WEP Warwick Evans Publishing

International Journal of Computer Science and Information Technology

ISSN: 3005-9682 (Print), ISSN: 3005-7140 (Online) | Volume 4, Number 1, Year 2024
DOI: https://doi.org/10.62051/ijcsit.v4n1.02
Journal homepage: https://wepub.org/index.php/IJCSIT/index

Research on LCD Defect Detection Algorithm Based on YOLOv8s

Qi Li [1], Yan Shi [1], Yinghua Liao [1], Chengye Shu [2], Wei Tang [1]

[1] College of Light and Chemical Technology, Mechanical Engineering University, Yibin Sichuan 644000, China

[2] Sichuan Jin-glong Optoelectronic Technology Co., LTD, Yibin Sichuan 644000, China

ABSTRACT

Aiming at the problems of small LCD display defects and low contrast, which are easy to be confused with background and lead to unsatisfactory detection effect, a liquid crystal display defect detection algorithm based on YOLOv8s was proposed. Firstly, a new CBLGhost module is pro-posed to facilitate the design core of GhostNet network, which uses the operation of convolution and linear change to generate the feature map, effectively reducing the computing resources re-quired by the model. Secondly, the HorNet module is introduced into the neck network to realize the modeling of high-order spatial interaction, and improve the recognition ability of the model for tiny features. Finally, CFNet module is introduced to balance the proportion of parameters between backbone network and fusion module network, so as to reduce the number of algorithm parameters and improve the detection speed of the algorithm. The experimental results on the self-made LCD defect data set show that the proposed algorithm can improve the detection ac-curacy without sacrificing FLOPs. Compared with the original algorithm, the accuracy is signif-icantly improved, with mAP reaching 93.7%, an increase of 3.8%. Compared with the mainstream target detection algorithms, the results show that the proposed algorithm has better performance in detecting the display defects of LCD.

KEYWORDS

Defect detection; YOLOv8; LCD; Cascade fusion network

1. INTRODUCTION

In Liquid Crystal Display (LCD) is a popular display device with the advantages of high resolution, no geometric distortion, thin and light, and low power consumption, which is used in smartphones, digital cameras, flat panels, monitors and other electronic devices [1]. The complex manufacturing process causes LCD defects such as strobe, colour difference, screen light leakage and Mura defects, which reduces the qualification rate of LCDs shipped from the factory. With the gradual improvement of consumer experience requirements, the need for defect detection is becoming more and more prominent, and it plays an increasingly important role in the quality control of LCD screens.

Currently the main popular inspection methods are: manual inspection and machine vision inspection [2]. In manual inspection, not only the labor cost is exponentially higher, not conducive to large-scale production, and high-resolution LCD single liquid crystal particles as small as a few tens of microns [3], the detection of such small defects has reached the limit of naked-eye detection, the human eye is difficult to directly find small defects, and can not be accurate judgment of the results of the detection of small defects. In terms of machine vision detection, the algorithmic framework based on deep learning [4] to realize online detection with higher accuracy and efficiency is a major

direction for defect detection. Qian [5] et al. proposed a fast method for sub-pixel level defect detection in LCD based on improved gradient algorithm in response to the problem of low defect detection efficiency. Tao [6] et al. proposed a novel DeepScratchNet, which realizes automatic weak scratch detection by aggregating rich multidimensional features for scratch representation. Du [7] et al. proposed an end-to-end spatio-temporal feature extraction and target detection framework based on the inter-frame energy accumulation (IFEA) enhancement mechanism, which improves the feature extraction capability of tiny defects. Wang [8] et al. proposed a feature attention pyramid-based target detection method (FAPNet), which enables to strengthen the target region features and weaken the background region features, and nearly improves the detection accuracy of tiny defects. For tiny defect detection, deep learning-based defect detection algorithms have become mainstream [9].

In this paper, in order to improve the detection performance of small defects on LCD screen, an improved LCD defect detection algorithm of YOLOv8s (You Only Look Once Version 8) is proposed. The work of this paper is summarized as follows: in this paper, at the backbone network, a new CBLGhost module is designed, and the CBL module is replaced by the CBLGhost module, which effectively reduces the computational resources required by the model; at the neck network, the convolutional block of the cross-stage local area network (CSPLayer_2Conv, C2f) is replaced by the HorNet module, which realizes the modeling of the higher-order spatial interactions, and improves the model's performance on the micro-defects. modeling, which improves the model's ability to recognize tiny features; at the backbone network, the C2f module is replaced by the Cascade Fusion Network (CFNet), which balances the parameter ratio of the backbone network and the fusion module network, thus reducing the number of algorithm parameters and improving the detection speed of the algorithm.

2. MATERIALS AND METHODS

2.1. Image Dataset Construction for LCD Defects

The image data set acquisition device for LCD defects is shown in Fig. 1, which is mainly composed of an industrial camera, a lightening device, and a test box for lighting up the LCD screen. In this paper, the LCD defects are divided into three major categories according to their shapes, which are line defects, point defects and area-like defects. The image format is PNG with a resolution of 3024 pixels × 3024 pixels. The images were collected under natural conditions with different angles and a total of 800 image samples were collected. Insufficient diversity of image samples can lead to overfitting in network model training. To prevent the overfitting phenomenon, this paper adopts cutting, rotating, mirroring, and contrast transformation methods on the collected image samples to increase the number of image samples from 800 to 2,674, and the various types of defective samples as well as the number of samples are shown in Figure 2. Using the LabelImg annotation tool to label the LCD defective data set, the labelled data set is randomly disrupted, and then divided into training set, validation set, and test set according to the ratio of 8:1:1.

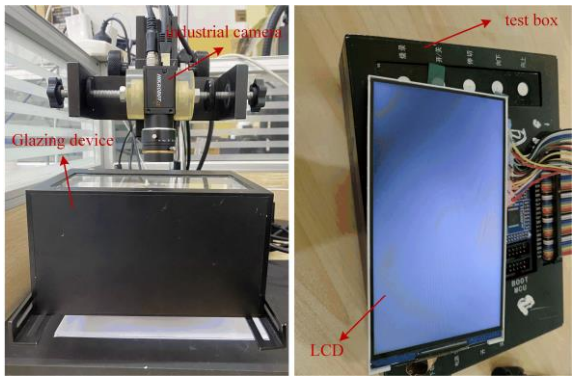
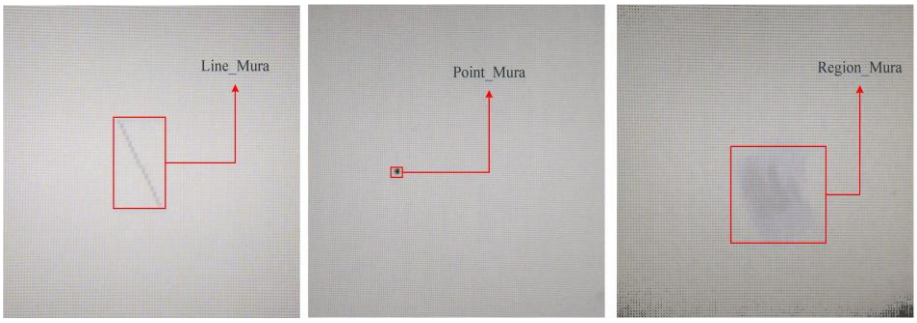
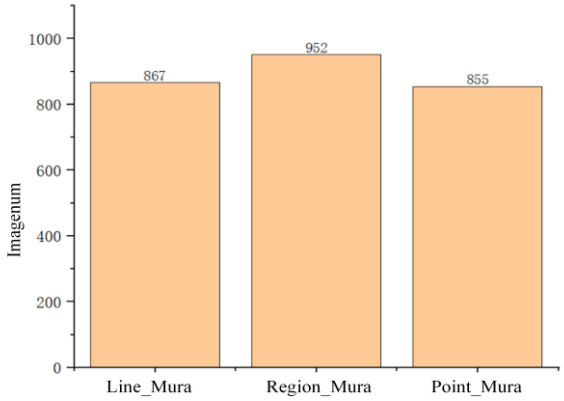


Figure 1. LCD defect data set acquisition device



(a) Sample pictures of LCD defects



(b) Types and quantities of LCD defective labels

Figure 2. LCD defect sample picture and number

2.2. Network Modeling and Improvements

2.2.1. Improved YOLOv8 target detection model

The YOLO (You Only Look Once) family of algorithms [10, 11], as one of the classic single-stage detection algorithms, has been upgraded to YOLOv8. Compared with the previous networks, the YOLOv8 algorithm improves the detection accuracy and real-time performance while reducing the number of network parameters. Therefore, YOLOv8s algorithm is selected as the base model of this paper. However, in the actual detection process, YOLOv8s algorithm still has some problems. First, the YOLOv8s algorithm uses a large number of standard convolution and C2f modules, which improves the accuracy of the algorithm, but reduces the running speed and increases the number of parameters of the model. Second, the YOLOv8s algorithm is not ideal for detecting such tiny and low-contrast point defects and line defects on the LCD screen, which are easy to be confused with the background, and is prone to leakage and misdetection.

Aiming at the above problems, this paper optimizes the YOLOv8s algorithm, and its main optimizations are as follows:(1) For the YOLOv8s algorithm uses a large number of standard convolutions, which increases the number of parameters of the algorithm. In this paper, at the backbone network (Backbone), the CBL module is replaced by the CBLGhost module, which uses half of the standard convolution operation with half of the lighter linear variation operation to generate a complete feature map, effectively reducing the computational resources required by the model. This is shown in Fig. 3(I). (2) The needle YOLOv8s algorithm is easy to be confused with the background when detecting point defects and line defects, resulting in unsatisfactory detection results. In this paper, at the neck network (Neck), the C2f module is replaced by the HorNet module, which realizes the modeling of higher-order spatial interactions and improves the feature recognition of the model for small defects, as shown in Fig. 3 (II). (3) The YOLOv8s algorithm uses a large number of C2f modules to improve the detection accuracy, which reduces the running speed of the algorithm and is

not conducive to the real-time detection. In this paper, some of the C2f modules are replaced by CFNet modules at the backbone network to balance the parameter ratio of the backbone network and the fusion module network, so as to reduce the number of algorithm parameters and improve the detection speed of the algorithm, as shown in Fig. 3 (III).

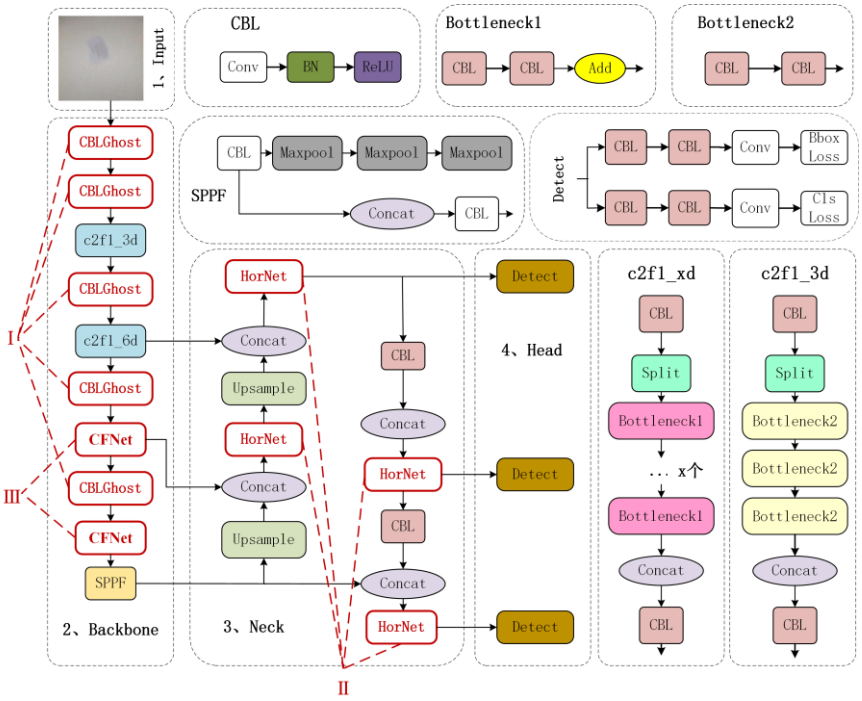


Figure 3. Network structure diagram of improved YOLOv8s algorithm

2.2.2. Introduction of the HorNet module

From the perspective of modelling spatial interactions, there are complex and often higher-order interactions between two spatial locations in a depth model, and the success of self-attention and other dynamic networks has shown that structural design that introduces higher-order spatial interactions is beneficial for improving the modelling of visual models. The standard convolution operation in YOLOv8s does not explicitly take into account spatial locations (i.e., the yellow features) and their neighbouring regions (i.e., the red regions) in terms of spatial interaction, as shown in Fig. 4(a). In this paper, recursive gated convolution ( ) is introduced, which uses gated convolution and recursive design, as shown in Fig. 4(b), to improve the modelling ability of the visual model and enhance the performance of detecting tiny defects through higher than second-order spatial interactions.

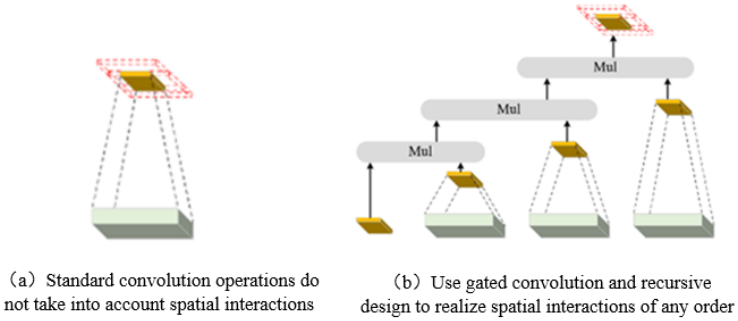


Figure 4. Schematic comparison between standard convolution operations and recursion-gate convolution operations

g^nConv principle is as follows: $x \in R^{HW \times C}$ is the input feature and a set of projected features P_0 and q_k are first obtained from Eq. (1)

$$[P_0^{HW \times C_0}, q_0^{HW \times C_0}, ..., q_{n-1}^{HW \times C_{n-1}}] = \Phi_{in}(x) \in R^{HW \times (C_0 + \Sigma_{0 \le K \le n-1} C_k)} \quad (1)$$

Where: Φ_{in} is a linear projection layer that performs channel blending, $(k=0,....,n-1)$.

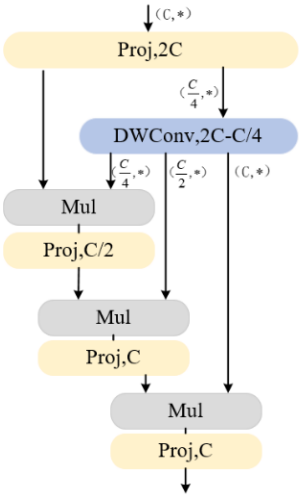
Performing the gated convolution in a further loop yields, in turn, P_{k+1}.

$$P_{k+1} = f_k(q_k) \odot g_k(p_k) / \alpha \text{ , k=0, 1, …, n-1} \tag{2}$$

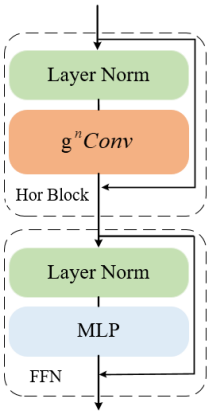
$$g_k = \{^{Identity, k=0}_{Linear(C_{K-1}, C_K), 1 \le k \le n-1} \tag{3}$$

Where: f is the depth convolution and g is used for dimensional alignment at different orders.

It can be seen that the order increases by 1 for each cycle, and X is able to model n-order spatial information interactions. The schematic of its X module is shown in Fig. 5(a). The HorNet module is built based on X. The basic blocks include using X as a spatial mixing layer and a feed-forward network (FFN), and its overall architecture is shown in Fig. 5(b). The HorNet module is highly flexible, compatible with various variants of convolution, and extends the second-order interactions in self-attention to an arbitrary order, which realizes modeling higher-order spatial information interactions and improves the performance of detecting tiny defects.



(a) $g^n Conv$ -module overall architecture diagram



(b) Overall Architecture Diagram of HorNet Module

Figure 5. $g^n Conv$ and HorNet module architecture diagram

2.2.3. Introduction of the CBLGhost module

The conventional feature extraction approach is to utilize multiple standard convolutions to perform convolutional mapping operations on all channels in the input feature map to produce a large number

of feature maps. However, stacking a large number of convolutional layers requires a huge amount of parametric counts and computation, which reduces the computational speed of the model. Therefore, in this paper, the design core of the GhostNet network [12] is introduced to propose a structure that can generate a large number of feature maps by only a small amount of computation, effectively reducing the computational resources required by the model, as shown in Fig. 6.

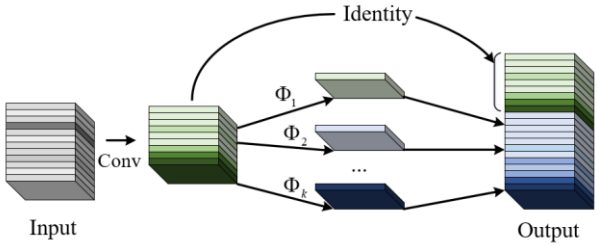


Figure 6. GhostNet design core diagra

As shown in Figure 7, the GhostConv module firstly uses the convolution operation of half the size of the original convolution to extract the rich feature information and generate half of the feature map, secondly, for the redundant feature information, it utilizes the lighter linear transformation operation to get the other half of the feature map, and finally passes the splicing operation to generate the final feature map.

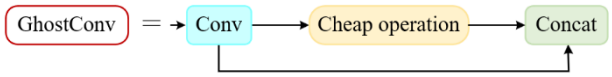


Figure 7. Overall architecture of GhostConv

As shown in Fig. 8, this paper designs a new CBLGhost module to replace the CBL module in the original network. Due to the introduction of the GhostConv module which reduces a large number of standard convolutions in the original structure, it greatly compresses the size of the network model, reduces the number of parameters and the amount of computation, and replaces the conventional convolution by the standard convolutional operation and the lighter linear change operation, which not only effectively reduces the computational resources required for the model but also does not affect the performance of the model. This not only effectively reduces the computational resources required by the model, but also does not affect the performance of the model.

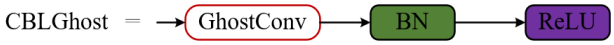


Figure 8. Overall architecture of CBLGhost

2.2.4. Introduction of a cascading converged network architecture

The network structure of YOLOv7 algorithm is to extract features by Backbone first, and then the multi-scale features in Backbone are fused by the feature fusion module. However, the number of Backbone network parameters is much larger than that of the feature fusion module, therefore, Cascade Fusion Net-work [20] (CFNet) is introduced in this to balance the parameter occupancy of Backbone and fusion module and reduce the number of algorithmic parameters [13]. The structure of CFNet is shown in Fig. 9, which mainly consists of two Block network structure blocks, two Downsampling blocks, Focus block and Aggregation block.

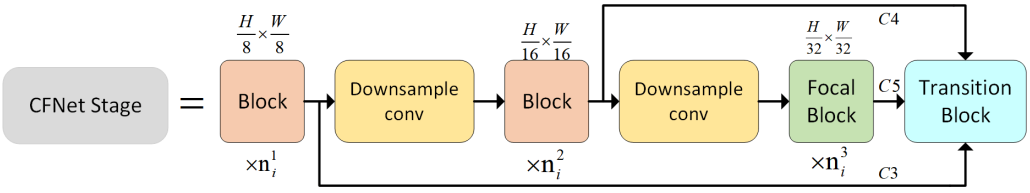


Figure 9. Cascade Fusion network structure

Block network structure is composed of Swin Transformer [14] structural blocks, the main role is to extract the deep semantic information, increase the sensory field, and does not perform downsampling operations on the feature map. The network structure diagram of the aggregation module for the CFNet Stage structure is shown in Fig. 10. C3, C4, C5 are used as inputs, and the aggregation module outputs P3, P4, and $conv-d_x$ denotes the 1×1 convolution with output channel number d_x, and 2x denotes up-sampling of features with scale factor 2. In this paper, C5 and C4 are used to perform the feature sum operation, and the result obtained aligns the spatial size of C3 then performs the feature sum operation with C3 to output the final result. The network structure diagram of the focus module of the CFNet Stage structure is shown in Fig. 11. N is the number of channels of the output features, D7×7 denotes the 7×7 depth convolution, r is the expansion rate of the additional convolution, and 1×1 denotes the 1×1 convolution. The main purpose is to add 7x7 depth separable convolutional layers in the yellow region, the use of depth separable convolution serves to reduce the number of references, and the 7x7 depth convolution serves to increase the network's perceptual field of view and extract rich semantic information.

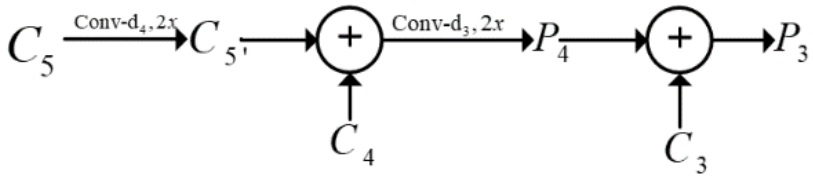


Figure 10. Transition Block network structure

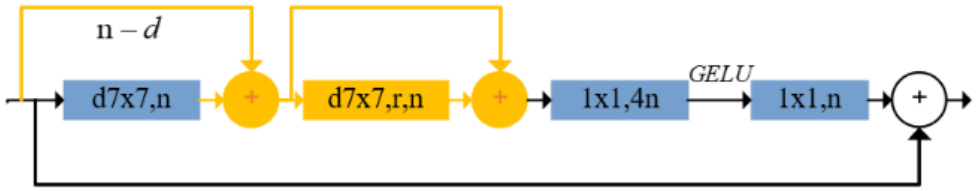


Figure 11. Focal Block network structure

3. EXPERIMENT AND RESULT ANALYSIS

3.1. Experimental Environment Configuration and Data Preparation

3.1.1. Experimental environment for network modeling

The GPU in the hardware environment of the experiment is NVIDIA 1660Ti; the RAM is DDR4 16GB. the operating system in the software environment of the experiment is Windows 10; the deployment environment is Python 3.9; the deep learning framework is PyTorch 1.12.0; and the accelerated computing architecture is CUDA 12.0.

3.1.2. Selection of hyperparameters for network models

The input image size in the experimental hyper-parameter settings is 640×640; the parameter learning-rate (learning-rate) is 0.01; the momentum (momentum) is 0.937; the weight-delay coefficient (weight-delay) is 0.0005; the batch size (batch size) is 8; and the number of iterations (Epoch) is 400.

3.1.3. Evaluation indicators for network models

This experiment uses six evaluation metrics: computational mean average precision (mAP), precision (P), evaluation accuracy (Average Precision, AP), recall (R), computational FLOPs reflecting the

computational complexity of the model, and detection speed (Frame per Second, FPS) as the judging criteria of the algorithm. Its calculation formula is as follows:

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP + FN}$$

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

$$A_{AP} = \int_0^1 P(R)\mathrm{d}R$$

FLOPs —— the amount of computation in response to the complexity of the modeling algorithm

Among them:

TP (True Positive) indicates that the detection target is correctly detected, FP (False Positive) indicates that the non-detection target is correctly detected, and FN (False Negative) indicates that the detection target is not correctly detected; in the P-R curve, the area surrounded by the P-R curve and the axes is equal to the magnitude of the AP value; i stands for the number of detections, n represents the number of number of images, and AP(i) represents the average accuracy of the ith category.

3.2. Training Results and Analysis

3.2.1. Impact of the HorNet module on the algorithm

To seek the optimal position of the HorNet module in the algorithm, three sets of experiments were conducted by replacing the C2f module with the HorNet module in the backbone network, the neck network, and the backbone and neck network of the algorithm, respectively. The results are shown in Table 1. The overall performance of the model is lower than that of the benchmark model with the introduction of the HorNet module in the backbone network of YOLOv8s, indicating that the feature extraction performance of the backbone network of the improved network is insufficient, resulting in a decrease in the accuracy of the model. The detection accuracy of the HorNet module is best at the combination of the backbone and the neck network of YOLOv8s, but the overall performance of the model is lower than the baseline model concerning that of the backbone and neck network with the introduction of the HorNet module in the neck network of YOLOv8s. The HorNet module also showed only a 0.1% increase in mAP and a 0.4% decrease in P, sacrificing 6.8% of FLOPs but with insignificant improvement. In contrast, the replacement of the neck network compared to the original network's P, R and mAP rose by 1.1%, 3.2% and 4.9%, respectively, and the identification of line defects that can be easily confused with the background is more effective, compared to the original network's 85.9% directly improved to 98.2%. The experimental results show that the best performance can be obtained by replacing the HorNet module with the C2f module in the neck network.

Table 1. Influence of optimization position of HorNet module on algorithm

	Location	P/%	R/%	mAP/%	FLOPs/G	AP/%		
						Line defect	Point defect	Regional defect
YOLOv8	-	91.5	88.8	89.9	28.7	85.9	86.2	97.6
HorNet	Backbone	94.4	83.9	89.7	30.8	85.9	85.6	97.7
	Neck	92.6	92	94.8	32.3	98.2	88.4	97.9
	Backbone+Neck	92.2	93.2	94.9	34.5	98.1	88.6	98.2

3.2.2. Impact of the CBLGhost module on the algorithm

To seek the optimal position of the CBLGhost module in the algorithm, three sets of experiments were conducted by replacing the CBL module with the CBLGhost module at the backbone network, the neck network, and the backbone and neck network of the algorithm, respectively. The results are shown in Table 2, where it can be seen that the CBLGhost module at the backbone network has a significantly better boosting effect than at the neck network, and the backbone and neck network. Compared with the replacement with the neck network, P and mAP are boosted by 1.4% and 2%, respectively, and FLOPs are reduced by 6.9%; compared with the replacement with the backbone and neck network, FLOPs are unchanged, but the algorithms P and mAP are boosted by 2.6% and 1.5%, respectively. Compared with the original network P and mAP are improved by 2.2% and 2.7%, respectively, and FLOPs are reduced by 7.3%.

Therefore, in this paper, the CBL module is replaced with the CBLGhost module at the backbone network of the algorithm.

Table 2. Influence of optimization position of CBLGhost module on algorithm

	Location	P/%	R/%	mAP/%	FLOPs/G	AP/%		
						Line defect	Point defect	Regional defect
YOLOv8	-	91.5	88.8	89.9	28.7	85.9	86.2	97.6
CBLGhost	Backbone	93.7	87.7	92.6	26.6	89.7	90.3	97.8
	Neck	92.3	87.3	90.6	28.2	86	88.7	97.9
	Backbone+Neck	91.1	89.5	91.1	26.4	86.9	87.8	98.5

3.2.3. Ablation experiment

To verify whether the improved algorithm proposed in this paper is effective, each improved network is compared with the original network in ablation experiments, and the results are shown in Table 4. First, the fusion experiments of HorNet module and CBLGhost module are conducted, as shown in Table 3 Experiment 4, compared with Experiment 3 (only HorNet module is added), the improvement effect of P, R and mAP is small, which is only 0.6%, 0.9% and 0.1%, but the FLOPs decreased by 5.9%, which indicates that this improvement reduces the model's without increasing the network's parameter overall size, allowing the network to understand the redundant information faster and faster. Next, the C2f module was replaced with the CFNet module, as shown in Experiment 5, the algorithm in Experiment 5 reduces FLOPs by 15.1% at the expense of accuracy relative to the original network, and the algorithm's detection speed becomes faster but the detection accuracy decreases, making it unsuitable for direct adoption. As shown in Experiment 6, the HorNet module, CBLGhost module and CFNet module are fused in the experiment, and the experimental results show that the FLOPs of the improved algorithm are unchanged compared with the original network, but the mAP of the improved algorithm improves by 3.8% and the R improves by 3%, which enhances the detection accuracy of the algorithm without sacrificing the number of parameters of the algorithm, and verifies that the improved algorithm is superior for performance enhancement of the superiority of the improved algorithm.

Table 3. Comparison of ablation experiments

	CBLGhost	HorNet	CFNet	P/%	R/%	mAP/%	FLOPs/G
1				91.5	88.8	89.9	28.7
2	√			93.7	87.7	92.6	26.6
3		√		92.6	92	94.8	32.3
4	√	√		93.2	92.9	94.9	30.5
5			√	94.8	81.8	87.2	25.9
6	√	√	√	91.3	91.8	93.7	28.7

To demonstrate the optimization effect of the improved module on the algorithm even further, the values of the corresponding pixel points of all channels of the feature map are averaged to produce the visualization results. As shown in Fig. 12(b) and (c), the YOLOv8-G algorithm with the addition of the CBLGhost module has a more concentrated region of interest to focus on the target defects compared with the YOLOv8s algorithm. It shows that the improved CBLGhost module not only effectively reduces the computer resources, but also improves the algorithm's defect detection performance. As shown in Fig. 12(d), the algorithm in this paper further focuses on the target defects and accurately depicts the region of interest of the defect features, which improves the detection performance of tiny defects.

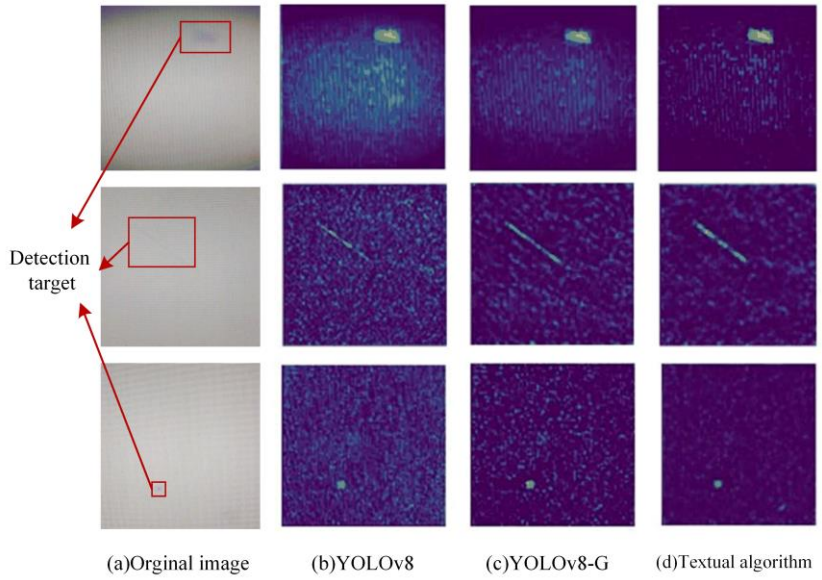


Figure 12. Network feature map visualization

The P-R curve results of the network before and after optimization are shown in Fig. 13 and Fig. 14, where the horizontal coordinate is R, the vertical coordinate is P, and the part of the curve surrounded by the horizontal and vertical coordinates is the detection accuracy. From the results of the P-R curves of this paper's algorithm and the original algorithm, it can be seen that the overall detection accuracy of the network using the optimized network has been improved by 3.8%. The greatest improvement is seen for line defects with an improvement of 12.5 %.

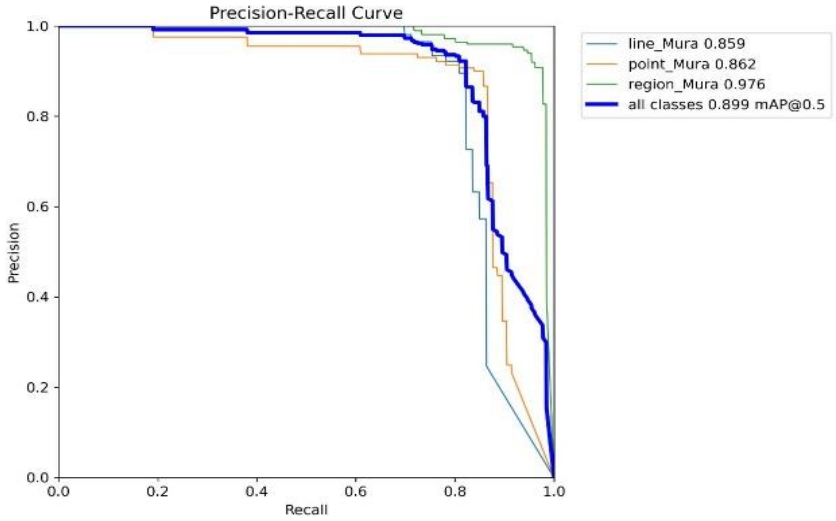


Figure 13. P-R results of the original algorithm

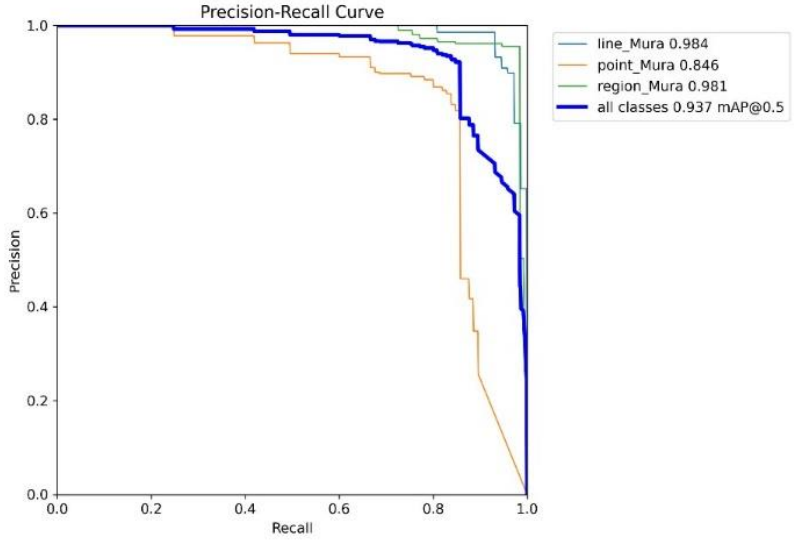


Figure 14. P-R results of the algorithm in this paper

3.2.4. Performance comparison with other algorithms

To ensure the fairness of the algorithm comparison, the YOLOv4s algorithm [15], YOLOv5s algorithm [16], YOLOv7-tiny algorithm, YOLOv7 algorithm [17], YOLOv8s algorithm and this paper's algorithm are trained respectively, under the unchanged parameter index. The experimental results are shown in Table 4, which shows that the mAP of this paper's algorithm is the best. the mAP of YOLOv4s algorithm, YOLOv5s algorithm and YOLOv7-tiny algorithm is reduced by 3.6%, 1% and 0.8%, respectively, relative to YOLOv8 algorithm, and although the mAP of YOLOv7 algorithm is 1.4% higher than that of YOLOv8 algorithm, the FLOPs are sacrificed by 72%, therefore, YOLOv8s algorithm is selected as the base algorithm in this paper. The comparison between this paper's algorithm and the YOLOv8s algorithm shows that the mAP of this paper's algorithm is improved by 3.8% and the FPS reaches 105, which indicates that the introduction of the HorNet module can improve the model's ability to recognize tiny features, and the introduction of the CFNet module can effectively improve the detection speed of the algorithm. Therefore, the improvement of this paper's algorithm is effective, and this paper's algorithm improves the detection speed of the algorithm while improving the accuracy.

Table 4. Performance comparison of different basic algorithms

Models	P/%	mAP/%	FLOPs/G	FPS
YOLOv4s	87.2	86.3	19.6	69
YOLOv5s	90.4	88.9	15.8	62
YOLOv7	92.1	91.3	105.2	45
YOLOv7-tiny	87.1	89.1	13.0	102
YOLOv8s	91.5	89.9	28.7	93
Textual algorithm	91.6	93.7	28.7	105

The improved model was utilized to train and test the dataset of LCD defects, and the detection results of the three types of defects are shown in Fig. 15. Among them, the locations of defects are marked by rectangular boxes, and different coloured rectangular boxes are used for different types of defects, and the probability of each type of defect being detected is generated in the upper left corner of the rectangular detection box, compared with YOLOv8s algorithm this paper's algorithm can better detect the location and type of defects, which indicates that the network has a better ability to detect the defects, and this shows that the introduction of the HorNet module is successful in improving the model s ability to recognize tiny features.

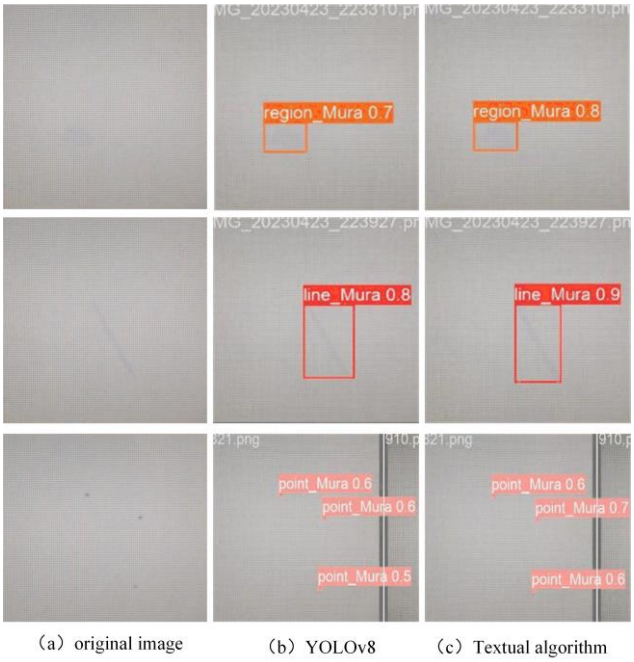


Figure 15. Comparison of detection results between the proposed algorithm and the original algorithm

4. CONCLUSION

In this paper, in order to improve the performance of the detection of small defects on LCD screens, accelerate the detection speed, and facilitate real-time detection. YOLOv8s-based LCD defect detection algorithm is proposed, based on the YOLOv8s model, the CBLGhost module is constructed, which generates feature maps through half of the standard convolution operation and half of the lighter linear change operation, effectively reducing the computational resources required by the model; YOLOv8s algorithm is easy to confuse with the background when detecting the point and line defects, resulting in the problem of unsatisfactory detection results. In order to solve the problem of unsatisfactory detection effect, the HorNet module is introduced to realize the modelling of high-order spatial interactions and improve the feature recognition ability of the model for small defects;

the CFNet module is introduced to balance the parameter ratio of the backbone network and the fusion module network, so as to reduce the number of algorithm parameters and improve the detection speed of the algorithm. Compared with other typical algorithms, the results show that using the algorithm in this paper improves the detection accuracy of the algorithm without sacrificing the number of algorithmic parameters, and compared with the original algorithm, the accuracy of the algorithm is significantly improved, and the mAP reaches 93.7%, which is an improvement of 3.8%. Further optimization of the network model is needed in future work to improve the detection accuracy.

REFERENCES

[1] Lin G, Kong L, Liu T, et al. An antagonistic training algorithm for TFT-LCD module mura defect detection [J]. Signal Processing: Image Communication, 2022, 107: 116791.

[2] Kesen, Wang Xiaoqiang, Lin Hao et al. A review of single-stage small target detection methods in Deep Learning [J]. Exploration of Computer Science and Technology, 202, 16(1):41-58.

[3] Chen Xin, Huang Dejun, Fang Chenggang, LI Shuaikang. Character defect detection of LCD screen based on multi-feature matching [J]. Packaging Engineering, 2023, 44 (03): 157-163.

[4] Athanasios V, Nikolaos D, Anastasios D, et al. Deep Learning for Computer Vision: A Brief Review [J]. Computational intelligence and neuroscience, 2018. (2018):7068349.

[5] Qian Jide, Chen Bin, Qian Jiye, et al. Sub-pixel defect detection of liquid crystal display based on improved gradient algorithm [J]. Journal of Computer Applications, 2017, 37(S1): 201-205.

[6] Tao X, Zhang D, Hou W, et al. Industrial weak scratches inspection based on multifeature fusion network [J]. IEEE Transactions on Instrumentation and Measurement, 2020, 70:1-14.

[7] Du J, Lu H, Zhang L, et al. A spatial-temporal feature-based detection framework for infrared dim small target [J]. IEEE Transactions on Geoscience and Remote Sensing, 2021, 60: 1-12.

[8] Wang Xili, LIANG Zhengyin, LIU Tao. Object detection method in remote sensing image based on feature attention pyramid [J]. Journal of Remote Sensing, 2023, 27(02): 492-501.

[9] Yang Honggang, Chen Jiejie, Xu Mengfei. Application of bilinear in-roll neural networks to fundus disease image classification [J]. Journal of Computer Applications, 2023, 43(1):259-264.

[10] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection [C]// Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 779-788.

[11] REDMON J, FARHADI A. YOLO9000: better, faster, stronger [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 7263- 7271.

[12] HAN K, WANG Y, TIAN Q, et al. Ghostnet: More features from cheap operations [C]// Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 1580-1589.

[13] Yifan Z, Wenhao Y, Yifeng Z, et al. CFNet: Conditional filter learning with dynamic noise estimation for real image denoising [J]. Knowledge-Based Systems, 2024, 284111320-.

[14] LIU Z, LIN Y, CAO Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows [C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. Virtual, 2021: 10012-10022.

[15] YU J, ZHANG W. Face mask wearing detection algorithmbased on improved YOLO-v4 [J]. Sensors, 2021, 21(9): 3263.

[16] SONG Q, LI S, BAI Q, et al. Object detection method forgrasping robot based on improved YOlOv5 [J]. Micromachines, 2021, 12(11):1273.

[17] WANG C Y, BOCHKOVSKIY A, LIAO H Y M. YOlOv7: trainable bag-of-freebies sets new state-of-the-art for realtime object detectors [C]// Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition. Washington D.C., USA: IEEE Press, 2023: 7464-7475.