Pill Defect Detection Based on Improved YOLOv5s Network

AI Sheng[1], CHEN Yitao[1], LIU Fang[1], ZHU Aoxiang[2]

(1. *School of Mechanical Engineering and Automation*, *Wuhan Textile University*, *Wuhan* 430200;
2. *Yunmeng County Hospital of Chinese Medicine*, *Xiaogan* 432000)

Abstract: To address the problems of low detection accuracy and slow speed of traditional vision in the pharmaceutical industry, a YOLOv5s-EBD defect detection algorithm: Based on YOLOv5 network, firstly, the channel attention mechanism is introduced into the network to focus the network on defects similar to the pill background, reducing the time-consuming scanning of invalid backgrounds; the PANet module in the network is then replaced with BiFPN for differential fusion of different features; finally, Depth-wise separable convolution is used instead of standard convolution to achieve the output Finally, Depth-wise separable convolution is used instead of standard convolution to achieve the output feature map requirements of standard convolution with less number of parameters and computation, and improve detection speed. the improved model is able to detect all types of defects in tablets with an accuracy of over 94% and a detection speed of 123.8 fps, which is 4.27% higher than the unimproved YOLOv5 network model with 5.2 fps.

Keywords: Pill Defect Detection, Channel Attention Mechanism, Differentiation Fusion, Depth-separable Convolution

1 Introduction

The production of oral tablets may produce defects such as color, borders, contamination, scratches, markings, etc. These defects need to be rigorously tested and removed before the tablets are boxed. Current methods used to detect defects in tablets include the least squares method[1], the template matching method[2] and feature analysis[3], support vector machine method etc. Although these algorithms can detect defects, they require high image resolution and identify individual types of defects with low accuracy and slow speed.

Deep learning-based defect detection is widely used in steel, bridges, PCB boards, and other objects[4]: Li improved the YOLO network to achieve 97.55% mAP and 95.86% recall for the detection of small defects on steel surfaces[5]; The YOLOv3 method was used by Zhang to detect bridge defects and improve the detection accuracy through migration learning[6]; Li used the YOLOV4 method to detect surface defects on PCB boards and used clustering to obtain a better prior frame and improve detection accuracy[7] . Current YOLOv5 has higher detection accuracy and speed, and greater flexibility. Among them, YOLOv5s network has smaller models and faster detection speed in the detection scenario, which is more suitable for defect detection of oral tablets.

Aiming at the current situation and characteristics of oral tablet defect detection, this paper proposes a YOLOv5s-EBD defect detection algorithm: introducing a channel attention mechanism in the network to improve the detection accuracy of defects similar to the background; using BiFPN instead of PANet module on the neck network, fusing PANet deep and shallow feature layers in both directions to enhance the information transfer between different layers and improve the detection of the algorithm accuracy; and later using

depth-separable convolution to replace the standard convolution module, which can reduce the number of parameters and computation in the model and improve the detection speed.

2 Introduction of the YOLOv5 Network

The YOLOv5 network consists of a Backbone, a Neck and a Head[8]: The Backbone is used for fine-grained aggregation of different images to form image features; the Neck is used to enhance the features of different feature layers and to pass the features from bottom to top to the output; the Head is used for image prediction, generating bounding boxes and prediction categories, where the three prediction heads are responsible for predicting small, medium and large image targets[9]. The CBL module is a standard convolution layer, consisting of a convolution layer (Conv), Normalized layer (BN) and an activation function (Relu)[10]. The CSP module performs the convolution so that the number of channels is halved. This module mainly divides the feature map into two paths, one passing through the CBL module and Resunit module for further convolution, and the other directly convolution the two parts, which are finally stitched together by Concat to continue down the execution[11]. The YOLOv5 network framework is as shown in Fig.1.

3 YOLOv5s Network Improvements

3.1 Introduction of the Attention Mechanism Module

The YOLOv5s network is prone to false detection and over-detection of defect less regions for defects that are similar to the background of the pill. To address this problem, a channel attention mechanism network (ECANet) was introduced[12] that allows valid features to be focused on and invalid features to be suppressed.

The attention mechanism is as shown in Fig.2. The average feature map (1×1×C) is obtained by first using global average pooling (GAP) for each channel of the input feature map (W×H×C); then the 1-dimensional convolution is used to interact across channels (the size of the convolution kernel k is determined by the adaptive function), so that the number of channels in a larger layer can interact more across channels, and the sigmoid function is used to find the channel weights of the feature map; finally, the weights of each channel and the original input feature map are multiplied channel by channel to produce a weighted feature map, where different colors represent channels with different weights[13] .The network focuses on the channels with higher weights. The network focuses on the channels with higher weights. (where the k-value is calculated as shown in equation 1, where γ= 2 and b=1, C is the channel dimension).

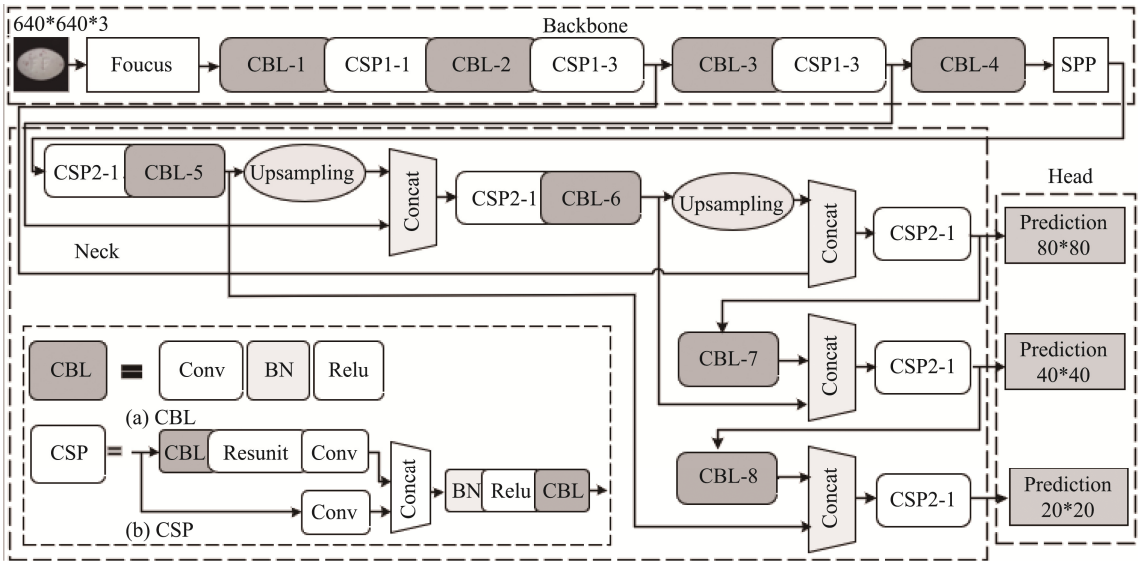


Fig.1 YOLOv5 Network Framework

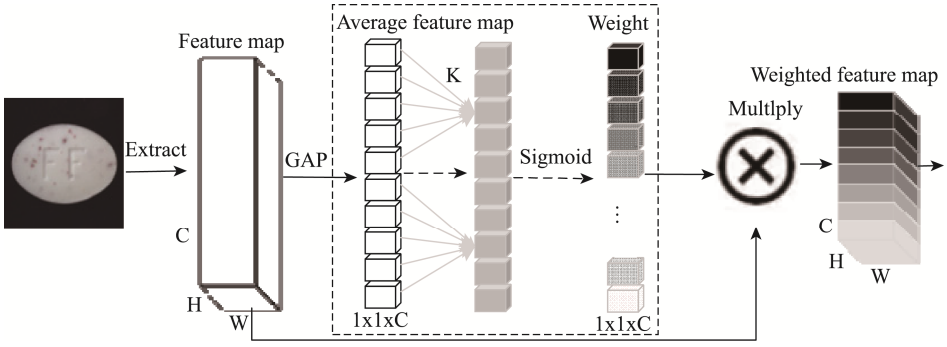


Fig.2 The Attention Mechanism Process

$$k = \Psi(C) = \left| \frac{\log_2(C)}{\gamma} + \frac{b}{\gamma} \right| \quad (1)$$

The best detection performance was found by placing the attention mechanism ECANet behind the CSP1-3 module in Backbone and the Concat module in Neck. The improvements in Backbone are as shown in Fig.3 and in Neck see YOLOv5s-EBD improvement figure.

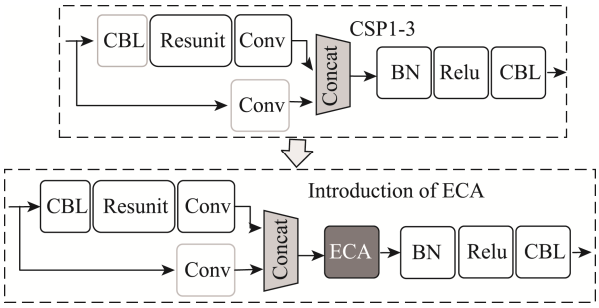


Fig.3 Introduction of Attention Mechanism in Backbone

3.2 Introduction of the BiFPN Module

The YOLOv5 network simply uses the Concat module in PANet to superimpose the feature maps without differentiating the features, resulting in its detection accuracy being compromised. The introduction of the weighted fusion mechanism BiFPN [13], which enhances higher-level feature fusion in processing paths, treats each bi-directional path as a feature network layer, learns the importance of different input features and performs differentiated fusion of different features[14].

Fig.4(a) shows the PANet module used in YOLOv5, which adds a bottom-up path fusion to the FPN so that the lower layer has higher semantic information while the upper layer gets higher positional information. The features are guaranteed to have both high positional information (for defect localisation) and semantic information (for defect classification). Fig.4(b) shows the BiFPN module, which has four improvements over PANet: firstly, the node with only one input edge is removed, which has little contribution to the fusion of different features, and its removal has little impact on the network, while simplifying the network model; secondly, without increasing the number of input nodes, the node leads from the input node to the output node in the same layer, ensuring that more features can be fused with less computation. Third, the BiFPN bidirectional path can be repeated many times, and the number of uses is calculated using NAS to add values to the network for adjustment to achieve better fusion of higher-level features; Fourth, the importance of learning different features is differentiated for different input features to be fused[15].

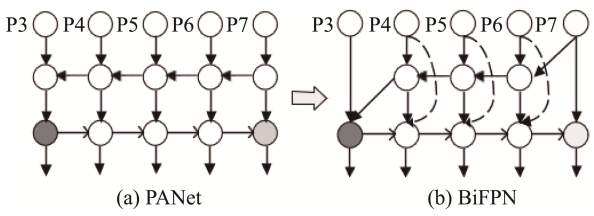


Fig.4 Improved Feature Fusion

From Fig.3 to Fig.7, they are the five effective feature layers extracted from the backbone network, with the arrows pointing to the upper and lower fusion of features. The fusion process sets weights to each node to balance the features at different scales. The thick arrow route indicates the top-down conveying high-level semantic information, while the thin arrow route indicates the bottom-up conveying low-level location information.

BiFPN adds additional weights to each input feature with the weighted fusion method shown in equation (2).

$$O=\sum_i \frac{w_i}{\varepsilon+\sum_j w_j}\cdot I_i \tag{2}$$

where w represents the weight parameter learned during the fusion process. The activation function is used so that $w_i \geqslant 0$, and ε is a small amount that keeps the value stable and has a value of 0.0001. Using P6 as an example, the calculation of cross-scale connectivity and weighted feature fusion in BiFPN is shown in equations (3) and (4).

$$P_6^{td}=Conv\left(\frac{w1\cdot P_6^{in}+w2\cdot \mathrm{Re}size(P_7^{in})}{w'1+w'2+\varepsilon}\right) \tag{3}$$

$$P_6^{out}=Conv\left(\frac{w'1\cdot P_6^{in}+w2\cdot P_6^{td}+w'3\cdot \mathrm{Re}size(P_5^{out})}{w'1+w'2+w'3+\varepsilon}\right) \tag{4}$$

Where P_6^{td} is the middle layer of the P6 feature fusion process, P_6^{in} and P_6^{out} are the input and output features of P6 respectively. Conv denotes the convolution process and resize denotes the up-sampling or down-sampling operation. The introduced BiFPN module mainly replaces the Concat module in Neck.

3.3 Introduction of Depth-separable Convolution Modules

In the industrial deployment of the pill defect detection model, the large model of the standard convolution can lead to slow industrial deployment and affect the speed of pill defect detection. Therefore, the standard convolution in the original network is replaced with a depth-separable convolution, which consists of Depth-wise Convolution for spatial filtering and Point-wise Convolution for feature generation[16-17].

3.3.1 Standard Convolution

Standard convolution directly selects the convolution kernel based on the output feature map, the flow is as shown in Fig.5. For example, for a 5*5*3 input feature map, if you want to get a 3*3*4 output feature map, you can directly use the 3*3*3*4 convolution kernel to convolution.

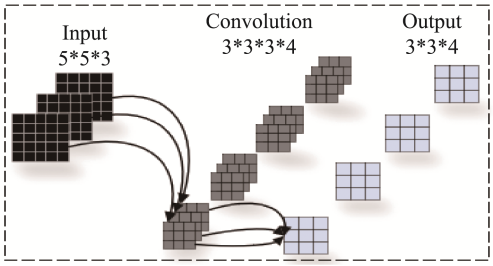


Fig.5 Standard Convolution

The number of standard convolution parameters and the amount of computation are calculated as shown in equations (5) and (6) [18].

$$N\_std=D_W\cdot D_H\cdot M\cdot N \tag{5}$$

$$C\_std=D_W\cdot D_H\cdot(I_W-D_W+1)\cdot(I_H-D_H+1)\cdot M\cdot N \tag{6}$$

where N_std and C_std represent the number of parameters and computation of the standard convolution; D_W and D_H represent the width and height of the convolution kernel; M and N represent the number of input and output channels; I_W and I_H represent the width and height of the input feature layer[19]. By feeding the data in the example into the above equation, the number of standard convolution parameters N_std is 108 and the amount of computation C_std is 972.

By the data in the example into the above formula, the number of standard convolution parameters N_std is 108 and the amount of computation C_std is 972.

3.3.2 Deeply Separable Convolution

The convolution first convolves each channel of the input feature map by depth-by-depth convolution, which does not change the depth of the feature map, and then increases the dimension of the input feature map by point-by-point convolution to obtain the same output feature map. As with the standard convolution example, for a 5*5*3 input feature map, a

depth-by-depth convolution of 3*3*3 kernels is used, followed by a point-by-point convolution of 1*1*3*4 kernels to obtain the same output feature map. The flow is as shown in Fig.6, with the depth-by-depth convolution flow on the left and the point-by-point convolution flow on the right.

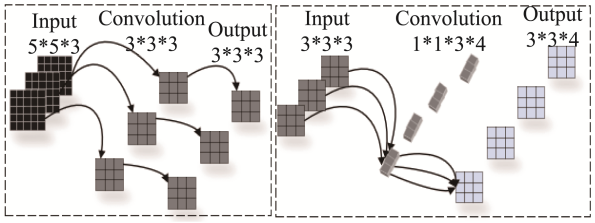


Fig.6 Deeply Separable Convolution

The depth separable convolution number of parameters and the amount of computation are calculated in two parts, and the formulae are shown in (7), (8), (9) and (10).

$$N\_dw = D_{W1} \cdot D_{H1} \cdot M \tag{7}$$

$$C\_dw = D_{W1} \cdot D_{H1} \cdot (I_W - D_{W1} + 1) \cdot (I_H - D_{H1} + 1) \cdot M \tag{8}$$

$$N\_pw = D_{W2} \cdot D_{H2} \cdot M \cdot N \tag{9}$$

$$C\_pw = D_{W2} \cdot D_{H2} \cdot I_W \cdot I_H \cdot M \cdot N \tag{10}$$

where N_dw and C_dw are the number of parameters and computations for the deep convolution, N_pw and C_pw are the number of parameters and computations for the point-by-point convolution; D_{W1} and D_{H1} are the width and height of the deep convolution kernel, D_{W2} and D_{H2} are the width and height of the point-by-point convolution kernel, and I_W and I_H are the width and height of the input feature layer[20]. By substituting the data in the example into the above formula, the number of the number of depth-by-depth convolution parameters N_dw as 27 and the computation C_dw as 243, and the number of point-by-point convolution parameters N_pw as 12 and the computation C_pw as 108.

It can be seen that the total number of deeply separable convolution parameters is 39, and the total calculated amount is 351, which are about 1 / 3 of the standard convolution. By using the depth-separable convolution instead of the standard convolution in CBL-3 and CBL-4, the network model can be reduced and the speed of defect detection can be improved, which is more conducive to the deployment of the algorithm model in industrial scenarios.

After implementing the above three methodological improvements, the YOLOv5s-EBD model was obtained. The framework diagram is as shown in Fig.7 where the dark gray modules are the improved modules.

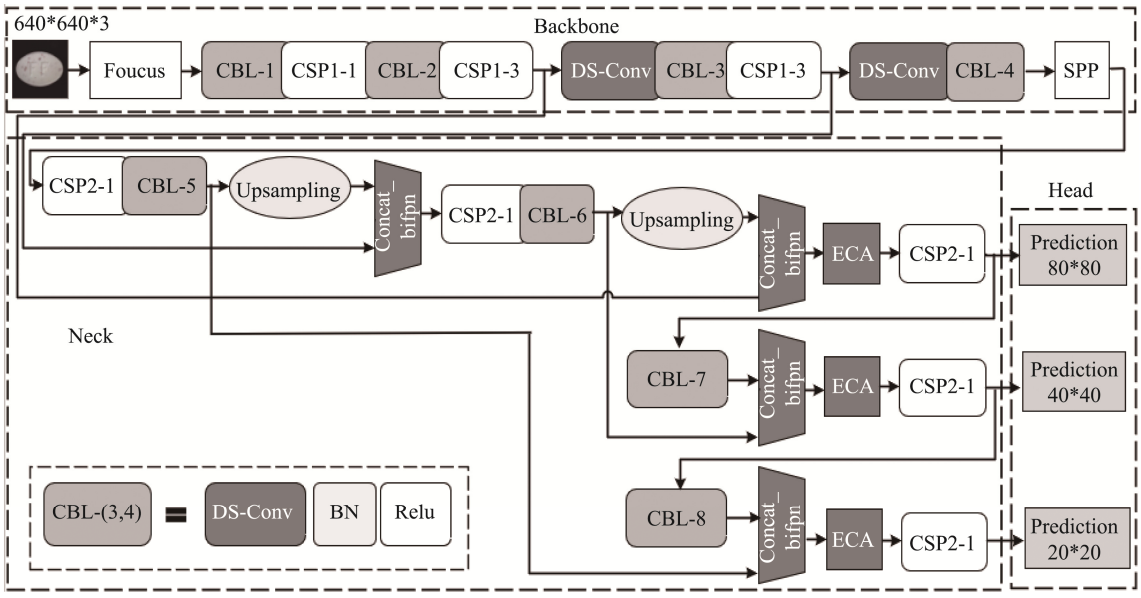


Fig.7 YOLOv5-EBD Network Framework

4 Data Processing and Training

4.1 Data Processing

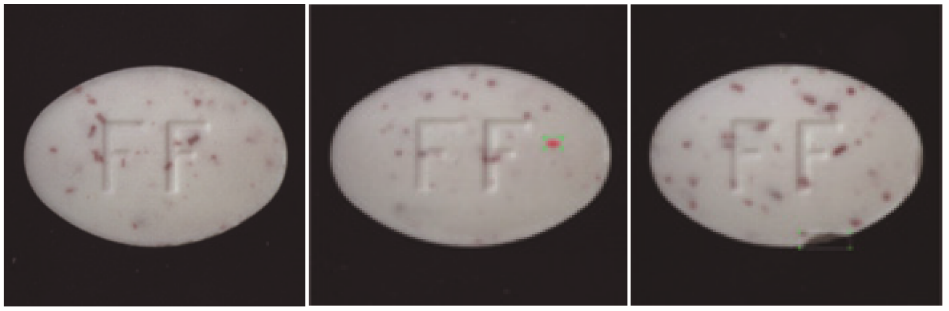
A vision acquisition system was set up on the tablet exit belt line at the pharmaceutical site. A Hikvision (MV-CS200-10GC) camera was used to take pictures of the tablets on the belt line for storage, ensuring that only one tablet was taken at a time by using light sources to fill in the light and controlling the flow rate of the tablets. After several months of on-site acquisition 53,800 pictures were obtained. From these, 1835 defect data were selected according to the pharmaceutical company's inspection quality requirements, including five categories of defects such as colour, borders, scratches, contamination and marks. In order to reduce training time and adapt YOLOv5s to network detection, the resolution of the collected images was cropped to 640 × 640. In order to improve the generalization capability of tablet defect detection, the data enhancement methods such as panning, flipping, scaling, changing colour difference and colour temperature were used to expand the images to 6400 in order to improve the generalization capability of tablet defect detection, see Table 1. Before network training, the defect data were randomly divided into 7:2:1 into three data sets, see Table 2. After the data was divided into three datasets, the labeling software LabelImg was used to manually label the location and type of defects, and the various types of defect labeling data are as shown in Fig.8.

Table 1 Defective Picture Statistics

Entry	Colour Pollution	Boundary Defects	Pharmaceuticals Pollution	Scratches Defects	Imprint Defects	Mixed Defects
Original Data Set	256	332	234	295	146	572
Extended Data Sets	1054	1096	1042	1076	1021	1111

Table 2 Dataset Statistics

Entry	Train	Validation	Test
Data Category	4480	640	1280



(a) Defective tablets (b) Color defect labeling (c) Boundary defect labeling

(d) Scratch defect labeling (e) Pollution defect labeling (f) Imprint defect labeling

Fig.8 Defect Labeling Data

4.2 Data Training and Testing

Data training and testing process is as shown in Fig.9 before training, the network parameters were set: the initial learning rate was set to 0.001 and decayed by 10% every 30 iterations; the SGDM gradient optimization algorithm was used, the momentum was set to 0.9, the decay factor was 0.0005, the batch size was 8, and the training Epoch was 200. [21] The training Epoch is 200. The YOLO network is then adjusted by using the loss function Loss to calculate the loss between the predicted value and the true value. The network decides whether to continue training based on the Epoch value, and the conditions are met to obtain the best weight file for data testing. In the testing process, the test images are fed into the trained YOLO network at to obtain defect images with a large number of predictor frames, which are then filtered using non-maximal suppression (NMS) to obtain the best defect detection image.

5 Evaluation Indicators and Test Results

5.1 Evaluation Indicators

In order to better detect tablet defects and to evaluate the performance of the detection model, five metrics were selected for evaluation: Precision, Recall, precision (AP), average precision (mAP) and F_1, which are calculated as shown in (11), (12), (13), (14), (15).

$$Precision = \frac{TP}{\mathrm{TP} + \mathrm{FP}} \times 100\% \tag{11}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \times 100\% \tag{12}$$

$$AP = \int_0^1 p(r)dr \tag{13}$$

$$mAP = \frac{\sum_{c=1}^{C} AP(\mathrm{c})}{C} \tag{14}$$

$$\mathrm{F_1} = \frac{2 \times \mathrm{Recall} \times Precision}{Recall + Precision} \tag{15}$$

where AP is the mean single defect accuracy value, while mAP is the mean of all APs and C is the number of detection categories. F_1 is used for the harmonic average of precision and recall, giving more weight to important values.

The training results obtained by training the defective data in the test set with the improved YOLOv5s-EBD model are as shown in Fig.10. The overall recognition accuracy exceeds 0.8, and the recall and average precision are close to 1.0.

5.2 Test Results

5.2.1 Results of Testing for Various Types of Defects

In order to verify the effectiveness of the improved model in detecting various types of defects, the data of 1280 pill defects in the test set were tested. The detection results for each defect are shown in Fig.11 and the statistics of the detection results for different defects are shown in Table 3.

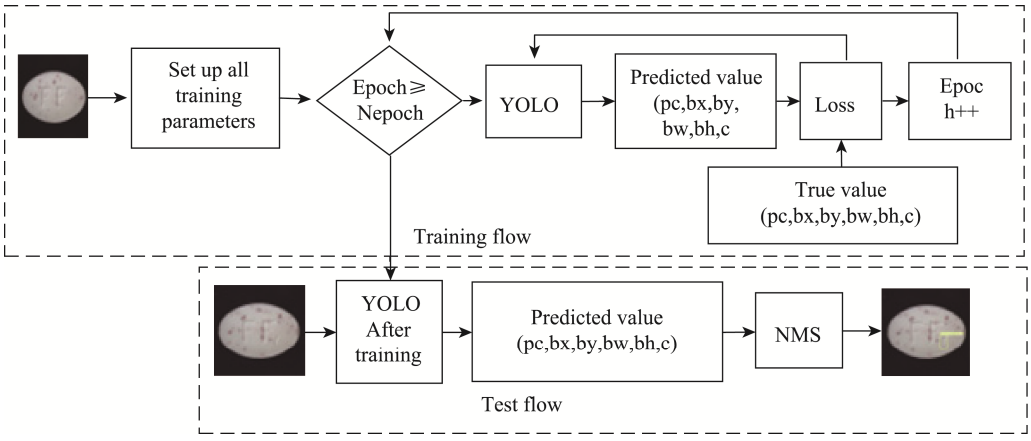


Fig.9 Training and Testing Process

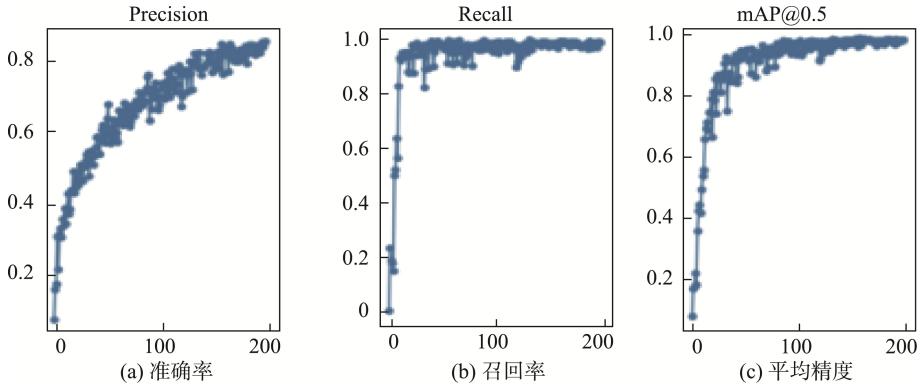


Fig.10 Training Results Data

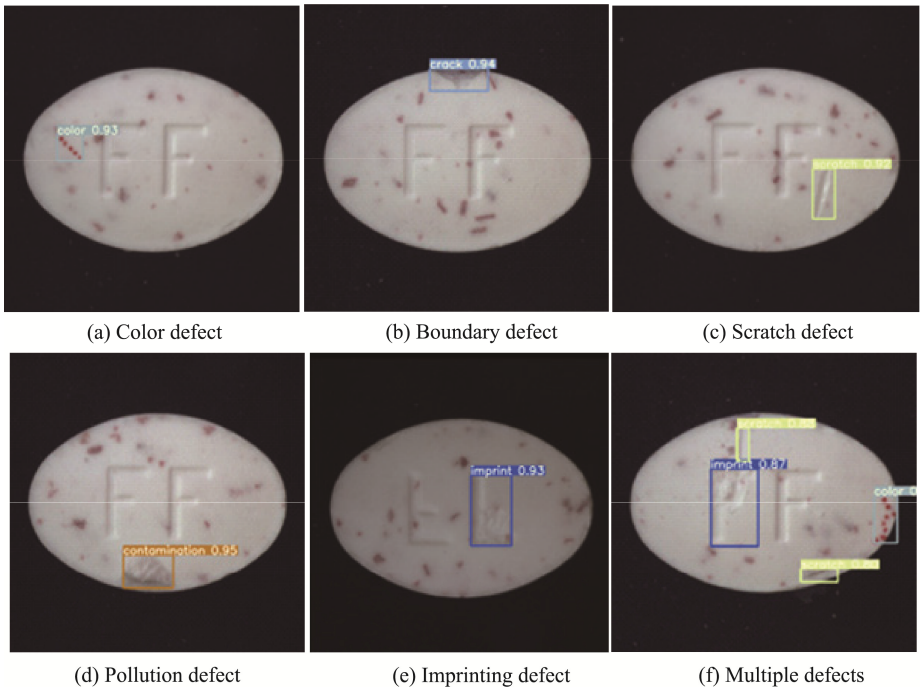


Fig.11 Tablet Defect Detection Display

Table 3 Statistics for Different Defect Detection Results

Tags	Colour Defects	Boundary Defects	Contamina-tion Defects	Scratch Defects	Impression Defects	False Detection	Precis ion (%)	Recall (%)	AP (%)	F_1
Colour Defects	216	1	3	1	1	7	94.32	92.3	94.18	0.93
Boundary Defects	1	277	2	5	0	2	96.51	95.18	95.82	0.95
Contamination Defects	4	2	237	1	2	4	94.8	93.3	94.45	0.94
Scratch Defects	2	5	2	264	4	3	94.28	92.31	94.03	0.93
Impression Defects	0	0	1	4	202	5	95.28	93.95	95.21	0.94
Not Detected	11	6	8	11	6					
Average							95.04	93.41	94.74	0.94

Table 3 shows that the detection accuracy of the five types of defects is above 94%, and there are relatively few misclassifications among each other, and more focus on their respective missed detection. Among them, the Precision and Recall values of boundary defects and imprint defects are higher, which is because the areas where they appear are relatively fixed, and the defect features are easier to detect. The relatively low Precision and Recall values of color defects and scratch defects, resulting in low AP and F1 values, are due to their unfixed defect locations, large shape randomness, and large size differences, which make the detection more difficult.

5.2.2 Test Results for Each Detection Model

To further illustrate the detection performance of this improved model in this study, comparative experiments were conducted between the improved model and other single-stage target detection models, including YOLOv4, YOLOv5s, YOLOv5m and SSD. All other conditions were the same in the experiments, and the results are shown in See Table 4.

Table 4 Experimental Results for Different Models

Models	Precision (%)	mAP (%)	Params (MB)	FPS (fps)
SSD	84.23	83.43	99.4	42.8
YOLOv4	90.23	90.14	246.7	46.3
YOLOv5s	91.16	90.61	14.2	118.6
YOLOv5m	92.90	92.32	41.6	96.8
YOLOv5s-EBD	95.04	94.74	12.4	123.6

In the table, Params is the number of network parameters and FPS is the image processing speed of the model. By comparing the five groups of experimental results in Table 4, the improved YOLOv5s-EBD model in this paper is superior to other models in all evaluation indicators. In particular, the advantages are most obvious compared with the SSD model. Compared with the unimproved YOLOv5s model, the accuracy and average precision are improved by more than 3%, and the model is smaller, easier to deploy in the production line, and has faster detection speed.

5.2.3 Test Results for Each Improvement Indicator

In order to verify the importance of each module, four sets of experiments are set up for comparison. They are the unmodified YOLOv5s model, the YOLOv5s-E model with the introduction of the attention mechanism ECANet, the YOLOv5s-EB model with PANet replaced by BiFPN in YOLOv5s-E, and the YOLOv5s-EBD model. Other conditions were the same for all four sets of experiments and the results obtained are shown in Table 5.

By comparison of YOLOv5s and YOLOv5s-EB in the table shows that although the improved model improves AP, mAP and F1 by 4.31%, 4.18% and 0.04 respectively, it also leads to an increase in the number of model parameters, a decrease in image processing rate and an overall performance impact. The introduction of the depth-separable convolution improvement, reduced the model size to 12.4 MB and increased the FPS to 123.8 fps. The three improvements resulted in a significant improvement in all metrics of the YOLOv5s-EBD compared to the YOLOv5s model.

6 Conclusion

In this paper, deep learning techniques are applied to pill surface defect detection, and an improved YOLOv5s-EBD model is proposed. The channel attention mechanism is introduced in the network, and the PANet in the network is replaced with a BiFPN module to make the network focus more on the defect features and improve the detection accuracy of the network; later, the size of the network model is reduced

Table 5 Statistics on the Results of the Four Experimental Groups

Models	ECANet	BiFPN	DSC	AP(%)	mAP(%)	F1	Params(MB)	FPS (fps)
YOLOv5s				91.11	90.61	0.91	14.2	118.6
YOLOv5s-E	+			93.36	92.83	0.93	21.3	109.4
YOLOv5s-EB	+	+		95.42	94.79	0.94	27.5	102.3
YOLOv5s-EBD	+	+	+	95.38	94.74	0.94	12.4	123.8

through convolution replacement to improve the defect detection speed. After experimental testing, the AP was 95.38%, F1 was 0.94, mAP was 94.74% and FPS was 123.8fps. The improved model in this paper can effectively achieve defect localization and identification. Compared to the original YOLOv5s model, AP, mAP, F_1 and FPS were improved by 4.27%, 4.13%, 0.03 and 5.2fps respectively. The model is also applicable to other tablet defect detection in the industry.

References

[1] Xu J, Yue Qiuyan, Ren X, Wang Q. Design of machine vision-based pill surface defect recognition and sorting system[J]. Sensors and Microsystems, 2017, 36(06): 90-93.

[2] Mozina M, Tomazevic D, Pernus F, et al. Automated visual inspection of imprint quality of pharmaceutical tablets[J]. Machine Vision and Applications, 2013, 24(1): 63-73.

[3] Gu Z, Tang Q, Li Z. Machine vision-based detection method for blister pill plate through-bubble defects[J]. Packaging and Food Machinery, 2019, 37(04): 58-63.

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

[5] Li J, Su Z, Geng J, et al. Real-time detection of steel strip surface defects based on improved YOLO detection network. IFAC PapersOnLine, 2018, 51: 76 -81.

[6] Zhang C, Chang C, Jamshidi M. Concrete bridge surface damage detection using a single-stage detector. Comput-Aided Civil Infrastruct Eng, 2020, 35: 389 -409.

[7] LI Chengfei, CAI Jialun, QIU Shihan. PCB defect detection based on improved YOLOv4 algorithm[J]. Electronic Measurement Technology, 2021, 44(17): 146 -153.

[8] Wen F, Chen Y. Research on surface defect detection technology of electronic components based on improved YOLOv4[J]. Journal of Shenyang University of Technology, 2021, 40(02): 1-7.

[9] Ying Z, Lin Z, Wu Z, et al. A modified-YOLOv5s model for detection of wire braided hose defects[J]. Measurement, 2022(190-): 190.

[10] Dong YF, Yang Y, Wang LQ. A semantic segmentation method for images based on multi-scale feature extraction and fully connected conditional random fields[J]. Advances in Lasers and Optoelectronics, 2019, 56(13): 109-117.

[11] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia Path Aggregation Network for Instance Segmentation IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018), pp. 8759-8768.

[12] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, Q. Hu ECA-Net: Efficient Channel Attention for Deep Convolutional Neural Networks[C] IEEE/CVF Conf. Computer Vis. Pattern Recogn. (CVPR)., 2020 (2020), pp. 11531-11539.

[13] Chen Z, Zhang H, Zeng N Yin, Li H. Multi-scale recovery of blurred images with fused attention mechanism[J]. Chinese Journal of Graphics, 2022, 27(05): 1682-1696.

[14] Tan, M.; Pang, R.; Le, Q.V. Efficient Det: Scalable and Efficient Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13-19 June 2020; pp. 10778-10787.

[15] WANG Y, CHEN G, HE C, HAO T, MA J. Intelligent recognition of scanning electron microscopy abrasive grain images based on improved YOLOv4[J/OL]. Journal of Friction: 1-20[2022-12-09].

[16] Du, F.J.; Jiao, S.J. Improvement of lightweight convolutional neural network model based on YOLO algorithm and its research in pavement defect Detection. *Sensors* 2022, *22*, 3537.

[17] Chollet F. Xception: deep learning with depth wise separable convolutions [C] //Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1251-1258.

[18] Wang Y, Zhao W, Xu H, Liu J. Non-fixed scene weather recognition algorithm based on lightweight convolutional neural network[J]. Electronic Measurement Technology, 2019, 42(17): 152-156.

[19] Cai F, Zhang Y, Huang J. A bridge surface crack detection algorithm based on YOLOv3 and attention mechanism[J]. Pattern Recognition and Artificial Intelligence, 2020, 33(10): 926-933.

[20] Li J, Li J, Duan Y, Ren G, Shi W. Deep learning-based detection of pipeline yarns and their colours [J]. Computer System Applications, 2021, 30(06): 311-315.

[21] Chen T, Zhou M, Han Q, Zhang X, Mao YB. Substation defect detection based on improved YOLOv4[J]. Computer System Applications, 2022, 31(06): 245-251.

Author Biographies

AI Sheng majored in defect detection at Wuhan Textile University.

E-mail: 1396914686@qq.com

CHEN Yingtao received Ph.D. from Huazhong University of Science and Technology (HUST) in 2005. He is currently a professor and Master's Advisor in HUST. His main research interests include precision blanking technology and automation, measurement, etc.

E-mail: to_cyt@163.com