Mathematical Modeling and Algorithm Application
ISSN: 3006-0842 | Vol. 2, No. 3, 2024

MUTI-YOLOV8: YOLOV8-based target detection algorithm for stacked objects

Yuchen Xu [1], Yun Xu [1], Qi Li [2],Liang Gao [1], Lang Wang [1]

[1]School of Mechanical Engineering, Sichuan University of Science & Engineering, Yibin Sichuan 644000, China
[2]Sichuan Changzheng Machine Tool Group Co.,Ltd., Zigong Sichuan 643000,China

Abstract: In order to solve the problems of low accuracy, missed detection and false detection of stacked objects in the existing object detection, an improved algorithm model based on YOLOV8 was proposed. The model introduces Deformable Convolutiona Networks. The Shueffle Attention mechanism is added to reduce the complexity of the model and improve the ability to express features. The Dysample upsampling module is introduced to replace the original Upsample module, which reduces the requirement of computing resources. The improved model is trained on the stacked object dataset, and compared with the original model, the accuracy of the improved network model is improved by 1.7%, and the accuracy of mAP50-95 is increased by 1%. It provides a theoretical reference for the study of object detection of stacked objects.

Keywords: YOLOV8 ; Deep learning ; Object detection ; Attention mechanisms.

1. Introduction

With the rapid development of intelligent manufacturing technology and robotics industry, there is an increasing demand for intelligent robots in the industrial field. Intelligent robots realize target grasping through visual perception, target detection and localization and robotic arm grasping technologies. In structured environments, vision-based technology-guided robotic arms can replace manual labor to a certain extent to complete repetitive and mechanical work. However, in some unstructured industrial environments, such as complex stacking, random placement and other scenarios, target recognition still exists, with slow detection speed, low accuracy and robustness [1].

In recent years, with the rise of computer vision technology and the wave of deep learning, more and more researchers have started to use deep learning for computer vision research. Deep learning methods are relative to some traditional detection methods, the focus of the work is mainly concentrated on the design and training of the detection network, which greatly reduces the dependence on domain knowledge due to the direct learning of relevant feature representations from the original image. In the field of target detection, they can be broadly categorized into two-stage algorithms represented by R-CNN (Region Convolutional Neural Networks) and single-stage algorithms represented by SSD, YOLO family of algorithms [2].

Girshick[3] et al. first proposed the application of deep learning algorithms to object detection and proposed R-CNN detection network.He[4] et al. designed SPPNet network, which uses spatial pyramid pooling to enable the network to handle images of arbitrary sizes, which significantly improves the speed of detection but the detection accuracy is not very high. During the same period related researchers proposed some single-stage network applications for object detection.Redmon [5] et al. proposed YOLOv1 released in 2015, which uses a single convolutional neural network to achieve accurate object detection by partitioning the input image into SxS lattices, each predicting B bounding boxes and C category probabilities.Liu et al[6] proposed the SSD network architecture, SSD designs a detection network with feature pyramid structure, which ensures similar detection accuracy as Faster R-CNN and improves faster detection speed than YOLOv1. YOLOv2, on the other hand, uses a deeper network structure and introduces more techniques, such as improved bounding-box prediction, multi-scale training, and suppression of overlap, which reduces the algorithm complexity and improves the detection accuracy.YOLOv3, on the other hand, further improves the detection accuracy by introducing multi-scale prediction, improved bounding box prediction, suppression of overlap, and improved loss function. In addition, YOLOv3 [7] introduced a new detection framework which can better handle complex scenes. Since then, the YOLO series has gradually become a popular method in single-stage detection networks, among which YOLOv5 further improves the detection speed and accuracy by improving the backbone network and loss function, etc.

In this paper, the MUTI-YOLOV8 algorithm is proposed for the recognition problem of stacked objects. (1) Shueffle Attention attention mechanism is added to the backbone network, which reduces the model complexity and improves the expression of features. (2) Deformable Convolutiona Networks Deformable Convolutiona Networks are added to strengthen the feature learning of defective targets. (3) Dysample upsampling module is introduced in the neck instead of the original Upsample upsampling module, which reduces the demand of computational resources and effectively improves the defect detection accuracy.

2. YOLOv8 algorithm

YOLOv8 is the latest version of the YOLO (You Only Look Once) series of object detection and image segmentation models developed by Ulitralytics. It is a real-time target detection algorithm based on a one-stage algorithmic model that incorporates many SOTA techniques and is highly scalable, in the same vein as the YOLOv3 algorithm and the YOLOv5 algorithm [8].

YOLOv8 combines context and features, uses a C2f structure with richer gradient flow at the backbone (backbone network) and Neck side, and improves the overall

performance of the model by setting different number of channels for different scales of the model; and uses a decoupled header structure to split the detection and classification, and handles the visual task independently.

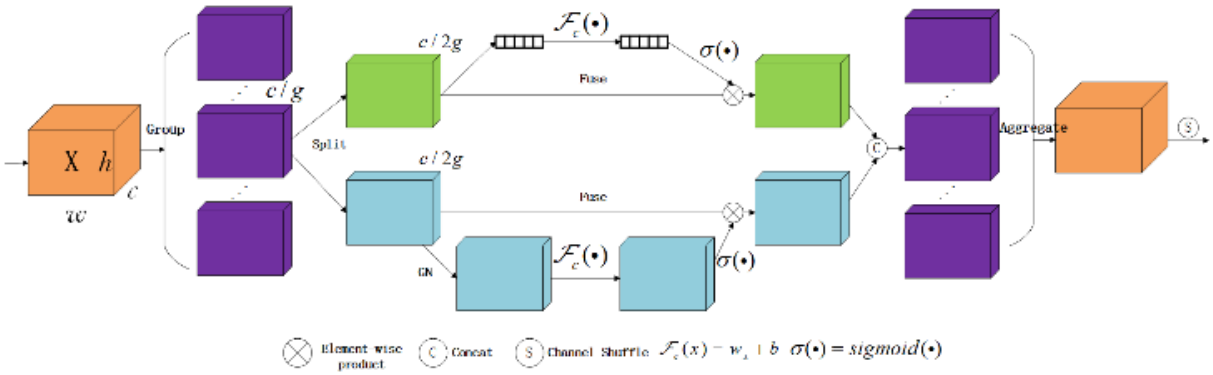


Figure 1 ShuffleAttention Network Architecture

3. MUTI-YOLOv8 Algorithm

3.1. Shuffle Attention

ShuffleAttention [9] proposed by Zhang et al. is a very lightweight hybrid attention structure, as shown in Fig. 1, which divides the channel dimension into multiple sub-features and processes them in parallel. For each sub-feature, SA utilizes hybrid units to describe the feature dependencies in both spatial and channel dimensions. Finally, all sub-features are aggregated, and the "channel mixing" operator is used to realize the communication between different sub-features.

The SA first divides X into G groups along the channel dimension, i.e.,$X = [X_1, \cdots, X_G], X_k \in \mathbb{R}^{C/G \times H \times W}$.where each sub-feature X_k gradually captures a particular kind of semantic information during the training process, and a corresponding importance coefficient is generated for each sub-feature.At the beginning of each attentional unit, the input of X_k is divided into two branches of spatial attention and channel attention， where $X_{k1}, X_{k2} \in \mathbb{R}^{C/2G \times H \times W}$.One branch is used to generate channel attention maps through interrelationships between channels, while the other branch is used to generate spatial attention maps by utilizing spatial relationships between features so that the model can focus on the target more effectively.

In channel attention, the statistics of the channel approach are generated by embedding the global average pool using global information， which is $s \in \mathbb{R}^{C/2G \times 1 \times 1}$， and is computed by contracting X_{K1} by the spatial dimension $H \times K$:

$$s = \mathcal{F}_{gp}(X_{k1}) = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} X_{k1}(i, j). \quad (1)$$

Through the sigmoid mechanism, a compact feature is created to achieve accuracy and adaptability.

$$X'_{k1} = \sigma(\mathcal{F}_c(s)) \cdot X_{k1} = \sigma(W_1 s + b_1) \cdot X_{k1} \quad (2)$$

Where $W_1 \in \mathbb{R}^{C/2G \times 1 \times 1}$， $b_1 \in \mathbb{R}^{C/2G \times 1 \times 1}$.

In contrast, in spatial attention branching, Group Norm is first performed on the input feature map [25].The representation of the input $\hat{X}_{k2}$ is then augmented by the transformation $\mathcal{F}_c(\cdot)$. The final output formula for spatial attention is as follows

$$X'_{k2} = \sigma(W_2 \cdot GN(W_{K2}) + b_2) \cdot W_{k2} \quad (3)$$

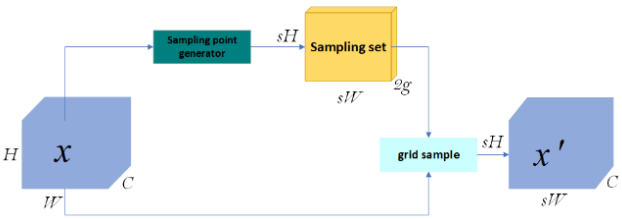
Finally, the results of the two attention spans are concatenated together to make the final output the same size, where$X'_k = [X'_{k1}, X'_{k2}] \in \mathbb{R}^{C/G \times H \times W}$.

3.2. Dysample

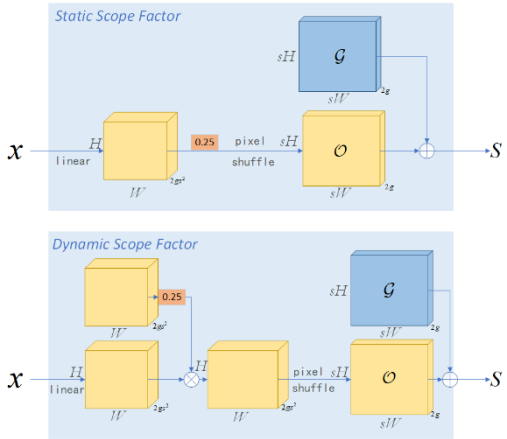
Feature up-sampling is an important component in dense predictive models used to gradually recover feature resolution.

Kernel-based dynamic upsampling such as CARRAFE, FADE, and SAPA have achieved highly desirable performance gains, but their workload has increased dramatically due to the excessive time-consuming dynamic convolution and the additional sub-network used to generate the dynamic kernel. This problem can be well addressed by utilizing the Dysample upsampling module proposed by Liu [10].

Considering the heavy workload associated with dynamic convolution, the Dysample network structure is shown in Fig. 2.



(a)Sampling based dynamic upsampling



(b)Sampling point generator in Dysample

Figure 2 Dysample network architecture

Bypassing the kernel-based paradigm and redirecting to the essence of upsampling, i.e., sampling points. It is assumed that the input features are interpolated to continuous features

using bilinear interpolation, and content-aware sampling points are generated to resample continuous images.

From this perspective, a simple design is first performed as shown in Fig. so that the point-by-point offsets are generated by a linear projection and used to resample the point values using the grid sample function in PyTorch with the following formula.

$$\mathcal{X}' = grid\_samle(\mathcal{X}, S). \quad (4)$$

And then, given an upsampling scale factor s and a feature map $\mathcal{X}$ of size $C \times H \times W$, whose input and output channel numbers are C and $2s^2$, is used to generate offsets of size $2s^2 \times H \times W$.The sample set $\mathcal{S}$, which is the sum of the offset $\mathcal{O}$ and the original sample $\mathcal{G}$, is then obtained by Pixel-shuffling [11].

$$\mathcal{O} = linear(\mathcal{X}) \quad (5)$$

$$\mathcal{S} = \mathcal{G} + \mathcal{O} \quad (6)$$

Compared to other upsamplers, DySample does not require high-resolution bootstrap inputs, does not require additional CUDA packages, and reduces parameters, FLOPsD, GPU & latency. Also, DySample outperforms other samplers in semantic segmentation, target detection, instance segmentation, panorama segmentation and monocular depth estimation.

3.3. DCNv2

In unstructured environments, the types and morphologies of stacked targets are mostly different. The traditional convolution used in YOLOv8 uses a fixed-size convolution kernel for feature learning, which makes it difficult to effectively obtain the boundary information of the targets. For this reason, this paper utilizes Deformable Convolution Network DCN (Deformable Convolution) proposed by DAI et al [12] and others as shown in Fig. 3, which can solve this problem well.

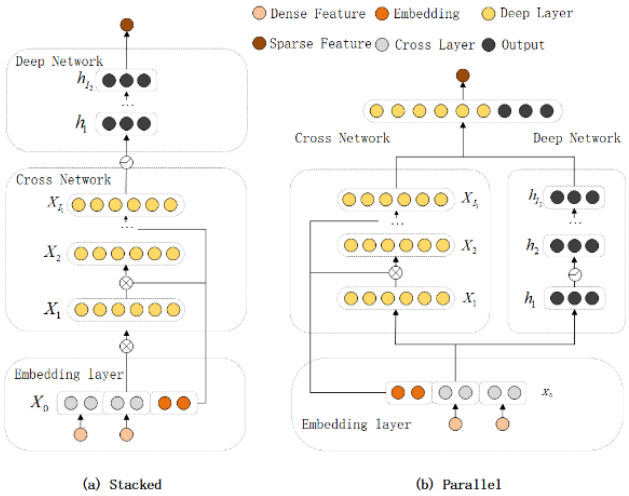


Figure 3 DCN network architecture

In DCNv1, the offset is obtained by learning the sampling points and adding them to the sampling positions during the standard convolution operation, which is able to automatically adjust the size of the convolution kernel according to different detection targets, with the following equation.

$$y(p) = \sum_{k=1}^{K} \omega_k \cdot x(p + p_k + \Delta p_k). \quad (7)$$

where x and y are the input and output feature maps, respectively, K is the total number of sampling points, k is the location of the sampling point, p is the current position of the output feature map, ω_k and p_k are the weight and offset of the k position, respectively, Δp_k is the learnable offset of the k position.

And in DCNv2, in order to further enhance the control of deformable convolution over the spatial support region, a modulation mechanism is introduced by adding weights for each sampling point and setting the weights of uninterested locations to 0. The modulated deformable convolution can be expressed as follows:

$$y(p) = \sum_{k=1}^{K} \omega_k \cdot x(p + p_k + \Delta p_k)\Delta m_k. \quad (8)$$

Among them, Δm_k is the modulation scalar at the k position in the range [0,1].

The introduction of deformable convolution into the feature layer of YOLOv8 is able to increase the range of its receptive field, better adapt to the geometrical deformation of the target object, and improve the convolutional learning ability by adapting to the size of the defective target, which achieves the purpose of optimizing the algorithm.

3.4. MUTI-YOLOv8 Infrastructure

Based on the YOLOv8-Seg model, use DCNv2 deformable convolution to integrate into the original C2f, replacing Conv2d in C2f with DCNv2 to form the C2f_DCNv2 module, which improves the sensory field, enhances the feature extraction capability of the model, and more accurately fits targets of different directions and sizes.

Use DySample instead of the original upsample structure in YOLOv8 to enhance the target sensing ability and better aggregate contextual information in the sensing field. The ShuffleAttention attention mechanism is inserted between the fourth layer C2f_DCNv2 and SPPF to reduce the model complexity and achieve higher accuracy.The MUTI-YOLOv8 network structure is shown in Figure 4.

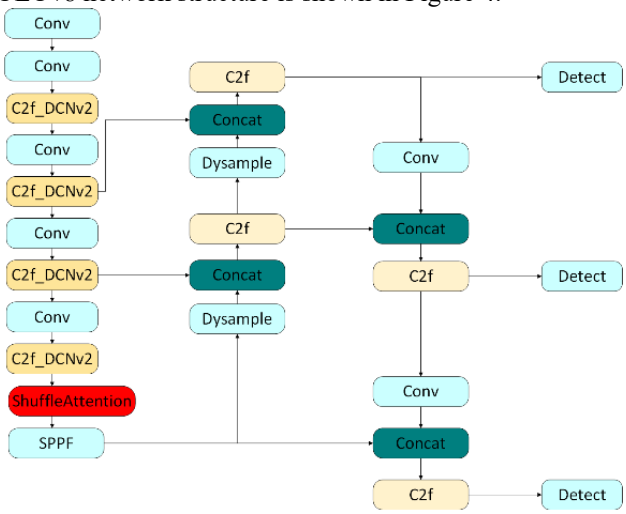


Figure 4 network infrastructure

4. Results and Analysis

4.1. Experimental environment

The hardware environment of this study is, under Windows 11 operating system, with a CPU of i5-12490F, a graphics card of NVIDIA GeForce RTX3060 (12G video memory), and a CUDA version of 1.13.1+cu117.

Table 1 experimental environment

environment	Configuration
system	Windows11
IDE	Pycharm
CPU	i5-12490F
GPU	NVIDIA GeForce RTX3060
CUDA+cudnn	1.13.1+cu117

The training parameters for this experiment are configured as, input image resized to 640x640, training batch of 16, number of rounds of 300, initial learning rate of 0.01, weight decay of 0.0005 and learning momentum of 0.937.

4.2. Experimental dataset

In this experiment, Astra Pro Plus depth camera is utilized to simulate the construction of unstructured stacked environments using a variety of industrial environments and common objects in life, a total of 1,000 photos are collected, and the final dataset is 5,000 photos by data enhancement. In order to make the model training more effective, according to the ratio of 7:2:1, 3500 sheets are used as the training set, 1000 sheets are used as the testing set, and 500 sheets are used as the validation set.

4.3. Evaluation indicators

In this paper, the experiments choose the accuracy rate (Precision) P: as the ratio of the number of correctly detected objects to the total number of detected objects, the recall rate (Recall) B: the ratio between the number of correctly recognized objects and the total number of actual objects. Average precision (average precision) AP: the average detection precision of the single-category model, which establishes a coordinate system with the recall rate as the horizontal axis and the accuracy rate as the vertical axis, and the area enclosed by the PR curve formed within a certain threshold.

Mean average precision (mean average precision) mAP: indicates the average precision of all categories, the higher the precision of the target model, the larger its value, the formula is as follows.

$$Recall = \frac{TP}{TP+TN} \tag{9}$$

$$AP = \int_0^1 P(r)dr \tag{10}$$

$$mAP = \frac{1}{N}\sum_{i=1}^{N} AP_i \tag{11}$$

Wherein, TP denotes the number of correctly detected defective targets, FP denotes the number of incorrectly detected defective targets, and TN denotes the number of missed defective targets.

4.4. Attentional Experiments

In order to verify the superiority of introducing the ShuffleAttention attention mechanism and to compare the effects of different attention mechanisms on target detection performance. In this paper, GAM,CA,ECA, and CBAM attention mechanisms are selected for comparison experiments under YOLOv8s model.

Table 2 Attention Contrast Experiments

model	P%	R%	f_{mAP}%
-	93.7	94.2	96.9
CBAM	93.4	95.1	96.6
GAM	96.1	96.2	97.8
CA	94.3	93.5	96.8
ECA	95.4	95.1	97.3
SA	**96.3**	**96.3**	**98.2**

From the data in Table 2, among all the attention mechanisms, only the ShuffleAttention attention mechanism improves the training effect, and the mAP reaches 98.2. It can be seen that, compared with other attention mechanisms, the effect improvement of SA for target detection is more superior, which enhances the feature extraction ability of the model.

4.5. Ablation Experiment

In order to verify the performance advantage of the new module and improved model introduced in this paper in multi-object detection, the YOLOv8n model is used as the base network, c2f is replaced by C2F_DCNv2 in the original network backbone, SA attention mechanism is introduced, and UpSample up-sampling is replaced by DySample in the head. yolov8n_1

Indicates the introduction of the SA attention mechanism. Model YOLOv8n_2 represents the replacement of the DySample upsampling module. Model YOLOv8n_3 denotes replacement of DCNv2 deformable convolution module. Model YOLOv8n_4 denotes the replacement of DySample upsampling on top of YOLOv8n_1. YOLOv8n_5 denotes the replacement of DCNv2 deformable convolution on top of YOLOv8n_2. YOLOv8N_6 denotes the addition of SA attention mechanism on top of YOLOv8n_2. Finally, all the improved modules are introduced by the MUTI-YOLOv8 model. The obtained ablation experimental results are shown in Table 3. From the experimental results in Table 3, it can be seen that the accuracy of the model is improved by 0.9% and the mAP is improved by 0.5% when only the ShuffleAttention attention mechanism is added to the backbone compared to YOLOv8n. This implies that the SA attention mechanism improves the information capture of the feature map by mixing the attention structures, thus improving the feature extraction capability of the model. In the base model of YOLOv8n, replacing the C2F convolution module with the DCNv2 variable convolution module improves the precision by 2.3%, recall by 0.6%, and mAP by 0.5%. This shows that when replacing C2F with DCNv2, it plays an active role in target detection and improves the detection performance of the model. When both the SA attention mechanism and the replacement of the DCNv2 variable convolution module were added, the model precision increased by 3.8%, recall increased by 2%, and mAP increased by 1.3%, which further demonstrates the gainful effect of the addition of the attention mechanism and the replacement of the variable convolution module on the network model enhancement. By replacing the UpSample upsampling with the DySample upsampling module, the precision is improved by 2.1% compared to the original network. By also adding the SA attention mechanism, the DCNv2 module with DySample upsampling, the improved overall network precision is increased by 4.6%, recall by 2.3%, and mAP by 1.7%. The ablation experiments validate that the proposed improved model shows considerable improvement in three important metrics: precision, recall and mAP of the network model.

Table 3 ablation experiment

model	SA	DySample	DCNv2	P/%	R/%	mAP50/%
YOLOv8n	-	-	-	93.3	94.2	96.9
YOLOv8n_1	✓	-	-	94.2	94.2	97.4
YOLOv8n_2	-	✓	-	95.4	93.0	97.3
YOLOv8n_3	-	-	✓	95.6	94.8	97.4
YOLOv8n_4	✓	✓	-	95.5	93.4	97.5
YOLOv8n_5	-	✓	✓	94.8	96.1	97.7
YOLOv8n_6	✓	-	✓	97.1	96.2	98.2
MUTI-YOLOv8	✓	✓	✓	**97.9**	**96.5**	**98.6**

4.6. Comparison

In order to further verify the superiority of the improved model, it is compared and experimented with the algorithms of the current mainstream models: YOLOv3, YOLOv5, YOLOv6, YOLOv8,SSD, and the original network YOLOv8, etc., and the experimental results are shown in Table 4. As can be seen from Table 4, compared with the original network YOLOv8n, this paper's algorithm, although the number of parameters has increased, in terms of precision, recall and mAP, it improves by 4.6%, 2.3% and 1.7%, respectively, which proves the effectiveness of this paper's algorithm. As for the detection precision, the precision of the MUTI-YOLOv8 algorithm proposed in this paper is higher than that of Faster-RCNN, YOLOv3, YOLOv5n, YOLOv6n, YOLOv8n, which is improved by,, 0.3%, 2.3%, 3.2% and 4.6%, respectively. Among them, YOLOv3 has the lowest improvement effect, only 0.3%, but its number of parameters is 8.5 times that of this paper's algorithm.Although the Faster-RCNN and YOLOv3 algorithms are similar to this paper's algorithms in terms of their detection accuracy, the size of their number of parameters is large, which is not conducive to the model's deployment and use. In summary, compared with the current mainstream model, the improved algorithm in this paper has better detection performance.

Table 4 Comparison of different algorithms

algorithm	P/%	R/%	mAP50/%	Params/10^6
Fsater-RCNN	61.9	95.0	96.6	137.1
YOLOv3	97.6	96.1	98.6	103.7
YOLOv5n	95.6	94.8	95.0	9.1
YOLOv6n	94.7	93.8	97.0	16.3
YOLOv8n	93.3	94.2	96.9	11.1
MUTI-YOLOv8	97.9	96.5	98.6	12.2

4.7. Comparison

In order to present a more intuitive comparison between the detection effect of the improved model and the basic YOLOv8n model, the target pictures in the stacked environment are selected for the comparison of the detection effect, and the results are shown in Fig. 5.

By observing the detection effect of the two models, it can be found that the confidence effect of MUTI-YOLOv8 is better than that of YOLOv8n in the task of target detection under the stacked environment.The results show that the MUTI-YOLOv8 model proposed in this paper possesses a more effective detection in the practical application, shows higher robustness and accuracy, and has a higher application value.

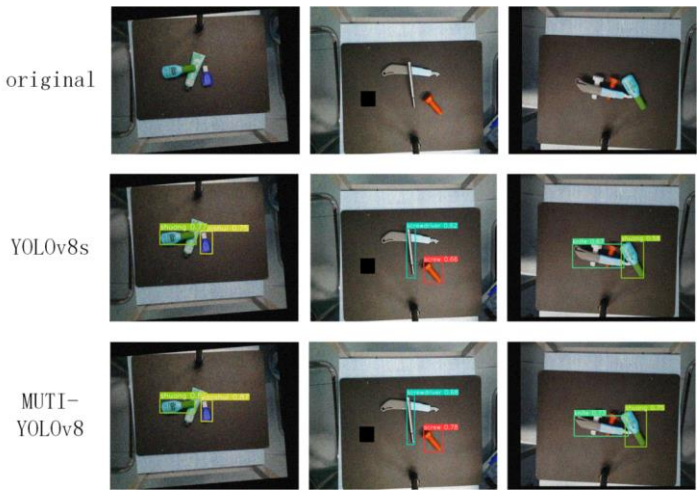


Figure 5 Comparison of detection effect

5. Summarize

To address the problems of target detection algorithms in complex unstructured environments in practical detection. In this paper, the MUTI-YOLOv8 target detection algorithm is proposed. On the basis of YOLOv8s model, we improve it by introducing SA attention mechanism in the backbone part to improve the model accuracy. The C2F convolution module is replaced with DCNv2 variable convolution module to enhance the feature extraction capability. The UpSample module is replaced by the DySample module in the head to reduce the computational cost and storage overhead of the model. This makes the algorithm more adaptable to target detection in unstructured environments. However, due to its relatively simple dataset, target detection in more complex environments requires further research.

References

[1] Luo Xiongwei. Zhu Zhengtao." A cascade network-based segmentation algorithm for scattered stacked objects." Foreign Electronic Measurement Technology 43. 02 (2024): 66-73.doi: 10.19652/j.cnki.femt.2305474.

[2] Gui Jiayang,Wang Shunji,Zhou Zhengkang,et al. Foreign object detection algorithm in tunnel based on improved YOLOv8n

[J/OL]. Computer Application1-7[2024-08-08]. http://kns.cnki.net/ kcms/ detail/51.1307.TP.20240424.1642.010.html.

[3]Girshick R, Donahue J, Darrell T, et al.Rich feature hierarchies for accurate object detection and semantic segmentation [C]// Proceedings of the IEEE conference on computer vision andpattern recognition.2014:580-587.

[4]He K, Zhang X, Ren S, et al.Spatial pyramid pooling in deep convolutional networks for visual recognition[J]. IEEE transactions on pattern analysis and machine intelligence, 2015, 37(9):1904-1916.

[5] Redmon J, Divvala S, Girshick R, et al. You only look once: Unified,real-timeobjectdetection[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[6] Liu W, Anguelov D, Erhan D, et al. Ssd: Single shot multibox detector[C]/ComputerVision-ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14,2016, Proceedings, Part I 14. Springer International Publishing, 2016: 21-37.

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

[8] Han Q. Research on improved YOLOv8 algorithm for small target detection[D]. Changchun:Jilin University,2023

[9] Zhang, Qing-Long and Yubin Yang. "SA-Net: Shuffle Attention for Deep Convolutional Neural Networks." ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (2021): 2235-2239.

[10] Liu, Wenze et al. "Learning to Upsample by Learning to Sample." 2023 IEEE/CVF International Conference on Computer Vision (ICCV) (2023): 6004-6014.

[11]Wenzhe Shi, Jose Caballero, Ferenc Husz´ar, Johannes Totz,Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. InProceedings of IEEE Conference on Computer Vision Pattern Recognition (CVPR), pages 1874–1883, 2016.

[12] Wang, Ruoxi et al. "DCN V2: Improved Deep & Cross Network and Practical Lessons for Web-scale Learning to Rank Systems." Proceedings of the Web Conference 2021 (2020): n. pag.