Hindawi
Advances in Materials Science and Engineering
Volume 2021, Article ID 5586615, 11 pages
https://doi.org/10.1155/2021/5586615

Research Article

Pixel-Level Recognition of Pavement Distresses Based on U-Net

Deru Li,[1] **Zhongdong Duan**,[1] **Xiaoyang Hu**,[2] **and Dongchang Zhang**[2]

[1]*School of Civil and Environmental Engineering, Harbin Institute of Technology, Shenzhen 518055, China*
[2]*China Merchants Roadway Information Technology (Chongqing) Co., Ltd., Chongqing 400067, China*

Correspondence should be addressed to Zhongdong Duan; duanzd@hit.edu.cn

Received 28 January 2021; Revised 22 February 2021; Accepted 26 February 2021; Published 15 March 2021

Academic Editor: Yubo Jiao

This study develops and tests an automatic pixel-level image recognition model to reduce the amount of manual labor required to collect data for road maintenance. Firstly, images of six kinds of pavement distresses, namely, transverse cracks, longitudinal cracks, alligator cracks, block cracks, potholes, and patches, are collected from four asphalt highways in three provinces in China to build a labeled pixel-level dataset containing 10,097 images. Secondly, the U-net model, one of the most advanced deep neural networks for image segmentation, is combined with the ResNet neural network as the basic classification network to recognize distressed areas in the images. Data augmentation, batch normalization, momentum, transfer learning, and discriminative learning rates are used to train the model. Thirdly, the trained models are validated on the test dataset, and the results of experiments show the following: if the types of pavement distresses are not distinguished, the pixel accuracy (PA) values of the recognition models using ResNet-34 and ResNet-50 as basic classification networks are 97.336% and 95.772%, respectively, on the validation set. When the types of distresses are distinguished, the PA values of models using the two classification networks are 66.103% and 44.953%, respectively. For the model using ResNet-34, the category pixel accuracy (CPA) and intersection over union (IoU) of the identification of areas with no distress are 99.276% and 99.059%, respectively. For areas featuring distresses in the images, the CPA and IoU of the model are the highest for the identification of patches, at 82.774% and 73.778%, and are the lowest for alligator cracks, at 14.077% and 12.581%, respectively.

1. Introduction

The traditional way of pavements distress evaluation involves manual visual inspection and measurement. This is labor intensive, hinders traffic, and poses risk to workers' safety. It is also inefficient and inaccurate and makes it difficult to objectively assess the pavement condition. To improve the situation, cameras [1] and intelligent vehicles [2] have been used to capture images of the surfaces of pavements to obtain a large amount of data that are then analyzed manually. Such semiautomatic (in terms of image acquisition) and semimanual (in terms of distress identification) methods still require a large amount of labor. To address this issue, researchers have proposed methods for the automatic identification of pavement distresses using image analysis.

Li [3] and Li et al. [4] used an eight-directional Sobel operator and the maximum intercluster variance algorithm to develop an edge detection method suitable for processing images showing damage to pavements. Li et al. [5] proposed a method to detect pavement cracks based on the minimal-cost path search for strong speckle noise and low-contrast and poor continuity of pavement cracks. These image segmentation methods are complex and difficult to achieve rapid batch detection. Lin and Liu [6] used a nonlinear support vector machine (SVM) to identify potholes, and Shen et al. [7] used the SVM to recognize damage on pavement images. Acosta [8] proposed a horizontal and vertical segmentation algorithm, which divides the road damage image pixels into background, foreground, and possible foreground. The road crack image is obtained through the connection of the adjacent foreground and the possible foreground area. Chu et al. [9] used the optimal threshold algorithm to remove noise pixels in the road image and realized image binarization through online learning. Wu et al. [10] used the CCOI algorithm to realize the connection

of the cracks according to the degree of connectivity between the binary image and the surrounding objects. The algorithm improves the accuracy and efficiency of the pavement distress recognition but cannot distinguish among the types of distresses or meets the requirements for real-time detection.

With the rapid development of artificial intelligence, in particular the image classification and object segmentation technology based on the convolutional neural network (CNN) in recent years, researchers have applied computer vision and machine learning technology to achieve automatic detection of pavements distresses [11–16]. Maeda et al. [12] proposed an object classification method based on the CNN. The authors used images captured by a smartphone mounted on a car and classified eight categories of pavement damage with an image dataset containing 9,053 images showing damage to pavements and 15,435 instances of such damage. Zhang et al. [13] proposed a model that contains four convolutional layers, a maximum pooling layer, and two fully connected layers for pavement crack detection. Sha et al. [14] introduced the CNN to the image analysis based on pavement distress recognition and measurement and proposed models to extract crack and pothole features. Their experiments showed that CNN can achieve accurate results for the identification and measurement of pavement cracks and potholes. Shi et al. [15] detected bare surface distress on concrete pavements using deep learning and achieved an accuracy of 90.2%. Tang et al. [16] cut the pavement images into subblocks of 128 pixels × 128 pixels, manually labeled them as crack and noncrack images, and used a variety of deep learning models to identify the subblocks, showing cracks with recognition accuracy of 92%.

Existing methods that use deep neural networks to process images to identify pavement distresses have the following problems: (1) There is a lack of high-quality, large-scale, multidistress pavement datasets for model training. Most of current researches on pavement distress recognition have been on a single type of distress, such as cracks [17–20]. (2) When the deep neural networks are used to analyze images of pavement distress, the images need to be manually divided into subblocks to reduce their size while maintaining high pixel. It helps to improve the distress images recognition accuracy, but it is labor intensive as it is done manually. (3) Advanced technologies for image segmentation, such as the U-net, can significantly improve the efficiency of image detection while maintaining a high accuracy, and it has been used for crack detection in concrete structures [21] and pavements [22–24]. Ji et al. [21] achieved an accuracy of 99.56% on their test set, and Chen et al. [22] obtained an accuracy of 89.92% in their asphalt pavement dataset. However, this model has not been applied to the identification of multiple distresses of pavements.

For multidistress identification of pavement images, a large number of pavement images from four asphalt highways in three provinces and cities in China were collected and labeled on pixel level. Six types of distresses are considered in this study, namely, alligator cracks, longitudinal cracks, transverse cracks, block cracks, potholes, and patches. The U-net model is combined with the ResNet, a deep convolutional neural network (CNN), to train the pavement image recognition model. Techniques such as data augmentation, batch normalization, momentum learning, and other regularization techniques are used to enhance the model training. The developed model does not require manual preprocessing of the pavement images and delivers pixel-level distress location, shape, and size in the images, which could automate the distress detection procedure and improve the distress identification accuracy.

This paper trained a new model to identify the distress area and classify the distress types for multidistress pavement images. In Section 2, a pixel-level dataset of six types of distresses is built. In Section 3, a U-net model combined with the ResNet is introduced. Section 4 demonstrates the training and verification of pixel-level pavement distress recognition model. Section 5 summarizes the outcomes and concludes the paper.

2. Dataset of Pavement Distresses

The images of pavement distress used in this article were collected by the Intelligent Road Measuring Vehicle (Luxin-CT616, see Figure 1), manufactured by the China Merchants Roadway Information Technology (Chongqing) Co., Ltd. The vehicle was equipped with one CCD (charge coupled device) camera at a right angle to cover the pavement surface. A 3,662 × 2,032-pixel pavement image was captured every 2 m along the driving direction. Each image covered a pavement patch of 3.5 m × 2 m at a resolution of 0.96 mm/px × 1 mm/px. The frame rate changed with vehicle speed.

A total of 10,097 pavement images featuring six types of distresses, alligator cracks, longitudinal cracks, transverse cracks, block cracks, potholes, and patches, were collected on four asphalt expressways in three provinces and cities in China (G15 Yueqing section and Ounan section in Zhejiang Province, G243 Meitan-to-Yuqing highway in Guizhou, Chongqing Inner Ring Expressway, and Chongqing Yuchang Expressway. The images were captured over 567.33 km).

Each image might have contained no, one, or more than one type of distress. Each type of distress was labeled at pixel level as shown in Figure 2. A total of 5,427 images contain one type of distress, accounting for 53.75% of all images, and 4,670 images contain two or more types of distresses, accounting for 46.25%.

The 10,097 images of the pavement distress dataset feature 14,697 cases of distress. The number of occurrences of each type of distress is shown in Table 1. Patches accounts for the largest portion of the dataset, and block cracks is the smallest.

3. Deep Neural Network for Pixel-Level Pavement Distress Recognition

In traditional methods of image recognition, such as feature engineering tasks, the extraction of image features and tags needs to be carried out manually. Recent years have witnessed the rapid development of convolutional neural network (CNN) [25] that can automatically extract image features. The deep CNN contains multiple CNNs, which

Figure 1: Intelligent road measuring vehicle (luxin-CT616).

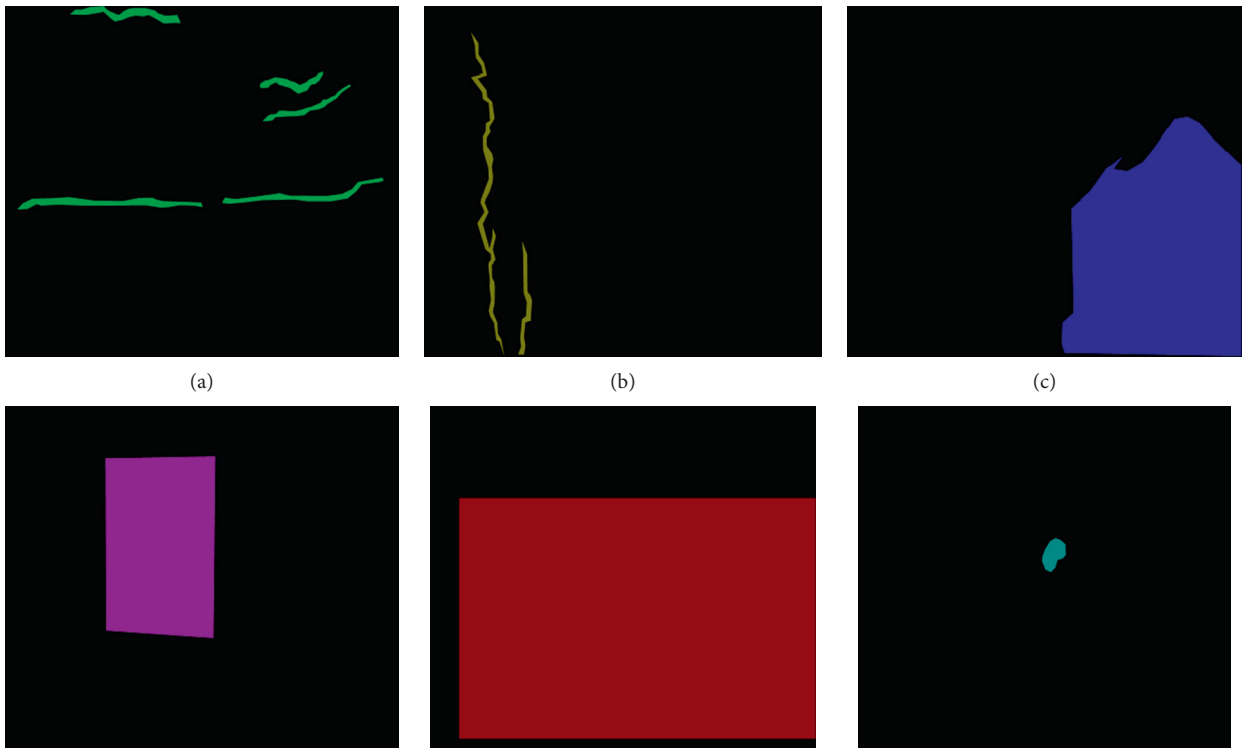

Figure 2: Six types of pavement distress. (a) Transverse crack. (b) Longitudinal crack. (c) Alligator crack. (d) Patch. (e) Block crack. (f) Pothole.

enables the extraction of local and global features from images in one model. Through the automatic feature extraction of images by a low-level CNN, some basic edge-related features, such as points and lines, can be constructed and converted into higher-level features through a high-level CNN for the accurate recognition of images. Therefore, when a deep CNN is used to train a model to detect damage to pavements, the images are input to the model for automatic distress identification without the need of manual feature extraction. To improve the identification accuracy, the semantic segmentation of images is used, and the pavement distresses are identified at pixel level. Specifically, the U-net network is employed as the main algorithmic architecture of the model and ResNet as its basic classification network.

3.1. ResNet. There is a consensus in image recognition that the greater the number of layers used for image recognition with deep CNN networks, the higher the recognition

Table 1: The numbers of all distress types.

Distress	Number	Percent (%)
Transverse crack	1786	12.09
Longitudinal crack	2348	15.90
Alligator crack	1662	11.31
Block crack	126	0.86
Patch	6924	47.11
Pothole	1919	13.06

accuracy. However, when the network reaches a certain depth in image recognition experiments, further increasing network depth degrades its performance [26]. Simply increasing the number of layers does not yield an improvement in recognition performance but reduces the speed of convergence and the classification accuracy of the network.

To solve this problem, He et al. [26] proposed a residual network (namely, ResNet) that uses residual blocks by adding an identity mapping to the usual deep CNN, as shown in Figure 3. Take a deep CNN network with output x for an example. ResNet is to add a residual block as shown in Figure 3 after the last layer of the deep CNN network. The input to the residual block is the output x of the original deep CNN, its learning feature is the output $F(x)$ of the two middle CNN layers, and the final output $H(x)$ of the residual block is $H(x) = F(x) + x$. When $F(x)$ is zero, the residual block is subjected to only identity mapping, and the two convolutional layers in it have no effect on the output of the original depth of the CNN. Thus, the performance of the CNN network does not suffer if the depth of the residual block is increased. When $F(x)$ is not zero, the residual block learns a new feature $F(x)$, and the entire deep CNN network can learn $F(x)$ based on the input feature x, which improves the classification performance. In this paper, ResNet is used as the classification network of the semantic segmentation model for pavement distress identification.

Table 2 shows the network structures of ResNet-34 and ResNet-50. Both contained 16 residual blocks. ResNet-34 has 36 convolution layers, a max pooling layer, two average pooling layers, and a fully connected layer. The total number of parameters is 21,813,570. ResNet-50 has 53 convolution layers, a max pooling layer, two average pooling layers, and a fully connected layer, and its total number of parameters is 30,682,729.

3.2. U-Net Neural Network. The probability of occurrence of different distresses in asphalt pavements varies. For example, a patch has a higher incidence probability than that of block cracks. The imbalance in the dataset can cause the deep CNN algorithm to attend more to the distresses with more instances and ignore those with fewer exemplars when recognizing image features. The consequence is to lower the recognition accuracy of the latter. The U-net model [27] could help solve this problem by a symmetrical U-shaped structure.

ResNet-34 is shown to have high accuracy in image classification, and it is used as the backbone of the U-net network, which is shown in Figure 4. It is a fully convolutional semantic segmentation network that has a symmetrical U-shaped structure containing a compression path and an expansion path. The left side of Figure 4 shows the path of contraction and the right side shows that of expansion. The contraction path consists of a ResNet-34 with repeated applications of residual blocks. Some residual blocks are followed by maximum pooling layers for downsampling, and a skip connection is used to splice the feature map in the expansion path. In the expansion path, each repeated step involves firstly upsampling the feature map and then performing a 2×2 deconvolution to halve the number of feature channels and double the size of the feature map. This feature map is spliced using the corresponding feature map in the contraction path. After stitching, two 3×3 convolutions are performed, and each convolution layer is followed by a ReLU activation function. In the last layer of the network, a 1×1 convolution is used to map each 64-channel feature map to the required number of categories. The blue arrow in Figure 4 indicates the feature splicing operation.

The network structure of U-net based on ResNet-34 is shown in Table 3. It contains 54 convolutional layers, and the total number of parameters is 41,132,518. The U-net network structure based on ResNet-50 is similar, with 71 convolutional layers and 338,306,566 parameters.

Before the pavement distress images are input to the model in the form of a pixel matrix, the value of each pixel in the image is converted. Pixel values of the no-distress areas (the remaining part of the image except the distress) are all set to zero, and those of block cracks, longitudinal cracks, transverse cracks, alligator cracks, potholes, and patches are set to 1, 2, 3, 4, 5, and 6, respectively. Hence, each pixel in the images is an integer value between 0 and 6. The converted image set is then input to the U-net model for training and validation.

4. Level Pavement Distress Recognition Model

4.1. Techniques for Training. To improve the model's pixel-level recognition accuracy of pavement distresses and speed-up training, techniques such as data augmentation [28], batch normalization [28], momentum learning [29], fine-tuning [30], and a discriminative learning rate [31] are used.

Data augmentation [28] enables a limited amount of data to produce value equivalent to larger amounts of data without substantially increasing data. It is an important regularization technique in computer vision which applies geometric transformations (such as flipping, rotation, cropping, deformation, and scaling) and color transformations (such as pertaining to noise, blur, erasure, and filling) to images. Because the pavement distress images are captured vertically and longitudinal cracks and transverse cracks have obvious directionality, geometric transformations such as rotation and deformation cannot be used for data augmentation. In this paper, each image is randomly flipped vertically with a probability of 0.75. Figure 5 compares the longitudinal cracks after flipping with the original image.

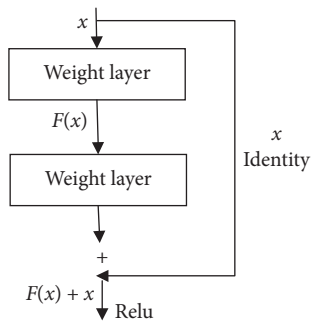

Figure 3: Residual model: a building block.

Table 2: The architecture of ResNet-34 and ResNet-50.

Layer name	34-layer	50-layer
Conv1	$[7\times7, 64]$[a], stride 2	
Max pool	$[3\times3]$, stride 2	
Conv2_x	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
Conv3_x	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$
Conv4_x	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$
Conv5_x	$\begin{bmatrix} 3\times512 \\ 3\times512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
Adaptive average pool	Output_size = 1	
Adaptive max pool	Output_size = 1	
Fully connected layer	Dimensionality = 7	

[a]The number of the convolutional filters is 64, each of size 7×7.

Batch normalization [28] uses normalization and linear transformation to constrain the mean and variance of the input data of each layer to within a certain order of magnitude, which could avoid the mean and variance being too large or too small, so that subsequent layers of the network do not need to adapt to changes in the input to the previous network, and each layer can independently learn the data. This improves the learning speed of the entire neural network. Batch normalization after each convolutional layer is used.

When applying the gradient descent method, if the absolute value of the slope of the objective optimization function in the vertical direction is greater than that in the horizontal direction at a certain position, the gradient descent causes the variable to move to a greater extent along the vertical direction than the horizontal direction at a given learning rate. Therefore, a lower learning rate needs to be set to prevent the independent variable from crossing the optimal solution of the objective function in the vertical direction. However, this causes the independent variable to move slowly in the horizontal direction and prolonged the time needed for convergence. In order to solve this problem, the momentum method uses the weighted average of the gradient of the past time step, where the weight decays exponentially according to the time step, so that the independent variable updates of adjacent time steps are more consistent in direction. Therefore, a higher learning rate is used to enable the independent variable to move more quickly to the optimal solution and accelerate convergence. According to Smith's experiments [29], a cyclical momentum of 0.95–0.85 provides better performance than a constant momentum. Therefore, the momenta used in this paper are (0.95, 0.85).

When recognizing an image, the system first learns low-level features, such as lines, and then learns specific abstract features. Model parameters that have been trained on large image sets are well-learned low-level features. Hence, they can be migrated to other, smaller image sets to speed up the training of the model. This paper uses the transfer learning method [30] to transfer parameters of the ResNet-34 and ResNet-50 models, which have been trained on ImageNet (containing 14,197,122 images in 21,841 categories), to the model for the semantic segmentation of images of pavement distress. The parameters are retrained by fine-tuning and using a discriminative learning rate [31] to obtain values suitable for images of pavement distress. Fine-tuning is used to lock parameters of part of the network layer in the model and to train only the last layer or few layers of parameters; the discriminative learning rate is used to divide the parameters in the model into three parts according to the depth of the CNN model (each part is assigned a different learning rate). The changes in the parameters decrease with decreasing distance to the bottom of the model, where the learning rate is lower. Thus, it becomes easier for the model parameters to reach a stable state, which speeds up the training process.

4.2. Measures of Model Performance. Pixel accuracy (PA), category pixel accuracy (CPA), intersection over union (IoU), and mean intersection over union (MIoU) are commonly used as measures of semantic segmentation. They are defined as follows.

Suppose that there are $k+1$ categories $(0 \sim k)$ in a dataset, and "0" usually represents the background. P_{ii} represents the number of pixels that are originally i type and are predicted to be so, P_{ij} represents the number of pixels that are originally i type but predicted to be j type, and P_{ji} is the number of pixels that are originally j type but predicted to be i type.

PA is defined as the ratio of the number of pixels correctly predicted to the total number of pixels defined in the following:

$$\mathrm{PA} = \frac{\sum_{i=0}^{k} p_{ii}}{\sum_{i=0}^{k}\sum_{j=0}^{k} p_{ij}}. \tag{1}$$

The larger PA is, the greater the number of pixels that the model has predicted correctly is, and the stronger the classification ability is for the model.

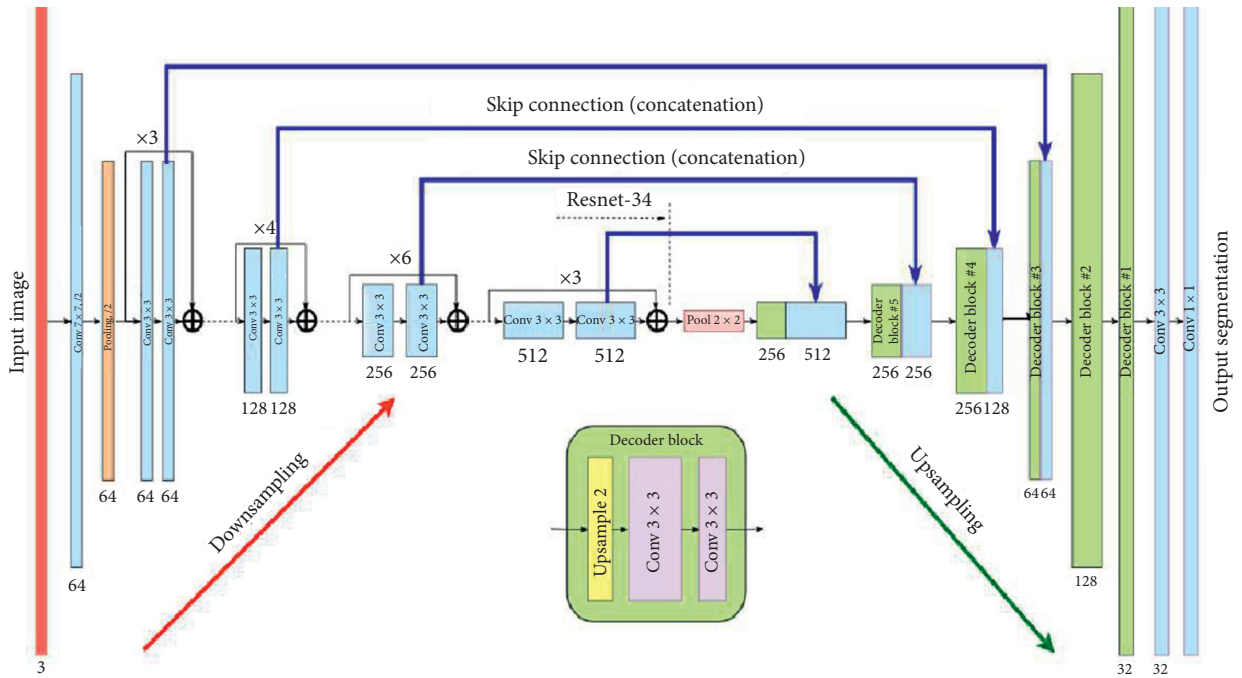

Figure 4: U-net architecture based on ResNet-34.

Table 3: The architecture of U-net based on ResNet-34.

The contraction path	Layers	The expansion path	Layers
Conv2d	$[7\times7,\ 64]$, stride = 2	Pixel shuffle	$[1\times1,\ 1024]$
Max pool	$[3\times3]$, stride = 2	Average pool	$[2\times2]$
Conv2d	$\begin{bmatrix}3\times3,64\\3\times3,64\end{bmatrix}\times3$	Conv2d	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$
Conv2d	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times2$	Pixel shuffle	$[1\times1,\ 512]$
Downsample	$[1\times1,\ 128]$	Average pool	$[2\times2]$
Conv2d	$\begin{bmatrix}3\times3,128\\3\times3,128\end{bmatrix}\times3$	Conv2d	$\begin{bmatrix}3\times3,384\\3\times3,384\end{bmatrix}\times2$
Conv2d	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$	Pixel shuffle	$[1\times1,\ 768]$
Downsample	$[1\times1,\ 256]$, stride = 2	Average pool	$[2\times2]$
Conv2d	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times5$	Conv2d	$\begin{bmatrix}3\times3,256\\3\times3,256\end{bmatrix}\times2$
Conv2d	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	Pixel shuffle	$[1\times1,\ 512]$
Downsample	$[1\times1,\ 512]$, stride = 2	Average pool	$[2\times2]$
Conv2d	$\begin{bmatrix}3\times3,512\\3\times3,512\end{bmatrix}\times2$	Conv2d	$\begin{bmatrix}3\times3,96\\3\times3,96\end{bmatrix}\times2$
Conv2d	$[3\times3,\ 1024]$	Pixel shuffle	$[1\times1,\ 384]$
Conv2d	$[3\times3,\ 512\ ]$	Average pool Conv2d Conv2d Conv2d	$[2\times2]$ $[3\times3,\ 49]$ $[3\times3,\ 99]$ $[1\times1,\ 7\ ]$

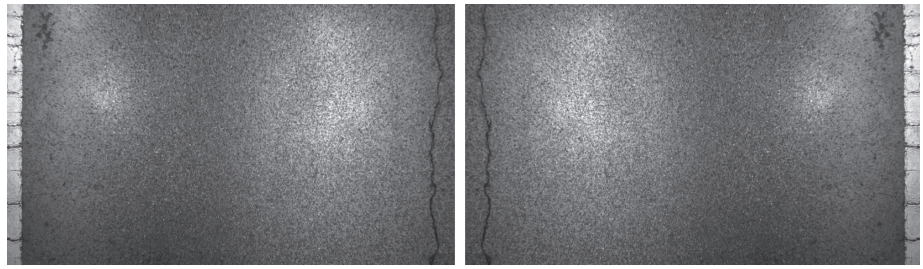

Figure 5: Images of longitudinal crack using flip. (a) Original image. (b) Flipped image.

Table 4: PA values of different models using images of different sizes.

Model	Input image size (pixel)	PA	
		Detecting distress with six types	Detecting with or without distress
ResNet-34	256×256	**0.66103**	**0.97336**
	512×512	0.62716	0.97113
ResNet-50	256×256	0.44953	0.95772

CPA is defined as the ratio of pixels predicted to be correct in one category defined in the following:

$$\text{CPA} = \frac{p_{ii}}{\sum_{j=0}^{k} p_{ji}}. \tag{2}$$

The larger CPA is, the less likely it is that the model predicts pixels of other categories as belonging to the given category, and the stronger the model's classification ability is.

IoU is defined as the ratio of the intersection and union of the ground truth and the predicted value of one category, as defined in the following:

$$\text{IoU} = \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}}. \tag{3}$$

The closer the IoU is to one, the more consistent the model's predicted position for the category is with the true position, and the stronger the model's ability is to locate the category.

MIoU is the average of the IoU of all classes defined by

$$\text{MIoU} = \frac{1}{k+1} \sum_{i=0}^{k} \frac{p_{ii}}{\sum_{j=0}^{k} p_{ij} + \sum_{j=0}^{k} p_{ji} - p_{ii}}. \tag{4}$$

The closer the MIoU is to one, the stronger the model's ability to locate all classes is.

4.3. Analysis of Results

4.3.1. Training Process. A total of 80% of images in the dataset are used to train the models and the remaining 20% are used for validation. Cross-entropy loss is used as loss function for the training process. The image sizes are set to 256×256-pixel and 512×512-pixel, respectively. Using the above training techniques, the pixel-level recognition model of pavement distresses is trained based on ResNet-34 and ResNet-50. All code is implemented on Fast.ai with a PyTorch backbone on an NVIDIA RTX 2080Ti GPU. The training of the model includes the three following steps.

Table 5: CPA values of different labels using ResNet-34.

Distress	CPA
No distress	0.99276
Patch	0.82774
Pothole	0.31902
Transverse crack	0.19139
Block crack	0.25135
Alligator crack	0.14077
Longitudinal crack	0.27439

Table 6: IoU of different labels using ResNet-34.

Distress	IoU
No distress	0.99059
Patch	0.73778
Pothole	0.28436
Transverse crack	0.21429
Block crack	0.19445
Alligator crack	0.23038
Longitudinal crack	0.12581
Average (MIoU)	0.3968

The first step is to randomly flip the pavement images for data augmentation and set the original image size to 256×256-pixel and 512×512-pixel, respectively.

The second step is to load the U-net model with ResNet-34 and ResNet-50, where the parameters have been pre-trained on a large dataset [32] and [33] input the images of step 1 into the model and specify the evaluation index of the model.

The last step is to train the model using fine-tuning and discriminating the learning rate. Firstly, the parameters of the network layer in the model except the last layer or a few layers are locked using fine-tuning method, and only the

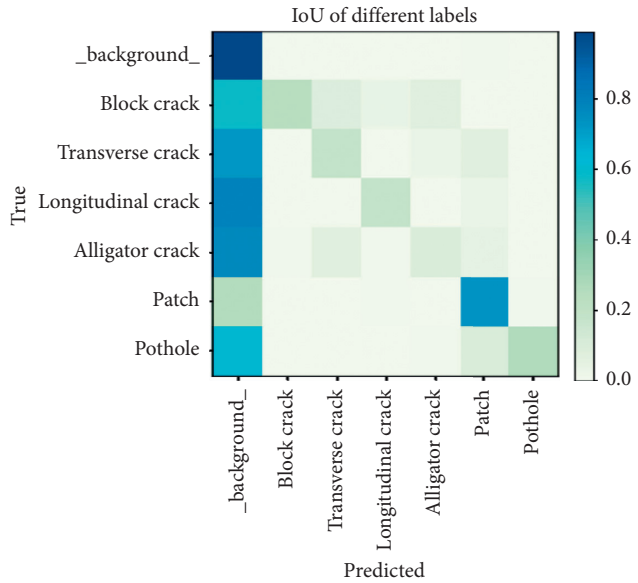

Figure 6: Confusion matrix of IoU values of different labels.

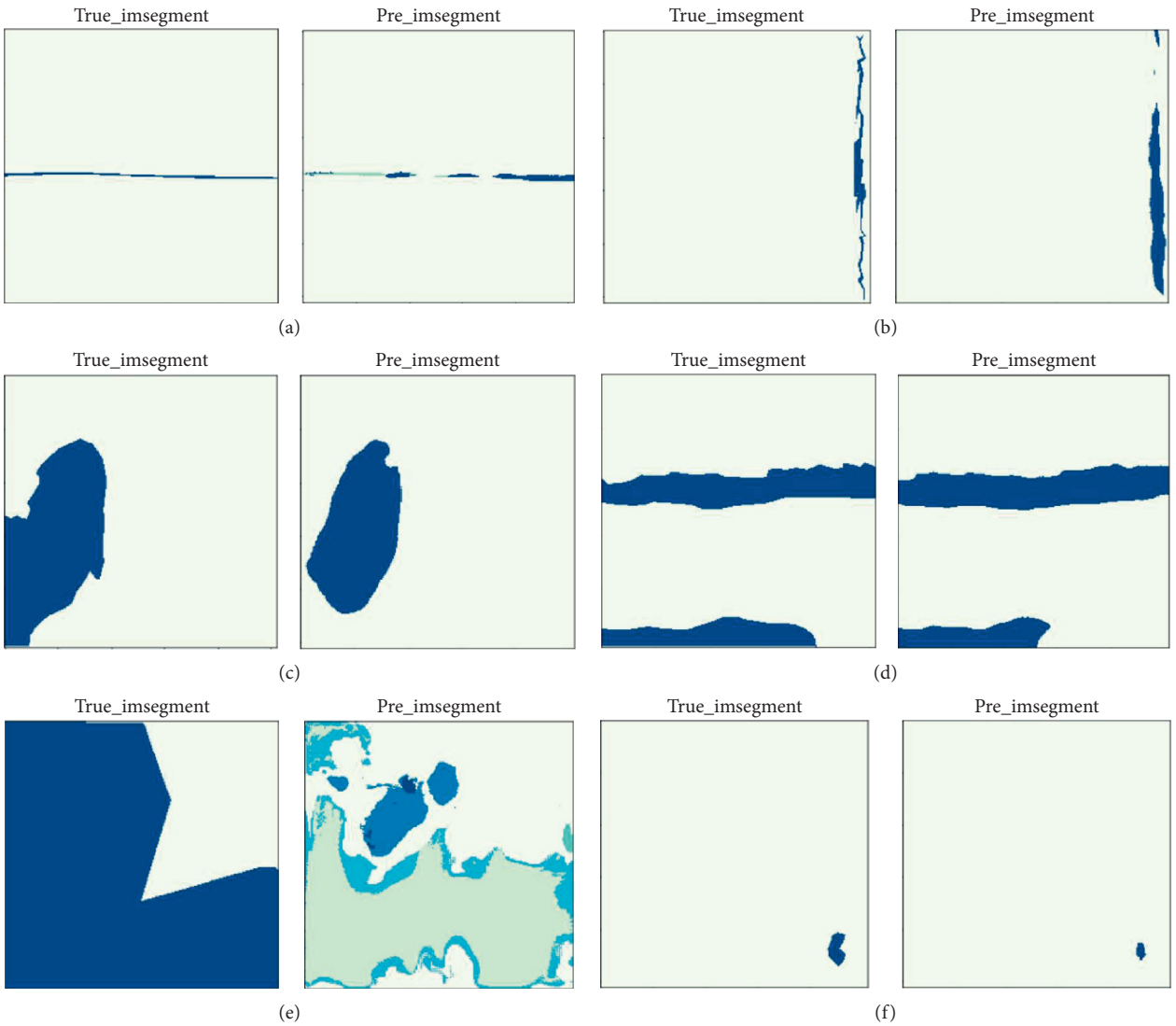

Figure 7: Examples of empirically determined and predicted labels. (a) Transverse crack. (b) Longitudinal crack. (c) Alligator crack. (d) Patch. (e) Block crack. (f) Pothole.

unlocked layers parameters are trained. Then the parameters of the model are divided into three parts by discriminating the learning rate, and the model is trained with each part assigned a different learning rate.

4.3.2. Test of the U-Net Model. Table 4 shows the PA values of the pavement distress pixel-level recognition model using ResNet-34 and ResNet-50 by inputting images with different sizes. When only the distressed or nondistressed areas in an image are to be distinguished, the PA values obtained on the verification set of the two models are 97.336% and 95.772%, respectively. This is higher than the single-category model (89.92%) for crack recognition on an asphalt pavement in Chen et al.'s work [22] and the single-category model (92.12%) for crack recognition of Fitchburg Municipal Airport runways as reported in Cheng et al.'s work [23].

When the distress types detection are concerned, for example, in our case, six types of distresses are to be categorized, the PA of ResNet-34 is evaluated to be 66.103%, higher than the PA of ResNet-50 (44.953%) on the 256×256-pixel image size dataset. However, when 512×512-pixel image size is used, the PA of the model based on ResNet-34 is 62.716%, lower than that on 256×256-pixel image size.

For the six types of distresses detection, the CPA and IoU of each distress obtained by the ResNet-34 model on the validation set with 256×256-pixel image size are shown in Tables 5 and 6, respectively. Compared with distress detection model based on SSD (Single-Shot MultiBox Detector) [34] using the same distress images dataset, where the distressed areas are labeled with rectangle bounding boxes, the IoU is set to 0.5 (the common value in object detection), the average precision (AP) of patches, potholes, transverse cracks, block cracks, alligator cracks and longitudinal cracks is 0.53061, 0.00263, 0.30748, 0.28322, 0.41175, and 0.09820, respectively [35]. The U-net model has great improvement on patches (0.82774 versus 0.53061), potholes (0.31902 versus 0.00263), and longitudinal cracks (0.27439 versus 0.09820) detection over the SSD model but poorly performs on transverse cracks (0.19139 versus 0.30748), block cracks (0.25135 versus 0.28322), and alligator cracks (0.14077 versus 0.28322). The poor performance of the U-net model on the three types of distresses may be due to the fact that some image pixels within the distress outline have the same features as the no-distress pixels outside the outline on the distress images.

Table 6 shows the IoU of different categories (including no-distress areas, that is, the background) on the verification set. The IoU of the no-distress area is 99.059%, and the average is 39.68% (MIoU). Compared with the IoU value of 0.6 for pavement cracks obtained by Jiang et al. [24] with a single-class model, the IoUs of various cracks achieved in this study are low, showing that the detection of location, shape, and size of distressed area of images is still a challenge for multiple classifiers.

Figure 6 shows the IoU confusion matrix diagram of different categories (including the no-distress areas, that is, the background) on the validation set. The horizontal axis in the figure represents the predicted category and the vertical axis represents the true category. The darker the color is, the closer the IoU is to one. In the confusion matrix, the darker the diagonal block is, the higher the accuracy of the model is, and the darker the nondiagonal block is, and there is a greater probability of being incorrectly recognized. Pixels with various distresses have a significant probability of being recognized as no-distress pixels (background pixels). In addition, block cracks are more likely to be mistaken for transverse and alligator cracks, while patches and potholes are rarely mistaken for other types of distresses.

Figure 7 shows a comparison between the manually determined area of pavement distress and the predicted area by the trained model for various types of distresses. It shows that the developed model could recognize multiple distress labels in the images but incur certain errors in identifying the specific type and location of the distresses. This might have occurred for a number of reasons. Firstly, the size of certain pavements distresses occupied a relatively small part of an image. When the image of the pavement distress is reduced, the distress characteristics become less prominent, which might have affected the recognition performance of the model. Secondly, for such distresses as alligator and block cracks, some image pixels within the distress outline have the same features as the no-distress pixels, which might have reduced the recognition accuracy of the model.

5. Conclusions

The development of pavement distress inspection technology and AI, especially convolutional neural networks (CNN), makes it possible to automate the road inspection and evaluation process from road image collection, distress detection, and road condition assessment. To achieve this goal, an efficient and high-precision pavement distress classification and detection model is needed. Toward this purpose, road distress image detection models based on U-net are trained and tested with a road distress image dataset built by road images collected from 567.33 km asphalt expressways in China. The major conclusions of this study are drawn in the following:

(1) A total of 10,097 images of six common distresses captured from four asphalt expressways in three provinces in China are labeled in pixel level to construct a pavement distress image dataset.

(2) Using the U-net model and ResNet, an advanced semantic segmentation model for road distress detection is trained for pixel-level pavement distress recognition. The developed model does not require manual preprocessing of the pavement images and delivers pixel-level distress location, shape, and size in the images, which could automate the distress detection procedure and improve the distress identification accuracy.

(3) For binary classification, which classifies an image pixel as distressed or not distressed, the PA values are evaluated to be 97.336% for the ResNet-34 model using the test dataset. When the six types of common road distresses are to be classified, the trained

multiple feature classifier performs differently on the six types of distresses. It achieves an accuracy of 82.774% for patches but incurs fairly large errors in predicting the shape and location of other types of distresses.

(4) The U-net network structure used in this article contains 4 downsampling processes and 4 upsampling processes, and the image has undergone multilayer convolution before the first downsampling, which makes it extract a smaller number of the feature maps of small size distress such as pothole. On the other hand, cracks have more point and line characteristics, and shallow convolution layers will be helpful to extract the characteristics of cracks and potholes. Thus, changing the number of downsampling processes or upsampling processes, converting long skip connection in the U-Net structure into short skip connection, and integrating multilayer feature maps may help to improve the recognition accuracy.

Data Availability

Some or all data, models, or code supporting the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

The authors are grateful to Weixiang Liu, Ruibo Zhang, and Guocai Zheng for their labeling work for the dataset used in this study.

References

[1] H. D. Cheng and M. Miyojim, "Automatic pavement distress detection system," *Information Sciences*, vol. 108, no. 1–4, pp. 219–240, 1998.

[2] A. Zhang, K. C. Wang, Y. Fei et al., "Deep learning–based fully automated pavement crack detection on 3D asphalt surfaces with an improved cracknet," *Journal of Computing in Civil Engineering*, vol. 32, no. 5, Article ID 04018041, 2018.

[3] J. H. Li, "Pavement crack diseases detecting by image processing algorithm," *Journal of Chang'an University (Natural Science Edition)*, vol. 24, no. 3, pp. 24–29, 2004, in Chinese.

[4] L. Li, L. J. Sun, and C. Chen, "An edge detection method designed for pavement distress images," *Journal of Tongji University (Natural Science)*, vol. 39, no. 5, pp. 688–692, 2010, in Chinsese.

[5] Q. Q. Li, Q. Zou, and Q. Z. Mao, "Pavement crack detection based on minimum cost path searching," *China Journal of Highway and Transport*, vol. 23, no. 6, pp. 28–33, 2010, in Chinese.

[6] J. Lin and Y. Liu, "Potholes detection based on SVM in the pavement distress image," in *Proceedings of the 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science*, pp. 544–547, Hong Kong, China, August 2010.

[7] Z. Q. Shen, Y. H. Peng, and N. Shu, "Road damage identification method based on scale-span image and SVM," *Geomatics and Information Science of Wuhan University*, vol. 38, no. 8, pp. 993–997, 2013, in Chinese.

[8] J. A. Acosta, *Pavement Surface Distress Evaluation Using Video Image Analysis*, Case Western Reserve University, Cleveland, OH, USA, 1994.

[9] X. Chu, X. Yan, and M. Long, "The automatic search of pavement surface distress image based on on-line learning," in *Proceedings of the International Conference on Transportation Engineering 2007*, pp. 3282–3287, Chengdu, China, July 2007.

[10] M. Wu, X. Chen, and C. R. Liu, "Smart structures and materials 2002: smart systems for bridges, structures, and highways," *International Society for Optics and Photonics*, vol. 4696, pp. 293–300, 2002.

[11] S. Gao, Z. Jie, Z. Pan et al., "Automatic recognition of pavement crack via convolutional neural network," *Transactions on Edutainment XIV*, pp. 82–89, Springer, Berlin, Heidelberg, 2018.

[12] H. Maeda, Y. Sekimoto, T. Seto, and H. Omata, "Road damage detection and classification using deep neural networks with smartphone images," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 12, pp. 1127–1141, 2018.

[13] L. Zhang, F. Yang, D. Zhang et al., "Road crack detection using deep convolutional neural network," in *Proceedings of the IEEE International Conference on Image Processing (ICIP 2016)*, pp. 3708–3712, IEEE, Phoenix, AZ, USA, September 2016.

[14] A. M. Sha, Z. Tong, and J. Gao, "Recognition and measurement of pavement disasters based on convolutional neural networks," *China Journal of Highway and Transport*, vol. 31, no. 1, pp. 1–10, 2018, in Chinese.

[15] L. Shi, J. J. Hu, W. Li et al., "Automatic detection method of road surface distress based on deep learning," in *Proceedings of the 10th Annual Conference of the Maintenance and Management Branch of China Highway Society*, pp. 164–170, 2020, in Chinese.

[16] J. Tang, B. Peng, and Y. Zhang, "Automatic pavement crack detection based on deep learning," in *Proceedings of the 14th China Intelligent Transportation Conference*, pp. 458–468, 2019, in Chinese.

[17] W. Liu, Y. Huang, Y. Li et al., "FPCNet: fast pavement crack detection network based on encoder-decoder architecture," 2019, https://arxiv.org/abs/1907.02248.

[18] W. Song, G. Jia, H. Zhu, D. Jia, and L. Gao, "Automated pavement crack damage detection using deep multiscale convolutional features," *Journal of Advanced Transportation*, vol. 2020, Article ID 6412562, 11 pages, 2020.

[19] Z. Fan, C. Li, Y. Chen et al., "Ensemble of deep convolutional neural networks for automatic pavement crack detection and measurement," 2020, https://arxiv.org/abs/2002.03241.

[20] B. Li, K. C. P. Wang, A. Zhang, E. Yang, and G. Wang, "Automatic classification of pavement crack using deep convolutional neural network," *International Journal of Pavement Engineering*, vol. 21, no. 4, pp. 457–463, 2020.

[21] J. Ji, L. Wu, Z. Chen et al., "Automated pixel-level surface crack detection using U-net," in *Proceedings of the International Conference on Multi-Disciplinary Trends in Artificial Intelligence*, Springer, Hanoi, Vietnam, November 2018.

[22] Z. B. Chen, W. T. Luo, and L. Li, "Automatic identification of pavement crack using improved U-net model," *Journal of*

Data Acquisition and Processing, vol. 35, no. 2, pp. 260–269, 2020, in Chinese.

[23] J. Cheng, W. Xiong, W. Chen et al., "Pixel-level crack detection using U-net," in *Proceedings of the TENCON 2018-2018 IEEE Region 10 Conference*, IEEE, Jeju, South Korea, October 2018.

[24] L. Jiang, Y. Xie, and T. Ren, "A deep neural networks approach for pixel-level runway pavement crack segmentation using drone-captured images," 2020, https://arxiv.org/abs/2001.03257.

[25] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[26] K. He, X. Zhang, S. Q. Ren et al., "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[27] O. Ronneberger, P. Fisher, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, Munich, Germany, October 2015.

[28] I. Sergey and S. Christian, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceedings of the International Conference on International Conference on Machine Learning*, Lille, France, 2015.

[29] L. N. A. Smith, "Disciplined approach to neural network hyper-parameters: part 1-learning rate, batch size, momentum, and weight decay," 2018, https://arxiv.org/abs/1803.09820.

[30] J. Lu, V. Behbood, P. Hao, H. Zuo, S. Xue, and G. Zhang, "Transfer learning using computational intelligence: a survey," *Knowledge-Based Systems*, vol. 80, pp. 14–23, 2015.

[31] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," 2018, https://arxiv.org/abs/1801.06146.

[32] J. Deng, W. Dong, R. Socher et al., "Imagenet: a large-scale hierarchical image database," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, June 2009.

[33] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes challenge: a retrospective," *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, 2015.

[34] W. Liu, D. Anguelov, D. Erhan et al., "Single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Amsterdam, Netherlands, October 2016.

[35] D. R. Li, Z. D. Duan, X. Y. Hu, D. C. Zhang, and Y. Y. Zhang, "Automated classification and detection of multiply pavement distress images based on deep learning," *Journal of Traffic and Transportation Engineering (English Edition)*, 2020.