

Article

Research on Road Damage Detection in Multiple Countries Based on CAYOLO

Zhen Liu ¹, Fei-Fan Lao ¹, Sheng-Bing Che ^{1,*}, Shuo Zhang ¹, Sha Zhou ¹ and Yang-Zhuo Tuo ¹

¹ College of Computer and Information Engineering, Central South University of Forestry and Technology, ChangSha 410004, China

* Correspondence: t20050514@csuft.edu.cn

Abstract: Road damage detection is an international research focus, but the accuracy of damage detection has been low due to the complexity and diversity of damage features. This paper propose a CAYOLO deep learning network model based on the CAC3 and DBC3 structures in this paper. DBC3 captures the information of slight damages, while CAC3 enables the network to focus on the features of the damage in the channel and direction. The multi-scale feature pyramid structure shortens the in-formation transmission path and makes it easier to transmit information from the bottom to the top, boosting the model's ability to extract targets at different scales. Aiming at public datasets of road damages with features of India, the Czech Republic, and Japan, the research team collected 1981 image data of Chinese road damages in diverse parts of Hunan Province, China. It detected four typical damages: D00, D10, D20, and D40. It is discovered that the CAYOLO network model suggested in this paper can enhance overall detection results in the datasets of various nations than all known network models of the same type. The average detection time for a single image is 10.2ms, the D10 AP detects Japanese images at 45.56%, and the mAP detects Chinese images at 94.9%.

Keywords: Road damage; CAYOLO; CAC3; DBC3; deep learning

1. Introduction

Roads play an indispensable role in every country as infrastructure that connects various parts of people's daily lives, such as schools, companies, and shopping malls. They bring great convenience to our lives and are an essential part of modern transportation systems. However, due to aging, lack of regular maintenance, and some natural causes, pavements can deteriorate and wear out to varying degrees. Road maintenance departments worldwide are faced with an increasing number of maintenance tasks. More serious, the pothole type of damage is a potential threat to the safety of people driving on the road, increasing the incidence of road accidents and may even cause severe casualties. According to the data made available by the Ministry of Transport of the People's Republic of China, although the cost of road maintenance in 2020 is 8.45 billion yuan less than in 2019, the total amount still reaches as much as 149.95 billion yuan[1]. The country spends so much on road maintenance every year that it is vital to research a model that can be generalized and can accurately detect road damage.

Many studies and investigations have been conducted into road damage in several countries[2]. Among the road damage detection methods, the most commonly used methods are manual assessment[3], semi-automatic assessment[4], and fully automatic assessment[5]. The first approach, which primarily relies on the road assessor stepping on the road or watching the road surface from a slow-moving car, is vulnerable to traffic jams

Received:

Revised:

Accepted:

Published:

Citation: Lastname, F.; Lastname, F.; Lastname, F. Road Damage Detection. *Sensors* **2025**, *1*, 0. <https://doi.org/>

Copyright: © 2025 by the authors. Submitted to *Sensors* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

and threats to the assessor's safety. The second is a fully automatic distress assessment method with high detection accuracy. However, it needs to be equipped with expensive infrared sensor equipment, and the detection object is only cracked. The third is semi-automatic assessment, where the image collection equipment is attached to the vehicle and collects data while the car is in motion, while the data processing process is handled indoors, but it also takes much time in the data processing. Since fully automated inspection vehicles are exceedingly expensive to purchase and operate, reaching \$30–50 per kilometer[6], few national highway agencies have such fully automated inspection vehicles. However, many countries and municipalities lack the relevant technology and sufficient funds to purchase expensive devices. Roadway inspection using smartphones is becoming cheaper to execute as mobile devices like smartphones progress, many of which feature high-resolution digital cameras and processors. Puspita et al.[7] proposed to use smartphones in vehicles to obtain road images and detect road damage. In this context, developing a general road assessment model is crucial.

In general, the following three points can be made out of this question's contributions:

- (1) The DBC3 structure was designed to extract the details of the minor cracks based on the disease's characteristics.
- (2) To maintain the feature's information in the channel and direction, two structures have been used: the feature pyramid structure and the attention CAC3 structure.
- (3) Experiments on multiple algorithms and datasets show that our proposed method can get better results.

The structure of this paper is as follows: The second section discusses some work related to this study, the third section introduces the dataset used and the evaluation method of the experimental results, the fourth section discusses the design of the model, and the fifth section explains the experimental results and analysis, and the conclusions of the sixth section.

2. Related Work

2.1. Study of Damages Based on Images Taken on the Vertical Road

Most of the algorithmic models were designed based on machine learning. Aravind et al.[8] developed an advanced machine learning-based framework for concrete crack detection and failure pattern recognition, employing six classifiers to analyze experimental images of concrete beams. Their approach achieved 100% accuracy in failure pattern identification using support vector classifiers, demonstrating superior performance in classifying flexural, shear, and compression failures. Sun et al.[9] proposed an optimized crack detection method utilizing histogram of oriented gradient (HOG) features with support vector machine classification, demonstrating superior accuracy and computational efficiency compared to conventional methods through Bayesian-optimized hyperparameter selection. De León et al.[10] developed a novel crack segmentation algorithm combining minimal path selection theory with Gabor filter-based texture analysis, achieving a competitive F1-score of 0.839 while maintaining full workflow transparency and operational flexibility across various infrastructure monitoring scenarios.

With the proposal of deep learning neural networks, many road damages based on neural networks have emerged. Zhang et al. first used deep convolutional neural networks to bridge the gap with traffic studies, used ConvNet to compare with SVM and Boosting methods, and experimentally demonstrated that the ConvNet-based method can correctly identify the presence of cracks in pictures. Samma et al.[11] developed an optimized VGG-19 model using a novel two-layer optimizer for drone-based road damage detection, achieving 96.4% F1-score accuracy while reducing the number of convolutional filters by 52% compared to the original architecture, outperforming conventional

optimization methods including PSO and reinforcement learning-based approaches. Luo et al.[12] proposed E-EfficientDet, an enhanced lightweight network featuring a novel feature extraction enhancement module (FEEM) and semi-dense connected pyramid structures, achieving 2.41% higher accuracy than EfficientDet-D0 while maintaining real-time performance (27.0 FPS) and compact model size (32.31 MB), suitable for deployment on mobile inspection platforms. He and Chen[13] introduced a new neural network for road crack detection, AttentionCrackNet, which uses an encoder-decoder structure and inserts an attention mechanism named the attention gate, which enables the network to focus more on crack features.

However, the two critical features of weak semantic information and geometric feature anomalies are not considered in most CNN-based road crack detection studies. Xu, Chen, and Qin[14] accordingly proposed CrdNet, a backbone consisting of LrNet[15] with the addition of a multiscale aspect ratio mechanism, length awareness, and a cascade detection head, and demonstrated that using the CrdNet network to detect pavements can achieve better results than the Faster R-CNN algorithm[16], and mAP can reach 90.92%. Dorafshan et al.[17], who compared several common edge detectors and neural network models AlexNet DCNN[18], discovered that the deep learning-based approach outperformed various edge detection methods to detect cracks smaller than 0.04 mm. However, DCNN takes more time to acquire pre-trained models. Kulambayev et al.[19] introduced an optimized Mask R-CNN framework for real-time road surface damage detection, demonstrating superior accuracy and computational efficiency in detecting various road aberrations including cracks, potholes, and rutting across diverse environmental conditions. Hacefendiolu and Baaa[20] developed a comprehensive crack detection system using Faster R-CNN, achieving robust performance across varying weather conditions and illumination levels, with detailed analysis of detection rates under different environmental scenarios including sunny, cloudy, and foggy conditions, as well as sunset and moonlight periods.

Although studies using images taken vertically to the road surface can produce good detection results, studies using such images have several limitations: the expense of regularly collecting new images using specialized vehicles is too high; the collected images do not cover the entire road surface, and the constructed models of this type are better suited for dealing with classification problems. Due to these problems, disease studies based on images taken from dashboards have emerged.

2.2. Study of Damages Based on Images Taken on the Dashboard

The second kind of detection study is the semi-automatic assessment approach discussed in the first section, which uses a smartphone fixed to the dashboard or windshield to take pictures and then performs image processing and detection model building in the room. In addition to the road, the photos obtained using this method also include distracting things like trees, buildings, and automobiles, which makes it challenging to detect pavement illness. Despite these limitations, this study is made possible because it only needs a smartphone to gather the photographs, and the research data may be easily updated.

In 2018, public datasets of these photographs were made available, and most of them were analyzed using deep learning techniques. Road damage types were divided into eight categories by Maeda et al.[21] in 2018, trained on the eight categories using SSD Inception V2[23, 24][22,23] and SSD MobileNet[24], resulting in recall and precision rates of 71% and 77%. The dataset, RDD-2018, was collected using cell phones placed on windshields. Wang et al.[25] conducted experiments using SSD and Faster R-CNN based on the dataset RDD-2018 and performed data augmentation operations with brightness ad-

justment and Gaussian blurring, and the F1-score reached 0.6255. Wang et al.[26] fused 14 base models using pre-trained ResNet[27] and VGG[28] as the backbone to handle damage detection and classification tasks to enhance the performance of the models. Chen et al.[29] trained Mask R-CNN[30] with DenseNet[31] as the backbone, used a feature pyramid network combined with multi-scale features for detection, and used Labelme to polygon label the identified diseases again based on RDD-2018. To improve the F1 score, Maeda et al.[32] first cropped the damaged part of the D40 category into pixel blocks, then generated more pothole-type diseases using PG-GAN[33], and finally combined the generated images with non-damage images using Poisson fusion. When there were fewer original images, the F1 score increased by 5%, and the F1 score increased by 2% when the number of training images was increased. Additionally, the authors proposed RDD-2019 based on RDD-2018, which meant that the annotations of the RDD-2018 dataset were updated and expanded in number. To increase the precision of local bounding boxes in the dataset, Yin et al.[34] suggested a multilayer feature pyramid method based on M2det[35] with an attention mechanism.

Arya et al.[36] created RDD-2020 by combining road pictures from the Czech Republic and India with the Japanese dataset to identify cracks and potholes. This dataset was based on RDD-2019. Additionally, they examined how changing the number of training photographs affected the model's performance. The trials revealed that using more training images improved the model's performance. Chen et al.[37] developed an optimized YOLOv5-based road damage detection system using the RDD-2020 dataset, implementing a comprehensive data preprocessing pipeline to address labeling inconsistencies and achieving efficient real-time detection capabilities for both damage localization and classification. Hasan et al.[38] conducted extensive experiments with five pre-trained models on the RDD-2020 dataset, proposing a fine-tuned Faster RCNN with ResNet-101 architecture that achieved improved F1-scores of 0.47, 0.41, 0.41, and 0.35 for longitudinal cracks, transverse cracks, alligator cracks, and potholes respectively, outperforming previous approaches in detecting overlapping damage regions. Wu et al.[39] proposed Flexi-Weather Hard Detection, an innovative algorithm integrating AIGC-based data augmentation and corner point feature aggregation, achieving 64.9% in Test1 metric and 40.6% F1-Score on RDD-2020 and CNRDD datasets, with enhanced robustness in adverse weather conditions through Weather Trim Augment module and improved local feature learning via Flexi Corner Block.

Shim et al.[40] designed a disease segmentation block with four dense blocks containing encoders and decoders based on DenseNet-121 to achieve an F1-score of 79.33 with only 3.56M parameters. However, the algorithm only detects the presence of damage in the images. Lin et al.[41] performed road picture capturing from two angles, 70° and 30°. They collected 620,000 damages in Taiwan, China, and classified the damages into six categories, taking utility hole covers and repairs into consideration. They conducted experiments using YOLOv3 and achieved 71% accuracy in detecting pothole-type damages. Rehana et al.[42] applied only 2607 photos to train the network to detect seven damage categories, with a 92% overall accuracy rate. Su et al.[43] introduced the FSRDD model to remove redundant information and extract features by ghost attention and proposal feature metric, show the methods applicability through ablation experiments, and reach an mAP of 51.7% on public datasets. Chen et al.[44] proposed LAG-YOLO, a lightweight attention-based YOLO variant for road damage detection, achieving 52.35% mAP on RDD-2020 dataset with only 4.19 million parameters through the integration of an attention ghost module and SimAM mechanism, demonstrating superior efficiency and accuracy for real-time processing and lightweight deployment. Deepa and Sivasangari[45] developed a hybrid deep learning framework combining adaptive density-based fuzzy c-means

clustering with a novel Hybrid Deep Capsule Autoencoder-based Convolutional Neural Network (Hybrid DCACN), achieving 98.815% accuracy on the RDD-2020 dataset through an optimized feature extraction pipeline incorporating Laplacian edge detection and hybrid wavelet-Walsh transform. Rakshitha et al.[46] conducted comprehensive experiments with deep CNN and ResNet architectures on the RDD2022[?] dataset, demonstrating that ResNet152 achieved superior performance with an F1-score of 0.6954 through advanced data augmentation techniques including contrast transformation and Gaussian blur, outperforming ResNet50 and ResNet101 models in pavement damage classification.

Road damages are today the focus of numerous studies, and many solutions have been proposed. Nonetheless, few studies report the disease as a binary classification problem by masking and labeling it, which can only establish whether a disease exists. The different damage kinds, however, cannot be classified. Although the properties of the cracks have always been unclear, several studies are suggested for their extraction. Other network models have been explored by researchers to extract disease features. Two-stage networks have a long training period yet have good detection accuracy. Single-stage models take less time to train but are less accurate. In this study, several typically used object detection methods are compared. The network model CAYOLO put forward in this research stands out from the mass and achieves the best detection speed and accuracy. CAYOLO outperforms other evaluation parameters. Although CAYOLO can produce excellent results, it still misses detections during testing, leaving much space for performance enhancement.

3. Datasets and Methods

3.1. Datasets

The dataset used in this paper was provided by the 2020 IEEE Big Data Road Damage Detection Challenge[35, 53-55][32,36,47,48]. The images in this dataset were captured with a smartphone fitted to the dashboard or windshield of a car. The dataset mainly includes images from Japan, the Czech Republic and India, and the official dataset was divided into one training set and two test sets. The labeling file of the training set was PASCAL VOC format and included unified names for various types of damage. Due to privacy reasons, only the annotation files of the training set are officially disclosed, so this paper only uses the data provided in the training set for experimental research.



(a) Japan road



(b) Indian road



(c) Czech road



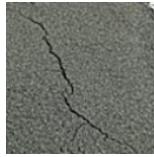
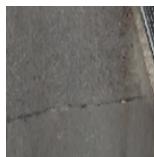
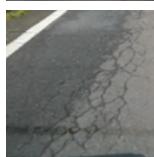
(d) China road

Figure 1. Examples of road pictures from several countries.

Since the data provided in the 2020 challenge mainly identifies and detects four categories of road damages: longitudinal cracks(D00), transverse cracks(D10), alligator cracks(D20), and potholes(D40), as shown in [Table 1](#), this paper also addresses these four categories of damage that were tested. In addition, since the surface conditions of roads vary significantly from country to country, as shown in [Figure 1](#), there are considerable differences between Indian roads and the rest of the nations. The number of Japanese datasets is the largest, with only the training set images provided by the public dataset being used for experiments in this thesis. The five categories that are not considered are dropped. There are a total of 7900 images containing damage in the end, and the training set, validation set, and test set are divided according to the ratio of 8:1:1. The number of training sets, verification sets and test sets were 6320, 790 and 790 images respectively.

215
216
217
218
219
220
221
222
223
224
225

Table 1. Damage types and illustrations.

Damage Name	Damage Type	Damage Detail Illustration
D00	Longitudinal cracks	
D10	Transverse cracks	
D20	Alligator cracks	
D40	Potholes, bumps, rutting, separation	

Meanwhile, this study collected data on partial highways and urban roads throughout Hunan Province, China, using smartphones placed on the dashboard, and the collected data size was 1920×1080 . The image size was adjusted to a uniform 600×600 after processing, consistent with the publicly available RDD-2020 dataset. When labeling the data, nine damage types were labeled according to the criteria given in the public dataset, but only four damage types were used for the experiment, and the data is presented in [Table 3](#), with a total of 1981 images and 2551 damages. The number of various damage types in each country is listed in detail in [Table 2](#), and it can be noted that Japan has the most significant total number of D20 categories and relatively few D40 types; India has a severe imbalance in the data of various kinds of damage, with only 60 D10 categories; and there is also an imbalance in the Czech Republic and China. From the given data, it is apparent that there are significant variations in each country's scenario, making it imperative to research a detection model that is widespread worldwide.

Table 2. A number of road damage data by category in a different countryt.

	D00	D10	D20	D40
Japan	3644	3577	5590	2061
India	1388	60	1827	2852
Czech	892	352	136	175
China	749	926	193	683

Even though this study only collected a limited number of photographs compared to that public dataset, the D10 category with less obvious damage characteristics received relatively more data. We carefully comply with the criteria during the labeling process and repeatedly examine the labeled dataset to avoid false detection and missed detection. As shown in [Figure 2](#), while even human eyes can recognize erroneous labeling immediately, learning models find it challenging to do so. Additionally, we strive to alter the damage's features while labeling it so that as little noise as possible enters the groundtruth bounding box.

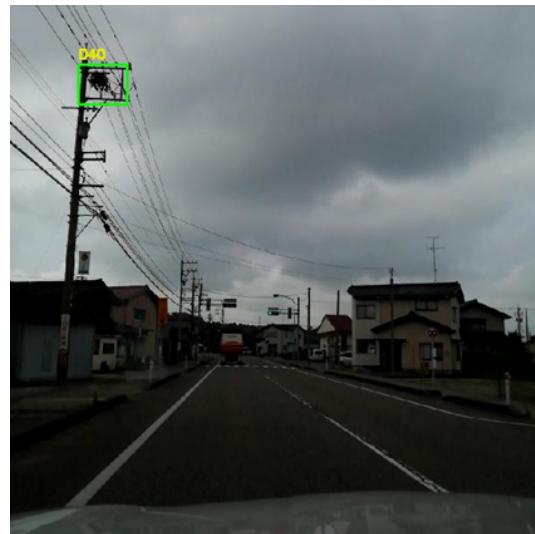


Figure 2. Error-tagging.

3.2. Methods

Four traditional road damage types are chosen as detection targets in this work. The YOLOv5 network meets the detection requirements well and is vulnerable to problems like the overfitting phenomena and poor detection performance because pavement cracks are thin and challenging to differentiate compared to many other objects, and the amount of available datasets is constrained. On this basis, this study proposed the CAYOLO model, which can effectively address these problems.

3.2.1. Design of CAC3 Module

The Coordinate Attention Concentrated-Comprehensive Convolution(CAC3) module, a lightweight attention module with Coordinate Attention(CA)[49], is embedded into the C3 structure, which divides the channel attention into two one-dimensional feature coding processes for feature aggregation in different spatial directions, allowing the network to acquire long-term information in one direction and retain location information in the other direction, followed by encoding the generated feature maps separately to generate a pair of feature maps interesting in direction and location.

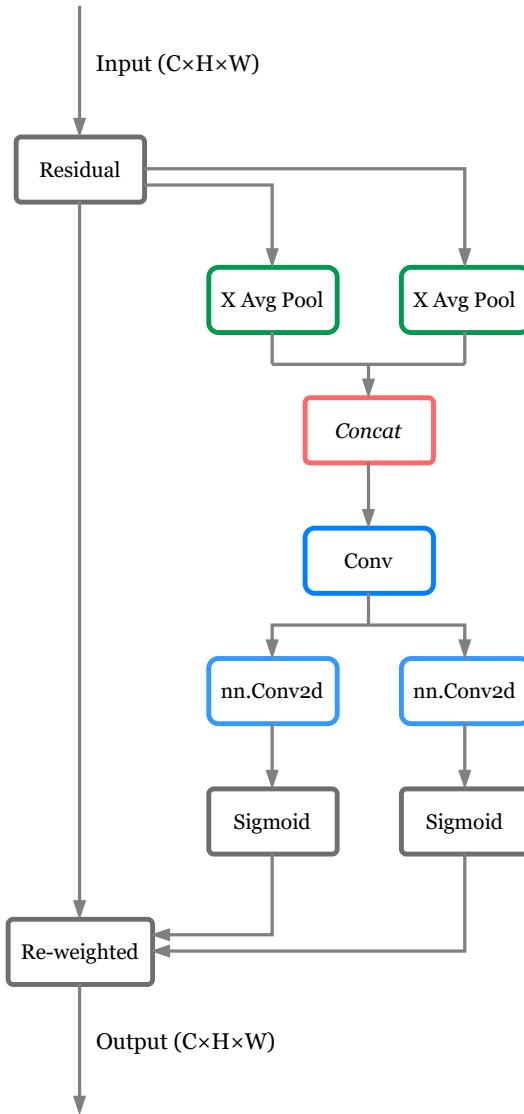


Figure 3. Error-tagging.

The CA attention mechanism is shown in Figure 3. The attention module can capture the feature information in spatial order thanks to the adaptive averaging pooling operation in the horizontal and vertical dimensions. Subsequently, the two feature maps are concatenated in the channel orientation operation, followed by encoding the spatial information of the just generated feature maps with 1×1 convolution and then tensor transforming the feature information in the horizontal and vertical directions using 1×1 convolution, respectively, to obtain feature maps that are sensitive to the location information. The computational cost and model complexity can be efficiently reduced by CA attention since it mainly utilizes 1×1 sized convolution kernels and divides the number of channels by the scale factor (the default is 32) before implementing the first convolution. SENet attention[50] operates on channels, so it focuses more on inter-channel correlation; CBAM attention[51] uses large-size convolution to obtain location information, but it focuses less attention on channel information; while CA attention is used in this experiment, it can extract not only inter-channel information but also direction and location information, effectively improving the model's ability to locate and identify damages. This paper proposes a CAC3 structure, which embeds CA attention into the Bottleneck structure of the C3 architecture, as shown in the figure. During the computation of the Bottleneck, the input information is divided into g groups for processing, and residual structures with

convolution kernels of size 1×1 and 3×3 are used for feature extraction. The residual structures are repeated n times, and the CA attention mechanism is added without changing the size and number of channels of the feature map. When the residual structure in the Bottleneck structure extracts deep-level information, the attention mechanism transmits the extracted spatial and positional information to the Bottleneck, which can effectively alleviate the problem of gradient vanishing. Additionally, the CAC3 structure pays more attention to the texture direction and position information of road damage during the backbone phase of the network, which can effectively extract the feature information of road damage. Experimental results show that this method can significantly improve the performance of the network and achieve good detection results.

3.2.2. Design of DBC3 Module

According to the slender traits of pavement cracks, this study designs the Double Bottleneck Concentrated-Comprehensive Convolution module(DBC3) in the neck region of the CAYOLO backbone network to extract the details of the damage and designs the Double Bottleneck inserted into the C3 structure with the convolution kernel size set to 1×1, 3×3, and 5×5, respectively. The small convolution kernel is more sensitive to slight cracks and potholes features during the training process. After Conv1, the feature vector obtained is inputted into the Double Convolution structure. Before that, the channel number is divided into two and inputted into two residual structures. The 1×1 and 3×3 small convolutions extract texture features of cracks and pits, while the 5×5 convolution focuses on the larger mesh crack features. The feature mapping network deepens the network's ability to extract deep features, and the Double Convolution structure does not cause network degradation due to the deep layers. The double convolution Double Bottleneck structure is shown in Figure 4.

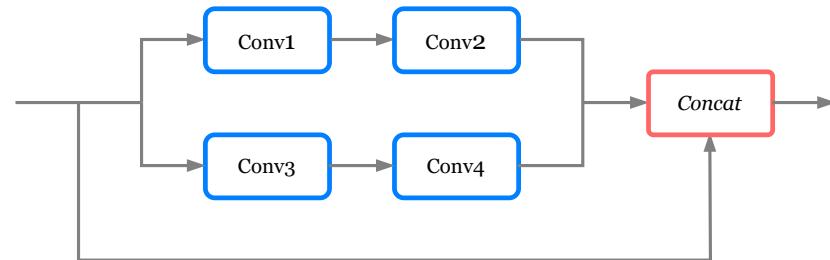


Figure 4. The Double Bottleneck module.

After embedding the Double Convolution structure into the C3 structure, in order to prevent the image size from decreasing during the convolution operation and to avoid losing pixel information at the edge of the image, automatic padding is performed during the convolution operation.

Afterward, the feature maps obtained are stacked on the channel axis. After the Double Convolution operation, some feature information may be lost and the network may suffer from model degradation due to its depth. Therefore, a shortcut operation is performed after the stacking operation to preserve the feature information of the disease. Although the addition of the Double Convolution operation may increase some computational complexity, it can improve the network's attention to small targets, so the computational complexity it brings is acceptable.

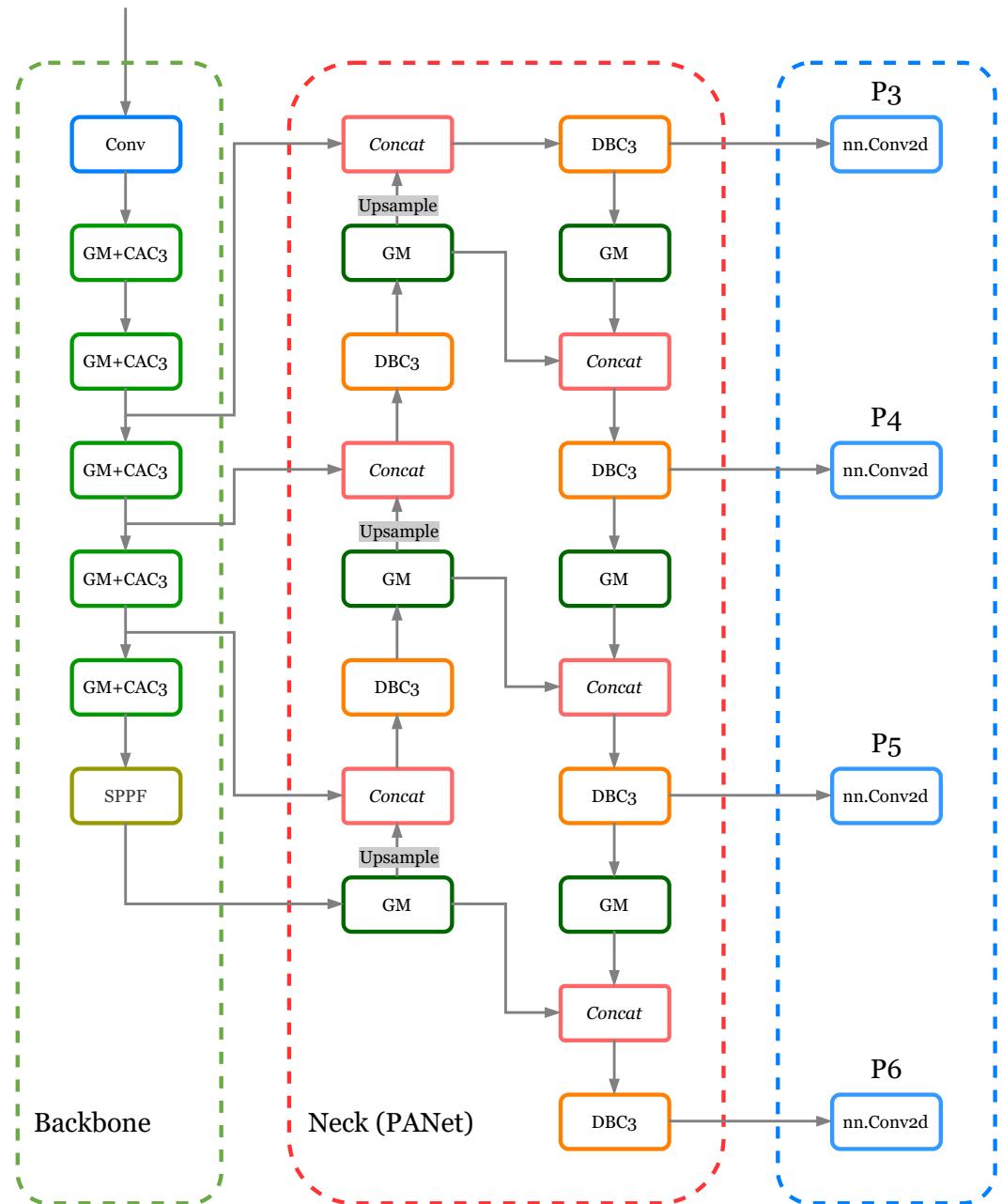


Figure 5. CAYOLO network structure.

CAYOLO's backbone structure is shown in Figure 5. The PANet[52] structure enhances the whole feature hierarchy with accurate localization signals in the lower layers through top-down path augmentation, which shortens the information path between the lower and top layer features. Shallow feature maps have small receptive fields, contain more location information, have a strong ability to capture geometric capability, and are more suitable for detecting small targets. Deep feature maps have more robust semantic information but are more suitable for detecting large targets due to their poor detail perception ability for low-resolution images.

The GM in Figure 5 stands for Ghost Module[53]. The bottom-up structure of PANet is the bottom-up structure of PANet is primarily introduced to preserve the vital shallow feature information of the network. After all, the shallow features contain much information, such as edge geometry, since instance segmentation is based on pixel-level classification, preserving more shallow features through this bottom-up process is beneficial, and more shallow features can be conserved after this bottom-up process. Considering the

315

316

317

318

319

320

321

322

323

324

325

326

327

328

comprehensive damage characteristics, the damage targets of potholes and crack classes are small. At the same time, some of the D20 types damages have a more comprehensive range of damage, so the feature layers [P3, P4, P5, P6] are used in this experiment to detect damage targets of potholes and crack classes, which are small in size, as well as some of the D20 types that have a more comprehensive range of damage.

4. Evaluation

Model performance is estimated by two metrics, average precision(AP) and mean average precision(mAP), F1-Score. Since the classification is imbalanced in the dataset and accuracy-based metrics alone may be biased, mAP is used as the primary criterion for comparing the performance of different models in this paper, and it can be used to assess the risk of incorrect classification by using a confidence score assigned to each detected bounding box.

The precision rate P represents the percentage of correctly predicted instances to the total number of predicted features, and the recall rate R represents the percent of correctly predicted features to the total number of features present in the actual class. Both precision rate and recall rate are based on IoU, which is the ratio of the area of overlap of the region between the predicted border and the actual edge to the area of its conjunctive region, and the IoU threshold is set to 0.5 in this paper. Where the precision and recall rates are computed as follows:

$$P = \frac{TP}{TP + FP} \quad (1)$$

$$R = \frac{TP}{TP + FN} \quad (2)$$

Among these, TP means the number of damages that were accurately identified (true positives), and FP denotes the number of damages that were mistakenly identified as damages (false positives). FN is the count of damages identified as non-damages (false negatives). The formulas for AP, mAP and F1-Score are as follows:

$$AP = \frac{1}{m} \sum_{r \in \{0.0, 0.1, \dots, 1.0\}} P_{interp}(r) \quad (3)$$

$$mAP = \frac{1}{N} \sum_{i=1:n}^N AP^i \quad (4)$$

$$F1 = 2 \times \frac{P \times R}{P + R} \quad (5)$$

Among them, $P_{interp}(r)$ is the maximum accuracy rate by a certain condition. AP^i is the average precision value of the i-th. The higher the AP value, the better the model performance. m is made up of 11 equally spaced numbers in the range $[0.0, 0.1, \dots, 1.0]$, where n is the total number of categories.

5. Experimental Results

5.1. Performance Comparison of Different Optimizer

This experiment was trained for 500 epochs, and [Figure 6](#) depicts the curve of the change of the mAP value of the model during the training process. From the figure, we can see that about the 200th epoch, the network has stabilized, and the value of CAYOLO proposed in this paper is marginally higher than that of the original YOLOv5. In about

400 epochs, YOLOv5 begins to exhibit the overfitting phenomenon, and the value of mAP has a significant downward trend.

Additionally, the network's effects of the SGD and Adam optimizers were investigated. As seen in Figure 7, although using the SGD optimizer can bring the network to equilibrium faster, the network achieves more stable performance with less fluctuation in detection values when using the Adam optimizer, and the network is more prone to overfitting when using the SGD optimizer. This situation is likely caused by the network training for too long, the parameters updating too frequently, and the existence of noise or outliers in the training data, which leads to the occurrence of overfitting. As a result, this study uses the Adam optimizer based CAYOLO to conduct experimental work.

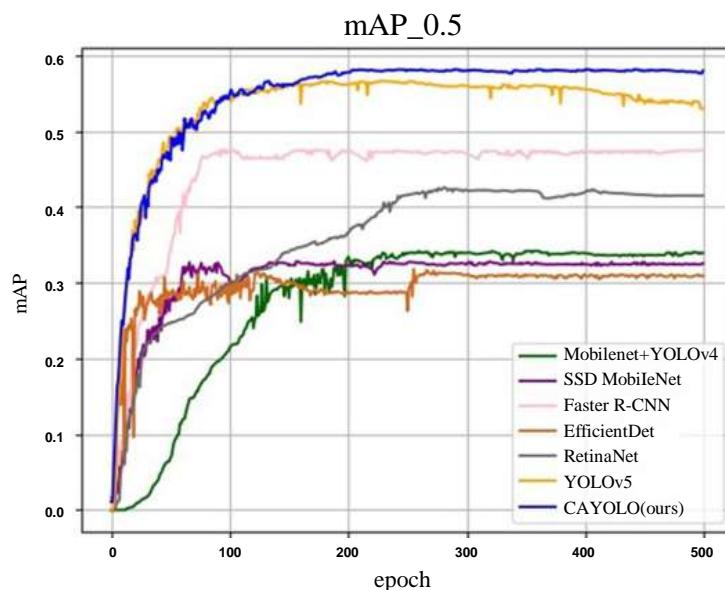


Figure 6. Comparison of the mAP training process of the different algorithms.

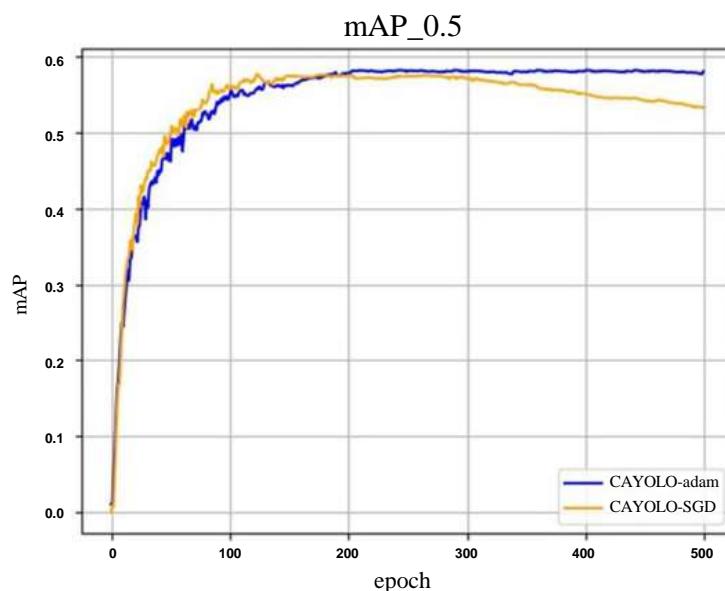
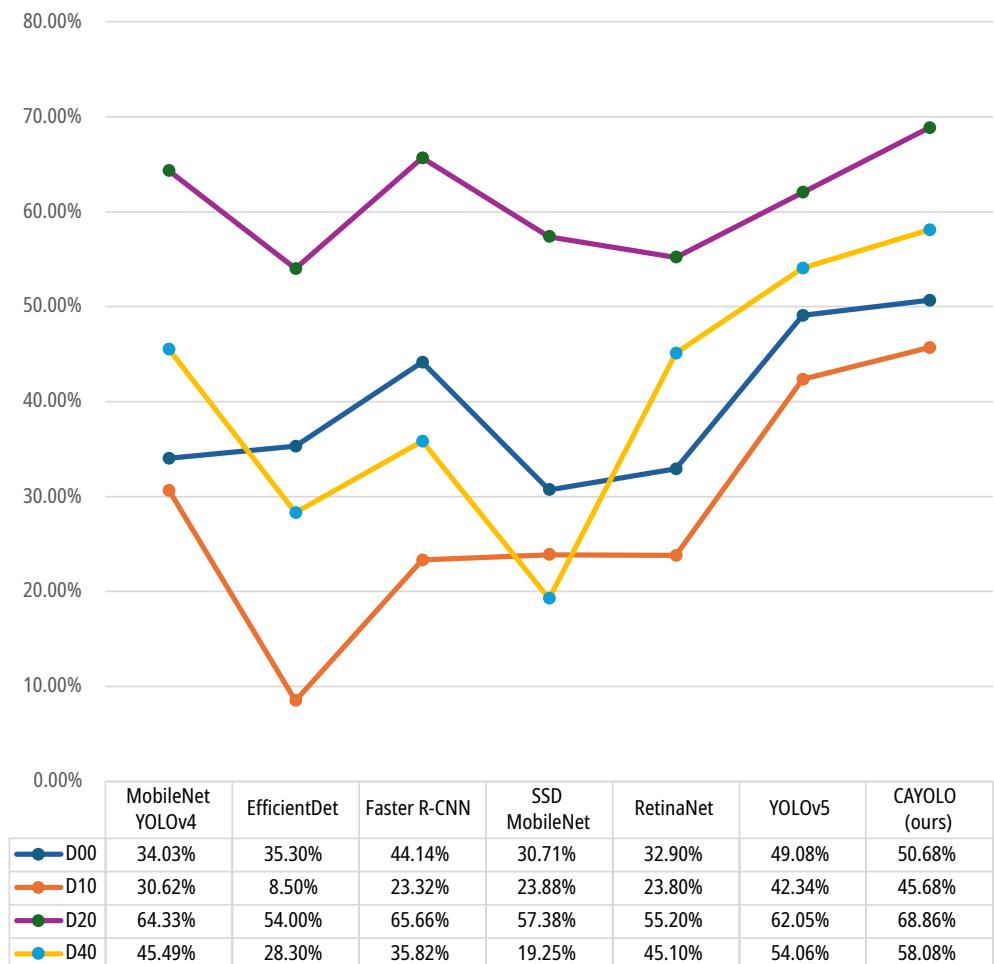


Figure 7. mAP changes during training with different optimizers.

5.2. Comparison of Results of Different Detection Algorithms

To demonstrate the advantages of the model proposed in this paper, the model CAYOLO in this paper is experimentally compared with Faster R-CNN, RetinaNet, EfficientDet, MobileNet+YOLOv4, SSD MobileNet, and YOLOv5. [Figure 8](#) shows the test set's average precision results for detection using various algorithms.

Analyzing the information in [Figure 8](#) demonstrates how the suggested methodology in this research can produce better outcomes for the detection of each category. The two-stage networks, Faster R-CNN and MobileNet+YOLOv4, showed comparable performance to the YOLOv5 network for detecting the D20 category. Neither EfficientDet nor SSD MobileNet showed good performance in detecting each category, the mAP of EfficientDet for the D10 was only 8.5%. RetinaNet shows that D40 performs dramatically better than SSD MobileNet, while other categories also perform as well as SSD MobileNet. CAYOLO improved the outcomes of YOLOv5 for detecting the D20 and D40 classes. The graphs show that each network's detection performance for the D10 category is less than ideal. This phenomenon is likely due to several factors, including the D10 category's limited data supply, the lack of evident fracture features in the images, and the network's difficulty in learning the characteristics of lateral cracks.



[Figure 8](#). AP values for different networks per category.

[Figure 9](#) shows the evaluation of the test set using several algorithms to examine the performance of the training model. The major problem with MobileNet+YOLOv4 is that it missed all D40 and D10 class damages, as demonstrated in [Figure 9\(a\)](#) and [Figure 9\(c\)](#). In [Figure 9\(a\)](#), it shows that EfficientDet incorrectly classifies the object on the

side wall as class D00, and Figure 9(b) misidentifies D01 as class D00 damage, which is the part of the longitudinal joint between the concrete during the construction of the road. This misidentification phenomenon also exists in Faster R-CNN and YOLOv5, and in Figure 9(c) also misses multiple instance of class D10 damages and misidentifies the gutter cover as D20 damage. Faster R-CNN has the highest detection accuracy; D40 is missed in Figure 9(a), while D01 and D11 are wrongly detected in Figure 9(b), identified as D00 and D10 in the figure. Class D11 is the part of transverse joints between concrete during road construction, respectively. Only D20 was detected by using SSD MobileNet, and all the other three classes were missed. There are also problems with false and missing tests and the RetinaNet. YOLOv5 did not detect D10 and D40, and no instances of D10 were identified. CAYOLO missed detecting instances of class D40 in both instances Figure 9(a) and Figure 9(c).

According to Figure 6 and Figure 8, it can be seen that CAYOLO proposed in this study has significant advantages in detecting various types of damages, and it can reach an average precision of 45.56% for the D10 category, which is less numerous. EfficientDet is a well-known lightweight network. Some information from the feature layers will be reused during the BiFPN process. However, doing so will increase the computational load. Therefore, some information is dropped randomly during this process via dropout, but some features are not apparent. The network can extract fewer features after the dropout operation, and since there are fewer significant features, the model cannot effectively learn the features of D10 during training. The data for D10 and D40 are sparse, as shown in Table 2 and Figure 8. D40 has more distinct characteristics in comparison. Hence its mAP is marginally greater than D10.

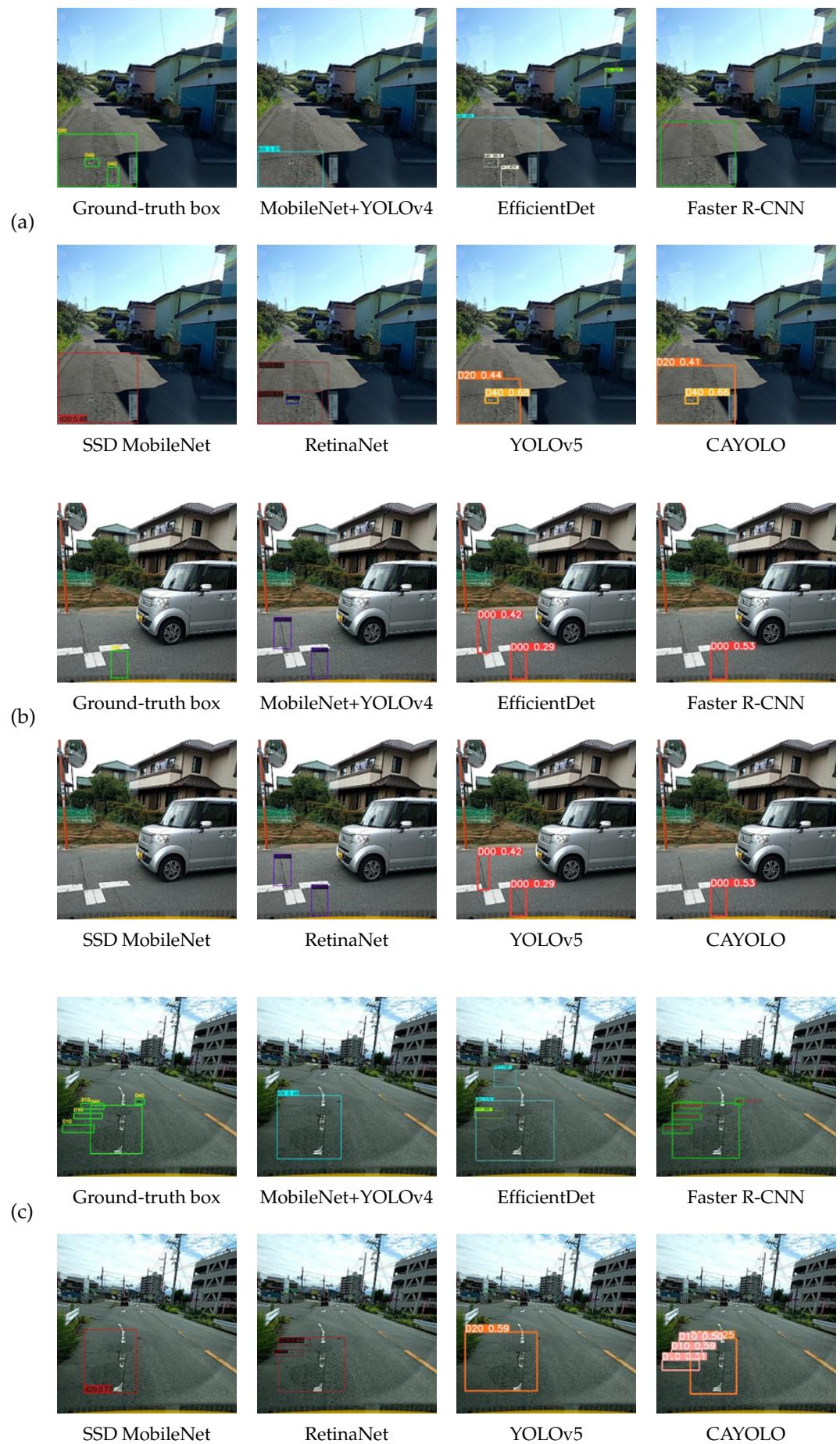


Figure 9. Comparison of different network test results.

The mAP values of each network's training process are shown in [Figure 6](#). According to [Figure 6](#), the CAYOLO network suggested in this article has considerably improved the detection performance of various networks. The CAYOLO network provided in this paper is more stable and effective than several other networks.

Table 3. Individual training model parameters and test single image time.

Model	Paramerters	Avg time per image
MobileNet+YOLOv4	51.1MB	19.6ms
EfficientDet	15.0MB	34.7ms
Faster R-CNN	360MB	92.4ms
SSD MobileNet	23.8MB	44.3ms
RetinaNet	277MB	12.95ms
YOLOv5	14.2MB	11.1ms

[Table 3](#) shows the model size and time parameters of the test dataset obtained by training with each network, and the test image size is set to 640×640. The table shows that the model obtained by training with YOLOv5 is the smallest, while the model obtained by training with Faster R-CNN with resnet101 as the backbone is the largest, with a single model reaching 360MB. The time used to test a single image in Compared with YOLOv5, the average detection time of CAYOLO for a single image is improved by 8.82%, sacrificing a petite model size. They are bringing an increase in model testing speed.

[Table 4](#) shows that although the detection speed is slower, the F1-Score in [44] is similar to the CAYOLO result. In [46], the maximum value is obtained when the IoU is 0.75, but this method takes a long time. The highest F1 value of 0.5814 was attained in [41] using the pre-training model and extensive training sets. [Table 4](#) demonstrates that when the IoU is 0.75, the mAP value of each network decrease by at least half. As a result, when the IoU is 0.5, this study performs model training and evaluation. This study proposed in this work can achieve good performance and the quickest detection speed among the current methods.

Table 4. Comparison of the results of CAYOLO and different papers'.

Methods	mAP@.5	mAP@.75	F1-score	Avg test time per image
Binyi Su[52]	0.517	-	-	60.9ms
Minh-Tu Cao[45]	0.547	0.249	0.5145	17s
Naddaf-Sh[43]	0.542	0.152	0.566	17.8ms
Mandal[40]	-	-	0.5814/0.5751	15.3ms
YOLOv5	0.519	0.227	0.545	11.1ms
CAYOLO	0.558	0.241	0.567	10.2ms

5.3. Road Damage Detection Results in Different Countries

This section conducts experiments using the models trained by YOLOv5 and CAYOLO, respectively, in the previous sections to assess whether the proposed model approach can adapt to the road conditions in various nations. The datasets used are the training images of the Czech Republic and India provided by RDD-2020 and the 1981 images collected by this research team from the Hunan Province of China. These data are randomly divided in the ratio of 8:1:1 to assess the model's effectiveness for detecting roads in various nations after the five more irrelevant categories have been discarded, similarly.

[Figure 10](#) and [Figure 11](#) display the fluctuation in mAP values for each country during the training process and the PR plots for the detection of the test set, respectively.

Figure 10(a) shows that using CAYOLO is more stable and better in the training process; Figure 10(b) illustrates that using YOLOv5 to train the Czech road can stabilize earlier compared to CAYOLO, but using CAYOLO can get better results; Figure 10(c) proves that using YOLOv5 can stabilize earlier, and after training up to 400 epochs, the two can be used interchangeably. The two networks can produce virtually the same outputs after 400 training epochs. Figure 11(a) exemplifies that when testing the Indian road images, CAYOLO can achieve better performance than YOLOv5 in detecting D10 class damages because the total number of D10 class damages is too minimal, the road conditions are more complicated, and the network has difficulty learning the relevant features of D10 class damages; Figure 11(b) shows that when testing the Czech road images, CAYOLO can achieve better performance in detecting D10 and D40 damages; and YOLOv5 can achieve better performance in detecting D00 and D20 categories of damages; Figure 11(c) demonstrates that when examining Chinese road photos, the two network models can almost always perform equally well. In general, CAYOLO, used in this study, can produce better road detection outcomes.

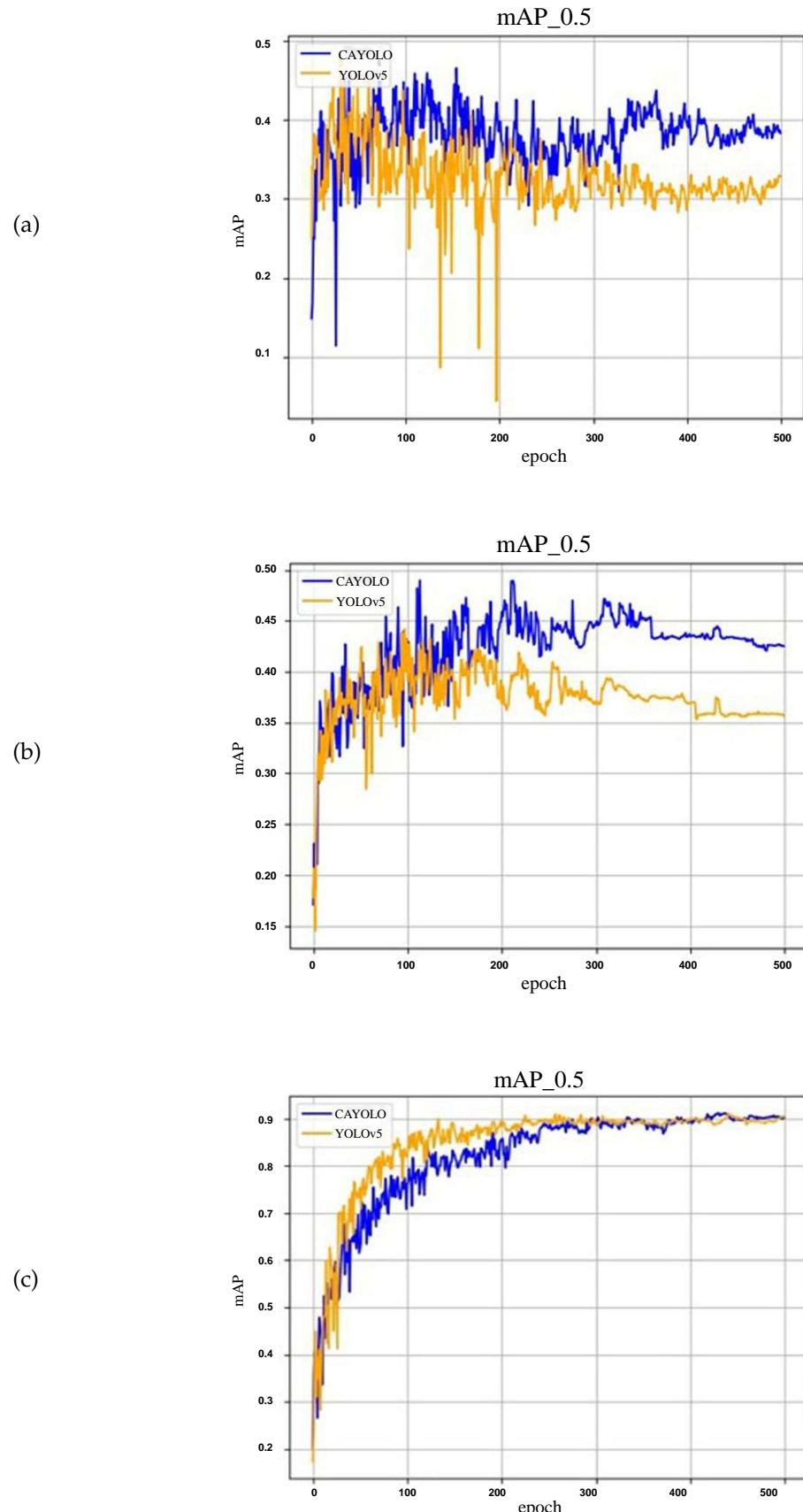


Figure 10. (a) (b) (c) is the mAP changes during training for the Indian, Czech, and Chinese road images.

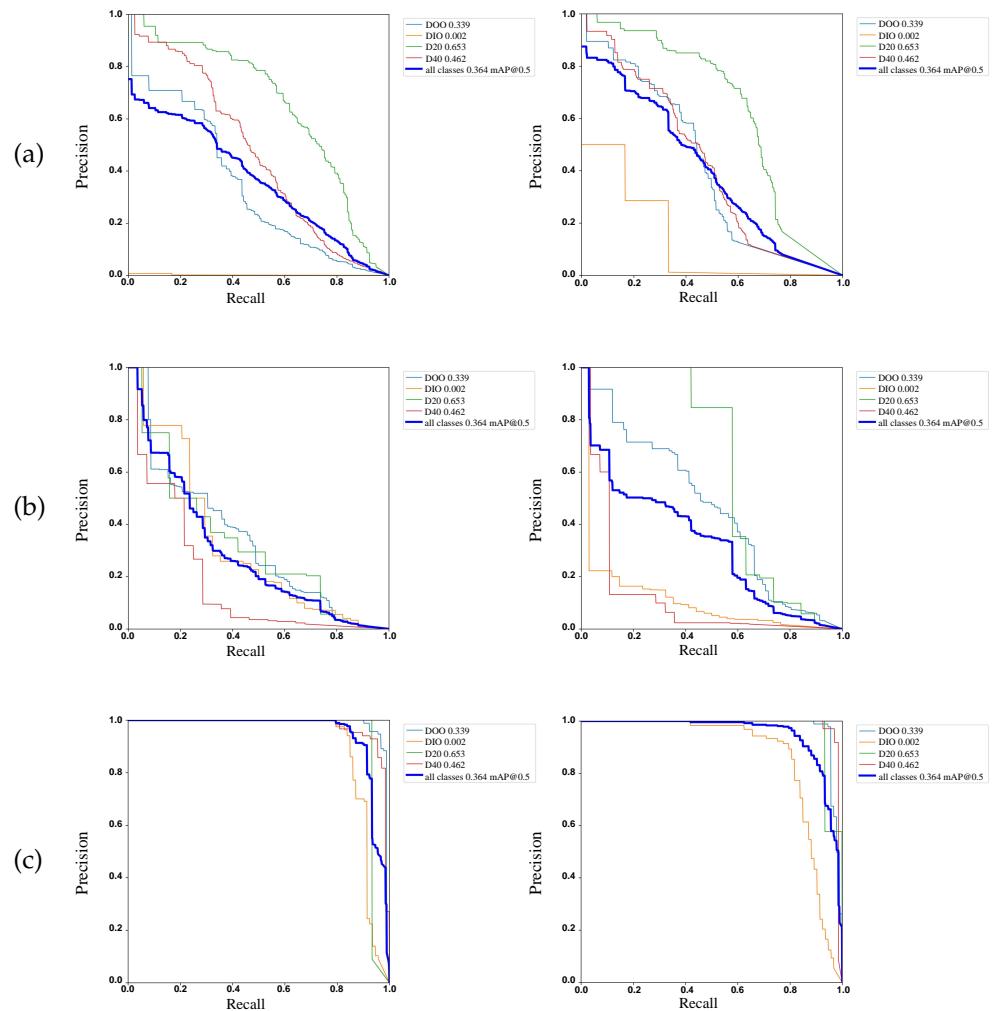


Figure 11. (a) (b) (c) are PR maps for India, the Czech Republic, and China, respectively, using YOLOv5 on the left and CAYOLO on the right.

In Figure 12(c), YOLOv5 fails to identify the damage, CAYOLO misidentifies D10 as D40 damage, there are too few D10 damages in the Indian road data, the network has low sensitivity to D10-like damages, and in Figure 12(e), it is a non-concrete pavement and misidentifies multiple D40 damages. In Figure 13(a), CAYOLO correctly identifies all the damages, while YOLOv5 has some problems in detecting D20 damages; in Figure 13(b) and Figure 13(e), YOLOv5 can accurately identify D10 damages while CAYOLO fails to do so. In Figure 14, both YOLOv5 and CAYOLO have some deficiencies in detecting D10 damage but present the same performance in detecting other damages.

We can see that CAYOLO can achieve better detection performance in most cases. CAYOLO utilizes an attention mechanism that can capture directional and channel information, and extract information that is often overlooked as the network deepens. It also uses a double convolution structure with small convolution kernels to extract features of small cracks. In addition, the Ghost Module is used to make the network lightweight and reduce the unnecessary computation brought by traditional convolutions during training. However, there is still significant room for improvement in detecting transverse cracks with CAYOLO. As shown in Figure 6, it has oscillation problems in the early stages of training, and in Table 4, its F1-score is slightly lower than some other studies. This is because the CAYOLO model has not achieved a good balance between precision and recall, and there are still some defects in the network's ability to recognize certain diseases. Nev-

ertheless, it has also made good progress in detection speed and some aspects, although there is still some distance to go in improving the F1-score.

480
481

(a)



(b)



(c)



(d)



(e)



Figure 12. Test Indian Road picture results. The left side is the actual label of the disease; the middle uses the YOLOv5 model test picture; the right uses the CAYOLO model test picture.

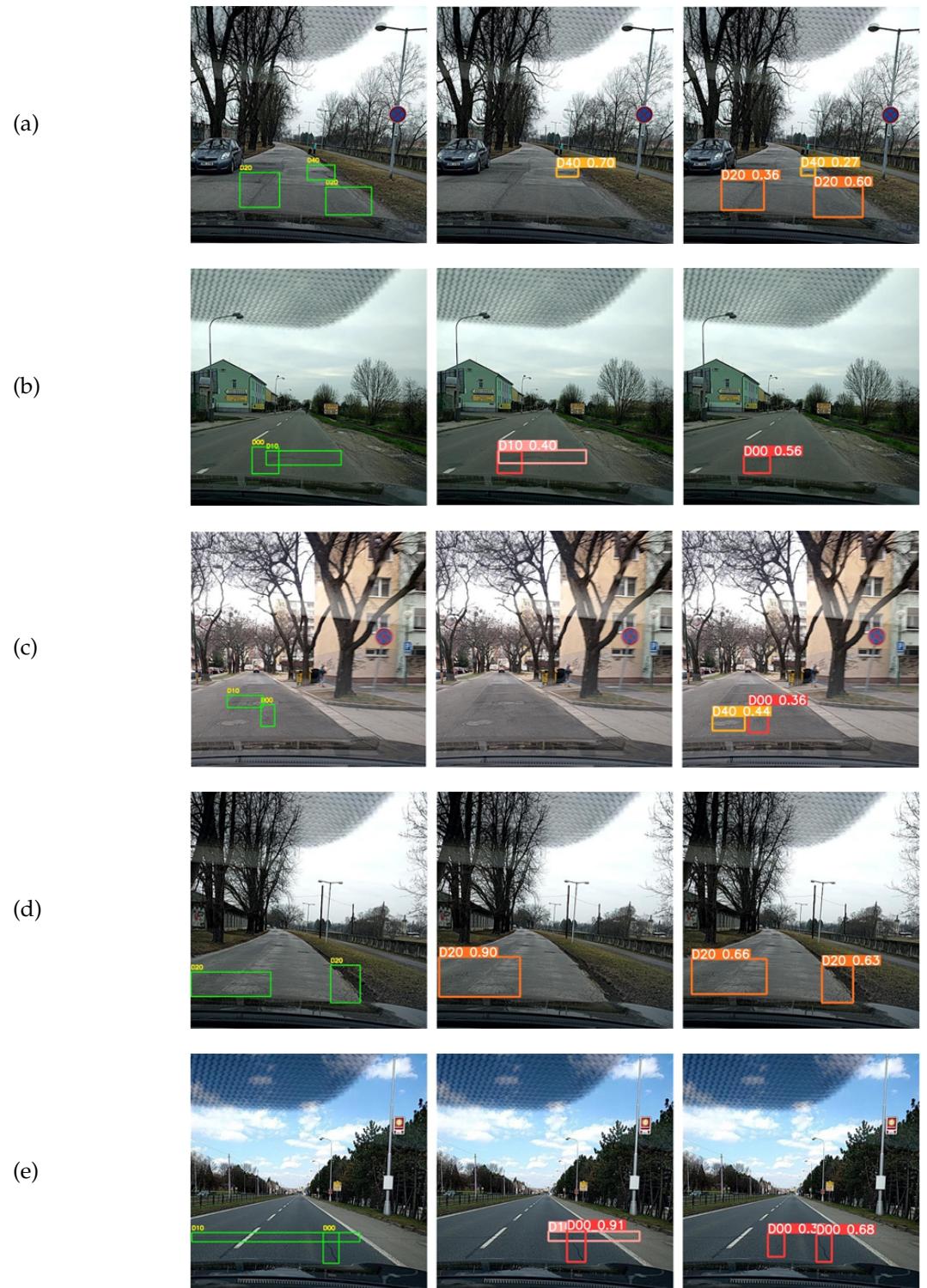


Figure 13. Test Indian Road picture results. The left side is the actual label of the disease; the middle uses the YOLOv5 model test picture; the right uses the CAYOLO model test picture.

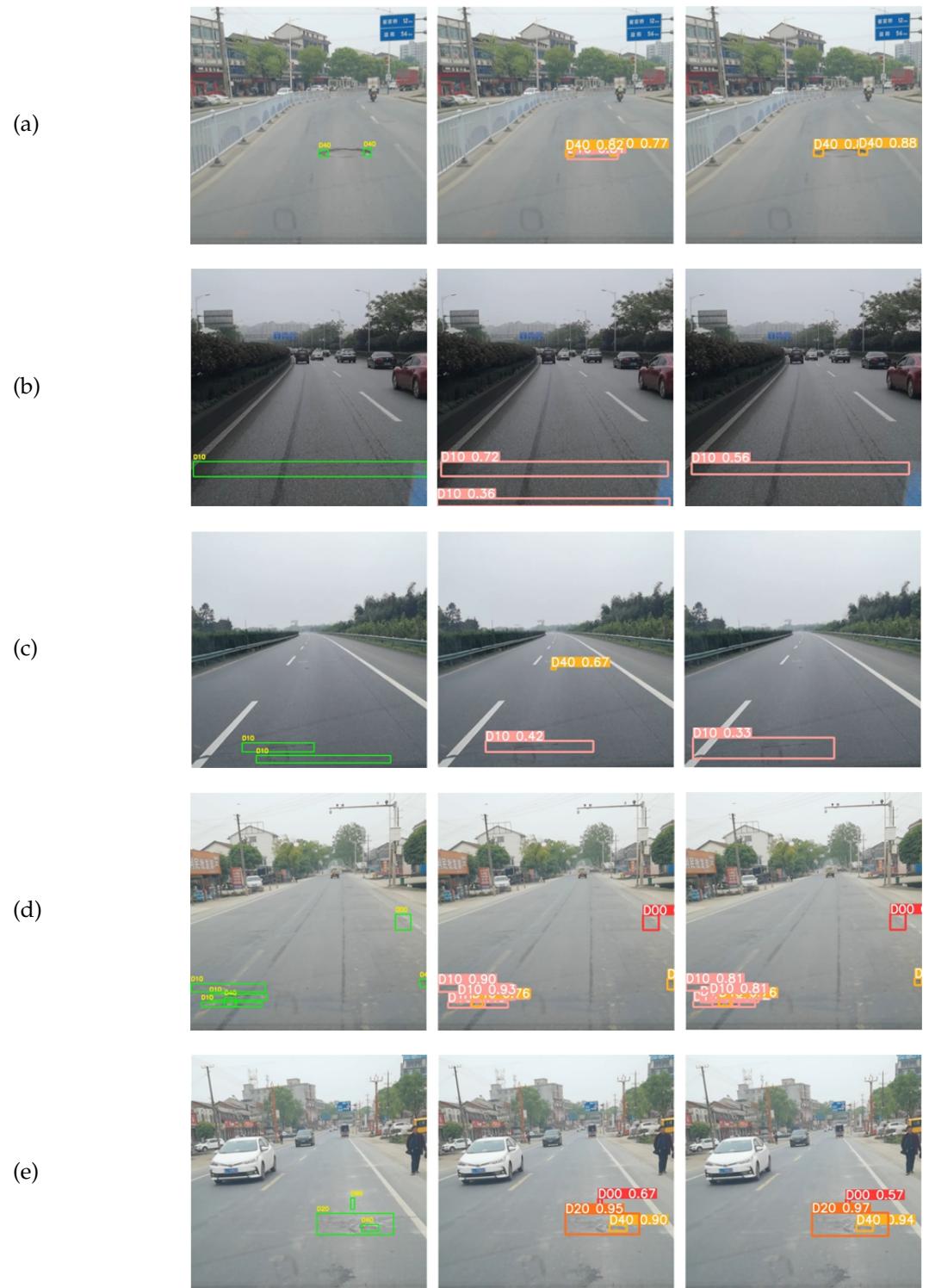


Figure 14. Test Indian Road picture results. The left side is the actual label of the disease; the middle uses the YOLOv5 model test picture; the right uses the CAYOLO model test picture.

6. Discussion and Conclusions

The DBC3 and CAC3 modules are created to construct the CAYOLO deep neural network according to the damage's characteristics. The network in this article focuses on the extraction of pavement crack features. The designed network model is compared with several mainstream algorithms in the field of road damage detection. The results show that using the CAYOLO network model can achieve significantly improved results on the publicly available dataset RDD-2020, with an mAP value of 68.86% for the alliga-

tor crack damage. CAYOLO also achieves the fastest detection speed among the same type of networks, with the time taken to detect a single image being 10.2ms. The trained YOLOv5 and CAYOLO models are retrained separately using data from several nations. The performance on Chinese road photographs is the best, with mAP reaching 94.9%, thus demonstrating the CAYOLO's ability to deliver noticeably enhanced results. Additionally, we'll make the relevant information we've gathered available to the public for use by other academics and keep gathering and updating the dataset.

We may infer from the trials that most network models challenge to extract lateral fractures. Thus in the subsequent studies, we must concentrate on enhancing the algorithm's power to do so. To increase the effectiveness of road damage identification in multiple nations, we will keep looking for better road damage detection methods in the future. Additionally, the dataset's size and quality must be increased, and the photos that were incorrectly labeled must be fixed. By doing so, the effect of mislabeling on the network can be partly mitigated. More data could be collected in the future via smartphones or cameras installed on buses. The government might also encourage citizens to capture road photographs for research by adopting suitable welfare policies.

Author Contributions: Conceptualization, Shuo.Z., S.C. and Z.L.; methodology, Z.L.; validation, Shuo.Z., S.C. and Z.L.; formal analysis, Sha.Z.; investigation, Y.Z.T.; resources, Z.L.; data curation, Shuo.Z. and Z.L.; writing original draft preparation, Z.L.; writing review and editing, Shuo.Z. and Z.L.; visualization, Shuo.Z.; supervision, S.B.C.; project administration, S.B.C. All authors have read and agreed to the published version of the manuscript.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available upon request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

- of Transport of the People's Republic of China, M. Interpretation of "2020 National Toll Road Statistical Bulletin". https://xxgk.mot.gov.cn/2020/jigou/glj/202110/t20211027_3623202.html, 2021. Accessed: 2025-03-16.
- Miller, T.R.; Zaloshnja, E. On a crash course: The dangers and health costs of deficient roadways **2009**.
- Chatterjee, A.; Tsai, Y.C. A Fast and Accurate Automated Pavement Crack Detection Algorithm. In Proceedings of the 2018 26th European Signal Processing Conference (EUSIPCO), 2018, pp. 2140–2144. <https://doi.org/10.23919/EUSIPCO.2018.8553388>.
- Shim, S.; Kim, J.; Lee, S.W.; Cho, G.C. Road damage detection using super-resolution and semi-supervised learning with generative adversarial network. *Automation in construction* **2022**, *135*, 104139.
- Liang, H.; Lee, S.C.; Seo, S. Automatic recognition of road damage based on lightweight attentional convolutional neural network. *Sensors* **2022**, *22*, 9599.
- Radopoulou, S.C.; Brilakis, I. Detection of multiple road defects for pavement condition assessment. *Eindhoven, The Netherlands* **2015**.
- Puspita, D.N.; Basuki, D.K.; Dewantara, B.S.B. Road damage detection and alert application using smartphones built-in sensors. In Proceedings of the 2022 International Electronics Symposium (IES). IEEE, 2022, pp. 506–510.
- Aravind, N.; Nagajothi, S.; Elavenil, S. Machine learning model for predicting the crack detection and pattern recognition of geopolymers concrete beams. *Construction and Building Materials* **2021**, *297*, 123785. <https://doi.org/https://doi.org/10.1016/j.conbuildmat.2021.123785>.
- Sun, Z.; Caetano, E.; Pereira, S.; Moutinho, C. Employing histogram of oriented gradient to enhance concrete crack detection performance with classification algorithm and Bayesian optimization. *Engineering Failure Analysis* **2023**, *150*, 107351. <https://doi.org/https://doi.org/10.1016/j.engfailanal.2023.107351>.
- de León, G.; Fiorentini, N.; Leandri, P.; Losa, M. A New Region-Based Minimal Path Selection Algorithm for Crack Detection and Ground Truth Labeling Exploiting Gabor Filters. *Remote Sensing* **2023**, *15*. <https://doi.org/10.3390/rs15112722>.

11. Samma, H.; Suandi, S.A.; Ismail, N.A.; Sulaiman, S.; Ping, L.L. Evolving Pre-Trained CNN Using Two-Layers Optimizer for Road Damage Detection From Drone Images. *IEEE Access* **2021**, *9*, 158215–158226. <https://doi.org/10.1109/ACCESS.2021.3131231>. 537
538
539
12. Luo, H.; Li, C.; Wu, M.; Cai, L. An Enhanced Lightweight Network for Road Damage Detection Based on Deep Learning. *Electronics* **2023**, *12*. <https://doi.org/10.3390/electronics12122583>. 540
541
542
13. Chen, J.; He, Y. A novel U-shaped encoder-decoder network with attention mechanism for detection and evaluation of road cracks at pixel level. *Computer-Aided Civil and Infrastructure Engineering* **2022**, *37*, 1721–1736. <https://doi.org/10.1111/mice.12826>. 543
544
545
14. Xu, H.; Chen, B.; Qin, J. A CNN-based length-aware cascade road damage detection approach. *Sensors* **2021**, *21*, 689. <https://doi.org/10.3390/s21030689>. 546
547
548
15. Sun, Z.; Han, Y.; Hua, Z.; Ruan, N.; Jia, W. Improving the efficiency and robustness of deepfakes detection through precise geometric features. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 3609–3618. <https://doi.org/10.48550/arXiv.2104.04480>. 549
550
551
16. 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*. 552
553
554
17. Dorafshan, S.; Thomas, R.J.; Maguire, M.J.C. Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete. *Construction and Building Materials* **2018**, *186*, 1031–1045. 555
556
557
18. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE* **1998**, *86*, 2278–2324. 558
559
560
19. Kulambayev, B.; Nurlybek, M.; Astaubayeva, G.; Tleuberdiyeva, G.; Zholdasbayev, S.; Tolep, A. Real-time road surface damage detection framework based on mask r-cnn model. *International Journal of Advanced Computer Science and Applications* **2023**, *14*. 561
562
563
20. Hacıefendioğlu, K.; Başağa, H.B. Concrete road crack detection using deep learning-based faster R-CNN method. *Iranian Journal of Science and Technology, Transactions of Civil Engineering* **2022**, *46*, 1621–1633. 564
565
566
21. Maeda, H.; Sekimoto, Y.; Seto, T.; Kashiyama, T.; Omata, H. Road damage detection and classification using deep neural networks with smartphone images. *Computer-Aided Civil and Infrastructure Engineering* **2018**, *33*, 1127–1141. 567
568
569
22. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2818–2826. 570
571
572
23. 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. 573
574
575
24. Howard, A.G.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, 2017, [arXiv:cs.CV/1704.04861]. 576
577
578
25. Wang, W.; Wu, B.; Yang, S.; Wang, Z. Road Damage Detection and Classification with Faster R-CNN. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 5220–5223. <https://doi.org/10.1109/BigData.2018.8622354>. 579
580
581
26. Wang, Y.J.; Ding, M.; Kan, S.; Zhang, S.; Lu, C. Deep Proposal and Detection Networks for Road Damage Detection and Classification. In Proceedings of the 2018 IEEE International Conference on Big Data (Big Data), 2018, pp. 5224–5227. <https://doi.org/10.1109/BigData.2018.8622599>. 582
583
584
27. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2016, pp. 770–778. 585
586
587
28. Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition, 2015, [arXiv:cs.CV/1409.1556]. 588
589
590
29. Chen, Q.; Gan, X.; Huang, W.; Feng, J.; Shim, H. Road Damage Detection and Classification Using Mask R-CNN with DenseNet Backbone. *Computers, Materials & Continua* **2020**, *65*. <https://doi.org/10.32604/cmc.2020.011191>. 591
592
593
30. He, K.; Gkioxari, G.; Dollar, P.; Girshick, R. Mask R-CNN. In Proceedings of the Proceedings of the IEEE International Conference on Computer Vision (ICCV), Oct 2017, pp. 2961–2969. 594
595
596
31. Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K.Q. Densely Connected Convolutional Networks. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017, pp. 4700–4708. 597
598
599
32. Maeda, H.; Kashiyama, T.; Sekimoto, Y.; Seto, T.; Omata, H. Generative adversarial network for road damage detection. *Computer-Aided Civil and Infrastructure Engineering* **2021**, *36*, 47–60. <https://doi.org/10.1111/mice.12561>. 600
601
602
33. Karras, T.; Aila, T.; Laine, S.; Lehtinen, J. Progressive Growing of GANs for Improved Quality, Stability, and Variation, 2018, [arXiv:cs.NE/1710.10196]. 603
604
605
34. Yin, J.; Qu, J.; Huang, W.; Chen, Q. Road damage detection and classification based on multi-level feature pyramids. *KSII Transactions on Internet and Information Systems (TIIS)* **2021**, *15*, 786–799. <https://doi.org/https://doi.org/10.3837/tiis.2021.02022>. 606
607
608

35. Zhao, Q.; Sheng, T.; Wang, Y.; Tang, Z.; Chen, Y.; Cai, L.; Ling, H. M2Det: A Single-Shot Object Detector Based on Multi-Level Feature Pyramid Network. *Proceedings of the AAAI Conference on Artificial Intelligence* **2019**, *33*, 9259–9266. <https://doi.org/10.1609/aaai.v33i01.33019259>. 591
592
36. Arya, D.; Maeda, H.; Ghosh, S.K.; Toshniwal, D.; Sekimoto, Y. RDD2020: An annotated image dataset for automatic road damage detection using deep learning. *Data in Brief* **2021**, *36*, 107133. <https://doi.org/https://doi.org/10.1016/j.dib.2021.107133>. 593
594
37. Chen, Z.; Wang, D.; Wang, Y.; Lin, S.; Jia, H.; Lin, P.; Liu, Y.; Chen, L. Research and Implementation of Road Damage Detection Algorithm Based on Object Detection Network. In Proceedings of the 2023 4th International Seminar on Artificial Intelligence, Networking and Information Technology (AINIT), 2023, pp. 446–450. <https://doi.org/10.1109/AINIT59027.2023.10212930>. 595
596
38. Hasan, M.M.; Sakib, S.; Deb, K. Road Damage Detection and Classification using Deep Neural Network. In Proceedings of the 2022 4th International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE), 2022, pp. 1–6. <https://doi.org/10.1109/ICECTE57896.2022.10114508>. 599
600
601
39. Wu, Z.; Li, Z.; Sun, X.; Zhao, R.; Bai, Y.; Na, J. Utilize Data Augmentation and Flexi Corner Block for Road Damage Detection. *Research Square* **2024**. <https://doi.org/10.21203/rs.3.rs-4570475/v1>. 602
603
40. Shim, S.; Kim, J.; Lee, S.W.; Cho, G.C. Road surface damage detection based on hierarchical architecture using lightweight auto-encoder network. *Automation in Construction* **2021**, *130*, 103833. <https://doi.org/10.1016/j.autcon.2021.103833>. 604
605
41. Lin, Y.C.; Chen, W.H.; Kuo, C.H. Implementation of Pavement Defect Detection System on Edge Computing Platform. *Applied Sciences* **2021**, *11*. <https://doi.org/10.3390/app11083725>. 606
607
42. C, R.K.; G, R. Road Damage Detection and Classification Using YOLOv5. In Proceedings of the 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT). IEEE, 2022, pp. 489–494. <https://doi.org/10.1109/ICICICT54557.2022.9917763>. 608
609
610
43. Su, B.; Zhang, H.; Wu, Z.; Zhou, Z. FSRDD: An Efficient Few-Shot Detector for Rare City Road Damage Detection. *IEEE Transactions on Intelligent Transportation Systems* **2022**, *23*, 24379–24388. <https://doi.org/10.1109/TITS.2022.3208188>. 611
612
44. Chen, J.; Yu, X.; Li, Q.; Wang, W.; He, B.G. LAG-YOLO: Efficient Road Damage Detector via Lightweight Attention Ghost Module. *Journal of Intelligent Construction* **2024**, *2*, 1–10. <https://doi.org/10.26599/JIC.2023.9180032>. 613
614
45. Deepa, D.; Sivasangari, A. An effective detection and classification of road damages using hybrid deep learning framework. *Multimedia Tools and Applications* **2023**, *82*, 18151–18184. 615
616
46. Rakshitha, R.; Srinath, S.; Kumar, N.V.; Rashmi, S.; Poornima, B. Deep Convolutional Neural Networks for Automated Road Damage Detection. In Proceedings of the International Conference on Advanced Communications and Machine Intelligence. Springer, 2024, pp. 155–165. 617
618
619
47. Arya, D.; Maeda, H.; Ghosh, S.K.; Toshniwal, D.; Mraz, A.; Kashiyama, T.; Sekimoto, Y. Deep learning-based road damage detection and classification for multiple countries. *Automation in Construction* **2021**, *132*, 103935. <https://doi.org/https://doi.org/10.1016/j.autcon.2021.103935>. 620
621
622
48. Arya, D.; Maeda, H.; Kumar Ghosh, S.; Toshniwal, D.; Omata, H.; Kashiyama, T.; Sekimoto, Y. Global Road Damage Detection: State-of-the-art Solutions. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data). IEEE, 2020, pp. 5533–5539. <https://doi.org/10.1109/BigData50022.2020.9377790>. 623
624
625
49. Hou, Q.; Zhou, D.; Feng, J. Coordinate Attention for Efficient Mobile Network Design. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2021, pp. 13713–13722. 626
627
50. Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018, pp. 7132–7141. 628
629
51. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional Block Attention Module. In Proceedings of the Proceedings of the European Conference on Computer Vision (ECCV), September 2018, pp. 3–19. 630
631
52. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2018, pp. 8759–8768. 632
633
53. Han, K.; Wang, Y.; Tian, Q.; Guo, J.; Xu, C.; Xu, C. GhostNet: More Features From Cheap Operations. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2020, pp. 1580–1589. 634
635

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. 636
637
638