

Article

Not peer-reviewed version

---

# PRC-Light YOLO: An Efficient Lightweight Model for Fabric Defect Detection

---

[Baobao Liu](#)\*, [Heying Wang](#), Zifan Cao, Yu Wang, Lu Tao, Jingjing Yang, Kaibing Zhang

Posted Date: 20 December 2023

doi: 10.20944/preprints202312.1579.v1

Keywords: Fabric defect detection; YOLOv7; Lightweight network; HardSwish; Wise-IOU v3



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

# PRC-Light YOLO: An Efficient Lightweight Model for Fabric Defect Detection

Baobao Liu <sup>1,2,\*</sup>, Heying Wang <sup>1</sup>, Zifan Cao <sup>1</sup>, Yu Wang <sup>1</sup>, Lu Tao <sup>1</sup>, Jingjing Yang <sup>1</sup>  
and Kaibing Zhang <sup>1</sup>

<sup>1</sup> School of Computer Science, Xi'an Polytechnic University, Xi'an 710048, China

<sup>2</sup> Shaanxi Key Laboratory of Clothing Intelligence, School of Computer Science, Xi'an Polytechnic University, Xi'an 710048, China

\* Correspondence: liubaobao1222@163.com

**Abstract:** Defect detection plays a pivotal role in quality control for fabrics. In order to enhance the accuracy and efficiency of fabric defect detection, we have proposed the PRC-Light YOLO model for fabric defect detection and have established a detection system. Firstly, we add new convolution operators in Backbone for the YOLOv7 and integrate them with Extended-Efficient Layer Aggregation Network. This combination constructs a new feature extraction module that not only reduces the computational effort of the network model, but also extracts spatial features effectively. Secondly, we employ multi-branch dilated convolutions feature pyramid, and introduce lightweight upsampling operators to improve performance of the feature fusion network. This module achieves an expanded receptive field by generating real-time adaptive convolution kernels, allowing for the collection of crucial information from regions with a richer contextual context. To further minimize the computation during network model training, we adopt the HardSwish activation function. Finally, we apply the Wise-IOU v3 bounding box loss function as a dynamic non-monotonic focusing mechanism, which decreases adverse gradients from low-quality instances. We conduct data augmentation on real fabric dataset to raise the generalization capability of the PRC-Light YOLO model. Compared with the YOLOv7 model, numerous simulation experiments show that our proposed methods reduce the model's parameters and computation by 18.03% and 20.53%, respectively. Simultaneously, there has been a 7.6% improvement in mAP.

**Keywords:** fabric defect detection; YOLOv7; lightweight network; HardSwish; Wise-IOU v3

## 1. Introduction

Fabric defect detection plays a significant role in industrial field. In the process of fabric manufacturing, the defects on the surface of fabric are inevitable. Therefore, the effective identification and precise localization of fabric defects have drawn the attention of scholars [? ].

Fabric defect detection can be mainly categorized into traditional methods and methods based on deep learning. The traditional methods include texture structure method, histogram statistics, spectral method, modeling method and adaptive dictionary learning method for fabric defect detection. Li et al. put forward a method based on saliency region and similarity localization detection to solve the problem of defect detection and to contour accurate segmentation of periodic textured fabric images[? ]. Xiang et al. proposed a defect detection algorithm based on Fourier convolution and convolutional self-encoder that avoids the conventional method of introducing noise in the training phase and employs a random masking method to generate image pairs for training instead[? ]. Wu et al. roposed a method of fabric defect detection based on K-Singular Value Decomposition dictionary learning[? ], in which the dictionary is trained using defect-free fabric samples and patch size to detect fabric defects. Kanwal et al. utilized the K-means algorithm to develop a bag-of-words model and proposed a significance-based bag-of-words model for fabric defect detection[? ]. This approach significantly improved the distinction between regions of images with and without defects. Although the detection speed and performance of the methods based on traditional image processing are better than those of

manual detection, the methods of generalization ability are inadequate. In past two decades, deep learning has become a trend for fabric defect detection, and the techniques have achieved significant advancements in this field[? ]. The detection methods based on deep learning can be divided into single-stage algorithms and two-stage algorithms. The single-stage algorithms mainly include Solid State Drive (SSD)[? ] and You Only Look Once (YOLO)[? ? ] series, the two-stage algorithms comprise R-CNN[? ], Fast R-CNN[? ], Faster R-CNN[? ] and so on. Zhao et al. utilized ResNet50 network and ROI Align to replace VGG16 features extraction network and the region of interest pooling layer in Faster R-CNN[? ]. At the same time, the softmax classifier is used to identify the image and obtain the prediction results. Zhang et al. introduced the channel attention mechanism in MobileNetV2-SSDLite and redefined the loss function by Focal Loss[? ]. Based on YOLOv5s, Zhou et al. used K-Means++ clustering to acquire the anchor frames, followed by combining the CARAFE upsampling operator and adding a small target detection layer in the detection head, which improved detection accuracy while retaining a certain computational efficiency[? ]. Wu et al. embedded classification-Aware regression losses into YOLOF to establish a relationship between classification and target localization[? ]. Lin et al. proposed a weighted bidirectional feature network by replacing the main module in the original YOLOv5 structure with Swin Transformer[? ]. In addition, they used the generalized focal loss to improve the learning of positive sample instances and reduce the missed detection rate. The methods mentioned above can achieve higher defect detection accuracy or the quantity of computation on the condition of the increased number of parameters in the original network.

In this paper, we research the structure of YOLOv7[? ] and propose Partial Convolution (PConv) to replace the  $3 \times 3$  Conv of Extended-Efficient Layer Aggregation Networks (E-ELAN) in the Backbone module. This replacement aims to reduce the number of parameters and network model calculation. In the Neck module, we import Receptive Field Block(RFB)[? ] and integrate Content-Aware ReAssembly of FEatures (CARAFE) lightweight upsampling operator. Thus, we have proposed the PRC-Light YOLO lightweight model. Moreover, the HardSwish activation function is used to curtail the computational cost and memory access of the network, and Wise-IOU v3 with dynamic non-monotonic focusing mechanism is applied as the bounding box loss function. The experiment verifies that PRC-Light YOLO has less parameters and calculation than the YOLOv7, which improve the performance of fabric defect detection.

The remaining sections of this article are organized as follows: In Section 2, we introduce the YOLOv7 model and its image processing. Section 3 presents the PRC-Light YOLO model, which encompasses the construction of lightweight modules, feature fusion network, optimization of activation functions and loss functions. Section 4 provides a detailed account of the experimental results and analysis. In Section 5, we construct fabric defect detection system. Section 6 discusses further research based on the experimental results. Finally, a conclusion is provided.

## 2. YOLOv7 Model Structure

The YOLOv7 model is mainly composed of three parts: Backbone, Neck, and Head, the network structure is shown in Figure ??.

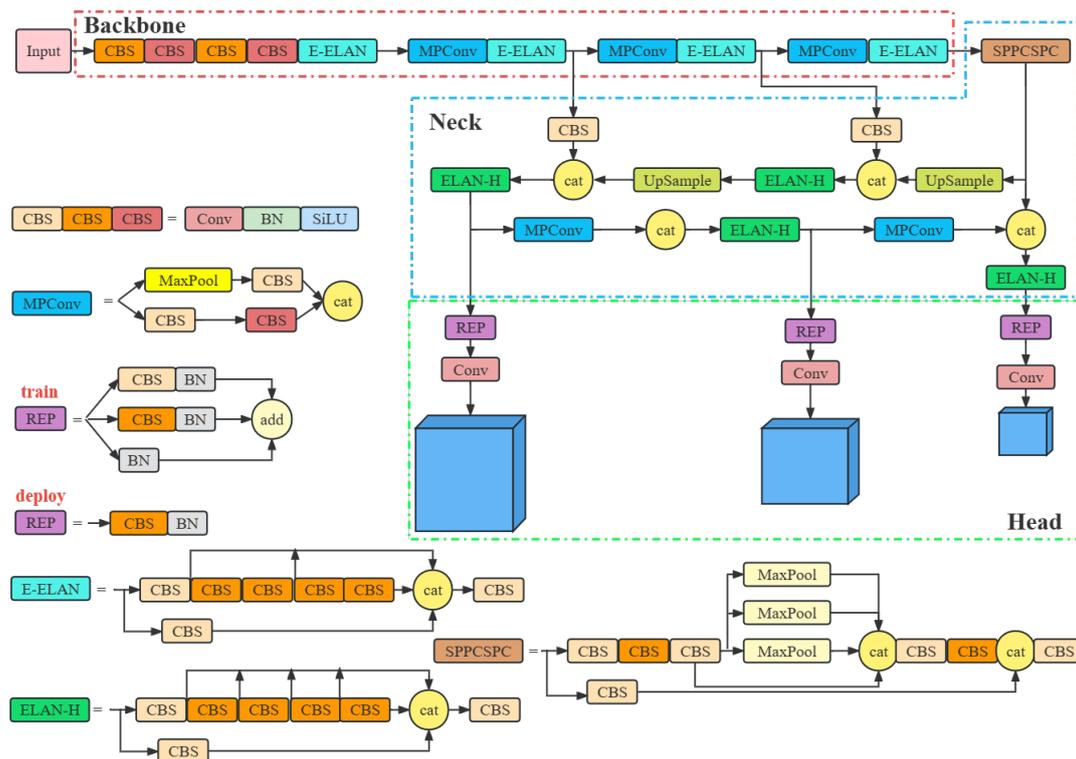


Figure 1. The network structure of YOLOv7.

### 2.1. Backbone

The Backbone part mainly comprises CBS(Convolution, Batch normalization, SiLU) layer, E-ELAN module, and MPCConv(Max Pool Convolution) module[? ]. Among them, different CBS colors signify varied convolution kernels. The MPCConv module is divided into upper and lower branches. In the upper branch, the feature map via MaxPool operation reduces spatial dimensions by half, followed by a CBS operation to halve the number of channels. In the lower branch, the feature map is processed by the first CBS to reduce the number of channels, followed by another CBS to further reduce the number of channels. Lastly, the upper and lower branches are tensor spliced. The E-ELAN module contains two branches[? ], and this module enhances the ability of the network to learn more features.

### 2.2. Neck

Path Aggregation Feature Pyramid Network (PAFPN) structure is used in the Neck module for feature fusion[? ]. This involves transmitting and fusing higher-level feature information through an upsampling process, followed by feature map predictions obtained through a down-sampling fusion method. The final output comprises results from three feature layers. As image processing can result in significant image distortion, the SPPCSPC module incorporates three parallel MaxPool operations during a series of convolutional operations, effectively reducing the occurrence of image distortion.

### 2.3. Head

In the head detection section, the REP module differs between the training and inference stages. During the training stage, it comprises three branches. While in the inference stage, the REP module undergoes reparameterization and consists of only a  $3 \times 3$  Conv.

Thus, fabric defect detection process based on YOLOv7 is outlined as follows. Firstly, before an image being input into the Backbone, the model performs a series of operations, including Mosaic data enhancement, adaptive anchor frame calculation, adaptive image scaling and so on. Secondly, the Backbone module extracts features from the processed image. Thirdly, the Neck module carries out

feature fusion processing for the features extracted from the Backbone, to get the feature maps of three sizes, large, medium, and small. Finally, the Head layer detects the fused features, and outputs the final results.

### 3. Fabric defect detection based on PRC-Light YOLO

Although YOLOv7 has excellent detection accuracy and speed on datasets such as COCO, PASCAL VOC[? ], its performance diminishes for fabric defect detection in fabric datasets. Thus, we study the theoretical foundation and network architecture of YOLOv7 and propose the PRC-Light YOLO for fabric defect detection model.

#### 3.1. Lightweight backbone network

The reduction of floating-point operations (FLOPs) can alleviate the computational burden of neural networks, shorten both forward and backward propagation times, and contribute to achieving low latency. Simultaneously, lower FLOPs can enhance the parallelism of neural networks on hardware, increasing throughput and enabling faster processing of input data streams. Therefore, to achieve lightweight neural networks while satisfying the requirements of low latency and high throughput, most researchers are committed to reducing the total number of FLOPs. This is because there exists the following relationship between neural network latency and FLOPs:

$$Latency = \frac{FLOPs}{FLOPS} \quad (1)$$

Floating-point operations per second (FLOPS) is a measure of effective computing speed. As indicated by the equation ??, reducing FLOPs does not necessarily lead to a decrease in neural network latency. Hence, it is essential to concurrently reduce FLOPs and optimize FLOPS to truly achieve low latency.

Chen et al. demonstrated that frequent memory accesses by convolution operations reduced FLOPS values in neural networks[? ]. MobileNet[? ], ShuffleNet[? ], and GhostNet[? ], for example, embed Depthwise Convolution (DWConv) or Group Convolution (GConv) to extract spatial features. For an input  $I \in R^{c \times h \times w}$ , with  $c$  convolution kernels  $W \in R^{k \times k}$ , get an output  $O \in R^{c \times h \times w}$ . Then the FLOPs for regular convolution is  $h \times w \times k^2 \times c^2$ , and the DWConv's FLOPs is  $h \times w \times k^2 \times c$ . Despite the lower FLOPs, the DWConv operation cannot be directly followed by the regular convolution operation, or it will lead to a significant degradation of model accuracy. To compensate for the drop in precision, the PointWise Convolution (PWConv) operation is typically employed after DWConv, with the channel count in DWConv being augmented to  $c$ . As a result, the Memory Access Cost (MAC) of DWConv is:

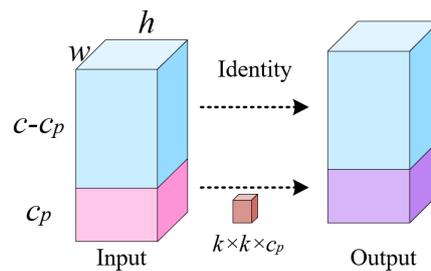
$$h \times w \times 2c' + k^2 \times c' \approx h \times w \times 2c' \quad (2)$$

The MAC for regular convolution is:

$$h \times w \times 2c' + k^2 \times c \approx h \times w \times 2c \quad (3)$$

A comparison of the equation ?? and the equation ?? suggests that the DWConv operation may elevate the frequency of memory accesses in the process of reducing FLOPs. Furthermore, since these models typically involve extra data processing operations such as splicing, disambiguation, and pooling, the inclusion of DWConv does not invariably curtail the neural network's latency.

Therefore, we adopt a novel partially convolution PConv in the Backbone, which reduces the FLOPs and MAC of neural networks. The principle of the PConv is illustrated in Figure ??.



**Figure 2.** The principle of PConv.

As can be seen from Figure ??, PConv only performs convolution operation on part of the input channels by using the regular conventional Conv for spatial feature extraction, while keeping the remaining  $c - c_p$  channels Identity mapping to Output. The FLOPs of PConv is calculated to be  $h \times w \times k^2 \times c_p^2$ , and the MAC is  $h \times w \times 2c_p + k^2 \times c_p^2 \approx h \times w \times 2c_p$ . Where  $c_p$  is the number of channels in the regular convolution action, which is usually taken to be  $1/4c$ . So the FLOPs of PConv are only  $1/16$  of those of the regular Conv, it has a smaller memory access. The paper replaces all  $3 \times 3$  convolution operations of E-ELAN in the YOLOv7 Backbone network with PConv to reduce the frequency of memory accesses by the neural network while maintaining higher FLOPs with fewer FLOPs, reducing neural network latency and making the network to extract spatial features more efficiently.

### 3.2. Improved feature fusion network

#### 3.2.1. RFB feature pyramid

The Feature Pyramid SPPCSP is composed of Single-Point Perfection (SPP)[?] ] and Complexity-Sensitive Perfection (CSP)[?] ], so it has a larger number of model parameters and computation which leads to slower inference speed of the YOLOv7 model. Additionally, during the sampling process, SPPCSP may cause a reduction in the size of input feature maps, potentially resulting in a certain degree of feature information loss, which have an impact on the detection of small objects. Consequently, we integrate RFB feature pyramid in the Neck of YOLOv7 model, which is able to learn deeper features from the lightweight CNN model, making the detection model faster and more accurate with less calculation.

The RFB is a multi-branch convolution[?] ] block inspired by Inception[?] ] in network construction. Its internal structures are divided into two parts, the multi-branch convolution with different convolution kernels and dilated convolutions with variable rates. The primary function of the dilated convolution process is to broaden the receptive field so that capture essential information in regions with more context. The structure of RFB is illustrated in Figure ??.

In the RFB, we employ branches with varying dilation factors to capture multiscale information and different ranges of dependencies. While these branches have distinct receptive fields, they share the same weights, which helps reduce the number of parameters, thereby lowering the risk of overfitting and maximizing the utilization of information from each sample. Finally, the RFB concatenates the computation results from different branches, preventing the model from encountering gradient explosion and vanishing gradient issues during training, ultimately achieving the goal of fusing different features.

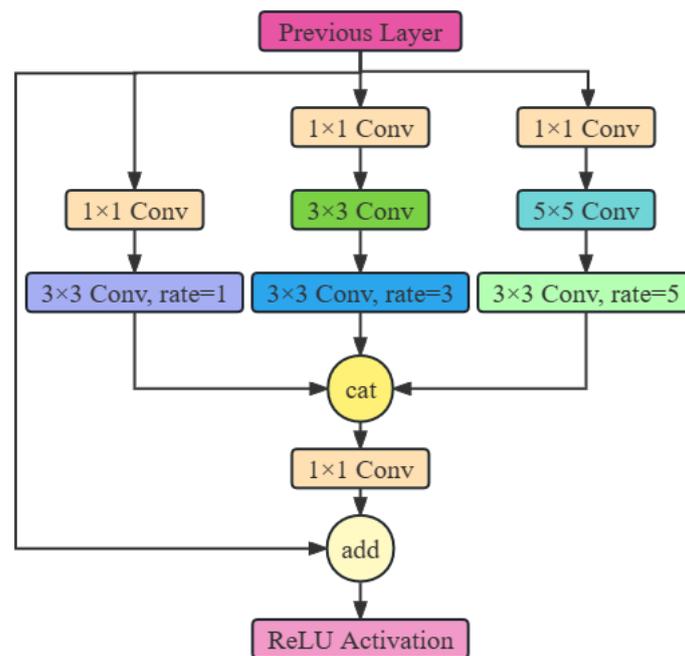


Figure 3. The structure of RFB.

### 3.2.2. CARAFE upsampling operator

The upsampling method in YOLOv7 feature fusion network is the nearest neighbor interpolation method[? ]. It works by finding the nearest neighbor pixel in the target image based on the position of each pixel in the original image, and then replacing the value of the corresponding pixel in the original image with the value of the nearest neighbor pixel. However, the image quality after scaling by this method is poor, and obvious jagged edges may appear. So, we introduce the CARAFE upsampling operator, which significantly enhances the semantic feature extraction capability of the neural network during the upsampling process in object detection tasks, without substantially increasing the number of parameters and computation. The CARAFE upsampling process is depicted in Figure ??.

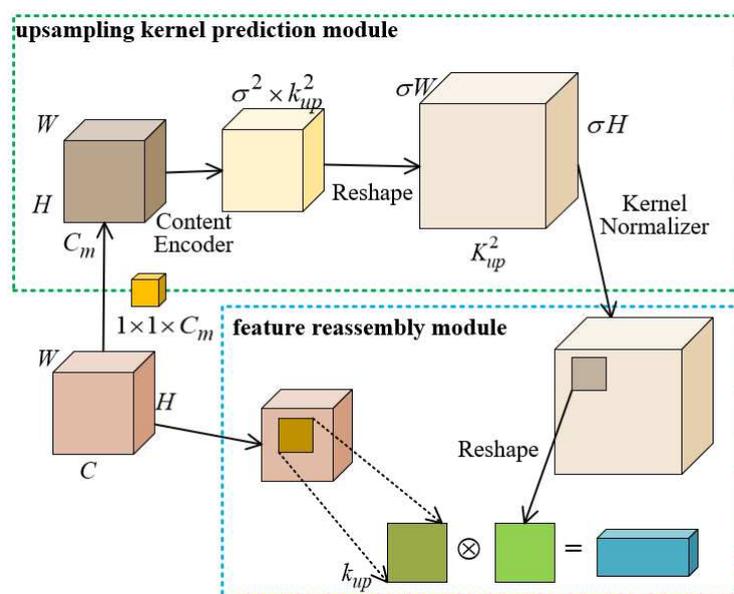


Figure 4. Upsampling process on CARAFE.

The CARAFE is divided into two main modules, which are the upsampling kernel prediction module and the feature reassembly module. In the upsampling kernel prediction module, assuming the upsampling ratio is  $\sigma$ , the process begins with channel compression through a  $1 \times 1$  convolution operation on the input feature map of size  $H \times W \times C$ , resulting in a feature map of size  $H \times W \times C_m$ . Subsequently, encoding the feature map using a  $k_{enc} \times k_{enc}$  convolutional kernel generates the recombined kernel  $k_{up} \times k_{up}$ , obtaining a feature map with  $\sigma^2 \times k_{up}^2$  channels. The channels are then spatially unfolded and subjected to softmax normalization, producing a feature map of size  $\sigma H \times \sigma W \times K_{up}^2$ , where each feature value corresponds to an upsampling kernel. In the feature recombination module, through the mapping relationship, the region  $k_{up} \times k_{up}$  centered in the input feature map is taken out, and then dot product operation with the upsampling kernel at the same position in  $\sigma H \times \sigma W \times K_{up}^2$  to obtain the output value.

The CARAFE achieves instance specific content-aware processing by generating real-time adaptive convolution kernels, and can aggregate contextual information over a larger receptive field. Therefore, we introduce CARAFE upsampling operator to replace two upsampling operations in the Neck structure of the YOLOv7 model. According to the trade-off relationship between  $k_{enc}$  and  $k_{up}$  to maintain the performance and efficiency,  $k_{up}$  takes the value of 7 and  $k_{enc}$  takes the value of 5 in the experiments.

### 3.3. HardSwish activation function

Activation functions are significant in neural networks. They are usually employed to introduce non-linearity, enabling neural networks to learn and represent more complex functions, ultimately enhancing the network's representational capacity[? ]. Without activation functions, deeper neural networks can also only handle linearly differentiable problems[? ]. Commonly used activation functions are Sigmoid, ReLU, Swish, Mish, GELU and so on. The YOLOv7 model uses the SiLU activation function with the following expression.

$$SiLU(x) = x \cdot sigmoid(x) = \frac{x}{1 + e^{-x}} \quad (4)$$

The computation process of the Sigmoid function involves exponentiation, which undoubtedly increases the computational complexity of the SiLU activation function. However, the Sigmoid activation function can be approximated using a piecewise linear function called HardSigmoid[? ], which significantly reduces the computational cost. The HardSigmoid activation function and HardSwish activation function expressions are as follows:

$$HardSigmoid(x) = \begin{cases} 0, & x \leq -3 \\ 1, & x \geq 3 \\ (x + 3)/6, & otherwise \end{cases} \quad (5)$$

$$HardSwish(x) = \begin{cases} 0, & x \leq -3 \\ (x^2 + 3x)/6, & -3 < x < 3 \\ x, & x \geq 3 \end{cases} \quad (6)$$

The HardSwish and SiLU activation function output curves are shown in Figure ?? . HardSwish and SiLU are differentiable at all points and have no upper bounds, enabling them to avoid overfitting while making the model more generalizable. Furthermore, the segmented function reduces the number of memory accesses, which significantly reduces the latency cost.

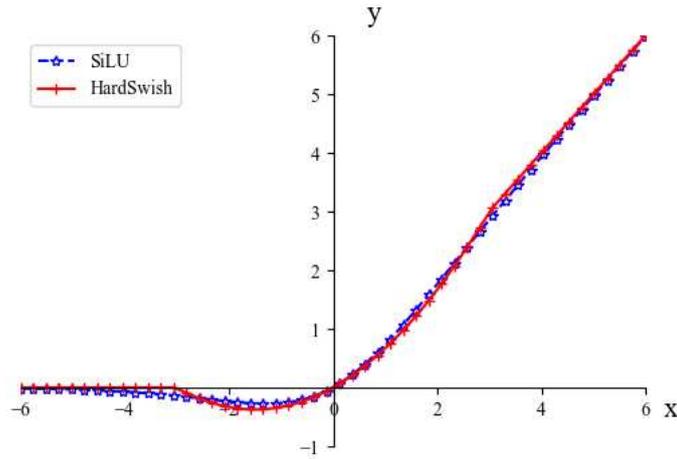


Figure 5. Activation function output curves.

### 3.4. Wise-IOU v3 Bounding Box Loss

The loss function of YOLOv7 consists of three parts: Object Confidence Loss (ObjLoss), Classification Loss (ClsLoss) and Bounding Box Loss (BoxLoss). Among them, the confidence loss and classification loss are calculated using binary cross entropy loss function, and the BoxLoss uses the Complete Intersection Over Union (CIOU) loss[? ]. YOLOv7 total loss expression is shown in equation ??.

$$Loss = Loss_{Obj} + Loss_{cls} + Loss_{Box} \quad (7)$$

The CIOU loss takes into account three geometric factors: overlap area, center point distance and aspect ratio. Given a prediction frame and a target frame, the CIOU loss is calculated as:

$$L_{CIOU} = 1 - IOU + \frac{\sqrt{(i_x - j_x)^2 + (i_y - j_y)^2}}{c^2} + \alpha v \quad (8)$$

$$\alpha = \frac{v}{(1 - IOU) + v} \quad (9)$$

$$v = \frac{4}{\pi^2} \left( \arctan \frac{w^{st}}{h^{st}} - \arctan \frac{w}{h} \right)^2 \quad (10)$$

The gradient of  $v$  with respect to  $w$  and  $h$  is calculated as follows:

$$\begin{cases} \frac{\partial v}{\partial w} = \frac{8}{\pi^2} \left( \arctan \frac{w^{st}}{h^{st}} - \arctan \frac{w}{h} \right) * \frac{h}{w^2 + h^2} \\ \frac{\partial v}{\partial h} = \frac{8}{\pi^2} \left( \arctan \frac{w^{st}}{h^{st}} - \arctan \frac{w}{h} \right) * \frac{w}{w^2 + h^2} \end{cases} \quad (11)$$

$(i_x, i_y), (j_x, j_y)$  are the center point coordinates of the real frame and the prediction frame respectively,  $\alpha$  is the balance parameter,  $v$  is used to measure the similarity of the length ratio, and  $c$  is denoted by the diagonal length of the smallest outer rectangle containing the true and predicted frames. Analysis of equation ?? shows that when the aspect ratio of the predicted frame and the real frame is equal, the value is 0, the penalty term of the aspect ratio loses its effect, which reduces the optimization performance of the model for similarity. From equation ??, the relationship between  $\frac{\partial v}{\partial w}, \frac{\partial v}{\partial h}$ , is inversely proportional, during training, the width and height of the prediction frame cannot be increased or decreased at the same time, which is not conducive for the prediction frame to approximate to the real frame[? ].

The CIOU employs a monotonic focus mechanism, which is predicated on the assumption

that high-quality examples exist within the training data. Its primary objective is to enhance the fitting capability of bounding box loss. However, when the target detection training dataset includes low-quality examples, blindly emphasizing the regression of bounding boxes for these low-quality instances evidently lead to a decline in the model's detection performance[? ]. Therefore, we use Wise-IOU v3 based on a non-monotonic dynamic focusing mechanism as the bounding box loss. The dynamic non-monotonic focusing mechanism uses the outlier degree instead of IOU to assess the quality of anchor boxes, which reduces the competitiveness of high-quality anchor boxes and at the same time reduces the harmful gradient generated by low-quality examples and improves the overall performance of the detector. The formula is as follows:

$$R_{WIOU} = \exp\left(\frac{(i_x - i_y)^2 + (j_x - j_y)^2}{(W^2 + H^2)^*}\right) \quad (12)$$

$$L_{w1} = R_{WIOU} * L_{IOU} \quad (13)$$

$$r = \frac{\beta}{3 \times 1.9^{\beta-3}} \quad (14)$$

$$\beta = \frac{L_{IOU}^*}{L_{IOU}} \in [0, +\infty) \quad (15)$$

$$L_{WIOU} = r * L_{w1} \quad (16)$$

$W$  and  $H$  are the width and height of the minimum outer rectangle of the prediction box and the real box respectively,  $(W^2 + H^2)^*$  denotes the separation of  $W$ ,  $H$  from the Computational Graph to prevent  $R_{WIOU}$  from making gradients that impede convergence.  $L_{IOU}^*$  is the monotonic focusing factor of  $L_{w1}$ .  $\overline{L_{IOU}}$  denotes the sliding mean with momentum  $m$ , and its dynamic updates contribute to maintaining an elevated overall  $\beta$ , effectively resolving the issue of slow convergence in the later stages of training. This empowers Wise-IOU to flexibly adjust gradient gain allocation strategies during the training process to best adapt to the current situation.

### 3.5. PRC-Light YOLO Model Structure

In summary, the paper proposes the PRC-Light YOLO. A new partial convolution is utilized to replace the  $3 \times 3$  convolution operations of E-ELAN in Backbone to form PE-ELAN module. Meanwhile, the RFB feature pyramid is inserted in the Neck module, and the nearest neighbor interpolation is replaced with CARAFE to improve the semantic extraction ability of the network. At the last, the SiLU of CBS in the whole model is replaced with HardSwish activation function, Convolution, Batch Normalization and HardSwish activation function constitute the CBH module. The PRC-Light YOLO structure is shown in Figure ??.

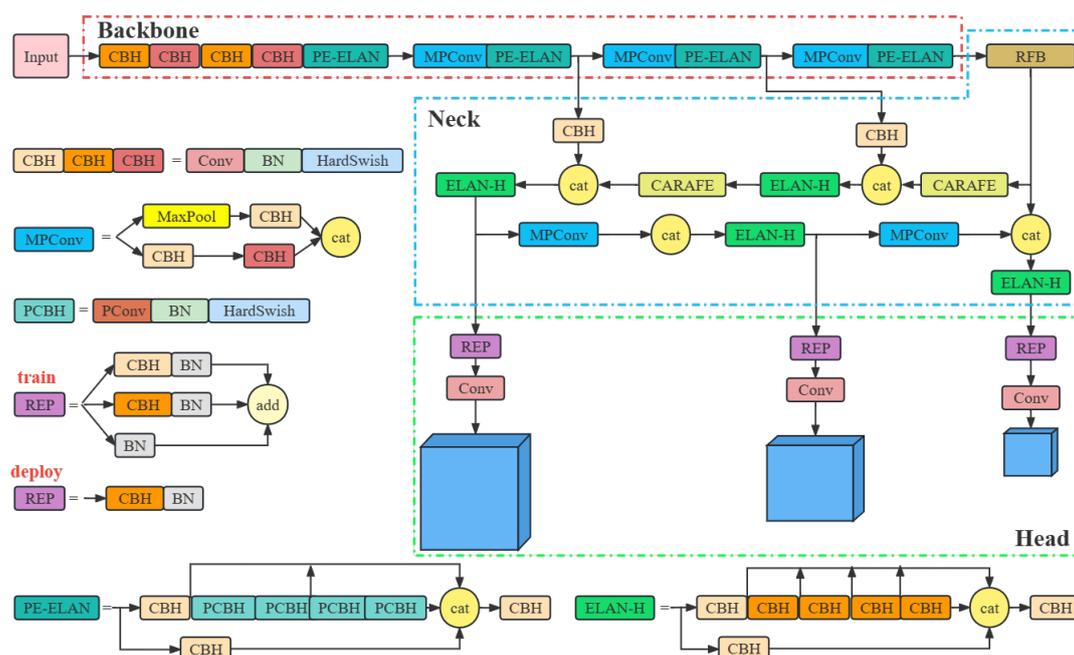


Figure 6. The PRC-Light YOLO structure.

## 4. Experiment

### 4.1. Fabric defect image dataset

The experiment selected four categories of Warp hanged, Yard defects, Holes and Stains from the measured dataset[?] and named them as dj, ds, pd, and wz, respectively. We select 1061 images from the above dataset and use MakeSense to label the defects types in the images, then extracting the txt annotation information files required for YOLOv7 model training. To solve the problem of insufficient number of these four datasets, we conduct data augmentation on the 1061 datasets, expanding the dataset to 4244 so that enhances the generalization ability of the network model, then dividing the training set, validation set, and testing set according to the ratio of 8:1:1. Parts of the samples are shown in Figure ??.



Figure 7. Defective fabric image samples.

#### 4.2. Experimental environment and parameter configuration

The experimental equipment parameters are shown in Table ???. The input image size is 320×320, with a total of 150 epochs of training with batch size 16, the initial learning rate is set to 0.01.

Table 1. Experimental equipment parameters.

Name	Operating System	RAM	Graphics Card	CUDA	Python	Framework
Parameter	Windows X64	12G	NVIDIA Quadro P4000	11.3	3.8	PyTorch

#### 4.3. Evaluation metrics

To verify the performance of the improved model, we use Precision (P), Recall (R), F1-score, and mean Average Precision (mAP) as the measures of detection accuracy, the formulas are calculated as follows:

$$P = \frac{TP}{TP + FP} \quad (17)$$

$$R = \frac{TP}{TP + FN} \quad (18)$$

$$F_1 - score = \frac{2 * P * R}{P + R} = \frac{2TP}{2TP + FP + FN} \quad (19)$$

$$mAP = \frac{\sum_{i=1}^N AP_i}{N} \quad (20)$$

TP shows the number of positive samples predicted as positive samples, FP denotes the number of negative samples predicted as positive samples, and FN indicates the number of positive samples predicted as negative samples. Each category can draw PR curves based on Precision and Recall, every

category corresponds to one PR curve, the area covered by which indicates the Average Precision (AP) of the category. mAP is the average value after summing up the APs of all categories.

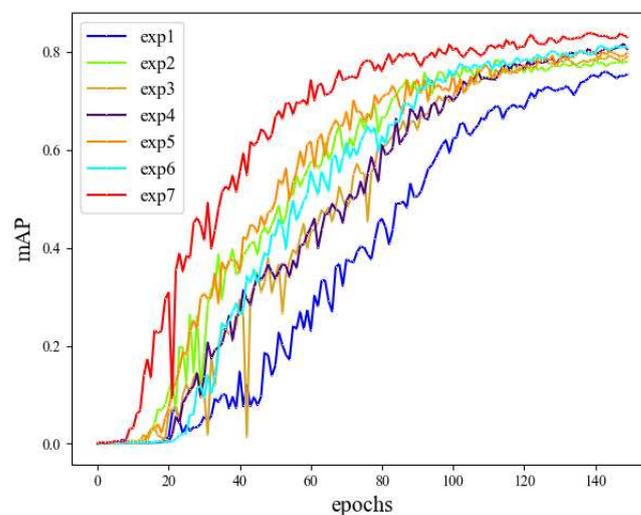
#### 4.4. Ablation experiment

Ablation experiment is an important method in deep learning, by improving the modules of the YOLOv7 network structure and verifying the contribution of different improvement methods to the performance of this model, we carried out seven sets of ablation experiments in all. The experimental results are shown in Table ??.

**Table 2.** Results of ablation experiments.

Experiment	HardSwish	Wise-IOU v3	RFB	CARAFE	PConv	Params/M	GFLOPs/G	mAP/%
exp1	×	×	×	×	×	37.21	105.2	75.4
exp2	✓	×	×	×	×	37.21	105.2	78.1
exp3	×	✓	×	×	×	37.21	105.2	79.1
exp4	×	×	✓	×	×	33.94	102.5	80.6
exp5	×	×	×	✓	×	37.87	106.5	79.8
exp6	✓	✓	✓	✓	×	34.6	103.8	81.1
exp7	✓	✓	✓	✓	✓	30.5	83.6	83

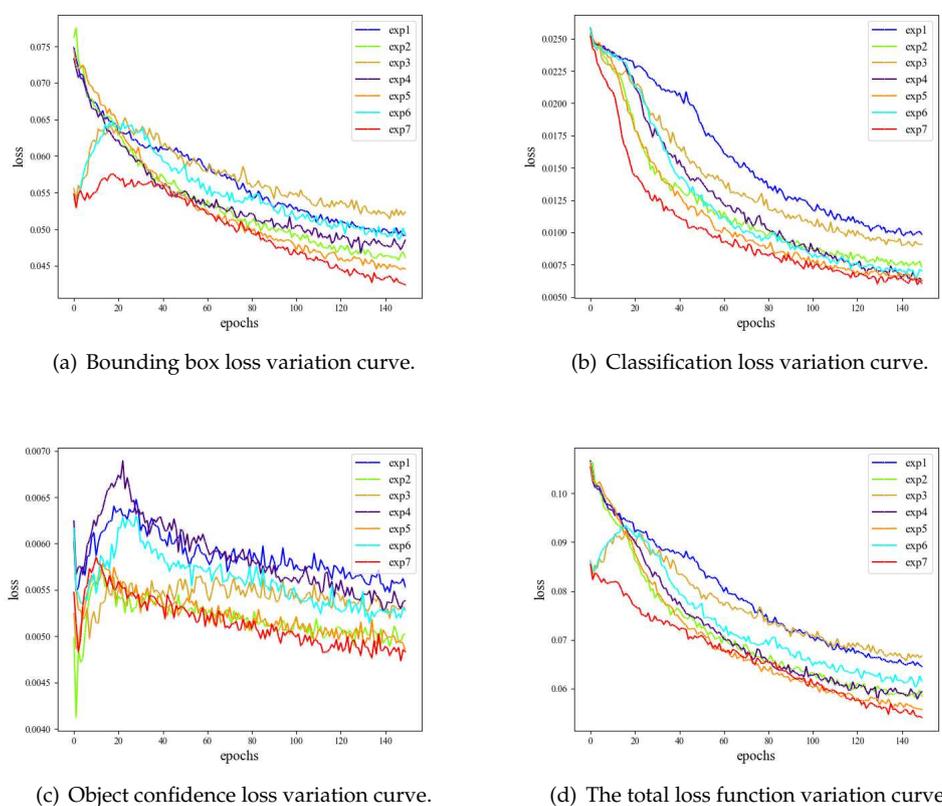
As can be seen from Table ??, the first set of experiments is based on the YOLOv7, the model parameter number is 37.21M, the GFLOPs is 105.2G, and the mAP is 75.4%, followed by subsequent experiments which are all based on the results of the first set of experiments as a benchmark for comparison. Experiments 2 and 3, without increasing the number of model parameters, respectively increased the mAP values by 2.7% and 3.7%. In Experiment 4, the introduction of the RFB feature pyramid reduces the parameters by 8.79% compared to the SPPCSPC module in YOLOv7. while simultaneously improving the mAP by 5.2 percentage points. Experiment 5 utilizes CARAFE lightweight upsampling module in the feature fusion module, and the mAP value goes to 79.8% with little increase in the number of parameters and GFLOPs, which is a significant improvement. In order to verify the lightweight effect of the PConv, Experiment 6 integrates HardSwish, Wise-IOU v3, RFB and CARAFE into the YOLOv7. By comparing Experiment 7 with Experiment 6, we obtained that the number of parameters and computation of the network were reduced respectively by 11.85% and 19.46%, and the mAP value was increased by 1.9%.



**Figure 8.** Change curve of mAP value.

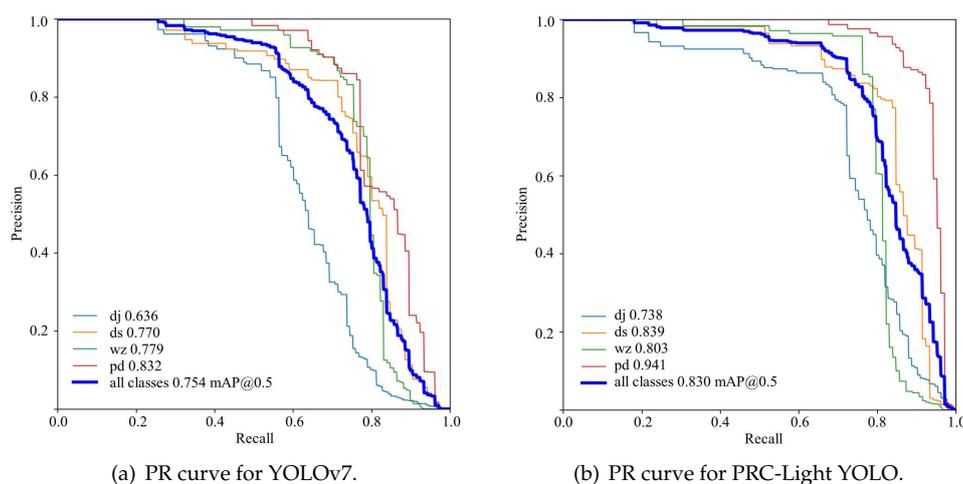
The variation curves of mAP values in 7 groups of experiments are shown in Figure ???. Compared with YOLOv7, the number of parameters and GFLOPs of PRC-Light YOLO are reduced by 18.03% and 20.53% respectively, while the mAP value is improved by 7.6%. Experiments show that by improving the network structure of YOLOv7, the lightweight effect is significant, moreover, the defect detection effect of the improved model on fabrics has been obviously improved.

According to the loss curve chart above, it is evident that the PRC-Light YOLO model exhibits smaller loss values after convergence compared to other models. Specifically, experiments 3, 6, and 7 employ the Wise-IoU bounding box loss function, which dynamically adjusts the emphasis on different input regions over time. For anchor boxes with lower outliers, a smaller gradient boost is allocated, thereby directing the boundary regression towards normal-quality anchor boxes. This adjustment enhances the convergence speed of the network model. The Figure ?? demonstrates that the PRC-Light YOLO model achieves lower loss values, thereby delivering superior performance in fabric defect detection.



**Figure 9.** Loss Function. The loss function of the model consists of three parts, (a), (b) and (c) respectively show the change curves of the BoxLoss, ClsLoss and ObjLoss for YOLOv7 and PRC-Light YOLO. (d) shows the total loss function variation curve.

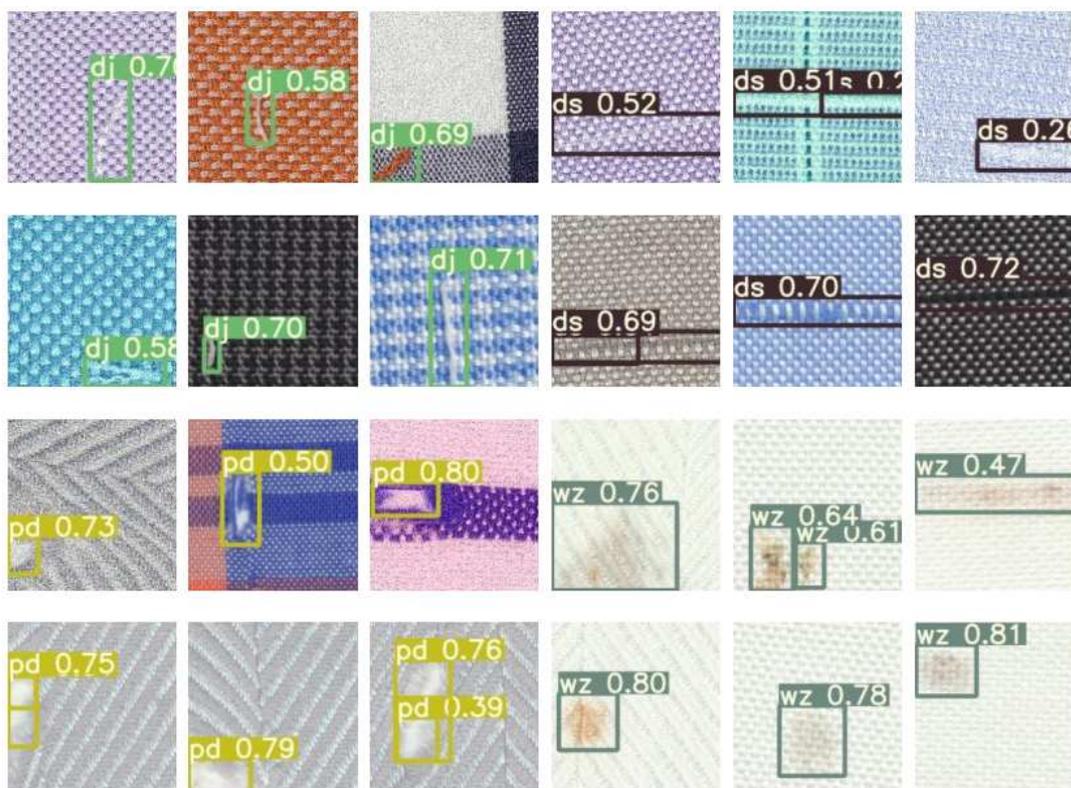
According to Figure ??, it can be obtained that the mAP value of PRC-Light YOLO is increased from 75.4% at the beginning to 83%, and the experimental results verified that the PRC-Light YOLO has improved the effect of defect detection. The AP values of threading, yard defects, stains and holes increased by 10.2%, 6.9%, 2.4%, 10.9%, respectively, with the most noticeable improvement in the detection of hanging warp and holes, the AP values of the four categories of defects all are improved.



**Figure 10.** Precision-Recall.(a) and (b) respectively show the PR curves of YOLOv7 and PRC-Light YOLO on the fabric dataset.

#### 4.5. Comparison of detection effect

Figure ?? and Figure ?? show the YOLOv7 and PRC - Light YOLO effects of four different types of defect detection, including the location of the defect by box selection, as well as above the box to display the type of defect and confidence. From the comparison in the figure, it can be observed that the PRC-Light YOLO detection model has improved the accuracy of fabric defect detection.



**Figure 11.** Detection effect of the YOLOv7.

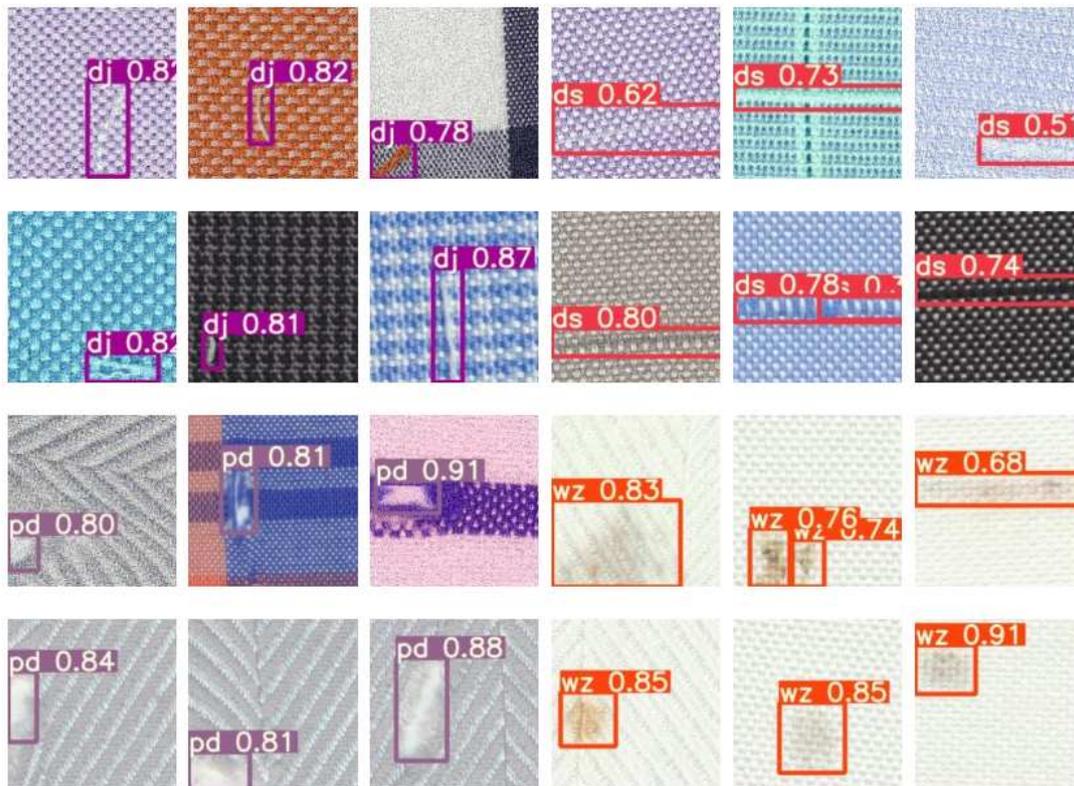


Figure 12. Detection effect of the PRC-Light YOLO.

#### 4.6. Comparison experiment

Through comparison experiments, the performance of different models can be compared so that the better method can be selected. Therefore, we choose to conduct comparison experiments with Faster R-CNN, SSD, EfficientDet, and CenterNet target detection models.

Table ?? shows a performance comparison of defect detection among various object detection models, with P, R, F1-score, and mAP selected as evaluation metrics.

Table ?? shows the comparison of the experimental results of different target detection models for the fabric defect dataset.

Table 3. Defect detectability of different models.

Models	P	R	F1	Params/M	GFLOPs/G	mAP/%
Faster R-CNN	0.784	0.491	0.6	136.98	370.3	0.696
SSD	0.832	0.529	0.847	105.2	87.41	0.731
EfficientDet	0.787	0.492	0.605	52.11	34.97	0.697
CenterNet	0.838	0.547	0.662	125	69.66	0.756
YOLOv7	0.826	0.697	0.76	37.21	105.2	0.754
PRC-Light YOLO	<b>0.859</b>	<b>0.784</b>	<b>0.82</b>	<b>30.50</b>	<b>83.6</b>	<b>0.83</b>

Table 4. Detection effects of different models.

Categories	Faster R-CNN	SSD	EfficientDet	CenterNet	YOLOv7	PRC-Light YOLO
Warp hanged	0.738	0.756	0.731	<b>0.802</b>	0.636	0.738
Yard defects	0.582	0.596	0.599	0.652	0.77	<b>0.839</b>
Stain	0.692	0.759	0.686	0.765	0.779	<b>0.803</b>
Hotel	0.773	0.814	0.773	0.804	0.832	<b>0.941</b>

As it can be seen that the two-stage detection model Faster R-CNN has the lowest mAP value compared to the other detection algorithms, besides, the two-stage detection model has one more step of operation compared to the single-stage model: the candidate regions are first generated, and then the regions are classified and identified, which also causes the slower training speed of Faster R-CNN. Compared with the target detection models of Faster R-CNN, SSD, EfficientDet, EfficientDet and YOLOv7, the precision of PRC-Light YOLO is improved by 7.5%, 2.7%, 7.2%, 2.1% and 3.3%. The recall is increased by 29.3%, 25.5%, 29.2%, 23.7%, 8.7%; The F1-score is increased by 20%, 17.3%, 21.5%, 15.8%, 6%, and params is decreased by 77.7%, 71%, 41.5%, 75.6%, 18%. In terms of GFLOPs, PRC-Light YOLO is lower than Faster R-CNN, SSD, and YOLOv7 but higher than EfficientDet and CenterNet, whereas in terms of mAP PRC-Light YOLO outperforms EfficientDet and CenterNet by 13.3% and 7.4%. Through the AP values of four defect types in the fabric data set, we can obtain that CenterNet is effective in detecting the warp hanged, but the PRC-Light YOLO is more effective in detecting Yarn defects, Stains, and Hole types. Comprehensive indications reveal that our proposed PRC-Light YOLO has an excellent detection effect on the fabric dataset, which confirms the effectiveness of the improved method.

## 5. Fabric Defect Detection System

In order to further illustrate that the model has a certain value of engineering applications, and at the same time more intuitively show the detection process and detection results. We use PyQt5 to realize the design of fabric defect detection system based on PRC-Light YOLO. The UI interface of the system is shown in Figure ??.

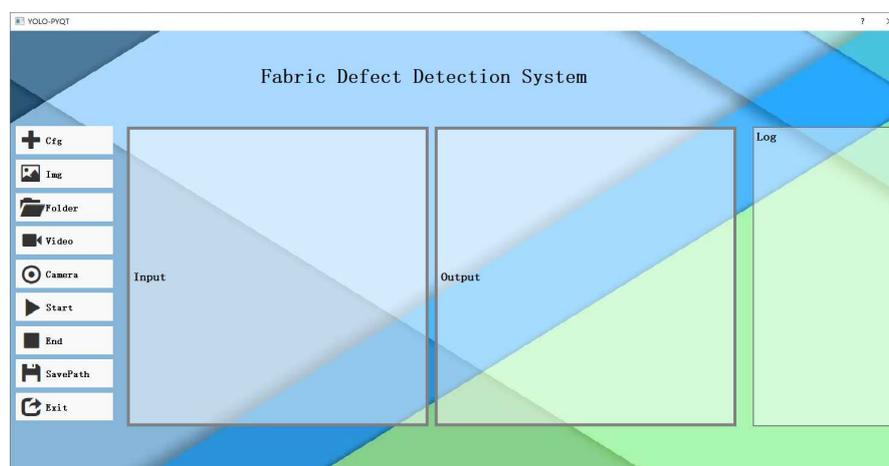


Figure 13. Detection System UI.

Clicking the "Cfg" button to choose the configuration file required by the detection system. The file format should be in YAML format. The path to the selected file will be displayed in the Statistical window. If the loading is successful, it will show "Load yaml success". The application effect of YOLO model is shown in Figure ??.

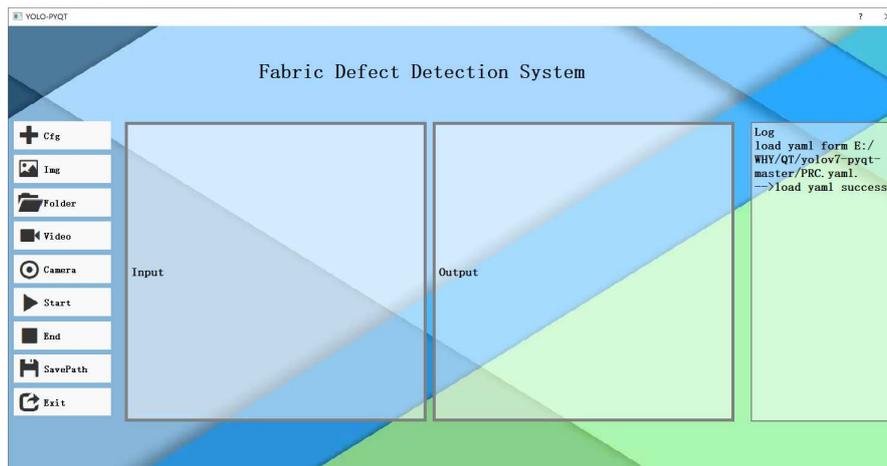


Figure 14. Application effect of PRC-Light YOLO model.

The content written in the yaml file includes the path to the weight file, the category of the blemish, and the confidence threshold and IOU threshold. The confidence threshold is used to filter the target detection results from the model's output. Only when the probability of detecting a target is greater than the confidence threshold will the target box be displayed. The IOU threshold primarily controls how the target detection algorithm handles overlapping targets. When two predicted bounding boxes intersect, if the IOU value between the two boxes is greater than the IOU threshold, lower-confidence prediction boxes will be filtered out.

After selecting the configuration file, you can set the path where the detection results will be saved. Additionally, the system is capable of processing images, videos, and real-time data captured through a camera feed. The data path will be displayed in the Statistical window accordingly. By clicking the "Start Detection" button, the detection results will be displayed to the right of the input image. The YOLO model detection effect is shown in Figure ??.



Figure 15. PRC-Light YOLO model checking effect.

In the Statistical window, you will also find information such as the time taken for detection, the path where the results are saved, as well as the categories and quantities of defects detected. This information is essential for subsequent and timely manual inspection.

## 6. Discussion

The detection of defects in fabrics holds significant importance for ensuring fabric quality, costing reduction, enhancing production efficiency, and preserving brand reputation. It is pivotal for the

competitiveness and sustainable development of the fabric industry. This article presents a fabric defect detection model based on PRC-Light YOLO, which reduces the parameters of the YOLOv7 while improving the accuracy of fabric defect detection. However, certain limitations still exist in some aspects.

(1) In the study of fabric defect detection, the size and variety of the dataset play a crucial role in the effectiveness of the model training. In practical industrial scenarios, fabric backgrounds are often characterized by complexity and diversity. However, the dataset utilized in this paper suffers from limitations such as a uniform background, limited quantity, and insufficient diversity in defect types. Consequently, the exploration of methodologies for obtaining and constructing a substantial dataset of fabric defects becomes imperative. Furthermore, investigating fabric defects detection the background of intricate patterns and textures holds significant research significance.

(2) In the research findings of this paper, it is evident that PRC-Light YOLO requires improvement in its capability to detect Warp hanged. Therefore, for the detection of minute defects, further investigation can be conducted into image preprocessing, network feature extraction capabilities, or loss functions, with the aim of enhancing the precision of the network's detection capabilities.

(3) The fabric defect detection system developed in this study places a primary focus on interactivity and intuitiveness, representing an initial construction of the detection system. While this system introduces new perspectives for real-time fabric defect detection, it has not yet been integrated with hardware infrastructure. To achieve full-scale industrial applications, future efforts can involve further refining the detection platform and completing the setup of the system's hardware components.

## 7. Conclusion

In response to the complex nature of defect types in fabrics, which vary in size, the computational intensity of detection models, and the lower detection accuracy, this paper proposes the defect detection model PRC-Light YOLO. The PRC-Light YOLO achieves network lightweight by replacing certain Conv layers with PConv layers in the Backbone. During the feature fusion stage, the RFB pyramid and the CARAFE upsampling operator are employed, this approach enlarges the neural network's receptive field and enhances detection accuracy without significantly increasing the model's parameter count or computational load. PRC-Light YOLO adopts the HardSwish activation function, which exhibits piecewise characteristics that reduce computational costs and minimize memory access. Additionally, it utilizes Wise-IOU v3 as the bounding box loss function during model training, demonstrating robust performance in scenarios with varying dataset scales and highly imbalanced positive-to-negative sample ratios. Finally, we design a fabric defect detection system that improves interaction.

To sum up, PRC-Light YOLO effectively reduces the network's parameter count and computational load while simultaneously improving its performance in detecting defects in fabrics.

**Author Contributions:** Conceptualization, B.L.and H.W.; methodology,B.L.and H.W.; software, Z.C.; validation, H.W., Z.C. and Y.W.; investigation, H.W., L.T.and J.Y.; writing—original draft preparation, H.W.; writing—review and editing, H.W.and B.L.; supervision, B.L.and K.Z.; project administration and funding acquisition, B.L. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China, grant number 61971339.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data used to support the findings of this study are available from the corresponding author upon request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- . Zhang, H.; Qiao, G.; Lu, S.; Yao, L.; Chen, X. Attention-based Feature Fusion Generative Adversarial Network for yarn-dyed fabric defect detection. *Textile Research Journal* **2023**, *93*, 1178–1195.
- . Li, W.; Zhang, Z.; Wang, M.; Chen, H. Fabric Defect Detection Algorithm Based on Image Saliency Region and Similarity Location. *Electronics* **2023**, *12*, 1392.
- . Xiang, J.; Pan, R.; Gao, W. Yarn-dyed fabric defect detection based on an improved autoencoder with Fourier convolution. *Textile Research Journal* **2023**, *93*, 1153–1165.
- . Wu, J.; Li, P.; Zhang, H.; Su, Z. CARL-YOLOF: A well-efficient model for digital printing fabric defect detection. *Journal of Engineered Fibers and Fabrics* **2022**, *17*, 15589250221135087.
- . Kanwal, M.; Riaz, M.M.; Ali, S.S.; Ghafoor, A. Saliency-based fabric defect detection via bag-of-words model. *Signal, Image and Video Processing* **2023**, *17*, 1687–1693.
- . Kahraman, Y.; Durmuşoğlu, A. Deep learning-based fabric defect detection: A review. *Textile Research Journal* **2023**, *93*, 1485–1503.
- . Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14. Springer, 2016, pp. 21–37.
- . Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
- . Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767* **2018**.
- . Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.
- . Girshick, R. Fast r-cnn. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
- . Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* **2015**, *28*.
- . Zhang, J.; Jing, J.; Lu, P.; Song, S. Improved MobileNetV2-SSDLite for automatic fabric defect detection system based on cloud-edge computing. *Measurement* **2022**, *201*, 111665.
- . Zhou, S.; Zhao, J.; Shi, Y.S.; Wang, Y.F.; Mei, S.Q. Research on improving YOLOv5s algorithm for fabric defect detection. *International Journal of Clothing Science and Technology* **2023**, *35*, 88–106.
- . Wu, Y.; Lou, L.; Wang, J. Cotton fabric defect detection based on K-SVD dictionary learning. *Journal of Natural Fibers* **2022**, *19*, 10764–10779.
- . Lin, G.; Liu, K.; Xia, X.; Yan, R. An efficient and intelligent detection method for fabric defects based on improved YOLOv5. *Sensors* **2022**, *23*, 97.
- . Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 7464–7475.
- . Liu, S.; Huang, D.; et al. Receptive field block net for accurate and fast object detection. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), 2018, pp. 385–400.
- . Baranwal, N.; Singh, K.N.; Singh, A.K.; et al. YOLO-based ROI selection for joint encryption and compression of medical images with reconstruction through super-resolution network. *Future Generation Computer Systems* **2024**, *150*, 1–9.
- . Wang, C.Y.; Liao, H.Y.M.; Yeh, I.H. Designing network design strategies through gradient path analysis. *arXiv preprint arXiv:2211.04800* **2022**.
- . Zhu, L.; Lee, F.; Cai, J.; Yu, H.; Chen, Q. An improved feature pyramid network for object detection. *Neurocomputing* **2022**, *483*, 127–139.
- . Chen, J.; Kao, S.h.; He, H.; Zhuo, W.; Wen, S.; Lee, C.H.; Chan, S.H.G. Run, Don't Walk: Chasing Higher FLOPS for Faster Neural Networks. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12021–12031.

- . Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, 2019, pp. 1314–1324.
- . Ma, N.; Zhang, X.; Zheng, H.T.; Sun, J. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), 2018, pp. 116–131.
- . Tang, Y.; Han, K.; Guo, J.; Xu, C.; Xu, C.; Wang, Y. GhostNetv2: enhance cheap operation with long-range attention. *Advances in Neural Information Processing Systems* **2022**, *35*, 9969–9982.
- . He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence* **2015**, *37*, 1904–1916.
- . Wang, C.Y.; Liao, H.Y.M.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A new backbone that can enhance learning capability of CNN. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, 2020, pp. 390–391.
- . Ding, P.; Qian, H.; Bao, J.; Zhou, Y.; Yan, S. L-YOLOv4: lightweight YOLOv4 based on modified RFB-s and depthwise separable convolution for multi-target detection in complex scenes. *Journal of Real-Time Image Processing* **2023**, *20*, 71.
- . Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A. Inception-v4, inception-resnet and the impact of residual connections on learning. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2017, Vol. 31.
- . Jiang, N.; Wang, L. Quantum image scaling using nearest neighbor interpolation. *Quantum Information Processing* **2015**, *14*, 1559–1571.
- . Manzhos, S.; Ihara, M. Neural network with optimal neuron activation functions based on additive Gaussian process regression. *arXiv preprint arXiv:2301.05567* **2023**.
- . Polyzos, E.; Nikolaou, C.; Polyzos, D.; Van Hemelrijck, D.; Pyl, L. Direct modeling of the elastic properties of single 3D printed composite filaments using X-ray computed tomography images segmented by neural networks. *Additive Manufacturing* **2023**, *76*, 103786.
- . Siddique, A.; Vai, M.I.; Pun, S.H. A low cost neuromorphic learning engine based on a high performance supervised SNN learning algorithm. *Scientific Reports* **2023**, *13*, 6280.
- . Zheng, Z.; Wang, P.; Liu, W.; Li, J.; Ye, R.; Ren, D. Distance-IoU loss: Faster and better learning for bounding box regression. In Proceedings of the Proceedings of the AAAI conference on artificial intelligence, 2020, Vol. 34, pp. 12993–13000.
- . Sravya, N.; Lal, S.; Nalini, J.; Reddy, C.S.; Dell'Acqua, F.; et al. DPPNet: An efficient and robust deep learning network for land cover segmentation from high-resolution satellite images. *IEEE Transactions on Emerging Topics in Computational Intelligence* **2022**, *7*, 128–139.
- . Tong, Z.; Chen, Y.; Xu, Z.; Yu, R. Wise-IoU: Bounding Box Regression Loss with Dynamic Focusing Mechanism. *arXiv preprint arXiv:2301.10051* **2023**.
- . Chen, M.; Yu, L.; Zhi, C.; Sun, R.; Zhu, S.; Gao, Z.; Ke, Z.; Zhu, M.; Zhang, Y. Improved faster R-CNN for fabric defect detection based on Gabor filter with Genetic Algorithm optimization. *Computers in Industry* **2022**, *134*, 103551.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.