Proceedings of the 2021 IEEE International
Conference on Intelligence and Safety for Robotics
Nagoya, Japan, March 4-6, 2021

Improved YOLOv5 Network Model and Application in Safety Helmet Detection

Shilei Tan, Gonglin Lu, Ziqiang Jiang, and Li Huang

2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR) | 978-1-6654-3862-9/21/$31.00 ©2021 IEEE | DOI: 10.1109/ISR50024.2021.9419561

Abstract—**Due to the complex background of the building site and the diverse sorts of construction personnel, relying on traditional manual inspections and video surveillance methods to detect the wearing of personnel helmets has poor timeliness and missed inspections. This article provides a method based on deep learning to resolve the above issues. First, improvement based on YOLOv5, added a functionality detection scale to allow it to get smaller targets; second, by introducing the DIoU-NMS instead of NMS, DIoU also considers the overlap area and the center distance of the two boxes, making it more accurate in suppressing the predicted bounding box. The experimental results show that the proposed algorithm significantly improves the accuracy compared to the YOLOv5 network model, and detection speed is 98 frames per second, which can meet the needs of real-time detection.**

I. INTRODUCTION

In recent years, accidents caused by violations of construction site rules and regulations have occurred frequently, causing massive loss of lives and property. Among them, the compliant wearing of helmets is the most basic regulations and rules for engineering construction. However, due to ineffective onsite supervision and low safety awareness among workers, safety accidents are frequently caused. To improve this type of situation, it is necessary to detect the state of wear of the helmets of construction workers in real-time, to discover it and to prevent it in time and to prevent the accident.

Currently, the construction site relies primarily on manual inspection and video monitoring for detection. This method does not detect all meteorological conditions, and at the same time, it is easy to miss the inspection because of the angle of the camera. With the development of deep learning technology over the past few years, it has been extensively used in the area of target recognition [1-4]. Deep learning is broken down into two categories. Category one has two steps. Firstly, candidate frames are generated and then classified. The first category is based on region generation. Typical methods are R-CNN [5], Faster R-CNN [6-7], SPP-net [8]. This kind of method has a high precision of detection, but it takes a lot of time. The other type uses an end-to-end method to predict the whole image and finish detecting and classifying the target position at the same time. Representative methods include primarily SSD [9-10], YOLO [11-14]. Such methods have a rapid detection rate, but the precision is reduced. Most of the researches has gone into the detection of motorcycle helmets, but the detection of helmets is relatively rare. In [15], a detection algorithm for motorcycle helmets is proposed, using a Hough circular transformation and directional gradient histogram descriptor to extract the helmet image attributes, and use multi-layer perceptron classifiers to classify targets. Linu Shine [16] proposed two helmet detection methods, one based on manual features, and the other using CNN. The experimental results indicate that the proposed CNN model has the best execution in terms of accuracy, while the feature-based model has a faster detection speed. The system proposed by B. Yogameena [17] is based on faster R-CNN to detect motorcycles in labeled foreground targets to ensure the presence of motorcyclists. Subsequently, faster R-CNN has also been used for detecting motorcyclists with or without helmets. Finally, a string of characters encoding the CNN model and a space converter is used to identify the license plate numbers of motorcyclists without helmets.

In this letter, we have proposed an improved YOLOv5 algorithm that is primarily reflected in two aspects. First, a model detection scale was added to make it more suitable for detecting small targets such as safety helmets at construction sites to avoid missed detection. Secondly, using DIoU-NMS instead of NMS as the predicted bounding box suppression makes it more suitable for boundary box suppression. Adding the distance between the center points of the two frames makes the predicted bounding box more accurate.

II. IMPROVEMENT OF YOLOV5 NETWORK MODEL

YOLOv5 was offered by Ultralytics in May of this year, and its detection speed and precision is superior to the previous YOLOv3 and YOLOv5 algorithms on the COCO dataset. The YOLOv5s weight data are 27M, which equates to 1/9 of the YOLOv4 version, which allows it to have a faster

Research supported by the National Natural Science Foundation of China under Grant 61933012 and 61860206008.

The authors are with the School of Automation, Chongqing University, Chongqing 400044, China.

Shilei Tan (e-mail: tanshilei@cqu.edu.cn); Gonglin Lu (e-mail: lugonglin @cqu. edu.cn); Ziqiang Jiang(e-mail:jiangziqiang@cqu.edu.cn); Li Huang (e-mail: huang_li@cqu.edu.cn).

detection speed. Agreeing to the official information, it can process an image at the fastest speed of 0. 007s, which can meet real-time detection. At the same time, it is suitable for deployment in onboard devices with limited computation, memory, and energy consumption requirements. For example, this section may apply the training template for monitoring equipment at construction sites.

YOLOv5 is more effective in detecting large targets, but it is easy to miss detection due to the constraints of the detection scale in terms of compact arrangement and multiple overlapping small targets. This article improves the model's scale and loss function to make it more responsive to small target detection, which will be presented separately below.

A. Improved Network Model Detection Scale

The map of the high-level characteristics of the network model has a wider receptive field, focusing on the expression of abstract semantic information, which is appropriate to the target classification tasks, but the resolution is low, and the ability to represent localization detail and information is poor. The deeper the network layer, the greater the loss of information to smaller targets in the downward sampling process. The low-level features map has a small receiver field and a high resolution, which corresponds to the size of the small target. The low-level features map gives more attention to the details and position information of the target, which is more beneficial for extracting the contour, color, and other detailed features and regression of the small target position. YOLOv5 contains three feature detection scales. For example, when the input image size is 640*640, the three detection scale ranges are 80*80, 40*40, 20*20. This means that when the small target is smaller than 8*8 pixels, the template is missed. Aimed at this situation, this article adds a detection scale to the model, 160*160, 640 divided by 160 is 4， so that the model can detect targets of 4*4 pixels or more, which may meet the detection needs of smaller targets. Fig.1 shows the improved network pattern.

It can be seen from the figure that the model is mainly composed of Input, Backbone, Neck, and Output. The input terminal mainly carries out the improvement of the Mosaic data and the adaptive calculation of the anchoring frame. Backbone is mainly composed of Focus and CSP. The key to Focus is slicing operations and increasing the depth of the network. For instance, the original image 416*416*3 is inserted into the Focus structure, and through the slicing operation, it becomes a feature map of 208*208*12 to deepen the network depth. CSP has two structures, CSP1_X and CSP2_X, which are used in the Backbone and Neck networks, respectively. CBL is mean Convolution, Batch normalization, and LeakyRelu. Neck strengthens the fusion power of network features through up-sampling and down-sampling operations. The output side uses CIoU as the loss function of the bounding box and uses linear weighted non-maximum suppression to filter the target box.

B. DIoU-NMS Prediction Bounding Box Suppression

In traditional NMS, the IoU index is often used to suppress redundant detection boxes, in which the overlapping area is the only factor, and error suppression is often caused for occlusion. DIoU-NMS regards DIoU as the criterion of NMS, because not only the overlap area should be considered in the suppression criterion, but also the center point distance between the two boxes, and DIoU considers the overlap area and the center distance of the two boxes at the same time. For the predicted box N with the highest score, the k_i update formula of DIoU-NMS can be formally defined as:

$$k_i = \begin{cases} k_i, IoU - R_{DIoU}(N, B_i) < \varepsilon \\ 0, IoU - R_{DIoU}(N, B_i) > \varepsilon \end{cases} \tag{1}$$

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{2}$$

$$R_{DIoU} = \frac{\rho^2(b, b^{gt})}{c^2} \tag{3}$$

In the formula, IoU represent the intersection ratio between the bounding box and ground truth; $\rho^2(b, b^{gt})$ represent the distance between the bounding box and the center end of the ground truth; c^2 represents the diagonal length of the minimum enclosing rectangular box of the two boxes; ε is the manually set NMS threshold; B is the anchor number corresponding to each grid; k_i is the classification score for different categories. DIoU-NMS suggests that the two predicted bounding boxes with farther center points may be located on different objects and should not be deleted like the traditional NMS method.

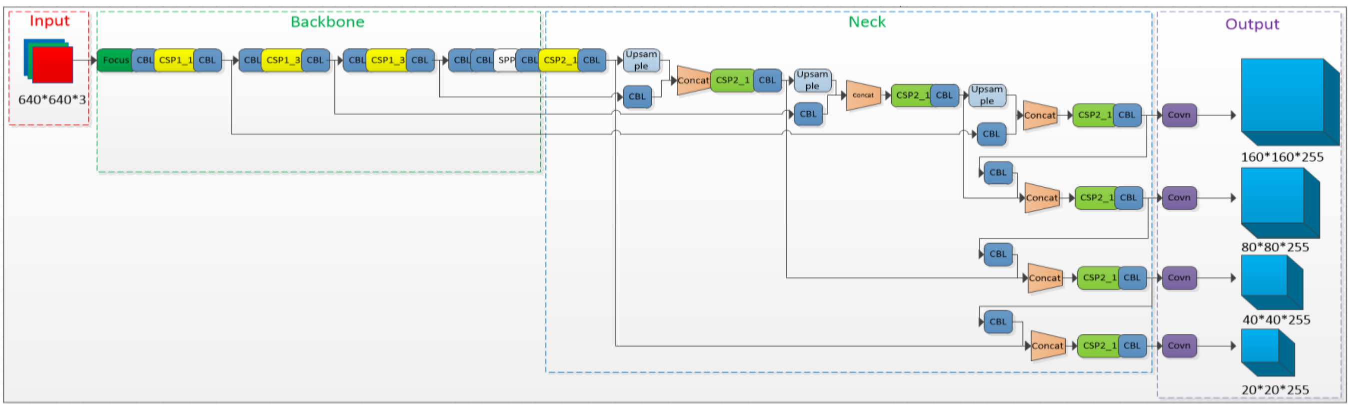

Fig.1 Improved YOLOv5 network model

III. Experimental Results and Analysis

A. Dataset Creation and Environment Construction

The data source of the safety helmet wearing dataset is mainly obtained through Internet crawlers. In total, 6,000 images have been collected to create training and test datasets. The datasets include two categories, Helmet and Head. Of these, the helmet indicated that staff wore the helmet properly, and the head indicated that staff did not wear the helmet. The image target label uses LabelImg, and the labeled file uses xml as the suffix, and the file name is consistent with the image name. The data set is divided into 5000 pictures as the training set and 1000 pictures as the test set.

The network model is based on Tensorflow architecture. The configuration of the experimental environment is set out in the table below.

TABLE I
EXPERIMENTAL ENVIRONMENT CONFIGURATION

Parameter	**Configuration**
CPU	Intel(R) Core (TM) i7-3770 CPU @ 3.40GHz
GPU	NVIDIA GeForce GTX TITAN X
System Environment	Ubuntu 16.04
Language	Python 3.6
Acceleration Environment	CUDA 9.0

B. Network Model Training

In the network model training phase, set the iteration batch size to 64, the attenuation coefficient to 0.0005, the total number of iterations to 1700, and the initial learning rate to 0.001. When the number of iterations reaches 1200 and 1500, the learning rate is reduced to 0.0001 and 0.00001. The CIoU loss function in the training phase is shown in Fig.2. It can be seen from the figure that the model has converged after 1400 iterations.

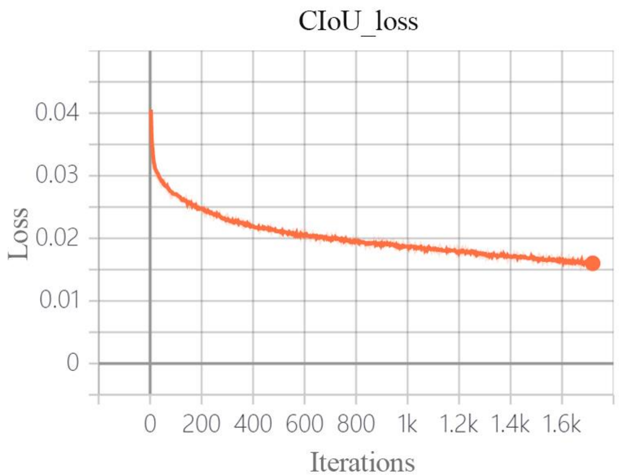

Fig.2 The model training CIoU loss function.

The class loss function is illustrated in Fig.3. The model makes a good distinction between categories. The target loss function is illustrated in Fig.4. After 1000 iterations, the loss is close to zero, which indicates that the network model can precisely identify the target in the training graph.

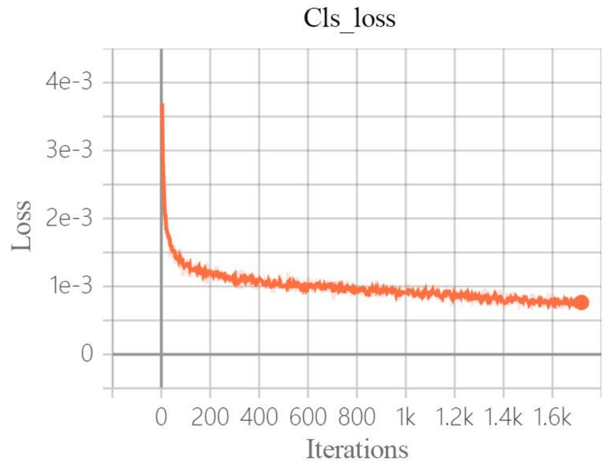

Fig.3 The model training classes loss function.

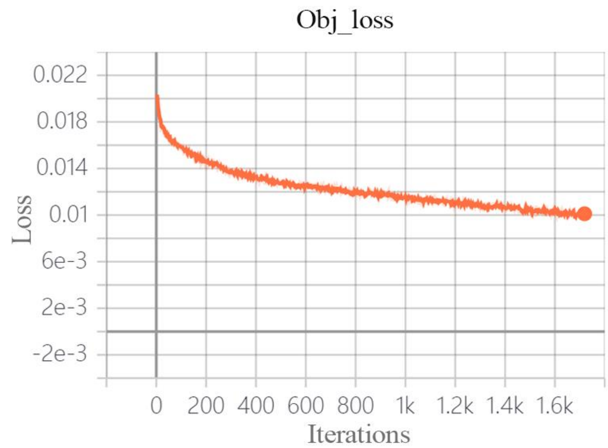

Fig.4 The model training object loss function.

C. Experimental Results and Analysis

To evaluate the performance of the model, the precision rate (P_r), the recall rate (R_e), the average precision (AP) and detection speed (S_p)are used as evaluation indicators. The specific expression is as follows.

$$P_r = \frac{TP}{TP+FP} \quad (4)$$

$$R_e = \frac{TP}{TP+FN} \quad (5)$$

$$AP = \int_0^1 P_r(R_e)\, dR_e \quad (6)$$

In the formula, True Positive (TP), is judged as a positive sample, it is also a positive sample in fact; False Positive (FP), is judged as a positive sample, but it is a negative sample in fact; False Negative (FN), is judged as a negative sample, but it is a positive sample in fact. The test results of the model are shown in Table II.

TABLE II
MODEL SAFETY HELMET WEARING DATASET TEST RESULTS

Models	P_r/%	R_e/%	AP /%	S_p (Frame/sec)
YOLOv5	86.28	94.15	92.12	106
Improved YOLOv5	92.24	96.27	95.68	98

(a1) (b1)

(a2) (b2)

(a3) (b3)

Fig.5 Comparison of recognition effects of different algorithms. (a1-a3) and (b1-b3) are YOLOv5 and improved YOLOv5, respectively, to identify the effects of the two algorithms on the safety helmet data sets.

We can see that the improved algorithm has a certain degree of improvement in the accuracy. Although the increased detection layer reduces the detection speed, it can still meet the needs of real-time detection.

The actual detection effect of the algorithm is shown in Fig.5. It can be seen from the figure that YOLOv5 has missed the inspection in the pictures (a1) and (a2), including the safety helmet and the head respectively, and the wrong inspection in (a3), which regards the wooden pile as the safety helmet. However, the improved algorithm effectively solves the above problems and has good robustness in a complex environment.

IV. CONCLUSIONS

Targeting the problems of missed detection and slow convergence of the models in the detection of small targets by the YOLOv5 algorithm, an improved schema has been proposed. Adding a detection scale and using DIoU-NMS instead of NMS as the predicted bounding box suppression makes it more suitable for detecting small targets. With the verification of the model in the detection of the wearing of the helmet, the experimental results show that the improvement of the precision of detection of the algorithm is relatively ideal. Later, the algorithm can be transplanted to monitoring equipment or inspection robot for practical application.

REFERENCES

[1] Han, Xiaofeng, et al. "Research on Remote Sensing Image Target Recognition Based on Deep Convolution Neural Network." *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 34, no. 5, May. 2020.

[2] Ma Y, Zhu L, Li Y. "HRRP-based target recognition with deep contractive neural network." *Journal of Electromagnetic Waves & Applications*, vol. 33, no. 7, pp. 911–928, May. 2020.

[3] Wang, Jia, et al. "Research on Automatic Target Detection and Recognition Based on Deep Learning." *Journal of Visual Communication and Image Representation*, vol. 60, pp. 44-50, Apr. 2019.

[4] YANG Y F, Cao Y. "Image Insulator Target Detection Based on Deep Learning for UAV". *Laser Journal*, vol41, no.10, pp.63-66, Oct. 2020.

[5] R.B. Girshick, J. Donahue, T. Darrell, J. Malik. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587, Jun. 2014.

[6] Liu, Zhigang, et al. "Detection Approach Based on an Improved Faster RCNN for Brace Sleeve Screws in High-Speed Railways." *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 7, pp. 4395-4403, Jul. 2020.

[7] Han, Cen, G. Gao, and Y. Zhang. "Real-time small traffic sign detection with revised faster-RCNN." *Multimedia Tools and Applications*, vol. 78, no.10, pp. 13263-13278, May. 2018.

[8] K. He, X. Zhang, S. Ren, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 37, no.9, pp.1904–1916, Sep. 2015.

[9] Hu, Xun, et al. "MobileNet-SSD MicroScope using adaptive error correction algorithm: real-time detection of license plates on mobile devices." *IET Intelligent Transport Systems*, vol. 14, no.2, pp.110-118, Feb. 2020.

[10] Chen, Zuge, et al. "SSD-MSN: An Improved Multi-Scale Object Detection Network Based on SSD." *IEEE Access*, vol. 7, pp. 80622-80632, 2019.

[11] Yin, Yunhua, H. Li, and W. Fu. "Faster-YOLO: An accurate and faster object detection method." *Digital Signal Processing*, vol.102, Jul. 2020.

[12] Huang, Zhanchao , Wang, Jianlin. "DC-SPP-YOLO: Dense Connection and Spatial Pyramid Pooling Based YOLO for Object Detection." *Information Sciences*, vol. 522, pp.241-258, Jun. 2020.

[13] Sadykova, Diana, et al. "IN-YOLO: Real-time Detection of Outdoor High Voltage Insulators using UAV Imaging." *IEEE Transactions on Power Delivery*, vol. 35, no. 3, pp.1599-1601, Jun. 2020.

[14] Nguyen, Duy Thanh, et al. "A High-Throughput and Power-Efficient FPGA Implementation of YOLO CNN for Object Detection." *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 8, pp.1861-1873, Aug. 2019.

[15] Silva, Romuere R. V. E, K. R. T. Aires, and R. D. M. S. Veras. "Detection of helmets on motorcyclists." *Multimedia Tools and Applications*, vol. 77, no. 5, pp.5659-5683, Mar. 2018.

[16] Shine, L, Jiji, CV. "Automated detection of Helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN." *Multimedia Tools and Applications*, vol. 79, no. 19-20, pp.14179-14199, May. 2020.

[17] Yogameena, B., K. Menaka, and S. S. Perumaal. "Deep learning-based helmet wear analysis of a motorcycle rider for intelligent surveillance system." *Intelligent Transport Systems Iet*, vol. 13, no. 7, pp.1190-1198, Jul. 2019.