International Journal of Advanced Engineering Research and Science (IJAERS)
Peer-Reviewed Journal
ISSN: 2349-6495(P) | 2456-1908(O)
Vol-11, Issue-8; Aug, 2024
Journal Home Page Available: https://ijaers.com/
Article DOI: https://dx.doi.org/10.22161/ijaers.118.2

Fire detection algorithm based on the fusion of YOLOv8 and Deformable Conv DCN

Lin Po Shang[1], Yan Zuo Chang[2,*], Yi Chen[1], Yong Shan Ou[3]

[1]Department of Energy and Power Engineering, Process Equipment and Control Engineering major, Guangdong University of Petrochemical Technology, China
[2]Department of Energy and Power Engineering, Guangdong University of Petrochemical Technology, China
[3]Department of Electronic Information Engineering, Electronic Information Science and Technology major, Guangdong University of Petrochemical Technology, China
*Corresponding author: ros1517877219@outlook.com

Received: 29 Jun 2024,
Receive in revised form: 31 Jul 2024,
Accepted: 08 Aug 2024,
Available online: 15 Aug 2024

***Keywords*— Fire identification, Deep learning, YOLOv8, Deformable Conv**

***Abstract*—** *With the progress of fire monitoring and Coping technique, image recognition based on deep learning has shown great potential in the field of fire detection. Aiming at the accuracy and efficiency problems existing in the existing object detection algorithms, this study proposed an improved YOLOv8 algorithm to improve the real-time recognition capability in the fire scene. Through experimental verification on standard fire data sets, this study evaluated the detection performance of the improved YOLOV8 algorithm fused with Deformable Conv. The experimental results show that the improved YOLOv8 has improved the fire identification accuracy compared with the traditional version, and has certain potential for practical application in fire monitoring system.*

I. INTRODUCTION

In modern society, fire occurs everywhere and crises abound, and almost any environment with heat sources or combustible materials has hidden safety hazards, which seriously affect the living environment of human beings. Human contact with fire may lead to burns of different degrees, ranging from minor surface burns to serious deep tissue damage. Severe burn require long-term medical treatment and may result in disability or life threatening [1]. In addition, fire also has irreversible damage to buildings, production and environment. Fire can cause internal and external burning of buildings, resulting in partial or complete damage to building structures [2]. High temperature can also make building materials lose strength such as steel and concrete, and even lead to structural collapse, posing a direct threat to the durability and safety of buildings [3]. In large-scale production environments such as warehouse workshops, chemical plants and assembly lines, fire spreads quickly and may cause violent explosions when it comes into contact with flammable and explosive substances, which damages important equipment in the production environment and threatens personnel's life and property safety [4].

With the rapid development of modernization, warning of fire is particularly important in industrial fields and large public places. At present, warning of fire is mainly based on sensors, such as smoke sensors and temperature sensors. Those mainly detect smoke and temperature [5], and the detection range is limited while the effect is difficult to guarantee. When the smaller flame appears, it is generally not immediately detected and reflected in time, which is often the cause of fire in the production workshop or chemical plant. The current development of AI technology is rapidly advancing, especially with breakthrough achievements in the field of mechanical learning. Computer vision technology is an application field of mechanical learning. In recent years, computer vision technology has had a mature development and has been applied in all walks of life, especially in the field of target detection and recognition. With the improvement of

technology, its accuracy and detection speed continue to improve. Instead of relying on sensors and manual inspection to prevent disasters, we proposed an improved visual recognition algorithm based on convolutional neural network (CNN) for fire detection in this paper.

At present, there are many kinds of object detection technologies. Compared with traditional object detection algorithms, the recognition algorithm based on convolutional neural network (CNN) has more advantages, which can not only better characterize complex features, but also greatly improve the accuracy and real-time performance. It is widely used in vehicle detection and recognition [6], automatic driving [7], attitude detection, fault diagnosis of manufacturing equipment and robot fields [8, 9]. Currently, the two main object detection algorithms based on convolutional neural network (CNN) are One-stage object detection algorithm and two-stage object detection algorithm. The detection process of two-stage detection algorithm is divided into two parts, as shown in Figure 1: (a) two-stage object detection algorithm and (b) single-stage object detection algorithm.

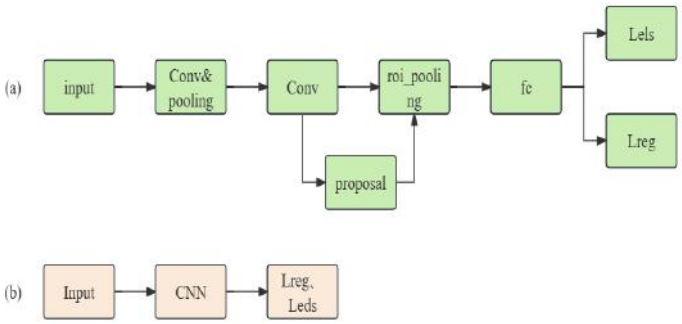


Fig. 1: Comparison between two-stage object detection algorithm and single-stage structure

First of all, we generate the candidate region, and then determine whether there is an object to be detected in the candidate region, and the category of the object. At present, the most popular Two-stage algorithm is R-CNN series, including R-CNN, SPP-Net, Faster-CNN, etc. [10]. Compared with Two-Stage algorithm, One-Stage algorithm carries out single regression and reduces Anchor steps. The category and location information is given directly through the Backbone network, so it is faster and easier to learn the generalization characteristics of the target, reducing the generation of false postive. The SSD and YOLO series are the mainstream of One-Stage. In 2016, Joseph Redmon et al. proposed a one-stage object detection network [11], which has a very fast detection speed and can process 45 frames of pictures per second. The author named it You Only Look Once, and the first generation of YOLO was born. With the subsequent in-depth research, YOLOv2 and YOLOv3 have been come out one after another, which have strong migration ability and are widely used in various fields. In terms of risk detection, YOLO has also played its role.Ying Liu developed a risk identification system in oil field production environment based on YOLOv3 [12]. Zizqiang Li et al. applied YOLOv5 in the intelligent detection of unsafe conditions on the construction site [13], Ruiguo Wei used YOLOv5 to develop a fire image recognition method [14], and Qingxu Li designed the cabin fire detection system with an improved YOLOv5 algorithm [15]. This study focused on the 8th generation YOLO algorithm (YOLOv8) is proposed an improved vision detection algorithm based on convolutional neural network (CNN) fused with Deformable Conv for fire identification.

II. INTRODUCTION OF YOLOv8 NETWORK ARCHITECTURE

YOLOv8 is a new generation of target detection algorithm launched by ultralytics. It is an innovation and improvement made by ultralytics team on the basis of YOLOv5 previously launched [16]. On the basis of inheriting the real-time detection characteristics of YOLO series, the model structure has been comprehensively optimized and improved to elevate the performance of the model. Its network structure diagram is shown in Figure 2.

Its general architecture is basically composed of Input, Backbone, Neck and Head. Compared with YOLOv5, the first layer convolution size of Backbone is changed from the original 6×6 to 3×3, and C3 module is changed into Cf2 module [17], which uses the relevant structure in YOLOv7. This part is replaced by multiple cross-layer connections instead of one convolutional module and one skip module, and the split part is added, which is based on

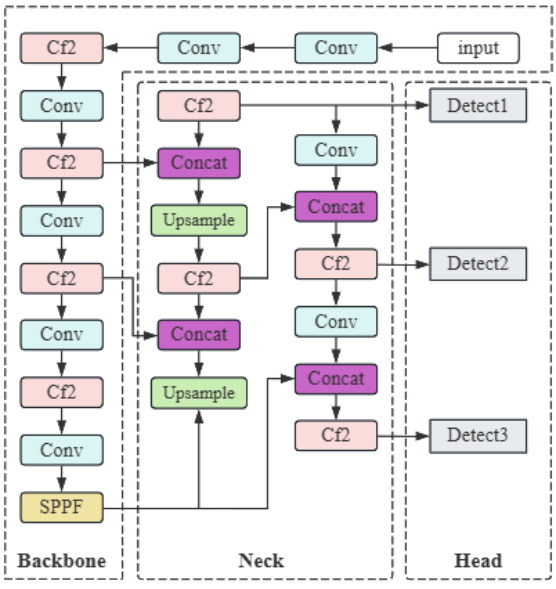


Fig. 2: YOLOv8 network structure diagram

the part of YOLOv7.The replacement helps the network to understand the characteristics of different scales, improving its accuracy and robustness. Neck, as an intermediate layer, fuses with the transmitted features of backbone, which is conducive to improving performance. While YOLOv8 replaces C3 module in Neck with Cf2 module, the original 1×1 convolution is removed [18].

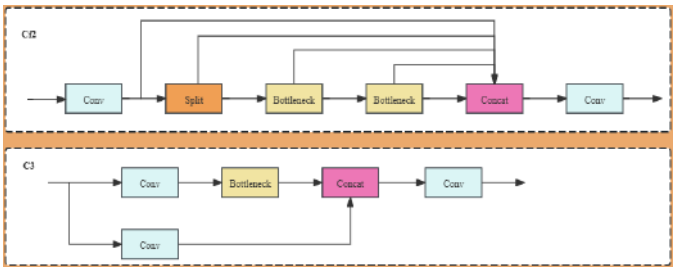


Fig. 3: Comparison between C3 module and C2f module

The Head is mainly responsible for generating the final target detection results from the features extracted by the Backbone network, helping to identify and locate various targets in the input image. Compared with Yolov5, the Coupled Head was changed to a similar to the Decoupled-Head structure. [19] As shown in Figure 3, the regression branch and prediction branch were separated.

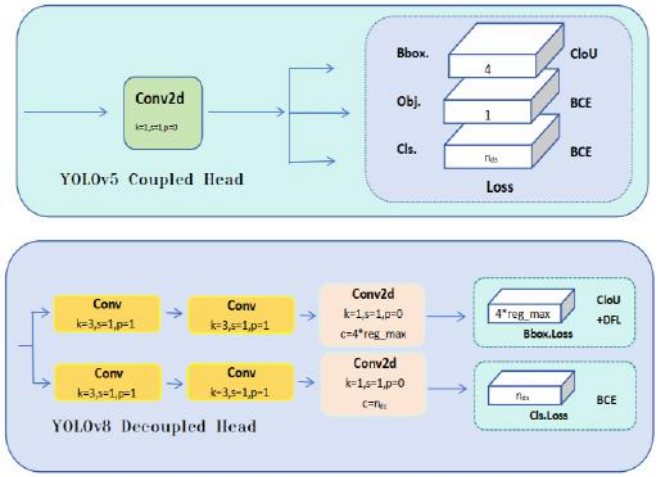


Fig. 4: Comparison between YOLOv5 and YOLOv8 Head

The original detection head was divided into multiple subtasks, each of which was responsible for detecting targets of related undetermined categories.This approach can improve the scalability and flexibility of the model, which is conducive to processing This approach can improve the scalability and flexibility of the model, which is conducive to processing multiple types of detection otargets, and ptimize each related subtask to improve the accuracy of detection. This improvement can also make the model better adapt to different application scenarios and target categories, and cope with the structure of different detection layers by adjusting the number and size of convolutional layer and fully connected layer.

III. ALGORITHM IMPROVEMENT AND OPTIMIZATION

3.1 Principle analysis of Convolution operation

In the convolutional neural network, a high value or a low value is needed to distinguish the feature region from other regions in order to extract features from the input image data. The process of generating values is called Convolution operation, and the core of the Convolution operation is a small matrix called Convolutional Kernel. This Convolutional Kernel slides over the entire input data to generate an output feature map by weighted summing local regions of the input data. At each step, the Convolutional Kernel is multiplied with a small window of the input data by elements part by part, and then the results are added to obtain a unit value of the output feature map. The sliding window usually moves on the input data by a certain translation unit, and moves a fixed unit distance each time [20]. This process allows the Convolutional Kernel to efficiently detect local features in the input data, such as edges, textures, or other higher-level features. For each input data, the Convolution operation will generate an output feature map, which has the same dimension as the input data in space, but the depth will be different. These feature maps contain different feature information of the input data, which can be used as input for the subsequent neural network layer.

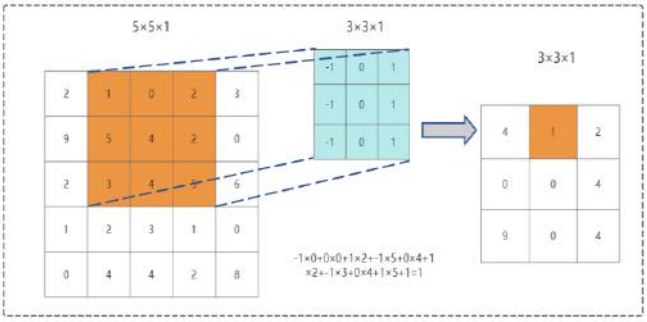


Fig. 5: General convolution process

Figure 5 shows the process of common convolution calculation on the input feature graph. The size of Convolutional Kernel is 3×3, and the size of the input feature graph is 5×5. The weight of the Convolutional Kernel is multiplied by the corresponding position element of the input feature graph, and the value of the output feature graph element is obtained by summating. The rest

of the input matrix is calculated step by step to get the output feature map.

Input any point P_0 on the feature graph, and the convolution process can be expressed as:

$$y(p_0)\sum_{p_n\in R}\omega(p_n)\times x(p_0+p_n) \quad (1)$$

Where Pn represents the offset of each point in the Convolutional Kernel relative to the center point, which can be expressed by the following formula (3×3 Convolutional Kernel for example) :

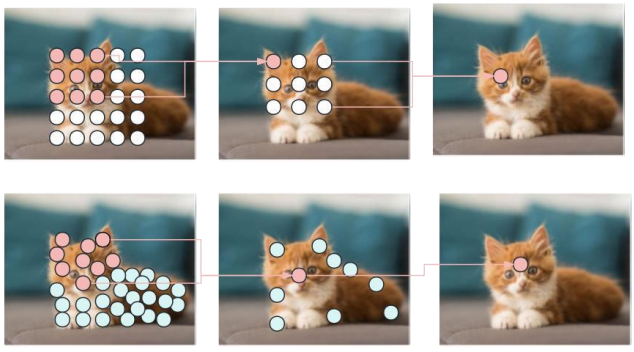
$$R=\{(-1,-1),(-1,0),\ldots,(0,0),\ldots,(1,0),(1,1)\} \quad (2)$$

We can take the center of the matrix as the origin, and each grid uses a coordinate system to represent the position of that part. w(pn) represents the weight of the corresponding position of the convolution check, and this value is generally a constant, x(p0+pn) represents the element value at p0+pn position on the input feature map, and y(p0) represents the element value at p0 position on the output feature map, which is obtained by the relevant convolution operation of the Convolutional Kernel.

3.2 Deformable Conv ideas

The Convolutional Kernel of conventional convolution has a fixed size and shape. For more complex image data, it is difficult for conventional Convolutional Kernel to extract more accurate features, especially the edge features of graphics. If Convolutional Kernel can be

adjusted according to the characteristics of input data during convolution operation, the characteristics of data can be better obtained and the detection accuracy can be improved. This is also the core of Deformable Conv.



(top) Standard convolution (bottom) Deformable Conv

Fig. 6: Convolution example of standard convolution and Deformable Conv

As can be seen from the upper and lower comparison shown in Figure 6, for this cat image, the features extracted from the standard convolution cured structure also have a certain curing effect. The collating features extracted from the detected objects in the image may not be obvious, and its rectangular plane arrangement cannot be well correlated with sampling. The sampling position of deforming convolution is more in line with the shape characteristics of the object itself [21], and it has certain elasticity and can better adapt to the boundary conditions of the image. Therefore, after incorporating deforming convolution for convolution operations, the output feature points correspond to the overall features of the data, which can improve the acquisition of useful information of the data and improve the accuracy of the model compared with conventional convolution.

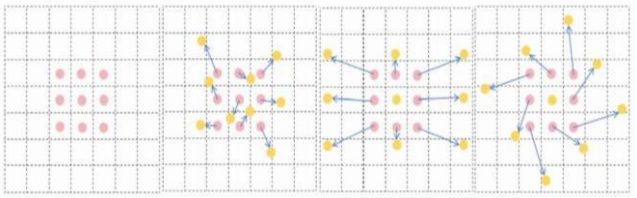


Fig.7: Different adoption points of Deformable Conv

As shown in Figure 7, (a) is the sampling method of a common 3x3 Convolutional Kernel, and (b) is the change of sampling point after Deformable Conv plus offset, where (c) and (d) are special forms of Deformable Conv. The light pink dots represent the 9 sampling points of the conventional square Convolutional Kernel, the light blue arrows represent the offset vector, and the yellow ones represent the offset sampling points. The Deformable Conv introduces an offset for each point based on formula 1, which is generated by the input feature map with another convolution, usually a decimal.

$$y(p_0)\sum_{p_n\in R}\omega(p_n)\times x(p_0+p_n+\Delta p_n) \quad (3)$$

In the Deformable Conv, the regular grid R increases the offset Δpn. Since the position after adding the offset is generally a decimal, it does not correspond to the actual pixel points on the input feature map.

Figure 8 shows the Deformable Conv diagram. It can be seen that offsets are generated by using an additional convolution, which is not a convolution operation. In the figure 8, N is used to represent the size of the Convolutional Kernel region, for example, the size of a convolution kernel is 3×3, N=9. The green process in the figure is the process of convolutional learning offset, where the channel size of the offset field is 2N and represents that the Convolutional Kernel learns the offset in the x direction and the y direction respectively.

On the input feature map, the convolution sampling region corresponding to common convolution operations is a square with the size of the Convolutional Kernel, while

the convolution sampling region corresponding to Deformable Conv is some points represented by blue boxes. This is the difference between Deformable Conv and ordinary convolution.

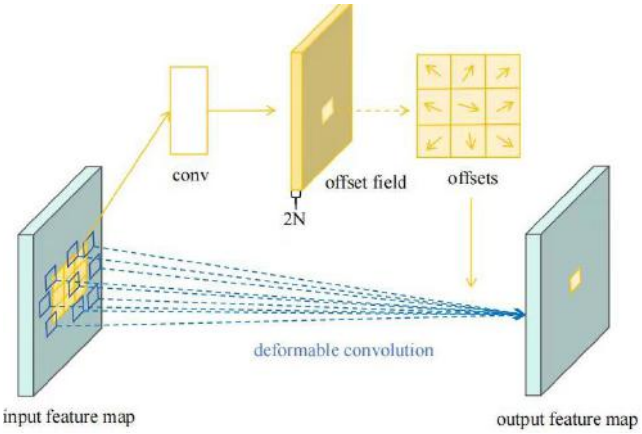


Fig.8: Schematic diagram of Deformable Conv

In terms of the specific details of the corresponding schematic diagram, the size of the convolution sampling area corresponding to a point on an output feature map and the input feature map is K×K. According to the operation of Deformable Conv, each convolution sampling point in the K×K region must learn a deviation offset. The offset is expressed in coordinates, so an output must learn 2×K×K parameters.If an output size is H×W, so a total of 2×K×K×H×W parameters must be learned. That is, the offset field(N=K×K) in the preceding figure has the dimensions B×2×K×K×H×W, where B represents batch_size.

IV. EXPERIMENT AND RESULT ANALYSIS

4.1 Experimental environment

The development language of the training model is Python, the compiler is Python3.9.13, the compilation tool is pycharm2023, the deep learning framework Pytorch1.13.0, and CUDA version 11.16. The experiment was carried out in Windows11 operating system. CPU was AMD Ryzen 7 5800H with Radeon Graphics 3.20GHz, and GPU was NVIDIARTX3060Ti with 16G video memory.

Related images were selected from the network in the experiment, and the data set contained more than 2000 pieces of two types of data. The selected images were annotated by labelimg tool in pycharm, and the annotated picture information was automatically converted into YOLO_txt format. Then the original image and YOLO_txt format images are divided into training set, verification set and prediction set proportionally.

During the training, Mosaic data was used to enhance the first 95% epoch. During the torch training, the parameter value was Epoch 100, the compressed size of the input image Imageinput 640×640, and the batch size of the training Batchsize 16.

4.2 Evaluation Indicators

In order to verify the improved performance of YOLOV8 model, this experiment verified the selection accuracy P(Precision), Recall R(Recall), Average Precision AP(Average Precision), and average precision mAP(mean Average Precision). Precision-confidence curve and loss line curve were used as evaluation indexes. The calculation of P, R, AP and mAP is as follows: The evaluation indexes in this experiment include accuracy rate, recall rate and average accuracy, and the formula is shown as follows.

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

$$\text{Recall} = \frac{TP}{TP+FF} \tag{5}$$

$$AP = \int_0^1 P(R)\,dR \tag{6}$$

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

TP (True Positive) indicates the number of Positive classes predicted as positive classes, that is, the number of positive classes predicted correctly. FP (False Positive) indicates the number of negative classes predicted as positive classes, that is, the number of negative class samples predicted incorrectly. The accuracy rate P represents the proportion of the predicted samples in the positive samples, and the actual proportion of the true positive samples. The recall rate R represents the proportion of truly positive samples in all predicted samples. While AP is the average accuracy of each category, mAP represents the average AP of multiple categories, and m is the number of categories. In this detection task, the types of input data are fire and spark, so m=2.

Compared with the experimental results shown in Figure 9-10, it can be seen that the precision, recall and mAP after the improvement have been improved to varying degrees, indicating that the overall accuracy of the model after incorporating the demorphable convolution has been improved to some extent.

The horizontal coordinate of the graph named Precision-confidence curve represents the detector's confidence, and the vertical coordinate represents the accuracy (or recall rate). The shape and position of the

curve reflect the performance of the detector at different confidence levels. It provides useful information for evaluating the performance of the detector at different confidence levels. It can be seen from this comparison experiment figure that the accuracy of the detector is improved after the use of Deformable Conv. The detector can maintain a low false positive rate while maintaining a high recall rate, and the recognition accuracy of the target is high.

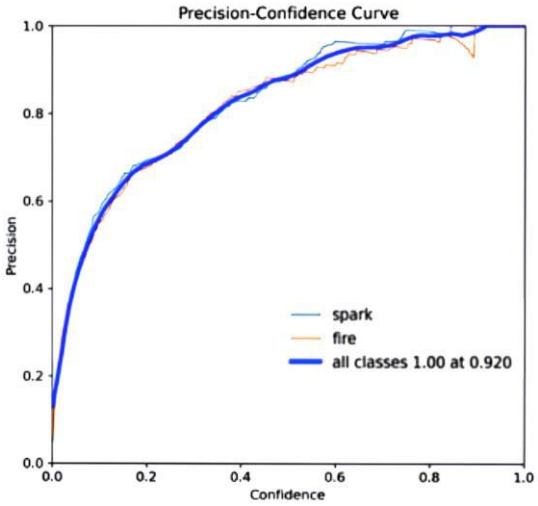


Fig.9: Feedback curve of the improved Precision-confidence curve

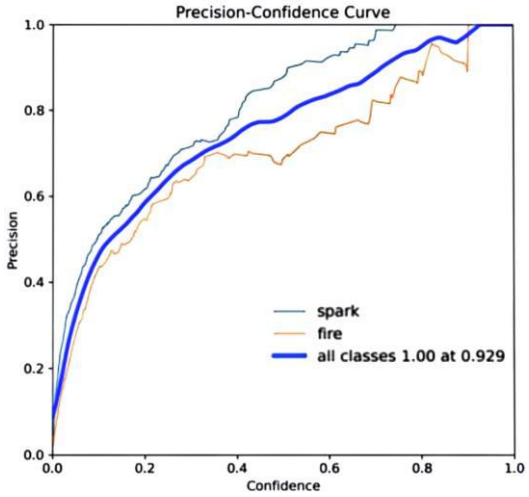


Fig.10: Precision-confidence curve Feedback curve before improvement

Loss function plays an important role in target detection tasks, which is used to measure the difference between the predicted value and the real value of the model. The resulting loss function includes box_loss (positioning loss), dfl_loss (feature point loss) and cls_loss (classification loss), as shown in Figure 11-12. Box_loss is used to calculate the difference between the predicted bounding box and the real bounding box, and IOU (Intersection overUnion) is used as a metric to measure the overlap between the two bounding boxes. Box_loss

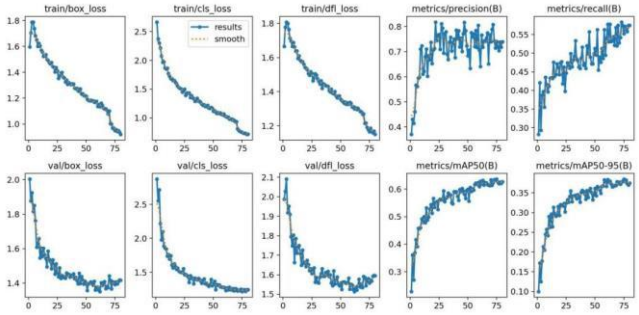


Fig.11: Visual analysis diagram of experimental results before improvement

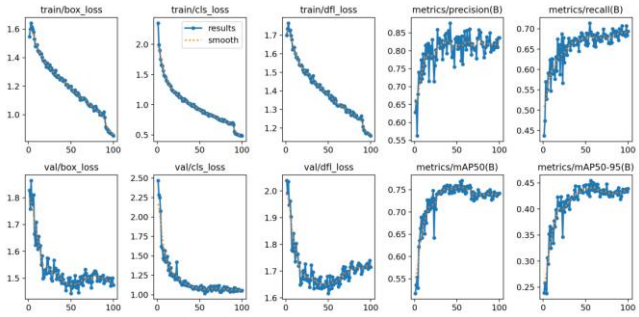


Fig.12: Visual analysis diagram of the improved experiment

calculates the IOU between the predicted box and the real bounding box. By minimizing box_loss, the model can learn a more accurate boundary box position, and the smaller the value, the more accurate the positioning. It can be clearly seen from image comparison that the improved box_loss mean is significantly smaller. Cls_loss is used to calculate the difference between the predicted class and the real class, and Cross Entropy Loss is used to measure the classification accuracy. The loss value of the class is calculated by comparing the difference between the predicted class distribution and the real class label. By minimizing cls_loss, the model can learn more accurate category classification, and the smaller the value, the more accurate the classification. Through comparison, it can be seen that the mean value of improved cls_loss is significantly smaller than that of pre-improved cls_loss, and df1_loss (feature point loss) is a custom loss function introduced in YOLOv8. YOLOv8 uses feature points to predict the direction and Angle information of the object, which is used to calculate the difference between the predicted feature points and the real feature points. By minimizing df1_loss, the model can learn more accurate

direction and Angle information of the object, and the smaller the value, the more accurate the feature acquisition. Through image comparison, it can be seen that the mean value of dfl_loss in the improved model is smaller than that before the improvement.

The detected training image is shown in Figure 13.

Fig.13: Training image detected

V. CONCLUSION

All data sets in this experiment were obtained legally from the Internet in terms of model environment construction, relevant data sets construction, model training, and evaluation results. After replacing standard convolution with Deformable Conv, the precision of YOLOv8 detection algorithm is 0.835, which is improved to a certain extent compared with the pre-improvement accuracy of 0.768. box_loss (positioning loss), dfl_loss (feature point loss) and cls_loss (classification loss) as a whole decreased by about 0.2, 0.1 and 0.3, indicating that the improved algorithm has improved the ability to locate objects in images, classify different kinds of images, and acquire features of objects.

In addition to the flame type data, this experiment also selected the complex data with relatively fine features - spark, which is still difficult to detect in the scope of fire prevention. However, from the experimental results, the improved algorithm also has a good normalization ability for spark. Compared with the original algorithm, the improved YOLOv8 algorithm can be applied to various scenes, such as shopping malls, chemical plants and other production environments, and has good accuracy. It also has good adaptability to some image data with not obvious feature points and enhances the algorithm's ability to extract target features.

ACKNOWLEDGEMENTS

This work was supported by the Research Funding of GDUPT, Horizon Patrol - Machine Inspection Expert Driven by YOLOV8 Engine (No. 71013513124014).

REFERENCES

[1] Xiaohuan Zhan, Junqin Gao&Jile Fu.(2018). Application of brace in the treatment of deep hand burns. Journal of Practical Hand Surgery, 32(1), 86-89.

[2] Kevin Luo&Iebin Lian.(2024).Building a Vision Transformer-Based Damage Severity Classifier with Ground-Level Imagery of Homes Affected by California Wildfires.FIRE, 7(4).

[3] Yinghao Shi . (2024). High temperature test and residual compressive strength study of C80 high performance concrete. Shanxi Architecture (15), 100-103.

[4] Conglin Zhou . (2023). Application of intelligent fire extinguishing technology in chemical fire rescue. Chemical Engineering Management (35), 85-87.

[5] Gefang Lei, Youlong Wu.(2024). Design of warehouse fire warning system based on Internet of Things technology. Internet of Things Technology (07), 35-38.

[6] Jin Yuchen&Rona (2019). Improved YOLOv2 vehicle real-time detection algorithm combining multi-scale features. Computer Engineering and Design, 40 (5), 1457-1463

[7] XinYu Zhang, ZhenHong Zou, ZhiWei Li, HuaPing Liu&Jun Li. (2020). Deep multi-modal fusion technology for automatic driving target detection. Journal of Intelligent Systems (04), 758-771.

[8] Yufeng Ding, Zhengru Liu&Lei Hu.(2021). Research on PCB test bench for temperature controller based on machine vision and FMEA. Experimental Technology and Management (11), 115-120.

[9] Huifeng Wang, Hao Du , Rong Gao, Yaxiong Tong Lu Peng, Yu Zhao&He Huang .(2024). A climbing robot system for detecting the apparent diseases of pier and column structures based on visual scanning. China Journal of Highway and Transportation (02), 40-52.

[10] Wang Yanya. (2022). Review of object detection algorithms based on Two-Stage. Journal of Hebei Academy of Sciences (02), 14-22.

[11] Joseph Redmon, Santosh Divvala, Ross GirshickAli Farhadi.(2016).You Only Look Once: Unified, Real-Time Object Detection.arxiv.

[12] Huan Liu.(2023). Implementation of Hazardous Operation Identification System for Oil Field Production monitoring Master (Dissertation, Xi 'an Shiyou University). master

[13] ZiQiang Li, Lei Ren, Li Liu&ZuoHua Miao.(2023). Intelligent detection of unsafe state in construction site based on YOLOv5 algorithm. Civil Engineering Information Technology (03), 20-26.

[14] Ruiguo Wei.(2022). Transfer Learning-based Fire image Recognition Method Research Master's degree thesis, Xi 'an Technological University. master

[15] QingXu Li.(2021). Master's Degree in Cabin Fire Detection System Design based on Improved YOLO Algorithm (Dissertation, Huazhong University of Science and Technology). master
[16] Zhihong Zhao&Ziye Hao.(2024). Improved YOLOv8 small target detection method for aerial photography: CRP-YOLO. Computer Engineering and Applications (13), 209-218.
[17] Wang, Xueqiu, Gao, Huanbing, Jia, ZemengLi, Zijian.(2023).BL-YOLOv8: An Improved Road Defect Detection Model Based on YOLOv8.SENSORS, 23(20).
[18] Jie Li, Yang Liu, Liang Liu, Bengan Su, Jialong Wei, Guangda Zhou ... & Zhen Zhao . Remote Sensing small target detection based on cross-stage two-branch feature aggregation. Journal of System Simulation 1-16.
[19] Yong Liu , Fengshun LV , Xuecun LI &Shiyu Zuo. Object detection algorithm of remote sensing image based on YOLOv8-LGA. Optoelectronics · Laser 1-12.
[20] Liwei Hu, Zhi Hou, XueTing Zhao, Bing Liu , Chen Chen , Yu He&Ruijie Zhang. Research on improving highway traffic risk prediction model based on traffic accident text mining. Journal of Southwest Jiaotong University 1-10.
[21] CunYi Liao, Yi Zheng, WeiJin Liu, Huan Yu&Shouyin Liu.(2024). Multi-task decoupling and fusion algorithm for automatic driving environment awareness. Computer Applications (02), 424-431.