J. Meas. Sci. Instrum., 2026, 17(1): 151-161. http://jmsi.publish.founderss.cn/
DOI: 10.62756/jmsi.1674-8042.2026013 jmsi@nuc.edu.cn

CSYOLO: a YOLOv8-based PCB defect detection model integrating composite backbone networks and dynamic snake convolution

LIU Chunjuan[1], ZHANG Mingxuan[1], YAN Haowen[2], WU Xiaosuo[1*], WANG Yixiang[1]

1. School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China;
2. School of Surveying, Mapping and Geographic Information, Lanzhou Jiaotong University, Lanzhou 730070, China

***Corresponding author:** WU Xiaosuo (wuxs_laser@lzjtu.edu.cn)

Received: January 19, 2025 **Revised:** March 22, 2025 **Accepted:** March 24, 2025

Abstract: An improved CSYOLOv8 model based on YOLOv8 model is developed specifically for identifying defects in printed circuit board (PCB). Firstly, a composite backbone network is designed to carry out additional feature extraction, which enriches the expression ability of features and enhances the detection accuracy of the model. Secondly, a YOLO-FPN (Feature pyramid network) structure is designed to supplant the original neck network, which enhances the feature fusion ability of the model and improves the detection accuracy of small target objects. Furthermore, to enhance the model's capability to extract tubular features, dynamic snake convolution is implemented. Finally, MPDIoU loss function is employed to enhance both the convergence rate and the precision of the model. Experiments show that the *mAP* of the improved model on the PCB defect dataset reaches 96.6%, which is 4.5% higher than that of the YOLOv8 model, and the number of parameters is only 3 256 862, and the average detection speed is 51.8 frames per second, which meets the requirements of detection accuracy and efficiency.
Key words: printed circuit board (PCB); deep learning; defect detection; YOLOv8; multi-scale feature fusion; loss function

0 Introduction

As the electronics manufacturing industry advances rapidly, electronic products are increasingly designed to be multifunctional, intelligent, and compact. Printed circuit boards (PCBs), being essential precision components in these products, significantly influence their overall performance. Consequently, the technology for detecting defects in PCBs has become crucial in today's electronics industry[1].

In recent years, advancements in deep learning have led to notable progress in target detection algorithms. Among the notable two-stage detection algorithms, Fast region-based convolutional neural network (Fast R-CNN) [2] and Faster R-CNN[3] stand out, while examples of single-stage detection algorithms include single shot multibox detector (SSD) [4] and you only look once (YOLO) [5-9], among others. These algorithms have been widely applied in various fields, including PCB defect detection. To tackle the difficulties associated with PCB defect detection, Ding et al. [10] proposed a tiny defect detection network (TDD-Net). The network was built on Faster R-CNN architecture and employed ResNet-101 as its backbone. It integrated feature pyramid networks[11] (FPN) for multi-scale feature fusion and makes use of "anchor" boxes of different sizes (152, 252, 402, 602, 802) for detection, yielding strong performance on a public PCB defect dataset. Shi et al. [12] enhanced the SSD algorithm by incorporating attention mechanisms and multi-layer feature fusion, which led to improved detection accuracy. The efficacy of this enhanced algorithm was demonstrated on both the PCB defect dataset and the Pascal VOC2007 dataset.

The YOLO algorithms, along with their enhanced variants, have yielded substantial advancements in the realm of object detection and have found extensive application in the detection of PCB defects[13]. For instance, Bian et al. [14] introduced the AT-YOLO algorithm, which built on YOLOv3 by substituting the backbone network with ResNeSt50[15] and incorporating a spatial pyramid pooling (SPP) module to combine features from various receptive fields. They also applied the *K*-means algorithm for re-clustering and generating anchor boxes, improving object detection accuracy. Tang et al. [16] enhanced YOLOv5 by adding a layer specifically for small object detection and incorporating a Transformer into the backbone network. This

integration helped minimize interference between the foreground and background, thereby boosting the network's analytical performance. Despite the good results achieved by these algorithms in PCB defect detection, there are still issues such as high computational costs and slow detection speeds.

To tackle these issues, we introduce an enhanced CSYOLO defect detection model derived from the YOLOv8 detection network. The model significantly improves the accuracy of PCB defect detection while not significantly increasing computational resources. The main improvements include: 1) Designing a composite backbone network that fuses the features extracted from the lead backbone and assistant backbone through multiple composite blocks, thereby enriching the network's feature expression capabilities and enhancing defect detection accuracy; 2) Developing a YOLO-FPN feature fusion network inspired by BiFPN[17] to establish cross-scale connections and execute weighted feature fusion. This approach aims to harness contextual information to enhance the model's feature integration capabilities; 3) Introducing dynamic snake convolution (DSC) [18] to enhance the extraction of tubular features; 4) Adopting a more efficient loss function, MPDIoU[19], to enhance the model's convergence speed and accuracy.

1 CSYOLOv8 model

The improved model CSYOLOv8 based on YOLOv8 mainly consists of four modules: input, composite backbone network, YOLO-FPN, and head. The overall structure is shown in Fig.1, where the improved blocks are highlighted by the red area, while the other unchanged YOLOv8 blocks are represented by the gray area.

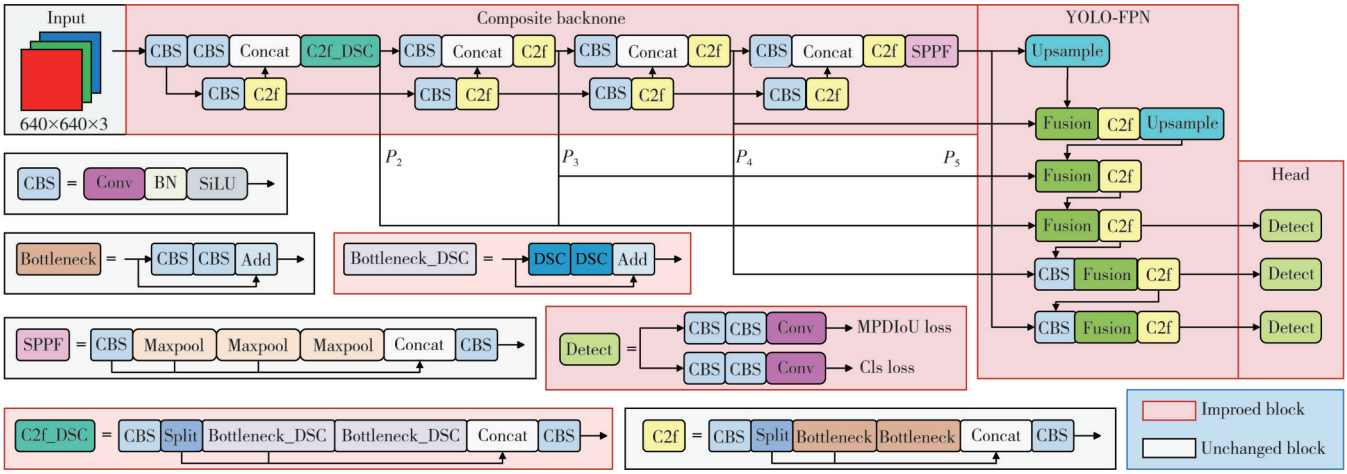


Fig. 1 Structure of CSYOLOv8 model

The input adopts the Mosaic data augmentation method, which significantly enhances the training dataset and boosts the network's generalization capabilities and robustness. The composite backbone is responsible for extracting defect features from the input PCB images and generating four feature maps of different scales (P_2, P_3, P_4, P_5), namely P_n, $n \in \{2, 3, 4, 5\}$, where P_n represents the size of the feature map being 2^{-n} times the size of the input image. The YOLO-FPN structure fully utilizes the multi-scale features extracted by the backbone network, enhancing the network's semantic expression and localization ability at different scales. Compared to the traditional coupled head design, CSYOLOv8 employs a decoupled head architecture, enabling the regression and classification branches to be processed separately. This approach not only significantly reduces the network's parameter count and computational demands but also enhances detection accuracy.

1.1 Improvement of backbone network

1.1.1 Composite backbone

The performance of modern target detectors depends heavily on the backbone networks. In general, the more the representative features are extracted from the backbone network, the better the performance of the detector is. To achieve higher accuracy, mainstream detectors tend to adopt wider and deeper backbone networks. But wider and deeper backbone networks may increase the complexity of the model, thus requiring large computational resources. To balance the performance and the model complexity, we design two different composite backbone networks, as shown in Fig.2.

The network cascades multiple identical composite blocks and utilizes the C2f module to further extract and transform the features extracted by the composite blocks, thereby generating outputs with stronger representational capabilities.

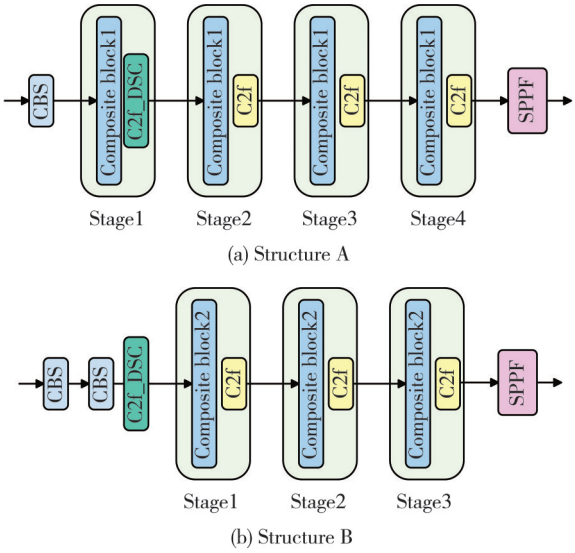
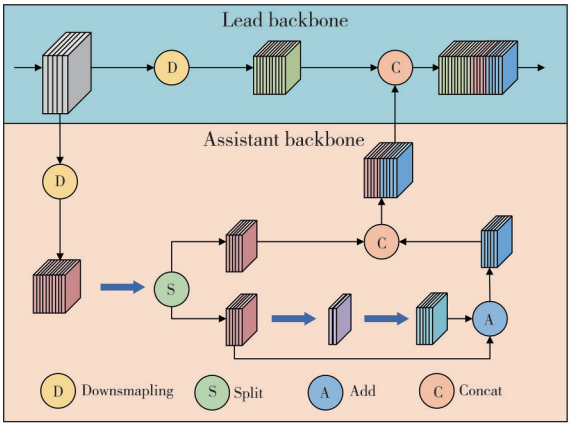


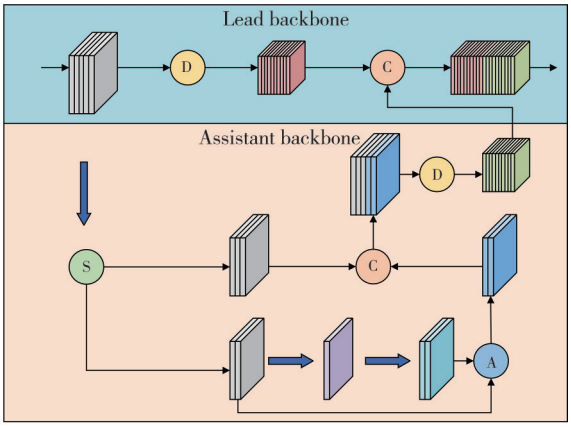
Fig. 2 Two structures of composite backbone network

The composite backbone network integrates the high-level and low-level features of two different backbone networks, enabling this network to perform PCB defect detection tasks more effectively. Specifically, structure A consists of four stages, each of which includes a composite module and a C2f module, where the C2f module in stage1 specifically introduces dynamic snake convolution to enhance the extraction effect of tubular features in PCB defects. In contrast, structure B is designed more succinctly with only three stages. To ensure the consistency of the output feature map size, structure B adds a convolutional module before stage1 and continues to use dynamic snake convolution in the C2f module.

Two different composite blocks are shown in Fig. 3. Each composite module consists of a lead backbone and an assistant backbone. The lead Backbone includes downsampling and concatenation operations, which are used to extract features and adjust the size of the feature maps.



(a) Composite block1



(b) Composite block2

Fig. 3 Structures of composite blocks

The assistant backbone is composed of two different structures: the first structure first performs downsampling to match the size of the feature maps with those of the lead backbone and increases the number of channels of the feature maps. Then, the feature maps are split into two parts, one of which is directly used for subsequent concatenation, while the other part undergoes convolution operations to extract deeper features. By element-wise addition, these features at different levels are fused, thus improving the efficiency of gradient propagation and information flow. Finally, the fused features are merged with the feature maps from the lead backbone through channel concatenation. The second structure first performs channel splitting to process features at different levels, then executes downsampling operations, and finally concatenates with the feature maps generated by the lead backbone.

Since the lead backbone and the assistant backbone networks jointly participate in feature extraction, even if one backbone network has deficiencies in some feature extraction, the other backbone network can supplement its shortcomings, so as to improve the robustness of the overall feature extraction. At the same time, using a composite backbone can effectively reduce the design and training costs required for designing new backbone networks, and while effectively controlling the complexity of the model, it improves the performance of the detection model.

1.1.2 Dynamic snake convolution

The characteristics of tubular targets are typically manifested as slender and complex, where standard convolutions and dilated convolutions cannot adaptively adjust their receptive fields to match the morphology of the target. In contrast, deformable convolutions can adjust the shape of the region of interest through adaptive learning, however, for tubular targets, deformable

convolutions cannot effectively restrict the connectivity of the receptive field. Dynamic snake convolution adopts an iterative strategy on the basis of introducing the deformation offset to ensure the continuity of the perception area during the deformation process and avoid the perception area deviating from the target due to excessive deformation offset. This continuity helps better capture the complete structural information of the target. Given the complexity of PCB defects, especially for the detection of slender defects (such as open circuit, short circuits, and spurious copper), we introduce dynamic snake convolutions into the first C2f module of the proposed CSYOLOv8 model to improve the detection effect of these slender defects.

The dynamic snake convolution process is as follows:

Given the center coordinates of the standard convolution kernel as $K_i = (x_i, y_i)$, a 3×3 convolution kernel is represented as

$$K=\{(x-1,y-1),(x-1,y),\cdots,(x+1,y+1)\}. \tag{1}$$

To endow the convolutional kernel with greater flexibility, enabling it to effectively focus on the complex geometric features of PCB defect targets, the dynamic snake convolution introduces a deformation offset Δ. The offset is obtained by applying a 3×3 convolution on the input feature map to obtain an offset map, and then limiting the offset to $[-1,1]$ by the tanh function, so that the final offset $\Delta\in[-1,1]$ is obtained.

Taking the transformation along the x-axis as an example, for each position $K_{i\pm c}$ within the convolutional kernel, $c\in\{1,2,3,4\}$ represents the horizontal distance from the central grid. The offset is obtained through iterative calculations. Starting from the central point K_i, the offset for each position is accumulated based on the offset of the previous position. This process ensures the continuity of the convolutional kernel and its adaptability to tubular structures. The offset is represented as

$$\Delta y=\sum_{i\pm c}\Delta y. \tag{2}$$

Updating the coordinates of each position in the kernel based on the calculated offset by

$$K_{i\pm c}=(x_i\pm c, y_i+\Delta y). \tag{3}$$

Similarly, in the y-axis direction, the offset and coordinate changes are represented as

$$\Delta x=\sum_{j\pm c}\Delta x, \tag{4}$$

$$K_{j\pm c}=(x_j+\Delta x, y_j\pm c). \tag{5}$$

Since offsets are usually fractional, bilinear interpolation is used to calculate the values of the convolution kernel at integer positions. Bilinear interpolation is represented as

$$\begin{aligned}I(x,y)=&(y_1-y)(x_1-x)\cdot I(x_0,y_0)+\\&(y_1-y)(x-x_0)\cdot I(x_1,y_0)+(y-y_0)(x_1-x)\cdot\\&I(x_0,y_1)+(y-y_0)(x-x_0)\cdot I(x_1,y_1),\end{aligned} \tag{6}$$

where (x_0, y_0), (x_1, y_0), (x_0, y_1), (x_1, y_1) are the four nearest integer coordinate points around the non-integer coordinate point (x,y).

Due to the stretching deformation of the dynamic snake convolution in both the horizontal and vertical directions, it is capable of covering a 9×9 area. This snake convolution based on dynamic structures exhibits higher adaptability when dealing with elongated tubular structures, enabling it to perceive key features more accurately, thereby enhancing the feature extraction capability for slender defects. Especially in PCB defect detection, dynamic snake convolution can effectively improve the extraction accuracy of elongated structural defects, thus increasing the overall detection accuracy.

1.2 YOLO-FPN feature fusion module

The detection of small objects has long posed a significant challenge in the realm of PCB defect detection. With the deepening of network layers, the features of large objects can be well preserved, while the features of small objects often gradually disappear after multiple convolutions. To address this issue, the introduction of structures like FPN and PANet[20] has improved the detection of small objects to some extent, as shown in Fig.4(a) and (b). However, the top-down feature fusion method in FPN is limited by a unidirectional information flow, and the improvement in accuracy is relatively limited. To further improve detection accuracy, PANet integrates a bottom-up information pathway in addition to FPN, enabling the feature maps to incorporate more comprehensive semantic and positional information. However, this improvement also results in an increase in the number of parameters. BiFPN, inspired by PANet, combines more features at a lower computational cost through weighted bidirectional cross-scale connections, as shown in Fig.4(c). Inspired by this, we propose a new feature extraction module, YOLO-FPN, under the BiFPN framework, by adding the input and fusion paths of shallow feature maps to better adapt to the output of composite backbone networks and further improve the effect of feature fusion. Its specific structure is shown in Fig.4(d).

In the YOLO-FPN fusion process, initially, the four different scale PCB feature maps P_2, P_3, P_4, P_5 generated by the composite backbone network need to

undergo 1×1 convolutions to adapt the number of channels, resulting in P_{2IN}, P_{3IN}, P_{4IN}, P_{5IN}. Then, P_{5IN} is upsampled and fused with P_{4IN} to obtain P_{4TD}. Next, P_{4TD} is upsampled and fused with P_{3IN} to obtain P_{3TD}, and then P_{3TD} is fused with P_{2IN} to get P_{3OUT}. After that, P_{3OUT} is downsampled and fused with P_{4IN} and P_{4TD} to obtain P_{4OUT}. Finally, P_{4OUT} is downsampled and fused with P_{5IN} to get P_{OUT}.

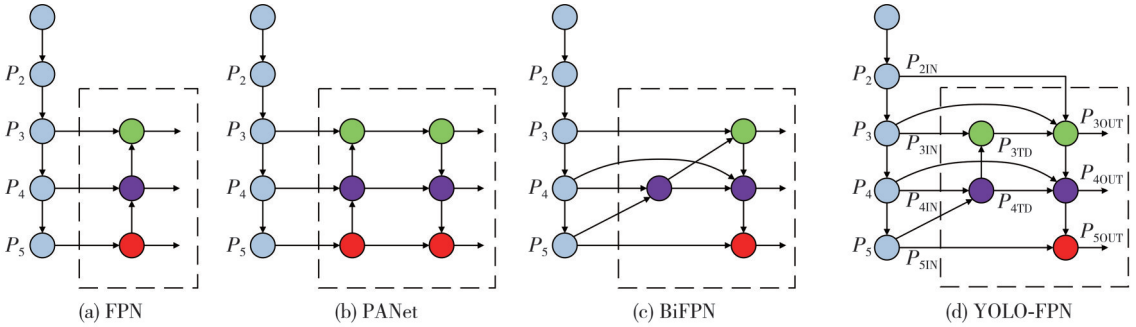


Fig. 4 Comparison of feature pyramid networks

In the course of the aforementioned procedure, the fusion and transference of PCB feature maps across various scales are accomplished, thereby enhancing the model's precision and robustness in detecting targets of varying sizes.

However, the four different scale PCB feature maps P_2, P_3, P_4, P_5 have different resolutions, and their contributions to feature fusion are different. To effectively learn the importance of various input features, it is essential to differentially integrate these features. To this end, we adopt a simple and efficient fast normalization fusion method—fusion, as shown below

$$O=\sum_i \frac{v_i}{\varepsilon+\sum_j v_j} I_i, \tag{7}$$

where v_i denotes a learnable weight corresponding to the ith input feature (i=1, 2, ⋯). To guarantee non-negative weights, a ReLU activation function is applied after v_i, ensuring $v_i>0$ and preventing the complete suppression of any input feature. The term $\sum_j v_j$ computes the sum of all learnable weights (j=1, 2, ⋯), which serves as the denominator for weight normalization. Ii represents the ith input feature map, which typically originates from different scales or branches within the network. ε is a small constant (e.g., 0.000 1) introduced to avoid numerical instability caused by division by zero.

Through the bidirectional (top-down and bottom-up) paths and weighting mechanism, YOLO-FPN can better aggregate feature information at different scales, highlight important features, and improve the accuracy and robustness of feature representation. At the same time, the network has a simple structure and efficient calculation, is easy to implement and optimize, and can better process the feature information from the composite backbone network, thereby improving the accuracy of PCB defect detection.

1.3 MPDIoU loss function

In PCB defect detection applications, most defects have small areas, which poses a significant challenge for detection algorithms when locating defects. To enhance the training effectiveness of bounding box regression, and to boost convergence speed and regression precision, we introduce a new intersection of union (IoU)-based metric method—MPDIoU. This approach enhances the regression process by directly minimizing the distance between the top-left and bottom-right points of the predicted bounding box and those of the actual bounding box.

The calculation for MPDIoU loss function is given as

$$B_{prd}=(x_1^{prd},y_1^{prd},x_2^{prd},y_2^{prd}), \tag{8}$$

$$B_{gt}=(x_1^{gt},y_1^{gt},x_2^{gt},y_2^{gt}), \tag{9}$$

where B_{prd} denotes the predicted bounding box, with (x_1^{prd}, y_1^{prd}) and (x_2^{prd},y_2^{prd}) representing the coordinates of the top-left and bottom-right points of the predicted bounding box, respectively. Conversely, B_{gt} signifies the ground truth bounding box, where (x_1^{gt},y_1^{gt}) and (x_2^{gt},y_2^{gt}) correspond to the coordinates of the top-left and bottom-right points of the ground truth bounding box, respectively. Assumming that d_1 represents the distance separating the top-left points of the predicted and ground truth bounding boxes and d_2 signifies the distance between their bottom-right points, they can be calculated by

$$d_1^2=(x_1^{prd}-x_1^{gt})^2+(y_1^{prd}-y_1^{gt})^2, \tag{10}$$

$$d_2^2=(x_2^{prd}-x_2^{gt})^2+(y_2^{prd}-y_2^{gt})^2. \tag{11}$$

Thus, the MPDIoU loss function is expressed as

$$L_{MPDIoU}=1-MPDIoU, \tag{12}$$

$$MPDIoU=IoU-\frac{d_1^2}{w^2+h^2}-\frac{d_2^2}{w^2+h^2}, \tag{13}$$

where w and h denote the width and height of the input image, respectively.

It is evident that the MPDIoU loss function takes into account the minimum point distance between any two bounding boxes, thereby enhancing accuracy and accelerating convergence in the regression of bounding boxes. This point-distance-based calculation method is more concise and more efficient than some complex geometric calculations, reducing the computational complexity and improving the computational speed. While retaining the advantage of the existing iou-based indicators. When being applied to PCB defect detection, it can markedly enhance the precision of detection and accelerate the convergence speed of the model.

2 Experiment and analysis

2.1 Dataset and experimental environment

The dataset employed in this experiment is the PCB defect dataset, provided by the Intelligent Robot Open Laboratory at Peking University. As shown in Fig.5, this dataset includes six types of PCB defects: "missing hole", "mouse bite", "open circuit", "short circuit", "spur", and "spurious copper", with a total of 693 images. Due to the small number of samples, the dataset was augmented by adjusting brightness, flipping, rotating, scaling, and cropping, resulting in an augmented dataset of 4 158 images.The experimental environment in this study is detailed in Table 1.

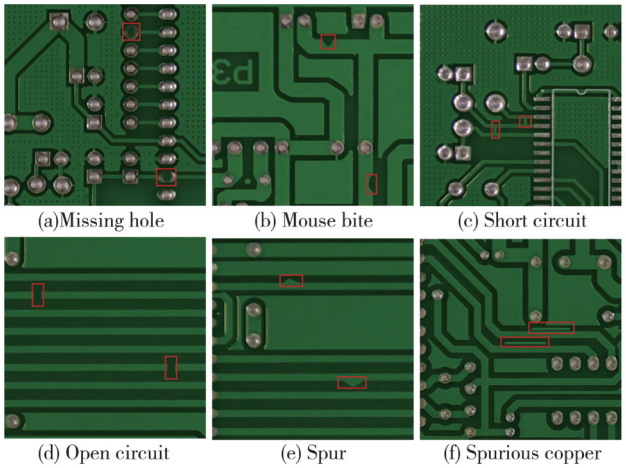


Fig. 5 Six types of defects

Table1 Experimental environment

Designation	Version
CPU	Intel(R) Xeon(R) Platinum 8255C CPU @ 2.50GHz
GPU	NVIDIA GeForce RTX 3080
Python	3.8
Deep learning framework	Pytorch 1.8.1 CUDA 11.1

2.2 Experimental parameter settings

In the training process, the input image dimensions were established at 640×640, and the dataset was partitioned into three subsets: training, validation, and test sets, in the ratio of 7∶1∶2. The experiment employed the stochastic gradient descent (SGD) algorithm for parameter optimization, initializing the learning rate at 0.01, momentum at 0.937, weight decay at 0.000 5, batch size at 16, and setting the total number of training epochs to 250. After training process, the optimal weight file and the last weight file were saved.

2.3 Evaluation metrics

We employed precision (P), recall (R), and mean average precision (mAP) as primary indicators to evaluate the detection accuracy of models, offering a comprehensive assessment of their performance. Among them, mAP represents the average precision at an intersection of union (IoU) threshold of 0.5. Additionally, the detection speed of the model was assessed by the number of image detection frames per second. Floating-point operations (FLOPs) were employed to measure the computational complexity of the model, whereas the total count of parameters was used to evaluate the model's overall complexity.

$$P=\frac{TP}{TP+FP}, \tag{14}$$

$$R=\frac{TP}{TP+FN}, \tag{15}$$

$$mAP=\frac{\sum_{i=1}^{n}AP(i)}{n}, \tag{16}$$

$$AP=\int_0^1 P(R)\mathrm{d}R=\sum_{k=1}^{N}P(k)\Delta R(k), \tag{17}$$

where P measures the accuracy of detection, whereas R indicates the rate of missed detection. In this context, TP signifies true positives, FP represents false positives, FN denotes false negatives, N is the total number of predition samples, and n denotes the total number of defect categories.

2.4 Comparison of experimental results

2.4.1 Module optimization experiment

As shown in Table 2, the design of structure is more concise, thus its parameters and GFLOPs are both lower than those of structure A, but its accuracy decreases. Among the different fusion modules, bottleneck structure is relatively simple and fails to fully process the fusion of feature maps of different sizes. Therefore, as an assistant backbone fusion module, its

contribution to the overall model accuracy is limited. In contrast, the C2f module is capable of capturing more comprehensive gradient flow information and improving the performance of the model without compromising on its lightweight nature. The C2f_Faster module, which is an improvement based on the C2f module, further diminishes the computational burden and complexity of the model. However, it does experience a marginal reduction in accuracy when contrasted with C2f.

Table 2 Results of different composite backbone experiments

Module	Structure	Number of Parameters	FLOPs/G	*mAP*/%
Bottleneck	A	4 265 458	10.9	94.4
	B	3 772 370	10.6	93.4
C2f	A	4 194 690	11.0	95.3
	B	3 754 626	10.7	94.1
C2f_Faster	A	3 833 490	9.9	95.0
	B	3 620 098	9.7	93.4

Since the CSYOLOv8 backbone network contains four C2f modules, we design a set of experiments aimed at exploring the impact of the addition position of DSC on the *mAP*. In the experiments, the position and number of DSCs were altered, and the experimental results are shown in Table 3, where "√" indicates the use of DSC at that position. By comparing the experimental results, it can be found that as the position of DSC addition becomes deeper and the number increases, the number of model parameters correspondingly increases. Specifically, adding DSC to the first C2f module significantly improves the *mAP*, reaching a value of 96.6%. At this point, although the changes to the network structure are minor, with only 3 256 862 parameters, the improvement in accuracy is the most significant. Therefore, DSC is chosen to be added to the first C2f module in our work as the final experimental scheme.

Table 3 Effect of DSC on experimental results

DSC location				*mAP*/%	Number of parameters
First C2f	Second C2f	Third C2f	Fourth C2f		
√				96.6	3 256 862
	√			94.9	3 289 414
		√		96.3	3 389 062
			√	95.2	3 511 906
		√	√	95.6	3 649 834
	√	√	√	95.8	3 688 114
√	√	√	√	95.9	3 693 862

2.4.2 Ablation experiment

To verify the impact of various improved modules on the model, a set of ablation experiments were conducted based on YOLOv8n as the baseline model. The specific results of the ablation experiments are shown in Table 4, where "√" indicates the use of that module.

Table 4 Results of ablation experiments

Model	Cb	YOLO-FPN	MPDIoU	DSC	Number of parameters	FLOPs/G	*P*/%	*R*/%	*mAP*/%	Frame rate/ frame per scond
Baseline					3 012 018	8.2	93.5	85.2	92.1	113.6
2	√				4 194 690	11.0	95.6	88.3	95.3	84.7
3		√			1 992 494	7.1	94.5	87.2	93.1	104.1
4			√		3 012 018	8.2	93.9	89.0	93.5	103.1
5				√	3 012 566	8.3	94.7	89.2	93.9	75.8
6	√	√			3 251 134	9.9	94.3	90.7	95.7	78.7
7	√	√	√		3 251 134	9.9	94.2	91.6	96.0	76.3
Ours	√	√	√	√	3 259 862	10.2	96.3	90.5	96.6	51.8

It can be seen that the proposed modules in this study exhibit superior performance on the PCB defect detection dataset. Among them, the composite backbone network contributes the most. By integrating the high-level and low-level features of two backbone networks, the improved model can more comprehensively capture various features of the target objects, thereby enhancing the richness and accuracy of feature representation. Although the addition of the assistant backbone increases the computational load and complexity of the model, the precision (*P*), recall (*R*), and mean average precision (*mAP*) are improved by 2.1%, 3.1%, and 3.2% respectively compared to those of the baseline model. This indicates that the composite backbone network plays a significant role in improving the accuracy of the model.

By replacing the neck network in the baseline model with YOLO-FPN, not only is the *mAP* increased by 1.0%, but the number of model parameters is also significantly reduced, from 3 012 018 to 1 992 494, and the number of floating-point operations is reduced from 8.2 G to 7.1 G. Therefore, the YOLO-FPN feature fusion module simplifies the network structure and reduces unnecessary computations, thereby lowering the model's complexity and computational cost. This allows the model to maintain high performance while adapting to various resource constraints.

Fig.6 shows the training results for the box loss. It can be observed that the model loss curve using the MPDIoU loss function drops faster and the curve is smoother. Furthermore, it can be seen from Table 4 that the model using the MPDIoU loss function has higher accuracy compared to the baseline model. This suggests

that the MPDIoU loss function is able to effectively reduce training time and improve convergence efficiency without sacrificing accuracy.

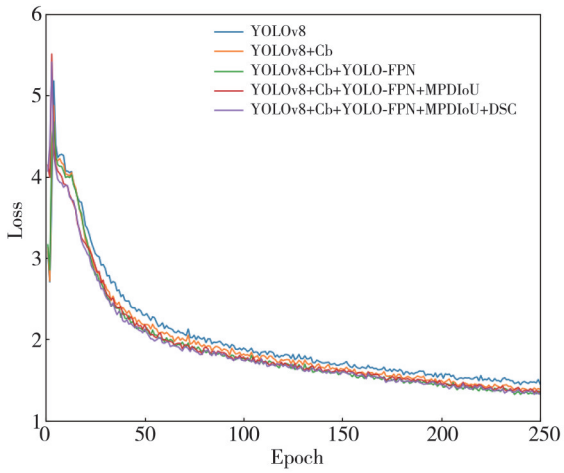
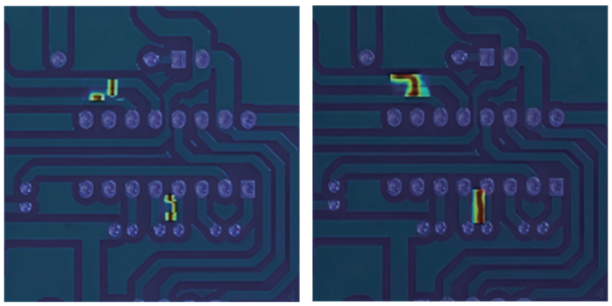


Fig. 6 Training results for box loss

As shown in Fig. 7, we demonstrate the different processing results of the C2f module and the C2f_DSC module when facing the same defect. With the introduction of DSC in the C2f module, the model can more effectively distinguish the key features of PCB defects from the background and maintain the continuity of perception, making it easier to extract elongated tubular features.



(a) C2f (b) C2f_DSC

Fig. 7 Heatmap visualization

With this improvement, the model significantly enhances the detection accuracy in PCB defect detection while maintaining almost the same number of parameters, resulting in an increase of 1.2%, 4.0%, and 1.8% in precision (P), recall (R), and mean average precision (mAP), respectively. As the various improved modules are gradually stacked, the average precision of the improved model in PCB defect detection increases significantly from 92.1% to 96.6%. In summary, through ablation experiments, the improved model is comprehensively verified. The results indicate that the improved CSYOLOv8 can significantly enhance the accuracy of PCB defect detection. Moreover, it maintains real-time performance, striking a favorable equilibrium between detection accuracy and operational efficiency.

Table 5 shows the impacts of the model proposed in this study against the YOLOv8 baseline model across six categories of PCB defect detection. The experimental results of AP across various defect categories reveal that the model presented in this paper outperforms the YOLOv8 baseline model in detecting all six categories. especially in the detection of "mouse bite", "open circuit", "short circuit", and "spur", with improvements of 8.9%, 4.2%, 3.6%, and 6.9%, respectively.

Table 5 *AP* with different defect categories

Defect category	*AP*/%		Improvement/%
	YOLOv8	Ours	
Missing hole	99.3	99.5	0.2
Mouse bite	86.8	95.7	8.9
Open circuit	92.0	96.2	4.2
Short circuit	93.3	96.9	3.6
Spur	87.4	94.3	6.9
Spurious copper	96.7	97.0	0.3

Particularly, as shown in the detection results of Fig. 8, the improved model not only effectively reduces the issues of missed and false detection of open circuits and spurious copper defects in the YOLOv8 baseline model, but also improves the problem of misjudging short circuits as spurs in short circuit defect detection. In addition, the improved model also enhances the overall detection accuracy.

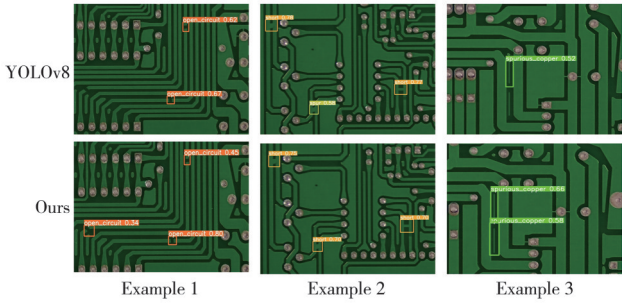


Fig. 8 Different defect test results

2.4.3 Comparative experiments

To ascertain the efficacy of the model, we conducted a comparative analysis between the proposed enhanced model based on YOLOv8 and various other object detection models. The experimental outcomes are presented in Table 6.

The experimental results indicate that in the application scenario of PCB defect detection, the detection accuracy values of Faster-RCNN, SSD, and RT-DETR models are relatively low, with $mAPs$ of 85.2%, 75.6%, and 83.4%, respectively. In contrast, the YOLO models have a clear advantage in accuracy, with the improved CSYOLO model achieving the highest mAP of 96.6%.

Table 6 Comparison of results of different models

Model	Model size/MB	Number of parameters	FLOPs/G	P/%	R/%	mAP/%
Faster-RCNN	108	136 811 934	369.9	92.3	79.8	85.2
SSD	93.2	26 285 486	62.7	79.7	67.7	75.6
RT-DETR[21]	108.4	29 276 390	105.2	81.0	76.2	83.4
YOLOv5	5.3	2 504 114	7.1	95.2	87.3	91.7
YOLOv6	8.7	4 500 080	13.1	90.9	81.9	88.6
GOLD-YOLO[22]	11.9	5 620 162	11.1	92.8	86.9	93.6
Mamba-YOLO[23]	87.2	21 837 562	49.7	93.9	89.0	93.5
YOLOv8n	6.1	3 012 018	8.2	93.5	85.2	92.1
YOLOv8s	21.5	11 137 922	28.4	94.8	94.2	96.4
Ours	6.5	3 256 862	10.2	96.3	90.5	96.6

Compared with the YOLOv8s model, the improved model maintains the same accuracy, but the model size, number of parameters, and GFLOPs are only 0.29, 0.36, and 0.3 times that of YOLOv8s, respectively. This means that the model size of the improved model is smaller, and the requirements for storage and computing resources are lower, which is conducive to deployment on resource-constrained devices. Furthermore, compared to models such as YOLOv5, YOLOv6, and YOLOv8n, the improved model also demonstrates superior performance in accuracy. Overall, the improved model provides excellent performance while keeping computational resource consumption low, which is a significant advantage in the field of PCB defect detection.

2.4.4 DeepPCB dataset

To verify the generalization ability of the model, we designed a set of experiments on the DeepPCB dataset[24]. The DeepPCB dataset comprises 1 500 pairs of images, each consisting of a defect-free template image and a corresponding test image annotated with defects. These images cover six of the most common types of PCB defects: "mouse bite", "short circuit", "spur", "spurious copper", "pin hole", and "open circuit". The experimental results are presented in Table 7.

Table 7 Comparison of accuracy of the proposed model with other models on the DeepPCB dataset

Model	AP/%						mAP/%
	Mouse bite	Short circuit	Spur	Spurious copper	Pin hole	Open circuit	
Faster-RCNN	94.2	91.0	93.8	93.8	94.6	94.7	93.7
SSD	91.4	93.5	92.7	95.2	93.2	95.3	93.2
RT-DETR	94.4	91.2	94.4	94.1	94.1	95.3	94.8
YOLOv5	98.3	94.8	98.3	98.7	97.9	98.5	97.8
YOLOv6	98.3	93.3	97.4	98.3	98.0	98.6	97.3
GOLD-YOLO	97.6	92.7	98.6	98.3	97.7	98.1	97.2
Mamba-YOLO	95.2	96.9	96.3	98.8	97.9	98.9	97.3
YOLOv8n	96.7	95.3	96.8	98.9	97.6	97.9	97.2
YOLOv8s	98.5	94.8	98.5	98.5	98.4	98.6	97.9
Ours	96.4	98.5	97.7	99.2	98.2	99.3	98.2

The model enhanced in this study demonstrates superior performance on the DeepPCB dataset, with a mAP of 98.2%, which is a 1.0% improvement over that of the baseline model. Especially in short circuit and open circuit defects, the target feature size is small, feature similarity is high, and feature extraction is difficult, which is not conducive to detection. The improved model also demonstrates superior performance in addressing the issue, achieving 3.2% and 1.4% increases in detection accuracy for the two types of defects.

Fig. 9 shows the detection results of the improved model and the baseline model. This suggests that the model proposed herein possesses commendable generalization capabilities.

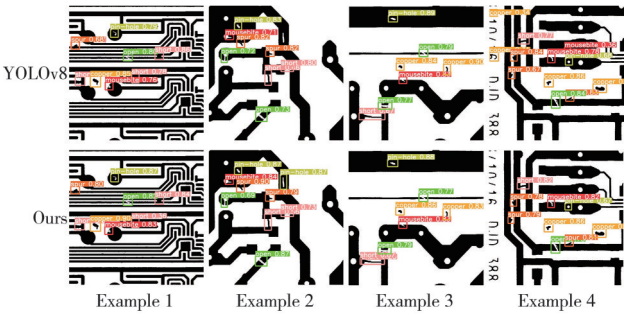


Fig. 9 Experimental results on DeepPCB dataset

3 Conclusions

To address the problem of PCB defect detection, we proposed an improved model based on YOLOv8, which significantly enhances detection accuracy while ensuring

detection efficiency. The model enhances feature expression by introducing a composite backbone network and dynamic snake convolution, thereby improving detection accuracy. At the same time, by replacing the traditional neck network with YOLO-FPN, the feature fusion capability is further strengthened, improving the detection accuracy for small objects. In terms of the loss function, replacing the original YOLOv8 loss function with the MPDIoU loss function significantly improves the convergence rate and accuracy of the model. Experiments on the PCB defect dataset have verified that the improved CSYOLOv8 model performes well and is effective in the task of PCB defect detection.

The next step will continue to study the improvement of model performance and optimize the detection capability for small defects in densely packed PCBs. It is expected to address defect detection tasks in different scenarios.

Acknowledgement

This work was supported by Natural Science Foundation of Gansu Province (No. 22JR5RA320).

Declaration of conflicting interests

The authors have no conflict of interests related to this publication.

References

[1] ZHOU Y B, YUAN M H, ZHANG J, et al. Review of vision-based defect detection research and its perspectives for printed circuit board. Journal of Manufacturing Systems, 2023, 70: 557-578.

[2] GIRSHICK R. Fast R-CNN//2015 IEEE International Conference on Computer Vision, December 7-13, 2015, Santiago, Chile. New York: IEEE, 2016: 1440-1448.

[3] REN S Q, HE K M, GIRSHICK R, et al. Faster R-CNN: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6): 1137-1149.

[4] LIU W, ANGUELOV D, ERHAN D, et al. SSD: single shot MultiBox detector//Computer Visio-ECCV 2016. Cham: Springer International Publishing, 2016: 21-37.

[5] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: unified, real-time object detection//2016 IEEE Conference on Computer Vision and Pattern Recognition, June 27-30, 2016, Las Vegas, NV, USA. New York: IEEE, 2016: 779-788.

[6] REDMON J, FARHADI A. YOLOv3: an incremental improvement. arXiv:1804.02767.

[7] REDMON J, FARHADI A. YOLO9000: better, faster, stronger//2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 6517-6525.

[8] LI C Y, LI L L, JIANG H L, et al. YOLOv6: A Single-Stage object detection framework for industrial applications. arXiv:2209.02976.

[9] WANG C Y, BOCHKOVSKIY A, LIAO H M. YOLOv7: trainable bag-of-freebies sets new state-of-the-art for real-time object detectors//2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 17-24, 2023, Vancouver, BC, Canada. New York: IEEE, 2023: 7464-7475.

[10] DING R W, DAI L H, LI G P, et al. TDD-net: a tiny defect detection network for printed circuit boards. CAAI Transactions on Intelligence Technology, 2019, 4(2): 110-116.

[11] LIN T Y, DOLLÁR P, GIRSHICK R, et al. Feature pyramid networks for object detection//2017 IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 936-944.

[12] SHI W, LU Z S, WU W, et al. Single-shot detector with enriched semantics for PCB tiny defect detection. The Journal of Engineering, 2020, 2020(13): 366-372.

[13] WANG L Y, BAI J, LI W J, et al. Research progress of YOLO series target detection algorithms. Computer Engineering and Applications, 2023, 59(14): 15-29.
WANG L Y, BAI J, LI W J, et al. Research progress of YOLO series target detection algorithms. Computer Engineering and Applications, 2023, 59(14): 15-29.

[14] BIAN B C, CHEN T, WU R J, et al. Improved YOLOv3-based defect detection algorithm for printed circuit board. Journal of Zhejiang University (Engineering Science), 2023, 57(4): 735-743.

[15] ZHANG H, WU C R, ZHANG Z Y, et al. ResNeSt: split-attention networks//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, June 19-20, 2022, New Orleans, LA, USA. New York: IEEE, 2022: 2735-2745.

[16] TANG J L, LIU S B, ZHAO D X, et al. PCB-YOLO: an improved detection algorithm of PCB surface defects based on YOLOv5. Sustainability, 2023, 15(7): 5963.

[17] TAN M X, PANG R M, LE Q V. EfficientDet: scalable and efficient object detection////2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 13-19, 2020, Seattle, WA, USA. New York: IEEE, 2020: 10778-10787.

[18] QI Y L, HE Y T, QI X M, et al. Dynamic snake convolution based on topological geometric constraints for tubular structure segmentation//2023 IEEE/CVF International Conference on Computer Vision, October 1-6, 2023, Paris, France. New York: IEEE, 2024: 6047-6056.

[19] MA S L, XU Y. MPDIoU: A Loss for Efficient and Accurate Bounding Box Regression. arXiv:2307.07662.
MA S L, XU Y. MPDIoU: a loss for efficient and accurate bounding box regression. 2023: arXiv: 2307.07662. https:

//arxiv. org/abs/2307. 07662

[20] LIU S, QI L, QIN H F, et al. Path aggregation network for instance segmentation//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June. 18-23, 2018, Salt Lake City, UT, USA. New York: IEEE, 2018: 8759-8768.

[21] ZHAO Y A, LV W Y, XU S L, et al. DETRs beat YOLOs on real-time object detection//2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition, June 16-22, 2024, Seattle, WA, USA. New York: IEEE, 2024: 16965-16974.

[22] WANG C C, HE W, NIE Y, et al. Gold-YOLO: efficient object detector via gather-and-distribute mechanism// NIPS'23: 37th International Conference on Neural Information Processing Systems, December. 10-16, 2023, New Orleans, LA, USA. Curran Associates Inc, 2023: 51094-51112.

[23] WANG Z Y, LI C, XU H Y, et al. Mamba YOLO: a simple baseline for object detection with state space model. 2024: arXiv: 2406. 05835. https://arxiv. org/abs/2406. 05835

[24] TANG S L, HE F, HUANG X L, et al. Online PCB defect detector on a new PCB defect dataset. 2019: arXiv: 1902. 06197. https://arxiv. org/abs/1902. 06197

CSYOLO: 融合复合主干网络与动态蛇形卷积的 YOLOv8 PCB缺陷检测模型

刘春娟[1], 张明璇[1], 闫浩文[2], 吴小所[1*], 王毅翔[1]

1. 兰州交通大学 电子与信息工程学院, 甘肃 兰州 730070;
2. 兰州交通大学 测绘与地理信息学院, 甘肃 兰州 730070

摘　要： 为解决印刷电路板(Printed circuit board, PCB)缺陷检测中精度与效率之间的平衡问题, 提出了一种基于YOLOv8的改进模型—CSYOLOv8, 用于PCB缺陷检测。首先, 设计了一种复合主干网络, 进行额外的特征提取, 以增强特征表达能力, 进而提升检测精度。其次, 提出了YOLO-FPN结构用以替代传统的颈部网络, 增强了特征融合能力, 并提升了对小目标的检测精度。此外, 为进一步优化模型对管状特征的提取能力, 引入了动态蛇形卷积。最后, 采用MPDIoU损失函数以提高模型的收敛速度与精度。实验结果表明, 改进后的模型在PCB缺陷数据集上的平均精度达到96. 6%, 相较于原YOLOv8模型提高了4. 5%, 且参数量为3 256 862, 平均检测速度为51. 8帧/秒, 能够有效满足精度与效率的双重需求。

关键词： 印刷电路板; 深度学习; 缺陷检测; YOLOv8; 多尺度特征融合; 损失函数

引用格式： LIU Chunjuan, ZHANG Mingxuan, YAN Haowen, et al. CSYOLO: a YOLOv8-based PCB defect detection model integrating composite backbone networks and dynamic snake convolution. Journal of Measurement Science and Instrumentation, 2026, 17(1): 151-161. DOI: 10.62756/jmsi.1674-8042.2026013