applied sciences

MDPI

*Article*

# STrans-YOLOX: Fusing Swin Transformer and YOLOX for Automatic Pavement Crack Detection

**Hui Luo *, Jiamin Li, Lianming Cai and Mingquan Wu**

School of Information Engineering, East China Jiaotong University, Nanchang 330013, China
* Correspondence: lh_jxnc@163.com

**Abstract:** Automatic pavement crack detection is crucial for reducing road maintenance costs and ensuring transportation safety. Although convolutional neural networks (CNNs) have been widely used in automatic pavement crack detection, they cannot adequately model the long-range dependencies between pixels and easily lose edge detail information in complex scenes. Moreover, irregular crack shapes also make the detection task challenging. To address these issues, an automatic pavement crack detection architecture named STrans-YOLOX is proposed. Specifically, the architecture first exploits the CNN backbone to extract feature information, preserving the local modeling ability of the CNN. Then, Swin Transformer is introduced to enhance the long-range dependencies through a self-attention mechanism by supplying each pixel with global features. A new global attention guidance module (GAGM) is used to ensure effective information propagation in the feature pyramid network (FPN) by using high-level semantic information to guide the low-level spatial information, thereby enhancing the multi-class and multi-scale features of cracks. During the post-processing stage, we utilize $\alpha$-IoU-NMS to achieve the accurate suppression of the detection boxes in the case of occlusion and overlapping objects by introducing an adjustable power parameter. The experiments demonstrate that the proposed STrans-YOLOX achieves 63.37% mAP and surpasses the state-of-the-art models on the challenging pavement crack dataset.

**Keywords:** pavement crack detection; object detection; Swin Transformer; YOLOX; global guidance attention; multi-scale feature fusion; NMS; complex scenes

check for updates

**Citation:** Luo, H.; Li, J.; Cai, L.; Wu, M. STrans-YOLOX: Fusing Swin Transformer and YOLOX for Automatic Pavement Crack Detection. *Appl. Sci.* **2023**, *13*, 1999. https://doi.org/10.3390/app13031999

Academic Editor: Zengshun Chen

Received: 12 January 2023
Revised: 27 January 2023
Accepted: 29 January 2023
Published: 3 February 2023

**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

During the extensive use of highway transportation, road pavement defects may be formed due to climatic conditions, traffic loads and the service life of roads. Cracking is one of the common defects in pavements, and it can provide important information for the evaluation of road maintenance. The timely detection and repair of pavement cracks is of great significance in terms of reducing road maintenance costs and ensuring transportation safety. The traditional manual inspection method uses pavement photographs captured by road scanners for manual analysis. However, manual involvement in inspection is associated with high costs and low efficiency, and the detection results rely on the subjectivity of the inspectors. Thus, there is an important need to design an efficient and accurate pavement crack detection approach.

Benefiting from advancements in computer vision, the utilization of image processing techniques for pavement crack detection has been widely studied in recent years to overcome the drawbacks of the manual detection method. Oliveira et al. [1] put forward the first model for crack detection using image threshold segmentation, which combines the dynamic threshold and image entropy. However, methods based on threshold segmentation require different threshold values that depend on the background conditions. Considering the obvious crack edges, Zhao et al. [2] proposed an improved Canny operator for edge detection. Although edge-based detection methods are effective and fast, the crack region information and background region information is easily affected by noise,

which leads to a reduction in segmentation accuracy. In addition, some traditional machine learning methods have also been used for the pavement crack detection task, such as the support vector machine (SVM) and random forest (RF). For example, Shi et al. [3] proposed the CrackForest framework by introducing a random structured forest for integrating different features into a high-performance crack detector, which can effectively detect pavement cracks with uneven edges and complicated topological structures. However, methods based on traditional machine learning are not universal and rely on hand-crafted features. Moreover, complex pavement conditions, variable illumination and the feature inhomogeneity of cracks make the pavement crack detection task very challenging.

Recently, deep learning has achieved huge success in many computer vision tasks. By establishing a hierarchical structure from low-level features to high-level features, it can automatically learn and construct the complete features of objects. Deep learning, especially a deep convolutional neural network (CNN), has been applied in pavement crack detection [4,5], and it is superior to traditional image processing and machine learning. Therefore, employing a deep CNN for pavement crack detection is the general trend within industry.

Nevertheless, there are still some issues concerning existing CNN-based crack detection methods. First, a CNN cannot adequately model the long-range dependencies due to its inherent locality [6], especially when uneven illumination results in low contrast between cracks and background and when there is background noise (leaves, oil stains, water stains, etc.). Moreover, cracks are a kind of strip defect with strong boundary information, and the long-range dependencies are crucial to capturing the boundary details. In addition, cracks are characterized by a large variety of shapes and scales. Objects of different scales need to be recognized on the corresponding feature layers, that is, small-sized objects are recognized on low-level feature maps and large-sized objects are recognized on high-level feature maps. Moreover, for the same object, low-level feature maps with high resolution can generate detailed information such as boundaries and textures, while high-level feature maps with low resolution can capture advanced semantic information and so are suitable for object classification. Therefore, the effective fusion of different levels of feature maps helps improve the detection accuracy of multi-class and multi-scale cracks. Most methods mainly rely on a feature pyramid network (FPN) [7,8] to realize the fusion of feature maps with different scales. However, the inputs of the FPN are only achieved through simple upsampling and the concatenation of feature maps without considering the adaptability between different scales of feature maps and cracks. These issues make the performance of CNN-based models decline.

Recently, transformers [9] have been applied in the field of computer vision with increasing successes. A transformer makes up for the deficiency of a CNN, and its unique self-attention mechanism can learn the long-range dependencies between image pixels and capture global information. Particularly for the objects that vary greatly in terms of their textures, shapes and scales, a transformer usually produces better performance. Based on the transformer concept, Ze et al. [10] designed Swin Transformer, which uses a hierarchical architecture resembling a CNN to build a hierarchical transformer with significant computational efficiency. Swin Transformer achieves state-of-the-art performance in computer vision tasks.

In object detection tasks, the YOLO series [11–14] plays a vital role in single-stage detectors. In particular, the latest YOLOX [15], with its quick detection speed and high detection accuracy, has strong recognition ability in vision tasks. Motivated by the above analysis, an automatic pavement crack detection architecture is proposed and named STrans-YOLOX. It fuses Swin Transformer and YOLOX to address existing challenges. The network adds the strength of a self-attention mechanism to learn the long-range dependencies based on the efficient local modeling of a CNN. During the feature extraction stage, the last CSPLayer module in the backbone is replaced by the Swin Transformer block to compensate for the insufficient long-range dependencies of the convolution operations and to extract rich feature information from crack images. In addition, some CSPLayer

modules in the FPN are replaced with Swin Transformer blocks to obtain the self-attention of multi-scale features to preserve the edge detail information. It can also explore the potential of Swin Transformer to integrate multi-scale features in the FPN. Then, a global attention guidance module (GAGM) is designed to change the original input way of different scales of feature maps and enhance the feature fusion ability of multi-class and multi-scale cracks. By introducing the GAGM between feature maps of adjacent scales, the global context information of high-level feature maps is obtained, and the advanced semantic information is used to weight the low-level feature maps so as to guide the effective multi-scale feature fusion between low-level feature information and high-level feature information. Thus, it can overcome the semantic and scale inconsistency when multi-class and multi-scale crack features are input into the FPN. In addition, to alleviate the phenomenon of missing detection caused by occlusion and overlapping objects, we adopt the $\alpha$-IoU-NMS algorithm during the post-processing stage. By introducing an adjustable power parameter to meet the different levels of suppression accuracy of the detection boxes, $\alpha$-IoU-NMS can suppress the redundant boxes more accurately and reduce the missing detection of occlusion and overlapping objects. Compared with the original YOLOX, our proposed STrans-YOLOX further improves the detection accuracy of pavement cracks with acceptable detection speed.

To sum up, our major contributions are as listed below:

1. We propose an automatic pavement crack detection network that fuses Swin Transformer and YOLOX. It complements the long-range dependencies between pixels compared with pure CNN-based networks. Compared with pure CNNs, it supplements the long-range dependencies between pixels. Moreover, compared with pure transformer-based networks, it preserves the advantage of the CNN in local modeling, not impairing the backbone to learn the local feature information.
2. Swin Transformer is introduced to the model to make up for the insufficient modeling of the long-range dependencies of the convolution operations, thereby solving the problems of low contrast and background noise. Thus, it can better retain the edge detail features of the input images and increase the crack detection accuracy.
3. A GAGM is designed to guide the low-level detail information with the high-level semantic information, which effectively overcomes the semantic and scale inconsistency when multi-class and multi-scale features are input into the FPN.
4. $\alpha$-IoU-NMS is utilized to suppress the redundant detection boxes more accurately, which can reduce the missing detection of occlusion and overlapping objects.
5. STrans-YOLOX achieves mAP of 63.37% on the challenging pavement crack dataset, which surpasses the state-of-the-art models.

## 2. Related Work

### *2.1. Pavement Crack Detection Methods Based on CNNs*

By sharing the convolutional kernel parameters, a CNN has the ability to learn object features efficiently. Therefore, CNNs have also been extensively studied in research on pavement crack detection. Zhang et al. [16] first adopted a supervised deep CNN in the field of pavement crack detection, and they successfully achieved crack image classification with better detection performance than methods that are dependent on hand-crafted features. However, the designed network was primitive, resulting in a slow detection speed and low detection accuracy. Tang et al. [17] introduced a crack detection network using Faster R-CNN, which improved the detection accuracy of cracks with various lengths under different illuminations by transfer learning and the multi-task enhancement method, although the algorithm was not good in terms of its detection speed. Maeda et al. [18] designed a detection network based on SSD Inception V2 and SSD MobileNet for the fast defect detection of vehicle-mounted pavement images and achieved acceptable accuracy and recall. Mandal et al. [19] could accurately identify and locate transverse cracks, longitudinal cracks and alligator cracks in different positions by exploiting YOLOv2. Du et al. [20] employed YOLOv3 to predict defect locations and classes, which achieved a further in-

crease in the detection speed and accuracy, and they also discussed the applicability of the model under different lighting conditions. In summary, crack detection methods based on deep CNNs have been proven to be superior to traditional image processing-based methods and machine learning-based methods. However, due to the intrinsic locality of convolution operations, CNNs mainly focus on a small range of features, which leads to insufficient modeling of the long-range dependencies. To alleviate this problem, Yan et al. [21] introduced deformable convolution into the model to improve the crack detection accuracy in complex scenes, while Wang et al. [22] introduced dilated convolution with different dilation rates into the FPN to improve the global information and representation ability of feature maps. Although some studies have been conducted, they still have limitations.

### 2.2. Transformer-Based Methods

Vaswani et al. [9] first put forward the transformer, which is completely composed of the self-attention mechanism, to solve natural language processing (NLP) tasks. A transformer overcomes the slow training speed of an RNN and achieves rapid parallel computation and the ability to capture long-range dependencies. Dosovitskiy et al. [23] designed Vision Transformer (ViT), which can efficiently learn image features by directly applying the image block sequence to the transformer. Beal et al. [24] proposed ViT-FRCNN for object detection through utilizing ViT as the feature extraction network, that is, replacing the CNN backbone with the transformer, and it produced a competitive result with regard to CNNs. Considering the large-scale differences between visual entities and the high computation complexity caused by the high resolution of images, Ze et al. [10] proposed Swin Transformer. Swin Transformer improves the multi-scale feature extraction quality of high-resolution images via introducing the hierarchical structure typically utilized in CNNs for constructing a hierarchical transformer. Thus, it significantly reduces the computation complexity and realizes state-of-the-art performance in computer vision tasks.

To address the issue of object detection in complex scenes, Zhu et al. [25] put forward TPH-YOLOV5, which uses the transformer prediction head (TPH) to replace the original detector, and they proved that it has better performance than the baseline model. Zhang et al. [26] proposed a transformer FPN with full active feature interaction across space and scale, which was able to extract rich context information. Srinivas et al. [27] proposed the BoTNet backbone, which simply and effectively improved the performance by replacing the last three bottleneck blocks of ResNet with a global self-attention mechanism. Thus, combining the advantages of a CNN in processing the underlying information with the strength of a transformer in modeling the long-range dependencies can extract powerful features.

## 3. Methodology

### 3.1. Overview of YOLOX

Based on the different proportions of the model depth and width, we can divide YOLOX into four versions: YOLOX-X, YOLOX-L, YOLOX-M and YOLOX-S. The whole structure of YOLOX (Figure 1) is composed of a backbone, FPN and prediction head. Among them, the structure of CSPDarkNet [28] with an SPPBottleneck is exploited as the backbone and the PANet used in YOLOv4 is utilized as the FPN. The model includes three prediction heads, which are employed to detect small-, medium- and large-size targets, respectively. The YOLO series are the most extensively utilized single-stage detectors, which offer the optimal trade-off between speed and accuracy. Moreover, YOLOX ingeniously integrates the excellent advances in object detection with the YOLO network, including the decoupling head, data augmentation, anchor-free and SimOTA. It also achieves the best results of the current YOLO series. Therefore, we choose YOLOX as the baseline model.

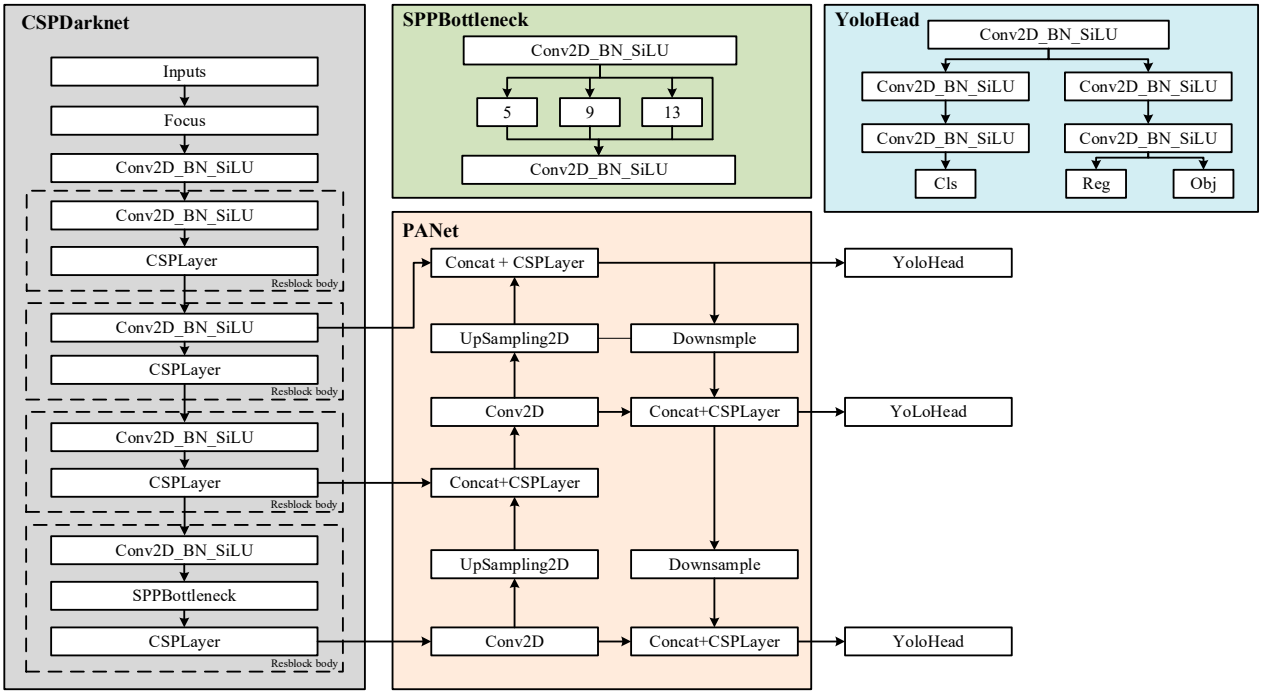



**Figure 1.** The architecture of YOLOX.

By comparing the experimental results, we observe that YOLOX-M performs better on the pavement crack dataset than YOLOX-X, YOLOX-L and YOLOX-S. Therefore, we choose to improve YOLOX-M to make it more suitable for detecting the crack features.

### *3.2. STrans-YOLOX*

The framework of STrans-YOLOX is displayed in Figure 2. The structure mainly includes three sections: feature extraction network, FPN and prediction head. In the feature extraction network, the CNN is first adopted to extract the object features to preserve its local modeling ability. Then, in the last residual stage of the backbone, Swin Transformer is used to transform the feature maps into sequence embedding to capture the long-range dependencies between pixels. In the FPN, the extracted multi-level features are passed through the GAGM modules to obtain the global context information of the high-level features, and the advanced semantic information is utilized for weighting the low-level feature maps, thereby guiding the effective multi-scale feature fusion between the low-level feature information and high-level feature information. The fused features are input into Swin Transformer again for the multi-scale feature aggregation to obtain more comprehensive and richer crack features. Finally, three enhanced feature maps in different scales are input into the prediction heads for the regression of the detection results. In addition, the $\alpha$-IoU-NMS algorithm is utilized during the post-processing stage for accurate box suppression to further improve the detection accuracy.

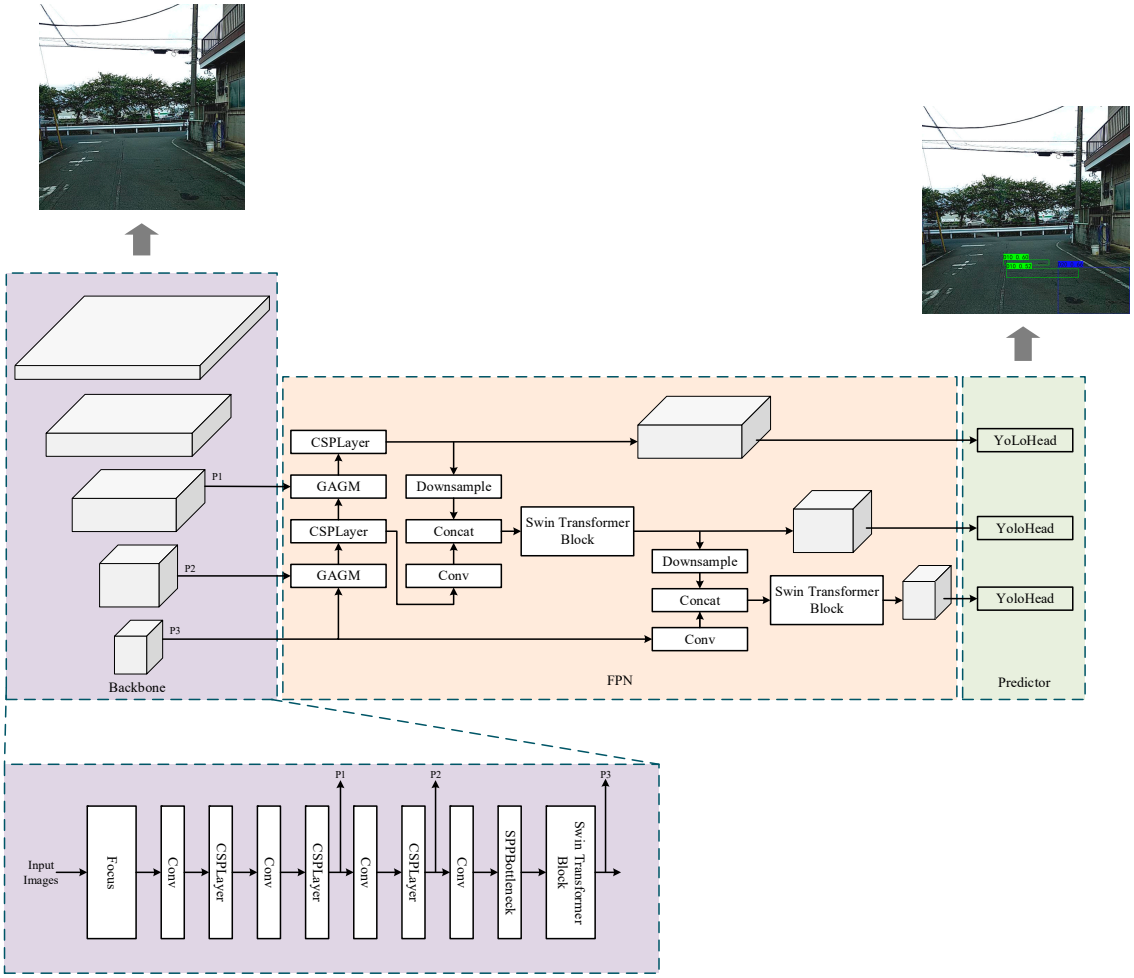



**Figure 2.** The architecture of STrans-YOLOX.

#### 3.2.1. Swin Transformer Block

A CNN can effectively capture local information, although it has limitations in terms of modeling the long-range dependencies. To solve this issue, Swin Transformer blocks are added into the CNN-based structure, which can enhance the long-range dependencies and detailed information. Our design ideas are: (1) to use a CNN to efficiently learn the spatial features from high-resolution images; and (2) to use Swin Transformer to process and aggregate the feature information captured by the CNN. The fusion design combines the advantages of convolution and self-attention. By means of spatial downsampling using the CNN, the model allows the self-attention to focus on small-resolution feature maps with long-range dependencies and an appropriate computational cost. Therefore, we introduce the Swin Transformer block in the last residual CSP bottleneck block and replace some CSPLayer modules in the FPN with the Swin Transformer blocks. The architecture of the Swin Transformer block is illustrated in Figure 3, which consists of windows multi-head self-attention (W-MSA), shifted windows multi-head self-attention (SW-MSA), and multilayer perceptron (MLP). The LayerNorm (LN) layer is inserted between the modules to make the training process steadier. A residual connection is employed after each module. The W-MSA constrains the computation of the self-attention to a window, with the following output:

$$\hat{x}^l = \text{W-MSA}(LN(x^{l-1})) + x^{l-1} \tag{1}$$

The output $x^l$ of the $l^{th}$ MLP layer is computed as:

$$x^l = \text{MLP}(LN(\hat{x}^l)) + \hat{x}^l \tag{2}$$

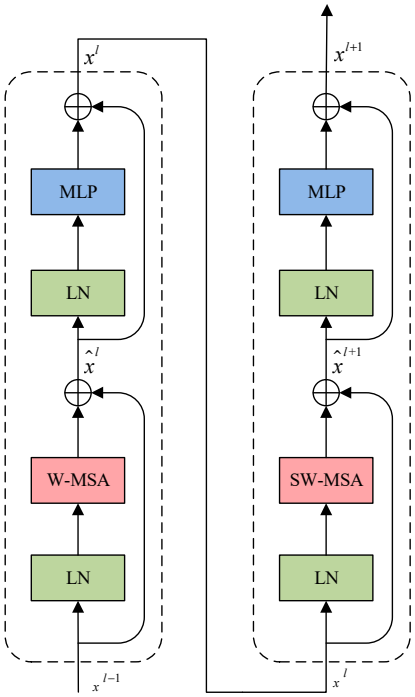



**Figure 3.** The architecture of the Swin Transformer block.

Then, in the next cascaded Swin Transformer module, SW-MSA is added with a shift operation compared with W-MSA to establish information interaction between the different windows without increasing the computation overhead. The process can be expressed as follows:

$$\hat{x}^{l+1} = \text{SW-MSA}(LN(x^l)) + x^l \tag{3}$$

$$x^{l+1} = \text{MLP}(LN(\hat{x}^{l+1})) + \hat{x}^{l+1} \tag{4}$$

where $\hat{x}^{l+1}$ denotes the output of $l+1^{th}$ SW-MSA, and $x^{l+1}$ is the output of $l+1^{th}$ MLP. Swin Transformer solves the problem of the large computation of the transformer by dividing the image into several windows and constraining the computation of the self-attention in the windows. In the second calculation operation of the self-attention, the shifted window method is utilized to segment image patches with different boundaries so as to realize cross-window communication. By introducing Swin Transformer, the model can effectively model the long-range dependencies, and it has better performance for pavement crack and edge detection in complex scenes.

#### 3.2.2. Global Attention Guidance Module (GAGM)

The effective fusion of feature maps in different scales is helpful for improving the detection accuracy of cracks with multi-class and multi-scale features. However, many existing methods do not take into account the semantic and scale inconsistency between feature maps and real cracks, resulting in the insufficient extraction of multi-scale features. In order to solve this issue, and inspired by [29,30], we propose the GAGM. The corresponding architecture is illustrated in Figure 4. By introducing the GAGM between feature maps of adjacent scales, the advanced semantic information of high-level feature map guides the detailed information of low-level feature maps for effective multi-scale feature fusion. Specifically, the low-level feature map $f_l$ (high-resolution) and high-level feature map $f_h$ (low-resolution) are input into different branches of the GAGM, respectively. First, a $3 \times 3$ convolution is applied to adjust the channel dimension of the $f_h$, and the adjusted feature map $X_H$ can be formulated as:

$$X_H = \sigma(BN(Conv_{3\times3}(X_h))) \tag{5}$$

where, $\sigma$ refers to the SiLU activation.

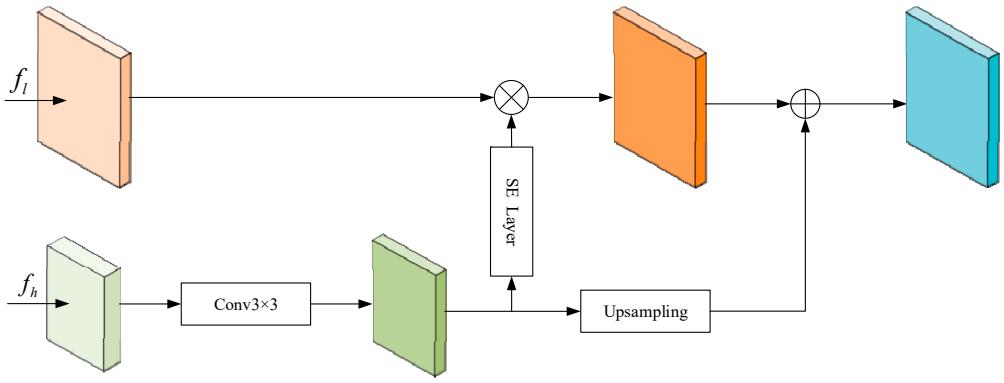



**Figure 4.** The architecture of the GAGM.

Then, the global context information of the high-level feature map is obtained through the SE attention mechanism, which can be calculated as:

$$X_a = sigmoid(BN(Conv_{1\times1}(\sigma(BN(Conv'_{1\times1}(G(X_H))))))) \quad (6)$$

where $G(\cdot)$ refers to the global pooling operation. The kernel size of the $Conv_{1\times1}$ is $C/r \times 1 \times 1$ and the kernel size of the $Conv'_{1\times1}$ is $C \times 1 \times 1$. Moreover, $r$ denotes the reduction rate required to control the size of the attention mechanism module.

The obtained attention feature map $X_a$ is multiplied by the low-level feature map $f_l$ for weighting, so the channel attentional feature information is applied to guide the low-level feature map to enhance the class information. Here, $X_H$ is upsampled to the same resolution with the low-level feature map and then added to the weighted $f_l$ so that the class information of the high-level feature map can be fully integrated with the position detailed information of the low-level feature map. The final output $X_{out}$ is formulated as:

$$X_{out} = (X_a \otimes X_l) \oplus upsample(X_H) \quad (7)$$

where $\otimes$ and $\oplus$ represent the element-wise multiplication and element-wise addition, and $upsample(\cdot)$ represents 2 times upsampling.

In a word, instead of directly feeding the FPN with feature maps of different scales for the purpose of feature fusion, the proposed GAGM takes into account the semantic and scale inconsistency of the features. It can effectively guide and propagate information among different levels of feature maps, thereby enhancing the fusion ability of multi-class and multi-scale feature information.

#### 3.2.3. $\alpha$-IoU-NMS

During the post-processing stage, the NMS algorithm sorts all the detection boxes according to their scores. Then, it selects the box with the maximum score and suppresses the other redundant boxes. However, NMS simply employs the IoU index to suppress the redundant boxes by setting the scores of the boxes with large overlaps to zero. Therefore, when there are occlusion and overlapping objects, NMS is prone to suppress the boxes wrongly, resulting in missing detection. Inspired by [31], we propose the $\alpha$-IoU-NMS algorithm in this article, which combines the $\alpha$-IoU and NMS algorithms.

(1) Traditional NMS

The traditional NMS algorithm sorts the detection boxes of each class according to their scores in descending order. Then, it selects the detection box with the maximum score and suppresses the other detection boxes whose overlaps with the maximum score detection

box exceed the setting threshold. Suppose there are two detection boxes, $A$ and $B$, and $iou$ is the ratio of the intersection and union of $A$ and $B$, as shown in the following formula:

$$iou = \frac{in\,sec\,tion}{union} = \frac{A \cap B}{A \cup B} \tag{8}$$

The traditional NMS method is defined by:

$$S_i = \begin{cases} S_i, iou(M, bi) < N_t \\ 0, iou(M, bi) \geq N_t \end{cases} \tag{9}$$

where $M$ is the box with the maximum current confidence score, $b_i$ is the other detection box, $N_t$ denotes the setting threshold, $S_i = 0$ represents the redundant box, and $S_i = S_i$ means the different object box.

(2) α-IoU-NMS

In order to obtain more accurate bounding box suppression, we apply the power transformation to the IoU in the NMS to generate the power IoU, which is called α-IoU. The α-IoU can be calculated by:

$$\alpha - iou = iou^{\alpha} \tag{10}$$

The formula for the α-IoU-NMS is defined by:

$$S_i = \begin{cases} S_i, \alpha - iou(M, bi) < N_t \\ 0, \alpha - iou(M, bi) \geq N_t \end{cases} \tag{11}$$

In the α-IoU-NMS algorithm, an adjustable power parameter is introduced to meet different levels of box suppression accuracy. Experiments show a positive effect in the case of $\alpha < 1$ for the pavement crack detection task, since α-IoU is a monotonically increasing function with a limit when $\alpha < 1$. When the IoU of the object is large and the overlap degree is high, the suppression will be strict. When the IoU of the object is small and it is more likely to detect potential objects, the suppression will be not so strict. Thus, it can alleviate the phenomenon of missing detection, especially when there are occlusion and overlapping objects. Therefore, it is shown that the proper selection of $\alpha$ ($\alpha < 1$) helps to improve the box suppression accuracy by adaptively increasing the probability of low IoU objects.

## 4. Experiments and Results

### *4.1. Dataset Acquisition*

The experimental dataset utilized in this article was selected from RDD2020 [32]. The images in RDD2020 were captured using vehicle-mounted smartphones and annotated with multiple road damage labels, including longitudinal cracks, transverse cracks, and alligator cracks. These images were collected from the asphalt pavement of urban roads, meaning they can be used for crack detection in asphalt pavement. We first picked out the images with crack labels, and we then removed the images with low quality. Moreover, in order to ensure the diversity of the pavement crack samples, the dataset we selected includes all kinds of weather (sunny, cloudy and rainy), illumination conditions and multiple noise interference, such as lane lines, water stains, shadows and ruts. We finally selected 7428 images from RDD2020 to form the pavement crack dataset, which contains three types of pavement cracks: Longitudinal Cracks (D00), Transverse Cracks (D10), and Alligator Cracks (D20). Visual examples of these classes of cracks are shown in Figure 5. It can be seen from the figure that the pavement crack dataset with complex scenes is challenging. Moreover, the resolution of each image is 600 × 600 pixels. From the dataset, we randomly selected 81% (6016 images) for the training set, 9% (669 images) for the validation set and 10% (743 images) for the test set. The distribution of instances in the dataset is detailed in Table 1.




**Figure 5.** Sample images for pavement cracks and schematic diagram of each class.

**Table 1.** The instance distribution of each class in the pavement crack dataset.

| Class Name | Train Set | Validation Set | Test Set | Total |
|---|---|---|---|---|
| D00 | 3262 | 399 | 388 | 4049 |
| D10 | 3243 | 348 | 388 | 3979 |
| D20 | 5009 | 542 | 648 | 6199 |
| Total | 11,514 | 1289 | 1424 | 14,277 |

### *4.2. Evaluation Metrics*

To test the model performance in the task of pavement crack detection, we use the Average Precision (AP), mean Average Precision (mAP), and Frames Per Second (FPS) as the evaluation metrics. According to the combinations of the ground truth annotation and the prediction result, the samples can be classified as follows: true positive samples (true positive, *TP*), negative samples that are mistakenly identified to be positive (false positive, *FP*), positive samples that are mistakenly identified to be negative (false negative, *FN*), and true negative samples (true negative, TN).

The Precision (*P*) is the proportion of positive samples to all the positive samples in the prediction. The Recall (*R*) is the proportion of positive samples to all the samples that are truly positive. They are calculated with follows:

$$P = \frac{TP}{TP + FP} \tag{12}$$

$$R = \frac{TP}{TP + FN} \tag{13}$$

The AP measures the performance of the model in terms of the *P* and *R*, and the mAP represents the average value of the AP for all the categories. A higher mAP value indicates that the model has better comprehensive detection performance for all the categories. The calculation formulas are as follows:

$$AP = \int_0^1 P(R)dR \tag{14}$$

$$mAP = \frac{\sum_{i=1}^{c} AP_i}{c} \tag{15}$$

where $AP_i$ denotes the AP value of each class, and $c$ denotes the class number.

### *4.3. Implementation Details*

We implement STrans-YOLOX on the PyTorch 1.8.1 framework using Python 3.6. All the experiments run in an NVIDIA GeForce RTX 2080 GPU (8G RAM) on a Windows 10 operating system.

During the training stage, to save on training time and reflect the idea of transfer learning, we use some parameters trained by YOLOX-M on a large dataset as a pre-trained model, as STrans-YOLOX and YOLOX share most of the parameters and there are many weights that can be transferred from YOLOX-M to STrans-YOLOX. An Adam optimizer with a weight decay factor of 0.0005 is adopted for the training, and a learning rate decay strategy of cosine annealing with an initial learning rate of 0.001 is used. We adopt a freeze training strategy. The backbone is first frozen with a batch size of 8 for 50 training epochs. Then, the backbone is unfrozen and the whole model is trained with a batch size of 4 until it converges. In freeze training, the backbone of the model is frozen, so the backbone parameters are not learned. The memory occupation is small, and only the parameters of the remaining part of the model are learned. In unfreezing training, the backbone of the model is unfrozen and all the parameters are learned, which takes up a lot of memory. Thus, the batch size for freeze training can be set larger compared with that for unfreezing training. The reason for using freeze training is that some features extracted from the backbone are generic. Freeze training can speed up the training efficiency and prevent the weight from being destroyed, which essentially also reflects the idea of transfer learning. The image size is resized to 640 × 640, and Mosaic data augmentation is enabled. The training curve of the loss function is mapped in Figure 6, and the loss value tends to be stable after 120 epochs.

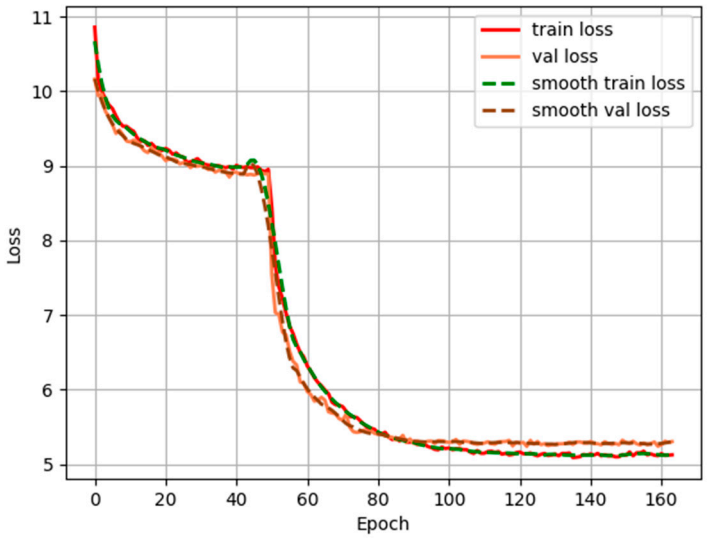



**Figure 6.** The training curve of the loss function.

### *4.4. Experimental Results and Analysis*

(1) Comparison of the Results with the State-of-the-Art Models

To assess the effectiveness of STrans-YOLOX, we compared it with other state-of-the-art object models on the same test set, including SSD [33], CenterNet [34], RetinaNet [35], YOLOv3 [13], YOLOv4 [14], and YOLOv5. The specific detection results are shown in Table 2. In terms of the mAP, STrans-YOLOX achieves 63.73%, which is superior to other models. Our model achieves a significant improvement in the AP values of all three classes, and the value of mAP is improved by 3.17% compared with the original YOLOX. STrans-YOLOX strengthens the extraction ability of crack features in complex scenes and further enhances the detection performance for multi-class and multi-scale cracks, while

also dealing with occlusion objects more appropriately. With respect to the detection speed, although the generated FPS of STrans-YOLOX is reduced compared to the original YOLOX, it can reach 26.59, which is acceptable in terms of meeting the requirement of real-time detection. Therefore, when comprehensively compared with the current mainstream detection models, STrans-YOLOX is effective in detecting pavement cracks in the challenging dataset.

**Table 2.** Comparison results of different models on the pavement crack test set.

| Method | Accuracy (%) | | | | FPS |
|---|---|---|---|---|---|
| | AP (D00) | AP (D10) | AP (D20) | mAP | |
| SSD | 50.37 | 30.14 | 68.57 | 49.70 | 27.65 |
| CenterNet | 55.56 | 44.92 | 65.34 | 55.28 | 37.32 |
| RetinaNet | 58.30 | 23.61 | 73.33 | 51.75 | 21.82 |
| YOLOv3 | 55.40 | 44.88 | 70.47 | 56.91 | 26.17 |
| YOLOv4 | 54.30 | 46.08 | 70.61 | 57.00 | 19.99 |
| YOLOv5-M | 57.17 | 38.51 | 65.33 | 53.67 | 39.43 |
| YOLOX-M | 59.92 | 50.98 | 70.18 | 60.36 | 39.01 |
| STrans-YOLOX | 62.11 | 55.78 | 73.30 | 63.73 | 26.59 |

Figure 7 is a comparison of the F1-Score curves of YOLOX and STrans-YOLOX of three classes of cracks. The F1-Score, also known as the balanced F1-Score, is defined as the harmonic average of the Precision and Recall. The F1-Score takes values ranging from 0 to 1, with 1 representing the best output of the model and 0 representing the worst output of the model. It can be seen from the graph that STrans-YOLOX greatly improves the F1-Score in all three classes compared with the original YOLOX.

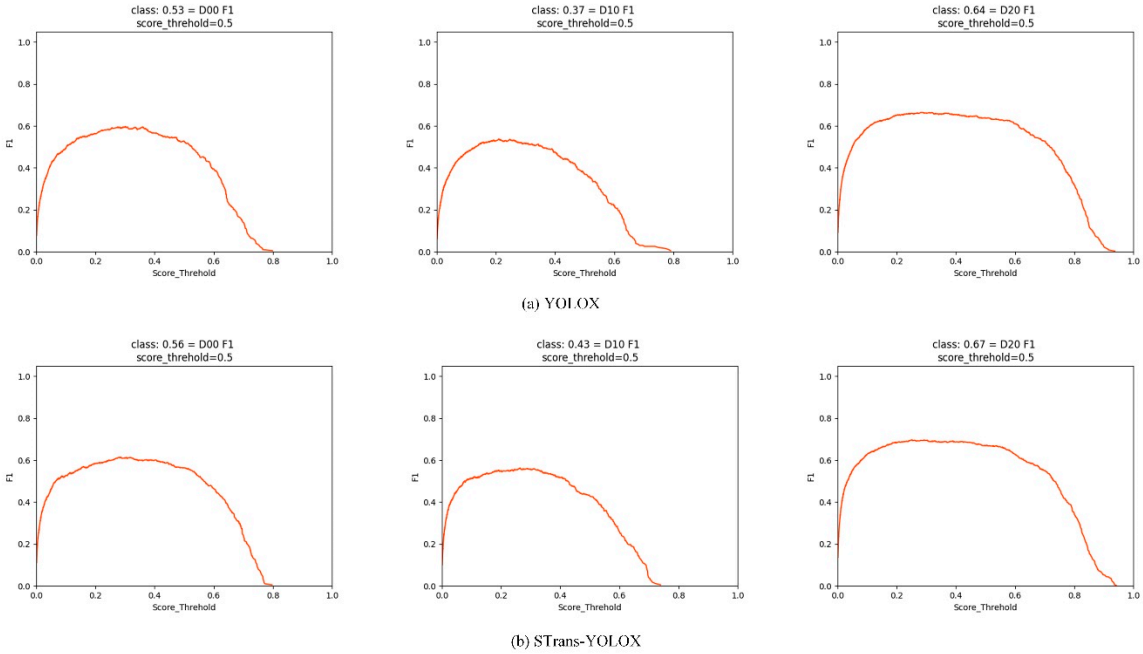



**Figure 7.** F1-Score curves of YOLOX and STrans-YOLOX of three classes of cracks.

(2) Ablation Study

To verify the effectiveness of our proposed model, we conduct ablation experiments to analyze the importance of each module, including the Swin Transformer block, GAGM and $\alpha$-IoU-NMS. The impact of each module is listed in Table 3.

**Table 3.** Comparison results of the ablation experiments.

| Method | AP (D00)/% | AP (D10)/% | AP (D20)/% | Map/% |
|---|---|---|---|---|
| YOLOX | 59.92 | 50.98 | 70.18 | 60.36 |
| YOLOX + Swin Transformer | 60.01 (↑0.09) | 53.48 (↑2.5) | 72.59 (↑2.41) | 62.03 (↑1.67) |
| YOLOX + Swin Transformer + GAGM | 61.44 (↑1.43) | 55.63 (↑2.15) | 72.63 (↑0.04) | 63.23 (↑1.2) |
| YOLOX + Swin Transformer + GAGM + α-IoU-NMS | 62.11 (↑0.67) | 55.78 (↑0.15) | 73.30 (↑0.67) | 63.73 (↑0.5) |

(a) Effect of Swin Transformer Blocks

After using the Swin Transformer blocks to replace some CSPLayer modules, the model mAP is increased by 1.67%, and the AP values of the transverse cracks and alligator cracks are greatly increased. The experiment demonstrates that the method combining Swin Transformer and a CNN has better performance than the pure CNN-based network. This is mainly because Swin Transformer can effectively improve the feature extraction ability in complex scenes by introducing long-range dependencies, and it is able to keep detailed information concerning input images.

(b) Effect of GAGM

After the GAGM modules are added into the FPN, the mAP of the model is improved by 1.2%. The GAGM has a good detection effect, especially for longitudinal cracks and transverse cracks, and it plays a role in the detection of multi-class and multi-scale objects.

(c) Effect of α-IoU-NMS

By employing the α-IoU-NMS algorithm, STrans-YOLOX increases the mAP by 0.5% with no increase in the parameter number. This demonstrates that the improved NMS is more effective than the traditional NMS in dealing with the error suppression caused by occlusion and overlapping objects. We also explore the influence of different values of α on the detection results. As shown in Table 4, the performance of STrans-YOLOX varies with the values of α. With an increase in the values of α, the mAP first increases and then decreases. When α is 1, the performance is equivalent to using the traditional NMS. When α exceeds 1, the mAP decreases significantly. The reason for this may be that the box suppression is loose when α is too small, which leads to false detection. When α is large, the box suppression is strict, resulting in missing detection. Therefore, the value of α is finally set as 0.75 based on the experimental results.

**Table 4.** Effects of different values of α on the detection results.

| α | 0.5 | 0.6 | 0.7 | 0.75 | 0.8 | 0.9 | 1 | 2 | 3 |
|---|---|---|---|---|---|---|---|---|---|
| mAP/% | 63.19 | 63.63 | 63.62 | 63.73 | 63.71 | 63.62 | 63.23 | 58.93 | 52.78 |

(3) Visual Detection Results on the Pavement Crack Dataset

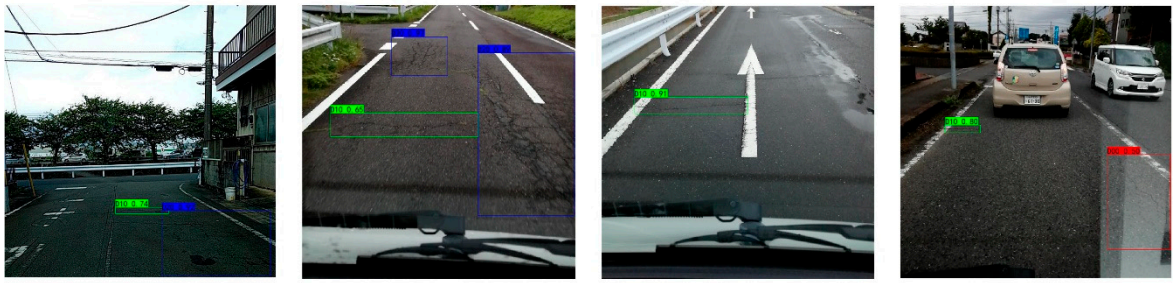
To visually assess the utility of STrans-YOLOX, we pick some representative images of the detection results obtained using different models, which are depicted in Figure 8, where (a) is the original crack images to be detected, (b) is the detection results based on RetinaNet, (c) is the detection results based on YOLOv4, (d) is the detection results based on YOLOX, and (e) is the detection results based on STrans-YOLOX proposed in this paper. From the detection results, it is clear that some models have missing detection and false detection in complex scenes, and our proposed model can accurately detect and locate multi-class and multi-scale crack objects. In addition, the heatmap visualization results obtained using different detection networks are illustrated as Figure 9. The figure demonstrates that STrans-YOLOX can accurately focus on regions of interest in complex scenes.



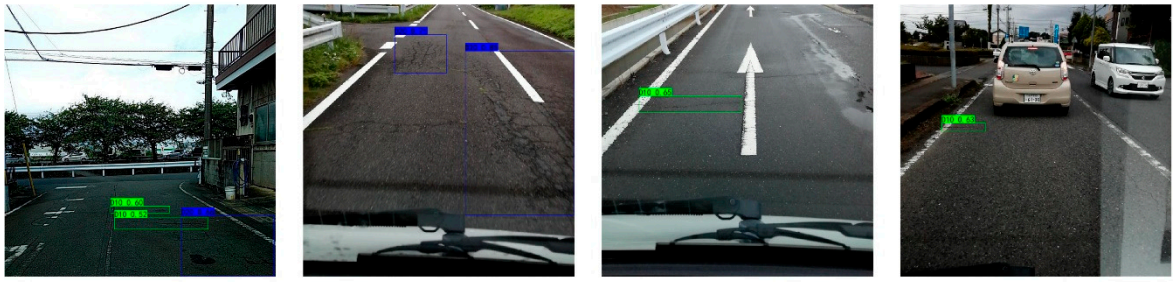
(a) original images



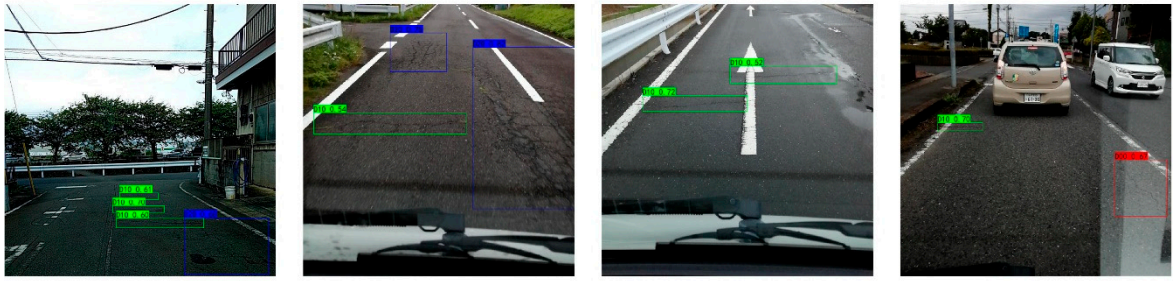
(b) RetinaNet



(c) YOLOv4



(d) YOLOX



(e) STrans-YOLOX

**Figure 8.** The visual detection results of different models.




**Figure 9.** The heatmap results of different models.

## 5. Conclusions

We propose an automatic pavement crack detection model that fuses Swin Transformer and a CNN, which can deal with different kinds of cracks. Aiming to address the problem of the insufficient modeling of long-range dependencies of pure CNNs, the proposed model introduces Swin Transformer to enhance the learning ability of the feature information in complex scenes and better keep the edge detailed information in input images. A GAGM is proposed to improve the means of feature fusion in the FPN through guiding the low-level spatial information with advanced semantic information. Thus, it can effectively improve the feature fusion ability of multi-class and multi-scale cracks. In the post-processing stage, $\alpha$-IoU-NMS is utilized to improve the detection accuracy of occlusion and overlapping objects by introducing an adjustable power parameter. Experiments demonstrate that the proposed STrans-YOLOX achieves a mAP of 63.37% and surpasses other state-of-the-art models on the challenging pavement crack dataset. We also notice in the experiments that

some linear joints on the roads are falsely identified as cracks, increasing the number of false positives. This indicates that the effects of other crack-like defects on cracks should be considered in future studies to further improve the detection accuracy of pavement cracks. In conclusion, this research can provide some reference significance for vehicle-mounted and low-cost pavement crack detection in urban roads.

**Author Contributions:** Conceptualization, H.L., J.L. and L.C.; data curation, J.L. and M.W.; formal analysis, H.L.; funding acquisition, H.L.; methodology, J.L. and H.L.; software, J.L. and H.L.; validation, L.C. and M.W.; investigation, J.L. and M.W.; resources, H.L. and L.C.; writing—original draft, H.L., J.L., L.C. and M.W.; writing—review and editing, H.L. and M.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was supported by the National Natural Science Foundation (NNSF) of China (No. 62262021), the Key R&D Program of Jiangxi Province under Grant No. 20202BBEL53001 and the Key Project of Science and Technology Research of Jiangxi Department of Education under Grant No. GJJ200603.

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The dataset generated and analyzed during the current research was selected from open-source information available online. RDD2020: https://github.com/sekilab/RoadDamageDetector/#road-damage-dataset-2019 (accessed on 20 November 2021).

**Acknowledgments:** The authors gratefully acknowledge the reviewers and editor-in-chief for their careful work, and they also thank the data sharer for their selfless dedication.

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

## References

1. Oliveira, H.; Correia, P.L. Automatic road crack segmentation using entropy and image dynamic thresholding. In Proceedings of the 7th European Signal Processing Conference, Glasgow, Scotland, UK, 24–28 August 2009; pp. 622–626.
2. Zhao, H.; Qin, G.; Wang, X. Improvement of canny algorithm based on pavement edge detection. In Proceedings of the 3rd International Congress on Image and Signal Processing, Yantai, China, 16–18 October 2010; pp. 964–967.
3. Shi, Y.; Cui, L.; Qi, Z.; Meng, F.; Chen, Z. Automatic Road Crack Detection Using Random Structured Forests. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3434–3445. [CrossRef]
4. Li, S.; Zhao, X. Convolutional neural networks-based crack detection for real concrete surface. In Proceedings of the SPIE Conference on Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems, Denver, CO, USA, 5–8 March 2018; pp. 105983V-1–105983V-7.
5. Han, Z.; Chen, H.; Liu, Y.; Li, Y.; Du, Y.; Zhang, H. Vision-Based Crack Detection of Asphalt Pavement Using Deep Convolutional Neural Network. *Iran. J. Sci. Technol. Trans. Civ. Eng.* **2021**, *45*, 2047–2055. [CrossRef]
6. Wang, X.; Girshick, R.; Gupta, A.; He, K. Non-local neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 7794–7803.
7. Lin, T.Y.; Dollar, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 936–944.
8. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path Aggregation Network for Instance Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8759–8768.
9. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS), Long Beach, CA, USA, 4–9 December 2017; pp. 6000–6010.
10. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, QC, Canada, 10–17 October 2021; pp. 9992–10002.
11. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
12. Redmon, J.; Farhadi, A. YOLO9000: Better, Faster, Stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 22–25 July 2016; pp. 7263–7271.
13. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
14. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. YOLOv4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:360200410934.

15. Ge, Z.; Liu, S.; Wang, F.; Li, Z.; Sun, J. YOLOX: Exceeding YOLO Series in 2021. *arXiv* **2021**, arXiv:2107.08430.
16. Zhang, L.; Yang, F.; Zhang, Y.D.; Zhu, Y.J. Road Crack Detection Using Deep Convolutional Neural Network. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3708–3712.
17. Tang, J.; Mao, Y.; Wang, J.; Wang, L. Multi-task Enhanced Dam Crack Image Detection Based on Faster R-CNN. In Proceedings of the 4th International Conference on Image, Vision and Computing (ICIVC), Xiamen, China, 5–7 July 2019; pp. 336–340.
18. Maeda, H.; Sekimoto, Y.; Seto, T.; Kashiyama, T.; Omata, H. Road damage detection and classification using deep neural networks with smartphone images. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 1127–1141. [CrossRef]
19. Mandal, V.; Uong, L.; Adu-Gyamfi, Y. Automated Road Crack Detection Using Deep Convolutional Neural Networks. In Proceedings of the IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 10–13 December 2018; pp. 5212–5215.
20. Du, Y.; Pan, N.; Xu, Z.; Deng, F.; Kang, H. Pavement distress detection and classification based on YOLO network. *Int. J. Pavement Eng.* **2021**, *22*, 1659–1672. [CrossRef]
21. Yan, K.; Zhang, Z.H. Automated Asphalt Highway Pavement Crack Detection Based on Deformable Single Shot Multi-Box Detector Under a Complex Environment. *IEEE Access* **2021**, *9*, 150925–150938. [CrossRef]
22. Wang, H.; Wang, Z.; Yu, L. YOLO Object Detection Algorithm with Hybrid Atrous Convolutional Pyramid. In Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA), Guilin, China, 7–10 August 2022; pp. 940–945.
23. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, arXiv:2010.11929.
24. Beal, J.; Kim, E.; Tzeng, E.; Dong, H.P.; Kislyuk, D. Toward Transformer-Based Object Detection. *arXiv* **2020**, arXiv:2012.09958.
25. Zhu, X.; Lyu, S.; Wang, X.; Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW), Montreal, BC, Canada, 11–17 October 2021; pp. 2778–2788.
26. Zhang, D.; Zhang, H.; Tang, J.; Wang, M.; Hua, X.; Sun, Q. Feature Pyramid Transformer. In Proceedings of the European Conference on Computer Vision, Glasgow, Scotland, UK, 23–28 August 2020; pp. 323–339.
27. Srinivas, A.; Lin, T.Y.; Parmar, N.; Shlens, J.; Abbeel, P.; Vaswani, A. Bottleneck Transformers for Visual Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Virtual Event, Nashville, TN, USA, 20–25 June 2021; pp. 16514–16524.
28. Wang, C.Y.; Mark Liao, H.Y.; Wu, Y.H.; Chen, P.Y.; Hsieh, J.W.; Yeh, I.H. CSPNet: A New Backbone that can Enhance Learning Capability of CNN. In Proceedings of the EEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Seattle, WA, USA, 14–19 June 2020; pp. 1571–1580.
29. Cao, Y.; Xu, J.; Lin, S.; Wei, F.; Hu, H. GCNet: Non-Local Networks Meet Squeeze-Excitation Networks and Beyond. In Proceedings of the IEEE/CVF International Conference on Computer Vision Workshop (ICCVW), Seoul, Republic of Korea, 27–28 October 2019; pp. 1971–1980.
30. Mohammadi, S.; Noori, M.; Bahri, A.; Majelan, S.G.; Havaei, M. CAGNet: Content-aware guidance for salient object detection. *Pattern Recognit* **2020**, *103*, 107303. [CrossRef]
31. He, J.; Erfani, S.; Ma, X.; Bailey, J.; Chi, Y.; Hua, X.S. Alpha-IoU: A Family of Power Intersection over Union Losses for Bounding Box Regression. *arXiv* **2021**, arXiv:2110.13675v2.
32. Arya, D.; Maeda, H.; Ghosh, S.K.; Toshniwal, D.; Sekimoto, Y. An annotated image dataset for Automatic Road Damage Detection using Deep Learning. *Data Brief* **2021**, *36*, 107133. [CrossRef] [PubMed]
33. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. SSD: Single Shot Multi-box Detector. In Proceedings of the European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, 8–16 October 2016; pp. 21–37.
34. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. CenterNet: Keypoint Triplets for Object Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 6568–6577.
35. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal Loss for Dense Object Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 318–327. [CrossRef] [PubMed]


**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.