International Journal of Innovative Computing, Information and Control
Volume **19**, Number **2**, April **2023**

ICIC International ©2023 ISSN 1349-4198
pp. **447–463**

APPLICATION OF YOLOV4 ALGORITHM WITH INTEGRATED ATTENTION MECHANISM IN METAL SURFACE DEFECT DETECTION

Xikun Xie[1], Changjiang Li[1,*], Yang Liu[1], Junjie Song[1], Jonghyun Ahn[2] and Zhong Zhang[2]

[1]School of Mechanical and Power Engineering
Chongqing University of Science and Technology
Huxi University Town, Shapingba District, Chongqing 401331, P. R. China
{ 2020203050; 2021203204; 2021203242 }@cqust.edu.cn
*Corresponding author: 1993013@cqust.edu.cn

[2]Department of Intelligent Mechanical Engineering
Hiroshima Institute of Technology
2-1-1 Miyake, Saeki-ku, Hiroshima 731-5193, Japan
{ j.ahn.h2; t.sho.g4 }@cc.it-hiroshima.ac.jp

Received August 2022; revised November 2022

Abstract. *Deep learning methods based on YOLOV4 are widely used for defect detection. Still, problems with large backbone networks and high computation make it difficult to deploy on mobile-embedded devices with limited computing power. In this paper, we propose a DC-EMV2YOLOV4 model based on Deep Convolutional Generative Adversarial Network (DCGAN) to improve the sample quality imbalance problem. Meanwhile, it follows the EMV2-YOLOV4 model structure with an effective channel attention network for defect detection tasks. To improve the accuracy of the model for hard-to-detect defect classes under deployment requirements, the experiments use the generative adversarial network for sample expansion and accuracy improvement. The average accuracy of the final reconstructed image is the highest in all defect types of GCT10 and NEU, two types of metal surface defect datasets, with mAP of 0.747 and 0.862, respectively, which is higher than the detection accuracy of other detection networks. The DC-EMV2YOLOV4, compared to the EMV2-YOLOV4 model, increased the mAP values by 6.3% and 0.7% on both datasets, respectively. The number of parameters and inference time can meet the requirements of lightweight deployment and accuracy of metal surface defect detection.*
Keywords: Defect detection, DC-EMV2YOLOV4, Reconstructed image, Lightweight

1. **Introduction.** Deep learning methods have played a role in industrial image detection, with good results achieved through convolutional neural network training for feature extraction and classification [1,2] in recent years. Based on traditional vision inspection detection's high cost and low fitness problems, Balcioglu et al. [3] used the Deep Convolutional Neural Network (DCNN) approach for defect detection on gear metal manufacturing surfaces to meet the durability and efficiency requirements. However, there is still space for improving defect detection compared to the generic DCNN model. The reflective properties of general metal surfaces affect the accuracy of detection algorithms. Zhang et al. [4] used Mask R-CNN network to construct a surface defect detection system for mirror parts and performed migration learning with a small number of samples. However, there are problems of slow detection speed and low real-time performance in two-stage detection. Recently, originating from GAN's powerful capability in image generation, many

DOI: 10.24507/ijicic.19.02.447

GAN-based works are also applied to surface defect sample generation. Deep Convolution Generative Adversarial Network (DCGAN) [5] is able to expand the negative sample dataset and improve the accuracy of the defect detection algorithm.

A computer vision method with YOLO series single-stage detection achieved the extraction and classification of defective features. Cheng et al. [6,7] improved the YOLOV3 algorithm for small-size targets and introduced DIOU to accelerate model convergence to lift small-target defect feature recognition. However, the average detection rate and accuracy were still not high. Adding a visual attention mechanism to the defect detection algorithm can significantly lift feature extraction and defect detection accuracy. Hu et al. [8] proposed that the SENet (Squeeze-and-Excitation Network) mainly improves the network's performance by modeling the channel relationships, but it does not fully use global contextual information. The YOLO model can greatly enhance the detection accuracy requirements and achieve real-time detection speed to meet industrial needs. Therefore, many scholars have addressed the problems of insufficient information processing capability at small scales and the tricky balance between detection accuracy and speed in the improvement based on the YOLO model. In the modified YOLOV4 [9] model proposed by Liu et al. [10], the K-means clustering method is used to redefine the confidence loss as well as the MobileNetV3 lightweight network to replace the YOLOV4 backbone feature extraction network. However, the poor detection effect due to sample imbalance is not addressed, and the accuracy does not meet the online detection requirements. To achieve the balance between accuracy and speed in aluminum strip defect inspection, Ma et al. [11] proposed a lightweight detection method based on an attention mechanism. The backbone network from the YOLOV4 model was replaced with YOLO-DCSAM. Eventually the detection speed achieved greatly improved. However, the model parameters are complex in small-scale defect detection, and the lower frame rate leads to a slower operation speed problem.

Therefore, the paper proposes the improved DC-EMV2YOLOV4, which has achieved the latest results in defect morphology detection. DC-EMV2YOLOV4 effectively balances the detection accuracy and real-time dynamic performance of metal-like products with surface defects. Our main contributions include the following.

(i) This paper proposes the DC-EMV2YOLOV4, a lightweight defect inspection model for metal surfaces based on the EMV2-YOLOV4 [12] algorithm. We continue with an ECA-MobileNetV2 (EMV2) backbone to enhance feature extraction from multi-category morphology by replacing MobileNetV2 network with an ECA module. Finally, the algorithm can solve the problem of the low accuracy of small-scale defect recognition.

(ii) We design to reconstruct the defective sample dataset using the DCGAN approach to improve the accuracy of the model. Eventually, the problem of low recognition rate of the model due to insufficient negative samples is solved.

(iii) Experiments are conducted on the GCT10 and NEU datasets, the latest results show that DC-EMV2YOLOV4 effectively solves the problem of low recognition accuracy due to the influence of defect features by background pixels and brightness.

In this paper, we first introduce the background of the study and the proposed innovation in the introduction section. Then we introduce the dataset of DCGAN image synthesis expanded samples and the basic model YOLOV4 in the methodology section, based on which the improved EMV2-YOLOV4 model is obtained. In the third section, the experimental environment is set up as well as the comparison of the frontier algorithms and the comparative ablation experiments are made. Finally, the fourth section is the paper's conclusion, and the future improvement directions are discussed.

2. Methodology.

2.1. **Sample dataset expansion.** In order to improve the quality and training stability of GAN-generated samples, the classical DCGAN is used for image dataset expansion. It is a neural network architecture that combines convolutional neural networks and unsupervised learning. Combining convolutional networks with powerful feature extraction and GAN specialized in learning data distribution can significantly improve the quality of the generated samples. On the one hand, compared with normal GANs, DCGAN uses convolutional layers instead of fully connected and up-sampling. On the other hand, Batch Normalization (BN) is applied to the network architecture, which maps the resultant output of feature layers to a specific space and improves the training speed of the model. The DCGAN model architecture is shown in Figure 1.

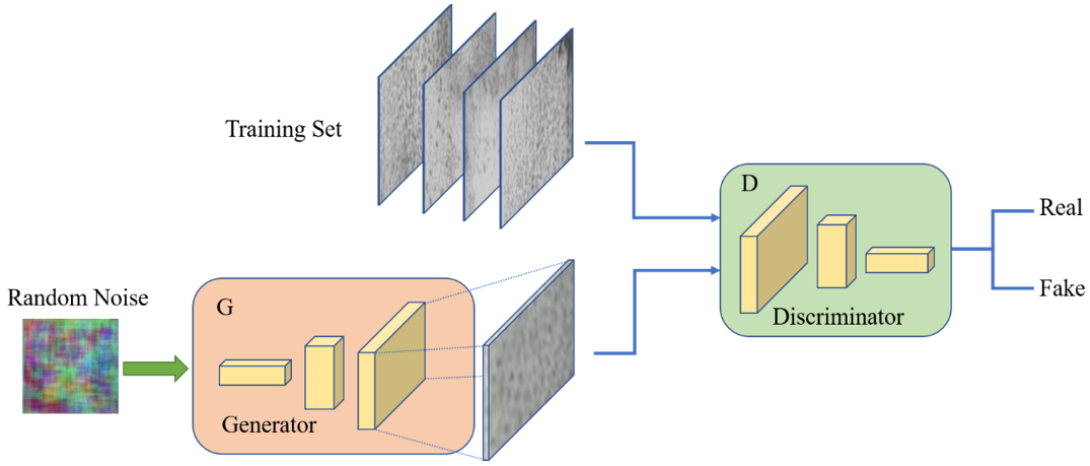


FIGURE 1. DCGAN model architecture

The G is a Generator responsible for capturing the probability distribution of the training data and generating an actual image similar to the source image by simulating the real morphological distribution of the sample space. The other one is a Discriminator to discriminate whether it is synthetic data. The framework training process alternates iterations similar to a gaming game, and the cycle repeats itself, eventually reaching Nash equilibrium, leading to model convergence. Finally, the training data distribution is successfully simulated to achieve the goal of generating real images. We use the NEU dataset as an example for data generation objectives. The NEU dataset generation results are shown in Figure 2.

2.2. **YOLOV4 and ECA structure.** YOLOV4 is the target detection algorithm that upgrades the performance of the YOLO series. Based on the original target detection framework, it has been optimized to enhance feature extraction, enhance network model nonlinearity, and prevent overfitting, respectively. Many defects have been remedied during the continuous iteration of the version. However, YOLOV4 has problems such as high model complexity and low running speed on hardware devices. Therefore, the improved YOLOV4 algorithm is proposed. The network architecture is shown in Figure 3.

Channel attention mechanisms have been shown to have great potential in improving the performance of deep Convolutional Neural Networks (CNNs), with representative approaches being SENet and Effective Channel Attention (ECA) networks for deep CNNs.

The SE module first employs global average pooling for each channel independently, and following uses two Fully Connected (FC) layers and a nonlinear sigmoid function to generate the channel weights as shown in Figure 4(a). The two FC layers are designed to

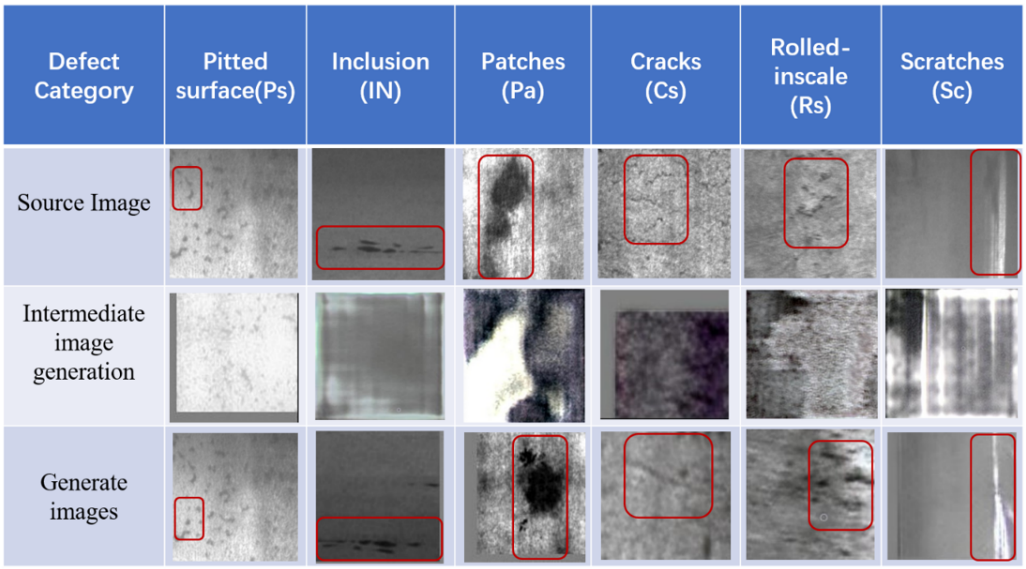


FIGURE 2. NEU dataset generation results

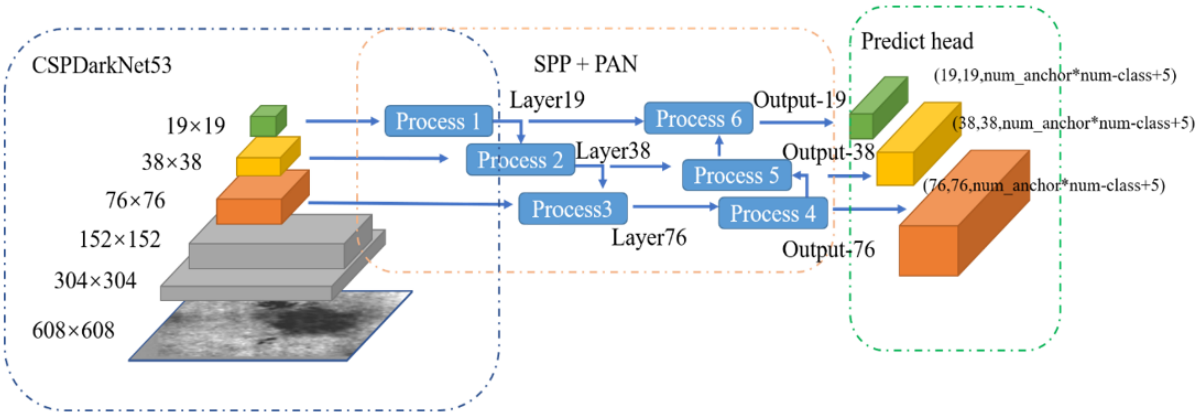


FIGURE 3. YOLOV4 model structure

capture nonlinear cross-channel interactions, which involve dimensionality reduction to control the complexity of the model. Although this strategy is widely used in subsequent channel attention modules, dimensionality reduction can have side effects on channel attention prediction, and dependencies are inefficient and unnecessary for capturing all channels. The ECA attention mechanism uses a local cross-channel interaction strategy without dimensionality reduction by considering each channel and its k nearest neighbors, which effectively improves the feature extraction capability and efficiency of the detection model.

As shown in Figure 4(b), after performing the channel-by-channel global average pooling gap, ECA performs a one-dimensional convolution of convolution size k. As in Equation (1), k represents local cross-channel interaction coverage. The k value is determined by an adaptive method, as in Equation (2), and there is a nonlinear relationship between k and the number of output channels c. Subsequently, the relationships between the convolution kernel size k and the number of channels c and the hyperparameters γ and b are obtained in Equation (3). The k represents local cross-channel interaction coverage. We set k as the convolution kernel size, c as the number of output channels, as well as γ and b as the

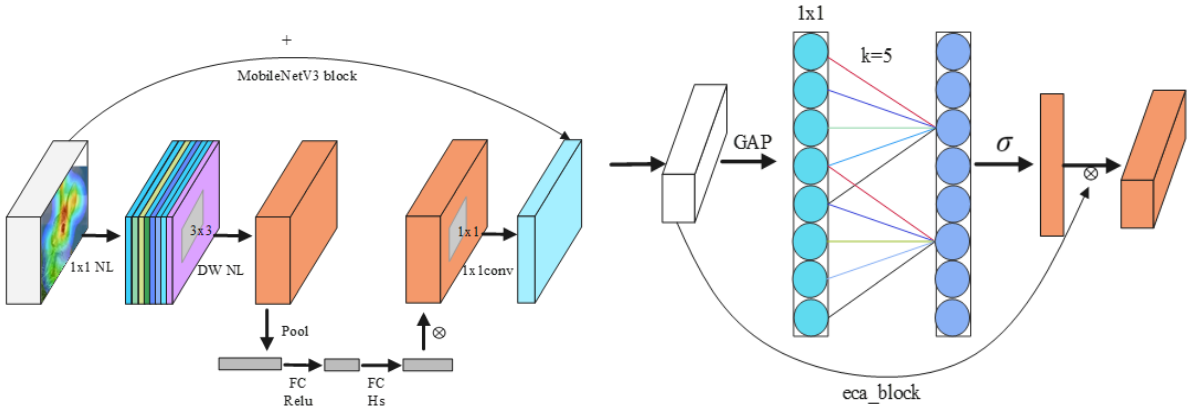


FIGURE 4. Comparison of attention mechanism structure: (a) SE attention mechanism; (b) ECA attention mechanism

model hyperparameters.

$$w = \begin{bmatrix} w^{1,1} & \cdots & w^{1,k} & 0 & 0 & \cdots & \cdots & 0 \\ 0 & w^{2,2} & \cdots & w^{2,k+1} & 0 & \cdots & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 & \cdots & w^{c,c-k+1} & \cdots & w^{c,c} \end{bmatrix} \tag{1}$$

$$c = \phi(k) = 2^{(\gamma * k - b)} \tag{2}$$

$$k = \psi(c) = \left| \frac{\log_2(c)}{\gamma} + \frac{b}{\gamma} \right|_{odd} \tag{3}$$

SENet uses Equation (4) sigmoid function as the activation function, which can decrease the speed of the inference process in the derivation and quantization. Compared with the sigmoid function, the ReLu function enables the addition of grid-fitting nonlinear mapping in the neural network. Still, the lack of upper limit value of ReLu tends to cause gradient explosion. Introducing the empirical function Equation (5) ReLu6 as the transition function to construct the ECA Equation (6) h-sigmoid function as the activation function can improve the convergence speed. As shown in Figure 5, the h-sigmoid function is a segmented linear approximation of the logistic sigmoid activation function, which is easier to compute than the ordinary sigmoid function and can effectively reduce the gradient disappearance during the model training process, making the learning computation faster.

$$\text{sigmoid: } \sigma(x) = \frac{1}{1 + e^{-x}} \tag{4}$$

$$\text{ReLu6}(x) = \min(\max(x, 0), 6) \tag{5}$$

$$\text{h-sigmoid: } f(\text{ReLu6}(x+3)) = \begin{cases} 1, & (x > 3) \\ \text{ReLu6}(x+3)/6 + 0.5, & (3 \geq x \geq -3) \\ 0, & (x < -3) \end{cases} \tag{6}$$

MobileNetV2 [13] network is a lightweight deep neural separable network model built using deep separable convolution and inverse residual structure. Depth-Separable Convolution (DwConv2D) consists of Depth-Wise Convolution (DW) and Point-Wise Convolution (PW), the number of convolution kernels is the same as the input channel, and the

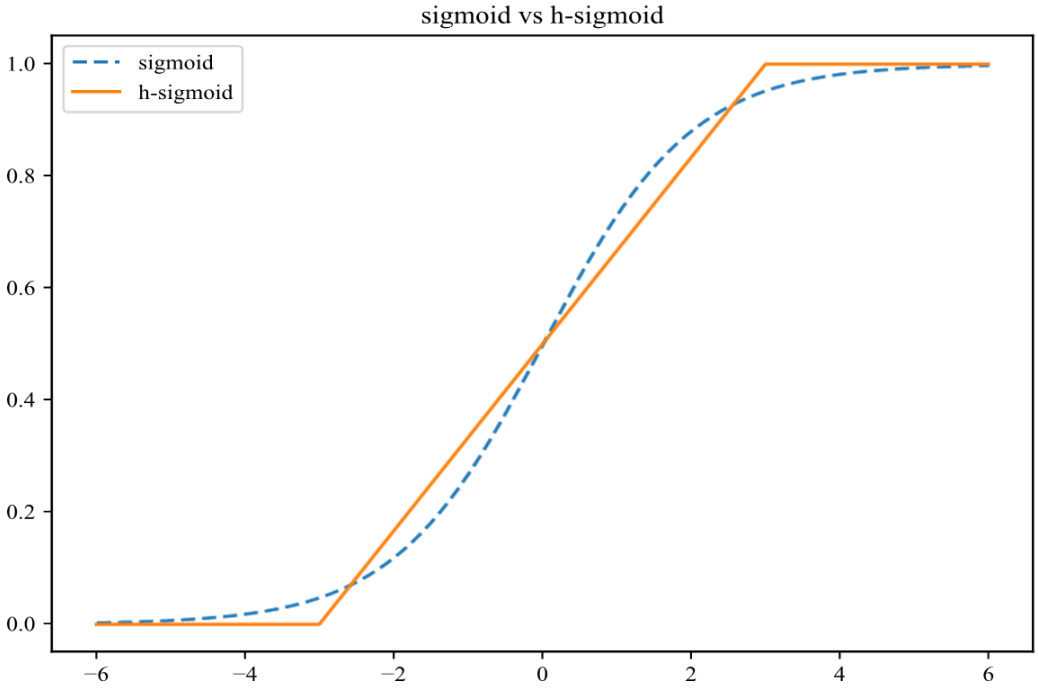


FIGURE 5. The sigmoid and h-sigmoid activation function contrast

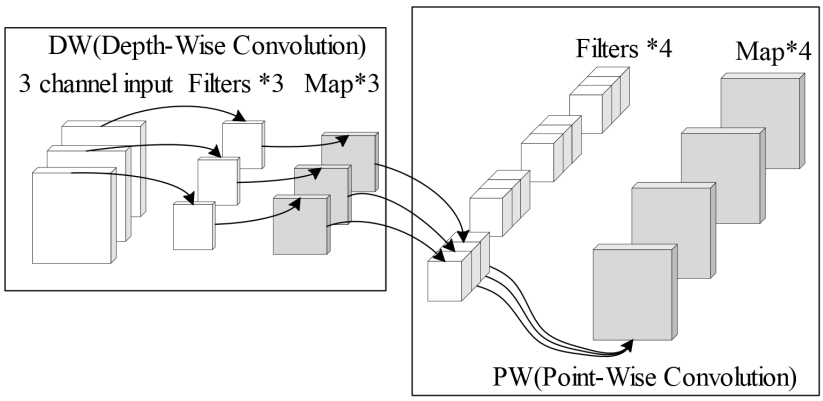


FIGURE 6. Depth-Separable Convolutional (DwConv2D) structures

size is usually 3×3 in Figure 6. Equation (7) is the depth convolution DW, where G is the output feature map, K is the convolution kernel of $W \times H$, X is the input feature map, m denotes the m channel of the feature map, and i, j denotes the coordinates of the output feature map on the m channel. w, h are the coordinates of the convolution kernel weight elements of the first channel. Point-by-point convolution is basically the same as normal convolution, with the size of the convolution kernel set to 1×1. The depth-separable convolution DwConv2D replaces the standard convolution with fewer parameters and less computation, and the computation is only 1/9 of the standard convolution.

$$G_{i,j,m} = \sum_{w,h}^{W,H} K_{w,h,m} \cdot X_{i+w,j+h,m} \tag{7}$$

The ECA-MobileNetV2 convolutional structure consists of a standard convolutional module CBR (Conv-BN-ReLu), and multiple EIbneck (ECA_Inverted_bottleneck) stacked convolutional blocks, as shown in Figure 7. Figure 7(a) shows the structure of EIbneck,

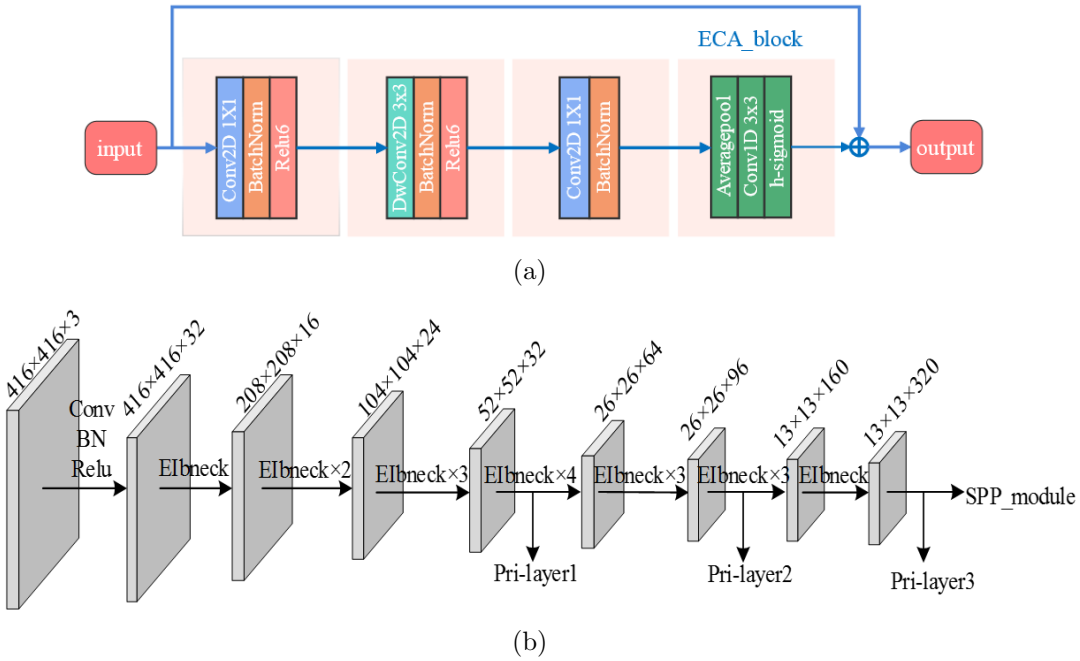


FIGURE 7. The structure of the ECA-MobileNetV2: (a) The EIbneck structure from the ECA-MobileNetV2; (b) the backbone structure from the ECA-MobileNetV2

which integrates the ECA_block attention mechanism and the feature extraction network of MobileNetV2. The CBR module is primarily run for feature extraction, followed by the deep separable convolution module DwConv2D convolution of size 3×3 and Conv2D normal convolution for adjusting the number of channels. The output feature layer fuses the final ECA_block convolution operation. The EIbneck module combines the lightweight feature of MobileNetV2 and the effective feature extraction capability of ECA. It is a lightweight feature extraction network model with local cross-channel interaction, which can effectively enhance useful feature information and suppress useless feature information, improving the detection accuracy and efficiency of the model.

The backbone structure of ECA-MobileNetV2 is shown in Figure 7(b). First of all, the backbone sets input size as $416 \times 416 \times 3$ image, and $208 \times 208 \times 16$ scale feature image is obtained by CBR convolution and once EIbneck, and the first preliminary feature layer Pri-layer1 with the scale of $52 \times 52 \times 32$ is obtained after applying five times EIbnecks. Furthermore, Pri-layer1 outputs the second preliminary feature layer Pri-layer2 with the scale of $26 \times 26 \times 96$ after 7 times of EIblock convolutions. Finally, Pri-layer2 outputs the third preliminary feature layer Pri-layer3 of $13 \times 13 \times 320$ after 4 times of EIblock convolution. To further improve the detection accuracy from small-scale, low-contrast and other hard-to-detect defects, the output preliminary feature layers of different scales need further feature enhancement.

2.3. **DC-EMV2YOLOV4 network architecture.** The overall structure of the DC-EMV2YOLOV4 detection model is shown in Figure 8. The whole network structure is divided into three parts: the backbone feature extraction network ECA-MobileNetV2; the enhanced feature extraction networks SPP (Spatial Pyramid Pooling) and PANet (Path Aggregation Network); and the prediction network YOLO_predict, which uses the obtained features for prediction. The DC-EMV2YOLOV4 network model algorithm

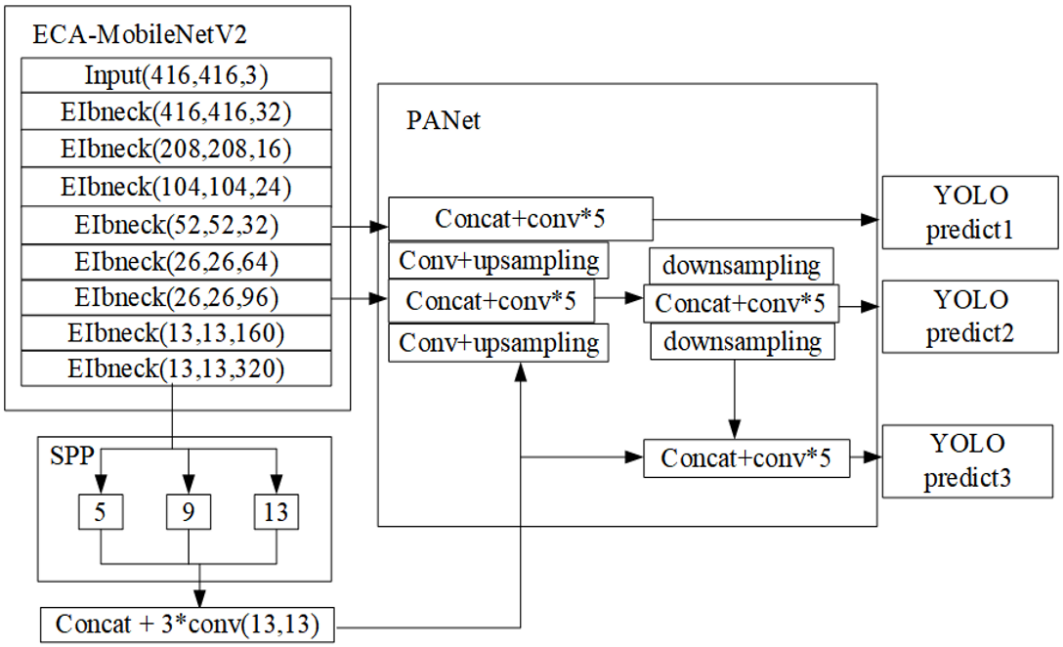


FIGURE 8. Overall structure diagram

introduces the feature strengthen networks SPP and PANet of YOLOV4. The SPP network can extract multi-scale depth features with different sensory fields and stitch on the channel dimension of the feature mapping to fusion. To adapt to the various shape and size of defects in surface defects, the small-scale feature layer Pri-layer3 output by ECA-MobileNetV2 is first enhanced using the SPP module. Pri-layer3 is pooled at three scales of 5×5, 9×9, and 13×13, and the processing results are data spliced to obtain feature enhancement. Finally, the output scale is 13×13. The feature enhancement network PANet is to enhance the whole feature level by bottom-up path enhancement using an accurate bottom-localization signal. Therefore, the information path between bottom and top-level features can be shortened. There are three different scales of the output feature map outputs through multi-scale features fusion of the target feature and repeatedly upsampling and downsampling operations.

The first prediction feature (YOLO_predict1) is obtained by splicing Pri-layer2 and Pri-layer3 features which is obtained from the extraction network ECA-MobileNetV2 with the features obtained from SPP. The Pri-layer2 features are fused with the features from the SPP augmentation network after up-sampling, and the second prediction feature (YOLO_predict2) is obtained by down-sampling with Pri-layer1. The third prediction feature (YOLO_predict3) is obtained fusing Pri-layer2, Pri-layer1 features and SPP network feature enhancement several times. The convolution operation and decoding operation are performed on the prediction features respectively, and finally the prediction suggestion box anchor box corresponding to three different scale defects is obtained.

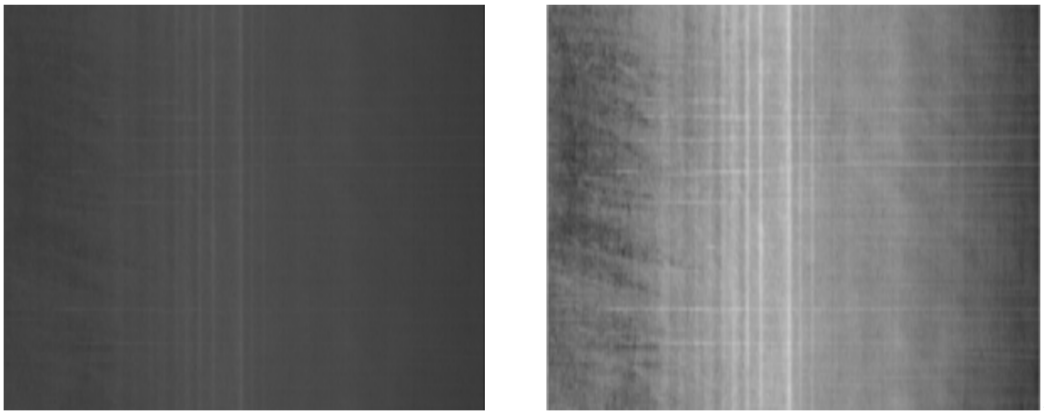
3. **Experiments.**

3.1. **Experimental settings.** The experiments use NVIDIA GTX1660Supper (6G) graphics card, Intel(R) Core (TM) i7-7700 CPU @ 3.60GHz as the experimental hardware device, and PyTorch learning framework to build the detection model on the Win10 OS platform. This paper conducts the detection experiments which are performed on two types of metal defect datasets, comparing the traditional multi-scale fusion detection network FasterRCNN, SSD300, and quantization network MV3yoloV4 incorporating

attention [14] Light, EfficientDet [15], and the light-weight network GhostyoloV4. Experimentally compare the variation of DC-EMV2YOLOV4 (ours) detection accuracy, number of parameters, and frame resolution FPS. Experiments found that the paper's network model does not need to iterate multiple training, resulting from the model fully converging with the actual model training model in 100 epochs. In addition, the experiment using the model MobileNetV2-YOLOV4 based on the VOC dataset training weights as pretraining parameters transfer learning to the defect detection experimental model. The size of BatchSize is set to 16, the initial 50 epochs using a learning rate of 1e-3, 50 to 100 epochs using a learning rate of 1e-4 for freeze weight training, and the BatchSize is set to 8.

3.1.1. *Datasets pre-processing.* The base dataset originated from the Tianjin University industrial metal surface defect dataset (GCT10 dataset) and the Northeastern University steel plate surface defect dataset (NEU dataset). The GCT10 dataset uses 512×512 size input, with a total of 3500 images, and has 10 categories of surface defects, including Punch-hole (Pu), Welding-line (Wl), Crescent-gap (Cg), Water-spots (Ws), Oil-spots (Os), Silk-spots (Ss), Inclusions (In), Rolled-pit (Rp), Crease (Ce), and Waistfloding (Wf). Similarly, NEU dataset uses 200×200 size as input, and has 6 categories of surface defects, 300 images each, including Rolled-inscale (Rs), Patches (Pa), Cracks (Cs), Pitted-surface (Ps), Inclusions (IN), Scratches (Sc).

Defect images are disturbed by the production line site environment, resulting in uneven image grayscale and low contrast between defects and background. Each defect image contains at least one defect, the defect edge is also not obvious, the shape is complex and variable, and some images contain multiple defects of different scales. Since the presence of defect contrast is not obvious, image will affect the model training accuracy, and the image is preprocessed with the Retinex [16] algorithm to enhance the contrast of defects in the original image. The defect image enhancement contrast results are shown in Figure 9. To accommodate multi-type defect detection at different scales in complex backgrounds, the enhanced dataset is used to train the EMV2 classification model, which is then fed into the EMV2-YOLOV4 detection model for training.



(a) (b)

FIGURE 9. Image enhancement effect contrast diagram: (a) Original image of the waist crease defect; (b) waist crease notch treated by the Retinex algorithm

3.1.2. *Experimental evaluation.* The accuracy P (Precision) and recall R (Recall), as well as the mAP (mean Average Precision) of the two datasets, are used as the model judging criteria. Precision indicates the proportion of classifiers that classify accurately and Recall indicates the proportion of classifiers that identify correctly as a positive class. The mAP of the area classes consisting of Precision and Recall is used as the overall performance index of the network model, where Precision and Recall are calculated as Equations (8) and (9).

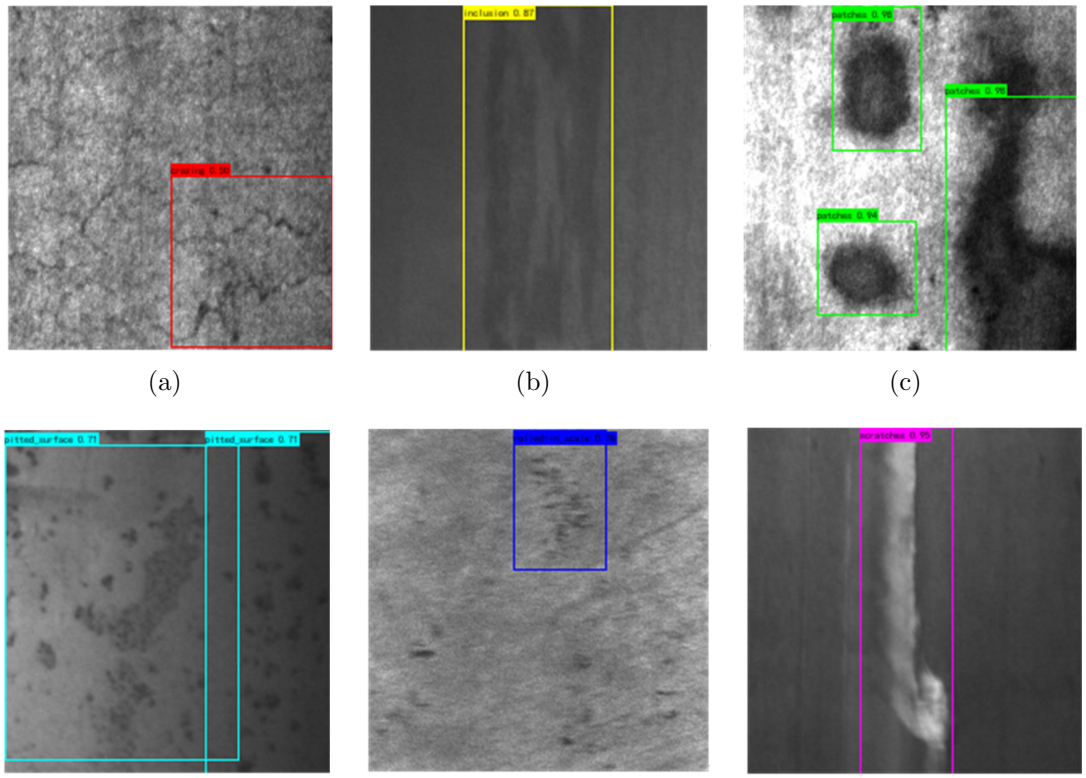
$$P = \frac{TP}{TP + FP}, \quad R = \frac{TP}{TP + FN} \tag{8}$$

$$mAP = \frac{\sum_{i=0}^{n} AP(i)}{n} \tag{9}$$

where TP is the number of correct predictions, and FP is the number of correctly predicted samples. FN is the number of incorrectly predicted negative samples, n is the number of detected defect categories, and AP is the accuracy of each defect category.

3.2. **Baseline comparison.**

3.2.1. *DC-EMV2YOLOV4 prediction results.* When measuring accuracy in the application of lightweight models, the single image inference time (ms), as well as FPS, is used for the model indicator in detecting speed. The detections of the algorithm DC-EMV2YOLOV4 on the NEU and the GCT10 in this paper are shown in Figure 10 and Figure 11. We point out in the abstract and introduction section that DC-EMV2YOLOV4



(a) (b) (c)

(d) (e) (f)

FIGURE 10. NEU prediction of various defects: (a) Cracks0.50, (b) Inclusions0.87, (c) Patches0.98, (d) Pitted-surface0.71, (e) Rolled-inscale0.78, and (f) Scratches0.95

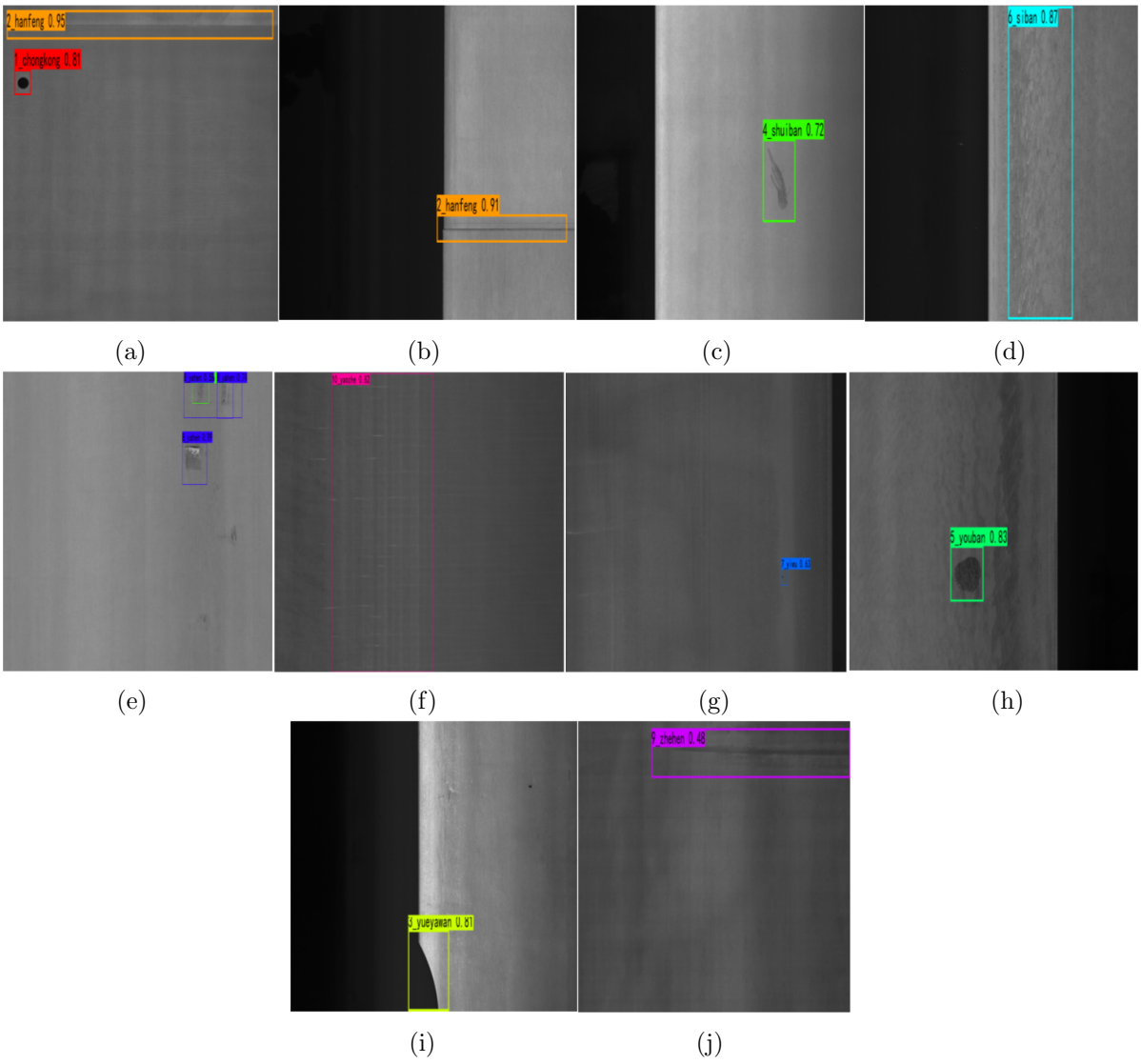


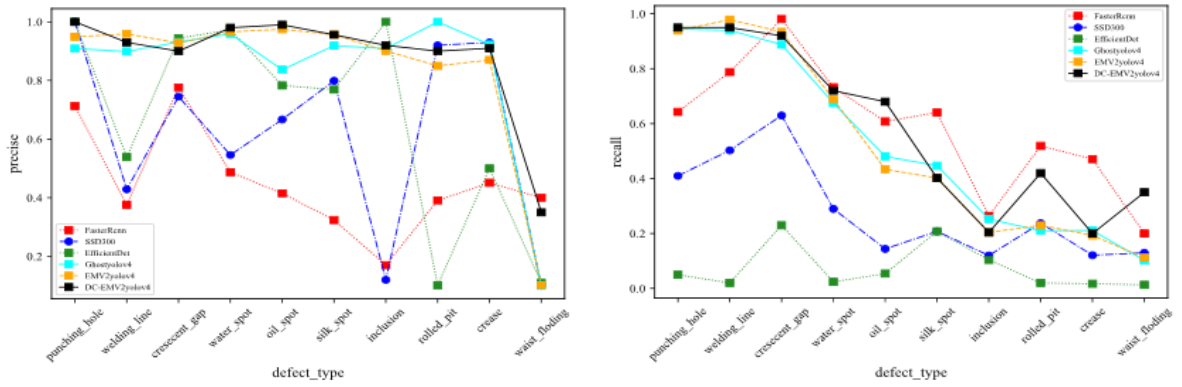
FIGURE 11. GCT10 prediction of various defects: (a) Punch-hole0.81, (b) Welding-line0.91, (c) Water-spots0.72, (d) Silk-spots0.87, (e) Rolled-pit0.74, (f) Waistfloding0.82, (g) Inclusions0.63, (h) Oil-spots0.83, (i) Crescent-gap0.81, and (j) Crease0.48

still uses the EMV2-YOLOV4 model algorithm and that DCGAN expansion of the model will change the accuracy of the model. However, it does not affect the model's overall detection speed and detection time. Therefore, only the accuracy effect is compared and analyzed in the experimental section.

3.2.2. *Comparison with state-of-the-art methods.* The experiments compare improved EMV2-YOLOV4 and other mainstream lightweight network algorithms and the traditional algorithms in Table 1. In the original experiment, the detection accuracy AP of the EMV2-YOLOV4 algorithm is the highest for all defects on GCT10 except Wf. In contrast, Pu, Wl, and Cg category defects are above 97% detection accuracy. Average detection accuracy mAP is up to 0.684. The value increases by 22.7% and 35.1% respectively, for FasterRCNN [17] and SSD300 [18], rising by 34.6% compared to EfficientDet with the multiscale lightweight network, and increases by 5.1% for the GhostyoloV4 model incorporated into the lightweight module GhostNet [19].

TABLE 1. Dataset accuracy

Types	AP					
	FasterRCNN	SSD300	EfficientDet	GhostyoloV4	EMV2-YOLOV4	DC-EMV2YOLOV4
Pu	0.55	0.59	0.61	0.94	**0.99**	**1.0**
Wl	0.52	0.70	0.03	0.95	**0.98**	0.96
Cg	0.88	0.58	0.93	0.96	**0.97**	0.95
Ws	0.70	0.34	0.71	0.78	**0.84**	**0.88**
Os	0.51	0.28	0.30	0.68	**0.81**	**0.92**
Ss	0.43	0.44	0.52	0.70	**0.80**	0.80
In	0.07	0.03	0.02	0.35	**0.44**	**0.63**
Rp	0.40	0.14	0.10	0.22	**0.32**	**0.48**
Ce	0.36	0.20	0.13	0.58	**0.59**	0.50
Wf	0.15	0.03	0.03	**0.17**	0.10	**0.35**
GCT10_mAP	0.457	0.333	0.338	0.633	**0.684**	**0.747**
Cs	0.50	0.24	0.38	0.46	**0.59**	**0.72**
IN	0.86	0.35	0.65	0.88	**0.89**	0.83
Pa	0.96	0.66	0.87	0.97	**0.96**	**0.98**
Ps	0.83	0.61	0.67	0.85	**0.88**	0.85
Rs	0.73	0.34	0.59	0.80	**0.83**	**0.87**
Sc	0.96	0.41	0.40	0.97	**0.98**	0.92
NEU_mAP	0.807	0.435	0.593	0.822	**0.855**	**0.862**

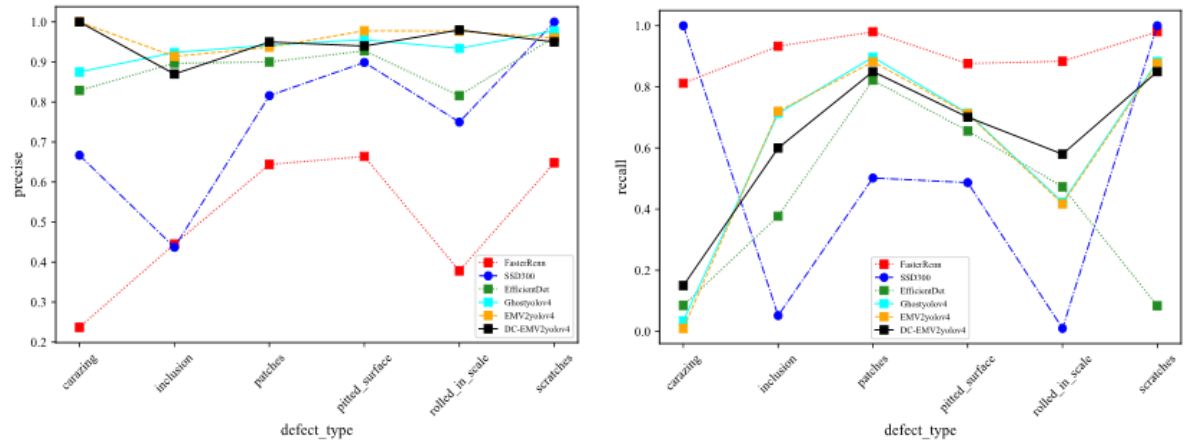


(a) (b)

FIGURE 12. Comparison from accuracy and recall in GCT10 sets: (a) Accuracy comparison; (b) recall comparison

On the dataset NEU, the detection accuracy is the highest for all defects with mAP of 0.86. The visualization of precision and recall in the accuracy metrics in the experiments is shown in Figure 12 and Figure 13. Combined with the dual metrics analysis EMV2-YOLOV4 still achieves the best performance on mAP.

DC-EMV2YOLOV4 expands the experimental data, and the accuracy is improved for most types of defects compared to that on the original dataset. In particular, the effect is obvious for the Wf and Ce categories of defects with low accuracy on the original dataset. We analyze that the sample size of these two types of defects in the original data is small, and it is difficult for the network model to identify the morphology of the features. The model learning performance can be improved by adding negative samples. As shown in Table 1, the DC-EMV2YOLOV4 algorithm represents the accuracy of the expanded dataset.



(a) (b)

FIGURE 13. Comparison from accuracy and recall in NEU sets: (a) Accuracy comparison; (b) recall comparison

The experiments show that EMV2-YOLOV4 is better than other mainstream lightweight network algorithms and other algorithms. The EIbneck module of EMV2-YOLOV4 adopts the attention mechanism ECA and uses cross-channel information fusion and adaptive convolution methods to assign adaptive weights to important regions of images. Experiments also show that the multi-scale feature fusion and different-scale feature detection and rescreening algorithms of EMV2-YOLOV4 can adapt to different types and scales of defects.

Furthermore, we contrast the detecting speed in Table 2. The number of parameters of EMV2-YOLOV4 is 10.424M, which meets the requirements of lightweight deployment. Meanwhile, the defect detection inference time in NEU and GCT10 datasets is 18.44 ms/img and 18.50 ms/img. The frame rate also reaches 54.25 f/s and 54.09 f/s, respectively, which meet the demand of industrialized real-time detection.

TABLE 2. Comparison of time and detection speed algorithms on the two datasets

Algorithm	Dataset				
	NEU		GCT10		
Type	Infertime (ms/img)	FPS	Infertime (ms/img)	FPS	Light-parameter (M)
FasterRCNN	143.17	6.98	167.41	5.97	–
SSD300	44.95	22.25	46.65	21.44	–
EfficientDet	46.11	21.7	45.19	22.13	–
YOLOV4	83.42	11.99	83.3	12.01	64.04
MV1yoloV4	38.49	26.0	36.77	27.21	12.692
MV2yoloV4	21.84	45.44	21.91	45.73	10.801
MV3yoloV4 (SE)	36.56	27.36	38.16	26.24	11.729
GhostyoloV4	24.76	40.42	25.86	38.71	11.052
EMV2-YOLOV4 [12]	18.44	54.25	18.50	54.09	10.424

Compared with the traditional network models such as FasterRCNN and EfficientDet, the EMV2-YOLOV4 from this paper can significantly reduce the inference time. FasterRCNN as a two-stage network structure in the number of parameters is large, the effect

on the deployment of model lightweight is not apparent. In contrast, the lowest number of parameters in the light structure of EfficientDet is only 9.84M, but the accuracy is poor. Although the backbone network from EfficientDet is lightweight and efficient, the multi-scale fusion information in each channel of the BiFN feature module will increase the inference time. EMV2-YOLOV4 algorithm has a lower inference time than other lightweight network models such as GhostyoloV4 and MV3yoloV4. MV3yoloV4 will have improved model accuracy after imposing the attention mechanism, but there will be an increase in the number of parameters, accompanied by an increase in memory consumption. In contrast, the ECA module incorporated in EMV2-YOLOV4 uses one-dimensional convolution for channel weight sharing, reducing the loss of channel information and still achieving higher accuracy with fewer model parameters. The incorporation of the attention mechanism can improve the lightweight effect, and the fusion of ECA module is better than the incorporation of the SE module.

Ultimately, comparison between the number of parameters and the average accuracy of defects is shown in Figure 14. Compared with MV2yoloV4, the lowest number of EfficientDet parameters in the lightweight model is only 9.84M to meet the lightweight requirements. However, the model accuracy is lower than other networks. The MV3yoloV4 with attention mechanism is more accurate, but the number of parameters is 11.8M, and the model running speed is reduced. Comprehensive analysis of the model accuracy and the number of parameters shows that the EMV2-YOLOV4 model with the attention mechanism has the highest accuracy and relatively low number of parameters, which is suitable as the most optimal choice for the lightweight deployment model.

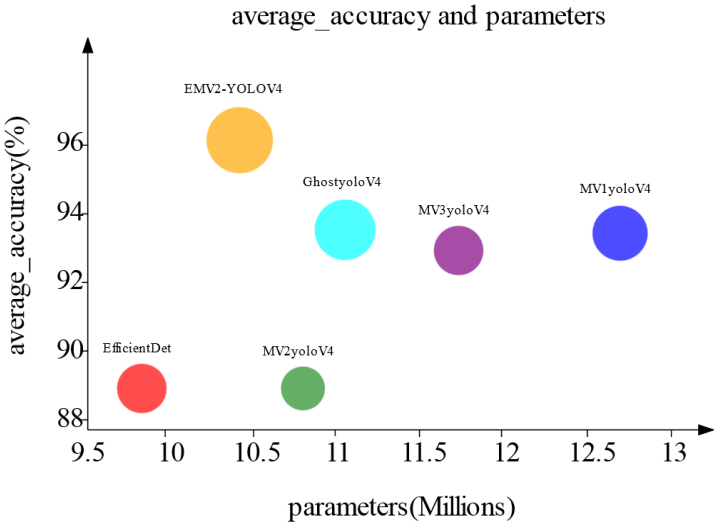


FIGURE 14. Comparsion of average accuracy with the number of parameters

3.3. **Ablation experiments.** The ablation experiment results for the algorithm in this paper are shown in Table 3. The average detection accuracy mAP of the metal surface defect dataset is used to compare the ECA of the EIbenck module of this paper, SE module of MV3yoloV4, migration transfer learning, and different α (width factors). It can demonstrate the performance improvement contribution to the various improvements of the YOLOV4 model. The outcome includes the comparison of the contribution of the ECA module to the MobileNetV2 network, the detection of the migration of the model weight parameters from the GCT dataset to the NEU dataset using migration training, and the comparison of the contribution of the validation migration learning method to EMV2-YOLOV4, MV3yoloV4, and MV2yoloV4.

Table 3. Ablation experiments results

Network-models	Factors					
	α	SE	ECA	Transfer learning	GCT_map/%	NEU_map/%
YOLOV4	–	–	–	–	47.2	64.5
MV3yoloV4	–	–	–	–	56.74	77.71
	0.5	–	–	–	53.3	73.0
	–	√	–	–	56.74	75.89
	–	–	–	√	56.74	75.68
MV2yoloV4	–	–	–	–	57.49	74.97
	0.5	–	–	–	56.90	79.3
	–	–	–	√	57.49	77.61
EMV2-YOLOV4 [12]	0.5	–	√	√	68.4	**85.5**

The lightweighting effect of MV3yoloV4 with the addition of SE is similar to that of EMV2-YOLOV4, but the improvement in detection accuracy is much lower than that of EMV2-YOLOV4; the detection accuracy of EMV2-YOLOV4 with the addition of ECA is improved by 6% compared with MV2yoloV4, and the parameter volume is reduced by 1.3M. The training using the transfer learning EMV2-YOLOV4 with width factor α on top of YOLOV4 will sacrifice some accuracy and increase the model inference time, but the final detection accuracy is improved by 21.2%, and the number of parameters is only 1/6 of YOLOV4.

4. **Conclusion.** To balance the defect detection accuracy and lightweight configuration conditions and to solve the problem of low accuracy of the YOLO algorithm on small-size datasets, this paper proposes the DC-EMV2YOLOV4 model. Compared with the original YOLOV4 model, which has the problem of low model extraction accuracy, the improved backbone network EIbneck instead of the CSPDarkNet network can improve the detection accuracy. At the same time, the experimental comparison of two public datasets shows that the proposed model can perform best in terms of the number of parameters and the accuracy of each category. The experimental results show that mAP detection accuracy is still the highest on the GCT10 dataset as well as the NEU dataset after sample expansion, with 0.747 and 0.862, respectively. Meanwhile, the study shows that the original EMV2-YOLOV4 algorithm has a low number of parameters (only 10.4M), which can improve the detection accuracy as well as the adaptability to the defect type and scale. Finally, the DC-EMV2YOLOV4 detection network was built based on the deep learning PyTorch framework. We provide the following direction for future improvement: The task of detecting defects resourced from metal products in industrial environments has serious noise interference already environmental lighting problems. This factor can greatly affect the model's accuracy and actual detection speed proposed in the paper. Different illumination methods are chosen for other metal surface properties. Therefore, we suggest using a coaxial light source illumination method under reflective data samples to weaken the effect of environmental noise. This behavior can ensure that the same detection accuracy remains optimal under natural light conditions.

Acknowledgment. This work is supported by National Key R&D Program Projects (Grant Number: 2017YFC0805900).

REFERENCES

[1] T. N. Zhang, E. Q. Chen and W. F. Xiao, Fast target detection method for improving MobileNet_YOLOv3 network, *Journal of Chinese Computer Systems*, vol.42, no.5, pp.1008-1014, 2021.

[2] Y. Fu, A. R. J. Downey, L. Yuan et al., Machine learning algorithms for defect detection in metal laser-based additive manufacturing: A review, *Journal of Manufacturing Processes*, vol.75, pp.693-710, 2022.

[3] Y. S. Balcioglu, B. Sezen, M. S. Gok et al., Image processing with deep learning: Surface defect detection of metal gears through deep learning, *Materials Evaluation*, vol.80, no.2, 2022.

[4] Z. Zhang, T. Shirai and T. Akiduki, Development of a surface defect inspection method and system by deep learning, *ICIC Express Letters, Part B: Applications*, vol.13, no.8, pp.827-835, DOI: 10.24507/icicelb.13.08.827, 2022.

[5] W. Fang, F. Zhang, V. S. Sheng et al., A method for improving CNN-based image recognition using DCGAN, *Computers, Materials and Continua*, vol.57, no.1, pp.167-178, 2018.

[6] Q. Y. Cheng, X. H. Duan and W. Zhu, Study on metal surface defect detection by improved YOLOv3, *Computer Engineering and Applications*, vol.57, no.19, pp.252-258, 2021.

[7] H. X. Huang and X. Jin, Small target defect detection based on YOLOv4, *Electronics World*, pp.146-147, 2021.

[8] J. Hu, S. Li, S. Gang et al., Squeeze-and-excitation networks, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2017.

[9] B. Alexey, C. Y. Wang and H. Y. M. Liao, Optimal speed and accuracy of object detection, *arXiv Preprint*, arXiv: 2004.10934, 2020.

[10] Y. Liu, Q. Wang, H. Zhang, Y. Liu and K. Zhao, Real-time defect detection of metal surface based on improved YOLOv4, *International Journal of Innovative Computing, Information and Control*, vol.18, no.4, pp.1329-1338, DOI: 10.24507/ijicic.18.04.1329, 2022.

[11] Z. Ma, Y. Li, M. Huang, Q. Huang, J. Cheng and S. Tang, A lightweight detector based on attention mechanism for aluminum strip surface defect detection, *Computers in Industry*, vol.136, DOI: 10.1016/j.compind.2021.103585, 2022.

[12] X. Xie, C. Li and M. Xu, Application of attention YOLOv4 algorithm in metal defect detection, *IEEE International Conference on Emergency Science and Information Technology (ICESIT)*, pp.465-468, DOI: 10.1109/ICESIT53460.2021.9696808, 2021.

[13] M. Sandler, A. Howard, M. Zhu et al., MobileNetV2: Inverted residuals and linear bottlenecks, *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[14] A. Howard, M. Sandler, G. Chu et al., Searching for MobileNetV3, *Proc. of the IEEE/CVF International Conference on Computer Vision*, pp.1314-1324, 2019.

[15] M. Tan, R. Pang and Q. V. Le, EfficientDet: Scalable and efficient object detection, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.10778-10787, DOI: 10.1109/CVPR42600.2020.01079, 2020.

[16] Z. P. Zhao, J. G. Li and Y. Y. Pu, Clear processing of low illumination images based on improved Retinex, *Computer Applications and Software*, vol.38, no.8, pp.220-226, 2021.

[17] S. Ren, K. He, R. Girshick et al., Faster R-CNN: Towards real-time object detection with region proposal networks, *Advances in Neural Information Processing Systems*, vol.28, pp.91-99, 2015.

[18] W. Liu, D. Anguelov, D. Erhan et al., SSD: Single shot multibox detector, *European Conference on Computer Vision*, pp.21-37, 2016.

[19] K. Han, Y. Wang, Q. Tian et al., GhostNet: More features from cheap operations, *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1580-1589, 2020.

Author Biography

Xikun Xie received his B.S. and M.S. degrees from Chongqing University of Science and Technology, Chongqing, China. He specializes in machine learning image processing and image generation adversarial networks. His current research interests include digital cores and digital twins.

Changjiang Li received his B.S. and M.S. degrees in mechanical design and theory from University of Science and Technology Beijing, China. He received his Ph.D. degree in electronic information from Toyohashi University of Technology, Japan. He is currently a professor with the School of Mechanical and Power Engineering, Chongqing University of Science and Technology, Chongqing, China. His research interests include solid waste resource utilization technology and equipment, machine vision. He has authored or co-authored over 40 papers in these areas. He received more than 15 patents.

Yang Liu received his B.S. degree from Chongqing University of Science and Technology, Chongqing, China. His research interests include digital twins and mechanical industrial simulation. He is currently a Master student in Chongqing University of Science and Technology and majors in image processing.

Junjie Song received his B.S. degree from Chongqing University of Science and Technology, Chongqing, China. His research interests include machine learning, image generation. He is currently a Master student in Chongqing University of Science and Technology and majors in numerical simulation.

Jonghyun Ahn received the Ph.D. degree in field robotics from the Kyushu Institute of Technology, Kitakyushu, Japan, in 2017. From April to September 2017, he was a Researcher with the Center for Socio-Robotic Synthesis, Kyushu Institute of Technology. Then, from October to March 2019, he was an Assistant Professor with the Department of Human Intelligence Systems, Kyushu Institute of Technology. Since April 2019, he has been an Assistant Professor with the Department of Intelligent Mechanical Engineering, in Hiroshima Institute of Technology, Hiroshima, Japan. His research interests include intelligent sensing and field robotics system.

Zhong Zhang received his Bachelor engineering, Master engineering degrees in 1982 and 1984, respectively, from Chang'an University, China and his Doctor engineering degree in 1993 from Okayama University, Japan. He was a visiting scholar at the University of Melbourne, Australia in 1998. He engaged in researches regarding intelligent system and signal, image processing as a senior researcher at the Industrial Technology Center of Okayama Prefecture, an Associate Professor at Okayama Prefecture University, a Professor at Toyohashi University of Technology, and now a Professor at Hiroshima Institute of Technology.