Research Article

An Advanced Otsu Method Integrated with Edge Detection and Decision Tree for Crack Detection in Highway Transportation Infrastructure

Haihang Han,[1] Hanyu Deng,[2,3] Qiao Dong ⓘ,[2] Xingyu Gu,[2] Tianjie Zhang ⓘ,[1] and Yangyang Wang[1]

[1]*Zhejiang Scientific Research Institute of Transport,*
 Zhejiang Provincial Key Lab for Detection and Maintenance Technology of Road and Bridge, Hangzhou, Zhejiang 311305, China
[2]*School of Transportation, National Demonstration Center for Experimental Road and Traffic Engineering Education,*
 Southeast University, Nanjing, Jiangsu 211189, China
[3]*Department of Civil and Environmental Engineering, University of California, Davis, Berkeley, CA 95616, USA*

Correspondence should be addressed to Qiao Dong; qiaodong@seu.edu.cn

The detection of various cracks on pavement surfaces has drawn more and more attention from pavement maintenance engineers. In the traditional pavement image segmentation, due to the small area of the pavement cracks, the gray level of crack pixels only accounts for a very small portion in the grayscale histogram, making it difficult to segment. This paper developed an improved Otsu method integrated with edge detection and a decision tree classifier for cracking identification in asphalt pavements. An image preprocessing approach including Gaussian function-based spatial filtering and top-hat transform is firstly proposed to reduce the influence of poor shading and lighting effects significantly. Four edge detection operators including Prewitt, Sobel, Gauss–Laplace (LoG), and Canny are evaluated. The Canny edge detection has demonstrated outstanding performance in crack detection; this algorithm helps to obtain more details of both cracks and noises. The Sobel and LoG operators show similar image segmentation and retain fewer noises. The decision tree classifier based on the ID3 algorithm can effectively classify different types of cracks including transverse, longitudinal, and block ones.

1. Introduction

With the rapid development of the highway transportation infrastructure network and the increase of pavement service life, pavement distress including cracks, potholes, ruts, etc., increase rapidly. The detection and treatment of pavement distress have gradually become an important focus in the field of pavement engineering. As major pavement distress, cracks usually indicate the reduction of pavement performance and risk more serious pavement structural distresses. Therefore, the quick detection and treatment of pavement cracks at an early age is the key to extending the service life of pavements and saving maintenance funds [1]. Traditional crack detection methods rely on manual identification, which is inefficient and subjective. In recent years, a variety of intelligent detection equipment such as multifunctional pavement detection vehicles have been used in pavement distress evaluation, which usually involves an automatic collection of high-quality pavement images without the traffic's influence.

With the development of computer technology and digital image processing technology, digital pavement crack recognition methods began to appear and develop rapidly. Morphological operations like top-hat and bottom-hat are applied for the contrast enhancement of the image, which helps to achieve better efforts of the segmentation of verminous objects like retinal vessels [2]. Multiscale new top-hat transform was also conducted on infrared image

enhancement algorithm through contrast enhancement [3]. Experts and scholars have proposed many automatic crack extraction methods based on pavement surface images. The classical edge detection algorithm is used to obtain the crack edge by extracting the pixels with a larger gradient [4, 5]. Tsai et al. used different edge detection operators including Sobel, LoG, Canny, and Prewitt to detect the crack structure on the original concrete [6]. It was found that the LoG operator is ideal and relatively simple. Canny operator has the best capability to extract weak edges, but it is also more vulnerable to noise. The performance of six edge detectors and the deep convolutional neural networks (DCNN) for concrete crack detection was investigated and compared. A hybrid crack detector by combining the DCNN and the edge detector was proposed, which had 24 times less noise than the least noisy edge detector [7]. Wang et al. proposed an asphalt pavement crack detection algorithm based on multiscale ridge edge [8]. The filter was constructed by Gaussian function, and its first-order and second-order derivatives were used to convolute the rows and columns of the image respectively to determine the ridge edge center and width. Then, the ridge edge detected at each scale was fused to obtain the ridge edge image and finally denoised and connected to detect cracks by the expansion and minimum spanning tree algorithm. Mao-De et al. proposed a pavement crack edge detection algorithm based on the morphology [9]. For the pavement image after median filtering, the gradient operator and closing operator are appropriate for edge extraction and gap closure, which can better extract the skeleton of pavement cracks.

The threshold method segments pavement cracks and background by setting static or dynamic thresholds, realizing the automatic extraction of cracks [10, 11]. In 1979, Otsu proposed a classical threshold segmentation method based on a gray histogram [12]. The maximum value of the variance between classes was used as the criterion to obtain the threshold K, and then the image was binarized. This method is simple and easy to understand and has the potential to be extended to two or even multithreshold segmentation. Li et al. proposed a pavement crack image segmentation method related to the neighborhood difference histogram based on threshold idea, which has better effects for the early development of narrow cracks [13]. The steerable matched filtering and an active contour model [13] firstly enhances the contrast between cracks and surrounding pavement and captures crack discontinuity and curvature and then uses a region-based active contour model for crack segmentation. Ai et al. proposed an automatic image crack segmentation method based on probability and large-scale domain information on the pixel level using threshold segmentation [14]. Lau et al. proposed U-Net-based network architecture for the automated pavement crack segmentation [15]. The traditional convolutional network has a fully connected layer as the final layer, which the U-Net model only contains convolutional layers [15] and therefore is more efficient. The probability generation model and support vector machine were used to calculate the crack probability based on the pixel gray level and the domain information, employing the probability overlay map for crack segmentation and extraction.

After the segmentation results are obtained, the decision tree method is used to eliminate the noise part by further optimizing the segmentation results. Decision tree technology can establish classification models because of its simplicity and proximity like human thinking [16]. ID3 algorithm is one of the most representative decision tree algorithms. It adopts a branch strategy and constructs a decision tree through a selection window [17]. After that, the C4.5 algorithm [18] and SPRINT algorithm [19] were proposed. Qin et al. found that ID3 algorithm can quickly develop an accurate decision tree and is effective with a large number of attribute values [20]. The ID3 decision tree can effectively remove the noise from the segmentation results and improve the segmentation.

The deep learning-based method has achieved remarkable successes in computer vision, especially in the last five years [21–23]. However, it is still hard to interpret the detector based on the convolutional network. In addition, deep learning needs a large amount of high-quality images for training and requires extensive labeling work. Therefore, the traditional image process still has some advantages on pavement crack detection. The edge detection method and threshold method mentioned above are simple and highly efficient, but they are too sensitive to noise and have poor effects when the background is complex or the gray level of the background is close to that of the crack. In order to solve these problems, this study proposes a method combining the edge detection method with the threshold method. An improved Otsu method based on the edge detection algorithm is also proposed. A decision tree was then used to identify the cracks from background noises. Four edge detection methods including the Prewitt operator, Sobel operator, Laplace of Gauss (LoG) method, and Canny were evaluated with the improved optimal global threshold method in segmenting pavement images. The gray images of asphalt pavement surface collected by pavement evaluation vehicles are used for this study. The influences of different edge detection operators and the proposed algorithm on the final segmentation are discussed to validate the proposed method.

2. Methodology

There are several assumptions for crack detection based on image processing [4]. In the gray image of pavement surface including cracks, the gray values of crack pixels are deeper than those of pavement pixels, and the gray distribution of cracks and pavement backgrounds is independent. A crack is a narrow, continuous target and a group of interconnected segments with different directions. The width of a crack is not constant over the entire length. The pixels in the crack can be considered as the optical and/or geometric points of interest. In 1979, Otsu [12] proposed the optical global threshold image segmentation method, namely, the Otsu method. This method is considered to be the best method in global threshold processing under the condition of maximum interclass variance [4]. However, in the crack region segmentation of pavement images, if the crack area only accounts for a small proportion of the whole image, it is

difficult to obtain satisfying results by using the Otsu method. As shown in Figure 1, in this study, the edge detection method is firstly used to identify all edges in the image. The Otsu method is then used to select the optimal threshold value of the edge region in the pavement image for the segmentation, and the decision tree is adopted to further eliminate the noise from cracks.

2.1. Image Preprocessing. The quality of pavement surface images is usually reduced by many reasons including different lighting conditions such as sunny or cloudy, random grainy texture, nonuniform lighting, irregular shadows, pavement markings, watermarks, tire marks, oil stains, etc. These factors have a significant impact on the detection of cracks based on the image processing. The image preprocess mainly includes eliminating or reducing the negative effects of those factors and could significantly improve the image processing effectiveness. In this study, the Gaussian function-based spatial filtering and top-hat transform are utilized to preprocess the collected pavement images.

To filter out the noise to prevent false detection, a Gaussian filter kernel can be convolved with the image to slightly smooth the image to reduce the effects of obvious noise on the edge detector. Spatial domain filtering based on Gaussian function, including using a two-dimensional Gaussian function to construct a filter template in the spatial domain to smooth the image by spatial convolution of the input image, is described as follows:

$$G(x, y) = e^{\frac{x^2 + y^2}{2\sigma^2}}, \tag{1}$$

where σ is the kernel, a distribution parameter, with a default value of 2.5 and x, y are the input values.

In grayscale morphology, the combination of image subtraction and opening and closing operations produces the top-hat transform and bottom-hat transform. The top-hat transform and bottom-hat transformation have generally similar functions, while the difference is the object. The top-hat transform is used for light objects on a dark background, and the bottom-hat transform is used for dark objects on a light background. The top-hat of f is defined as f minus its opening operation:

$$T_{\text{tophat}}(f) = f - (f \circ b). \tag{2}$$

Similarly, the bottom-hat of f is defined as the closing operation minus f:

$$T_{\text{bothat}}(f) = (f \cdot b) - f, \tag{3}$$

where f is the original image and b is a structural element. The size of b is decided mostly based on the conversion relation between the pixel and the realistic scale and the common size of cracks. The default size of b is 150 mm.

2.2. Edge Detection. One drawback of using the Otsu method for pavement crack detection is that crack only accounts for a very small area in the image and is not very prominent in the gray-level histogram of the image. This study adopts edge
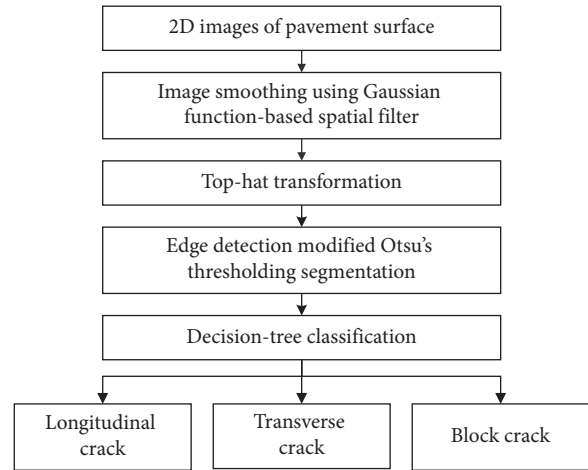

FIGURE 1: Flowchart of the methodology.

detection to identify the potential crack edge area first and then uses the Otsu method to only deal with the identified crack edge area to significantly improve the detection efficiency. Traditional edge detection algorithms in image processing include the Prewitt gradient operator, Sobel gradient operator, Gauss–Laplace (LoG) operator, and Canny operator. For edge detection, the edge in the image refers to an abrupt gray-level change. Both first-order and second-order differentiation can be used to detect gray-level change. The derivative of a function at a point can be defined by difference. There are several assumptions of the derivative. The approximation of the first derivative should be zero in the area of unique gray level and should not be zero at the start of gray steps or ramps and within the gray ramp. The approximation of the second derivative should be zero in the area of unique gray level, must not be zero at the start of gray steps or slops, and must be zero within the gray ramp. The derivative is approximated by Taylor expansion to construct the filter template. The approximates of the first and second derivatives of the function are shown as follows:

$$\frac{\partial f}{\partial x} = f'(x) = f(x + 1) - f(x), \tag{4}$$

$$\frac{\partial^2 f}{\partial^2 x} = f''(x) = f(x + 1) + f(x - 1) - 2f(x). \tag{5}$$

In edge detection, there are three commonly used edge models: step edge, ramp edge, and roof edge. Figure 2 shows the grayscale curves and the first- and second-order differential curves of the ramp and the roof edge models. As shown in Figure 2(b), in the second derivative, the two extreme points are the maximum and minimum points of the second derivative at the bottom and top of the gray ramp. The intersection of the straight line connecting the maximum and minimum points of the second derivative and the zero gray-level axis is called the zero-crossing point of the second derivative.

2.2.1. Prewitt and Sobel Operator. The intensity and direction of the gray-level gradient in images can be detected based on the first-order difference of the image gray
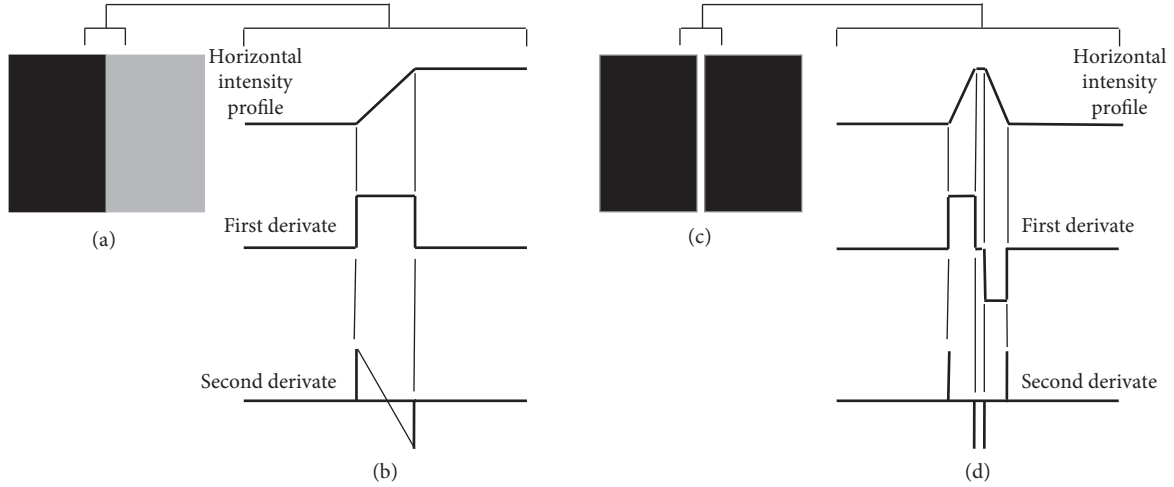
FIGURE 2: Schematic of the gray model and its first and second derivatives. (a) Ramp edge model. (b) Gray profile of the ramp edge model and its corresponding first and second derivatives. (c) Roof edge model. (d) Gray profile of the roof edge model and its corresponding first- and second-order derivative.

gradient. Both the Prewitt and Sobel operators are discrete differentiation operators for edge detection by gradient transform. Figure 3 shows the templates of the two gradient operators. The operators use two 3×3 kernels which are convolved with the image to calculate approximations of the derivatives—one for horizontal changes, and the other for vertical.

The grayscale gradient of the image can be obtained using the gradient operator shown in Figure 3. Usually, we can use the gradient images g_x and g_y, which at each point contain the horizontal and vertical derivative approximations, respectively, to calculate the gradient intensity at the gradient's direction based on equation (6). $M(x, y)$ is the magnitude of the gradient. To save calculation time, the intensity of the gradient can also be approximated by equation (7).

$$M(x, y) = \text{mag} \begin{pmatrix} \dfrac{\partial f}{\partial x} \\[2mm] \dfrac{\partial f}{\partial y} \end{pmatrix} = \sqrt{g_x^2 + g_y^2}, \qquad (6)$$

$$M(x, y) = |g_x| + |g_y|. \qquad (7)$$

The difference between the Prewitt operator and Sobel operator is that the Sobel operator has a larger coefficient of the center point. The central part of the pixel occupies a greater weight, which can smooth the noise better than the Prewitt operator [24]. The templates in Figure 3 can be revised to make the edge detection more sensitive to the diagonal direction.

2.2.2. Laplace of Gaussian (LoG) Operator.

In the gradient operator based on the second derivative, Marr and Hildreth combined the Laplace operator (equation (9)) and the two-dimensional Gaussian function with σ as the standard

deviation (equation (1) to form the LoG operator (equation (9)) [25]. Laplacian is also very sensitive to noise as other first-order methods. To reduce the noise effect, the two-step LoG operation is to first smooth the image with a Gaussian filter and then to detect the zero-crossings using Laplacian. After generating the spatial convolution template from equation (9), we can perform spatial convolution on the input image $f(x, y)$ to obtain the result $g(x, y)$ and find the zero-crossing point in $f(x, y)$ to identify the edge in the input image $f(x, y)$.

$$\nabla^2 = \frac{\partial^2}{x^2} + \frac{\partial^2}{y^2}, \qquad (8)$$

$$\begin{aligned} \nabla^2 G(x, y) &= \frac{\partial^2 G(x, y)}{x^2} + \frac{\partial^2 G(x, y)}{y^2} \\ &= \left[\frac{(x^2 + y^2) - 2\sigma^2}{\sigma^4} \right] e^{-((x^2 + y^2)/(2\sigma^2))}. \end{aligned} \qquad (9)$$

The LoG operator uses Gaussian low-pass filtering to smooth the image, effectively reducing noise interference. In addition, the LoG operator has an equal response to the gray change of any template direction in the original image, instead of using multiple templates to calculate the gray gradient in different directions of the image when using the operator based on the first derivative, and therefore is also very efficient.

2.2.3. Canny Operator.

The Canny operator uses the first-order directional derivative of the two-dimensional Gaussian function in any direction to reduce noise and compare it with the spatial convolution of the input image $f(x, y)$ to suppress noise and then find the maximum gradient to detect the edge of the image. The Canny operator firstly uses a Gaussian filter to smooth the input image and find the intensity gradients of the image. Then, it applies gradient magnitude thresholding or lower-bound cut-off

−1	−1	−1
0	0	0
1	1	1

−1	0	1
−1	0	1
−1	0	1

(a)

−1	0	1
−2	0	2
−1	0	1

−1	−2	−1
0	0	0
1	2	1

(b)

Figure 3: Prewitt and Sobel operator templates. (a) Prewitt operator. (b) Sobel operator.

suppression to get rid of spurious response to edge detection and applies a double threshold to determine potential edges. At last, it finalizes the detection of edges by suppressing all the other edges that are weak and not connected to strong edges. The Canny operator has a low error rate since all edges should be found, and there should be no spurious responses. It can also locate the edge close to the real edge. Therefore, it is one of the most strictly defined methods that provide good and reliable detection.

2.3. Otsu's Thresholding.

The result of edge detection is not the crack area itself, but the edge of the crack area. It still needs image segmentation to identify the crack area. The purpose of the improved Otsu method using edge detection is to find the edge in the image and only use the pixels near the edge area to construct a grayscale histogram and use the grayscale histogram as the object of the Otsu method to obtain the segmentation threshold. This can effectively reduce the influence of a relatively large background area on Otsu's best global threshold segmentation.

The basic principle of Otsu's method is to use a threshold to divide the image into two parts, the region, and the background by maximizing the between-class variance. Otsu's method is based on computations performed on the histogram of an image, which is a one-dimensional array. The corresponding threshold gray value for classification is called the optimal threshold. For an image with a total of N pixels, the probability p_i of each gray level in the gray image is calculated by

$$p_i = \frac{n_i}{N}, \quad i = 0, 1, 2 \dots, L-1, \tag{10}$$

where n_i is the number of pixels whose gray value is i and $L-1$ is the largest gray value.

Let K be the initial value of the threshold; use this threshold to divide all pixels into two parts with gray values from 0 to $K-1$ and from K to $L-1$. The between-class variance σ_B^2 is calculated by equation (11). Generally, the initial value K of the threshold is usually set to 1, and the maximum between-class variance when $K = 1, 2 \dots, L-2$ is calculated. A gray value K that maximizes σ_B^2 is calculated as the final segmentation threshold.

$$\sigma_B^2 = \omega_0 \left(\mu_0 - \mu_r\right)^2 + \omega_1 \left(\mu_0 - \mu_r\right)^2, \tag{11}$$

where $\omega_0 = \sum_{i=0}^{k-1} p_i$, $\omega_1 = \sum_{i=k}^{L-1} p_1 = 1 - \omega_0$, $\mu_0 = \sum_{i=0}^{k-1} i p_i / \omega_0$, $\mu_1 = \sum_{i=k}^{L-1} p_i / \omega_1$, and $\mu_r = \sum_{i=0}^{L-1} p_i$.

The improved Otsu method integrating edge detection includes the following steps.

(1) Obtain pavement crack image $f(x, y)$ after pre-processing and edge detection.

(2) Specify a threshold value of gray level T to threshold the image to obtain a binary image $gT(x, y)$.

(3) For the area where $gT(x, y)$ equals to the logical value 1, find the corresponding area in image $f(x, y)$ to calculate the gray-level histogram and use equation (10) to calculate the probability value of each gray level.

(4) For each gray-level probability value obtained in step (3), use equation (11) to calculate the maximum between-class variance σ_B^2 and obtain the best threshold K to segment image $f(x, y)$.

2.4. Decision Tree Classifier.

Pavement crack only accounts for a small part in pavement images. After the edge detection and Otsu's thresholding, the detected regions in the image may still include potholes and noises, other than cracks, and need to be further classified. Decision tree is a robust supervised learning classifier for pattern recognition, which relies on a labeled training set. Decision tree has small computation cost and high classification accuracy. It is also very easy to generate classification rules which are accurate and easy to understand.

Decision tree has a tree structure used for classification and prediction [16]. Generally, it consists of root nodes, decision nodes, branches, and leaves. The root node includes the full set of samples. Decision nodes and branches from the root are connected to each leaf. It represents the classification path of a sample. Each decision node represents a classification on a feature. Each branch represents a classification result, and leaves refer to a class or part of a class. Determination of the optimal subfeature is the key to the training of the decision tree.

The ID3 tree uses a greedy search approach to determine decision node selection. It picks an ideal attribute once and does not reconsider or modify its previous choices. ID3 algorithm uses entropy and information gain to determine which attributes best split the data. This algorithm can ensure that a decision tree is developed with the most simple path being found and the smallest number of branches. The expected information or entropy is a measure of uncertainty associated with a random number. Let the training set be X; the total number of samples is N, which contains M different classes w_i. Let N_i be the number of samples belonging to class w_i in $\bar{X}$. For a given sample classification, the expected information required is

TABLE 1: Description of features.

No.	Feature	Description
X1	Ratio of the major axis and the minor axis	The ratio of the length (in pixels) of the major axis to the length (in pixels) of the minor axis of the ellipse that has the same normalized second central moments as the region, returned as a scalar
X2	Angle between the horizontal axis and the major axis of the ellipse	Angle between the horizontal the x-axis and the major axis of the ellipse that has the same normalized second central moments as the region, returned as a scalar
X3	Area of the region	Actual number of pixels in the region
X4	Standard deviation of the gray level in the region	Standard deviation value of the gray histogram of pixels in the region
X5	Mean of the gray level in the region	Mean value of the gray histogram of pixels in the region
X6	Third-order moment of regional grayscale	Third-order moment value of the gray histogram of pixels in the region

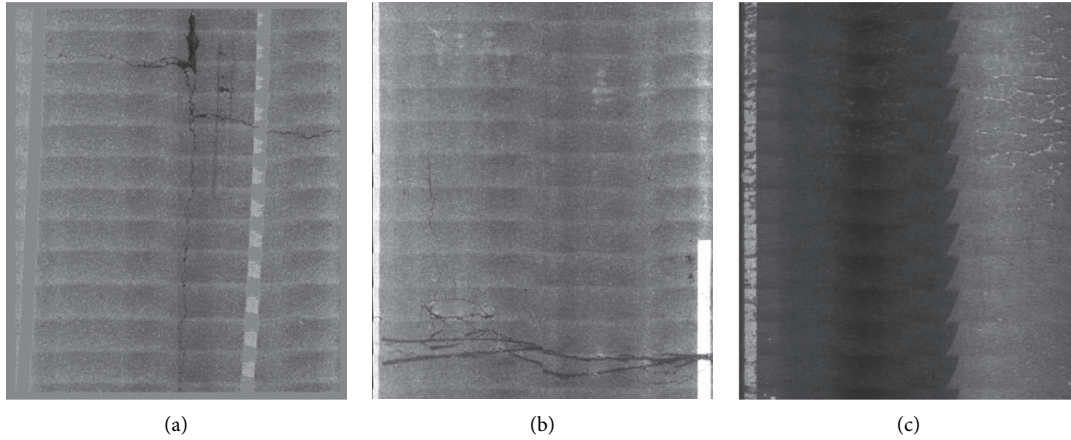(a)   (b)   (c)

FIGURE 4: Samples for each crack class. (a) Longitudinal crack. (b) Transverse crack. (c) Block crack.

$$I(N_1, N_2, \ldots, N_M) = -\sum_{i=1}^{M} p_i \log_2 (p)_i, \qquad (12)$$

where p_i is the probability that the sample belongs to class w_i, $p_i = N_i/N$.

In this study, six features are extracted as the predictors to train the decision tree, including the ratio of the major axis and the minor axis of the ellipse with the same second moment as the region; the angle between the horizontal axis and the major axis of the ellipse, the ellipse and the region have the same second moment; area of the region; the standard deviation of the gray level in the region; mean of the gray level in the region; and the third-order moment of regional grayscale. The detailed description of the features is listed in Table 1. A total of 251 pavement crack images including 131 transverse cracks, 92 longitudinal cracks, 45 block cracks, and some noises are labeled as the training set. The images with their original large size of 3.75 m*5 m were used to perform crack classification. Figure 4 shows the sample of each type of crack. The image segments are classified into four groups of transverse cracks, longitudinal cracks, block cracks, and noises. After training the decision tree, the pavement crack image can be reconstructed with only predicted cracks to calculate the location, length, and width of the cracks.

To evaluate the accuracy of pavement classifications, several performance measures including precision, recall, and F-measure as shown in equations (13)–(15) are frequently adopted. Precision is the ratio of the number of positive samples correctly classified as positive over the total number of positive samples. The recall is the ratio of the number of positive samples correctly classified as positive over the total number of positive samples. F-measure combines precision and recall:

$$\text{precision} = \frac{TP}{(TP + FP)}, \qquad (13)$$

$$\text{recall} = \frac{TP}{(TP + FN)}, \qquad (14)$$

$$F - \text{measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}. \qquad (15)$$

3. Discussion of Results

Pavement crack images were collected and processed with the proposed method including preprocess, edge detection, Otsu's thresholding, and ID3 decision tree classification. Different edge detection operators were evaluated and compared. The code for image preprocess and edge detection is shown in the Appendix section.

3.1. Preprocessing. Figure 5 shows a typical pavement surface image using the top-hat filtering, which is to remove the brightness in the background information from an image
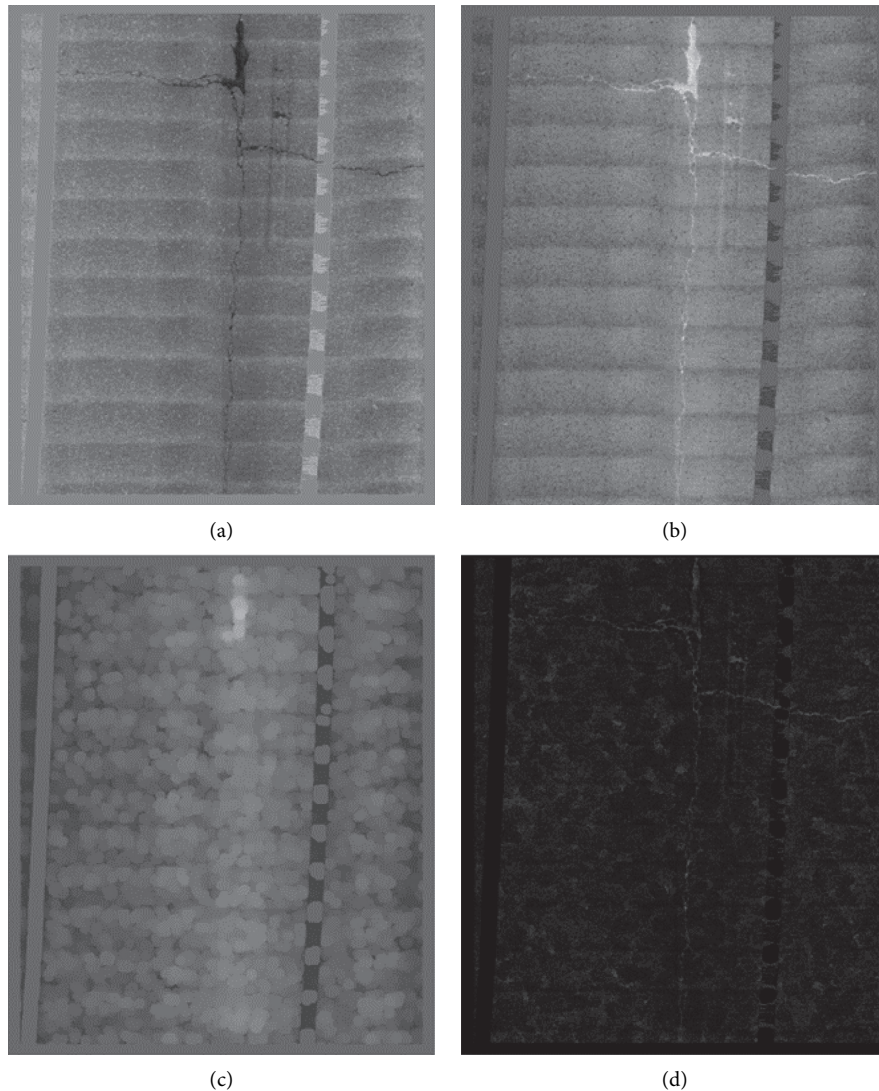
(a)

(b)

(c)

(d)

FIGURE 5: Effect of top-hat conversion. (a) Original image. (b) Image after color reversing. (c) Brighter areas in the image. (d) Result of the top-hat transformation.

through opening operations. The color of the pavement image was reversed, the crack was light, and the background was dark as shown in Figure 5(b). Figure 5(c) shows the brighter area in the image, which could be reduced by the top-hat transform. Figure 5(d) then can be obtained by reducing the brighter area in the original image, and the cracks become clearer.

3.2. *Influence of Edge Detections.* The original road image and image segmentation with preprocessing, edge detection, and Otsu's thresholding are shown in Figure 6. Figures 6(b)~6(e) clearly show that the developed Otsu's method effectively extracted the main crack area from the background area, showing a good segmentation effect. Generally, different edge detection methods have little effect on image segmentation. To further compare the results of image segmentation in more detail, the number of noise regions of different edge detection is calculated.

For the pavement image shown in Figure 6, the numbers of noise regions of the Prewitt, Sobel, LoG, and Canny operators are 143, 111, 123, and 159, respectively. The corresponding numbers of crack regions are 21, 21, 21, and 23, respectively. Generally, the four operators obtained the same crack regions. The Canny edge detection has a better effect on crack detection than the other methods, obtaining more details of the edge and crack area, while retaining more noises. The Sobel and LoG operators show similar image segmentations. The Prewitt and Canny operators have more noise in the image background. This is because the Sobel gradient operator and the spatial domain filter template in the LoG operator could reduce noise. In addition, by comparing Figures 6(a) and 6(f), it can be seen that preprocessing significantly improves segmentation effects. A large amount of noise remains without preprocessing. A good segmentation could not be obtained by solely using edge detection and Otsu's thresholding.
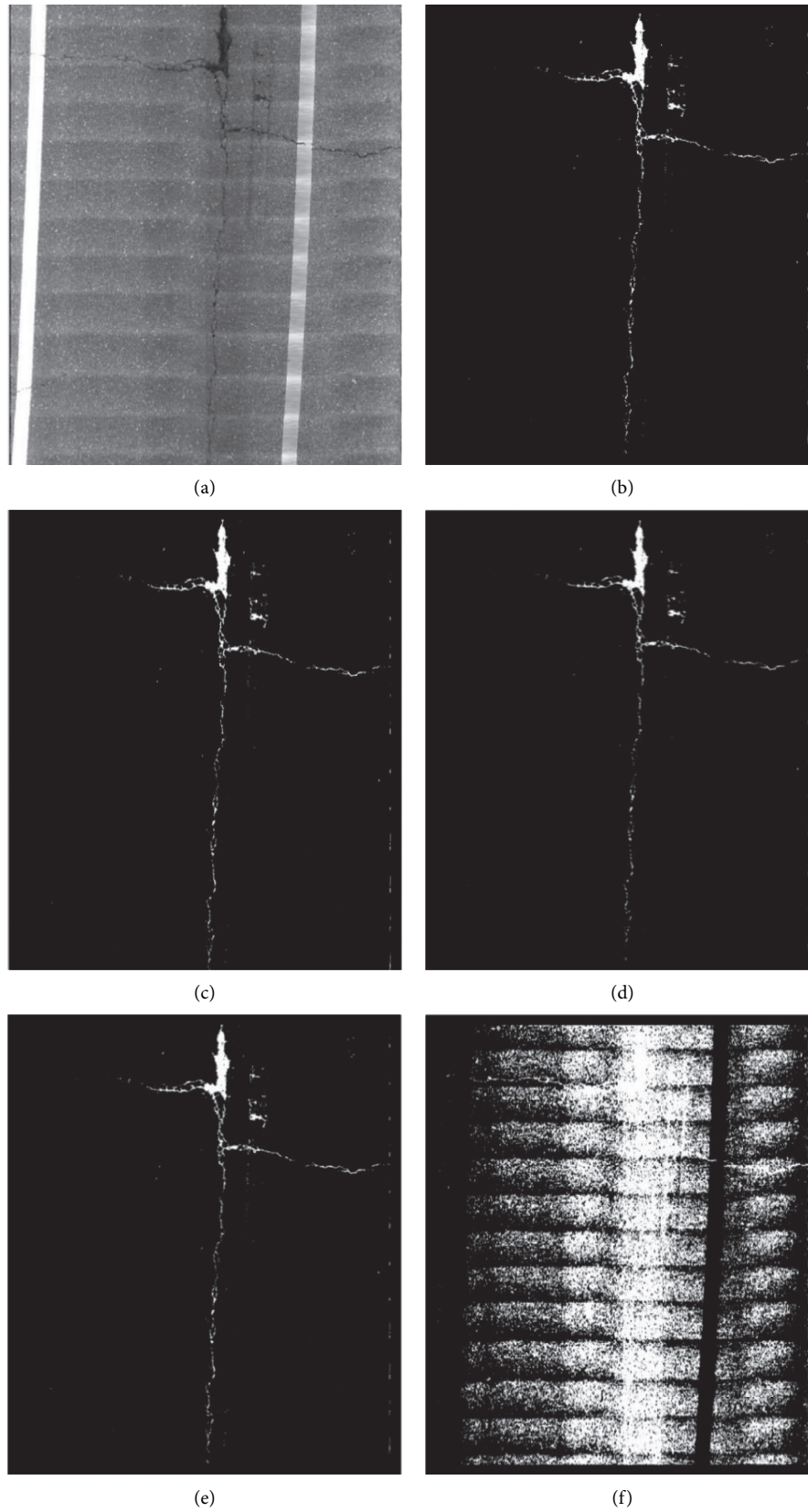
(a)

(b)

(c)

(d)

(e)

(f)

Figure 6: Image segmentation using different operators. (a) Original image. (b) Prewitt operator. (c) Sobel operator. (d) LoG operator. (e) Canny operator. (f) Segmentation without preprocessing.
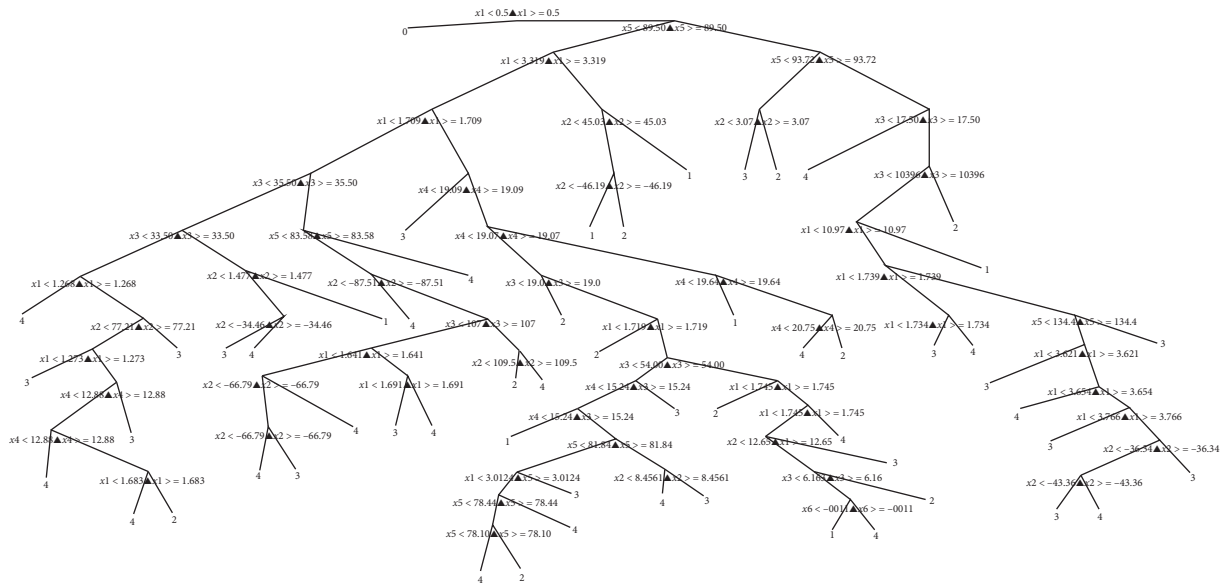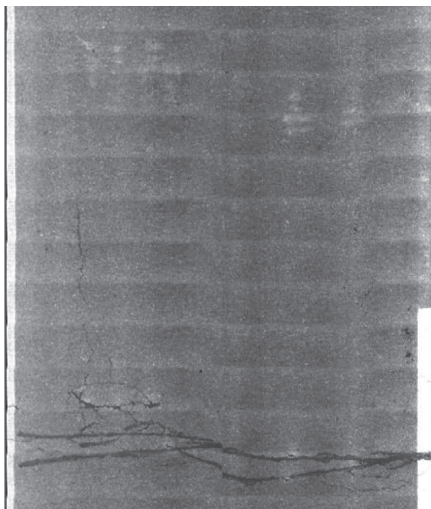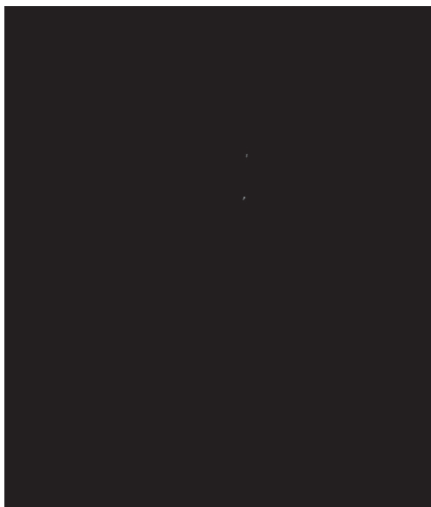
FIGURE 7: Structure of the decision tree.

(a)

(b)

(c)

(d)

FIGURE 8: Continued.

(e)   (f)
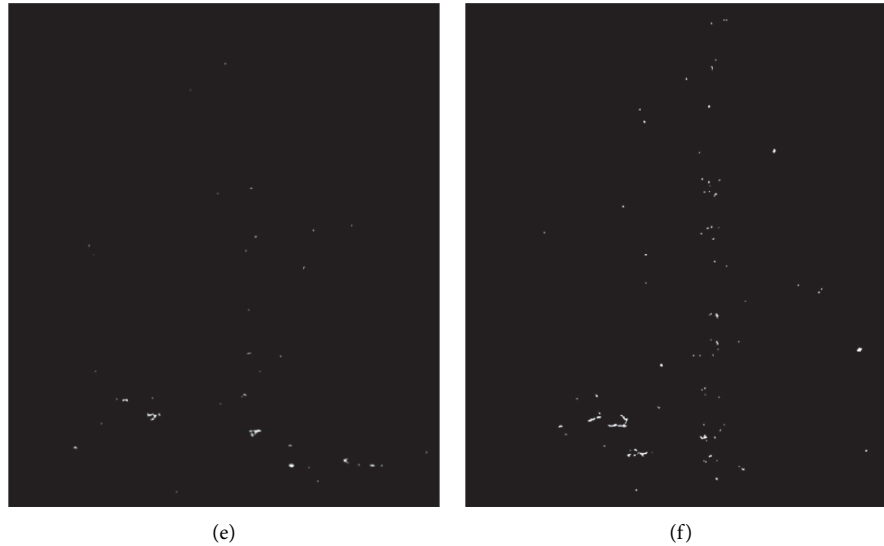
FIGURE 8: The effect of decision tree classifier. (a) Original image. (b) Segmentation. (c) Longitudinal crack classification image. (d) Transverse crack classification image. (e) Block crack classification image. (f) Noises.

3.3. Decision Tree Classification. Figure 7 shows the structure of the decision tree model. $x_1, x_2 \ldots, x_6$ are the six features defined above. Figure 8 shows the pattern recognition effect of the decision tree classifier with a pavement crack image containing transverse cracks. It can be seen from Figure 8(d) that the transverse crack regions in the image segmentation results are effectively classified and be separated from other types of cracks and noise. In Figure 8(d), the noises are significantly reduced, which shows that the secondary denoising effect of the decision tree classifier. Generally, different types of cracks and the corresponding regions in the image are successfully extracted, except that part of the branches of the transverse cracks are identified as block cracks, and a very small amount of noise appears in the longitudinal crack classification image. The proposed method achieved a precision of 88.9%, a recall of 82.8%, and an *F*-measure of 85.3%, indicating a comparable performance.

4. Conclusion

Because the area of the pavement crack is too small, comparing with the image background, the crack only accounts for a very small portion in the grayscale histogram and the pixels are highly concentrated, making it difficult to split effectively. This paper developed an improved Otsu method integrated with edge detection and decision tree classifier for cracking identification in asphalt pavements through image segmentation. An image preprocessing approach including Gaussian function-based spatial filtering and top-hat transform is also proposed.

The Gaussian function-based spatial filtering and top-hat transform significantly reduce the influence of poor shading and lighting effects and improve the image segmentation effects. The improved Otsu optimal global threshold segmentation method based on edge detection could effectively segment pavement crack images after valid preprocessing.

All the four edge detection operators have similar effects on segmentation. The Canny edge detection has a better effect on crack detection, obtaining more details of the edge and crack area, as well as more noises. The Sobel and LoG operators show similar image segmentation and retain fewer noises. The decision tree classifier based on ID3 algorithm can effectively classify different types of cracks including transverse, longitudinal, and block cracks, which also has high calculation efficiency.

The proposed method achieved a fairly high precision, indicating a comparable performance on the crack detection based on 2D pavement surface images. However, it is still sensitive to the quality of images, especially when the pavement surface image contains extensive dirty spots, water, pavement texture, or shadows. Recently, the high-resolution surface profile of pavement can be obtained with 3D cameras and laser line scanner. Those distress detection algorithms can be potentially directly used to process the data with depth information to evaluate pavement distress or texture. They can also be integrated with the deep learning-based methods to firstly identify the critical region to improve the calculation efficiency. In future studies, more types of cracks and other pavement distress including potholes and raveling could be potentially detected using the proposed methods with more pavement distress images for training the decision tree model.

Appendix

"Code for image preprocess and edge detection"

```
function [I] = autotophat(I, varargin)
%Gaussian filter and Top-hat transformation
if nargin = = 1
    A = 1; cj = 150;
    disp("Default A = 1, cj = 350");
```

```
else
    if nargin = = 2
    A = varargin{1};
    cj = 350;
    disp("A = input value");
    else
        if nargin = = 3
        A = varargin{1};
        cj = varargin{2};
        else
            disp("wrong input");
            return;
        end
    end
end
    ave = fspecial("Gaussian",3);
I = imfilter(I, ave);
[ri,~] = size(I);
se = strel("rectangle", [ri, round(cj*A)]);
w = fspecial("average", [10]);
Ibg = imopen(aveimf(I), se);
%I = medfilt2(I, [9]);
%I = imtophat(I, se);
I = I + mean(mean(Ibg))-Ibg;
%figure, imshow(Ibg);
%figure, imshow(I)
end
function F3 = cannyotsu(F2, n, varargin)
%Canny-revised Otsu
sigma = 2.5;
if nargin = = 3
    sigma = varargin{1};
end
lap = abs(edge(F2, "canny", [], sigma));
lap = lap/max(lap());
h = imhist(lap);
q = percentile2i(h, n);
markerimage = lap > q;
fp = F2.*uint8(markerimage);
hp = imhist(fp); %figure, plot(hp);
hp(1 : 12) = 0; %hp(2) = 0; hp(3) = 0;
%figure, bar(hp, 0)
t = otsuthresh(hp);
%hp(1:round(255*(1−t))) = 0;
%t = otsuthresh(hp);
F3 = imbinarize(F2, t);
end"
```

Data Availability

Access to data is restricted as the dataset is from a third-party company and is under commercial confidentiality.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] M. Huang, Q. Dong, F. Ni, and L. Wang, "LCA and LCCA based multi-objective optimization of pavement maintenance," *Journal of Cleaner Production*, vol. 283, p. 124583, 2021.

[2] R. Kushol, M. H. Kabir, M. S. Salekin, and A. B. M. A. Rahman, "Contrast enhancement by top-hat and bottom-hat transform with optimal structuring element: application to retinal vessel segmentation," in *Image Analysis and Recognition*, F. Karray, A. Campilho, and F. Cheriet, Eds., Springer International Publishing, New York, NY, USA, 2017.

[3] X. Bai, F. Zhou, and B. Xue, "Infrared image enhancement through contrast enhancement by using multiscale new top-hat transform," *Infrared Physics & Technology*, vol. 54, no. 2, pp. 61–69, 2011.

[4] H. Zakeri, F. M. Nejad, and A. Fahimifar, "Image based techniques for crack detection, classification and quantification in asphalt pavement: a review," *Archives of Computational Methods in Engineering*, vol. 24, no. 4, pp. 1–43, 2016.

[5] A. G. Ouyang and Y. P. Wang, "Edge detection in pavement crack image with beamlet transform," in *Proceedings of the 2nd International Conference on Electronic & Mechanical Engineering and Information Technology*, Atlantis press, Beijing, China, December 2012.

[6] Y.-C. Tsai, V. Kaul, and R. M. Mersereau, "Critical assessment of pavement distress segmentation methods," *Journal of Transportation Engineering*, vol. 136, no. 1, pp. 11–19, 2010.

[7] S. Dorafshan, R. J. Thomas, and M. Maguire, "Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete," *Construction and Building Materials*, vol. 186, pp. 1031–1045, 2018.

[8] S. F. Wang, Y. L. Che, N. Li, Z. G. Xu, and Y. S. An, "Asphalt pavement crack detection algorithm based on multi-scale ridges," *Zhongguo Gonglu Xuebao/China Journal of Highway and Transport*, vol. 30, pp. 32–41, 2017.

[9] Y. Mao-De, B. O. Shao-Bo, and H. E. Yu-Yao, "A method of image detection and analysis for pavement crack based on morphology," *Journal of Engineering Graphics*, vol. 28, 2008.

[10] A. Akagic, E. Buza, S. Omanovic et al., "Pavement crack detection using Otsu thresholding for image segmentation," in *Proceedings of the 2018 41st International Convention on Information and Communication Technology*, pp. 1092–1097, Electronics and Microelectronics (MIPRO), OPatija, Croatia, May 2018.

[11] M. Kamaliardakani, L. Sun, and M. K. Ardakani, "Sealed-crack detection algorithm using heuristic thresholding approach," *Journal of Computing in Civil Engineering*, vol. 30, no. 1, p. 10, 2016.

[12] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[13] S. Li, Y. Cao, and H. Cai, "Automatic pavement-crack detection and segmentation based on steerable matched filtering and an active contour model," *Journal of Computing in Civil Engineering*, vol. 31, no. 5, Article ID 04017045, 2017.

[14] D. Ai, G. Jiang, L. Siew Kei, and C. Li, "Automatic pixel-level pavement crack detection using information of multi-scale neighborhoods," *IEEE Access*, vol. 6, pp. 24452–24463, 2018.

[15] S. L. H. Lau, E. K. P. Chong, X. Yang, and X. Wang, "Automated pavement crack segmentation using U-net-based convolutional neural network," *IEEE Access*, vol. 8, pp. 114892–114899, 2020.

[16] S. B. Kotsiantis, "Decision trees: a recent overview," *Artificial Intelligence Review*, vol. 39, no. 4, pp. 261–283, 2013.

[17] J. R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, no. 1, pp. 81–106, 1986.

[18] S. Ruggieri, "Efficient C4.5 [classification algorithm]," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 438–444, 2002.

[19] J. C. Shafer, R. Agrawal, and M. Mehta, "SPRINT: a scalable parallel classifier for data mining," in *Proceedings of the VLDB'96, 22th International Conference on Very Large Data Bases*, Mumbai, India, September 1996.

[20] Z. Qin, C. Yu, L. Qingquan, M. Qingzhou, and W. Song, "CrackTree automatic crack detection from pavement images," *Pattern Recognition Letters*, vol. 33, no. 3, pp. 227–238, 2012.

[21] C. P. Wang Kelvin, A. Zhang, Q. Li Joshua, Y. Fei, C. Chen, and B. Li, "Deep learning for asphalt pavement cracking recognition using convolutional neural network," in *Proceedings of the Airfield and Highway Pavements*, pp. 166–177, Philadelphia, PA, USA, August 2017.

[22] A. Zhang, K. C. P. 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 4018041, 2018.

[23] J. Liu, X. Yang, and S. Lau, "Automated pavement crack detection and segmentation based on two-step convolutional neural network," *Computer-Aided Civil and Infrastructure Engineering*, vol. 35, no. 11, pp. 1291–1305, 2020.

[24] A. Zhang, Q. Li, and K. C. P. Wang, "Matched filtering algorithm for pavement cracking detection," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 2367, no. 1, pp. 30–42, 2013.

[25] R. C. Gonzalez and P. Wintz, *Digital Image Processing*, Addison-Wesley, Boston, MA, USA, 2010.