Academic Journal of Science and Technology
ISSN: 2771-3032 | Vol. 7, No. 1, 2023

Tile Surface Defect Detection Based on Improved Faster R-CNN

Xuan Che, Wenzhong Zhu

College of computer science and engineering, Sichuan University of Science and Engineering, Sichuan 64400, China

Abstract: With the increasing population, there is a high demand for housing, commercial buildings, and lifestyle facilities. The development of modern manufacturing technology has led to higher quality control requirements for tiles. This article presents a ceramic tile defect detection method based on Faster R-CNN, primarily focused on detecting surface defects or cracks on tiles. The method replaces VGG16 with the ResNet-101 network as a new backbone and utilizes depth-wise separable convolution to address efficiency issues resulting from the backbone replacement. Finally, Soft-NMS is employed to optimize regression boxes and prevent missed detections. Experimental results demonstrate that the improved algorithm achieves a mAP of 70.4%, a 13.2% enhancement over the original algorithm, highlighting the effectiveness and feasibility of the proposed approach.

Keywords: Defect detection, Tile defects, Faster-RCNN, BiFPN.

1. Introduction

China, as the world's leading producer of ceramic tiles[1], not only fulfills the substantial domestic construction demand but also exports its products abroad with a strong reputation and capability. Due to the unique properties of construction materials, ceramic tiles are susceptible to breakage, which can occur during production, packaging, and transportation due to mishandling. Surface defects such as cracks, scratches, and indentations on ceramic tiles can mar their aesthetic appearance and diminish their decorative impact. Furthermore, these flaws can lead to dirt accumulation and water infiltration, further compromising the quality of the tiles. Cracks on the surface or within the interior of ceramic tiles might compromise their structural integrity, reducing their strength and durability. These cracks can also serve as pathways for dirt and moisture, causing further deterioration. Defects in ceramic tiles can have adverse effects on their appearance, performance, and lifespan.Traditional manual inspection methods [2] rely on the subjective expertise of inspectors to identify flaws and defects. This approach is inefficient, resource-intensive, heavily reliant on human labor, and prone to errors in detection and classification. Therefore, replacing manual inspection with deep learning object detection algorithms holds significant importance.

At present, there have been numerous experiments on machine vision methods for object surface defect detection. For instance, Pu Yuxiang[3] employed image edge detection to determine the presence of defects on object surfaces based on contours. Through the utilization of the Canny algorithm for image binarization and overlaying the resulting binary image with the original, the defect positions could be intuitively displayed . With the advancement of deep learning in the field of computer vision, visual tasks similar to object detection have achieved breakthroughs using neural networks on various authoritative datasets. Yang Cui[4]proposed a machine vision-based method for detecting subtle defects in images, with the main idea of constructing a lightweight network model based on the Faster-RCNN framework. This method utilizes sample gradient feature information for non-end-to-end network training, effectively enhancing the model's inference capability. Maheshwari S. Biradar[5] utilized a supervised three-layer deep convolutional neural network (DCNN), with each convolutional layer containing a support vector machine (SVM) as a classifier. This network detected local connectivity between each pixel, aiding in learning object structures and discerning defects .

Based on deep learning, object detection algorithms are currently divided into two main categories: two-stage algorithms represented by RCNN[6] and Faster R-CNN[7], and one-stage algorithms represented by YOLO[8]、RetinaNet[9]. Two-stage algorithms extract region proposals through feature extraction and then employ convolutional neural networks for regression and classification. They exhibit high accuracy but are slower. On the other hand, the speed-focused improvements in one-stage detection algorithms, while addressing speed concerns, tend to sacrifice a certain level of accuracy. The Faster R-CNN network used in this study, which was once considered a two-stage network, boasts a favorable balance between accuracy and speed.

Given the characteristics and detection requirements of ceramic tile defects, this experiment primarily made the following improvements to the Faster R-CNN. In order to enhance the network's detection capability, the original VGG16 backbone was replaced with ResNet101[10]. The BiFPN[11] (bidirectional feature pyramid network) was introduced to enhance feature fusion, offering a more accurate response to the diverse sizes and shapes of ceramic tile defects. Simultaneously, to mitigate the parameter increase resulting from the aforementioned changes, depth-wise separable convolutions[12] were employed to reduce detection time. Finally, to prevent the removal of regression boxes with large overlapping areas, the Soft-NMS[13] technique was utilized to suppress the scores of regression boxes, thus increasing the detection rate.

2. Faster R-CNN Algorithm and Improvements

2.1. Faster R-CNN and the network improved in this article

CNN is an object detection network in the field of deep learning. It was proposed in 2015 by Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. It represents a significant advancement in detection algorithms, addressing a series of issues present in previous approaches. This innovation has led to improvements in detection speed and major breakthroughs in detection accuracy. Additionally, it has streamlined the previously complex multi-stage processes. One key addition introduced by Faster R-CNN is the Region Proposal Network (RPN), which enhances the efficiency of object detection. The diagram below depicts the network architecture.

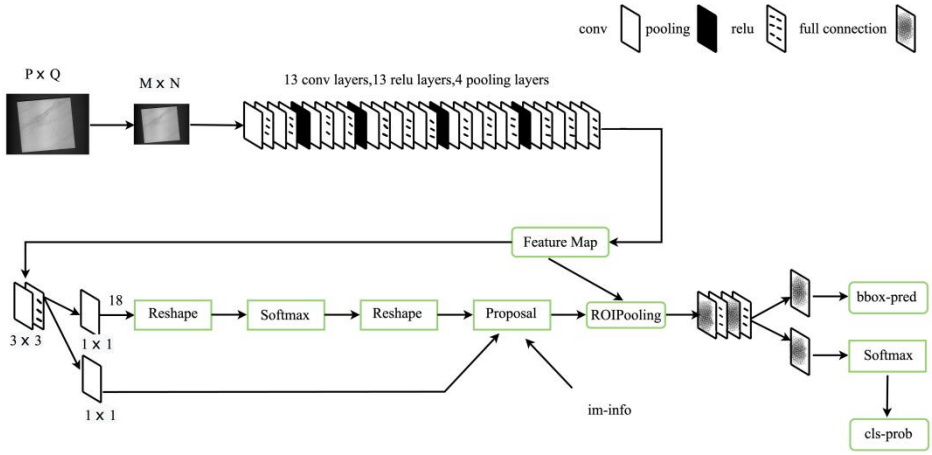


Figure 1. Faster-RCNN Structure

The main architecture of Faster R-CNN is shown in the diagram. Initially, the image size is resized to fit the network. After passing through the convolutional feature extractor, Faster R-CNN employs a pretrained convolutional neural network (such as VGG16, ResNet, etc.) to extract features from the input image. These features are then used for subsequent object detection tasks. The Region Proposal Network (RPN) stands as the core innovation of Faster R-CNN. It's a neural network responsible for generating candidate bounding boxes. By sliding a window across the feature map, the RPN predicts multiple candidate bounding boxes and their corresponding confidences for each window position. These candidate boxes serve as regions likely to contain objects.

In previous R-CNN models, each candidate box was handled separately, leading to redundant feature computation. To address this, Faster R-CNN introduces the RoI (Region of Interest) pooling layer, which maps each candidate box to a fixed-size feature map, allowing for shared computation. After RoI pooling, each candidate box is fed into two parallel branches of fully connected layers. One branch is for classification, determining whether the candidate box contains an object, while the other is for bounding box regression, refining the position of the candidate box more accurately.

In this article, an improved network is built upon the original architecture by utilizing ResNet as the backbone. The neck includes a bifpn (bi-directional feature pyramid network) that efficiently connects across scales and fuses features with weights. This results in better fused features for classification and regression tasks. Deep separable convolutions are introduced to reduce redundant parameter volume. Lastly, soft-NMS is incorporated, suppressing scores based on intersection over union (IoU) and adjusting confidences, significantly preserving true boxes and enhancing accuracy. The network structure diagram for the improved version is as follows:

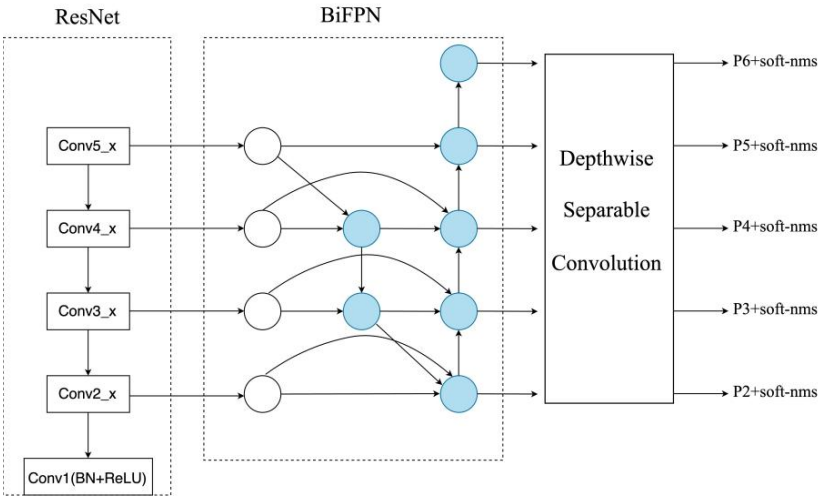


Figure 2. Improved Network

2.2. Backbone network ResNet

ResNet (Residual Networks) is a convolutional neural network (CNN) architecture in the field of deep learning, proposed by Kaiming He and others in 2015. The primary contribution of ResNet lies in its solution to the issues of vanishing gradients and exploding gradients during training of deep neural networks, enabling the training of even deeper networks and achieving improved performance. Traditional deep neural networks tend to encounter the vanishing gradient problem as the number of layers increases, where gradients diminish during backpropagation, rendering ineffective parameter updates. ResNet addresses this by introducing "residual blocks." Each residual block includes a skip connection (also known as a "shortcut" or "identity" connection), allowing gradients to flow directly, thereby preventing gradient vanishing.

A typical residual block consists of two convolutional layers. If the input is denoted as x and the output of the residual block is denoted as F(x), the computation of the residual block can be expressed as: F(x) = x + H(x), where H(x) represents the result of the convolutional layers within the residual block. If the network considers the identity mapping (input and output consistency) to be optimal, H(x) can be set to a function close to zero, effectively reducing the residual block to an identity mapping. This design allows the network to selectively learn residuals, adapting to various feature extraction needs.

ResNet is an architecture that tackles the problem of vanishing gradients in training deep neural networks by introducing residual blocks and skip connections. Its innovative design enables the training of very deep networks, leading to outstanding performance across a variety of computer vision tasks.

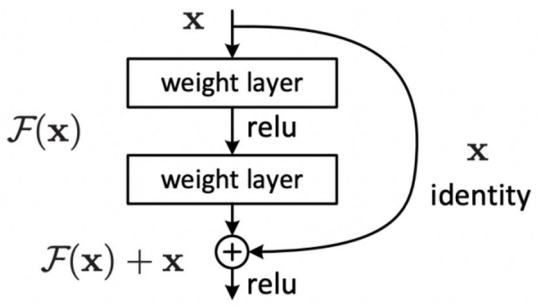
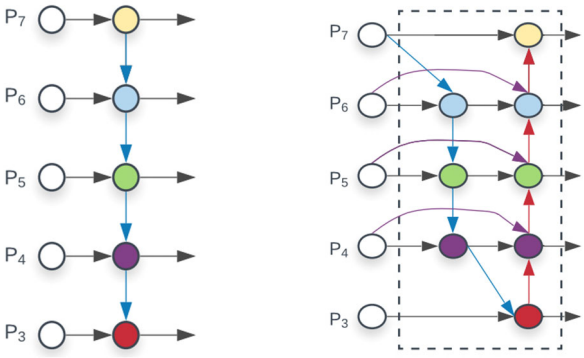


Figure 3. Residual Block

2.3. BIFPN

BiFPN stands for Bidirectional Feature Pyramid Network, which is a neural network architecture widely used in tasks such as object detection and semantic segmentation. It effectively integrates feature information across multiple scales. BiFPN draws inspiration from Feature Pyramid Networks (FPN) and aims to address the fusion of features at different scales, allowing the network to capture multi-scale information of objects in various levels of feature representation. BiFPN improves upon FPN by introducing bidirectional information flow, further enhancing feature representation.The following image illustrates the differences between FPN and BiFPN.



(a) FPN (b) BiFPN

Figure 4. Comparison between FPN and BiFPN

The key idea behind BiFPN is to perform not only top-down information fusion among higher-resolution feature pyramid levels, but also bottom-up information fusion among lower-resolution layers. This bidirectional flow structure enables better propagation of feature information at different scales while maintaining a balance between detailed and semantic information. The architecture of BiFPN involves the following key steps: top-down information flow, bottom-up information flow, and bidirectional information flow balance. These steps enable BiFPN to capture rich semantic information, convey detected details and local features, allowing the network to adapt better to targets of different scales.

2.4. Depthwise separable convolution

Depthwise Separable Convolution is at the core of MobileNet, an architecture that employs factorized convolution. Its goal is to reduce model computation and parameter count, thereby enhancing computational efficiency while maintaining a certain level of performance. It proves especially valuable in scenarios with limited computing resources, such as mobile devices. Traditional convolutional layers consist of two main components: Spatial Convolution and Cross-Channel Convolution. Depthwise Separable Convolution breaks these components apart into two steps: Depthwise Convolution and Pointwise Convolution. By combining these steps, Depthwise Separable Convolution effectively reduces computation and parameter requirements. The Depthwise Convolution learns spatial features, while the Pointwise Convolution learns relationships between channels. This separation allows the model to learn and represent features more efficiently. This structure is particularly suitable for resource-constrained environments like mobile devices, as it can maintain relatively high performance while reducing computational demands.

2.5. Soft-NMS

NMS (Non-Maximum Suppression) has consistently been a component of many object detection algorithms. The algorithm selects the proposal box with the highest score among the model's predictions and discards other boxes that significantly overlap with the chosen one, and this process is executed recursively. In the context of bottle cap detection, various categories of actual boxes exhibit overlapping and containing portions. However, in NMS algorithms for detecting overlapping objects, only boxes with higher scores are retained, which can often result in mistakenly discarding

target boxes. To prevent the omission of actual detected objects, the present algorithm employs Soft-NMS for suppressing predicted boxes.

Soft-NMS doesn't directly discard boxes with IoU above the threshold; rather, it reduces confidence scores for suppression. Building upon NMS, the algorithm designates M as the box with the highest score and bi as the box to be processed. After calculating their IoU, a weight function is applied. Boxes with higher overlap with M experience more significant score attenuation. Based on this attenuation, scores are determined, and boxes falling below the threshold are discarded. By scoring based on overlap instead of a uniform zeroing, Soft-NMS not only eliminates surplus boxes but also retains object boxes, effectively enhancing detection accuracy. Due to the possibility of discontinuity in linear weighting, Soft-NMS can also be modified to use Gaussian weighting. The specific formula is as follows:

$$s_i = \begin{cases} s_i, & iou(M,b_i) < \mathrm{N_t} \\ s_i(1-iou(M,b_i)), & iou(M,b_i) \geq \mathrm{N_t} \end{cases},$$

$$s_i = s_i e^{-\frac{iou(M,b_i)^2}{\sigma}}, \forall b_i$$

3. Analysis and Discussion

3.1. Dataset Preprocessing

The data is sourced from a well-known tile enterprise in Foshan, Guangdong Province. Data collection was conducted by setting up specialized photography equipment on the production line to gather real-time production process data. It covers a wide range of common defects in the tile production line, including powder spots, corner cracks, glaze drips, ink breaks, ink drips, B holes, soiling, edge cracks, chipping, tile residue, white edges, etc. The dataset has been publicly released on the Tianchi Data website.

(1) Data Details and Labeling Process

During the data collection process, certain defects could only be captured from specific angles. Three images were taken for each tile, including low-angle monochromatic images, high-angle monochromatic images, and color images.

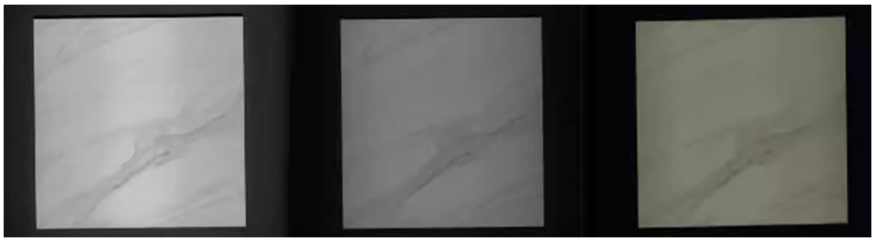
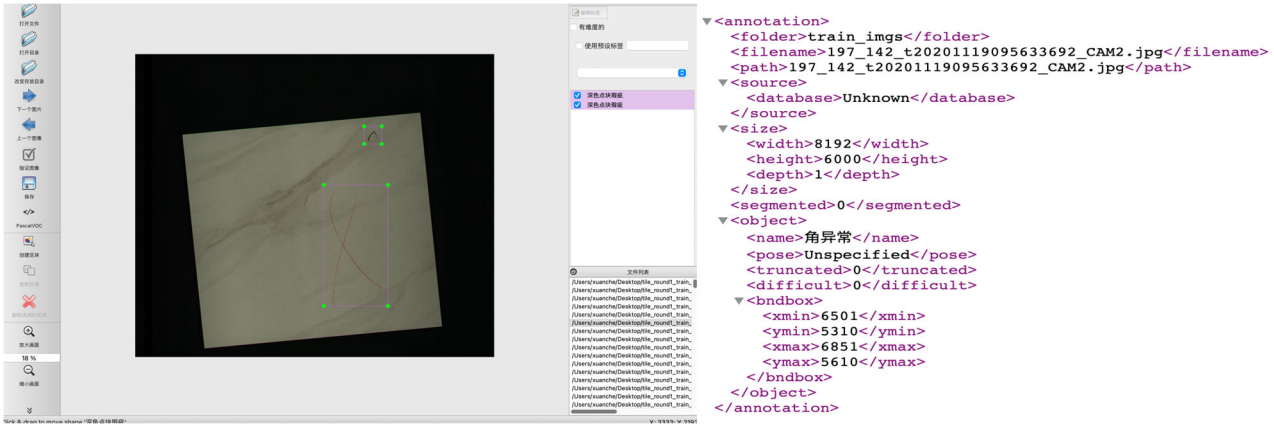


Figure 5. Example Fig

The data is annotated using the VOC format for detection. The images were processed into a standard VOC dataset format using the object detection annotation tool, LabelImg. Based on the annotated bounding box positions, new XML files were generated containing the file name, category number, image width and height, as well as the bounding box (bbox) coordinates information. The bbox includes four points (xmin, ymin, xmax, ymax) that determine the position of the detected tile defects. The annotation process is illustrated in the figure.

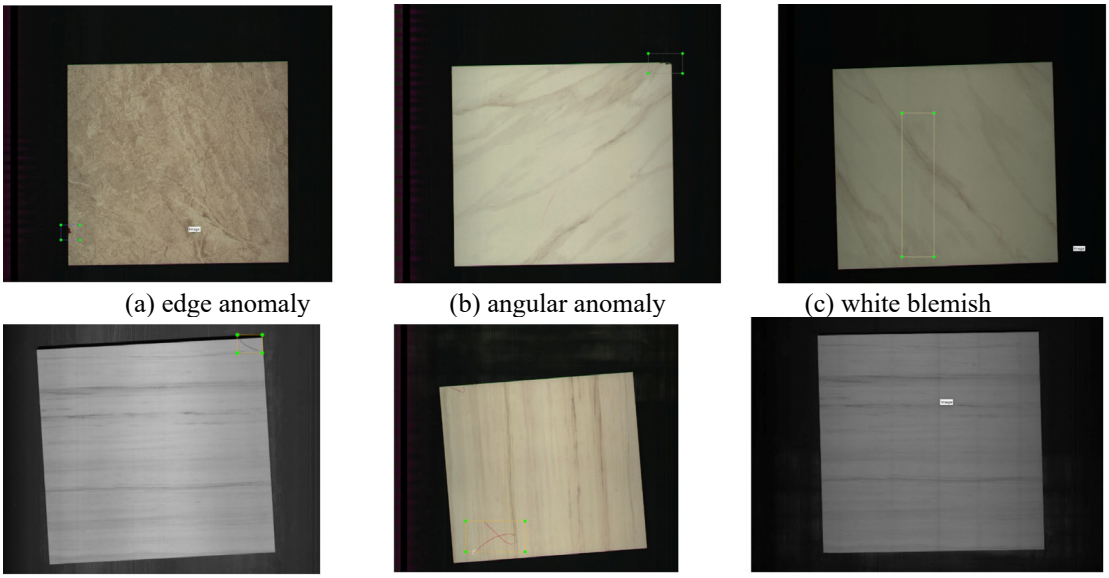


(a) LabelImg (b) xml document

Figure 6. Dataset Creation

(2) Tile Defect Classification

During the production process, based on the characteristics, sizes, and positions of tile defects, the defects other than the background were categorized into six classes: edge anomalies, angular anomalies, white blemish, dark blemish, light blemish and aperture defects.



(a) edge anomaly (b) angular anomaly (c) white blemish

(d) dark blemish (e) light blemish (f) aperture defects

Figure 7. Defect Category

(3) Dataset Partitioning

Reasonable dataset partitioning can enhance training efficiency and during the model construction process, it is important to assess the model's performance status and determine whether it is overfitting or underfitting. Therefore, in this experiment, the dataset was partitioned into training, validation, and test sets in an 8:1:1 ratio. The training set consists of 12,003 images, the validation set includes 1,345 images, and the test set contains 1,627 images. The specific partitioning is detailed in the table：

Table 1. Dataset partitioning table

category	background	edge anomaly	angular anomaly	white blemish	light blemish	dark blemish	aperture defects
num	0	1	2	3	4	5	6
Training	1290	1715	1428	1911	2301	1875	1928
Validation	175	215	201	195	242	151	174
Test	121	231	221	259	287	212	198

3.2. Experimental Evaluation Metrics

IOU refers to the Intersection over Union, which is the result of the intersection of predicted box A divided by the union of true box B. It measures the overlap between algorithm-predicted boxes and true boxes, serving as a measure of the model's performance. The experiment primarily employs IOU intersection-over-union ratios to compute the Average Precision (AP) and mean Average Precision (mAP) as well as recall rate as evaluation metrics. The calculation formulas are as follows, where TP (True Positive) represents true positive predictions, TN (True Negative) signifies true negative predictions, FP (False Positive) indicates false positive predictions, FN (False Negative) denotes false negative predictions, and n represents the number of detected sample categories, with 6 categories in this study.

$$IoU = \frac{A \cap B}{A \cup B}$$

$$\text{precision} = \frac{TP}{TP + FP}$$

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

$$\text{AP} = \int precision(recall)d(recall)$$

$$mAP = \frac{\sum AP}{n}$$

3.3. The Experimental Process of the Improved Network for Tile Defect Detection

In this experiment, the Faster R-CNN model was utilized with transferred learning from pre-trained weights on the PASCAL VOC dataset. During training, the dataset was augmented with random flips, enhanced brightness, and other data augmentation strategies. The input images underwent Multi-Scale Training (MST), which involves training the model with various input sizes to effectively improve multi-scale object detection and enhance the network's robustness. The input scales were set to (640, 640) and (1280, 1280). The batch size was set to 8, influenced by the image size. An initial learning rate of 0.0001 was employed, and a Warmup optimization algorithm, also known as a learning rate warmup method, was used for adjusting the learning rate. Warmup

involves starting the training process with a smaller learning rate, gradually increasing it to a predefined value after training for a certain number of epochs.

3.4. Comparison of Results Before and After Network Improvement

In order to verify the effectiveness of the improved RetinaNet network for bottle cap defect detection, this experiment trained the original network, the network with Swin-Transformer as the backbone, the network with the neural architecture search FPN as the neck, and the network with soft-NMS for suppressing predicted boxes. These networks were tested on the test set, and their performance was compared. The results are shown in the table:

Table 2. Detection effect table

	ResNet	BiFPN	DepthWise	soft-nms	mAP/%	memory
1					57.2	3598
2	√				62.3	3984
3		√			62.7	3894
4			√		60.7	3137
5				√	59.8	3598
6	√	√	√	√	70.4	3827

By employing ResNet as the backbone in Faster R-CNN, the mAP improved by 5.1%. Introducing BiFPN resulted in a 5.5% mAP enhancement. Finally, incorporating soft-NMS achieved an accuracy of 59.8%, yielding a 2.6% increase and an overall improvement of 13.2% in precision. Based on the experimental results, it is evident that each module showed varying degrees of progress, with the backbone showcasing the most noticeable improvement. The precision comparison chart further underscores the elevated recognition performance of the improved network on the tile defect detection dataset, along with the efficacy of the added modules.

3.5. Accuracy and Training Performance for Each Category

From Figure 8, it is evident that the accuracy for 'edge anomaly' reaches 80.3%, 'angular anomaly' achieves 89%, and even the 'aperture defects', which can be somewhat ambiguous even to human observers, achieve an accuracy of 82.1%. The accuracies for 'white blemish', 'light blemish', and 'dark blemish' have also shown significant improvement. These results indicate that the improved network in this study performs well in tile defect detection, effectively enhancing the network's capability to detect small target defects.

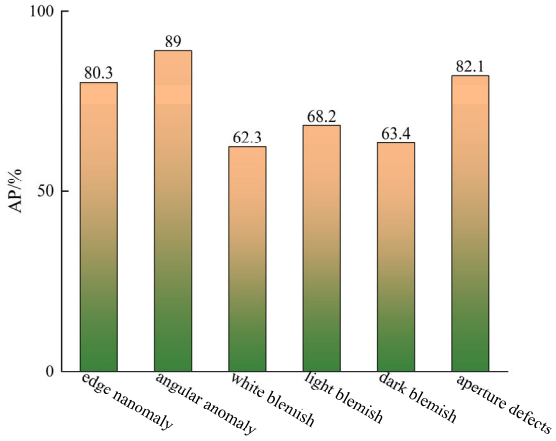


Figure 8. Defect type accuracy

To provide a clearer demonstration of the progress achieved by the improved network, the following figure presents a comparison between the original network and the network presented in this study. It is evident that the enhanced version of the network in this study achieves a higher level of defect detection recognition, effectively preventing false negatives, and meets the factory's requirements for tile defect detection.

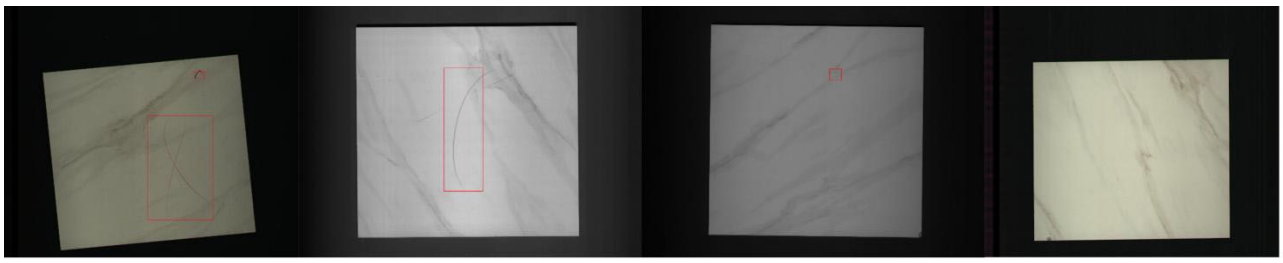


Figure 9. Faster-RCNN detection result

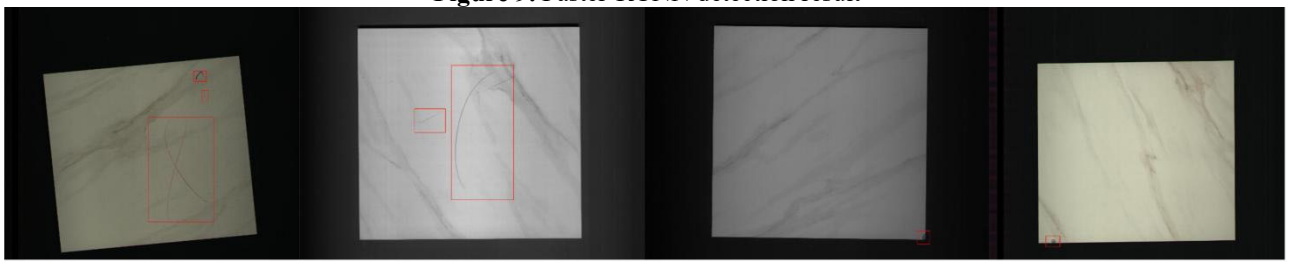


Figure 10. Improved Network detection result

4. Summary

The tile defect dataset exhibits a wide range of defect sizes and significant variations in their characteristics. The distribution of easy-to-detect and hard-to-detect categories is uneven. In this study, a Faster R-CNN model with ResNet as the backbone was employed to address these challenges. This approach not only enables the model to focus on less readily detectable features, resolving the issue of data imbalance, but also ensures stability and enhancement through the use of a BiFPN-based neck. The adoption of the Pascal VOC format for the dataset and the COCO evaluation metrics adheres to the common detection methodology in object detection algorithms, resulting in improved accuracy and dataset robustness compared to the original network. The algorithm's feasibility and effectiveness are validated.

Although the incorporation of ResNet and the more complex FPN increases the number of parameters, the use of depthwise separable convolution reduces detection time, still meeting the requirements for tile defect detection on factory production lines. This improvement offers significant practical significance for modern, intelligent construction factories. The current enhanced network focuses solely on tile defect detection. Future considerations include incorporating various features into detection targets and introducing suitable attention mechanisms into the neck component to optimize detection speed, aiming for a more broadly applicable and efficient network model.

References

[1] Huang HuiNing. Global Ceramic Tile Development Status and Implications[J].Foshan Ceramics.Vol. 25(2015),p.1-11.

[2] SHOCKLETTI.Review on application of surface defect detection[J].Electronic Technology.Vol. 49(2020),p.189-191.

[3] Pu YuXiang.Design of Bottle Cap Visual Inspection System Based on Machine Vision[J].Light Textile Industry and Technology. Vol. 49(2020),p.30-33.

[4] Yang Cui.Speckled Micro-defects Detection Based on Deep Neural Network Learning[J].Journal of Anqing Normal University(Natural Science Edition).Vol.28(2022)No.4,p.51-56.

[5] Biradara M, Shiparamattia B,Patilb B .Fabric Defect Detection Using Deep Convol-utional Neural Network[J].Optical Memory and Neural Networks.Vol. 30(2021),p.250-256

[6] Girshick R, Donahue J , Darrell T , et al.Rich feature hierarchies for accurate object detection and semantic segmentation[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.Beijing,Jul 6,p.580-587.

[7] Ren S Q, He K M , Girshick R, et al.Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J].Neural Information Processing Systems. 6(2017), p.1137-1149.

[8] 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.Santiago,Dec 7-13,p.779-788.

[9] Lin T, Goyal P, Girshick R, He K M, et al. Focal Loss for Dense Object Detection[C]. 2017 IEEE International Conference on Computer Vision. Venice.Italy,Oct 22-29, p.2999-3007.

[10] Zhang ZuoRen.ResNet-Based Model for Autonomous Vehicles Trajectory Prediction [C].2021 IEEE International Conference on Consumer Electronics and Computer Engineering . Guangzhou,Oct 3,p.565-568.

[11] Mingxing Tan, Ruoming Pang, Quoc V. Le.EfficientDet: Scalable and Efficient Object Detection[C].Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2020).Seattle,June 14-19,p.10781-10790

[12] Andrew G. Howard, Menglong Zhu, Bo Chen,et al.MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications.Goggle.

[13] Bodla N,Singh B,Chellappa R.Soft-NMS Im-proving Object Detection with One Line of Code [C].2017 IEEE International Conference on Computer Vision.Venice,Aug 9,p.5562-5570.