Hongyuan. ”Yolov4: Optimal speed and accuracy of object detection.” arXiv preprint arXiv:2004.10934 (2020).

[13] Shi,Wenzhe, Jose Caballero, Ferenc Huszár et al. ”Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[14] Hu, Jie, Li, Shen, Sun, Gang. ”Squeeze-and-excitation networks.” Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[15] Yu, Caixia, Zhao, Jingtao, Wang, Yanfei. ”Seismic detection method for small-scale discontinuities based on dictionary learning and sparse representation.” Journal of Applied Geophysics 137 (2017): 55-62.

[16] Gavrila, Dariu M., Stefan Munder. ”Multi-cue pedestrian detection and tracking from a moving vehicle.” International journal of computer vision 73 (2007): 41-59.

[17] Jiang, BoruiLuo, Rui,xuanMao, JiayuanXiao et al. ”Acquisition of localization confidence for accurate object detection.” Proceedings of the European conference on computer vision (ECCV). 2018. for object detection and instance segmentation.” IEEE Transactions on Cybernetics 52.8 (2021): 8574-8586.

[18] Sethi, Shilpa, Mamta Kathuria, Trilok Kaushik. ”Face mask detection using deep learning: An approach to reduce risk of Coronavirus spread.” Journal of biomedical informatics 120 (2021): 103848.

[19] Tzutalin, D. ”tzutalin/labelImg.” (2015).

[20] Zheng, Zhaohui, Wang, PingWei, Liu,Li, Ye, Jinze et al. ”Distance-IoU loss: Faster and better learning for bounding box regression.” Proceedings of the AAAI conference on artificial intelligence. Vol. 34. No. 07. 2020.

Yang Fan is Master's program at Yunnan Minzu University. His research interests include machine learning and computer vision.

Wu Wang received his B.S. and M.S. degrees from Yunnan University, China, in 2003 and 2007, respectively. Received his Ph.D. degree from Future University Hakodate, Japan, in 2018. He is currently an associate professor at the School of Mathematics and Computer Science at Yunnan Minzu University. His research interest includes ad hoc networks, neural network, and network security.