IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 42, No. 2, April, 2025, pp. 985-994
Journal homepage: http://iieta.org/journals/ts

YOLOv8 Algorithm Improvement and Its Application in Small Target Detection

Lu Gao[1], Junwei Zhao[2*]

[1] College of Physical Science and Technology, Bohai University, Jinzhou 121000, China
[2] College of Computer Science and Technology, North China Institute of Science & Technology, Langfang 065201, China

Corresponding Author Email: shower@yeah.net

https://doi.org/10.18280/ts.420232

Received: 9 October 2024
Revised: 13 February 2025
Accepted: 7 March 2025
Available online: 30 April 2025

Keywords:
YOLOv8, small target detection (STD), improved algorithm, DCNv4, deformable convolution, attention mechanism

ABSTRACT

In the trend of the continuous development of AI, the application range of small target detection (STD) is very wide. Improving the accuracy of small target detection is the focus of research, which has positive significance for the improvement of computer vision technology and multiple application scenarios. Based on the overall structure of YOLOv8 network, this paper introduces the deformable convolution and attention mechanism of DCNv4 to improve it. DCNv4 enhances the ability of the network to capture spatial structure, especially for objects of different scales or positions, so that the kernel can sample from any position in the input feature map, which can improve the performance of small target detection to a certain extent. The attention mechanism improves the ability of the network to focus on the key information in the detection task, and improves the efficiency and accuracy of the network detection. The experimental results show that the improved YOLOv8 algorithm significantly improves the detection performance of small targets, and achieves a good balance between detection accuracy and computational efficiency.

1. INTRODUCTION

Small target detection (STD) is the detection of small objects, because small objects are often far away from normal size objects, it is difficult to learn feature representations from their abstract structures, and challenges such as complex background and image quality problems also hinder deep learning-based detectors [1]. The wide range of practical application requirements of STD attracts more and more researchers to engage in STD to overcome the difficulties in STD. Typical application scenarios include many aspects: First, it is applied to aerial image analysis of unmanned aerial vehicles (UAVs) [2]. UAVs equipped with embedded devices can analyze captured data to complete corresponding tasks. In agriculture, drones can capture the color of crops, judge the maturity of fruits and guide agricultural production.

Second, applied to automatic driving [3], automatic driving requires continuous environmental perception to obtain the distribution of obstacles, so as to achieve safe driving. Through object detection technology, the class, location and size of surrounding objects can be predicted, so that a safe route can be formulated in advance to avoid the obstacles ahead.

Third, as applied to medical image analysis [4], the lesion sites on medical images are often small and difficult to be observed by naked eye. The use of STD technology can help find small lesions, so that doctors can detect problems in the early stages of the disease, reducing the risk of disease progression. In order to improve the accuracy and efficiency of STD, this paper studies the STD model improved by YOLOv8 algorithm.

2. RELATED WORKS

Detecting small targets using computer vision is a challenging task. Traditional STD algorithms rely on manual feature construction and need to design complex feature representations. Haar feature is a local feature descriptor based on the difference of image gray values, and it is also a rectangular feature. The rectangular feature value is the sum of the gray values of the white rectangle minus the gray values of the black rectangle. The rectangular feature is similar to some simple graphic structures [5]. HOG features extract image texture and shape features using local gradient direction [6]. SVM classifiers can use extracted HOG features for target detection. Because of the sensitivity of human visual system to color, color feature has always been the main factor considered in salience target detection algorithms. The classification model of target and background is established based on SVM algorithm, and the background model is iteratively optimized in combination with the information entropy evaluation feature map, and the significant target is obtained [7]. Traditional object detection algorithms use manual features with limited feature description ability, which makes it difficult to capture some high-level semantic information and sensitive to changes in illumination, scale and perspective [8]. The challenge of STD is the change of attitude, Angle and scale, complex background and so on, which makes the traditional target detection algorithm cannot be applied to the STD task.

Compared with traditional algorithms, deep learning can not only learn shallow features such as color and texture, and remove noise through appropriate training [9], but also learn

deep semantic information. As a downstream task of image classification, researchers began to consider using CNN in object detection. Faster R-CNN implements the first end-to-end real-time object detection algorithm. In the detection process, candidate bounding boxes of images are selected, and accurate object region and category labels are further generated by using Fast R-CNN [10]. YOLOv8 is a new model specially designed for scenes involving a large number of small objects, which enhances multipath fusion to integrate features at different levels, preserve shallower details and improve detection accuracy of small objects [11]. Compared with other mainstream target detection algorithms, YOLOv8 has excellent fast and accurate performance [12]. The Neck part of YOLOv8 includes an SPPF module [13], which changes the original extraction module into a series mode to greatly reduce the amount of computation. The head adopts the PAN structure based on FPN design idea, and integrates the upper layer features into the shallow network features. The detection head uses the head structure to calculate the losses respectively, so that the model attention to the key points and improves the operation efficiency.

New deep learning methods have been widely applied in the field of small object detection: Firstly, the multi-scale feature fusion method [14]. Small targets occupy a small proportion in the image, and detailed information is prone to be lost during the downsampling process of the network. To solve this problem, multi-scale feature fusion methods have been widely applied. By constructing a convolutional neural network model with convolutional layers and pooling layers of different scales, feature information of different scales is extracted, and then these feature maps are fused by operations such as upsampling and downsampling. This can integrate the rich detailed information at the bottom level and the semantic information at the top level, thereby better detecting small targets. Like the feature pyramid network, by means of top-down and horizontal connections, features at different levels are fused, and good results have been achieved in the task of small object detection. Secondly, data augmentation technology [15-21]. Due to the relatively small amount of training sample data for small targets, data augmentation techniques are crucial for improving the detection performance of small targets. In addition to traditional operations such as random flipping, rotation and zooming, some new data augmentation methods are constantly emerging. For example, the CutMix technology increases the diversity of data by fusing parts of different images, enabling the model to learn more features of small targets in different scenarios. The MixUp technique linearly combines the features of different images to generate new training samples, which helps improve the generalization ability of the model. Thirdly, model lightweighting and optimization. On some resource-constrained devices, such as mobile terminals and embedded systems, it is necessary to lightweight and optimize the small object detection model. On the one hand, by designing lightweight network structures, such as MobileNet, ShuffleNet, etc. [22, 23], the parameters and computational load of the model can be reduced. On the other hand, by adopting model compression techniques such as pruning and quantization, redundant parameters in the model are removed, and the parameters are represented as low-precision data types. Without affecting the performance of the model, the storage space and inference time of the model are reduced.

There have been many successful application cases of deep learning methods in the field of small object detection. The following are a few of them. The first one is the aerial photography scene captured by drones. In UAV aerial images, the targets present the characteristics of small targets due to their long distance and low resolution, and there are dense occlusions and complex background interferences [24, 25]. YOLOv5s+CBAM, on the VisDrone2021 dataset, enhanced small object feature extraction by introducing CBAM, mAP@0.5 reached 47.6%, which is 6.2% higher than the original YOLOv5. The FCOS-RetinaNet hybrid model, which integrates FCOS and FPN, achieves an AP_small of 38.9% in the intensive small vehicle detection task, outperforming YOLOv5's 34.5%. The image resolution of UAVs is high, and the real-time requirements are strict. The FPS of the existing models is generally lower than 30, and further lightweighting is needed. The second is the security monitoring scenario. Security monitoring needs to detect tiny targets in long-distance cameras and adapt to low-light conditions at night [26]. RetinaNet+CBAM, in the DOTA dataset, by improving the feature pyramid structure, the AP_small was increased to 41.3% (originally 36.1% for RetinaNet). NanoDet-M, an ultra-lightweight model (with 1.8M parameters), on the HRSC2016 ship dataset, has a 39.2% mAP@0.5 FPS of 152 and is suitable for edge device deployment, but AP_small is about 5% lower than that of the heavy model. The false detection rate increases in low-light scenarios (background false detection rate >20%), and it is necessary to combine infrared image fusion or GAN to enhance the data. Thirdly, medical imaging testing [27]. The precise detection of small targets in medical images directly affects the diagnostic accuracy, but data annotation is scarce and there is a lot of noise. U-Net++withCBAM, in the LiTS dataset, by introducing the dense attention module, the detection recalls rate of liver tumors increased to 82.3%, which was 11.5% higher than that of the traditional U-NET. TransUNet, an encoder-decoder structure based on Transformer, has a Dice coefficient of 0.78 for small lesions (volume <5cm³) in the BraTS brain tumor segmentation task, which is superior to the CNN-based model (0.72). The cost of medical data annotation is high. Semi-supervised learning can alleviate the shortage of data, but the generalization ability of the model still needs to be verified. Fourth, the autonomous driving scenario. Autonomous driving needs to detect small targets such as traffic lights, pedestrians and cones in real time, and also needs to handle dynamic blur and extreme weather. YOLOv8s+SWIN, combined with SwinTransformer 's visual backbone, achieved 68.4% in traffic sign detection mAP@0.5 in the KITTI dataset, a 4.1% improvement over YOLOv8. DETR-DC5, a detection model based on Transformer, has an AP_small of 43.7% in the Cityscapes dataset, but the inference speed is only 25FPS, and the inference engine needs to be optimized. The missed detection rate of small targets is high in dynamic scenes, and the robustness needs to be improved by combining time series information.

3. YOLOV8 ALGORITHM STRUCTURE

YOLOv8 algorithm uses some effective strategies of YOLO series algorithms, and the backbone network architecture is very clear. The original C3 modules are replaced by the new C2f modules, which greatly improves the model performance. The YOLOv8 algorithm is shown in Figure 1.

3.1 Backbone network

In YOLOv8, the C2f replaces the C3 in YOLOv5, and the ELAN in C2f replaces the CSP in C3. ELAN module by controlling the shortest path, the feature map of different layers is effectively fused, and more complex feature fusion is achieved through group convolution and scrambled operation, which eases the problem of gradient explosion, maintains the high performance of the network, and realizes the effective learning and convergence of the deeper network. C2f uses more Bottleneck structures, reduces or increases dimensions, and adopts more residual structures for feature extraction, which reduces the complexity of the model and improves the training stability to a certain extent. It is especially effective when dealing with large-scale data sets, making the network can be deployed efficiently on mobile devices [28].

3.2 Detect head

The detection head of YOLOv5 calculates a diverse set of anchor frame sizes based on the training set statistics, and generates potential target bounding boxes from the anchor frame as a starting point. There are some drawbacks to this approach. The selection and setting of anchor frames are quite complicated, and the rationalization of threshold selection is also challenging. YOLOv8 does away with the anchor frame mechanism, bypassing the problems associated with anchor frame design and its associated calculations. The detection head of YOLOv8 is shown in Figure 2.

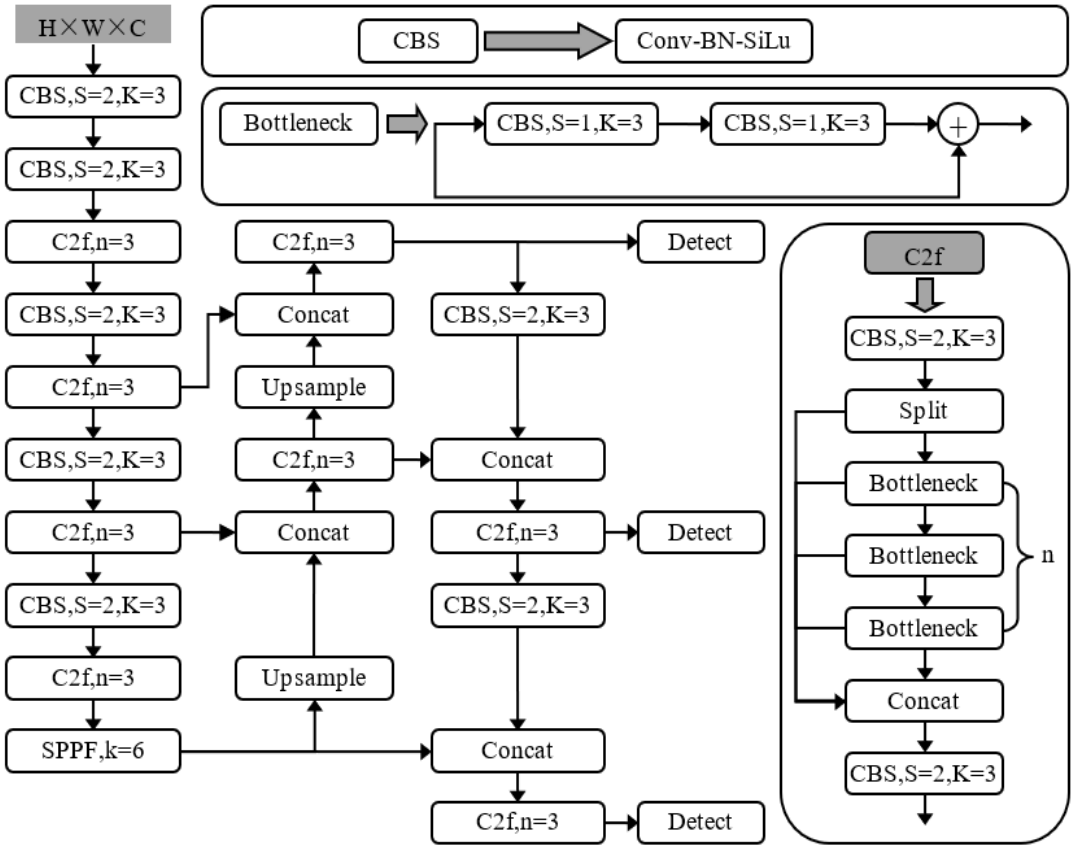

Figure 1. YOLOv8 algorithm structure

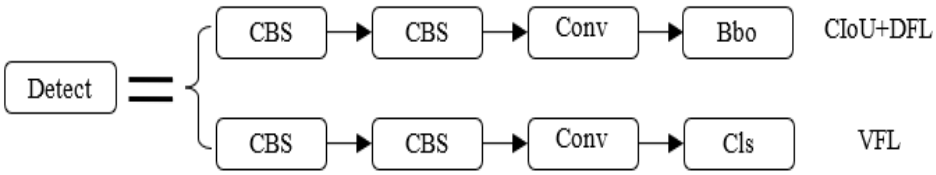

Figure 2. Detect head of YOLOv8

YOLOv8 adopts a completely decoupled detection head design strategy, in which the original bounding box position information and the probability distribution of the category of the object in the box are explicitly split into two independent processing branches. One branch is responsible for predicting accurate boundary box position information, and the other branch is responsible for identifying specific categories of objects in the box, which helps the model to focus more on their respective tasks during the training process, resulting in faster convergence speed and improved detection accuracy to a certain extent.

3.3 Loss function

YOLOv8 Loss includes classification and detection frame loss. Classification loss adopts Varifocal Loss (VFL) function, and detection frame loss adopts Distribution Focal Loss (DFL) function. VFL developed on the basis of Focal Loss (FL) [29].

(1) FL Loss Function

FL is based on Cross Entropy (CE). Although FL is derived from object detection, it can be applied to many other scenarios. The formula of CE is expressed as:

$$CE(p,y)=\begin{cases}-\log(p), & y=1\\ -\log(1-p), & otherwise\end{cases} \tag{1}$$

where, y represents the sample, p represents the probability, in order to facilitate the presentation, redefine p_t:

$$p_t=\begin{cases}p, & y=1\\ 1-p, & otherwise\end{cases} \tag{2}$$

Thus, the CE function can be expressed as:

$$CE(p,y)=CE(p_t)=-\log(p_t) \tag{3}$$

The model should put more effort into learning difficult samples and less effort into learning easy samples. CE functions treat hard samples and easy samples equally. This leads to the expression FL:

$$FL(p_t)=-(1-p_t)^{\gamma}\log(p_t) \tag{4}$$

In the above equation, γ is the regulator and its value is between 0 and 5. When the value is 0, it is equivalent to the CE function. The larger the value, the model more attention to the difficulty of the sample. The above formula only reflects difficult samples and easy samples, and does not distinguish between positive samples and negative samples. Thus, the complete FL formula is derived:

$$FL(p_t)=-\alpha_t(1-p_t)^{\gamma}\log(p_t) \tag{5}$$

It is insufficient to consider α_t solely as a mechanism for adjusting the weights; instead, it should be examined in conjunction with $-\alpha_t(1-p_t)^{\gamma}$ to fully understand its impact.

(2) VFL Loss Function

VFL developed on the basis of FL to ensure the accuracy of the detection box and the accuracy of category prediction.

p represents the classification score predicted by the model, and q represents the target IoU value. For a positive sample, q is the IoU value between the generated bounding box and the real bounding box, expressed as:

$$\mathrm{VFL}(p,q)=-q(q\log(p)+(1-q)\log(1-p)) \tag{6}$$

q of all classes is set to 0, and the formula is expressed as:

$$VFL(p,q)=-\alpha p^{\gamma}\log(1-p) \tag{7}$$

Here, α and γ are hyperparameters that adjust the weight to solve the problem of sample imbalance. Rather than simply treating all samples, this approach highlights the influence of positive samples by assigning different weights according to the IoU value.

(3) DFL Loss Function

In the process of image detection, when multiple objects block or overlap each other, the annotation frame and detection frame may not accurately reflect the true semantic of the image. Therefore, it is necessary to adopt a more accurate bounding box representation method, and the general probability representation formula is:

$$\hat{y}=\int_{-\infty}^{+\infty}P(x)x\,dx=\int_{y_0}^{y_n}P(x)x\,dx \tag{8}$$

Because the network cannot directly generate a continuous probability distribution, this distribution is approximated by a series of discrete probability points:

$$\hat{y}=\sum_{i=0}^{n}P(y_i)y_i \tag{9}$$

In the training process, if the above formula is directly used for training, the probability representation space will be too large, which makes the network difficult to optimize and converge. Therefore, using Distribution Focal Loss:

$$\begin{aligned}&\mathrm{DFL}(S_i,S_{i+1})=\\ &-\left((y_{i+1}-y)\log(S_i)+(y-y_i)\log(S_{i+1})\right)\end{aligned} \tag{10}$$

where, S represents the probability output value. When y_{i+1} is very close to y and the probability output of S is large, the DFL is small and the distribution tends to be closer to the center of the annotation. Therefore, DFL helps the network focus on the predicted values near the target more quickly and speeds up model convergence.

4. YOLOV8 ALGORITHM IMPROVED

The improvement of YOLOv8 algorithm is the introduction of DCNv4 deformable convolution and attention mechanism.

4.1 DCNv4 deformable convolution

Because of the fixed sampling grid of traditional CNN, it is often difficult to capture the feature information adequately for small targets. Especially for high-resolution images, the features of small targets are sparser. The introduction of Deformable Conv Nets (DCN) enhances the ability of the network to capture spatial structure, especially for objects of different scales or positions, so that the kernel can sample from any position in the input feature map, which can improve the performance of STD to a certain extent.

DCNv1 changes the sampling position of the standard convolution by predicting the offset. The calculation formula:

$$y(p)=\sum_{k=1}^{K}w_k\cdot x(p+p_k+\Delta p_k) \tag{11}$$

where, K represents the number of sampling points, w_k represents the weight of the k-th sampling point, p_k represents the predetermined bias amount, $x(p)$ represents the p feature of the input position, $y(p)$ represents the p feature of the output position, and Δp_k represents the position offset of the learned k-th sampling point.

DCNv2 enhances the fitting ability of irregularly shaped objects. Based on DCNv1, a weight value is added for each sampling. The calculation formula:

$$y(p_0)=\sum_{p_k\in R} w(p_k)\cdot x(p_0+p_k+Dp_k)\cdot \Delta m_k \tag{12}$$

where, Δm_k represents the modulation factor of the k-th sampling point.

Given $x\in H\times W\times C$, the process of aggregation of K sampling points for each point p_0, the operation of DCNv3 is defined as:

$$y_g=\sum_{gk}^{K} m_{gk}\left(p_0+p_k+Dp_{gk}\right) \tag{13}$$

Early DCN still has limitations in speed and efficiency in practical application, while DCNv4 has carried out significant optimization and innovation in core technology, greatly enhancing the dynamic and expressive ability of the network [30]. In DCNv4, the introduction of unbounded dynamic weights allows the model to make adaptive adjustments according to the actual distribution of input features, so as to capture key feature information more accurately and adapt to small targets of different sizes and shapes more accurately. DCNv4 is also optimized for computation and stored procedures, reducing the number of memory accesses and the computational burden of bilinear interpolation coefficients, reducing unnecessary memory access requests, and significantly improving the model's operating efficiency and memory utilization by reducing redundant operations, achieving a speed increase of up to 80%. DCNv4 provides a variety of pre-trained models and profiles, reducing the number of bytes that the kernel needs to read and write, reducing the workload per thread, especially in large-scale operations can accumulate significant efficiency gains, becoming an important cornerstone of future vision models.

4.2 Attention Mechanism (AM)

The attention mechanism mimics the visual function of the human eye, improves the ability of the network to focus on key information in the detection task, and improves the efficiency and accuracy of the network detection. The AM is shown in Figure 3.

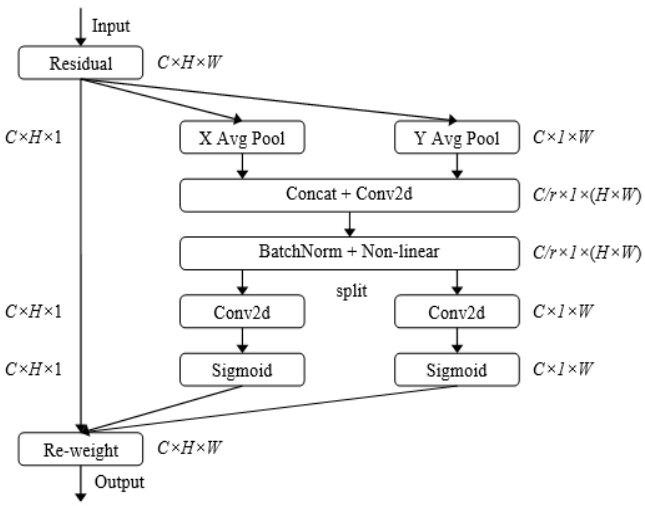

Figure 3. Attention mechanism structure

The improved attention mechanism of YOLOv8 algorithm consists of location information embedding and location attention generation.

(1) Location Information Embedding

The global average pooling operation is carried out in width and height respectively, and the feature map in each direction is obtained. The formula is expressed as:

$$z_c^h(h)=\frac{1}{W}\sum_{0\le i\le T} x_c(h,i) \tag{14}$$

$$z_c^w(w)=\frac{1}{H}\sum_{0\le i\le H} x_c(j,w) \tag{15}$$

(2) Location Attention Generation

First, the width and height direction feature maps are spliced, and then the feature maps are convolved. Then, BatchNorm operation is performed to get the feature graph F_1, and Sigmoid activation function is used to get the feature graph:

$$f=\delta\left(F_1\left(\left[z_c^h,z_c^w\right]\right)\right) \tag{16}$$

Then, the size of the feature graph remains unchanged, and the convolution calculation is carried out to obtain the feature graph F_h and F_w, and the attention weights are obtained respectively after Sigmoid activation function:

$$g^h=\sigma\left(F_h\left(f^h\right)\right) \tag{17}$$

$$g^w=\sigma\left(F_w\left(f^w\right)\right) \tag{18}$$

Finally, the original feature map is calculated using weighted multiplication to get the output:

$$y_c(i,j)=x_c(i,j)\times g_c^k(i)\times g_c^w(j) \tag{1}$$

4.3 Algorithm process and implementation details

The core improvement positions are the Backbone and the Neck. Among them, Backbone embeds DCNv4 deformable convolution in the residual block of CSPDarknet to enhance the perception ability of deformed targets. Neck inserts a cross-stage attention mechanism in the feature fusion path (FPN/PAN) to enhance multi-scale feature interaction. The flow of the improved YOLOv8 algorithm is shown in Figure 4.

DCNv4 improves the detection accuracy of the model for edge-blurred and rotating targets through dynamic offset adjustment. The attention mechanism optimizes the feature channels and spatial weights to suppress the interference of background noise. The AP indicators of the improved model in scenarios of dense targets, small targets and occlusion have been significantly enhanced. The specific implementation details include two aspects:

(1) CNv4 deformable convolution ensemble. The ordinary convolution of the residual block of CSPDarknet in Backbone is replaced with the DCNv4 module. Dynamic offset prediction generates spatial offsets through lightweight convolution to control the sampling positions of convolution kernels. Dynamic mask learning, introducing learnable Sigmoid masks to dynamically adjust the weights of different

sampling points. Support group convolution to reduce the computational load.

(2) Attention mechanism design. The FPN layer of the Neck adds cross-stage attention at the lateral connection of the feature pyramid. The PAN layer inserts spatial-channel attention in the top-down and bottom-up feature fusion paths. Among them, for cross-stage attention, when fusing features from different stages, channel weights are generated through global average pooling. Spatial attention generates spatial weights using deformable convolution on feature maps to focus on the target area. The specific implementation logic: Perform channel compression on the input features. Extract context information through global pooling or deformable convolution. Dynamically generate channel/space weights and multiply them element by element with the original features.

5. RESULTS AND DISCUSSION

In view of the existing problems of YOLOv8 algorithm, improvements are made from the two aspects of deformable convolution and attention mechanism of DCNv4. The effect of the improved results on STD needs to be verified by experiments.

5.1 Parameter setting

Before the experiment, the environment, including hardware environment and software environment, should be built first, and then the parameters of training YOLOv8 algorithm and its improved algorithm should be set up to lay the foundation for the experiment. The parameter Settings are shown in Table 1.

The specific reasons for the choice are described as follows: First, in terms of hardware. Powerful CPU and GPU can provide sufficient computing power to accelerate the training and inference process of the model. Sufficient memory and high-speed storage devices can ensure the rapid reading, writing and processing of data, and improve the efficiency of experiments. Second, in terms of software. The combination of Windows operating system, PyTorch framework, CUDA and cuDNN is a common choice in the field of deep learning, with good compatibility and performance optimization, and can provide a stable operating environment for experiments. Thirdly, in terms of parameter Settings. The selection of model parameters such as input image size, anchor box setting, and backbone network improvement is to better adapt to the requirements of small target detection and improve the detection accuracy of the model for small targets. The setting of training parameters is to ensure the convergence speed and generalization ability of the model and avoid overfitting and underfitting. The setting of data augmentation parameters can increase the diversity of the data set and improve the robustness and generalization ability of the model.

Table 1. Parameter settings

Class	Name	Configuration
Hardware environment	CPU	Inter(R) Core(R) i7-8700K
	GPU	NVIDIA V100
	Memory capacity	32GB
	Architecture	Volta
	CUDA core count	5120
	Hard disk	Solid state 512g
Software environment	OS	Windows 10 Enterprise Edition
	Language	Python 3.7
	Framework	Pytorch 1.4.0
	Development platform	PyCharm 2023
	Graphics processor	CUDA 11.6
	Data storage	MySQL 8.3
Parameter settings	Input image size	640×640
	Momentum for formal training	0.937
	Final learning rate	0.001
	Weight decay	0.0005
	Optimizer	SGD
	Batch size	96

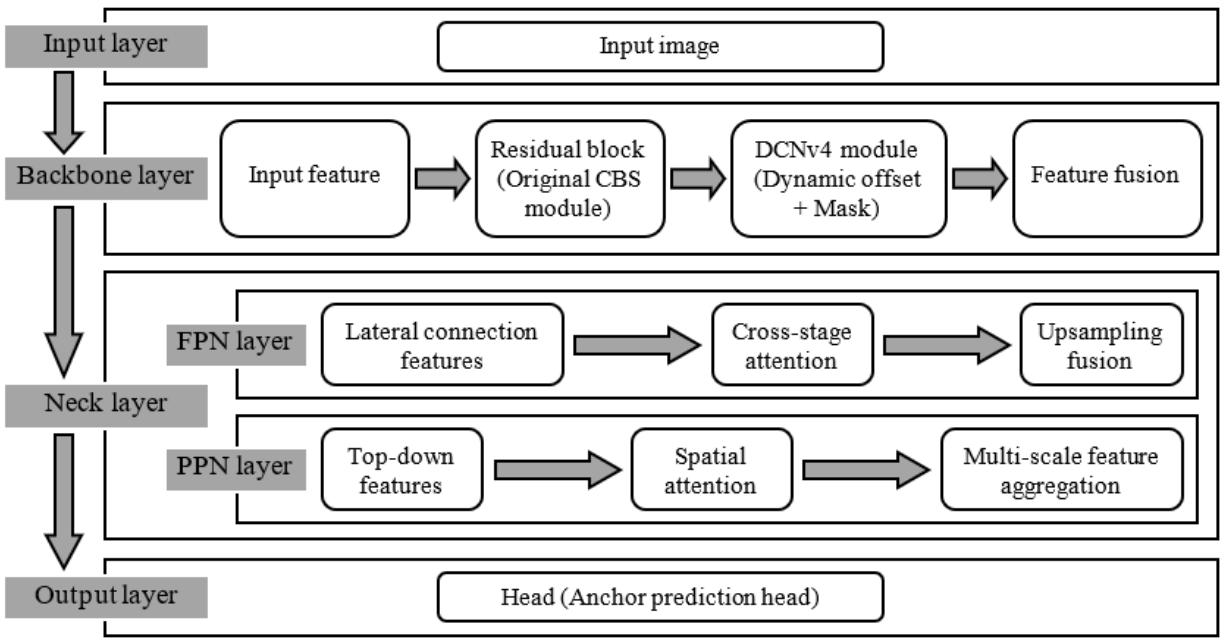

Figure 4. The improved YOLOv8 algorithm process

5.2 Evaluation index and data set

In order to evaluate the use evaluation effect, three indicators were used, namely Precision (P), Recall (R) and mAP. Where, P index represents the ratio of the number of correct samples detected to the total number of tested samples. The R index represents the ratio of the number of correct samples detected to the total number of samples in the test set. The mAP metric represents the mean of all categories of aps in the entire dataset. For mAP indicators, mAP@0.5 and mAP@0.5:0.95 are generally used in object detection. mAP@0.5 is the average accuracy for all classes when the IoU reaches 0.5. mAP@0.5:0.95 is the average accuracy for all classes as the IoU changes from 0.5 to 0.95. ASCAL VOC is a commonly used data set in the field of computer vision, including 20 common object categories, 1,000 images for testing, and 4,200 images for training and verification, which is convenient for comparing the performance of different algorithms [31]. And over time, the number of categories and the level of detail in the annotations continues to increase. PASCAL VOC data set uses XML file to label the label information. YOLOv8 algorithm trains the labeled image label file format to be TXT. Category information and four coordinate point information of each item need to be extracted from the XML file and saved in the corresponding TXT file.

5.3 Experimental results and analysis

The original model is denoted as YOLOv8, the model introducing deformable convolution with DCNv4 is denoted as YOLOv8+DCN, the model introducing attention mechanism is denoted as YOLOv8+AM, and the model introducing deformable convolution with DCNv4 and attention mechanism is denoted as YOLOv8+DCN+AM.

(1) Introducing DCNv4 Deformable Convolution

The experimental comparison results between the original model YOLOv8 and the deformable convolution model YOLOv8+DCN introduced by DCNv4 are shown in Table 2.

In Table 2, compared with YOLOv8+DCN, P, R and mAP have been improved to varying degrees. Among them, P index increased by 4.91%, R index increased by 3.58%, mAP@0.5 increased by 4.78%, mAP@0.5:0.95 increased by 2.91%.

(2) Attention Mechanism

The experimental comparison results between the original model YOLOv8 and the model YOLOv8+AM introduced with attention mechanism are shown in Table 3.

In Table 3, compared with YOLOv8+AM, P, R and mAP are all improved to varying degrees. Among them, P index increased by 3.07%, R index increased by 5.80%, mAP@0.5 increased by 3.67%, mAP@0.5:0.95 increased by 2.34%.

(3) The Deformable Convolution and Attention Mechanisms of DCNv4 are Introduced

The experimental between the original YOLOv8 and the YOLOv8+DCN+AM, which introduced deformable convolution and attention mechanism with DCNv4, are shown in Table 4.

In Table 4, compared with YOLOv8+DCN+AM, P, R and mAP are significantly improved. Among them, P index increased by 7.46%, R index increased by 8.97%, mAP@0.5 increased by 6.81%, mAP@0.5:0.95 increased by 5.65%.

Table 2. Comparison of YOLOv8 and YOLOv8+DCN

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8	52.12	41.37	40.36	24.41
YOLOv8+DCN	54.68	42.85	42.29	25.12
YOLOv8+DCN increase percentage	4.91%	3.58%	4.78%	2.91%

Table 3. Comparison of YOLOv8 and YOLOv8+AM

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8	52.12	41.37	40.36	24.41
YOLOv8+AM	53.72	43.77	41.84	24.98
YOLOv8+AM increase percentage	3.07%	5.80%	3.67%	2.34%

Table 4. Comparison of YOLOv8 and YOLOv8+DCN+AM

Model	P (%)	R (%)	mAP@0.5 (%)	mAP@0.5:0.95 (%)
YOLOv8	52.12	41.37	40.36	24.41
YOLOv8+DCN+AM	56.01	45.08	43.11	25.79
YOLOv8+DCN+AM increase percentage	7.46%	8.97%	6.81%	5.65%

Table 5. Comparison of the YOLOv8 algorithm with other small object detection algorithms

Algorithm	mAP@0.5	AP_small	FPS (on GPU)	Parameters (M)	FLOPs (G)	Background False Detection Rate (%)
YOLOv8	56.3	42.1	128	3.2	7.5	12.3
YOLOv7	54.7	39.8	95	2.5	5.8	14.1
YOLOv5	53.2	38.5	110	2.1	4.9	15.6
EfficientDet-D1	51.8	37.2	65	5.3	12.1	18.4
FCOS	50.1	35.9	82	4.7	9.3	16.8
NanoDet	49.5	34.7	145	1.8	3.2	17.9

5.4 Comparison of the YOLOv8 algorithm with other small object detection algorithms

On the ASCAL VOC dataset, a comparative experiment was conducted between the YOLOv8 algorithm and some other common small object detection algorithms. The experimental results are shown in Table 5.

The advantages of YOLOv8 are reflected in four aspects:

First, it has outstanding performance in small object detection. The AP_small reached 42.1%, significantly superior to YOLOv7 (39.8%) and FCOS (35.9%), mainly due to the improved Backbone (such as the CSPNet variant) and attention mechanisms (such as CBAM), which enhanced the extraction ability of tiny features. Second, there is a balance between real-time performance and accuracy. The FPS is as high as 128, far exceeding that of some high-precision models (such as 65FPS of EfficientDet-D1), and is suitable for real-time detection scenarios. Thirdly, lightweight design. The number of parameters is only 3.2M, which is superior to most SOTA models (for example, although the 1.8M of NanoDet is smaller, its AP_small is lower), and it has great deployment potential in edge devices. Fourth, training strategy optimization. By adopting self-supervised pre-training and data augmentation (such as Mosaic++), the robustness in complex backgrounds has been enhanced, and the background false detection rate (12.3%) is lower than that of YOLOv7 (14.1%).

The potential shortcomings of YOLOv8 are reflected in four aspects: First, there are still bottlenecks in the detection of extremely small objects. The detection effect on targets smaller than 16^2 pixels is poor (AP_tiny is not listed), and it needs to be further optimized by combining multi-scale feature fusion or Transformer. Second, the generalization ability for complex scenarios is limited. In densely occluded scenarios (such as the VisDrone dataset), mAP@0.5 drops by about 5%, weaker than targeted design models like TridentNet. Thirdly, the demand for computing resources is relatively high. Compared with lightweight models (such as NanoDet), its FLOPs are 134% higher and it is not friendly to the deployment of low-end devices. Fourth, there is room for post-processing optimization. The NMS threshold is fixed, which may lead to missed detection or false detection of small targets. Dynamic NMS strategies (such as Soft-NMS) may improve performance.

5.5 Potential limitations and future improvement directions

After the YOLOv8 algorithm introduces the DCNv4 deformable convolution and the attention mechanism, there is a significant improvement in small object detection. However, there are also some potential limitations, and it also provides a direction for future improvements.

The potential limitations are reflected in four aspects: First, the amount of calculation and the complexity of the model increase. Although the deformable convolution and attention mechanism of DCNv4 can improve the model performance, they also increase the complexity and computational load of the model. This may lead to the model requiring more computing resources and time during the training and inference processes. For some scenarios with high real-time requirements, such as mobile devices or resource-constrained embedded systems, it may be difficult to meet the performance requirements. The second is the risk of overfitting. After introducing new modules and mechanisms, the degree of freedom of the model increases, making it easier to fit the noise and details in the training data. If the training data is insufficient or diverse, it may lead to overfitting of the model and a decline in its generalization ability in the test set or practical applications. Thirdly, it has insufficient adaptability to extreme situations of small target sizes. Although the improved algorithm has a good detection effect on general small targets, for some small targets with extremely small sizes or in complex backgrounds, there may still be problems of insufficient detection accuracy. For example, when the size of a small target is less than a certain threshold, deformable convolution and the attention mechanism may not be able to fully capture its features, resulting in missed detection or false detection. Fourth, it is difficult to compress and optimize the model. Due to the increase in model complexity, the difficulty of model compression and optimization also increases accordingly. It is a challenging problem to compress the improved YOLOv8 model to a size suitable for a specific hardware platform without affecting the model performance and maintain a high inference speed.

The future improvement directions include five aspects: First, lightweight design. Study more efficient lightweight network structures and combine model compression techniques, such as pruning and quantization, to reduce the computational load and storage space of the model while ensuring detection accuracy. For example, it is possible to explore the lightweight transformation of deformable convolution and attention mechanisms, or to seek more lightweight alternatives to make them more suitable for resource-constrained environments. Second, data augmentation and optimization. Further enrich the data augmentation methods, specially design more effective augmentation strategies for the characteristics of small targets. For example, simulate the image changes of small targets under different lighting, occlusion, blur and other conditions, increase the diversity of training data, and improve the generalization ability of the model. Meanwhile, optimize the data preprocessing and annotation processes to ensure the quality and accuracy of the data. Thirdly, multimodal information fusion. Combining data from other modalities, such as infrared images and depth information, and fusing them with visible light images, more information is provided for the detection of small targets. Multimodal information can help the model better identify small targets, especially in complex backgrounds or low contrast situations, improving the accuracy and robustness of detection. Fourth, the adaptive mechanism. Design an adaptive model structure or parameter adjustment mechanism to enable the model to automatically adjust the detection strategy for small targets according to the characteristics of the input image. For example, the parameters of deformable convolution and the attention mechanism are dynamically adjusted according to the size, distribution density, etc. of small targets to better adapt to the detection requirements of small targets in different scenarios. The fifth point is cross-disciplinary knowledge transfer. Draw on the advanced technologies and methods in other fields, such as the experiences and achievements in small target detection in the fields of medical image analysis and remote sensing image detection, and transfer them to the improvement of the YOLOv8 algorithm to explore new research ideas and methods.

6. CONCLUSIONS

STD is an important research direction in target detection, which is mainly to detect, identify and locate small targets accurately. The traditional target detection algorithm extracts feature through a manually designed feature extractor. Although it does not require a large amount of data training, it has high time complexity and window redundancy, and is

sensitive to changes in illumination and angle. In the trend of the continuous development of AI, the application range of STD is very wide. Improving the accuracy of STD is the focus of research, which has positive significance for the improvement of computer vision technology level and multiple application scenarios. The STD algorithm based on improved YOLOv8 studied in this paper is improved from the introduction of DCNv4 deformable convolution and attention mechanism, so that the model can adapt to complex scenes and effectively solve the problem of poor STD performance. Future research can be carried out on the application of STD, and the model can be lightweight and deployed in embedded devices for practical production applications, including drone delivery and drone agricultural monitoring and other fields.

REFERENCES

[1] Bhanbhro, H., Hooi, Y.K., Zakaria, M.N.B., Kusakunniran, W., Amur, Z.H. (2024). MCBAN: A small object detection multi-convolutional block attention network. Computers, Materials & Continua, 81(2): 2243-2259. https://doi.org/10.32604/cmc.2024.052138

[2] Battish, N., Kaur, D., Chugh, M., Poddar, S. (2024). SDMNet: spatially dilated multi-scale network for object detection for drone aerial imagery. Image and Vision Computing, 150: 105232. https://doi.org/10.1016/j.imavis.2024.105232

[3] Nandal, P., Pahal, S., Malik, S., Sehrawat, N. (2025). Enhancing real time object detection for autonomous driving using YOLO-NAS algorithm with CLEO optimizer. International Journal of Information Technology, 17(3): 1321-1328. https://doi.org/10.1007/s41870-024-02296-w

[4] Ravi, S., Saranya, A. (2024). Breast cancer detection using machine learning in medical imaging–A survey. Procedia Computer Science, 239: 2235-2242. https://doi.org/10.1016/j.procs.2024.06.414

[5] Jothi, J.N., Nithila, E.E., Davix, X.A. (2024). Region duplication tampering detection and localization in digital video using haar wavelet transform. Wireless Personal Communications, 135(2): 655-674. https://doi.org/10.1007/s11277-024-11028-z

[6] Namah, A.A., Mirza, N.M., Al-Zuky, A.A. (2022). Target detection in video images using HOG-based cascade classifier. Revue d'Intelligence Artificielle, 36(5): 709-715. https://doi.org/10.18280/ria.360507

[7] Zhang, Y.B., Zhang, F., Zhang, J.J. (2022). Salient object detection algorithm based on SVM and background model. Electronic Design Engineering, 30(5): 17-21+27.

[8] Wang, Z.P. (2024). A small object detection algorithm based on improved YOLOv8s. Wuhan Polytechnic University.

[9] Umamageswari, A., Deepa, S., Hussain, F.B.J., Shanmugam, P. (2024). Enhancing underwater object detection using advanced deep learning de-noising techniques. Traitement du Signal, 41(5): 2593-2602. https://doi.org/10.18280/ts.410532

[10] Sagar, A.S., Chen, Y., Xie, Y., Kim, H.S. (2024). MSA R-CNN: A comprehensive approach to remote sensing object detection and scene understanding. Expert Systems with Applications, 241: 122788. https://doi.org/10.1016/j.eswa.2023.122788

[11] Khalili, B., Smyth, A.W. (2024). Sod-YOLOv8—enhancing YOLOV8 for small object detection in aerial imagery and traffic scenes. Sensors, 24(19): 6209. https://doi.org/10.3390/s24196209

[12] Kumar, N., Singh, P. (2025). Small and dim target detection in infrared imagery: A review, current techniques and future directions. Neurocomputing, 630: 129640. https://doi.org/10.1016/J.NEUCOM.2025.129640

[13] Kanade, K.A., Potdar, P.M., Kumar, A., Balol, G., Shivashankar, K. (2025). Weed detection in cotton farming by YOLOv5 and YOLOv8 object detectors. European Journal of Agronomy, 168: 127617. https://doi.org/10.1016/J.EJA.2025.127617

[14] Bhalla, S., Kumar, A., Kushwaha, R. (2024). Feature-adaptive FPN with multiscale context integration for underwater object detection. Earth Science Informatics, 17(6): 5923-5939. https://doi.org/10.1007/s12145-024-01473-6

[15] Shinde, A.S., Patil, V.V. (2024). Effect of data augmentation, cross-validation methods in robustness of explainable speech based emotion recognition. Traitement du Signal, 41(3): 1565-1574. https://doi.org/10.18280/ts.410344

[16] Sivapuram, A.K., Komuravelli, P., Gorthi, R.K.S. (2025). SA-LfV: Self-annotated labeling from videos for object detection. Machine Learning, 114(1): 21. https://doi.org/10.1007/s10994-024-06676-y

[17] Kim, K., Lee, S., Kakani, V., Li, X., Kim, H. (2024). Point cloud wall projection for realistic road data augmentation. Sensors, 24(24): 8144. https://doi.org/10.3390/s24248144

[18] Xue, Z., Yao, T. (2024). Enhancing occluded pedestrian re-identification with the motionblur data augmentation module. Mechatronics and Intelligent Transportation Systems, 3(2): 73-84. https://doi.org/10.56578/mits030201

[19] Silva, D.A., Smagulova, K., Elsheikh, A., Fouda, M.E., Eltawil, A.M. (2025). A recurrent YOLOv8-based framework for event-based object detection. Frontiers in Neuroscience, 18: 1477979. https://doi.org/10.3389/fnins.2024.1477979

[20] Nasehi, M., Ashourian, M., Emami, H. (2024). Vehicle type and speed detection on Android devices using YOLO V5 and MobileNet. Traitement du Signal, 41(3): 1377-1386. https://doi.org/10.18280/ts.410326

[21] Liu, C., Yang, J., Liu, Y., Zhang, Y., Liu, S., Chaikovska, T., Liu, C. (2023). A cervical lesion recognition method based on ShuffleNetV2-CA. Information Dynamics and Applications, 2(2): 77-89. https://doi.org/10.56578/ida020203

[22] Jaware, T.H., Patil, J.P., Badgujar, R.D. (2025). Robust deep learning approach for colon cancer detection using MobileNet. Journal of The Institution of Engineers (India): Series B. https://doi.org/10.1007/s40031-025-01217-0

[23] Srinivasan, D., Kiran, A., Parameswari, S., Vellaichamy, J. (2025). Bonevoyage: Navigating the depths of osteoporosis detection with a dual-core ensemble of cascaded ShuffleNet and neural networks. Journal of X-Ray Science and Technology, 33(1): 3-25. https://doi.org/10.1177/08953996241289314

[24] Battish, N., Kaur, D., Chugh, M., Poddar, S. (2024). SDMNet: spatially dilated multi-scale network for object

detection for drone aerial imagery. Image and Vision Computing, 150: 105232. https://doi.org/10.1016/j.imavis.2024.105232

[25] Tang, D., Tang, S., Fan, Z. (2024). LCFF-Net: A lightweight cross-scale feature fusion network for tiny target detection in UAV aerial imagery. PloS One, 19(12): e0315267. https://doi.org/10.1371/journal.pone.0315267

[26] Alamri, F. (2025). Comprehensive study on object detection for security and surveillance: A concise review. Multimedia Tools and Applications. https://doi.org/10.1007/s11042-025-20801-6

[27] Saidani, T. (2025). Efficient Tumor detection in medical imaging using advanced object detection model: A deep learning approach. International Journal of Advanced Computer Science & Applications, 16(1): 1139-1145. https://doi.org/10.14569/ijacsa.2025.01601109

[28] Meza, G., Ganta, D., Gonzalez Torres, S. (2024). Deep Learning Approach for Arm Fracture Detection Based on an Improved YOLOv8 Algorithm. Algorithms, 17(11): 471. https://doi.org/10.3390/A17110471

[29] Abdullahi, M., Oyelade, O.N., Kana, A.F.D., Bagiwa, M.A., Abdullahi, F.B., Junaidu, S.B., Iliyasu, I., Ore-ofe, A., Chiroma, H. (2024). A systematic literature review of visual feature learning: deep learning techniques, applications, challenges and future directions. Multimedia Tools and Applications, 1-58. https://doi.org/10.1007/s11042-024-19823-3

[30] Saad, A.M., Rahi, M.R.H., Islam, M.M., Rabbani, G. (2025). Diet engine: A real-time food nutrition assistant system for personalized dietary guidance. Food Chemistry Advances, 7: 100978. https://doi.org/10.1016/J.FOCHA.2025.100978

[31] Le Jeune, P., Bahaduri, B., Mokraoui, A. (2025). A comparative attention framework for better few-shot object detection on aerial images. Pattern Recognition, 161: 111243. https://doi.org/10.1016/j.patcog.2024.111243