mathematics

MDPI

Article

TransMF: Transformer-Based Multi-Scale Fusion Model for Crack Detection

Xiaochen Ju [1], Xinxin Zhao [1] and Shengsheng Qian [2,*]

[1] Railway Engineering Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing 100081, China; juxc2008@163.com (X.J.); xyzxx000@163.com (X.Z.)
[2] Institute of Automation, Chinese Academy of Sciences, Beijing 100090, China
* Correspondence: shengsheng.qian@nlpr.ia.ac.cn

Abstract: Cracks are widespread in infrastructure that are closely related to human activity. It is very popular to use artificial intelligence to detect cracks intelligently, which is known as crack detection. The noise in the background of crack images, discontinuity of cracks and other problems make the crack detection task a huge challenge. Although many approaches have been proposed, there are still two challenges: (1) cracks are long and complex in shape, making it difficult to capture long-range continuity; (2) most of the images in the crack dataset have noise, and it is difficult to detect only the cracks and ignore the noise. In this paper, we propose a novel method called *Transformer-based Multi-scale Fusion Model* (TransMF) for crack detection, including an Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM). The Encoder Module uses a hybrid of convolution blocks and Swin Transformer block to model the long-range dependencies of different parts in a crack image from a local and global perspective. The Decoder Module is designed with symmetrical structure to the Encoder Module. In the Fusion Module, the output in each layer with unique scales of Encoder Module and Decoder Module are fused in the form of convolution, which can release the effect of background noise and strengthen the correlations between relevant context in order to enhance the crack detection. Finally, the output of each layer of the Fusion Module is concatenated to achieve the purpose of crack detection. Extensive experiments on three benchmark datasets (CrackLS315, CRKWH100 and DeepCrack) demonstrate that the proposed TransMF in this paper exceeds the best performance of present baselines.

Keywords: crack detection; convolutional neural network; transformer; multi-scale fusion

MSC: 68T45

Citation: Ju, X.; Zhao, X.; Qian, S. TransMF: Transformer-Based Multi-Scale Fusion Model for Crack Detection. *Mathematics* **2022**, *10*, 2354. https://doi.org/10.3390/math10132354

Academic Editor: Teng Li

Received: 5 June 2022
Accepted: 4 July 2022
Published: 5 July 2022

1. Introduction

With the development of Deep Learning (DL), Artificial Intelligence (AI) has ushered in great prosperity. It has become a popular trend to find ways to solve tasks automatically instead of manually, permeating all aspects of our lives, such as Facial Recognition (FR) [1], Vehicle License Plate Recognition (VLPR) [2], Image Classification [3–5] and so on. More importantly, AI has been able to provide support for the safety of life and property, in which a relatively popular task is crack detection. A crack is a line structure and crack detection is a kind of segmentation task, or object detection task, which detects cracks on the object surface in an automatic way, and has practical significance for human survival and life. Public service infrastructures, such as bridges [6–8], and pavements [9–12], are directly related to the safety of human life, and cracks on the surface of which, to some extent, represent the degree of damage of these public facilities. Therefore, it is critical and important to detect cracks more quickly and efficiently.

The encoder–decoder framework is a popular method to solve crack detection, which has been widely used in an image segmentation domain. The encoder takes an input image and generates a high-dimensional feature vector and the decoder takes a high-dimensional feature vector and generates a semantic segmentation mask. High-dimensional features

can be aggregated with at multiple levels. U-Net [13] is a pioneering work in the field of crack detection, using a symmetric encoder–decoder framework with skip connection firstly, where both the decoder and encoder are implemented with Convolutional Neural Networks (CNNs). Based on U-Net [13], many excellent methods have been proposed [12,14,15]. However, the network framework of the above methods is relatively simple, and requires a large amount of data augmentation to improve the segmentation effect [16]. In addition, the Convolutional Neural Networks (CNN) have the limitation of the receptive field. During the convolution process, the weight calculation is performed in the receptive field of a certain size. Generally speaking, the receptive field is not very large, and combined with the slender feature of cracks, the convolution cannot capture the long-range dependencies of cracks, which may result in performance degradation. Recently, Transformer [17] was proposed to model long-range dependencies for contextual encoding of natural language, which has developed rapidly in the field of computer vision in the last 2 years, and a number of variants have been proposed, such as Vision Transformers [18], Swin Transformer [19], Star-Transformer [20], etc. CrackFormer [21] is a Crack Transformer network (CrackFormer) with a transformer encoder–decoder structure, which proposes a self-attention block and scaling-attention block for fine-grained crack detection. Today, there has been some research using a transformer-based multi-scale method on many applications. Kong et al. [22] proposed a multi-scale temporal transformer for skeleton-based action recognition.Xiao et al. [23] proposed a multi-scale spatiotemporal transformer to efficiently aggregate contextual information in long-time sequences of video frames. Yuan et al. [24] proposed a multi-scale adaptive segmentation network based on Swin Transformer for remote sensing image segmentation.

In addition, Deep Learning can obtain the deep contour features of an image, but the shallow features are rich in texture information of the image that contains unwanted noise. Noise is a thorny problem in crack detection [25], and how to design robust network architecture is very important for crack detection. A very common method is to simply fuse shallow features and deep features using a skip connection. For example, YOLOv3-Lite [26] adopts depthwise separable convolution, feature pyramid, and YOLOv3 to detect cracks in aircraft structures. CrackSeg [27] introduces a novel multi-scale dilated convolutional module to learn rich deep convolutional features under complex background. Although the above methods have achieved good results in solving the problem of background noise, they still cannot pay more attention to the detection object while removing the background noise.

Overall, there are two challenges that need to be addressed in order to effectively model crack detection:

- Challenge 1: Cracks on the surface of objects are thin and long with complex shapes, which makes it difficult to detect cracks. At present, many methods use Convolutional Neural Networks (CNNs) to extract deep features of cracks, but the convolutional features can only model local features, and ignore the global feature relationship that can capture long-range dependencies. The long-range dependencies can coordinate the overall characteristics of the cracks. Therefore, we conclude the first challenge is: how can we model the long-range dependencies of different parts in a crack image from a local and global perspective for a better crack image understanding?
- Challenge 2: Images of cracks are taken from various facilities, such as bridges, buildings, railways, roads and other public building facilities, or household items such as cups and tables. Therefore, the actual scene of cracks is complex and diverse, and the crack detection task cannot ignore these background noises, which leads to incorrect detection of cracks and reduces the detection efficiency. The crack features extracted by convolution are divided into shallow low-level features and deep high-level features. Shallow low-level features contain the texture information of cracks, and deep high-level features contain the general contour information of cracks. However, shallow low-level features are highly affected by the background noise, while deep high-level features are less affected by background noise. So, we concluded that the

second challenge is: how can we remove the effect of background noise from the low-level features of crack images, which is an important prerequisite to enhance the crack detection?

Motivated by the above discussions, we propose a novel method called *Transformer-based Multi-scale Fusion Model* (TransMF) for crack detection, which consists of three modules: Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM). For *Challenge 1*, we use a hybrid of convolution and transformer approaches, combining global and local perspectives, to explore the long-range dependencies of various parts in the crack. Specifically, we design an Encoder Module (EM) and Decoder Module (DM), which are symmetrical and contain multiple layers of Conv-Block and Swin Transformer block, respectively, as shown in Figure 1. For *Challenge 2*, we design a Fusion Module (FM) to fuse the multi-scale features from different layers in the encoder and decoder to mitigate the effect of background noise through fusing low-level and high-level features in the form of convolution, which can assist in strengthening the correlations between the relevant context for enhancing the crack detection. In general, the contributions in this paper are summarized as follows:

- We propose a novel *Transformer-based Multi-scale Fusion Model* (TransMF) for crack detection, including an Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM), which performs encoding and decoding symmetrically, and fuses at different levels to preserve the information of the underlying features and deep features.
- Both the Encoder Module (EM) and Decoder Module (DM) utilize a hybrid architecture of Convolutional Neural Networks (CNN) and Swin Transformers, which can capture both detailed spatial information from local features and the global context encoded by Transformers.
- In this paper, a multi-scale Fusion Module (FM) is designed, rather than a simple skip connection, to effectively fuse the encoder and decoder features of each layer with different scales to form a comprehensive representation, preventing wrong detection of noise.
- We evaluate TransMF on three benchmark datasets (CrackLS315 [28], CRKWH100 [28] and DeepCrack [14]). Experimental results demonstrate that the proposed TransMF exceeds the best performance of present baselines.

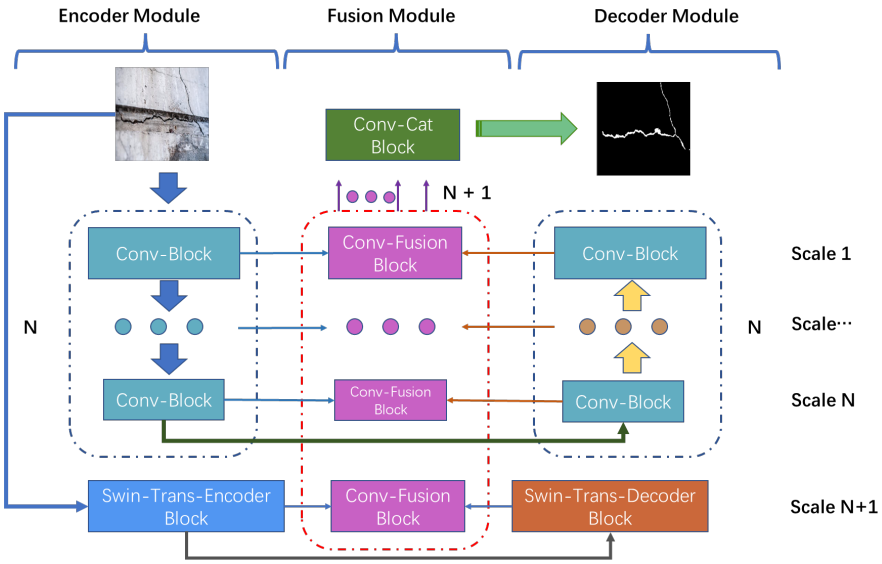


Figure 1. The overall framework of the proposed TransMF, which consists of three modules: an Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM). Both the Encoder Module (EM) and Decoder Module (DM) utilize a hybrid of Convolutional Neural Networks (CNNs) and Swin Transformers, which are symmetrical. The Fusion Module (FM) fuses multi-scale features to form a comprehensive representation.

2. Related Work

2.1. Crack Detection

Surface cracks are everywhere around us, being on things such as daily necessities, public building facilities, transportation tools and so on. The existence of cracks brings us great inconvenience and will endanger our lives and health to a certain extent. Crack detection has thus become a popular research field. The traditional crack detection method is that the inspector goes to the scene to detect cracks using the detection instrument, which is time-consuming, laborious and costly, also bringing great danger to the inspector. As Machine Learning [29] evolved, people designed manual features to train models for initial automatic detection. As Deep Learning [30] then came to a boom, models were learned in a black-box manner, facilitating the further development of crack detection, during which Convolutional Neural Networks (CNNs) [31] were widely used. In recent years, much excellent crack detection work has been proposed, such as [13,14,28,32].

Currently, crack detection based on deep learning can be divided into two kinds of methods [33] as shown in the left part in Figure 2: (1) image processing-based method for crack detection; (2) machine learning method for crack detection. In the first method, which utilizes handcrafted features, high-resolution images are preprocessed to remove noise and shadows using filters, segmentation and other approaches. Edge detection, segmentation, or pixel analysis are used to highlight or segment the cracked part in the image. In the second method, the dataset is preprocessed and a machine learning model is used to classify the cracked regions.

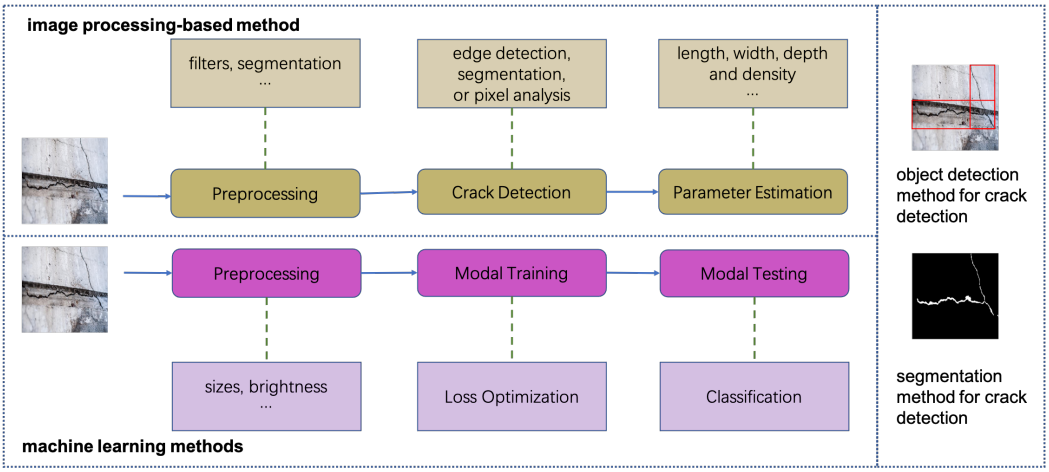


Figure 2. On the left are two methods of crack detection, including: (1) image processing-based method for crack detection, which utilizes handcrafted features; (2) machine learning method for crack detection, which utilizes learned features by the model. The right side shows that the methods based on machine learning include two categories, object detection method for crack detection and segmentation method for crack detection.

Among them, the methods based on machine learning include two categories, object detection method for crack detection and segmentation method for crack detection. The former detects regions containing cracks, and the latter segments crack contours, including semantic segmentation and instance segmentation, as shown as the right part in Figure 2. YOLOv3-Lite [26] uses the deep separable convolution to extract features, and utilizes the feature pyramid to preserve semantic information at different levels. A crack detection method based on the YOLOv4 algorithm is proposed in [34], which achieves good crack detection results with a lower trained model weight. To overcome the complicated and uneconomical disadvantages of traditional crack detection methods, a pavement crack detection network [35] is proposed to combine YOLOv5 and Transformer. Zhou et al. [36] propose a novel network architecture with richer feature fusion and attention mechanism and mixed pooling module for crack detection. Qu et al. [37] propose a deeply supervised convolutional neural network for crack detection via a novel multiscale convolutional

feature fusion module. A more fine-grained method is utilized in [38], where raw images are cropped into smaller images, and cracks are detected with a trained CNN classifier and an exhaustive search with a sliding window. U-Net [13] utilizes Convolutional Neural Networks (CNNs) to design encoder and decoder forming a 'U'-shaped net and detect a crack in form of segmentation. Based on U-Net [13], many excellent methods have been proposed [12,14,15]. For example, Liu et al. [14] utilize an encoder–decoder architecture to learn hierarchical features of cracks in multiple scenes and scales effectively for crack detection. CrackU-net [12] uses a 'U'-shaped model architecture to achieve crack detection, including convolution, pooling, transpose convolution, and concatenation operations in it. Liu et al. [15] propose a two-step pavement crack detection and segmentation method based on modified U-Net, in which a residual neural network (ResNet-34) pre-trained by ImageNet [39] is used as the encoder and convolution layers as the decoder. Dense Attention U-Net [40] proposes a encoder with multi-stage dense blocks to improve its capability for extracting informative contextual features. In this paper, we mainly focus on the segmentation method for crack detection.

2.2. Semantic Segmentation for Crack Detection

Semantic segmentation is a computer vision task, which performs binary classification for each pixel according to its semantics: '0' for the background and '1' for the foreground [5]. Generally speaking, the segmentation network designs a feature extraction network to obtain a feature map which is the same size as the original image, and performs a class prediction operation on each pixel. To enrich the channel information, down-sampling and up-sampling are chosen to form a feature extraction network. With the development of Convolutional Neural Networks (CNN), the down-sampling and up-sampling parts are replaced by various convolutional networks, called encoder and decoder. In recent years, transformers [17], originally used for Natural Language Processing (NLP), have set off a boom in the field of Computer Vision (CV), and more and more methods use a transformer [17] to complete segmentation tasks.

The object detection method for crack detection can only achieve the classification and rough location of cracks. More intuitive and accurate detection results are obtained by pixel-level crack detection [41]. There are three major types of approaches in the field of Semantic segmentation for Crack Detection, namely thresholding-based, edge-based, and data driven-based methods [42]. The first two are rule-driven segmentation methods. In this paper, we mainly focus on data-driven segmentation methods using neural networks. The fully connected segmentation method is popular with many researchers [41,43]. Dung et al. [43] propose a crack detection method based on deep Fully Convolutional Network (FCN). To solve time-consuming and labor-consuming problems, Yang et al. [41] propose a Fully Convolutional Network (FCN) with multiple steps to realize automatic pixel-level Crack Detection and Measurement. A modified FCN architecture is proposed in [44] to provide pixel-level detection of multiple damages. The U-Net [13] network expresses the encoding and decoding with a 'U'-shape and becomes the basis of many works [45–48], in which Convolutional Neural Networks (CNNs) make a good effect. As transformer [17] is widely used in the field of Computer Vision (CV), many works [21] also use transformer for crack segmentation, in which self-attention block and scaling-attention block are utilized for fine-grained crack detection.

Unlike the above methods, Convolutional Neural Network (CNN) and Transformer are both used in TransMF to jointly coordinate feature learning from both global and local perspectives, and to predict cracks by integrating features of different scales, in which long-range dependencies can be grasped and the impact of noise is minimized as much as possible.

3. Methodology

3.1. Problem Definition

Generally speaking, crack detection is a kind of image segmentation task, which applies image segmentation to the scene of detecting cracks on objects. Therefore, the

dataset and evaluation metrics of crack detection are basically the same as the requirements of image segmentation. Given an image $\mathbf{I} \in \mathbb{R}^{W \times H \times C}$, its label image is $\mathbf{L} \in \mathbb{R}^{W \times H}$, which is a binary image, and each pixel of the image belongs to a category, where W is image width, H is image height and C is the channel of the image. In this paper, the total number of categories is m. The trained model is used to predict the class of each pixel of the image and statistically evaluated using an evaluation metric.

3.2. Overall Framework

In this paper, we propose a novel Transformer-based Multi-scale Fusion Model (TransMF) to detect cracks on objects by designing an Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM).

- **Encoder Module (EM)** : The Encoder Module (EM) proposed in this paper is stacked by N Conv-Block and one Swin-Trans-Encoder Block. Each layer of Conv-Block has different image scales. By extracting deep features of images with convolution, and extracting feature relations with Swin-Trans-Encoder Block, the features with deep semantic relations are obtained. In this way, we can model the long-range dependencies of different regions in the crack image from a local and global perspective.
- **Decoder Module (DM)**: The Encoder Module (EM) and Decoder Module (DM) in TransMF proposed in this paper are symmetric, that is to say, the Decoder Module (DM) is also composed of Swin-Trans-Encoder Block and Conv-Block, and is consistent with the parameter Settings of the Encoder Module (EM). It is worth noting that the feature dimension of the Encoder Module (EM) decreases layer-by-layer, while the dimension of the Decoder Module (DM) increases layer-by-layer. The output of each layer of the Encoder Module (EM) and Decoder Module (DM) are jointly input to the Fusion Module (FM).
- **Fusion Module (FM)**: In order to achieve the representation fusing both low-level features and deep features, in this paper, we design a Fusion Module (FM), which fuses the outputs of encoder and decoder where images are different scales. To fuse those multiple scales feature maps, we concatenate the fusion features of different levels together to form the comprehensive representation, which can mitigate the effects of background noise for crack detection.

In this section, we will introduce our Transformer-based Multi-scale Fusion Model (TransMF) in detail.

3.3. Encoder Module (EM)

To model the long-range dependencies relation of different parts in the crack image from a local and global perspective, we propose a method which is a hybrid of convolution and Transformer, to explore those relationships of various parts in the crack, in which Conv-Block and Swin-Trans-Encoder Block are proposed.

As mentioned as Section 3.1, the input of our model are image label pair: $\{\mathbf{I}, \mathbf{L}\}$, $\mathbf{I} \in \mathbb{R}^{W \times H \times C}$, $\mathbf{L} \in \mathbb{R}^{W \times H}$. Encoder Module (EM) is consists of Conv-Block and Swin-Trans-Encoder Block, the output feature is f_{en}.

Conv-Block: In order to alleviate the influence of noise and obtain the local feature from a local perspective, in this paper, we design Conv-Block to obtain multi-scale feature maps. In Encoder Module (EM), Nx Conv-Block is used to extract the features of crack image $\mathbf{I}$ forming different scale feature maps. The structure of Conv-Block is shown in Figure 3a: each convolution operation is followed by a RELU activation function called Conv-RELU Block, in which the size of the convolution kernel is 3×3. After Mx convolutions, the max-pooling feature is sent into the next Conv-Block. The i-th output feature of Conv-Block is $I^i_{en_{conv}}, i \in [1, N]$ as Equation (1)

$$I^{i+1}_{en_{conv}} = \text{MaxPooling}(\underbrace{RELU(Conv(I^i_{en_{conv}}))}_{M\times}) \tag{1}$$

where $I_{en_{conv}^{i}} \in \mathbb{R}^{W^i \times H^i \times C^i}$ and $I_{en_{conv}}^{1} = \mathbf{I}$.

Swin-Trans-Encoder Block: In order to model the long-range dependencies relation of different crack regions from a global perspective, in this paper, we divide the output feature map of the last Conv-Block and design a Swin-Trans-Encoder Block to explore this relationship. The structure is shown in Figure 3c, in which ST block [19] is shown as (e).

The Swin-Trans-Encoder Block is composed of a Patch-Embedding layer and two ST blocks (Swin Transformer Block). Through the Patch Embedding operation, we split the feature map into 4×4 patches following Swin Transformer [19] and embed the feature getting $I_{em} \in \mathbb{R}^{\frac{W^N}{4} \times \frac{H^N}{4} \times C^N}$. However, we only use two ST blocks to encode these patch features. The ST block is calculated as Equation (2)

$$\begin{cases} \hat{z}^l = \text{W-MSA}(\text{LN}(z^{l-1})) + z^{l-1} \\ z^l = \text{MLP}(\text{LN}(\hat{z}^l)) + \hat{z}^l \\ \hat{z}^{l+1} = \text{SW-MSA}(\text{LN}(z^l)) + z^l \\ z^{l+1} = \text{MLP}(\text{LN}(\hat{z}^{l+1})) + \hat{z}^{l+1} \end{cases} \tag{2}$$

where $\hat{z}^l$ is the output for (S)W-MSA and z^l for MLP.

In summary, the Swin-Trans-Encoder Block is described by Equation (4).

$$f_{en_{st}} = \underbrace{ST\_Block}_{4\times}(PosEmbed(I_{en_{conv}}^{N})) \tag{3}$$

where $I_{en_{conv}}^{N}$ is the output feature of the N-th Conv-Block.

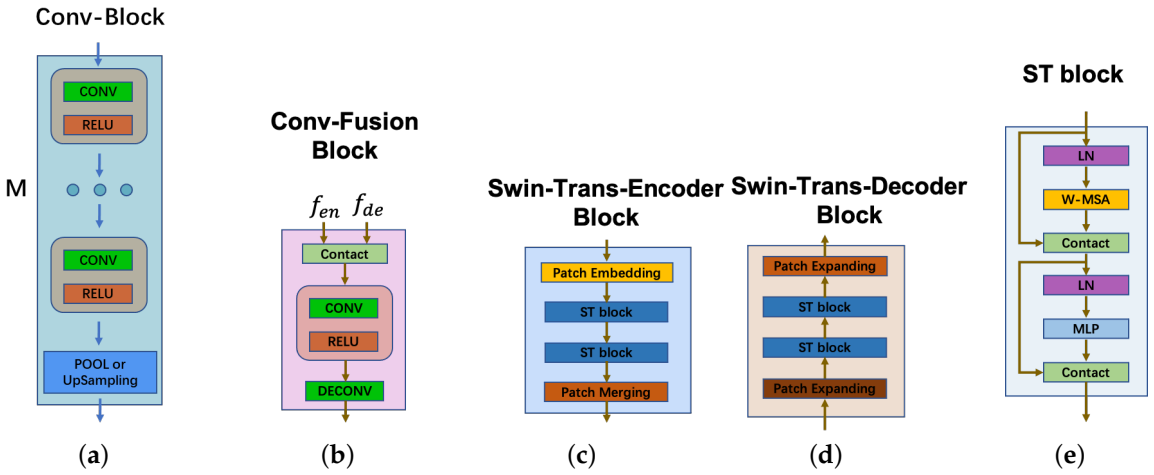


Figure 3. Subfigure (**a**) is Conv-Block, which consists of M convolution-activation blocks. The output in Encoder Module (EM) is the pooled feature, but the output in Decoder Module (DM) is the upsampling feature. Subfigure (**b**) is Conv-Fusion Block, which fuses the encoding feature and the decoding feature by concatenating, and then uses convolution fusing deeply. Subfigure (**c**) is Swin-Trans-Encoder Block, which uses Patch Embedding for image slicing and positional encoding, and uses two swin transformer blocks to extract features containing relationships, and uses Patch merging to merge image slices. Subfigure (**d**) is the Swin-Trans-Decoder Block, which uses the same number of swin transformer blocks as the Swin-Trans-Encoder Block. Subfigure (**e**) is the architecture of the swin transformer block.

3.4. Decoder Module (DM)

To decode the features from Encoder Module (EM), we design the Decoder Module (DM) symmetrically, including Swin-Trans-Decoder Block and Conv-Block, where the layer configuration of Conv-Block is symmetric to that in the Encoder Module (EM). Then we obtain the feature f_{de}.

Conv-Block: For details, refer to the explanation of Conv-Block in Encoder Module (EM). It should be noted that the pooling operation is performed when encoding, and the up-sampling operation is performed when decoding.

Swin-Trans-Decoder Block: The Swin-Trans-Decoder Block consists of two ST blocks and Patch Expanding, where the ST block is calculated as Equation (2). It is worth noting that the dimensions of the ST block in the Swin-Trans-Decoder Block and Swin-Trans-Encoder Block are the same.

For up-sampling, we design Patch Expanding through which we obtain the Swin-Trans-Decoder feature. The specific implementation is to use linear layers and normalization. After the Patch Expanding operation, the feature dimension is $W^{'} \times H^{'} \times C^{'}$.

$$f_{de_{st}} = \text{PatchExpanding}(\underbrace{ST\_Block}_{4\times}(f_{en_{st}})) \tag{4}$$

Then, the output feature $I^i_{de}, i \in [0, N]$ of Swin-Trans-Decoder Block is sent to the stacked Conv-Block layers and the i-th layer is calculated as Equation (5).

$$I^i_{de_{conv}} = \text{UpSampling}(\underbrace{RELU(Conv(I^{i-1}_{de_{conv}}))}_{M\times}) \tag{5}$$

where $I^1_{de_{conv}} = f_{de_{st}}$

3.5. Fusion Module (FM)

In order to better fuse the encoding features and decoding features of different scales, in this paper, we design a Fusion Module (FM), as shown as Figure 3b. First, the concatenated features of encoding and decoding of each scale from different layers are fused in the form of 1×1 convolution, and deconv is as up-sampling to obtain the same scale feature map. Finally, the convolutional fusion features at different scales are concatenated to obtain the final feature as described in Figure 1.

Given the encoding feature f^i_{en} and decoding feature f^i_{de} of the i-th layer, the fusion feature I^i_{fusion} is calculated as Equation (6).

$$I^i_{fusion} = \text{Deconv}(\text{RELU}(\text{Conv}(\text{Concat}(f^i_{en}, f^i_{de})))) \tag{6}$$

where $i \in [1, N+1]$. Note that when $i \in [1, N]$, $f_{en} = I^i_{en_{conv}}, f_{de} = I^i_{de_{conv}}$, and when $i = N+1$, $f_{en} = f_{en_{st}}, f_{de} = f_{de_{st}}$. As shown as Equation (7).

$$f^i_{en} = \begin{cases} I^i_{en_{conv}}, & i \in [1, N] \\ f_{en_{st}}, & i = N+1 \end{cases} \quad f^i_{de} = \begin{cases} I^i_{de_{conv}}, & i \in [1, N] \\ f_{de_{st}}, & i = N+1 \end{cases} \tag{7}$$

Finally, the predicted feature is calculated according to the following Equation (8) referring to Figure 1.

$$I_{fusion} = Concat(I^i_{fusion}), i \in [1, N+1] \tag{8}$$

3.6. Loss Function

Given predicted feature I_{fusion}, we chose Binary Cross Entropy to calculate the loss as Equation (9). Given the number of pixels in an input image, denoted as $M = W \times H \times C$, the value of the j-th pixel on the feature map is F_j, and its label is L_j, the loss is calculated as Equation (9).

$$l(F_j; W) = \begin{cases} log(1 - Sigmoid(F_j; W)), & if \quad L_j = 0 \\ log(Sigmoid(F_j; W)), & if \quad L_j = 1 \end{cases} \tag{9}$$

Then, the final loss is calculated as Equation (10).

$$Loss = \sum_{j=1}^{M}(\sum_{i=1}^{N} l(F_j^i; W) + l(F_j^{fusion}; W)) \tag{10}$$

4. Experiments

Extensive experiments are performed on three public datasets, and the results are compared with the current state-of-the-art baselines. In this section, the experimental results and result analysis will be presented in detail.

4.1. Dataset

To demonstrate the effectiveness and robustness of the method TransMF proposed in this paper, we compare it with the state-of-the-art baselines on three benchmark datasets (CrackLS315 [28], CRKWH100 [28] and DeepCrack [14]). Data augmentation is used in these three datasets all in form of random blur and random color jitter. The details are shown in Table 1.

Table 1. The statistics of two benchmark Datasets .

Split	CrackLS315	CRKWH100
# train	252	80
# test	63	20
# total	315	100

4.1.1. CrackLS315 Dataset

In CrackLS315 [28], 315 asphalt road pavement images are captured under laser illumination with a line-array camera at the same ground sampling distance. The size of each image is 512 by 512 pixels. This dataset is divided into 265 images for train, 10 images for validation and 40 images for test in [28]. In this paper, for simplicity, we randomly shuffle the dataset and divide it into a train set and test set in a ratio of 4:1.

4.1.2. CRKWH100 Dataset

CRKWH100 [28] consists of 100 road pavement images of size 512 $\times$ 512 pixels, all of which are captured by a line-array camera at a ground sampling distance of 1 millimetre under visible-light illumination. In [28], this dataset is used as a validation set, and in this paper, this dataset is divided into a train set and test set according to the same rules as CrackLS315 [28].

4.1.3. DeepCrack Dataset

In DeepCrack [14], a public benchmark dataset with cracks in multi-scale and multi-scene is established, which consists of 537 RGB color images with manually annotated segmentations. The images in this dataset are of a fixed size of 544 $\times$ 384 pixels. In our experiments, we divide it into a train set and test set in a ratio of 4:1 following [28].

4.2. Evaluation Metrics

In order to compare with the current baseline methods quantitatively, in this paper, several evaluation metrics are selected and calculated referring to [14], including Global accuracy, Class average accuracy, Mean intersection over Union, Precision, Recall and F-score.

Given an image I, the label image of which is L. The number of pixel categories is m, and in the background of Crack Detection in this paper, $m = 2$. For the i-th class pixels which are predicted to class j, the number of pixels is denoted as n_{ij} and $i, j \in [0, m-1]$.

4.2.1. Global Accuracy (G)

The percentage of the pixels correctly predicted is measured by Global accuracy (G), which is calculated as following Equation (11)

$$G = \frac{\sum_{i=0}^{m} n_{ii}}{\sum_{i=0}^{m} \sum_{j=0}^{m} n_{ij}} \tag{11}$$

4.2.2. Class Average Accuracy (C)

The predictive accuracy over all classes is called Class average accuracy (C), which is calculated as Equation (12).

$$C = \frac{1}{m} \times \frac{\sum_{i=0}^{m} n_{ii}}{\sum_{j=0}^{m} n_{ij}} \tag{12}$$

4.2.3. Mean Intersection over Union (I/U)

Mean intersection over union (I/U) over all classes is calculated as Equation (13).

$$I/U = \frac{1}{m} \times \frac{\sum_{i=0}^{m} n_{ii}}{\sum_{j=0}^{m} n_{ij} + \sum_{j=0}^{m} n_{ji} - n_{ii}} \tag{13}$$

Intersection-Over-Union is a common evaluation metric for semantic image segmentation. For an individual class, the IOU metric is defined as Equation (14):

$$\text{IOU} = \frac{TP}{TP + FP + FN} \tag{14}$$

Mean intersection over union first computes IOUs for all individual classes, then returns the mean of these values, which is the standard metric of segmentation and widely used in crack detection

4.2.4. Precision (P)

According to the definition of the confusion matrix of machine learning, the Precision (P) is calculated as Equation (15).

$$P = \frac{n_{TP}}{n_{TP} + n_{FP}} \tag{15}$$

where n_{TP} is the number of True Positives, n_{FP} is the number of False Positives.

4.2.5. Recall (R)

According to the definition of the confusion matrix of machine learning, the Recall (R) is calculated as Equation (16).

$$R = \frac{n_{TP}}{n_{TP} + n_{FN}} \tag{16}$$

where n_{TP} is the number of True Positives, n_{FN} is the number of False Negatives.

4.2.6. F-Score (F)

Given Precision (P) and Recall (R), F-score (F) is calculated as Equation (17).

$$F = \frac{2PR}{P + R} \tag{17}$$

4.3. Baselines

In order to prove the effectiveness of TransMF proposed in this paper, with the above datasets and evaluation metrics, we select several strong baseline methods for comparison, including: HED [49], U-Net [13], SegNet [50], DeepCrack [14]. The details of the baseline methods are as follows:

- HED [49]: HED is an edge-detection algorithm consisting of fully convolutional neural networks and deeply-supervised nets, which can detect edges at a speed of practical relevance.
- U-Net [13]: U-Net consists of a contracting path to capture context and a symmetric expanding path that enables precise localization, which achieves very good performance due to data augmentation.
- SegNet [50]: SegNet detects cracks using semantic pixel-wise segmentation, which consists of an encoder network, a corresponding decoder network followed by a pixel-wise classification layer, eliminating the need for learning to upsample.
- DeepCrack [14]: DeepCrack predicts pixel-wise crack segmentation in an end-to-end method, which proposes a CNN-based learning method for semantic segmentation using the 'U'-shaped model architecture.
- DAUnet [40]: DAUnet is proposed to use the dense attention with U-Net, and utilizes a encoder with multi-stage dense blocks to improve its capability of extracting informative contextual features.
- MPRA [36]: MPRA uses a novel network architecture with richer feature fusion and attention mechanism and mixed pooling module for crack detection.

To demonstrate the effectiveness of all components in TransMF, several variants are designed, which are introduced in detail in Section 5.2.

4.4. Experimental Setting

We use three datasets, CrackLS315 [28], CRKWH100 [28] and DeepCrack [14], which are often used as test sets. Six evaluation strategies are used to evaluate the prediction effect, including Global accuracy (G), Class average accuracy (C), Mean intersection over Union (*I*/*U*), Precision (*P*), Recall (*R*) and F-score (*F*). The first three are widely used to evaluate semantic segmentation, and the latter three are commonly used for crack detection. A better I/U can highlight the superiority of our method in the field of image segmentation, and a better F1 score can be a convenient comparison in the field of crack detection, because crack detection is not only implemented by the method of image segmentation, but also the method of image detection, which is introduced in related work section.

We implement the network using the PyTorch deep learning framework. The initial value of the learning rate is 1×10^{-3}, which decays every 1000 iterations with a decay rate of 0.1. The momentum is set to 0.9 and withou weight decay. We use Adam as the optimizer and a NVIDIA GeForce GPU for training.

5. Discussion

5.1. Quantitative Results

Detailed results on three datasets are shown in Table 2. In addition, we also draw PR curves to qualitatively compare the performance of different methods, as shown in Figure 4. From which we can obtain the subsequent observations:

(1) As can be seen from Table 2, our proposed TransMF achieves the best results on all metrics except Precision (P) on both CrackLS315 [28] and CRKWH100 [28] datasets. However, the F1 score indicates that our method is the best, and the low Precision indicates that many hard cases in the dataset are still problems to be studied in crack detection. SegNet [50] is better than Unet showing that a simple skip connection cannot fuse feature information of different scales. HED [49] is implemented by a full connection network and contains rich information, but there is still redundant information. Our TransMF is better than DeepCrack [14], indicating that the proposed Transformer-based Multi-scale Fusion Model could grasp long-range relation information from a local and global perspective.

Table 2. The results of comparison among baselines of TransMF on three datasets. The bold means that the best result for each baseline method.

Dataset	Methods	Metrics					
		G	C	I/U	P	R	F1
CrackLS315 [28]	HED	**99.81**	74.08	69.34	69.34	48.22	55.99
	Unet	99.75	72.32	65.42	50.45	44.75	47.43
	SegNet	**99.81**	69.15	66.79	**73.93**	38.34	50.5
	DeepCrack	99.60	73.99	61.51	31.28	48.25	37.95
	DAUnet	99.77	73.91	67.13	55.17	47.91	51.29
	MPRA	99.80	79.72	71.27	60.32	59.53	59.88
	TransMF	**99.81**	**80.78**	**72.24**	61.84	**61.67**	**61.75**
CRKWH100 [28]	HED	99.83	75.13	71.91	77.82	50.30	61.10
	Unet	99.78	72.84	67.54	60.72	45.75	52.18
	SegNet	99.84	73.97	72.40	**87.79**	47.96	62.03
	DeepCrack	99.59	73.97	61.83	32.48	48.22	38.81
	DAUnet	99.77	71.73	66.53	58.56	43.54	49.94
	MPRA	99.83	83.20	75.37	68.5	66.49	67.48
	TransMF	**99.85**	**84.08**	**77.37**	73.76	**68.22**	**70.88**
DeepCrack [14]	HED	98.55	90.54	81.76	75.86	81.97	78.80
	Unet	98.46	88.71	82.12	80.68	78.16	79.40
	SegNet	98.59	88.59	83.15	84.01	77.77	80.77
	DeepCrack	99.17	**94.57**	88.74	86.14	**89.64**	87.85
	TransMF	**99.27**	94.44	**89.78**	**88.89**	89.27	**89.08**

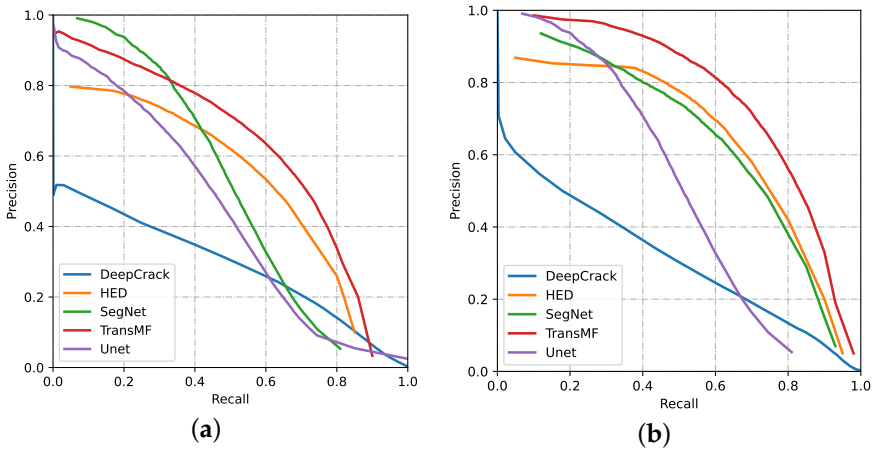


Figure 4. The Precision–Recall (PR) curve for crack detection. (**a**) on CrackLS315 dataset, (**b**) CRKWH100 dataset.

Compared to MPRA, which uses spatial attention and a channel-wise attention for low-level features and high-level features separately, our proposed TransMF utilizes an encoder–decoder structure to extract multi-scale visual features and construct the multi-scale targets sequentially, which can capture both high-level semantics and low-level details for crack detection. Compared to DAUnet, which utilizes a dense block for every encoder layer to extract contextual features, our proposed TransMF integrates a Swin-Transformer to capture long-term relations between all visual regions which can extract the richer contextual information.

(2) It can be seen from the PR curve in Figure 4, that the PR curve of TransMF completely wraps the other curves, showing that TransMF is completely better than other methods on the CRKWH100 [28] dataset. Although it cannot be distinguished from the performance of SegNet and the others on the CrackLS315 [28] dataset, our curve is convex and full, which means our method is better. Precision effectively describes the accuracy of

our positive predictions, i.e., all objects that we predicted in a given image. Recall effectively describes the completeness of our positive predictions relative to the ground truth. However, Precision and Recall can be adjusted by changing the value of the classification threshold. Usually, while the classification threshold increase, the Precision will increase and the Recall will decrease. Therefore, comparing methods via only Precision or Recall is not very meaningful and F-Measure is proposed to combine both Precision and Recall into a single measure that captures both properties. Although the Precision of our method is lower than SegNet and HED, both the Recall and the F1 score are optimal, which can prove the superiority of our method.

In addition, we run all methods on the same server with a GeForce RTX 3090 GPU and and a 2.3 GHz E5-2630 CPU. The results of time costs are reported in Table 3, where FPS means frames per second. As the input images are scaled to the same size on two datasets, the time costs of a specific method do not change on different datasets. While the proposed TransMF achieves significant performance improvements, its FPS score does not decrease a lot compared to baseline methods, which means the additional time costs are affordable.

Table 3. Time costs on two datasets.

Dataset	Method	FPS
CrackLS315	HED	18
	Unet	15
	SegNet	14
	DeepCrack	10
	TransMF	12
CRKWH100	HED	18
	Unet	15
	SegNet	14
	DeepCrack	10
	TransMF	12

5.2. Analysis of TransMF Components

In order to demonstrate the effectiveness of using Transformer and the Multi-scale Fusion model in TransMF, we design several variants for a common comparative study introduced as follows:

- TransMF$\neg f$: A variant of TransMF in which the Fusion Module is removed, and only uses the a Conv-Block in an Encoder Module (EM) and Decoder Module (DM).
- TransMF$\neg st$: A variant of TransMF which the Swin-Trans-Encoder Block and Swin-Trans-Decoder Block are removed in the Encoder Module (EM) and Decoder Module (DM).
- TransMF: the full TransMF.

Several variants of TransMF are performed and analyzed as follows, and experimental results are displayed in Table 4, from which we observe and draw the following conclusions:

(1) *Effects of Fusion Module*: From the comparison results of TransMF and TransMF$\neg f$, it can be seen that the Fusion Module plays a great role in feature representation for crack detection. With the help of the Fusion Module (FM), we obtain good features and the impact of noise can be minimized.

(2) *Effects of Transformer part*: We compare TransMF and TransMF$\neg st$, and the result of TransMF is higher than that of TransMF$\neg st$, illustrating the effectiveness of the Swin Transformer, indicating that TransMF can model the long-range dependencies relations of different parts in a crack image.

Table 4. The results of comparison among different variants of TransMF on CrackLS315 and CRKWH100 datasets. The bold means that the best result.

Dataset	Methods	Metrics					
		G	C	I/U	P	R	F1
CrackLS315 [28]	TransMF$\neg st$	99.76	74.04	66.59	52.16	48.19	50.1
	TransMF$\neg f$	99.75	71.23	65.06	51.48	42.56	46.6
	TransMF	**99.81**	**80.78**	**72.24**	**61.84**	**61.67**	**61.75**
CRKWH100 [28]	TransMF$\neg st$	99.83	82.78	75.32	69.21	65.64	67.38
	TransMF$\neg f$	99.76	71.45	66.13	57.12	42.98	49.05
	TransMF	**99.85**	**84.08**	**77.37**	**73.76**	**68.22**	**70.88**

5.3. Impacts of the ST Block Layers

In our framework, 4-layer ST blocks are used in both Swin-Trans-Encoder Block and Swin-Trans-Decoder Block. We conduct an experimental study on the number of layers of the ST block denoted as c, and the results are shown in Table 5. Of course, the more ST block layers are set, the better the effect will be. However, for the comprehensive time and efficiency, we select c = 4 in this paper.

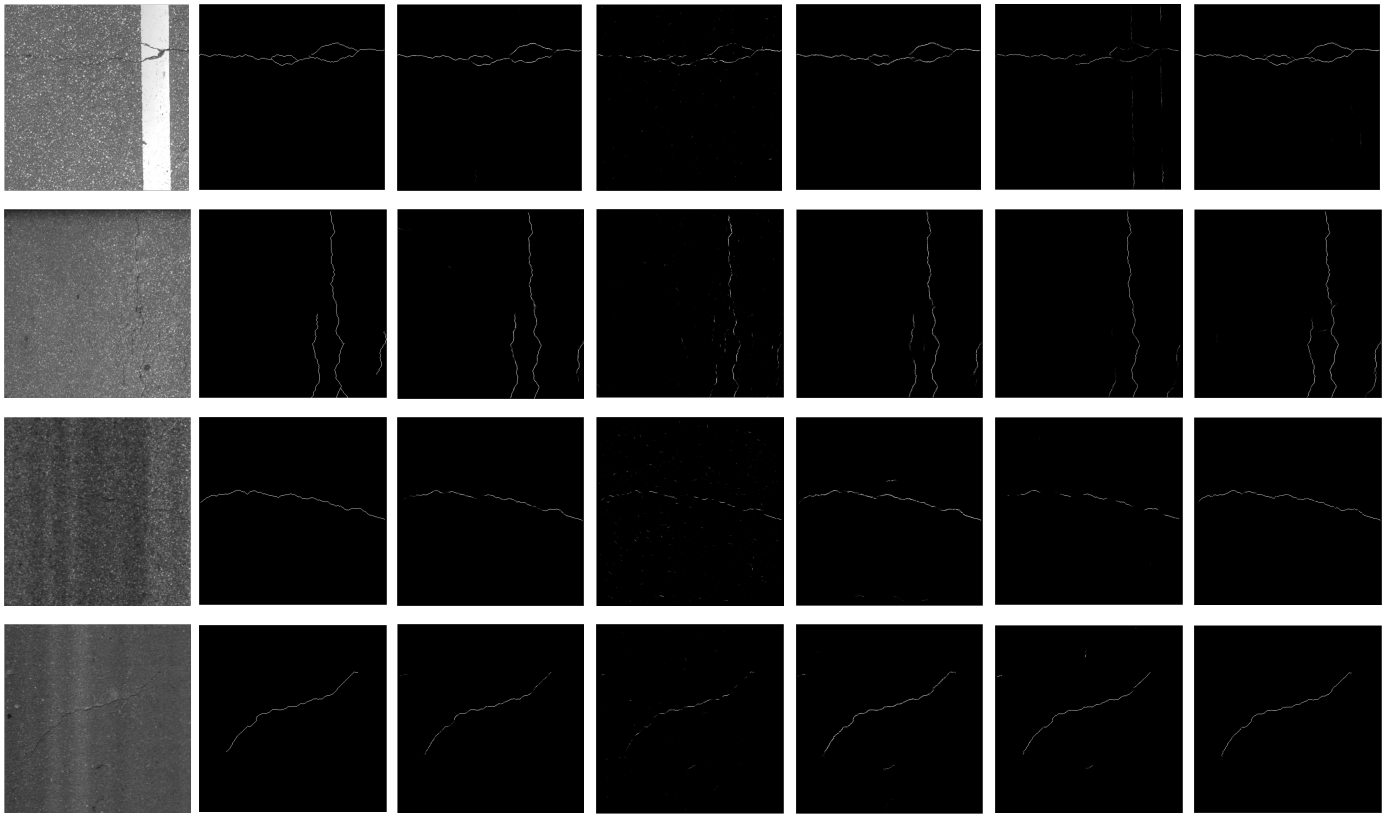
Table 5. The results of the number of the BT block layers on three datasets. The bold means that the best result for different block layers.

Dataset	c	Metrics					
		G	C	I/U	P	R	F1
CrackLS315 [28]	c = 1	99.82	81.83	74.52	68.36	63.74	65.97
	c = 2	99.85	81.02	75.83	75.78	62.09	68.25
	c = 3	99.84	83.09	76.52	72.98	66.25	69.45
	c = 4	**99.85**	**84.08**	**77.37**	**73.76**	**68.22**	**70.88**
	c = 5	99.80	79.29	71.37	61.58	58.67	60.09
CRKWH100 [28]	c = 1	99.78	78.17	69.82	57.56	56.45	57.00
	c = 2	99.8	79.49	71.09	60.01	59.08	59.54
	c = 3	99.79	79.02	70.34	57.96	58.14	58.05
	c = 4	99.81	80.78	72.24	61.84	61.67	61.75
	c = 5	**99.84**	**81.13**	**75.21**	**72.84**	**62.33**	**67.18**

5.4. Case Study

We picked a few images, visualized the predicted pictures, and compared different methods. As shown in the Table 6, it can be seen that U-Net [13] and DeepCrack [14] are poor for continuity detection of cracks and cannot capture long dependencies. SegNet [50] and HED [49] are better at capturing continuity, but still not as good as our proposed TransMF, which shows that long-range dependencies relationships are grasped by TransMF. As can be seen from the 4th image in Table 6, there is no noise crack in the image predicted by TransMF but exists in the image predicted by SegNet, which shows the robustness to noise of TransMF.

Table 6. Visualization of original and predicted segmentation images by all the methods.

Original Image	Ground Truth	HED	Unet	SegNet	DeepCrack	TransMF
						

6. Conclusions

In this paper, we propose a novel crack detection method called *Transformer-based Multi-scale Fusion Model* (TransMF), which detects a crack in form of semantic segmentation. The framework of TransMF includes an Encoder Module (EM), Decoder Module (DM) and Fusion Module (FM), in which the Encoder Module and Decoder Module use multiple convolution blocks and a Swin Transformer block to model the long-range dependencies of different parts in a crack image from a local and global perspective for a better crack image understanding. The output of the Encoder Module and the output of the Decoder Module at different scales are fused in the form of Convolution in the Fusion Module. The output of each layer of the Fusion Module is spliced to alleviate achieve the effect of background noise for the purpose of crack detection. Extensive experiments on three benchmark datasets (CrackLS315, CRKWH100 and DeepCrack) demonstrate that the proposed TransMF in this paper exceeds the best performance at present.

Author Contributions: Conceptualization, X.J. and S.Q.; methodology, X.J.; validation, X.J. and X.Z.; writing—original draft, X.J.; writing—review and editing, X.Z. and S.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation (NSF) of China (No. U1934209).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository. The data presented in this study are openly available at https://1drv.ms/f/s!AittnGm6vRKLtylBkxVXw5arGn6R (accessed on 1 July 2022).

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

Abbreviations

Abbreviation	Extension
EM	Encoder Module
DM	Decoder Module
FM	Fusion Module
DL	Deep Learning
AI	Artificial Intelligence
FR	Face Recognition
VLPR	Vehicle License Plate Recognition
CNNs	Convolutional Neural Networks
NLP	Natural Language Processing
CV	Computer Vision
FCN	Fully Convolutional Network
ST	Swin Transformer
G	Global accuracy
C	Class average accuracy
I/U	Mean intersection over Union
P	Precision
R	Recall
F	F-score
PR	Precision-Recall

References

1. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1701–1708.
2. Silva, S.M.; Jung, C.R. License Plate Detection and Recognition in Unconstrained Scenarios. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018; Lecture Notes in Computer Science; Volume 11216, pp. 593–609.
3. Li, T.; Wang, Y.; Hong, R.; Wang, M.; Wu, X. pDisVPL: Probabilistic Discriminative Visual Part Learning for Image Classification. *IEEE Multim.* **2018**, *25*, 34–45. [CrossRef]
4. Li, T.; Meng, Z.; Ni, B.; Shen, J.; Wang, M. Robust geometric ℓ_p-norm feature pooling for image classification and action recognition. *Image Vis. Comput.* **2016**, *55*, 64–76. [CrossRef]
5. Li, T.; Ni, B.; Wu, X.; Gao, Q.; Li, Q.; Sun, D. On random hyper-class random forest for visual classification. *Neurocomputing* **2016**, *172*, 281–289. [CrossRef]
6. Abdel-Qader, I.; Pashaie-Rad, S.; Abudayyeh, O.; Yehia, S. PCA-Based algorithm for unsupervised bridge crack detection. *Adv. Eng. Softw.* **2006**, *37*, 771–778. [CrossRef]
7. Xu, H.; Su, X.; Wang, Y.; Cai, H.; Cui, K.; Chen, X. Automatic Bridge Crack Detection Using a Convolutional Neural Network. *Appl. Sci.* **2019**, *9*, 2867. [CrossRef]
8. Tong, X.; Guo, J.; Ling, Y.; Yin, Z. A new image-based method for concrete bridge bottom crack detection. In Proceedings of the 2011 International Conference on Image Analysis and Signal Processing, Wuhan, China, 21–23 October 2011, pp. 568–571. [CrossRef]
9. Liu, H.; Yang, C.; Li, A.; Ge, Y.; Huang, S.; Feng, X.; Ruan, Z. Deep Domain Adaptation for Pavement Crack Detection. *arXiv* **2021**, arXiv:2111.10101.
10. Zhang, K.; Zhang, Y.; Cheng, H. CrackGAN: Pavement Crack Detection Using Partially Accurate Ground Truths Based on Generative Adversarial Learning. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1306–1319. [CrossRef]
11. Cao, W.; Liu, Q.; He, Z. Review of Pavement Defect Detection Methods. *IEEE Access* **2020**, *8*, 14531–14544. [CrossRef]
12. Huyan, J.; Li, W.; Tighe, S.; Xu, Z.; Zhai, J. CrackU-net: A novel deep convolutional neural network for pixelwise pavement crack detection. *Struct. Control Health Monit.* **2020**, *27*, e2551. [CrossRef]
13. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; Lecture Notes in Computer Science; Volume 9351, pp. 234–241.
14. Liu, Y.; Yao, J.; Lu, X.; Xie, R.; Li, L. DeepCrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing* **2019**, *338*, 139–153. [CrossRef]
15. Liu, J.; Yang, X.; Lau, S.; Wang, X.; Luo, S.; Lee, V.C.; Ding, L. Automated pavement crack detection and segmentation based on two-step convolutional neural network. *Comput. Aided Civ. Infrastruct. Eng.* **2020**, *35*, 1291–1305. [CrossRef]

16. Mohan, A.; Poobal, S. Crack detection using image processing: A critical review and analysis. *Alex. Eng. J.* **2017**, *57*, 787–798. [CrossRef]
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
18. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2021**, arXiv:2010.11929.
19. 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, Montreal, BC, Canada, 11–17 October 2021; pp. 9992–10002.
20. Guo, Q.; Qiu, X.; Liu, P.; Shao, Y.; Xue, X.; Zhang, Z. Star-Transformer. NAACL-HLT (1); Association for Computational Linguistics: Minneapolis, MN, USA, 7 June 2019; pp. 1315–1325. Available online: https://aclanthology.org/N19-1133/ (accessed on 2 June 2019)
21. Liu, H.; Miao, X.; Mertz, C.; Xu, C.; Kong, H. CrackFormer: Transformer Network for Fine-Grained Crack Detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021; pp. 3783–3792.
22. Kong, J.; Bian, Y.; Jiang, M. MTT: Multi-Scale Temporal Transformer for Skeleton-Based Action Recognition. *IEEE Signal Process. Lett.* **2022**, *29*, 528–532. [CrossRef]
23. Xiao, Y.; Yuan, Q.; He, J.; Zhang, Q.; Sun, J.; Su, X.; Wu, J.; Zhang, L. Space-time super-resolution for satellite video: A joint framework based on multi-scale spatial-temporal transformer. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *108*, 102731. [CrossRef]
24. Yuan, W.; Xu, W. MSST-Net: A Multi-Scale Adaptive Network for Building Extraction from Remote Sensing Images Based on Swin Transformer. *Remote. Sens.* **2021**, *13*, 4743. [CrossRef]
25. Jiaji, W.; Liu, Y.F.; Nie, X.; Mo, Y. Deep convolutional neural networks for semantic segmentation of cracks. *Struct. Control. Health Monit.* **2021**, *29*, e2850. [CrossRef]
26. Li, Y.; Han, Z.; Xu, H.; Liu, L.; Li, X.; Zhang, K. YOLOv3-Lite: A Lightweight Crack Detection Network for Aircraft Structure Based on Depthwise Separable Convolutions. *Appl. Sci.* **2019**, *9*, 3781. [CrossRef]
27. Song, W.; Jia, G.; Zhu, H.; Gao, L. Automated Pavement Crack Damage Detection Using Deep Multiscale Convolutional Features. *J. Adv. Transp.* **2020**, *2020*, 1–11. [CrossRef]
28. Zou, Q.; Zhang, Z.; Li, Q.; Qi, X.; Wang, Q.; Wang, S. DeepCrack: Learning Hierarchical Convolutional Features for Crack Detection. *IEEE Trans. Image Process.* **2019**, *28*, 1498–1512. [CrossRef]
29. Mitchell, T.M. *Machine Learning*; McGraw-Hill: New York, NY, USA, 1997.
30. Schulz, H.; Behnke, S. Deep Learning. *KI-Künstliche Intell.* **2012**, *26*. [CrossRef]
31. Valueva, M.; Nagornov, N.; Lyakhov, P.; Valuev, G.; Chervyakov, N. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Math. Comput. Simul.* **2020**, *177*, 232–243. [CrossRef]
32. Chen, H.; Lin, H. An Effective Hybrid Atrous Convolutional Network for Pixel-Level Crack Detection. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–12. [CrossRef]
33. Munawar, H.S.; Hammad, A.; Haddad, A.; Soares, C.; Waller, S. Image-Based Crack Detection Methods: A Review. *Infrastructures* **2021**, *6*, 115. [CrossRef]
34. Yao, G.; Sun, Y.; Yang, Y.; Liao, G. Lightweight Neural Network for Real-Time Crack Detection on Concrete Surface in Fog. *Front. Mater.* **2021**, *8*, 517. [CrossRef]
35. Xiang, X.; Wang, Z.; Qiao, Y. An Improved YOLOv5 Crack Detection Method Combined with Transformer. *IEEE Sens. J.* **2022**. [CrossRef]
36. Zhou, Q.; Qu, Z.; Cao, C. Mixed pooling and richer attention feature fusion for crack detection. *Pattern Recognit. Lett.* **2021**, *145*, 96–102. [CrossRef]
37. Qu, Z.; Cao, C.; Liu, L.; Zhou, D.Y. A deeply supervised convolutional neural network for pavement crack detection with multiscale feature fusion. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**. [CrossRef]
38. Li, S.; Zhao, X. Image-Based Concrete Crack Detection Using Convolutional Neural Network and Exhaustive Search Technique. *Adv. Civil Eng.* **2019**, *2019*, 1–12. [CrossRef]
39. Deng, J.; Dong, W.; Socher, R.; Li, L.; Li, K.; Fei-Fei, L. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 248–255.
40. Hsiel, Y.A.; Tsai, Y.C.J. Dau-net: Dense attention u-net for pavement crack segmentation. In Proceedings of the 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), Indianapolis, IN, USA, 19–22 September 2021; pp. 2251–2256.
41. Yang, X.; Li, H.; Yu, Y.; Luo, X.; Huang, T.; Yang, X. Automatic Pixel-Level Crack Detection and Measurement Using Fully Convolutional Network. *Comput.-Aided Civ. Infrastruct. Eng.* **2018**, *33*, 1090–1109. [CrossRef]
42. Kheradmandi, N.; Mehranfar, V. A critical review and comparative study on image segmentation-based techniques for pavement crack detection. *Constr. Build. Mater.* **2022**, *321*, 126162. [CrossRef]
43. Dung, C.V.; Anh, L.D. Autonomous concrete crack detection using deep fully convolutional neural network. *Autom. Constr.* **2019**, *99*, 52–58. [CrossRef]
44. Li, S.; Zhao, X.; Zhou, G. Automatic pixel-level multiple damage detection of concrete structure using fully convolutional network. *Comput.-Aided Civ. Infrastruct. Eng.* **2019**, *34*, 616–634. [CrossRef]

45. Zhang, L.; Shen, J.; Zhu, B. A research on an improved Unet-based concrete crack detection algorithm. *Struct. Health Monit.* **2020**, *20*, 1864–1879. [CrossRef]
46. Shokri, P.; Shahbazi, M.; Lichti, D.; Nielsen, J. VISION-BASED APPROACHES FOR QUANTIFYING CRACKS IN CONCRETE STRUCTURES. *ISPRS-Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2020**, *XLIII-B2-2020*, 1167–1174. [CrossRef]
47. Fang, J.; Qu, B.; Yuan, Y. Distribution equalization learning mechanism for road crack detection. *Neurocomputing* **2021**, *424*, 193–204. [CrossRef]
48. Kanaeva, I.; Ivanova, J. Road pavement crack detection using deep learning with synthetic data. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1019*, 012036. [CrossRef]
49. Xie, S.; Tu, Z. Holistically-Nested Edge Detection. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1395–1403.
50. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]