International Journal of Mechanical and Electrical Engineering
ISSN: 3005-9615 (Print), ISSN: 3005-7132 (Online) | Volume 4, Number 3, Year 2024
DOI: https://doi.org/10.62051/ijmee.v4n3.03
Journal homepage: https://wepub.org/index.php/IJMEE/index

Detecting Algorithm based on the Improved YOLOv8s for a Weak Feature Defect of Aviation Clamps

Qinpeng Luo, Yinhua Liao *

School of Mechanical Engineering, Sichuan University of Science & Engineering, Yibin, Sichuan, China

* Corresponding author: Yinhua Liao

ABSTRACT

There are some weak defects on the surface of aviation clamps. Because they are very weak, it is difficult to identify them efficiently and accurately by the existing visual detecting algorithms, and the existing methods have high arithmetic power requirements for vision detecting systems. So, this work proposes a detecting algorithm for a weak feature defect detection of aviation clamps (YOLO-OGS). Firstly, in order to improve the ability of convolutional operations of extract features and decrease the model's GFLOPs, the Multidimensional dynamic convolutional ODConv is added to the backbone network of YOLOv8. Then, in order to reduce the complexity of the model while increasing the effectiveness of feature fusion at various levels by keeping more of the hidden connections in the channels, the GhostSlimFPN paradigm network structure, which contains GSConv convolution and slim-neck structure, is introduced in the neck network. Finally, the Shuffle Attention module is used to widen the image's sensory field and enhance the details of weak flaws. Based on the aviation clamp defect data set, the comparative analysis results of YOLO-OGS and YOLOv8s algorithms show that YOLO-OGS decreases the GFLOPs by 14.4% and increases precision, recall, mAP@0.5, and GFLOPs by 4%, 6.4%, and 3.4%, respectively. And compared with the other existing mainstream networks YOLOv6, YOLOv8s, YOLOv8n, YOLOv5n, YOLOv5s, YOLOv7, YOLOv3-tiny. 8.3%, 3.4%, 4.4%, 15.4%, 8%, 13.4%, and 12% improvement in mAP@0.5.

KEYWORDS

Aviation Clamps; Weak Feature Defect; YOLOv8 Algorithm; Defect Detection.

1. INTRODUCTION

The aviation industry has become an essential part of the global economy due to the development of contemporary industries and technological breakthroughs. An aviation clamp is a kind of structural part widely used in aircraft. Because the performance and reliability of clamps are crucial to the flight safety of aircraft. Therefore, how to ensure the quality of the structural components of the clamps has become the focus of attention of the aircraft component manufacturers. During production and processing, clamps will inevitably show various defects such as scratches, pits, rust, pockmarks, and others. In the inspection of clamps, conventional inspection methods pose a potential risk of overlooking or misidentifying defects characterized by weak features. As the industrial sector continues to advance, the existing inspection technologies are no longer adequate to meet the demands of industrial production. Conventional inspection techniques have a low detection effect and are unable to reliably identify faint distinctive faults in the workpiece when it comes to clamp inspection. It has been challenging for the current inspection technology to keep up with the demands of industrial production as a result of the industry's ongoing development. As Industry 4.0 emerges[1], the manufacturing industry is undergoing a paradigm shift towards digital manufacturing [2], which

Content from this work may be used under the terms of CC BY-NC 4.0 licence (https://creativecommons.org/licenses/by-nc/4.0/).
Published by Warwick Evans Publishing.

necessitates enhanced precision in quality inspection tasks for industrial products. Currently, there are two groups of approaches to classify defects on workpiece surfaces: those based on conventional image processing techniques and those leveraging deep learning technology.

Traditional machine learning-based defect classification methods include Random Forest (RF) [3], K-Nearest Neighbor (KNN) [4], Support Vector Machine (SVM) [5], and others. Conventional machine learning techniques have demonstrated promising outcomes in the classification of surface defects on workpieces. For instance, Hu et al [6]proposed a segmentation method that employs local histogram adaptive thresholding and utilizes support vector machines for the classification of oxidation defects, enabling automatic detection of such defects on copper foil surfaces. Jian [7] et al eliminated the effect of ambient light fluctuation in cell phone screen glass (MPSG) images by employing the subtractive projection method (CSP). They also used an improved fuzzy mean clustering algorithm to extract the boundary defects from MPSG images. The detection time for this method was 1.6s, and the detection accuracy reached 97%. In order to segment and identify the surface flaws of lithium batteries, Xu et al [8]. Employed image subtraction, the Canny algorithm, AND logic operation, particle swarm optimization (PSO), and support vector machines. Their average detection precision was 98.3%. After applying the improved Canny edge detection algorithm in combination with probabilistic Hough transform, Guo et al [9] successfully achieved accurate splice detection of scratches in brake disc detection. Compared with the traditional Canny algorithm, the improved Canny operator exhibits a high accuracy of 97.96% and a miss detection rate of only 3.33% for brake disc scratch detection. Traditional machine learning-based methods for surface defect detection on workpieces have achieved significant results, effectively bridging the gap left by manual inspection methods to a large extent. However, these traditional methods require separate design for optimal performance and are limited in their application scenarios.

The advent of deep learning has significantly bolstered the robustness of surface defect detection algorithms, enabling automated and efficient extraction of workpiece surface defect features. Consequently, deep learning models have exhibited exceptional performance in tasks related to surface defect detection and classification. Xu et al [10] proposed an enhanced metal surface defect detection technology based on the refinement of the YOLO model, which significantly improves the precision in localizing minute defects. Wu et al [11]. proposed an enhanced SSD-assisted electromagnetic luminescence surface defect detection algorithm, which effectively improves the recognition rate for various types of defects. Cha et al [12]were the pioneers in directly applying Faster R-CNN for surface defect localization on bridges. In this approach, they substituted the Backbone network with ZF-net and employed a dataset comprising 2,366 images of bridge structures belonging to 5 distinct classes, each having a resolution of 500×375 pixels. The experimental findings demonstrated that the method achieved a mAP (mean average precision) value of 87.8%. Ye et al [13] used an improved Faster R-CNN algorithm for the detection of surface defects on ferrite shields, achieving an mAP of 81%. The enhanced YOLOv8 algorithm was proposed by Wang et al[14] for the purpose of road surface defect detection, achieving an mAP of 90.7%. After conducting a comprehensive analysis of common defects observed on steel plate surfaces and existing detection algorithms, Zhang et al [15]opted to enhance the YOLOv5 model, resulting in significant advancements. The enhanced YOLOv5 algorithm achieved an impressive a mAP of 83.8 in detecting surface defects on steel plates, thereby substantially improving the detection accuracy particularly for small-scale defects. The aforementioned approaches have exhibited certain advancements in the realm of small target detection; however, their progress in investigating weak feature defects remains insufficient. The computational efficiency of industrial inspection equipment imposes specific algorithmic requirements, as high GFLOPs and intricate calculation parameters can impede the algorithm's deployment and detection speed. Therefore, it is necessary to enhance the original algorithm in order to optimize its computational efficiency.

This paper proposes a YOLO-OGS model to enhance the detection capability of small defects in aviation clamp fittings, focusing on weak feature defects. The main contributions of this paper are outlined below:

1. To augment the feature extraction capacity of convolutional operations and alleviate the computational load of the model, we incorporated ODConv convolutions into the primary network. By leveraging ODConv's multi-dimensional convolution capabilities to extract features from all four dimensions of information, we maintained a lightweight architecture while enhancing image feature extraction prowess.

2. In order to enhance the transmission of small target feature information in convolutional neural networks and simultaneously reduce model complexity. We propose replacing the conventional Feature Pyramid Network (FPN) with the GhostSlimFPN (GS) architecture, which facilitates cross-scale connections between different layers and cross-layer connections within the same scale.

3. Incorporating the Shuffle Attention module into the neck network enhances the model's semantic understanding and information fusion capabilities for images.

The remaining sections of this paper are structured as follows: Section 2 presents a comprehensive overview of the proposed method's architecture; Section 3 outlines our approach to constructing the dataset; In Section 4, we showcase the exceptional performance of our model through ablation and comparative experiments; Finally, Section 5 concludes the paper and provides insights into future directions.

2. YOLOV8 ALGORITHM IMPROVEMENTS

2.1. YOLOv8

YOLOv8 is an improvement over the previous version of YOLO, enhancing performance and making the model fast, accurate, and easy to use. Additionally, the model can simultaneously implement classification, detection, segmentation, and pose estimation tasks. Compared to previous versions, YOLOv8 introduces a novel architecture that eliminates the use of anchor boxes, resulting in substantial enhancements in both speed and accuracy. The YOLOv8 model comprises four essential components: an input layer, a backbone network, a neck module, and a head module. The input is processed at a resolution of 640x640 in the backbone, which comprises multiple Conv and C2f modules. The C2f module is built upon the Efficient Layer Aggregation Network (ELAN) employed in YOLOv7 [16]. The primary objective of ELAN is to address the issue of convergence deterioration in deep models during model scaling. This goal is achieved by eliminating the 1×1 convolution sampling layer from the Neck component and substituting the C3 module with the C2f module. Furthermore, ELAN employs a decoupled head structure in the Head component, effectively separating the classification task from the regression task. Decoupled heads enable the segregation of feature extraction and pixel-level prediction, thereby enhancing the network's ability to process diverse scales and semantic information with greater flexibility. The proposed method is capable of effectively integrating feature information across multiple scales, thereby enhancing its capability in accurately segmenting targets with varying scales. During the training process, Mosaic data augmentation [17] was employed for image preprocessing, and its utilization was discontinued in the final 10 epochs to enhance detection accuracy.

2.2. Improved YOLOv8 Algorithm

This paper employs YOLOv8s as the foundation for enhancing the detection of weak defects on clamp surfaces. Figure 1 illustrates the overall architecture of the enhanced YOLO-OGS model built upon YOLOv8. First, introduce the multi-dimensional dynamic module ODConv to the backbone network, enhancing convolution operations from multiple dimensions and appropriately reducing the model's

computational complexity. To further improve the network performance, the PANet feature module in the neck network was replaced with GSFPN. This can enhance the network's detection performance for objects of different scales. At the same time, it can also reduce the network's computational load and parameters, thereby improving the overall network performance. Furthermore, introducing the Shuffle Attention module, which combines spatial and channel feature information, to enhance the network's ability to extract subtle defects.

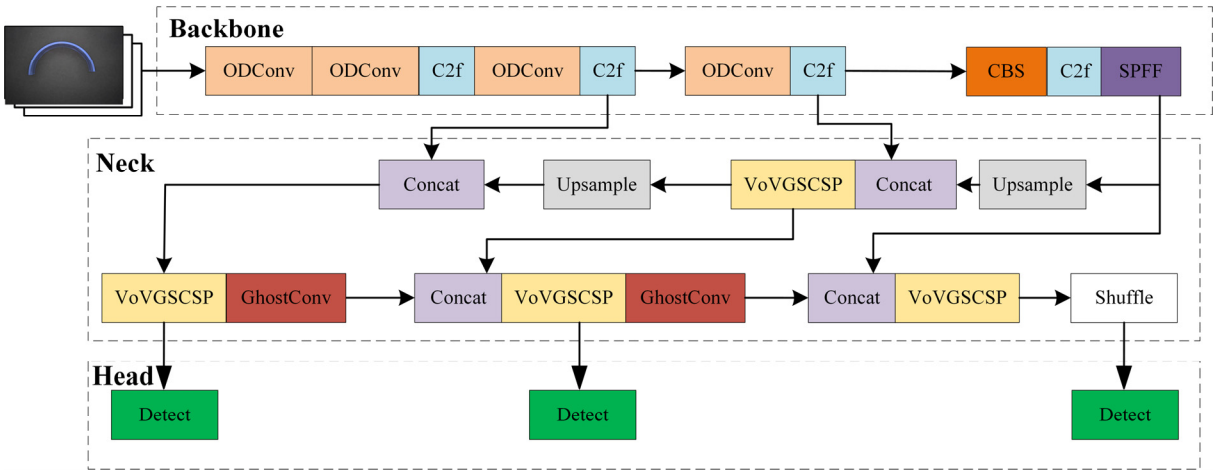


Fig 1. YOLO-OGS network model

2.3. Multidimensional Dynamic Convolution Module

In current neural networks, the majority of models employ static convolutional kernels to detect small targets [18-20]. However, this approach still exhibits certain limitations. The following are the primary drawbacks:

1. Difficulty in accurately localizing small objects: Because the convolutional networks used in the YOLO model typically have a large receptive field, this may lead to inaccurate positioning of the bounding boxes. The receptive field refers to the size of the input image area that a neural network can see in the feature map output at a certain layer. Having a large receptive field can make it difficult for the model to precisely locate small objects.

2. Limited feature representation: In the YOLO model, pooling operations and large stride convolutions are often used to reduce computational complexity and improve speed. However, these operations may result in the model losing details of small objects. Pooling operation reduces the resolution of feature maps, while large stride convolutions result in information loss. These factors may affect the accurate detection of small objects.

3. The categories in the training data are imbalanced: In training data, the number of samples for large objects is often significantly higher than that for small objects. As a result, the algorithm may prioritize the detection of large objects and have fewer training samples for small objects. The imbalance in this category can impact the model's capacity to learn about small objects, leading to a decreased accuracy in detecting small objects.

Dynamic convolution incorporates attention mechanisms and deformable convolution to overcome the aforementioned limitations. Specifically, the attention mechanism enables the model to prioritize important information during the convolution process, thereby enhancing the model's perception ability and accuracy. The deformable convolution enables the convolution kernel to undergo minor deformations within a specified range, thereby adapting to local variations in the data and enhancing the model's representational capability. However, dynamic convolutions such as DyConv[21] and CondConv[22] only focus on the dynamic nature of kernel size, while neglecting the dynamic characteristics of input channels, spatial dimensions, and output channels. We therefore introduce the

ODConv[23] multidimensional dynamic convolution module, which has the following characteristics:
1. Multidimensional Attention Mechanism: By using an innovative method for multidimensional attention, ODConv can simultaneously learn four different kinds of attention in each of the four convolutional kernel space dimensions. These four attentional modalities are applied methodically to the relevant convolutional kernels and complement each other.
2. Full Dimensionality Feature: ODConv is called a full-dimensional dynamic convolution because it considers the dynamic character of dimensions like space, input channels, and output channels. This method may greatly improve the convolutional neural network's (CNN) fundamental convolution operation's feature extraction capacity and improve scene capture.
Additionally, it achieves a better balance between model efficiency and precision, leading to enhanced recognition even for complicated models [24]. The formula for ODConv is as follows:

$$y = (\alpha_{w1} \odot \alpha_{f1} \odot \alpha_{c1} \odot \alpha_{s1} \odot W_1 + \ldots + \alpha_{wn} \odot \alpha_{fn} \odot \alpha_{cn} \odot \alpha_{sn} \odot W_n) * x \tag{1}$$

Where $\alpha_{wi} \in R$ stands for convolution kernel W_i's attention scalar. Three newly introduced attention points, denoted by $\alpha_{si} \in R^{K \times K}$, $\alpha_{fi} \in R^{C_{out}}$ and $\alpha_{ci} \in R^{C_{in}}$, are computed along the input channel dimension, the output channel dimension, and the spatial dimension of the convolution kernel W_i's kernel space. Figure 2 illustrates how the 4 dimensions are computed in ODConv. The network's sensitivity to small-scale targets is enhanced by these 4 computations, which retain a significant quantity of spatial information.

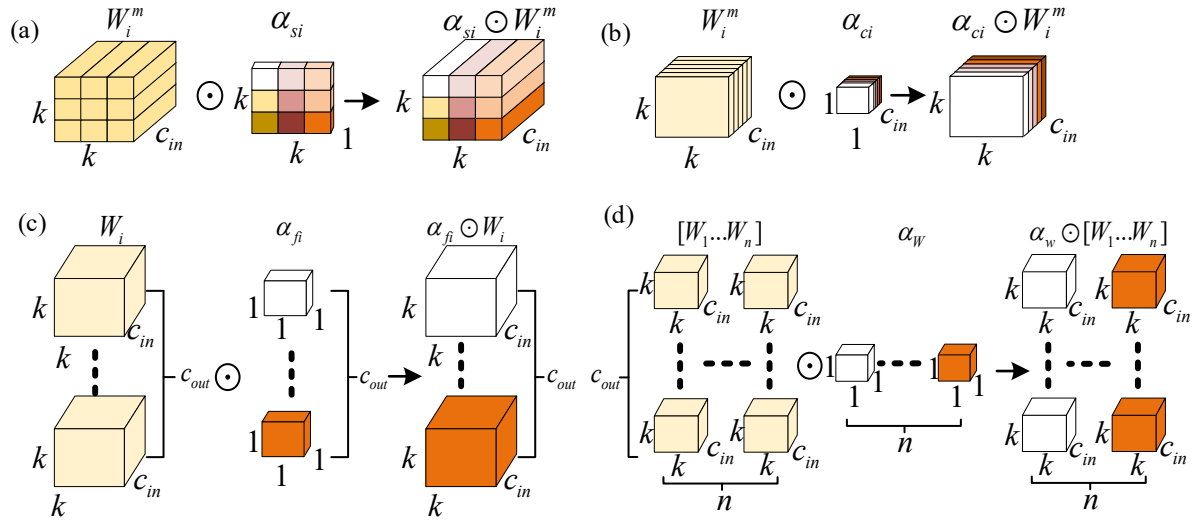


Fig 2. (a) The multiplication process along the spatial dimension, (b) The process of multiplying along the input channel dimension, (c) The process of multiplying along the output channel dimension, and(d)kernel-wise multiplication operations along the kernel dimension of the convolutional kernel space.

2.4. Feature Fusion Optimization

The original Yolov8 network utilizes the PANet network structure to facilitate feature fusion, thereby mitigating the potential loss of feature information that may occur in excessively deep networks. PANet only takes into account the impact of both the same scale between neighboring layers and the neighboring scales between the same layers. More information loss occurs in the process of feature fusion, especially the features of small targets. However, this approach results in greater loss of information, particularly for the characteristics of small targets. Feature fusion plays a crucial role in

object detection as it allows for the integration of feature information from different layers. The traditional Con-cat method simply connects features of different levels in the channel dimension. To enhance the accuracy of detecting weak defects and reduce the computational complexity in the network, this paper replaces the original PANet feature fusion module with the GSFPN [25]module. GSConv is a convolutional technique utilized to address the issue of separating channel information in deep convolutions. As illustrated in Figure 3, GSConv is capable of preserving the continuity of channel information across different stages of the model, effectively addressing the issue of channel information being fragmented in traditional convolution. It is worth noting that using GSConv throughout all stages of the model will increase the depth of the network layers. This may result in an increase in the time required for transmitting and inferring information. Therefore, when designing models, a trade-off must be made between depth and inference efficiency, and the choice should be based on the specific task. When using GSConv at the neck of the model, duplicate information can be effectively reduced and excessive compression of the feature map can be avoided by processing the feature map in series. Therefore, this approach can effectively preserve feature information and enhance the model's performance and inference efficiency in tackling complex tasks.

To maintain model accuracy and reduce complexity, this paper adopts GSConv and combines it with the VOV-GSCSP module within the slim-neck design paradigm. Figure.4(a) and (b) show the GSbottleneck and VoV-GSCSP structures, respectively. The cross-level partial networks (GSCSP) module is implemented on top of GSConv and is created via the VoV-GSCSP architecture, which combines the GSbottleneck and a one-shot aggregation technique. This module effectively balances model precision and speed, reduces computational complexity and network structure, and ensures adequate precision and reuse of extracted features.

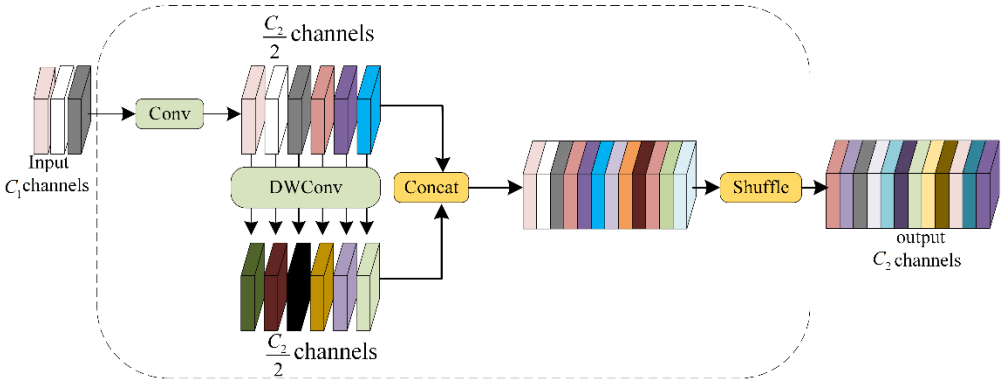


Fig 3. Detailed structure of GSConv

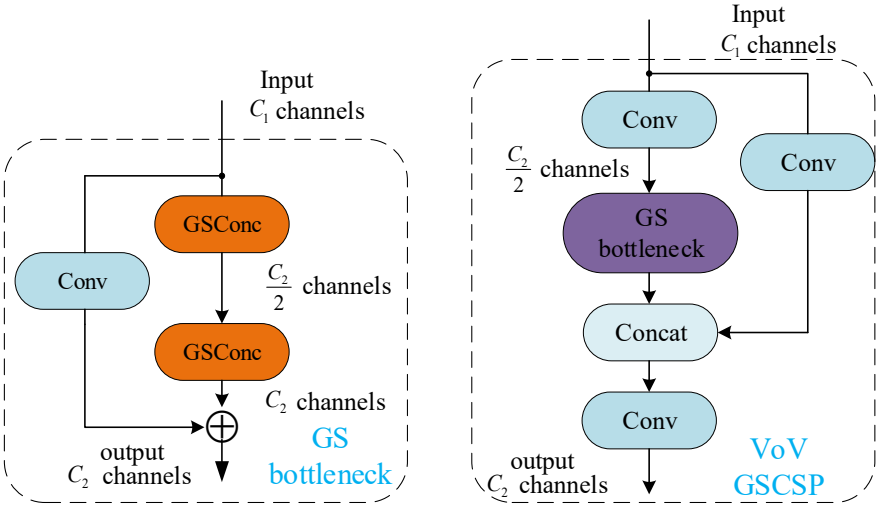


Fig 4. GSbottleneck module (left) and VoV-GSCSP module (right)

2.5. Introducing an Attention Module to the Neck Network

Attention mechanism plays a crucial role in the field of deep learning. It emulates the attention mechanism of human beings in processing information, allowing the model to selectively focus on the crucial parts of the input data. This enhances the model's performance and its ability to generalize. In the field of computer vision, numerous outstanding attention mechanisms have emerged. These include channel attention mechanisms such as Squeeze-and-Excitation Attention(SE) [26] , spatial attention mechanisms like Convolutional Block Attention Module(CBAM) [27] and Coordi−nate Attention(CA) [28], and attention mechanisms that integrate spatial and spectral information to process image feature information, such as S2 [29]. Although the combination of spatial attention and channel attention in a single attention mechanism has been shown to significantly enhance the network's feature extraction ability [30], it also presents the challenge of having an excessive number of computational parameters. Therefore, this paper adopts the Shuffle Attention (SA) module to address this issue.

Shuffle Attention (SA) [31]is a lightweight spatial channel attention mechanism as in Fig.5. The input feature tensor $X \in R^{C\times H\times W}$, where C represents the input feature tensor's channel and H and W stands for the input feature tensor's height and width, respectively, can be combined with the channel attention module and the spatial attention module efficiently. SA divides the feature tensor into G groups, such as, After that X_K divides into two branches along the channel dimension, i.e. X_{K1} , $X_{K2} \in R^{\frac{C}{2G}\times H\times W}$. X_{K1} denotes channel attention branching and X_{K2} denotes spatial attention branching.

Channel Attention Branch (black dashed box in Fig.5 channel feature information is generated using Global Average Pooling (GAP)[32]. As in equation (2):

$$S = F_{GP}\left(X_{K1}\right) = \frac{1}{H\times W}\sum_{i=1}^{H}\sum_{j=1}^{W} X_{K1}\left(i,j\right) \tag{2}$$

where $S \in R^{\frac{C}{2G}\times 1\times 1}$ denotes the feature channel information. The channel orientation weights are then recalibrated using sigmoid activation. Compact features are generated, which are defined by the formula:

$$X'_{K1} = \sigma\left(F_c\left(s\right)\right)\cdot X_{K1} = \sigma\left(W_{1S} + b_1\right)\cdot X_{K1} \tag{3}$$

where $W_1, b_1 \in R^{\frac{C}{2G}\times 1\times 1}$, W_1 and b_1 denote the weight matrix and bias term in the weighted summation formula, respectively.

Spatial attention branching (red dashed box in Fig. 5): The input feature map is normalized, and then it will be enhanced by transforming the input formulas as follows:

$$X'_{K2} = \sigma\left(W_2\cdot GN\left(X_{K2}\right) + b_2\right)\cdot X_{K2} \tag{4}$$

where $W_2, b_2 \in R^{\frac{C}{2G} \times 1 \times 1}$, W_2 and b_2 denote the weight matrix and bias term in the weighted summation formula, respectively. GN is the Group Norm. Finally, the two attentions are fused, i.e., $X'_K = [X'_{K1}, X'_{K2}] \in R^{\frac{C}{G} \times H \times W}$.

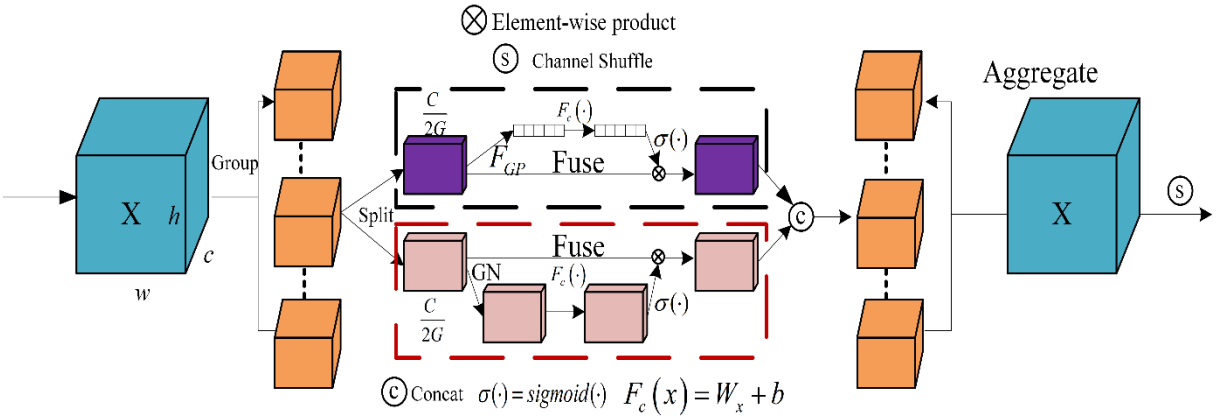


Fig 5. Structure of the SA attention mechanism

Finally, the SA module fuses the information within the groups by concatenating them, rearranges the groups using a channel mixing operation, and allows the information to flow between the different groups. In this paper, the model inserts SA in the neck network that can combine spatial and channel attention mechanisms for partitioning and parallelizing operations. This can improve the semantic representation of the weak features of the clamps and focus on the weak feature defects of the clamps.

3. CLAMP DEFECT DATA SET

To address the challenge of detecting defects in the weak features of clamps, we gathered 4877 images from the factory to assess the effectiveness of the proposed network. The images were captured by Hikvision's MV-S060 industrial camera equipped with the MVL-HF1628M-6MPE 6-megapixel industrial lens. With the assistance of a coaxial light source, position the clamp 5 cm away from the coaxial light source and set the light source to channel 2. This effectively reduces reflections from the metal surface, thereby enhancing the contrast between the defect and the background. The dataset comprises point defects, line defects, and regional defects, as illustrated in Figure 6. The specific criteria for evaluation are as follows:

1. Spot defects: Spot defects are pitting knocks on the surface of the clamp, which can be caused by human error during processing. This condition makes the surface plating of the clamp incomplete and can be classified as a spot defect if one or more knocks exceed 0.2mm.

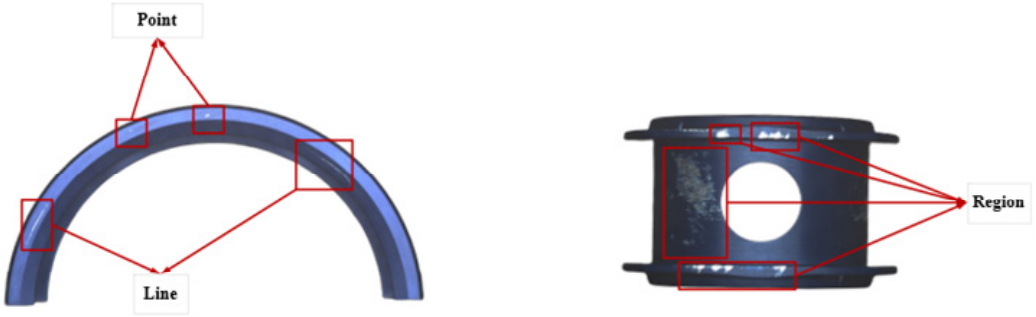
2. Linear defects: Linear defects are elongated scratches found on the surface of the clamp, typically concentrated in the center. These defects can significantly impact the overall service life of the clamp. Linear defects are considered to be linear when their length exceeds 3 mm and their width is greater than 0.1 mm.

3. Regional defects: Regional defects are large defects on the surface of the clamp, which will affect its normal assembly. If the defect area is large, the cumulative area is larger than 3×3 mm and the color is uneven, then it is a regional defect.

Three different types of defects in the dataset are illustrated in Figure 6, covering a wide range of clamp models. This leads to a data sample that is more representative and inclusive, providing a closer

reflection of the real production setting. Details of the dataset can be found in Table 1. Some of these images cover many different types of defects, providing a richer and more diverse data source for research. A dataset created from real-life production scenarios' photographs can better evaluate the strength and overall performance of defect detection algorithms. Offer more specific and feasible solutions for manufacturing companies. This dataset is closer to practical application scenarios and improves the applicability and reliability of the algorithms in real production.

This dataset has been split into a training set, validation set, and test set with the distribution of 8:1:1. During training, image-enhancing methods including brightness transformation and Gaussian blurring are used. These enhancements simulate blurring and uneven brightness in images, thereby increasing sample diversity and improving the model's generalization ability.



(a) Line defects and Point defects (b) Regional defects

Fig 6. Several instances of defects from the clamp dataset

Table 1. Type and number of defects in the clamp dataset

Clamp defects	Defect Name	Quantities
Line defect	L	2041
Regional defect	R	100
Point defect	P	2317

4. RESULTS

4.1. Performance Indicators

In this experiment, the algorithm's performance is evaluated using metrics such as Precision, Recall, Giga Floating-point Operations Per Second (GFLOPs), and mean Average Precision (mAP). The class-wide mean average precision is determined by the precision and recall, as depicted in Equation (5):

$$mAP = \frac{\sum_{i=1}^{N} \int_0^1 PdR}{N} \tag{5}$$

The precision rate is determined by dividing the total number of expected positive samples by the number of properly predicted positive samples, as expressed in equation (6):

$$precision = \frac{TP}{TP + FP} \tag{6}$$

In the formula mentioned above, the mAP value is calculated by averaging the AP (Average Precision) of each type of defect. Here, N represents the total number of categories detected by the network, while FP (False Positives), TP (True Positives), and FN (False Negatives) represent the counts of falsely detected, correctly detected, and missed frames, respectively. During the evaluation of the experiment's results, a higher mAP value signifies that the model exhibits better detection performance on the given dataset. This metric considers the precision and recall for each defect type and provides an overall measure of the model's effectiveness in detecting various defects.

4.2. Experimental Environment

To assess the proposed method's efficacy, we opt to utilize YOLOv8s as the foundational network model. The experimental environment's particular setup can be found in Table 2. Training was conducted with the following hyperparameters, taking into account the hardware conditions: a momentum of 0.937 was utilized, an initial learning rate of 0.01 was set, and optimization was carried out using the SGD optimizer with a weight decay of 0.0005. The diversity of images input into the network is maintained through the use of data enhancement techniques, including Gaussian blurring and rotational transformation, to enhance the data online. The dimensions of the input image for the model are 960×960×3, with a batch size of 16 and the network training is set to run for 300 epochs.

Table 2. Configuration and training environment

Name	Environmental Parameters
operating system	Windows 10 64-bit
GPU	NVIDIA RTX A5000 24GB
CPU	AMD Ryzen9 5950X 16-core 3.4GHz
RAM	64GB
Python	3.9.18
Pytorch	1.12.1

4.3. Examination of the Experimental Findings

1) Experimental level analysis of Backbone network improvements

We experimentally explore the impact of the multi-dimensional dynamic convolution ODConv on network performance. We embed the ODConv convolution into the YOLOv8s model's backbone network in order to reconfigure the network arithmetic. Utilizing the YOLov8s model as a foundation, the backbone network of the base model is enhanced with the addition of ODConv convolution. Six rounds of model comparisons were conducted, and the findings are presented in Table 3. Based on the data from Experiments No. 4 and 5, it can be concluded that Experiment No. 5 has shown improvements of 1.3%, 0.8%, and 0.6% over Experiment No. 4 in terms of recall, precision, and mAP@50, respectively. Experimental results 2, 3, 5, and 6 demonstrate that incorporating four ODConv layers enhances the network's detection accuracy while also decreasing its GFLOPs. Compared to Experiment 1, Experiment 6 improved accuracy by 1.9%, recall by 4%, and mAP@0.5 value by 0.7%, while GFLOPs decreased by 6.27%. Therefore, to simplify the model and enhance its performance, scheme number 6 was selected as the initial improvement outcome, and the model was

designated as YOLO-O. As shown in Fig. 7 (b) and (c) two sets of feature maps are visualized to demonstrate the capability of traditional convolution and ODConv convolution to extract defect features of different sizes. The dark blue area indicates the background area and the yellow area indicates the area of concern of the network about the defect. The location of the defect is marked by the red circle in the figure. The traditional convolutional figure 7(b) limits the ability of convolution to extract weak features, resulting in poor recognition of weak feature images. The ODConv convolutional map 7(c), on the other hand, has a wide coverage in the spatial dimension and is able to extract weak features better.

Table 3. Comparative experiments with the introduction of ODConv layers in different locations of the backbone network

Serial Number	Layer0	Layer1	Layer3	Layer5	Precision(%)	Recall(%)	GFLOPs	mAP@0.5(%)
1					90.0	86.0	28.7	90.3
2	✓				90.3	87.8	28.7	90.0
3	✓	✓			90.9	88.2	28.2	90.4
4		✓	✓		90.4	88.0	27.7	90.1
5	✓	✓	✓		91.2	89.3	27.7	90.7
6	✓	✓	✓	✓	91.9	90.0	26.9	91.0



(a) original figure (b)stage3 Conv_features (c)stage3 ODConv_feature

Fig 7. Scheme number 6 convolutional feature map visualization

2) Comparing the efficiency of various pyramid pooling techniques

Table 4. Improved feature fusion

Feature Fusion Method	Precision(%)	Recall(%)	mAP@0.5(%)	GFLOPs
PANet	93.1	91.5	92.9	27.0
BiFPN	93.5	90.8	92.8	29.1
GSFPN	93.7	91.7	93.0	24.3

This study adds the BiFPN module to the backbone network to perform related experiments to confirm the effectiveness of the GSFPN module. The results in Table 4 demonstrate the impact of utilizing two distinct FPN modules on the model's performance. After adding the GSFPN module to the YOLO-O based model, the precision rate is improved by 0.6%, recall by 0.2%, mAP@0.5 by 0.4%, while the GFLOPs of the network are reduced by 10% compared to the model with PANet feature fusion. The addition of the BiFPN module to the YOLO-O model resulted in a 0.2% decrease in precision, 0.9% decrease in recall, 0.2% decrease in mAP@0.5 and 16.49% decrease in network

GFLOPs compared to the GSFPN module. The findings indicate that the GSFPN module has the capability to enhance the effectiveness of multiscale fusion, leading to an enhancement in the model's detection accuracy. Also, this module is able to reduce the computational and parametric quantities of the network. Therefore, we refer to this improved model as YOLO-OG.

3) Comparing Various Attention Mechanisms

In order to improve the model's ability to accurately detect minor defects. CA[28], SE[26], SA[30], and CBAM [27] to the backbone network are based on the YOLO-OG model. Carry out the related experiments. Table 5 presents a comparison of the effects of four distinct attention modules on the model performance. To achieve the best outcomes in the attention module experiment, the SA module is included and given the name YOLO-OGS based on the thorough evaluation of accuracy, recall, and mAP@0.5. In comparison to the YOLO-OG model, there has been a 0.3% increase in precision rate, a 0.7% improvement in recall rate, and a 0.7% enhancement in mAP@0.5, leading to further improvements in the model's performance.

Table 5. Adding different attention modules

Attention Mechanism	Precision(%)	Recall(%)	mAP@0.5(%)
YOLO-OG	93.7	91.7	93.0
CA	93.4	91.9	93.1
CBAM	93.0	92.0	92.4
SE	93.8	91.2	92.5
SA	94.0	92.4	93.7

4) Ablation experiment

To assess the impact of the enhanced approach on the efficacy of identifying weak feature defects in clamp models, this paper conducted ablation experiments on the upgraded model. Improvements are made by adding multi-dimensional dynamic convolution ODConv, Shuffle Attention module, and GSFPN module to the YOLOv8s model to enhance the model performance. while maintaining the other training parameter configurations in line with the completed model that this study suggests. where "✓" represents that the corresponding improvement strategy is used in the network. As shown in the experimental data in Table 6, Group B introduces the multidimensional convolutional ODConv in the YOLOv8 backbone network. Resulting in a 6.27% decrease in GFLOPs, a 1.9% increase in precision, a 4% increase in recall, and a 1.3% increase in mAP@0.5. It can be seen that ODConv can improve model performance and reduce model complexity in a balanced way, but there is still room for improvement. On top of Group A, Group D has a separate shuffle attention module added to compare with the Group A YOLOv8s model. The results show that its precision, recall, and mAP@0.5 improved by 2.6%, 0.8%, and 2.4%, respectively. Based on the YOLO-O model in group B, group C adds the SA attention module to enhance the feature extraction of the model for weak target detection, which improves the precision rate by 1.4%, recall rate by 0.8%, and mAP@0.5 by 1.4%. Group E introduces the GSFPN module to reduce model complexity while improving performance. The improvement resulted in a 1.6% increase in precision, a 1.7% increase in recall, and a 1.4% increase in mAP@0.5 values, along with a 9% reduction in GFLOPs. recall by 1.7%, mAP@0.5 by 1.4%, and GFLOPs by 9%. After analysis, it was found that adding both the shuffle attention module and the GSFPN module improves the model performance and has less impact on the model complexity. The final improved YOLO-OGS model in Group F outperformed the YOLOv8s To assess the impact of the enhanced approach on the efficacy of identifying weak feature defects in clamp models, this paper conducted ablation experiments on the upgraded model. Improvements are made by adding multi-dimensional dynamic convolution ODConv, Shuffle

Attention module, and GSFPN module to the YOLOv8s model to enhance the model performance. while maintaining the other training parameter configurations in line with the completed model that this study suggests. where "✓" represents that the corresponding improvement strategy is used in the network. As shown in the experimental data in Table 6, Group B introduces the multidimensional convolutional ODConv in the YOLOv8 backbone network. Resulting in a 6.27% decrease in GFLOPs, a 1.9% increase in precision, a 4% increase in recall, and a 1.3% increase in mAP@0.5. It can be seen that ODConv can improve model performance and reduce model complexity in a balanced way, but there is still room for improvement. On top of Group A, Group D has a separate shuffle attention module added to compare with the Group A YOLOv8s model. The results show that its precision, recall, and mAP@0.5 improved by 2.6%, 0.8%, and 2.4%, respectively. Based on the YOLO-O model in group B, group C adds the SA attention module to enhance the feature extraction of the model for weak target detection, which improves the precision rate by 1.4%, recall rate by 0.8%, and mAP@0.5 by 1.4%. Group E introduces the GSFPN module to reduce model complexity while improving performance. The improvement resulted in a 1.6% increase in precision, a 1.7% increase in recall, and a 1.4% increase in mAP@0.5 values, along with a 9% reduction in GFLOPs. recall by 1.7%, mAP@0.5 by 1.4%, and GFLOPs by 9%. After analysis, it was found that adding both the shuffle attention module and the GSFPN module improves the model performance and has less impact on the model complexity. The final improved YOLO-OGS model in Group F outperformed the YOLOv8s model in Group A in terms of precision, recall, and mAP@0.5, with respective improvements of 4%, 6.4%, and 3.4%, while GFLOPs were reduced by 14.4%. The aforementioned results indicate that the YOLO-OGS model demonstrates superior accuracy and faster speed in detecting weak feature defects of clamps. Combining the above studies, it is determined that the final improved model, YOLO-OGS, achieves the combined best improvement.

Table 6. Performance results of ablation experiments

Serial Number	ODConv	SA	GSFPN	GFLOPs	Precision(%)	Recall(%)	mAP@0.5(%)	FPS
A				28.7	90.0	86.0	90.3	115.0
B	✓			26.9	91.9	90.0	91.0	122.3
C	✓	✓		26.9	93.3	91.9	92.6	123.4
D		✓		27.0	92.6	90.8	92.4	109.2
E	✓		✓	24.3	93.7	91.7	93.4	116.2
F	✓	✓	✓	23.8	94.0	92.4	93.7	123.6

5) Results and analysis of the clamp dataset

With the same experimental dataset, experimental hardware and software equipment, and relevant parameters, we conduct comparative experiments with algorithms such as YOLOv3-tiny, YOLO5s, and YOLOv7 in order to understand the gap between the proposed model and mainstream target detection algorithms. Appendix A shows the prediction results of different algorithms for the hoop dataset, and Table 7 compares the differences in performance of these algorithms. As can be seen in Appendix A, YOLOv3-Tiny had one missed detection and one false detection of a point weak feature defect on the first part. The second part had a location on the surface that contained line defects and area defects. YOLOv3-Tiny judged all the defects in that location as area-like defects. Although YOLOv5s was able to detect the faint characteristic defects of the first part, it was not accurate enough to localize the line defects of the second part. YOLOv8s has one missed detection for the weak characteristic defects of the first part. Although YOLOv8s was able to accurately locate the position of line and area defects for the second part, the recognition rate for line defects was low. The method in this paper is more accurate in detecting weak feature defects and can accurately detect and localize

different defects even in the case of dense defects. In Table 7, the final improved YOLO-OGS model in this paper has the highest precision, recall, mAP@0.5. The YOLO-OGS model is a detection algorithm specifically designed for detecting subtle feature defects in the card edge. By increasing the number of network layers and replacing different network modules, the algorithm enables its network architecture to further reduce resource consumption while satisfying high accuracy and low GFLOPs.

Table 7. Performance comparison of different algorithms for the clamp dataset

Model	Precision(%)	Recall(%)	mAP@0.5(%)	GFLOPs
YOLOv3-Tiny	85.7	78.9	81.7	19.1
YOLOv5n	80.0	77.2	78.3	7.1
YOLOv5s	83.3	80.1	85.7	24
YOLOv6	90.1	80.7	85.4	11.8
YOLOv7	82.7	79.6	80.3	105.1
YOLOv8n	91.0	85.3	89.3	8.1
YOLOv8s	90.5	86.0	90.3	28.7
YOLOv-OGS	94.0	92.4	93.7	24.5

6) Results and analysis of the NEU-CLS dataset

Table 8. Performance comparison of different algorithms for the NEU-CLS dataset

Model	Precision(%)	Recall(%)	mAP@0.5(%)	GFLOPs
YOLOv3-Tiny	46.9	50.8	49.0	19.1
YOLOv5n	62.4	61.9	68.1	7.1
YOLOv5s	60.6	68.7	70.2	24
YOLOv6	66.8	59.6	69.1	11.8
YOLOv7	67.2	65.3	68.5	105.1
YOLOv8n	66.7	67.6	70.0	8.1
YOLOv8s	66.6	67.4	70.1	28.7
YOLOv-OGS	68.4	68.5	72.9	24.5

The approach presented in this research is verified and contrasted using the public dataset NEU-CLS [33]at Northeastern University using the algorithms YOLOv5n, YOLOv6, and YOLOv8n. The heat maps of different algorithms are given in Appendix B. The heat maps can visually represent the focus of the model's attention when extracting features, and the brighter the color, the higher the model's attention. From Table 8, it can be seen that YOLOv3-tiny has low detection precision for the NEU-CLS dataset and is not suitable for steel weak feature defect detection. Additionally, the YOLOv3-tiny and YOLOv7 heat maps show how heavily the image background interferes with the model's ability to focus on the poor feature defect region. The mAPs of YOLOv8s and YOLOv5s are 70.1% and 70.2%, respectively. While there is still space for development, Appendix B demonstrates that YOLOv8s and YOLOv5s can reduce some background interference and enhance attention to the weak feature deficiencies. The YOLO-OGS model, on the other hand, has the highest precision, recall,

and mAP@0.5 among the compared models, and it can better suppress the background interference in the Heatmap and maximize the model's focus on weak feature defects.

5. CONCLUSION

To solve the problem of misdetection and omission detection of weak feature defects in aviation clamps. In the paper, an algorithm YOLO-OGS is proposed based on the improved YOLOv8s aviation clamp weak feature defect detection algorithm. The multidimensional dynamic convolution, or ODConv, improves the network model's basic convolution operation's feature extraction capability. It enriches the scene-capturing ability while reducing the computational parameters. Second, to decrease model complexity and increase feature fusion efficiency at every level, the GSFPN paradigm network structure is introduced. To enhance the network's capacity to identify weak defect features and to enrich the knowledge of weak defects, the shuffle attention module is implemented. The final improved YOLO-OGS network improves accuracy, recall, mAP@0.5, and FPS by 4%, 6.4%, 3.4%, and 6.95%, respectively, and reduces model GFLOPs by 14.4% when compared to YOLOv8s in the target detection task. The findings demonstrate that, without compromising inference speed, the final enhanced model in this work lowers model complexity and enhances algorithmic detection performance. It satisfies aircraft clamps' detection requirements.

Our core work in this study is to solve the problem of misdetection and omission of weakly characterized defects in aviation clamps. However, the proposed improvement strategy has general application value and broad prospects in cross-domain inspection applications. These strategies are not only effective in the detection of weak feature defects on the surface of clamps, but the core ideas and techniques behind them are also used with other small target detection scenarios (for example, in the areas of identification of partial disease symptoms in medical imaging diagnosis, detection of defects in electronic components, and detection of defects in engineered structural components, etc.).

Although the work in this paper has made some progress in detecting weak feature defects in clamps, there are some limitations. The proportion of weak feature defects in the clamp dataset is relatively small, and obtaining enough pictures of clamp defects and labeling their defects in practical applications requires high labor and time costs. In future research, we will try to apply semi-supervision to the field of clamp defect detection. Because the surface defects of clamps are of many types and shapes, it is a challenge to learn effective features in the absence of labels. So semi-supervised learning is chosen to meet this challenge, which can reduce the dependence on labeled data and save the time and cost of manual labeling. On the other hand training with labeled and unlabeled data can improve the generalization performance of the model for uncommon or unencountered defect types. This is of research interest for scenarios in an industry where labeling resources are limited and defect categories are constantly changing.

REFERENCES

[1] M. Ghobakhloo, "Industry 4.0, digitization, and opportunities for sustainability," Journal of cleaner production, vol. 252, pp. 119869, Apr.2020.

[2] G. Vial, "Understanding digital transformation: A review and a research agenda," Managing digital transformation, vol. 28, no. 3, pp. 13-66, Jun.2021.

[3] G. Li, C. Wang, D. Zhang, and G. Yang, "An improved feature selection method based on random forest algorithm for wind turbine condition monitoring," Sensors, vol. 21, no. 16, pp. 5654, Aug.2021.

[4] R. Shanmugamani, M. Sadique, and B. Ramamoorthy, "Detection and classification of surface defects of gun barrels using computer vision and machine learning," Measurement, vol. 60, pp. 222-230, Jan.2015.

[5] J.-K. Park, B.-K. Kwon, J.-H. Park, and D.-J. Kang, "Machine learning-based imaging system for surface defect inspection," International Journal of Precision Engineering and Manufacturing-Green Technology, vol. 3, pp. 303-310, Jul.2016.

[6] Z. Zhong and Y. Hu, "A fast oxidation region detection algorithm based on differential geometry approach for high-density flexible integrated circuit packaging substrates," Measurement Science and Technology, vol. 29, no. 10, pp. 105401, Aug.2018.

[7] C. Jian, J. Gao, and Y. Ao, "Automatic surface defect detection for mobile phone screen glass based on machine vision," Applied Soft Computing, vol. 52, pp. 348-358, Mar.2017.

[8] C. Xu, L. Li, J. Li, and C. Wen, "Surface defects detection and identification of lithium battery pole piece based on multi-feature fusion and PSO-SVM," Ieee Access, vol. 9, pp. 85232-85239, Mar.2021.

[9] J. Guo, Y. Yang, X. Xiong, Y. Yang, and M. Shao, "Brake disc positioning and defect detection method based on improved Canny operator," IET Image Processing, vol. 18, no. 5, pp. 1283-1295, Apr.2024.

[10] Y. Xu, K. Zhang, and L. Wang, "Metal surface defect detection using modified YOLO," Algorithms, vol. 14, no. 9, pp. 257, Aug.2021.

[11] Z. Xu, Z. Wu, and W. Fan, "Improved SSD-assisted algorithm for surface defect detection of electromagnetic luminescence," Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, vol. 235, no. 5, pp. 761-768, Feb,2021.

[12] Y. J. Cha, W. Choi, G. Suh, S. Mahmoudkhani, and O. Büyüköztürk, "Autonomous structural visual inspection using region-based deep learning for detecting multiple damage types," Computer-Aided Civil and Infrastructure Engineering, vol. 33, no. 9, pp. 731-747, Sept.2018.

[13] X. Ye, J. Ye, Z. He, D. Zhang, X. Hu, and Q. Chen, "A Novel Defect Detection Method for Ferrite Shield Surface Defects by Improved Faster R-CNN," Scientific Programming, vol. 2022, Nov.2022.

[14] X. Wang, H. Gao, Z. Jia, and Z. Li, "BL-YOLOv8: An improved road defect detection model based on YOLOv8," Sensors, vol. 23, no. 20, pp. 8361, Oct.2023.

[15] Y. Zhang, S. Shen, and S. Xu, "Strip steel surface defect detection based on lightweight YOLOv5," Frontiers in Neurorobotics, vol. 17, pp. 1263739, Oct.2023.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Jul.2023, pp. 7464-7475.

[17] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," arXiv preprint arXiv:2004.10934, Apr.2020.

[18] R. Sunkara and T. Luo, "No more strided convolutions or pooling: A new CNN building block for low-resolution images and small objects," in Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Aug.2022: Springer, pp. 443-459.

[19] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Scaled-yolov4: Scaling cross stage partial network," in Proceedings of the IEEE/cvf conference on computer vision and pattern recognition, Nov.2021, pp. 13029-13038.

[20] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, Jun.2018, pp. 7794-7803.

[21] Y. Chen, X. Dai, M. Liu, D. Chen, L. Yuan, and Z. Liu, "Dynamic convolution: Attention over convolution kernels," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Mar.2020, pp. 11030-11039.

[22] B. Yang, G. Bender, Q. V. Le, and J. Ngiam, "Condconv: Conditionally parameterized convolutions for efficient inference," Advances in neural information processing systems, vol. 32, Apr.2019.

[23] C. Li, A. Zhou, and A. Yao, "Omni-dimensional dynamic convolution," arXiv preprint arXiv:2209.07947, Sept.2022.

[24] X. Lin, L. Ma, W. Liu, and S.-F. Chang, "Context-gated convolution," in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVIII 16, 2020: Springer, pp. 701-718.

[25] H. Li, J. Li, H. Wei, Z. Liu, Z. Zhan, and Q. Ren, "Slim-neck by GSConv: A better design paradigm of detector architectures for autonomous vehicles," arXiv preprint arXiv:2206.02424, Aug.2022.

[26] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in Proceedings of the IEEE conference on computer vision and pattern recognition, Jun.2018, pp. 7132-7141.

[27] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in Proceedings of the European conference on computer vision (ECCV), Jul.2018, pp. 3-19.

[28] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, Mar.2021, pp. 13713-13722.

[29] T. Yu, X. Li, Y. Cai, M. Sun, and P. Li, "S^2-MLPv2: improved spatial-shift MLP architecture for vision," arXiv preprint arXiv:2108.01072, Aug.2021.

[30] Y. Yuan, L. Huang, J. Guo, C. Zhang, X. Chen, and J. Wang, "Ocnet: Object context network for scene parsing," arXiv preprint arXiv:1809.00916, Sept.2018.

[31] Q.-L. Zhang and Y.-B. Yang, "Sa-net: Shuffle attention for deep convolutional neural networks," in ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Jun.2021: IEEE, pp. 2235-2239.

[32] M. Lin, Q. Chen, and S. Yan, "Network in network," arXiv preprint arXiv:1312.4400, Dec.2013.

[33] Q. Luo, X. Fang, J. Su, J. Zhou, B. Zhou, C. Yang et al., "Automated visual defect classification for flat steel surface: a survey," IEEE Transactions on Instrumentation and Measurement, vol. 69, no. 12, pp. 9329-9349, Dec.2020.