DCMS-YOLOv5: A Dual-Channel and Multi-Scale Vertical Expansion Helmet Detection Model Based on YOLOv5

Yulu Liu, Ying Tian

Abstract—**This paper proposes Dual Channel Multi Scale YOLOv5 (DCMS-YOLOv5), an improvement of YOLOv5, to increase the recognition accuracy of helmet detection methods. The model has a dual-channel architecture, and feature extraction and fusion are performed in a lateral connection to enhance the model's ability to capture targets in complex scenes. The features and local dependencies are characterized at multiple scales to improve the model's ability to capture small targets. The model is validated with the safety helmet-wearing dataset (SHWD) and compared with other methods. The experimental results show that the DCMS-YOLOv5 model provides high helmet detection accuracy, performs excellent for detecting small targets, and has strong generalization ability.**

Index Terms—**Dual Channel, Multi-Scale Extension, Safety Helmet Detection, Small Target, YOLOv5**

I. Introduction

WRARING a helmet is a requirement in many industries, such as manufacturing and construction, for safety reasons. Workers often have accidents because they do not wear safety helmets as required [1]. Two difficulties exist in image-based helmet detection [2]. First, due to the complex environment and numerous people at the construction site, the model requires high feature extraction ability. Second, the lighting conditions at the construction site may not be optimal, and helmets have different styles and colors; thus, the misrecognition rate is high [3,4,5,6].

Research on safety helmet detection can be divided into two categories: two-stage target detection algorithms based on region generation and regression-based one-stage target detection algorithms [7]. Numerous scholars have conducted experimental research on the two-stage target detection algorithm faster regions with convolutional neural network features (Faster-RCNN) in recent years. For example, Long et al. proposed a Faster-RCNN-based helmet detection framework [8]. Wu et al. fused the feature layers obtained from multiple stages in the Faster-RCNN and performed multi-scale detection [9]. They modified the size of the candidate target frame and improved the detection scale of the helmet. Li et al. utilized multi-scale training and incorporated an anchor strategy to enhance the robustness of the original Faster-RCNN. They proposed the Online Hard Example Mining (OHEM) [10] framework for model optimization. Although the two-stage method has high recognition accuracy, there are many model parameters, and the running speed is slow. The one-stage target detection algorithm based on regression is a relatively popular detection method. For example, Zhang et al. increased the DenseNet structure by improving the structure of the one-stage target detection algorithm YOLOv3. The K-means algorithm was used to re-cluster the candidate frame to facilitate helmet detection. Yang et al. used the YOLOv3 algorithm to detect worker faces [12] and estimated the helmet area based on the relationship between the helmet and the face. They used the Histogram of Oriented Gradients (HOG) to extract the feature vector of the sample, adopted Support Vector Machine (SVM) to detect helmets. Xu et al. combined the Single-Shot Detector (SSD) algorithm with the improved MobileNet [13] to overcome model training difficulties with transfer learning strategies. Ben et al. proposed a helmet detection method based on an improved YOLOv4 algorithm [14]. It utilizes the K-means algorithm to cluster the dataset and multi-scale network training to improve the model's ability to detect helmets at different scales. Sadiq et al. further fine-tuned the fusion ability of YOLOv5, incorporated a blur-based image enhancement module, and proposed the FD-YOLOv5 M model for helmet detection [15]. Duan et al. proposed an end-to-end safety helmet detection algorithm based on scene correlation [16] and used a novel loss function and training strategy to improve the detection accuracy. Zhou et al. used YOLOv5 as the base model [17] for helmet detection by adjusting the parameters of the model class. Zhang et al. designed a deformable bilateral aggregation network (DBDA) based on MobileNetV3 for helmet detection [18] to improve the model's ability to detect helmets with different shapes and at different scales. Although these methods can detect safety helmets, most have a simple structure and low feature extraction ability. Even models with multi-scale strategies have provided low performance; thus, the recognition accuracy must be improved.

We propose a detection model (DCMS-YOLOv5) that uses a dual-channel strategy (stage 1 and stage 2) as the backbone network. The eigenvalues of the two channels are

Manuscript received September 2, 2022; revised January 28, 2023.

This work was funded by the foundation of Liaoning Educational Committee under the Grant No. LJKZ0310.

Yulu Liu is a postgraduate student of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China. (e-mail: lyl66289290@163.com).

Ying Tian is a Professor of School of Computer Science and Software Engineering, University of Science and Technology Liaoning, Anshan, 114051, China. (corresponding author to provide phone: 138-9801-5263; e-mail: astianying@126.com).

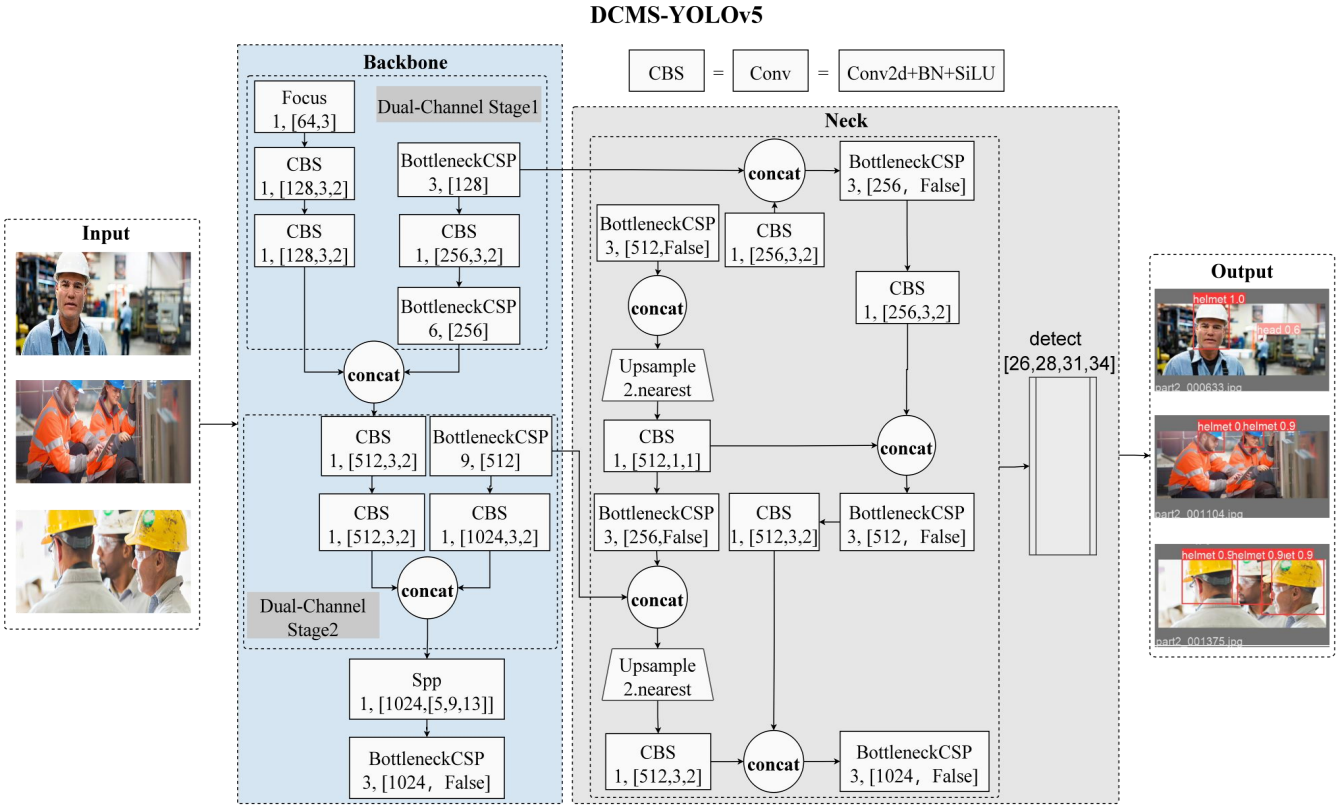


Fig. 1. Model overall architecture diagram

fused and interacted to improve the helmet detection rate and reduce background noise interference. A scale feature layer is incorporated into the neck, and an interaction channel between deep information and shallow information is established using multi-scale vertical expansion. The BottleneckCSP module is used to acquire deep information. The model is validated using the safety helmet-wearing dataset (SHWD), showing an average accuracy of 95.70%. The proposed model has significantly higher accuracy and better feature extraction performance at different scales than other methods [18].

II. Method

This section describes the baseline model (YOLOv5) and the proposed improved safety helmet detection model (DCMS-YOLOv5). The proposed two-channel lateral connection and multi-scale vertical scaling method are presented in detail.

A. Architecture

The DCMS-YOLOv5 includes the input, backbone, neck, and prediction modules. Data augmentation (Mosaic) is performed on the dataset in the input module. The adaptive method of YOLOv5 is used for anchor box calculation, enabling the model to determine the optimal frame value adaptively during training. The image scaling ratio is determined adaptively, and the image size is 640×640. A dual-channel lateral connection is used in the backbone for feature extraction. In order to prevent feature loss during feature extraction, a cross-channel multi-branch fusion model structure was constructed. A bottom-down lateral connection is established in the neck to create a feature pyramid network (FPN), and a bottom-up route is added to the path aggregation network (PAN) to extract the feature location. The model includes a detection layer that is extended vertically to enhance the ability to capture small targets. The model architecture diagram is shown in Fig. 1.

B. YOLOv5

YOLOv5 consists of YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The model is updated continuously. We used YOLOv5 version 6.0. YOLOv5 has a small model size and high recognition accuracy. The backbone consists of three modules: Conv, CSP1_X, and spatial pyramid pooling-fast (SPPF). The Conv module is the convolution layer. The batch normalization layer (BN) and activation function provide the final output information through matrix point multiplication and summation operations. The fuseforward function is used to integrate the Conv layer and BN layer to accelerate inference. The CSP1_X module divides the feature map into two parts. The first part passes through the Conv layer, bottlenecks, and Conv2d to obtain the output information y_1. The second part passes through the Conv to obtain y_2 and merges y_1 and y_2. The SPPF module converts the feature map into a fixed-size feature vector. There is no size limit for the feature map. The number of channels in the module is halved by performing CBL (Conv, BN, Leaky_ReLU), and maximum pooling is performed twice to obtain y_1 and y_2. The original x, y_1, and y_2 are pooled with the self.m(y_2) three times, and the final CBL operation is performed. The SPPF module is shown in Fig. 2.

The neck of YOLOv5s is a PAN that can detect objects of different sizes and at different scales. The bottom-up path of PANet segments shallow features to improve the information extraction output by the backbone and the generalization ability of the model.

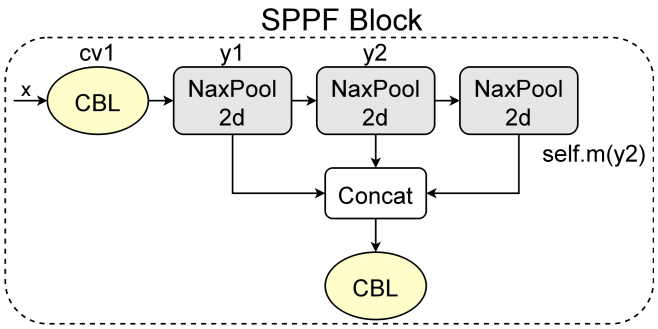


Fig. 2. SPPF module architecture diagram

C. Dual-channel lateral connection

A dual-channel architecture is used to capture more feature information and improve the helmet detection rate inthe complex working environment of the construction site. The model's backbone consists of dual channels (stage 1 and stage 2), and performs cross-channel multi-branch fusion using a lateral connection. Its structure diagram is shown in Fig. 3.

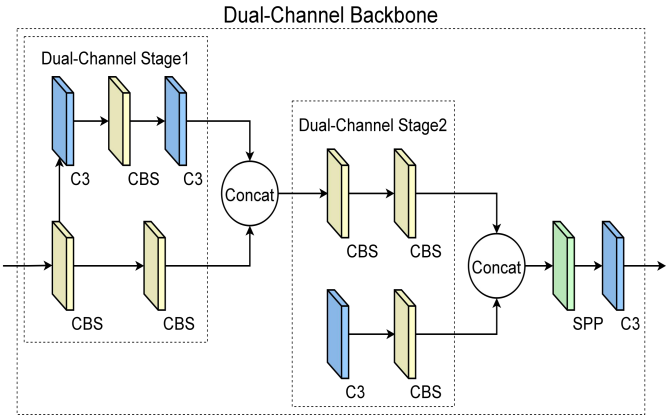


Fig. 3. Double-channel side connection structure diagram

Dual-channel stage 1 contains three CBS modules and two C3 modules. The C3 module performs three convolution operations and uses a double-branch cross fusion for residual feature learning. It uses multiple bottleneck stacks and three standard convolution layers. The other branch has only one convolution module, and the two branches are connected by a Concat layer. The CBS module performs convolution and BN on the input feature map, using a sigmoid-weighted linear unit (Silu) as the activation function. The first CBS module contains 64 convolution kernels, and the second and third CBS modules each contain 128 convolution kernels. During the convolution of the feature map, efficient training is performed by grouping and re-convolution so that the model parameters decrease with an increase in the number of filter groups. The input feature maps are equally divided into g groups to obtain the number of channels C of each convolution kernel $\frac{C_{input}}{g}$, and conventional convolution is performed on each group:

$$F = \frac{C_{input}}{g} \times C_{output} \times C_{input} \times O_1 \times \left(\left(\frac{H_{input} - K_1 + 2 \times P_1}{Stride_1} + 1 \right) \times \left(\frac{W_{input} - K_2 + 2 \times P_2}{Stride_2} + 1 \right) \right) \quad (1)$$

Where F is the number of convolutions on one pixel; C_{input} and C_{output} are the input and output of the number of channels, respectively; O_1 is the output value of a sliding window for a pixel; H_{input} and W_{output} are the input height and width values, respectively; K is the *Kernel* value; P is the Padding value. During the grouped convolution, the number of channels of a convolution kernel is reduced to the original, and the *Parameters* of the grouped convolution are as follows:

$$Parameters = \begin{cases} C_{out} \times \frac{C_{input}}{g} \times (K_1 \times K_2), & bias = False \\ C_{out} \times \frac{C_{input}}{g} \times (K_1 \times K_2 + 1), & bias = Tule \end{cases} \quad (2)$$

Dual-channel stage 1 connects the second CBS module and the first C3 module laterally to form the first channel. The Concat layer operation is performed on the third CBS module and the second C3 module, and the results is used as the output of the next channel.

Dual-channel stage 2 contains three CBS modules and one C3 module. The first and second CBS modules contain 512 convolution cores, and the third CBS module contains 1024 convolution cores. The output result of dual-channel stage 1 is used as the input of the first CBS module of dual-channel stage 2 and is laterally connected with the C3 module to form the second channel, creating the cross-channel multi-branch fused backbone network. The calculation formula for two-channel feature extraction is as follows:

$$Double\ Channel = DC_1 Feature_i \oplus DC_2 Feature_i, i \in [1, \infty) \quad (3)$$

Where DC_1 is the feature information extracted by dual-channel stage 1, and DC_2 is the feature information extracted by dual-channel stage 2.

D. Multi-scale vertical expansion

Since it is difficult to detect small objects in the complex scenes of construction sites, we propose a multi-scale expansion method to improve the detection of small objects. The bottleneck CSP module is used to replace the C3 module to extract deep semantic information from the image.

The bottleneck CSP module consists of a bottleneck module and a CSP module. The bottleneck module adjusts the number of channels to prevent gradient disappearance and explosion during training ss by the bottleneck-residual structure. The feature maps of various levels are fused with the bottleneck. First, the channel number of the image is reduced by half by a convolution layer composed of Conv, BN, and Leaky Relu. Then, the channel number is restored by a 3×3 convolution to ensure that features from different receptive fields are extracted. The CSP module can enrich the gradient splitter, combine the gradient information of different positions, perform feature fusion using, combine the n-th convolution of the first branch with the second branch, and output the final result. The diagram of the module of the library is shown in Fig. 4.

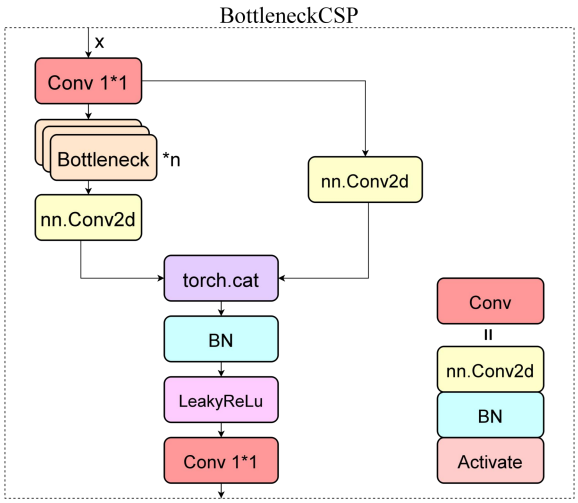


Fig. 4. Library module structure diagram

A focus layer is added to perform the slicing operation and prevent a decrease in the training speed with the increasing model depth. In the model, the input Image size is $640\times640\times3$. The difference between the two dimensions of the rows and columns is calculated to obtain a feature map with a size of $320\times320\times3$. In Focus, a total of Four slicing operations are performed by the focus layer, and the final feature map size is $320\times320\times12$. The focus operation prevents missing features and improves the model's running speed. The slice operation of the focus layer is shown in Fig. 5.

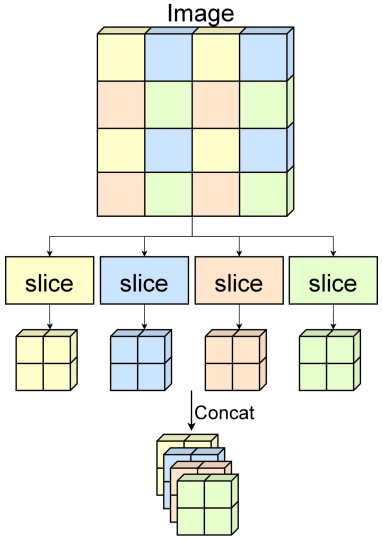


Fig. 5. Focus slice operation

Multi-scale vertical expansion and fusion of PANet are performed to integrate the shallow and deep information, and a new scale feature layer is added to the model. Four convolutions at different scales are used for prediction. This approach strengthens the information transfer between the deep and shallow information and improves the model's detection ability for small targets. The input feature map size of the backbone feature extraction network is 320×320 , and the final output feature map sizes for the 4 scales are 160×160 , 80×80 , 40×40 , and 20×20 . The diagram of the multi-scale vertical expansion is shown in Fig. 6.

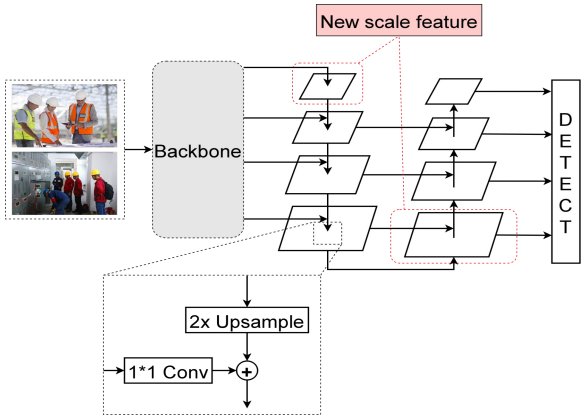


Fig. 6. Multi-scale extended fusion structure diagram

III. Experimental Results And Discussion

A. Datasets

We used the open-source dataset SHWD. The training and testing dataset contains 7,581 images, including 9,044 people wearing helmets and 111,514 people without helmets. The images were acquired under complex lighting conditions, and there are complex backgrounds and small targets. Workers wearing helmets are referred to as helmets, and workers without helmets are referred to as heads. The public dataset is an xml file in PASCAL VOC format and was converted into a txt tag file in the YOLO format. The dataset was divided into a training set and a test set with a ratio of 9:1. The number of training set images in the 7,581 image dataset is 6,823, and the number of test set images is 758 [16].

B. Experimental Configuration

This experiment was carried out on Intel E5-2650 v4 and dual 1080Ti servers, and the operating system was Centos7. Python version 3.8 and Pytorch version 1.9 were used. The learning rates were initialized to 0.01.

C. Evaluation Indicators

We used the Precision, Recall, Average Precision (AP), and mean Average Precision (mAP) as evaluation metrics to evaluate the model performance [19].

Precision: the ratio of the number of correct samples to the total number of samples, as defined in (4).

$$Precision = \frac{TP}{TP + FP} \tag{4}$$

Recall: the ratio of the number of correctly classified samples to the sum of the true positive (TP) and false negative (FN) samples, as defined in (5).

$$Recall = \frac{TP}{TP + FN} \tag{5}$$

AP: the area under the curve where the recall is on the horizontal axis and the precision is on the vertical axis. The larger the value of AP, the higher the average accuracy of the model, as shown in (6).

$$AP = \sum_{i=2}^{n} (r_i - r_{i-1}) p(r_i) \tag{6}$$

Where *TP* refers to the positive sample with correct allocation, *FP* (false positive) refers to the positive sample with incorrect allocation, *TN* (true negative) refers to the negative sample with correct allocation, and *FN* refers to the negative sample with incorrect allocation.

mAP: the average of the AP for all categories in the dataset. It includes the results for different intersection over union (IoU) thresholds, i.e., when 0.5 mAP is 0.5 at the IoU threshold, as shown in (7).

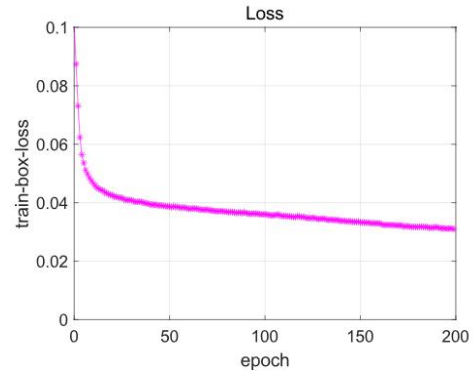
$$mAP = \frac{1}{m}\sum_{i=1}^{m} AP_i \tag{7}$$

Where m is the number of samples in the test set.

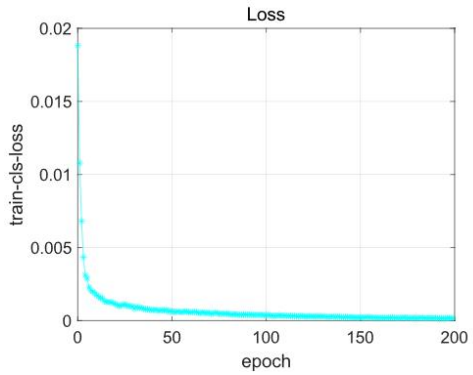
D. Ablation experiment

We first evaluated the performance of the dual-channel lateral connection model (DC-YOLOv5). The number of training epochs was 200, the batch size was -1, the number of epochs for the hyperparameter evolution was 300, and the size of the training set and test set images was 640×640. The loss value during training is shown in Fig. 7. It stabilizes around 0.03 after 200 iterations. The cls loss value approaches 0. The obj loss value stabilizes around 0.02.

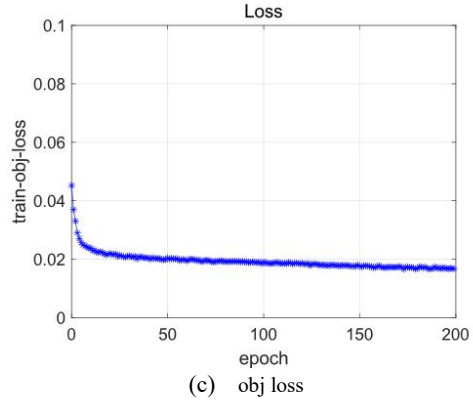
The generalized IoU (GIoU) error between the prediction frame and the calibration frame was calculated to determine whether the anchor frame and the classification were correct. The performance of the DC-YOLOv5 model is excellent. However, the model performance for capturing small targets is unsatisfactory.



(a) box loss



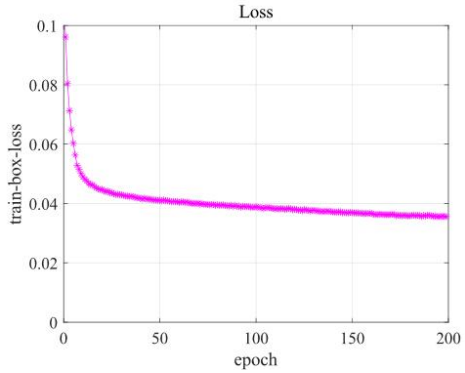
(b) cls loss



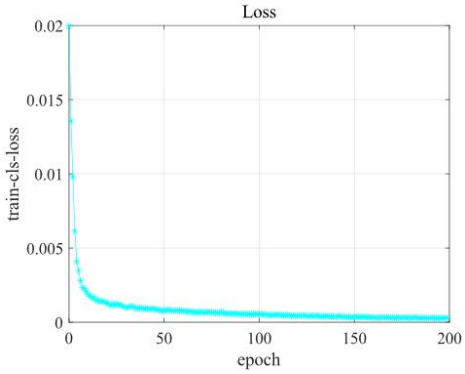
(c) obj loss

Fig. 7. DC-YOLOv5 convergence

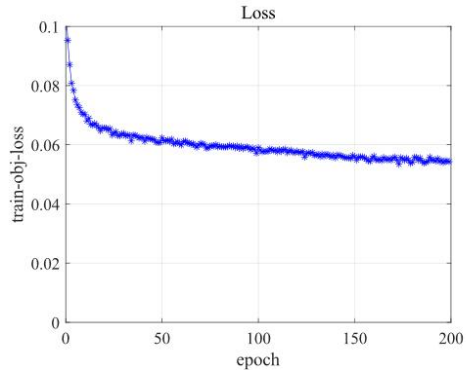
Therefore, the following changes were made to the DCMS-YOLOv5 model. First, a multi-scale structure is established by adding feature layers at different scales. The experiment showed that the depth information extraction performance of the C3 module in the DC-YOLOv5 model was worse than that of the bottleneck CSP module. Therefore, we replaced the C3 module and added the focus layer to improve the detection speed of the model. The performance evaluation indicator settings are the same for the DCMS-YOLOv5 model and the DC-YOLOv5 model. The training loss value is shown in Fig. 8. It stabilizes around 0.03. The cls loss value approaches 0, and the obj loss value stabilizes at 0.015.



(a) box loss



(b) cls loss



(c) obj loss

Fig. 8. DCMS-YOLOv5 convergence

The baseline model YOLOv5 was trained using the same configuration as the proposed model (200 training rounds). The average precision (IoU=0.5) is shown in Fig. 9. The DC-YOLOv5 and DCMS-YOLOv5 models converge rapidly after the first 50 epochs and stabilize after 100 epochs. The model performance is good without overfitting or underfitting. The improved model has significantly higher average accuracy than the original model, indicating that the improvement strategy is appropriate.

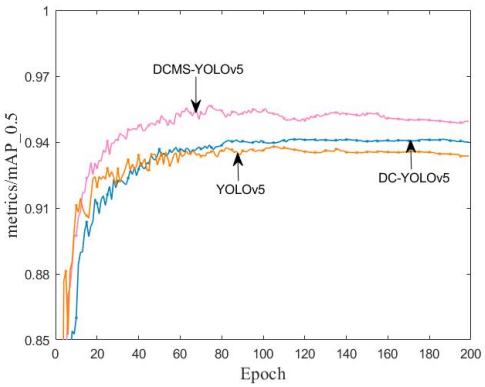


Fig. 9. Performance analysis of mAP_0.5 under different models

Ablation experiments were conducted on the test data set. Table I lists the performances of the improved modules. The DC-YOLOv5 model is based on YOLOv5 and was improved by adding the dual-channel lateral connection. The DCMS-YOLOv5 model is based on the DC-YOLOv5 model and was improved by incorporating vertical expansion and fusion. The average accuracy of the DC-YOLOv5 model (DCMS-YOLOv5 model) is 0.3% (1.9%) higher than the baseline model, indicating excellent performance of the proposed improvements.

TABLE I
ABLATION EXPERIMENTS ON SHWD

Approaches	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95(%)
YOLOv5	93.5	89.6	93.8	61.0
DC-YOLOv5	92.3	89.6	94.1	60.2
DCMS-YOLOv5	94.3	90.6	95.7	61.7

E. Compare of test results

Fig. 10 shows the visualization of part of the detection results. The left side is the DC-YOLOv5 detection diagram, and the right side is the DCMS-YOLOv5 detection effect diagram. Fig. 10(a) is an image of a small target at a short distance, Fig. 10(b) is an image of a small target at a long distance, and Fig. 10(c) is an image of a dense target. It can be seen that the DCMS-YOLOv5 detects the helmet-wearing targets missed by the DC-YOLOv5, and can accurately identify extremely small targets that are far away, and the confidence score is also improved. It shows that the DCMS-YOLOv5 has strong generalization ability in crowded target and small target scenarios.

(a) Small target at close range

(b) Small target at long range

(c) Dense target

Fig. 10 Visualization comparison of results

F. Compare with other models

The performance of the proposed model is compared with other methods. The results are listed in Table II. The proposed model has an mAP of 95.7% on the SHWD dataset. Its mAP is 20.2%, 19.7%, 17.6%, and 10.6% higher than that of SSD, Faster-RCNN, YOLOV3, and MobileNetV3, respectively, several popular target detection methods. It has a 4.7% higher mAP than the model described in Ref. [18]. The precision and recall of the proposed model are also better than that of the other models.

TABLE II
COMPARING WITH OTHER METHODS ON SHWD

Approaches	P (%)	R (%)	mAP@0.5 (%)
SSD [13]	83.2	76.7	76.0
Faster-RCNN[18]	80.8	78.6	75.5
YOLOV3[18]	85.1	80.1	78.1
MobileNetV3[18]	88.6	86.0	85.1
MobileNetV3+DBDA[18]	90.9	92.1	91.0
Ours model	94.3	90.6	95.7

IV. CONCLUSION

This paper proposed the DCMS-YOLOv5 model for helmet detection. The single-channel feature extraction method of the YOLOv5 model was improved. A dual-channel lateral connection backbone network was incorporated, and the feature map was sequentially used in the two channels for feature extraction and fusion. These improvements strengthen the model's ability to extract diverse features. The vertical expansion of the neck improves the model's detection ability by deepening the network level, enhancing the model's ability to detect small targets. A focus layer was added to improve the model's running speed. The model's performance was evaluated on the SHWD dataset, and the performance of the dual-channel lateral connection architecture and the multi-scale scale-up method was analyzed by an ablation study. The results and a comparison with other helmet detection methods indicate that the DCMS-YOLOv5 model has superior performance and excellent generalization ability.

REFERENCES

[1] B. Dai, Y. Nie, W. Cui, R. Liu, and Z. Zheng, "Real-time Safety Helmet Detection System based on Improved SSD," *International Conference on Artificial Intelligence and Advanced Manufacture*, pp. 95-99, 2020.

[2] M. Jin, J. Zhang, X. Chen, Q. Wang, and X. Wang, "Safety Helmet Detection Algorithm based on Color and HOG Features," *2020 IEEE 19th International Conference on Cognitive Informatics & Cognitive Computing*, pp. 215-219, 2020.

[3] G. Yan, Q. Sun, J. Huang, and Y. Chen, "Helmet detection based on deep learning and random forest on UAV for power construction safety," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 25, no. 1, pp. 40-49, 2021.

[4] N. Li, X. Lv, S. Xu, Y. Wang, Y. Wang, and Y. Gu, "Incorporate online hard example mining and multi-part combination into automatic safety helmet wearing detection," *IEEE Access*, vol. 9, pp. 139536-139543, 2020.

[5] F. Zhang, H. Fleyeh, X. Wang, and M. Lu, "Construction site accident analysis using text mining and natural language processing techniques," Automation in Construction, vol. 99, pp. 238-248, 2019.

[6] S. Tan, G. Lu, Z. Jiang, and L. Huang, "Improved YOLOv5 network model and application in safety helmet detection," *2021 IEEE International Conference on Intelligence and Safety for Robotics (ISR),* pp. 330-333, 2021.

[7] S. Huang, J. Huang, and Y. Kong, "Attention Guided YOLOv3 for Wearing Safety Helmet Detection," *2020 the 6th International Conference on Communication and Information Processing*, pp. 65-69, 2020.

[8] X. Long, W. Cui, and Z. Zheng, "Safety helmet wearing detection based on deep learning," *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 2495-2499, 2019.

[9] D.M. Wu, H. Wang, and J. Li, "Safety Helmet Detection and Identification Based on Improved Faster RCNN," *Information Technology & Informatization*, vol. 1, pp. 17-20, 2020.

[10] N. Li, X. Lv, S. Xu, Y. Wang, and Y. Gu, "Incorporate online hard example mining and multi-part combination into automatic safety helmet wearing detection," *IEEE Access*, vol. 9, pp. 139536-139543, 2020.

[11] Y. Zhang, K.P. Wu, K. Gao, and X. Yang, "Research on detection method of helmet based on improved YOLOV3," *Computer Simulation*, 2021.

[12] L.Q. Yang, L.Q. Cai, and S. Gu, "Behavior detection of helmet wearing based on machine learning method," *Chinese Safety Production Science and Technology,* 2019.

[13] X.F. Xu, W.F. Zhao, H.Q. Zou, L. Zhang, and Z.Y. Pan, "Detection Algorithm of Safety Helmet Wear Based on MobileNet-SSD," *Computer Engineering*, vol. 47, no. 10, pp. 298-305,313, 2021.

[14] B.Y. Deng, X.C. Lei, and M. Ye, "Safety helmet detection method based on YOLO v4," *2020 16th International Conference on Computational Intelligence and Security*, pp. 155-158, 2020.

[15] M. Sadiq, S. Masood, and O. Pal, "FD-YOLOv5: A Fuzzy Image Enhancement Based Robust Object Detection Model for Safety Helmet Detection," *International Journal of Fuzzy Systems*, pp. 1-17, 2022.

[16] Q.P. Duan, P. Kuang, L. Fan, and M.Y. He, "Method of Safety Helmet Wearing Detection based on Key-Point Estimation without Anchor," *2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing*, pp. 93-96, 2020.

[17] F.B. Zhou, H.L. Zhao, and Z. Nie, "Safety helmet detection based on YOLOv5," *2021 IEEE International Conference on Power Electronics, Computer Applications*, pp. 6-11, 2021.

[18] Y.T. Zhang, M.F. Zhang, X.Q. Shi, X.K. Chen, Y. Ren, and R. Liu, "Research on lightweight detection method of personnel helmet wearing," *Journal of Safety and Environment*, 2022.

[19] Christine Dewi, and Rung-Ching Chen, "Deep Learning for Advanced Similar Musical Instrument Detection and Recognition," IAENG International Journal of Computer Science, vol. 49, no.3, pp880-891, 2022.