Hindawi
Journal of Sensors
Volume 2022, Article ID 7530361, 12 pages
https://doi.org/10.1155/2022/7530361

Research Article

YOLOv5-PD: A Model for Common Asphalt Pavement Defects Detection

Yiming Xu, Fei Sun, and Li Wang

School of Electrical Engineering, Nantong University, Nantong, Jiangsu 226019, China

Correspondence should be addressed to Li Wang; lwee@ntu.edu.cn

Received 17 August 2022; Revised 26 September 2022; Accepted 12 November 2022; Published 29 November 2022

Academic Editor: Rajesh Kaluri

In asphalt pavement detection, the defect scale changes greatly, mainly including mesh cracks, patches, and potholes. In the case of large scale, the texture feature is not clear, and the information is easily lost in the feature extraction process. Correspondingly, the number of small-scale holes is often very large, which also puts forward higher requirements for the detection model. In view of the above problems, this paper proposed a model for common asphalt pavement defects detection called YOLOv5-PD. In order to reduce the loss of information and expand the receptive field of the model, Big Kernel convolution was used to replace a part of the convolution in the original CSPDarknet. The texture feature information of the cracks is retained to the greatest extent. In order to enhance the detection performance of small defects, convolution channel attention mechanism was added after each feature fusion layer, and performs attention processing on the feature map after concat to find the defect location. This study used a public pavement defect dataset from Brazil. In this work, ablation experiments were carried out according to the task scenario, and the improved effects were compared and analyzed. The proposed model is compared with other versions of models and advanced models, which proves the superiority of the proposed model. The mAP of proposed model reached 73.3% and the model inference speed reached 41FPS, which can meet real time engineering application requirements.

1. Introduction

In recent years, the road traffic has been in a state of rapid development, and the safety inspection in the field of road traffic has become increasingly important. It is put on the agenda to complete some specific tasks with efficient AI algorithms. Road maintenance is very important in the field of road traffic. With the increasing scale and number of highway projects, road maintenance tasks are also increasing, which brings about the problem of rapid rise in labor costs. Pavement defect detection is a necessary step for road maintenance and management [1]. Using deep learning algorithm to predetect the road surface can save a lot of costs for manual inspection and repair. Through the predetection of pavement defects, the road damage can be detected in advance on a large scale, and the defects can be screened in advance for the subsequent manual inspection. This makes it easier for road managers to understand road damage.

The focus of this study is to detect the common defects of asphalt pavement. There are many different subdirections in road safety inspection, including inspection of safe distance for multiple vehicles, and inspection of damage to bridges, tunnels, and roads. Workers use various sensors to collect distance data and image data, and use machine learning methods to process the data to obtain prediction or detection results, so as to reduce the risk of accidents. Compared with bridges and tunnels, the number of roads is very large. For example, the total mileage of roads in China has reached 5.28 million kilometers. If the defects of the bridge are not found in time, it may cause serious traffic accidents and major economic losses. The defects of tunnel may cause the loss of life and property safety of drivers. Therefore, the defect detection of bridges and tunnels needs more detection accuracy and recognition accuracy. On the contrary, it is the detection of road defects. The number of road defects is very large. Common road defects include mesh cracks and potholes. Considering that the patches of roads is often done

on significant defects, the repaired roads are more uneven and more prone to produce more defects, so the patches were added as the third type of pavement defects. Mesh racks, potholes, and patches are very common, and will not directly threaten the safety of personnel and vehicles. However, with the increase of time, the safety risks are also increasing, so the detection of road defects is also important.

The speed and accuracy of defect detection should be considered in the detection of three types of road defects in this study. Therefore, there are great differences in the selection of detection models between road defect detection and bridge and tunnel defect detection. The one-stage network in the deep learning model is more suitable for such tasks. The one-stage network gives consideration to both accuracy and speed, regards detection as a regression problem, and uses only one neural network to simultaneously predict the location and category of the boundary box. YOLO series has always been one of the representative algorithms of the one-stage network. In recent years, the algorithm has developed to YOLOv7 version. However, considering the robustness of the model, this study did not use the latest version of the model, but rather conservatively used YOLOv5, which has been verified by many people, and made corresponding improvements to the model for problems related to road defect detection task scenarios. While ensuring the detection speed, improve the detection accuracy of various types of defects as much as possible.

The contribution of the model proposed in this study is:

(1) An improved asphalt pavement defect detection system based on YOLOv5 model is proposed. Its accuracy and reasoning speed can meet the needs of actual pavement maintenance tasks

(2) Big Kernel convolution is used to replace the first layer convolution in CSPDarknet-53, which improves the detection accuracy of network crack defects and verifies the feasibility of large core convolution

(3) CBAM is added after each feature fusion layer to comprehensively improve the model detection effect and compensate the detection accuracy of path defects

(4) The results are compared with the commonly used advanced models such as Faster-RCNN and YOLOX, highlighting the performance enhancement of the proposed model

(5) Experiments prove that the proposed detection model is superior to other popular detection models, thus verifying the positive impact of the combination of Big Kernel convolution and CBAM

2. Literature Review

In this paper, the research background and existing work in this field have been extensively investigated. Relevant research is referred as the background support of this study.

Jhaveri et al. [2] discusses and summarizes the applicability and applications of machine learning to various problems in the real world. This study can be used as a benchmark for machine learning in a variety of applications and real-world situations.

Reddy et al. [3] studied the dimension reduction technology of big data applied to machine learning. This research has studied two outstanding dimension reduction techniques on the current popular machine learning algorithm, linear discriminant analysis and principal component analysis, and compared the dimension reduction selection and results of high-dimensional data and low-dimensional data.

Sagar et al. [4] elaborated the importance of machine learning technology from the perspective of data security, and reviewed the latest methods for more effective application of machine learning technology to meet the current world security requirements. And the vulnerabilities in the machine learning model are also evaluated.

Lakshmanna et al. [5] studied the application of deep learning technology in machine learning in various fields of the Internet of Things. This study discussed various deep learning methods and processes, and summarized the main report work of deep learning in the Internet of Things field.

Gadekallu et al. [6] proposed a deep learning model based on crow search for gesture recognition tasks in the field of human-computer interaction. The research uses open data sets and crow search algorithm to search the best super parameters of the convolutional neural network, which makes the model achieve 100% training and testing accuracy, and verifies the superiority of the deep learning model over the traditional machine learning model.

Kaluri et al. [7] conducted a series of studies on battery life prediction in the Internet of Things framework in the marine environment. In this study, data preprocessing is carried out first, then rough set theory is used to extract features, and the results are inputted into the depth neural network to obtain optimal prediction results. This research expands the research boundary of the Internet of Things.

In the field of road traffic in the Internet of Things, many researchers have applied the deep learning model to the detection of road defects.

Feng et al. [8] proposed a structure that uses information contained in feature maps at different levels so that all information can contribute to classification, avoiding the problem of losing original information during downsampling in traditional CNNs. However, the rationality of the structure still needs to be further proved, and the research did not give the reasoning speed of the proposed model.

Park et al. [9] proposed a deep learning model for automatic crack detection in car black box images using convolutional neural networks, which divides pavement features into cracks, pavement markings, and intact areas, and the architecture achieves an accuracy of 90.45%. However, this study did not make a further detailed division of road defects.

Ju et al. [10] proposed a Fast-RCNN-based Crack Depth Network (CrackDN), by embedding a sensitivity detection network in parallel with a feature extraction Convolutional

Neural Network (CNN), the two are then connected to a region proposal refinement network (Region Case Refinement Network, RPRN) for classification and regression, the final detection average accuracy is higher than 90%. However, the study found that various road markings have a great impact on the performance of the model.

Qu et al. [11] proposed a mixed-territory pavement crack detection algorithm based on convolutional neural network. Two different deep learning models are used in the two tasks of crack classification and detection. The classification part modifies the output dimension of the FC2 layer of the LeNet-5 model. The accuracy in the CFD dataset and Cracktree 200 dataset is higher than that of U-Net and Percolation. However, this model is greatly affected by the interference noise of the background, and its effect is poor for images with more complex cracks.

Du et al. [12] adopted the deep learning-based target detection framework YOLO network to detect pavement damage, the comprehensive detection accuracy reached 73.64%, and the processing speed reached 0.0347 s/pic. This study prepared a large PD dataset, but did not improve the proposed model.

The objects of the above research are all in the field of road surface detection, but their data sets are different, which leads to a great difference in their accuracy. In general, the model based on one-stage network has higher accuracy, while the model based on two-stage network has faster speed. As a representative of one-stage network, YOLO is widely used in various fields. On the premise that YOLO guarantees real-time, what researchers need to do is to modify the model according to the actual task scenario to meet the accuracy requirements.

This paper proposed YOLOv5-PD model based on YOLOv5 and is aiming at asphalt pavement defect scene to detect pavement defects, which are divided into three categories: mesh cracks, patches, and potholes [13]. In the system design of pavement detection, the improved YOLOv5 model was used to complete the identification and classification of defect detection, and the deep learning framework Pytorch was used to process a good dataset to train the model.

3. Background

3.1. Convolutional Neural Network. Artificial neural network is an algorithmic mathematical model that simulates the structure and behavior of biological nervous system and performs distributed parallel information processing. Early research uses fully connected neural network to process image data, but when fully connected neural network processes images, spatial information will be lost when the image is expanded into a vector. In addition, fully connected neural network has too many parameters, making training difficult, and too many parameters will also lead to loss of spatial information, which causes the network to overfit quickly. In order to solve the above problems of processing images, some studies have proposed convolutional neural networks [14].

Convolutional neural network is mainly composed of input layer, convolution layer, ReLU layer, pooling layer, and fully connected layer. The convolutional layer is the core layer of the convolutional neural network, in which the convolution kernels play a role similar to filters. These convolution kernels downsample the image step by step to extract the upper abstract information of the image. Generally, a pooling layer is periodically inserted between consecutive convolutional layers. The function of the pooling layer is to gradually reduce the spatial size of the data, reduce the number of convolution kernel parameters, reduce the amount of calculation, and control the overfitting of the model. The commonly used pooling layer has max pooling, average pooling, and L-2 normal pooling. The ReLU layer is a nonlinear activation function layer, which adds nonlinear expression capabilities to the model. In addition to ReLU, the commonly used activation functions include sigmoid, Tanh, and LeakyReLU. Finally, the fully connected layer in the convolutional neural network converts the two-dimensional feature map image processed by the convolution layer, ReLU layer, and pooling layer into a one-dimensional vector, multiplies this vector, reduces its dimension, and inputs it to softmax. The corresponding score of each category is obtained in the layer, that is, the probability of belonging to each category, and then the objects in the image are classified. The above are the main steps of image processing and image recognition by convolutional neural network. The general convolutional neural network structure is shown in Figure 1.

By taking a two-dimensional image of the pavement surface, inputting the image into the convolutional neural network, and training the parameters, the automatic identification, classification, and location of asphalt pavement defects can be realized.

3.2. YOLOv5 Model. The YOLO series algorithm is a typical one-staged target detection algorithm, which uses the anchor box to combine the problems of classification and object localization, taking into account both speed and accuracy [15]. The YOLOv5 model selected in this article is the latest version of the YOLO series. It inherits YOLOv3 and YOLOv4 [16]. The network structure is still divided into four parts: Input, Backbone, Neck, and Prediction. Compared with YOLOv3 and YOLOv4, YOLOv5 has certain innovations in different parts of the network.

On the data input side, YOLOv5 adopts the Mosaic data enhancement method and uses four pictures to randomly scale, randomly crop, and randomly arrange. Then, these pictures are spliced. A lot of small targets are added during the random scaling process to enrich the dataset and optimize the detection effect. Each time YOLOv5 is trained, it can adaptively calculate the best anchor box values in different training sets. In addition, in order to process image data faster, YOLOv5 uses adaptive image scaling, which adaptively adds the least black borders to the original image, and at the same time scales to a standard size, minimizing information redundancy.

In the Backbone part, YOLOv5 adopts the Focus structure to slice the input image. For example, YOLOv5s

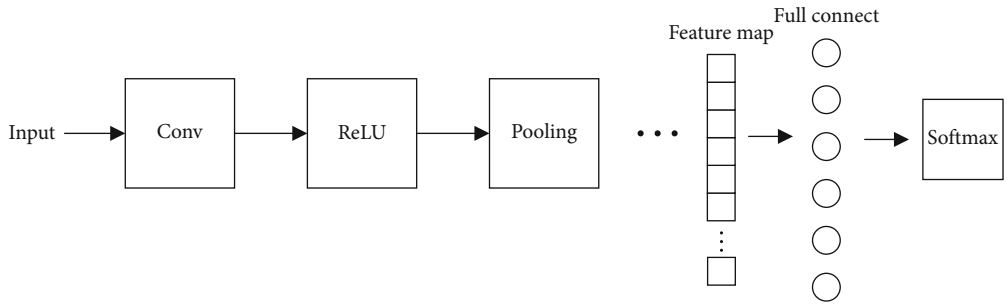


Figure 1: Convolutional neural network.

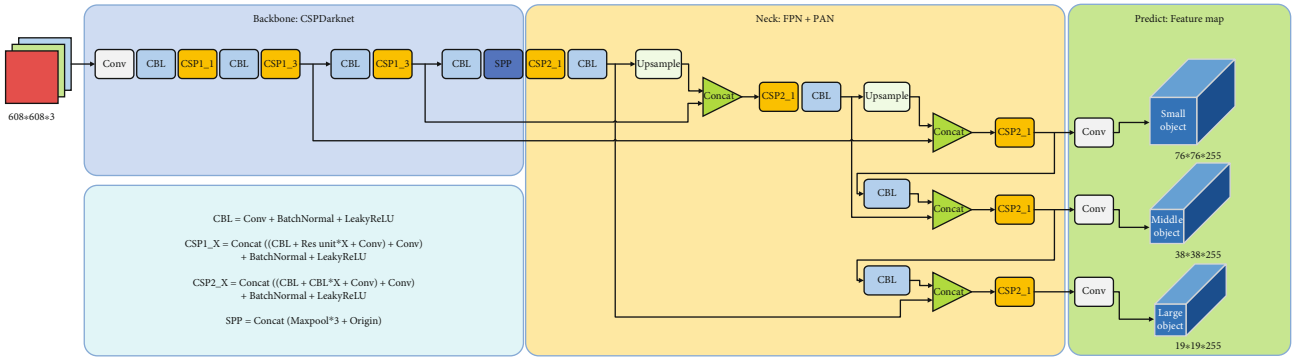


Figure 2: YOLOv5 model. Large feature maps predict small scale targets, and small feature maps predict large scale targets.

converts 608*608*3 image slices into 304*304*12 feature maps, and then passes through a volume of 32 convolution kernels product to obtain a feature map of 304*304*32.

In the Neck part, YOLOv5 uses the FPN and PAN structure for feature fusion at different scales. FPN is used to solve the multiscale problem in object detection [17]. It uses high-level features for upsampling and low-level features for top-down connections. Each layer is predicted, which greatly improves the performance of small object detection. In addition, YOLOv5 also adds two PAN structures (from PANet) [18]. Through the combination of FPN and PAN, different detection layers are aggregated from different backbone layers to improve the detection accuracy of objects of different scales.

In the final Prediction part, YOLOv5 uses DIOU_Loss as the loss function of bounding box, taking into account the overlap area between the prediction frame and the target frame, the distance between the center points, and the aspect ratio, which is better than IOU_Loss and GIOU_Loss [19]. The original YOLOv5 network structure is shown in Figure 2.

YOLOv5 has a series of model versions. Among them, YOLOv5s has the smallest depth and the smallest feature map width. *m*, *l*, and *x* are all deepened and widened on the basis of *s*. After comparing the four models, it can be found that from *s* to *m*, *l*, and *x*, the depth and width of the model are increasing, the parameters are getting more and more, the speed is getting slower and slower, but the accuracy is getting higher and higher. In this paper, YOLOv5l with moderate width and depth is used for pavement defect detection.

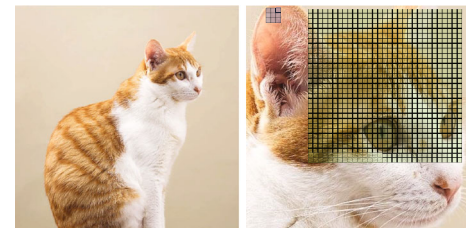


Figure 3: Big kernel convolution.

4. Model Improvements

4.1. Big Kernel Convolution. In the past, Alex-Net used 11*11 convolution, but after the advent of VGG, Big Kernels are gradually being phased out. Since then, the network structure design of CNN has gradually changed from the design of shallow and large convolution kernels to the design of deep and small convolution kernels. The reason for this phenomenon is that large kernels have been found to be less efficient and sometimes even reduce model accuracy. But as CNNs continue to develop, more and more training techniques are proposed. This conclusion may be changing.

Big Kernel convolution and structural reparameterization were recently proposed by Ding et al. [20]. During the development of CNN, different network depths, widths, and input resolutions were tried one by one, but the kernel size parameter setting is always defaults to 3*3 or 5*5. Ding et al. proposed that the large convolution was not used in the past, but it does not mean that it cannot be used now. With the blessing of modern CNN design, Big Kernel convolution can improve the receptive field of the model without

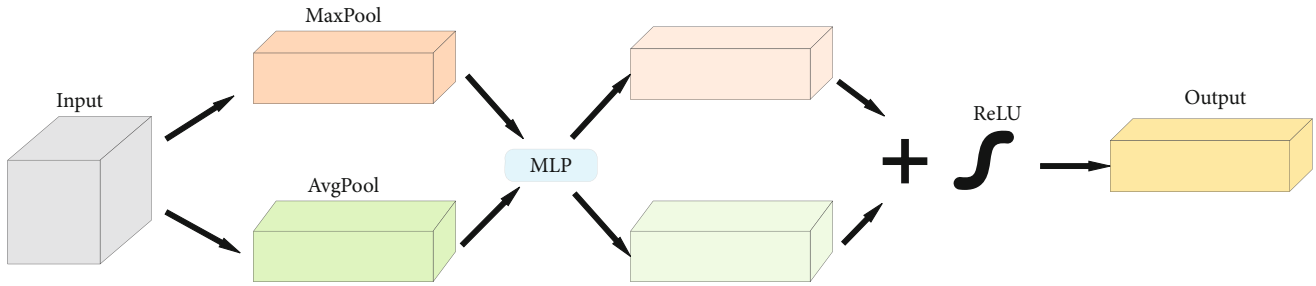


Figure 4: CAM.

increasing too many parameters, and finally improve the detection accuracy of the model.

At present, Transformer is very popular in academia, and its basic component is self-attention [21]. The essence of self-attention is to perform Query-Key-Value operation in a full-scale or larger window. Ding et al. conjectured that the full-scale or larger window is the key to such powerful performance of transformer. Because the receptive field of the detection model is increased. In CNN, use of Big Kernel convolution is a way to increase the receptive field and may inject new impetus into the current CNN development.

Inspired by the large kernel convolution proposed by Ding et al., this paper makes an attempt in the pavement defect model based on YOLOv5. The first convolutional layer in CSPDarknet of the original network is removed. A 31*31 Big Kernel convolution module is added. The effective receptive field of Big Kernel convolution is larger than the receptive field of original model receptive field, which can effectively reduce the information loss caused by layer-by-layer convolution to the image, and improve the precision of detecting large-scale objects such as mesh cracks and patches. The effect of Big Kernel Convolution is shown in Figure 3.

4.2. Convolutional Block Attention Module. Many different attention mechanisms have been proposed. Among them, CBAM (Convolutional Block Attention Module) is favored by many researchers, because CBAM is a general-purpose module that can be used in various models to improve the accuracy of model detection [22]. CBAM is the convolutional attention module, which consists of a spatial attention mechanism and a channel attention mechanism. In this paper, CBAM is added after the feature fusion layer to generate better attention maps.

CAM (Channel Attention Mechanism) takes the input feature map, performs average pooling and max pooling on it, then uses MLP (Multilayer Perceptron) for aggregation, and finally generates a channel attention map through a nonlinear activation function. The CAM module structure is shown in Figure 4.

SAM (spatial attention mechanism) is a supplement to channel attention. First, average pooling and max pooling are combined on the channel axis, and then the combined feature map is convolved, and spatial attention map is finally generated by a nonlinear activation function. The SAM module structure is shown in Figure 5.

Feature map has been processed by CAM and SAM; it has become a feature map that highlights the part of interest. The CBAM module structure is shown in Figure 6.

4.3. YOLOv5-PD. This paper improved the original YOLOv5 model to make it more suitable for the task scene of asphalt pavement defect detection. Finally, YOLOv5-PD was obtained. The cracks are small but dense, and it is easy to lose image feature information in the process of layer-by-layer convolution. In response to this problem, this paper considers improving the receptive field in the shallow stage of the model, so as to retain the shape characteristic information of the mesh crack to the greatest extent, and achieve the purpose of improving the crack detection accuracy. Inspired by RepLKNet, Big Kernel convolution is replaced to the first layer of convolution in CSPDarknet in the YOLOv5-PD model, increasing the effective receptive field of the model, and enhancing the model's response to mesh cracks. The texture information is retained, and the detection accuracy of the model for mesh cracks is improved.

There are many small defects in the asphalt pavement in the real scene. For these small defects, this paper considered adding a smaller anchor box to improve the detection effect, but the effect is not ideal. The reason for this phenomenon may be that the scale of defect target changes too much. If small anchor is added, the detection accuracy of large-scale flaws will even be reduced. Considering the above problems, this paper abandoned adding small-scale anchor box, but added convolution channel attention mechanism after the feature fusion layer, which comprehensively improves the sensitivity of the model to defect at various scales, so that the model can also take into account the small defects detection. Using CBAM can improve the detection accuracy of small defects without affecting the detection accuracy of large-scale defects.

The YOLOv5-PD model structure in this paper is shown in Figure 7. The input image is first convolved by a Big Kernel convolution, which preserves more information for the subsequent convolution. Next, the feature map is transformed into a tensor with 1024 channels through a series of CBL blocks and CSP blocks. In the Neck part, the feature map is concated with the feature map in the Backbone part through upsampling, and then the feature map after attention processing is generated through a CBAM. In the Neck part, four concats have been conducted. The purpose of concat is to fuse the information of shallow features and deep

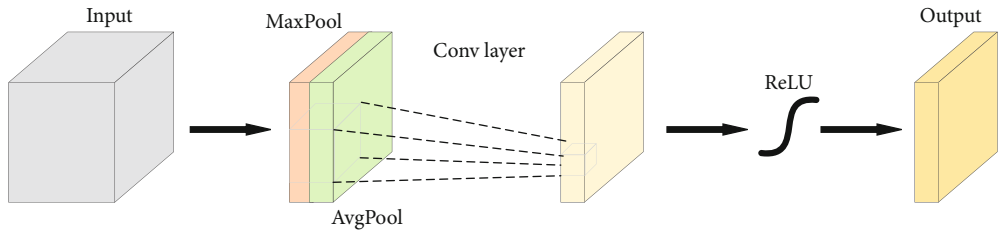


Figure 5: SAM.

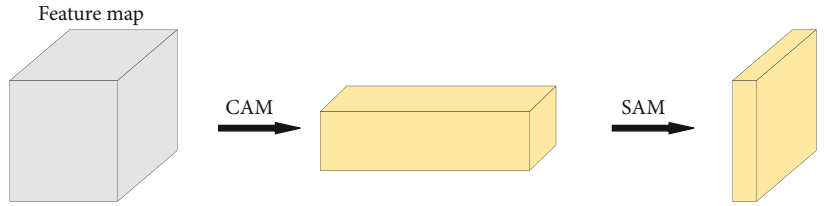


Figure 6: CBAM.

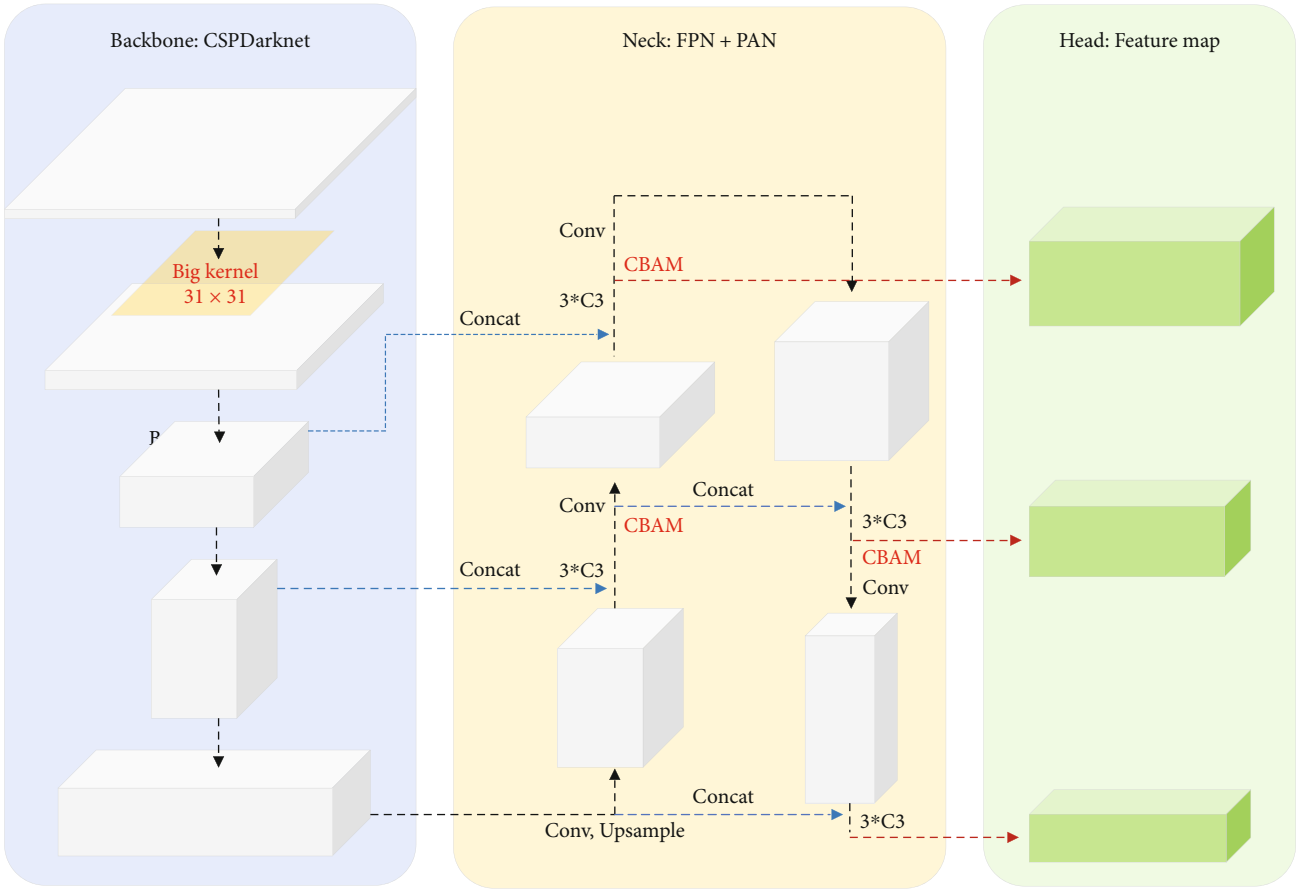


Figure 7: YOLOv5-PD.

features. The tensors of three sizes are processed by a C3 block (composed of three standard convolution layers and several Bottleneck blocks), and the feature map for predicting the corresponding size is obtained, which is the green rectangular block in Figure 7.

5. Experiments

5.1. Hardware Platform. The experimental platform in this experiment adopts Ubuntu18.04 operating system, CPU i9-10900K@3.7Ghz, GPU NVIDIA RTX3090, 64GB memory,

(a) Mesh cracks (b) Potholes

(c) Patches

Figure 8: Pavement defects.

and CUDA 11.6 and CUDNN 8.1.5 are installed to accelerate GPU computing.

5.2. Datasets. In the identification and processing of pavement surface defects, a large amount of data needs to be used to train the model. In this experiment, the dataset was a public dataset, and a total of 2235 pavement images were selected as the dataset [23]. Most of the pavement images with defects in the dataset needed to be marked, and a small part of the images were flawless and did not need to be marked, which was beneficial to the robustness of the data. The defects in the dataset were mainly divided into three categories: mesh cracks, potholes, and patches. The classification diagram is shown in Figure 8.

459 images from 2235 pavement defect maps were selected as the test set of the experiment, and 176 images from the remaining 1776 images were selected as the validation set to avoid model training falling into overfitting. The remaining 1600 images were used as the training set to train the model. The marking software used in this experiment was YOLO_Mark, which performed box selection on the pavement mesh cracks, potholes, and patches in the dataset. YOLO_Mark is a labeling software of YOLO dataset, which is easy to use and simple to operate. The image annotation effect is shown in Figure 9. The annotations distribution is shown in Figure 10.

5.3. Performance Evaluation. mAP is calculated using precision and recall, and used as a criterion for evaluating network model performance. mAP is the value of the average detection accuracy across all classes and is used to evaluate the overall performance of the detection model. Calculated as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{1}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{2}$$

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i. \tag{3}$$

Among them, true examples (TP) are positive examples that are correctly predicted, false positive examples (FP) are negative examples that are wrongly predicted as positive examples, false negative examples (FN) are positive examples that are wrongly predicted to be negative examples, and N is the number of detection categories, AP is the detection accuracy of various types, and the

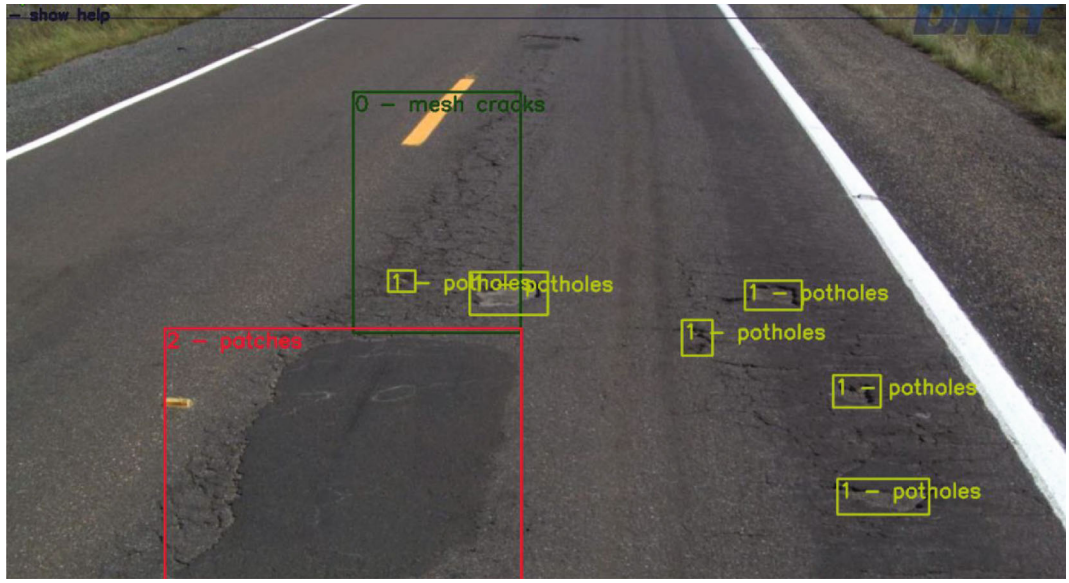
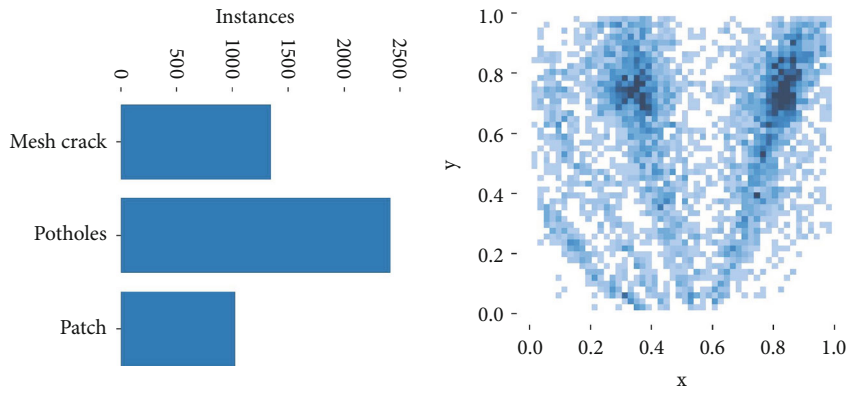
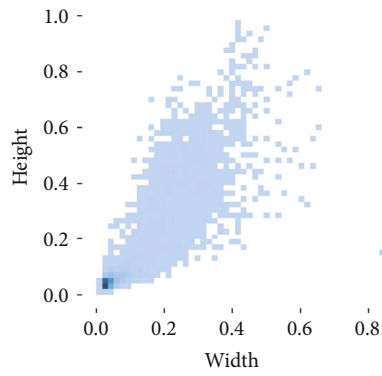


Figure 9: Image with annotations.



(a) Qty of category frames

(b) Distribution of target frame location



(c) Distribution of target frame size

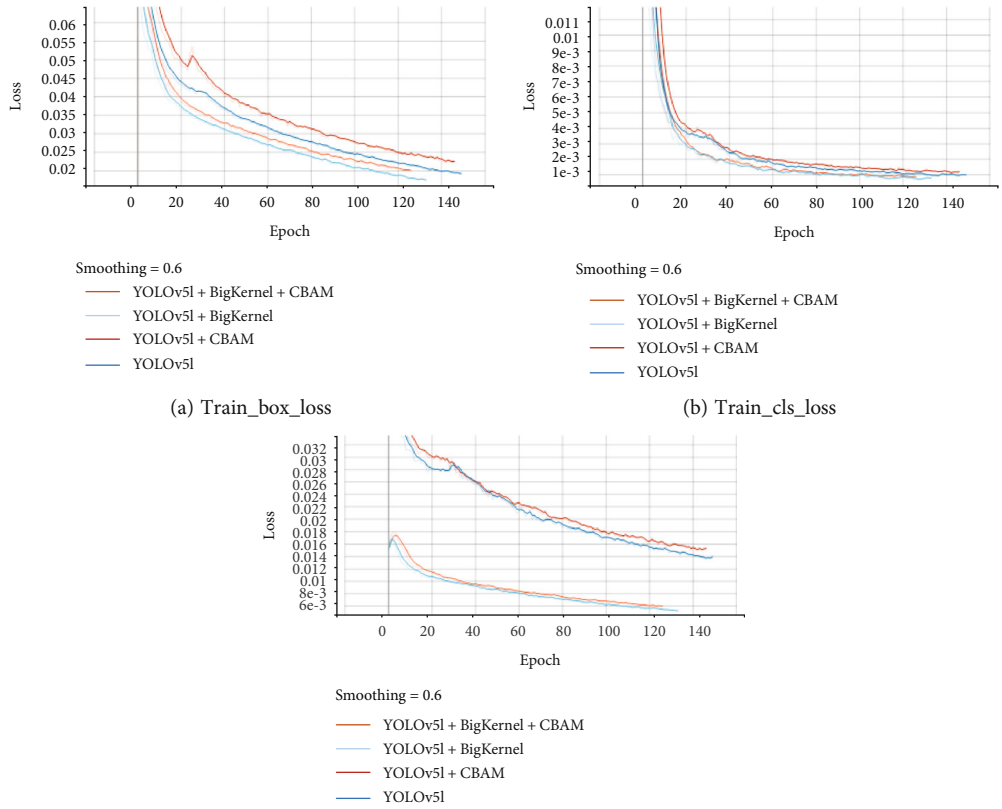
Figure 10: Annotations distribution.

calculation formula is:

$$\mathrm{AP} = \int_0^1 \mathrm{precision}(\mathrm{recall})\, d(\mathrm{recall}), \tag{4}$$

where AP can be expressed as the area of the curve made with recall as the horizontal axis and precision as the vertical axis, that is, the area of PR curve is calculated using the integral formula.

5.4. Training of the Model. This experiment chose the pretraining weight of YOLOv5 as the training weight of this



(a) Train_box_loss

(b) Train_cls_loss

(c) Train_obj_loss

Figure 11: Loss curve.

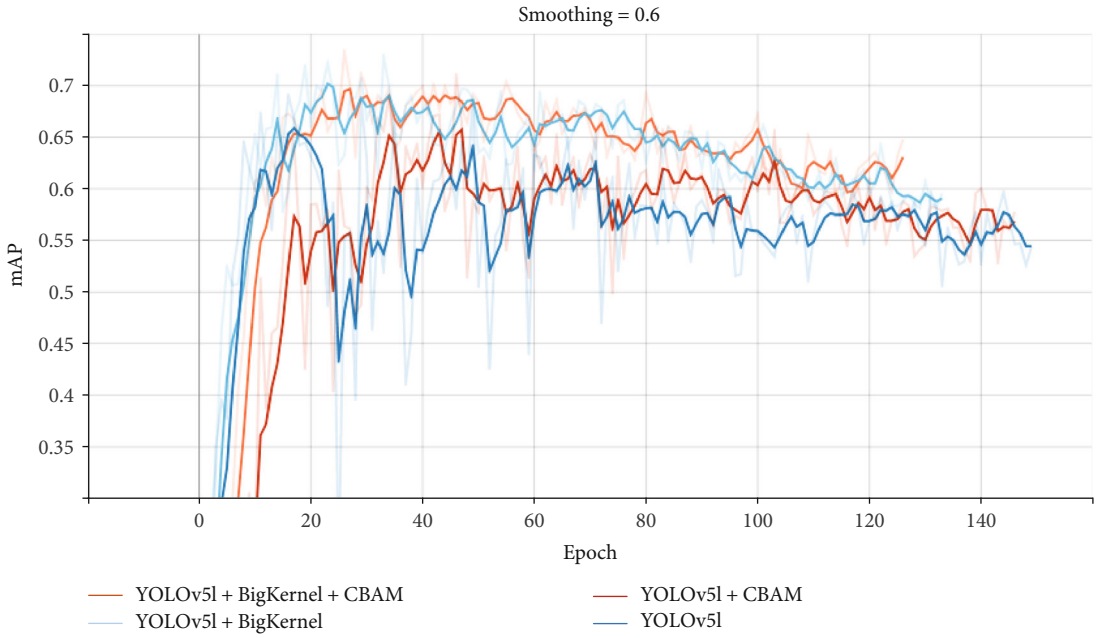


Figure 12: mAP.

experiment. The model and pretraining weights of YOLOv5 include YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. The training accuracy increases and the speed decreases. In this experiment, the actual model detection scene was considered, and the YOLOv5l model that took into account the accuracy and speed was used as the baseline comparison model of the experiment. YOLOv5l.pt was used as the experimental pretraining weight.

Table 1: Experiment results with different improvement methods.

	mAP %	MeshCracks AP%	Potholes AP %	Pacthes AP %	FPS
YOLOv5l	68.5 (0)	69.7 (0)	65.3 (0)	70.4 (0)	105
YOLOv5l + CBAM	71.2 (+2.7)	69 (-0.7)	68.9 (+3.6)	75.7 (+5.3)	102
YOLOv5l + BigKernel	72.8 (+4.3)	74.1 (+4.4)	65.1 (-0.2)	79.4 (+9.0)	48
YOLOv5l + BigKernel + CBAM	73.3 (+4.8)	72.1 (+2.4)	67.7 (+2.4)	80.1 (+9.7)	41

This experiment first used the YOLOv5l baseline model for experiments, the training parameter epoch was set to 300, the training optimizer selected SGD, and the initial learning rate was 1e-2. The GPU memory is 24GB, so the batch-size was set to 48. When the training reached 134th, the accuracy of the YOLOv5l model did not decrease significantly, and the training reached saturation, and the final best mAP was 68.5%.

The experiment was to detect potholes, cracks, and patches. The scale of the target changes greatly. Therefore, it is necessary to expand the effective receptive field of the model to better identify large-area cracks and pavement patches, so that the model can achieve higher accuracy. This experiment used a large kernel convolution block to replace the first layer of convolution in Backbone to enhance the model's ability to extract shallow semantics. Training epoch, optimizer, and learning rate are unchanged. After Big Kernel convolution is added, the memory required to optimize the model increased, and the batch-size was set to 12. At the 143rd round of training, the training of the model reaches saturation, and the best mAP is 72.8%.

In order to take into account the detection of small defects and further improve the accuracy of model detection, this experiment added CBAM to the improved YOLOv5l model. The added CBAM module enables the model to select the region of interest, making the model more sensitive to the defects to be detected, and improving the model detection accuracy under the premise of adding a small number of parameters. All hyperparameters remained unchanged. The model reached saturation at the 142nd round of training, and the best mAP was 73.3%.

5.5. Real-Time Operational Performance. Training loss function is shown in Figure 11. It can be seen from the figure that with the increase of the number of iterations, the value of the loss function of the four models gradually decreases and tends to converge. In this experiment, the SGD optimizer was used for gradient descent, and the training was stopped after 100 rounds if there was no obvious decline. The training of the four models was stopped around the 130th and 140th round.

The Figure 12 shows the mAP iteration curves of the four models with a smoothing rate of 0.6. It can be seen that the combination of YOLOv5l, Big Kernel convolution and CBAM can achieve the highest mAP. Both models with Big Kernel convolution replaced have higher mAP than the model without this module under the same number of iterations.

Table 2: Experiment results with YOLO versions.

	mAP %	FPS
YOLOv3	69.7	111
YOLOv4	61.3	71.4
YOLOv5	68.5	116
YOLOv6	63.8	49

Table 3: Experiment results with YOLOv5 versions.

	mAP %	FPS
YOLOv5n	68.0	200
YOLOv5s	67.8	192
YOLOv5m	66.8	133
YOLOv5l	68.5	116
YOLOv5x	64.3	74

Table 4: Experiment results between YOLOv5-PD and advanced model.

	mAP %	FPS
YOLOv5-PD	73.3	41
Faster-RCNN	60.1	26
YOLOX	58.0	13

5.6. Ablation Experiment. Next, this paper compared and analyzed the modified parts of the model to prove whether the improvements made in this experiment are effective. The effects of each part are shown in Table 1.

The effect of Big Kernel Convolution. It can be seen from the above table that after replacing the Big Kernel convolution, the detection accuracy of mesh cracks has been significantly improved by 4.4%, and the detection accuracy of patches is also improved by 9.0%, but the accuracy of potholes is slightly reduced by 0.2%.

The effect of CBAM. It can be seen from the above table that after adding CBAM, compared with the original YOLOv5l model, the improved model has improved detection accuracy of potholes and patches, the potholes accuracy has increased by 3.6%, and the patches accuracy is improved by 5.3%. CBAM makes the model more accurate and sensitive to the positioning of small pavement defects, and improves the detection accuracy of various types of defects.

With the addition of CBAM and big kernel modules, the map of the model was increased to 73.3%. Compared with the original model, YOLOv5-PD has a lower speed, but it

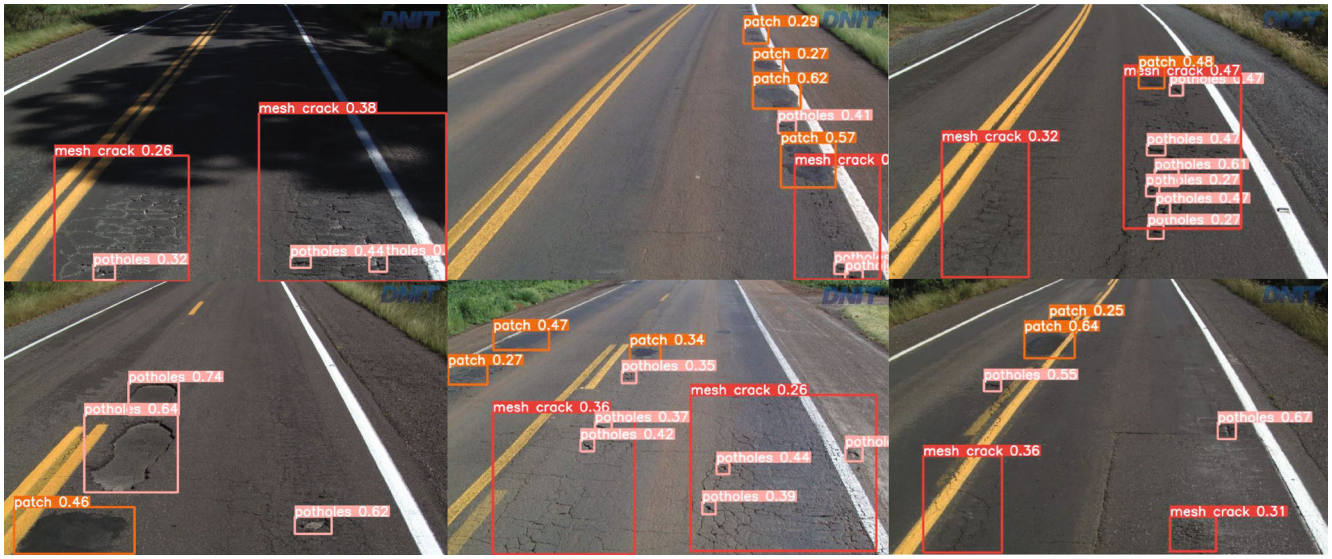


Figure 13: Effect of YOLO-PD detection.

can still reach 41FPS, obviously meeting the real-time requirements of the actual project.

5.7. Comparison Experiment. In addition, this study also made a comparison from YOLO versions, YOLOv5 versions to advanced models on the same dataset to verify the superiority of the model proposed in this experiment. The comparison results of different versions of YOLO are shown in Table 2.

YOLO series has developed to v6 version. In the development process, YOLO has been learning from the most advanced training skills and network modules. After comparison, YOLOv5 is determined as the experimental benchmark model in this paper.

According to the size of the parameter quantity, YOLOv5 has several minor versions. From *n* to *x*, the parameter quantity increases in turn and the speed decreases. This paper compares different versions and decides to use YOLOv5l model. The comparison results of different YOLOv5 are shown in Table 3.

The paper compared the proposed model with the most advanced model. It can be seen from Table 4 that YOLOv5-PD has superiority in precision and speed. The comparison between YOLOv5-PD and advanced model is shown in Table 4.

6. Conclusion and Future Work

This research proposed targeted improvements to the original YOLO model with reference to the defects in the actual road detection scene. And this paper compared it with common detection models, and the results have reached a balance in detection accuracy and speed, which confirms its advantages.

In the real pavement defect detection task, large-scale defects and small-scale defects coexist. Considering this situation, this paper proposed YOLOv5-PD model based on YOLOv5 and aiming at asphalt pavement defect scene. YOLOv5-PD greatly improves the detection accuracy of large-scale defects such as mesh cracks and patches, while small-scale defects such as potholes are also considered. The mAP of YOLOv5-PD reached 73.3%, and the detection speed reached 41FPS. The actual detection effect is shown in Figure 13. Thus, the model proposed in this study can meet the needs of complex pavement defect detection in the real situation.

In the process of improving the model, there are also some problems. Big Kernel convolution enhances the detection accuracy of large scale defects and reduces the accuracy of small scale defects, which seems to be unable to be satisfied at the same time. This research will continue to find improved methods to improve the accuracy of both in the future work, and add more kinds of road defects to improve the deployability of the model.

Data Availability

Data available on request from the authors. The data that support the findings of this study are available from the corresponding author, [Li Wang], upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (Grant No. 61973178, No. 62103205), the Smart Grid Joint Fund of State Key Program of National Natural Science Foundation of China (Grant No. U2066203), the Major natural science projects of colleges and universities in Jiangsu Province (Grant No. 21KJA470006), and the Postgraduate Research & Practice Innovation Program of Jiangsu Province (Grant No. KYCX22_3347).

Supplementary Materials

Datasets: https://biankatpas.github.io/Cracks-and-Potholes-in-Road-Images-Dataset/ *(Supplementary Materials)*

References

[1] C. Koch, K. Georgieva, V. Kasireddy, B. Akinci, and P. Fieguth, "A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure," *Advanced Engineering Informatics*, vol. 29, no. 2, pp. 196–210, 2015.

[2] R. H. Jhaveri, A. Revathi, K. Ramana, R. Raut, and R. K. Dhanaraj, "A review on machine learning strategies for real-world engineering applications," *Mobile Information Systems*, vol. 2022, Article ID 1833507, 2022.

[3] G. T. Reddy, M. P. Reddy, K. Lakshmanna et al., "Analysis of dimensionality reduction techniques on big data," *Access*, vol. 8, pp. 54776–54788, 2020.

[4] R. Sagar, R. Jhaveri, and C. Borrego, "Applications in security and evasions in machine learning: a survey," *Electronics*, vol. 9, no. 1, p. 97, 2020.

[5] K. Lakshmanna, R. Kaluri, N. Gundluru et al., "A review on deep learning techniques for IoT data," *Electronics*, vol. 11, no. 10, p. 1604, 2022.

[6] T. R. Gadekallu, M. Alazab, R. Kaluri, P. K. Maddikunta, S. Bhattacharya, and K. Lakshmanna, "Hand gesture classification using a novel CNN-crow search algorithm," *Complex & Intelligent Systems*, vol. 7, no. 4, pp. 1855–1868, 2021.

[7] R. Kaluri, D. S. Rajput, Q. Xin et al., "Roughsets-based approach for predicting battery life in IoT," no. 2, 2021https://arxiv.org/abs/2102.06026.

[8] F. Hui, X. Guo-sheng, and Y. Guo, "Multi-scale classification network for road crack detection," *IET Intelligent Transport Systems*, vol. 13, no. 2, pp. 398–405, 2019.

[9] S. Park, S. Bang, H. Kim, and H. Kim, "Patch-based crack detection in black box images using convolutional neural networks," *Journal of Computing in Civil Engineering*, vol. 33, no. 3, article 04019017, 2019.

[10] J. Huyan, W. Li, S. Tighe, J. Zhai, Z. Xu, and Y. Chen, "Detection of sealed and unsealed cracks with complex backgrounds using deep convolutional neural network," *Automation in Construction*, vol. 1, no. 107, article 102946, 2019.

[11] Z. Qu, J. Mei, L. Liu, and D. Y. Zhou, "Crack detection of concrete pavement with cross-entropy loss function and improved VGG16 network model," *Access*, vol. 8, pp. 54564–54573, 2020.

[12] Y. Du, N. Pan, Z. Xu, F. Deng, Y. Shen, and H. Kang, "Pavement distress detection and classification based on YOLO network," *International Journal of Pavement Engineering*, vol. 22, no. 13, pp. 1659–1672, 2021.

[13] S. Li, X. Gu, X. Xu et al., "Detection of concealed cracks from ground penetrating radar images based on deep learning algorithm," *Construction and Building Materials*, vol. 1, no. 273, article 121949, 2021.

[14] A. Shrestha and A. Mahmood, "Review of deep learning algorithms and architectures," *Access*, vol. 7, pp. 53040–53065, 2019.

[15] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, Santiago, Chile, December 2015.

[16] J. Redmon and A. Farhadi, "Yolov 3: an incremental improvement," 2018, https://arxiv.org/abs/1804.02767.

[17] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, July 2017.

[18] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8759–8768, Salt Lake City, UT, USA, June 2018.

[19] Z. Zheng, P. Wang, W. Liu, J. Li, R. Ye, and D. Ren, "Distance-IoU loss: faster and better learning for bounding box regression," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34no. 7, pp. 12993–13000, New York, USA, Feb 2020.

[20] X. Ding, X. Zhang, J. Han, and G. Ding, "Scaling up your kernels to 31x31: revisiting large kernel design in cnns," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11963–11975, New Orleans, LA, USA, June 2022.

[21] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[22] S. Woo, J. Park, J. Y. Lee, and I. S. Kweon, "CBAM: convolutional block attention module," *Proceedings of the European conference on computer vision (ECCV)*, vol. 11211, 2018, pp. 3–19, Munich, Germany, Sep 2018.

[23] B. K. Passos, M. Cassaniga, A. M. Fernandes, K. B. Medeiros, and E. Comunello, "Cracks and potholes in road images," *Mendeley Data*, 2020.