

Received 2 May 2024, accepted 17 May 2024, date of publication 21 May 2024, date of current version 29 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3403893

## RESEARCH ARTICLE

# Pavement Crack Classification and Recognition Algorithm Combined With Tensor Voting and RANSAC

HAIYING WANG<sup>1</sup>, JIANYU TAO<sup>1</sup>, AND HANSHENG LI<sup>2</sup>

<sup>1</sup>Key Laboratory of Road Construction Technology and Equipment of MOE, Chang'an University, Xi'an 710064, China

<sup>2</sup>Shaanxi Transportation Holding Group Company Ltd., Xi'an 710075, China

Corresponding author: Haiying Wang (whyng@chd.edu.cn)

This work was supported in part by the Traffic Research Project of the Department of Transport of Shaanxi Province under Grant 18-33X and Grant 21-04X.

**ABSTRACT** The existing methods for pavement crack classification and identification solely offer information about the crack type, neglecting size and direction details, which are essential for guiding repair efforts and forming the engineer digital information data. In response to the challenges posed by insufficient crack information, prolonged training time and intricate parameter adjustment inherent in employing deep learning algorithms for pavement crack classification and recognition, we propose an integrated approach combining tensor voting with the random sample consensus for pavement crack classification and recognition. The method involves pre-processing road images using gray value transformation and the K-Means clustering algorithm. Subsequently, the tensor voting algorithm is applied to enhance the linear features, resulting in the generation of linear saliency maps of cracks along with crack junction information. Furthermore, a non-maximum suppression method and the RANSAC algorithm are employed to refine and fit the crack skeleton curves respectively, accomplishing the crack classification and recognition. The outcomes demonstrate that the proposed integrated approach in the crack skeleton segmentation algorithm yields an average F1-score of 0.7879, outperforming traditional non-maximum suppression methods. The accuracy of crack classification and recognition reaches 96%, outperforming other crack classification and recognition algorithms grounded in digital image processing methods. Compared with the neural networks employed for classification and recognition, the proposed algorithm is able to capture direction and size details of cracks, which can provide guidance for intelligent crack repair. This additional information can offer valuable guidance for intelligent crack repair processes.

**INDEX TERMS** Crack classification and recognition, non-maximum suppression, random sample consensus (RANSAC), tensor voting.

## I. INTRODUCTION

The 2022 *Statistical Bulletin on the Development of the Transportation Industry*, released by the Ministry of Transport of China, reveals that as of the end of 2022, the total length of highways in China has reached 5.3548 million kilometers, resulting in an increasingly demanding road maintenance workload. Among various surface defects,

The associate editor coordinating the review of this manuscript and approving it for publication was Joewono Widjaja<sup>1</sup>.

cracks stand out as the most prevalent issue. Failure to promptly detect and assess these cracks, followed by necessary repairs, may contribute to the exacerbation of pavement deterioration. It has significant theoretical and practical implications to study the crack classification and recognition techniques to accomplish automatic crack detection and repair, elevating the efficiency of maintenance efforts.

Currently, the detection and recognition of road cracks generally involve the collection of road images through road inspection vehicles or unmanned aerial vehicles (UAV),

followed by the utilization of digital image processing techniques and deep learning methods for crack detection. In acquired road images, due to contrasting grayscale values between cracks and surrounding regions, digital image processing techniques can be employed by setting grayscale thresholds to achieve crack pixel detection and segmentation. Compared with the semantic segmentation network in deep learning algorithms [1], [2], [3], the threshold method exhibits lower segmentation accuracy and lacks continuity, but segments smaller cracks and saves network training time [4].

Many studies focus on enhancing the classical Otsu threshold method, K-means clustering, and other algorithms to determine the optimal gray value segmentation threshold. Li et al. [5] employed histogram equalization and linear gray value transformation to accentuate crack features, eliminate noise, and applied the Zhang-Suen refinement algorithm and connectivity domain threshold method for crack skeleton extraction and skeleton burrs removal. Vivekananthan et al. [6] utilized the Sobel filter for crack edge detection, combining max-min gray level discrimination with the Otsu method for improved segmentation accuracy. Xiong et al. [7] enhanced the pixel gray segmentation threshold in Otsu algorithm by introducing background tendency coefficients and bias parameters, resulting in superior crack segmentation compared to traditional digital image processing methods. Zhu et al. [8] initially partitioned images into blocks, and combined two-dimensional maximum entropy threshold segmentation method with Otsu algorithm for each image sub-block, realizing effective crack segmentation in identified sub-blocks with crack regions. References [9], [10], and [11] demonstrate that the optimal crack segmentation threshold was automatically determined through optimizing the initial clustering selection method of the K-Means algorithm and enhancing the algorithm's optimization capabilities, resulting in improved crack segmentation. However, despite the automatic segmentation achieved through the optimization of threshold methods, the resulting crack curves often contain noise points and face challenges in dealing with background interferences like shadows.

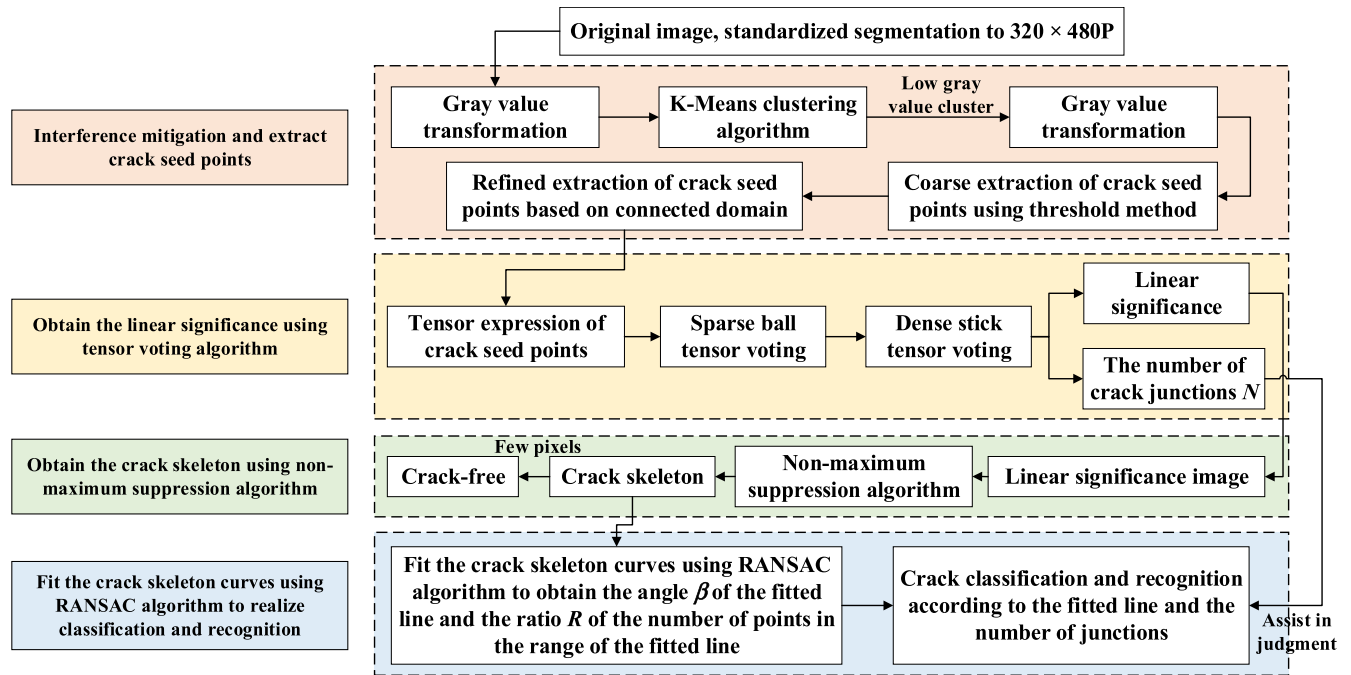
Additionally, some research focused on crack segmentation by analyzing geometric characteristics of cracks. Xu et al. [12] initially segmented cracks using the Otsu method and employed the K-Means clustering algorithm to cluster connected domain parameters of crack pixels, achieving segmentation based on connected domain characteristics. Peng et al. [13] introduced a three-threshold pavement crack segmentation method utilizing structured random forest for noise suppression and proposing a morphological operation technique to enhance crack curve continuity. Liu et al. [14] proposed an iterative tensor voting algorithm aimed at enhancing the linear characteristics of cracks, allowing for crack detection with the interference of inhomogeneous textures and complex backgrounds. Moreover, it has been observed that after the rough segmentation via traditional

threshold or clustering methods, accurate crack segmentation can be achieved by extracting crack seed points for regional growth based on crack characteristics [15], [16], [17].

In recent years, some studies have concentrated on combining digital image processing algorithms with deep learning methodologies. Zheng et al. [18] integrated a tensor voting module into semantic segmentation network, enhancing the feature map by incorporating significant domain maps generated through tensor voting. Choi et al. [19] utilized the ResNet50 network model to extract features from concrete crack images, subsequently applying the Sobel edge detection operator to amplify crack characteristics and diminish background noise, and employed the K-Means clustering algorithm to ascertain crack distribution, offering an automated and effective approach for assessing and monitoring damage in concrete structures. Luo et al. [20] combined the Canny result images with the low-level features within DeeplabV3+ network, enriching pavement crack location details and compensating for detail loss during the fusion of high-level and low-level feature layers.

Currently, deep learning methods are the primary approach for achieving classification and identification tasks related to pavement cracks, enabling the identification of whether the image contains cracks and crack types such as transverse, longitudinal, etc., by enhancing classical classification and recognition networks like AlexNet [21], [22], [23], VGG [24], [25], and ResNet [26], [27], [28], [29]. However, the utilization of deep learning methods for crack classification and recognition presents significant challenges: (1) The creation of training sets and model training demands substantial time and often requires high-end hardware systems. (2) The neural network designed for classification and recognition solely provides information about crack types. To gather additional details regarding direction and size, the use of supplementary algorithms becomes imperative, further increasing the algorithm's running time. (3) Model parameters are generally determined through iterative processes, with each parameter's practical implications not always clearly defined, making direct parameter adjustments challenging to achieve desired outcomes.

Accurate crack segmentation enables the analysis of geometric characteristics across various crack curve types, facilitating the use of digital image processing algorithms for crack classification and identification. Rodriguez-Lozano et al. [30] proposed a data dimensionality reduction method for crack classification and identification, calculating the maximum value and the difference between the maximum value by counting pixels in rows and columns of the crack binarization image, achieving fast and high-precision crack classification and identification. Sghaier et al. [31] employed crack length, width, surface area, projection attributes, and Hough attributes to determine crack severity and type, utilizing SVM for crack classification and identification. Qin et al. [32] introduced a crack classification and identification method using diagonal matching of bounding boxes, which quantified



**FIGURE 1.** Implementation pathway of the integrated pavement crack classification and recognition method using tensor voting and RANSAC.

the bounding box's aspect ratio, the number and location of crack segment intersections with the diagonal, and extracted crack geometric features by calculating intersection points, enabling crack classification and identification. Huyan et al. [33] employed the least squares method to determine crack curve inclination, initially categorized cracks as linear or mesh based on blank regions between crack pixels, and subsequently differentiated transverse and longitudinal cracks using length-to-width ratios and curve inclination.

Addressing the limitations of inconspicuous crack pixel detection in crack classification and recognition using digital image processing techniques, as well as the above-mentioned challenges associated with deep learning methods, this paper proposes an integrated approach combining the tensor voting and random sample consensus (RANSAC) for pavement crack classification and recognition. The entirety of this method relies on digital image processing techniques for crack segmentation and classification. The proposed approach eliminates the need for model training and parameter adjustment associated with deep learning algorithms. In addition to the information of crack type, the direction and size of the crack are also provided, offering guidance for intelligent road repair. This streamlining facilitates broader practical application, holding positive implications for the theoretical expansion of crack classification and recognition methodologies.

## II. PROPOSED ALGORITHM

The process of implementing the pavement crack classification and recognition method, which combines the tensor voting and RANSAC algorithms as proposed in this paper,

is illustrated in Fig. 1. This method initiates by employing grayscale value transformation and the K-Means clustering algorithm to extract crack seed points. Subsequently, the tensor voting algorithm is applied to acquire the linear saliency map of these seed points, followed by the introduction of a non-maximum suppression algorithm to obtain the crack skeleton diagram. Lastly, the RANSAC algorithm is employed to fit the crack skeleton diagram, culminating in the achievement of crack classification and recognition.

### A. INTERFERENCE MITIGATION AND EXTRACT CRACK SEED POINTS

Under natural illumination conditions, roadside trees or buildings cast shadows on the road surface. Due to the similarity in pixel grayscale values between these shadows and cracks, direct application of a threshold method for crack seed point extraction can often lead to the capture of a significant number of shadowed area pixels. This, in turn, can introduce interference to subsequent tensor voting algorithm. To address shadow interference, the conventional approach involves the segmentation of shadow regions, followed by subsequent brightness compensation. Huyan et al. [33] separately applied brightness compensation to the umbra and penumbra regions through the integration of illumination compensation models and interpolation techniques, effectively achieving shadow elimination. Li et al. [34] devised an efficient single-image shadow elimination algorithm based on hierarchical clustering, successfully mitigating the impact of shadows. However, it is important to note that images subjected to regional brightness compensation might exhibit unnatural transitions at shadow boundaries.

To mitigate shadow interference in pavement crack images, this paper introduces a combined approach involving gray value transformation and the K-Means clustering algorithm. The specific steps are as follows:

#### 1) GRAY VALUE TRANSFORMATION

After grayscale the pavement image (the gray value range is 0~255), calculate the maximum value  $m$  and minimum value  $n$  of the gray value of all pixels, and transform the gray value according to (1).

$$\text{Dst}(i, j) = \begin{cases} \frac{400\text{Src}(i, j)}{m-n} - \frac{400n}{m-n}, & n \leq \text{Src}(i, j) \leq \frac{m+n}{2} \\ \frac{110\text{Src}(i, j)}{m-n} + \frac{145m-255n}{m-n}, & \frac{m+n}{2} < \text{Src}(i, j) \leq m \end{cases} \quad (1)$$

where  $\text{Src}(i, j)$  represents the gray value of the pixel at the coordinate position of  $(i, j)$  in the original gray image, and  $\text{Dst}(i, j)$  is the gray value of the pixel at the coordinate position of  $(i, j)$  after the gray value transformation.

#### 2) K-MEANS CLUSTERING ALGORITHM

The K-Means clustering algorithm is used to segment a pavement image with shadow interference into pavement background areas and shadow areas. The main steps are as follows:

Step 1: Calculate the average  $\mu$  and the standard deviation  $\sigma$  of the gray values of all pixels, and divide the image into two clusters: low gray value and high gray value cluster. Initialize the iteration center of the low gray value cluster as  $\mu - \sigma$  (set to 50 if the  $\mu - \sigma$  is less than 0), and that of the high gray value cluster as  $\mu + \sigma$  (set to 200 if the  $\mu + \sigma$  is greater than 255).

Step 2: Based on the current iteration center, calculate the distance between each pixel's gray value and the iteration center of the two clusters (that is, the absolute difference between gray values), and assign each pixel to the nearest cluster.

Step 3: Calculate the average gray value of all pixels in each cluster, as the new iteration center for the next iteration.

Step 4: Calculate the distance between the two group of cluster centers obtained in Steps 2 and 3. If the distance is less than the given iteration accuracy (taken as 0.01), end the iteration, and record the coordinates of the pixels in the low gray value cluster at this time, otherwise repeat Steps 2 to 4.

#### 3) COARSE EXTRACTION OF CRACK SEED POINTS

Following K-Means clustering algorithm, a secondary gray value transformation is performed in the low gray value cluster utilizing (1), further enhancing contrast and emphasizing distinguishing features.

Crack seed points are coarsely extracted using a thresholding method. Calculate the average  $\mu_L$  and standard deviation  $\sigma_L$  of all pixels in the low gray value cluster after the secondary gray value transformation, and extract pixels with

gray values less than  $\mu_L - 2\sigma_L$  as coarse extraction result of crack seed points.

#### 4) REFINED EXTRACTION OF CRACK SEED POINTS

After obtaining the coarsely extracted crack seed points, we aimed to discern whether the image contained cracks and optimize the seed points extraction. Initially, the morphological close operation was applied to the coarsely extracted crack seed points to connect adjacent points into regions. Subsequently, all connected domains in the image were quantified, and the small connected domain suppressing method was employed to remove areas with fewer pixels than a specified threshold (which should be adjusted based on the image resolution, set at 500 in this paper). The remaining connected domains comprised the Region of Interest (RoI) of the crack seed points. Only the crack seed points within the RoI were retained as the refined extraction result of the crack seed points. For images lacking obvious cracks, the remaining crack seed points are typically few and dispersed, making them candidates for elimination in subsequent tensor voting algorithms. For images containing cracks, the noise in the refined crack seed points was suppressed.

By meticulously following these steps, the proposed method effectively eliminates the impact of shadow and noise interference in pavement crack images. Moreover, it is suitable for image pre-processing under various lighting conditions.

### B. TENSOR VOTING ALGORITHM

In this section, the tensor voting algorithm is applied to determine the linear and ball salience of crack seed points. Tensor voting is a classic image enhancement algorithm used to infer implicit structural features from noisy dot maps, finding broad application in image denoising and feature extraction domains [14], [18], [35], [36], [37], [38]. This approach enhances the identification and significance of linear features within the crack patterns, thereby contributing to improved accuracy and reliability in subsequent crack skeleton extraction and classification recognition procedures.

#### 1) TENSOR REPRESENTATION AND VOTING FIELD

In a two-dimensional image, the tensor of a pixel can be represented using a non-negative definite second-order symmetry matrix. Let the eigenvalues of the matrix be  $\lambda_1, \lambda_2$ , and  $\lambda_1 \geq \lambda_2$ , corresponding to their respective unit eigenvectors  $e_1, e_2$ , as presented in Table 1. There are three fundamental types of tensors: the unit ball tensor, the unit stick tensor and the ellipse tensor.

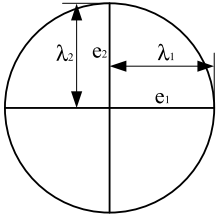
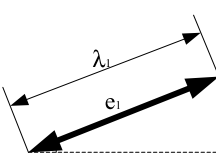
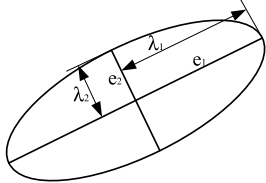
An ellipse tensor  $T$  can be represented as a superposition of the ball tensor and the stick tensor, and it can be decomposed according to its eigenvalues, as shown in (2).

$$T = \lambda_1 e_1 e_1^T + \lambda_2 e_2 e_2^T \quad (2)$$

Equation (2) can be further decomposed, as shown in (3).

$$T = (\lambda_1 - \lambda_2) e_1 e_1^T + \lambda_2 (e_1 e_1^T + e_2 e_2^T) \quad (3)$$

TABLE 1. Fundamental types of tensors.

Tensors	Unit ball tensor	Unit stick tensor	Ellipse tensor
Characteristics	There is no orientation information.	There is orientation information. Let the unit vector of the normal direction be $[n_x, n_y]^T$ .	Superimposition of ball tensors and stick tensors
Second-order matrix representation	$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$	$\begin{bmatrix} n_x^2 & n_x n_y \\ n_x n_y & n_y^2 \end{bmatrix}$	$\begin{bmatrix} a & c \\ c & b \end{bmatrix}$ , where $a, b, c$ are any positive numbers satisfying $ab - c^2 \geq 0$ .
Eigenvalues	$\lambda_1 = \lambda_2 = 1$	$\lambda_1 = 1, \lambda_2 = 0$	$\lambda_1 > \lambda_2 > 0$
Geometric representation			

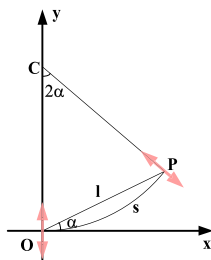


FIGURE 2. Diagram of the stick voting field.

where  $(\lambda_1 - \lambda_2) e_1 e_1^T$  represents a stick tensor, and  $(\lambda_1 - \lambda_2)$  signifies its linear saliency, and  $e_1$  represents its normal direction. Additionally,  $\lambda_2(e_1 e_1^T + e_2 e_2^T)$  represents a ball tensor, and  $\lambda_2$  signifies its ball saliency.

During the process of tensor voting, each individual pixel conveys directional and saliency information to neighboring pixels through a voting field. The voting field assumes two primary forms: stick voting field and ball voting field.

a: Stick voting fields

As illustrated in Fig. 2, consider point  $O$  as the voting point with its tangent direction along the  $x$ -axis and its normal direction along the  $y$ -axis. Let point  $P$  be the recipient of the vote, and let point  $C$  denote the center of the circle passing through the point  $O$  and  $P$ , with  $O$ 's normal direction as the radius direction.

The tensor voting field intensity produced by point  $O$  at  $P$  is determined by the distance between the two points. A greater distance corresponds to a weaker voting field intensity. The decay coefficient  $DF(s, k, \delta)$  of voting field intensity with increasing distance between the voting and recipient points is governed by the degradation function depicted in (4).

$$DF(s, k, \delta) = \exp\left(-\frac{s^2 + ck^2}{\delta^2}\right) \tag{4}$$

where  $\delta$  represents the sole parameter determining the scale of the voting field;  $s, k$  and  $l$  are the length of the arc, the curvature of the arc and the length of the line between points  $O$  and  $P$ , respectively.  $s = \alpha l / \sin\alpha$  and  $k = 2\sin\alpha / l$ , where  $\alpha$  is the angle between the line  $OP$  and the positive direction of the  $x$ -axis.  $c$  is the parameter controlling the rate of decay, determined by  $\delta, c = -16\log[0.1(\delta - 1)] / \pi^2$ .

The voting field intensity  $S$  of point  $O$  at any given point  $P$  is determined by (5).

$$S = \begin{cases} DF(s, k, \delta) \begin{bmatrix} -\sin(2\alpha + \theta) \\ \cos(2\alpha + \theta) \end{bmatrix} \\ [-\sin(2\alpha + \theta), \cos(2\alpha + \theta)], |\alpha| \leq 45^\circ \\ 0, |\alpha| > 45^\circ \end{cases} \tag{5}$$

where  $\theta$  represents the angle between the normal direction vector of point  $O$  and the positive direction of the  $y$ -axis. The scenario illustrated in Fig. 2 corresponds to when  $\theta = 0$ .

b: Ball voting fields

By rotating the stick voting field around the center point  $O$  at regular small intervals and superimposing the field intensities, a ball voting field can be obtained.

2) SPARSE BALL TENSOR VOTING

To initially obtain the direction and linear saliency of crack seed points, a sparse ball tensor voting process is performed. All crack seed points obtained from the threshold segmentation in Section II-A are initialized as spherical tensors without directional information. A ball voting field is formed at each of these points, and the values of the ball voting fields are calculated and superimposed. Subsequently, the tensors of



**TABLE 2. Classification of crack seed points based on eigenvalues.**

Eigenvalues		Categories
$\lambda_1 - \lambda_2 \geq \lambda_2$	$\lambda_2 \approx 0$	Points on the crack curve (not turning points)
	$\lambda_2 \gg 0$	Turning points on the crack curve
$\lambda_1 - \lambda_2 < \lambda_2$	$\lambda_1 \approx \lambda_2 > 0$	Junctions on the crack curve
	$\lambda_1 \approx \lambda_2 \approx 0$	Scatters (noise points)

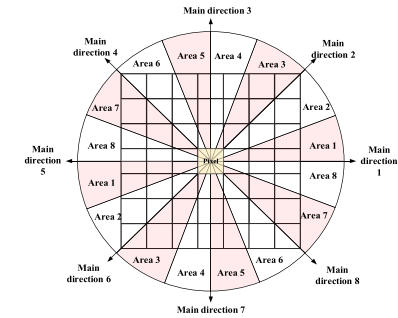
the non-crack seed points in the image are reset to  $\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$ .

The tensors resulting from the sparse ball voting process are subjected to the eigenvalue decomposition as shown in (3), yielding the linear saliency,  $\lambda_1 - \lambda_2$ , and the ball saliency,  $\lambda_2$ , for each pixel. Based on different eigenvalues, the seed points are categorized into four classes as shown in Table 2. Seed points with linear saliency lower than ball saliency ( $\lambda_1 - \lambda_2 < \lambda_2$ ) exhibit weaker linear features and are discarded to emphasize the linear characteristics of crack curves.

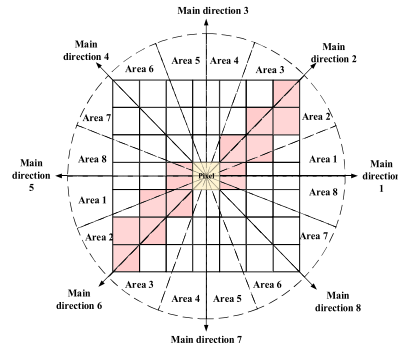
### 3) DENSE STICK TENSOR VOTING

The dense stick tensor voting is employed to further refine the linear saliency of crack seed points and simultaneously connect breakpoints between cracks. Initially, the ball saliency  $\lambda_2$  of the residual crack seed points with relatively high linear saliency from the sparse ball tensor voting is set to zero, transforming them into stick tensors with well-defined directional information, where the normal direction vector is denoted as  $e_1$ . Subsequently, stick voting field corresponding to its respective direction is established at each pixel, and the values of the stick voting fields are calculated and superimposed. After performing dense stick tensor voting, eigenvalue decomposition as shown in (3) is applied to the obtained tensors, yielding the linear saliency,  $\lambda_1 - \lambda_2$ , and ball saliency,  $\lambda_2$ , of each pixel. To minimize erroneous connections between breakpoints, normalized values of the linear saliency,  $\lambda_1 - \lambda_2$ , which less than 0.2 are set to zero. A linear saliency map is generated based on the values of  $\lambda_1 - \lambda_2$ .

As indicated in Table 2, the linear saliency,  $\lambda_1 - \lambda_2$  of crack intersections is approximately 0. This implies that at these intersection points, the linear saliency value is likely to be diminished and may be eliminated after performing sparse ball tensor voting. Consequently, critical information concerning the junctions of crack curves might be missing from the linear saliency map. The ball saliency,  $\lambda_2$ , of crack junctions typically exceeds that of noise points ( $\lambda_2 \approx 0$ ). Thus, using the value of  $\lambda_2$ , a distinction can be made between crack junctions and noise points. Nevertheless, portions with larger values might also correspond to turning points in the crack curves. In addressing this, the approach outlined in reference [37] is adopted. First, the ball saliency values,  $\lambda_2$ , of all retained crack seed points are normalized, and those exceeding 0.6 are extracted. Subsequently, connected domain analysis is conducted, and for each connected domain, the following steps are executed:



(a) Diagram of all neighborhoods of a pixel



(b) Diagram of adjacent points of a pixel

**FIGURE 3. Diagram of neighborhoods and adjacent points of a pixel.**

Step 1: Calculate the centroid coordinates of the connected domain and round them to the nearest integers. In the linear saliency map, draw a circular region with a radius of 10 pixels around this centroid point. Record the linear saliency,  $\lambda_1 - \lambda_2$ , of the pixels that the circular edge traverses, sequentially arrange them into an array.

Step 2: Count the number of local maxima points in the array. If the count is 2, then the connected domain is identified as a turning point in the crack curve. If the count is greater than 2, then it is determined to be a junction of crack curves.

Using the above method, the number of junctions of crack curves, denoted as  $N$ , is obtained. This value serves as an important parameter for the subsequent crack classification and recognition algorithm.

### C. NON-MAXIMUM SUPPRESSION ALGORITHM

The linear saliency map obtained from tensor voting is refined using a non-maximum suppression algorithm to obtain the crack skeleton. The crack skeleton is a one-pixel-wide curve that delineates the direction of the crack curve. The non-maximum suppression algorithm described in reference [38] employed an eight-neighborhood comparison approach. It searches along the pixel's normal direction and the normal direction  $+180^\circ$  for two adjacent pixels. If the linear saliency,  $\lambda_1 - \lambda_2$ , of the pixel is greater than or equal to the values of the two adjacent pixels, it is retained; otherwise, it is discarded. However, the crack skeleton generated using this method often exhibits numerous "spikes." To address this, a subsequent step of small connected domain

suppression is applied to remove the spikes. However, small connected domain suppression might inadvertently remove some short cracks, resulting in incomplete crack extraction. Therefore, building upon the eight-neighborhood comparison non-maximum suppression algorithm, this paper broadens the scope of pixel neighborhood searches to mitigate the occurrence of spikes. The specific method for determining adjacent pixels is as follows:

#### 1) DETERMINE THE NEIGHBORHOOD OF A PIXEL

All possible neighborhoods of a pixel are illustrated in Fig. 3(a). A pixel possesses eight main directions, and 16 areas are divided with each of the main direction rotating 22.5° clockwise. Symmetric areas about the pixel's center are merged into a single area. The area where the pixel's normal direction vector resides is considered the neighborhood of the pixel.

#### 2) DETERMINE THE K-RADIUS NEIGHBORHOOD OF A PIXEL

The square region formed by extending  $K$  pixel lengths in each of the four directions of top, bottom, left and right is called the  $K$ -radius neighborhood of a pixel, and the area size is  $(2K + 1) \times (2K + 1)$ . As depicted in Fig. 3(a), the square region illustrates the 3-radius neighborhood of the pixel.

#### 3) DETERMINE THE ADJACENT POINTS OF THE PIXEL IN THE K-RADIUS NEIGHBORHOOD

Within a pixel's  $K$ -radius neighborhood, points lying on the eight main direction lines, as well as points located one pixel distance away from each main direction line, are identified as all adjacent points of the pixel. As shown in Fig. 3(b), the red dots represent all adjacent points of the pixel in area 2. According to this rule, the number of adjacent points in different areas within the pixel's  $K$ -radius neighborhood is equal to  $4K$ .

The steps of the non-maximum suppression algorithm are as follows:

Step 1: Calculate the angle between the normal direction vector,  $e_1$ , and the main direction 1 of all pixels in the linear saliency map. Based on the angle, determine the neighborhood for each pixel. Specify  $K$  value to establish the  $K$ -radius neighborhood of each pixel, and further identify all adjacent points for each pixel.

Step 2: Compare the linear saliency,  $\lambda_1 - \lambda_2$ , of each pixel with its respective adjacent points, if the  $\lambda_1 - \lambda_2$  value of the pixel is greater than or equal to the value of all its adjacent points, the pixel is retained; otherwise, it is discarded.

Step 3: Display all retained pixels to obtain a crack skeleton image.

#### D. RANSAC ALGORITHM

The crack skeletons obtained in section II-C possess pronounced linear characteristics. Images are deemed crack-free when the number of pixels in the crack skeleton image falls below 500. In cases where cracks are present, cracks can be categorized into four types: transverse cracks, longitudinal

TABLE 3. Pavement crack classification and recognition rules.

Crack types	Classification criteria
Non-crack	The number of pixels in the crack skeleton image falls below 500.
Transverse cracks	$R \geq 80\%$ , angle $\beta$ with the range of $(-\pi/4, \pi/4)$ .
Longitudinal cracks	$R \geq 80\%$ , angle $\beta$ with the range of $[\pi/4, \pi/2] \cup [-\pi/2, -\pi/4]$ .
Block cracks	$R < 80\%$ , the number of crack junctions $N$ with the range of $N \leq 3$ .
Alligator cracks	$R < 80\%$ , the number of crack junctions $N$ with the range of $N > 3$ .

TABLE 4. Camera parameters of the UAV.

Parameters	Values
Type of sensors	CMOS, global shutter
Equivalent focal length of lens	24 mm
Target size	4/3 inch (18.5 mm × 13.5 mm)
Effective resolution	20 million pixels
Maximum frame rate	30 fps@1920 × 1080
Shutter speed	125 $\mu$ s~8 s
Maximum flight speed	15 m/s
Flight altitude	2 m

cracks, block cracks and alligator cracks. To facilitate classification and recognition, the crack skeleton is subjected to curve fitting using the RANSAC algorithm. The specific steps are as follows:

Step 1: Establish the coordinate origin of the crack skeleton image at the bottom left corner, with the x-axis aligned horizontally and the y-axis aligned vertically.

Step 2: Randomly select two points (ensuring that the Euclidean distance between the two points is required to be greater than  $\max\{\text{image height } h, \text{image width } w\}/3$ ), and connect them into a straight line.

Step 3: Set the threshold value  $s = \max\{h, w\}/10$ . Translate the straight line upwards and downwards by distance  $s$ , and count the number of points falling within the range enclosed by the two translated lines.

Step 4: Repeat Step 2 to Step 3  $n$  ( $n = \text{total number of pixel points of the crack skeleton image} / 3$ ) iterations. Identify the straight line with the highest count of enclosed points within the translation range as the final fitted straight line. Record the coordinates of the two points through which the fitted line passes, denoted as  $(x_1, y_1)$  and  $(x_2, y_2)$ . Calculate the angle  $\beta$  between the fitted line and the positive x-axis using the formula  $\beta = \arctan[(y_2 - y_1)/(x_2 - x_1)]$  (with a range of values calculated by the arctan function is  $[-\pi/2, \pi/2]$ ). Calculate the ratio  $R$ , which is the ratio of the count of points enclosed within the translation range of the line to the total number of pixels in the crack skeleton image.

Step 5: Determine the type of pavement cracks according to the angle  $\beta$ , ratio  $R$  and the number of crack junctions  $N$  obtained in section II-B.3. The specific classification rules are provided in Table 3. The threshold for each criterion may vary depending on the specific context and dataset.

III. EXPERIMENT RESULTS AND DISCUSSION

The proposed method, which combines tensor voting with the RANSAC algorithm for pavement crack classification and recognition, was implemented using Matlab R2018a software. We performed the experiments on a computer with a 3.2GHz CPU, 16GB RAM and NVIDIA RTX 3060 GPU. The pavement crack images were collected by the UAV, and the camera parameters of the UAV are shown in Table 4.

The imaging field of view of the camera is determined by (6) and (7).

$$FOV_H = \frac{H \cdot WD}{f} \tag{6}$$

$$FOV_W = \frac{W \cdot WD}{f} \tag{7}$$

where,  $FOV_H$  represents the imaging field of view of the camera along the flight direction of the UAV, while  $FOV_W$  represents the imaging field of view of the camera perpendicular to the flight direction of the UAV.  $H$  and  $W$  represent the vertical and horizontal target size of the selected camera, with  $H$  being 13.5 mm and  $W$  being 18.5 mm.  $WD$  represents the working distance of the camera, indicating the distance between the photographed object and the camera lens, approximately being the flight altitude 2 m.  $f$  represents the focal length of the selected lens, which is 24 mm.

The equation (6) and (7) yield  $FOV_H = 1542$  mm and  $FOV_W = 1125$  mm, indicating that the real-world size of the image captured by the camera is 1542 mm × 1125 mm. This size corresponds to the camera resolution of 1920 × 1080 pixels. Consequently, the size of the real object corresponding to each pixel in the image is approximately 1.0 mm × 0.8 mm. Given this resolution, the smallest detectable crack size in the image is about 1 mm, which fulfills the requirements for most pavement crack detection scenarios.

To capture images without pixel ghosting, the maximum flight speed of the UAV, denoted as  $v_{max}$ , is determined by (8).

$$v_{max} = \frac{\mu_c}{t_{min}} \tag{8}$$

where,  $\mu_c$  represents the minimum actual size represented by a pixel in the image, taken as 1mm.  $t_{min}$  represents the minimum exposure time of the camera, set as 1.25 μs according to Table 4.  $v_{max}$  is calculated as 8 m/s or 28.8 km/h.

When a road inspection vehicle is equipped with industrial cameras with similar parameters, it can capture a clear and comprehensive image of pavement cracks by ensuring that the vehicle’s speed remains below 30 km/h.

To enhance the algorithm’s processing speed and facilitate its verification, the collected images are resized and cropped. The resolution of the pavement crack images used in this paper is 480 × 320 pixels.

A. RESULTS OF GRAY VALUE TRANSFORMATION

As illustrated in Fig. 4, the gray value transformation method proposed in Section II-A.1 was applied to asphalt pavement

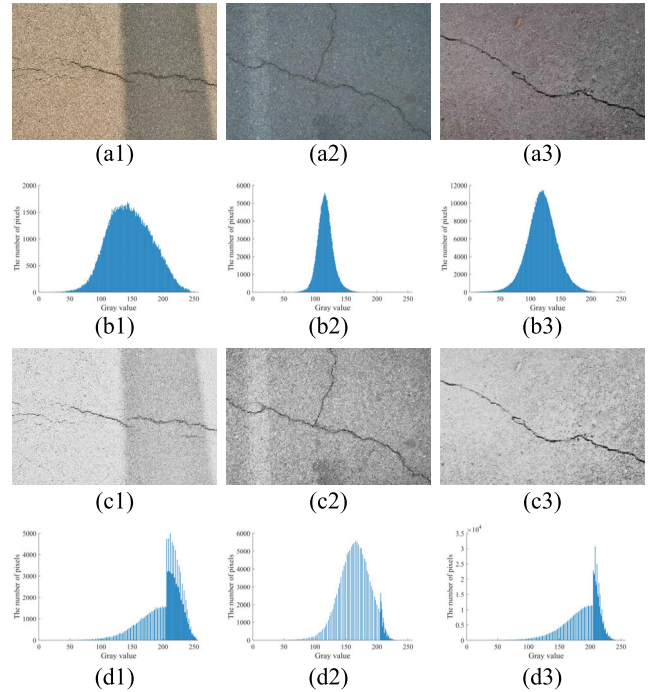


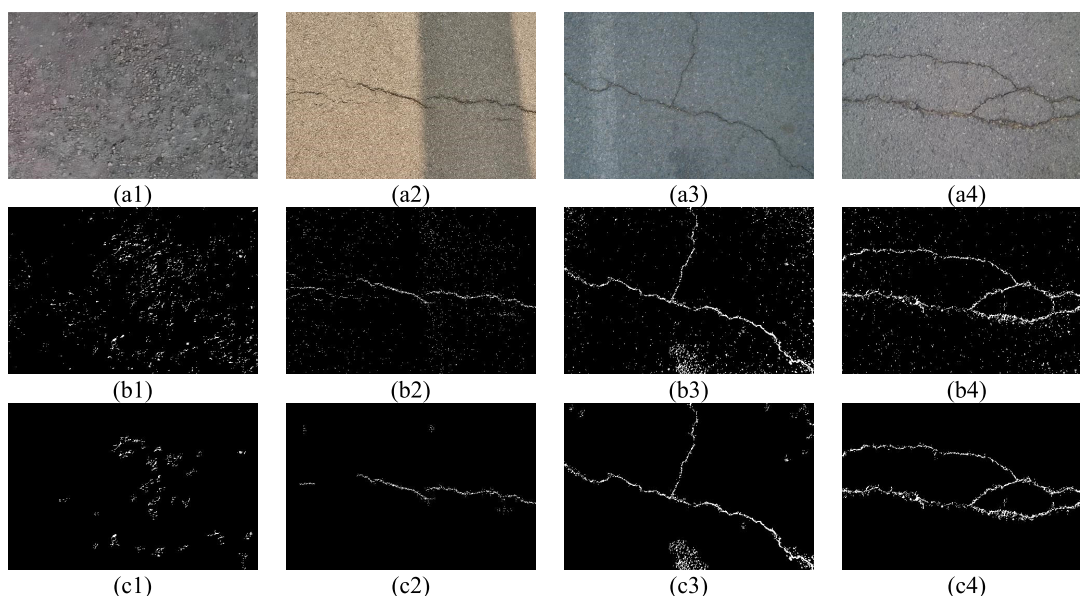
FIGURE 4. Result of gray value transformation concerning pavement cracks under varying lighting conditions and materials. (a1), (a2) and (a3): Original pavement images. (b1), (b2) and (b3): Gray histograms of grayscale images before the transformation. (c1), (c2) and (c3): Grayscale images after the transformation. (d1), (d2) and (d3): Gray histograms of grayscale images after the transformation.

cracks under sunny conditions (Fig. 4(a1)), asphalt pavement cracks under cloudy conditions (Fig. 4(a2)), and cement pavement cracks under cloudy conditions (Fig. 4(a3)). Subsequently, the effectiveness of the gray value transformation method was analyzed concerning pavement cracks under varying lighting conditions and materials based on the gray histogram before and after the transformation.

From the perspective of different lighting conditions, images may exhibit shadows and a broad range of gray values under good sunlight on sunny days. In contrast, the distribution range of gray values narrows on cloudy days, leading to low contrast between cracks and the surrounding background. As depicted in Fig. 4(c1), (c2), and (c3), the contrast of cracks is significantly improved following the gray value transformation under varying lighting conditions. Fig. 4(d1), (d2), and (d3) illustrate that the gray value distribution range of the image is expanded through stretching, resulting in an increased number of pixels with high gray values and a wider gray value gap with crack pixels. This stretching process proves convenient for the extraction of crack seed points.

From the perspective of different pavement materials, the gray value distribution range in asphalt pavement images is narrower than that in cement pavement images. In asphalt pavement, the majority of pixels fall within the range of [100, 150] in terms of gray value (in Fig. 4(b2)), resulting in lower contrast for cracks and making crack seed point extraction more challenging compared to cement pavement. However,





**FIGURE 5.** Results of crack seed points extraction. (a1), (a2), (a3) and (a4): Original pavement crack images. (b1), (b2), (b3) and (b4): Results of coarse extraction of crack seed points. (c1), (c2), (c3) and (c4): Results of refined extraction of crack seed points.

after applying the gray value transformation, the gray value range of most pixels in asphalt pavement crack images shifts to [150, 170] (in Fig. 4(d2)), distinctly differentiating them from low-gray-value crack pixels. This observation demonstrates the effectiveness and versatility of the gray value transformation method proposed in this paper across various lighting conditions and pavement materials.

### B. RESULTS OF CRACK SEED POINTS EXTRACTION

The results of crack seed points extraction using the method proposed in Section II-A are illustrated in Fig. 5.

As depicted in Fig. 5, the coarsely extracted crack seed points exhibit considerable noise interference points, particularly noticeable in images without cracks. This is primarily due to the direct extraction of pixels by the K-Means clustering algorithm and threshold method, leading to misidentifications in crack-free images. In comparison, the refined extracted crack seed points show a notable reduction in noise points, resulting in significantly fewer seed points in crack-free images as well, which can be further eliminated using the following tensor voting algorithm.

### C. RESULTS OF CRACK SKELETON SEGMENTATION

Fig. 6 illustrates the crack skeleton extraction results of crack seed points in Fig. 5(c1), (c2), (c3) and (c4). Fig. 6(e1), (e2), (e3) and (e4) depict crack skeleton images manually labeled using the Image Labeler App in Matlab, serving as ground truth references for crack skeleton segmentation. Within the tensor voting algorithm, the scale determining parameter  $\delta$  for the voting field is set as 15. In the context of the non-maximum suppression algorithm, the parameter  $K$  is selected as 7.

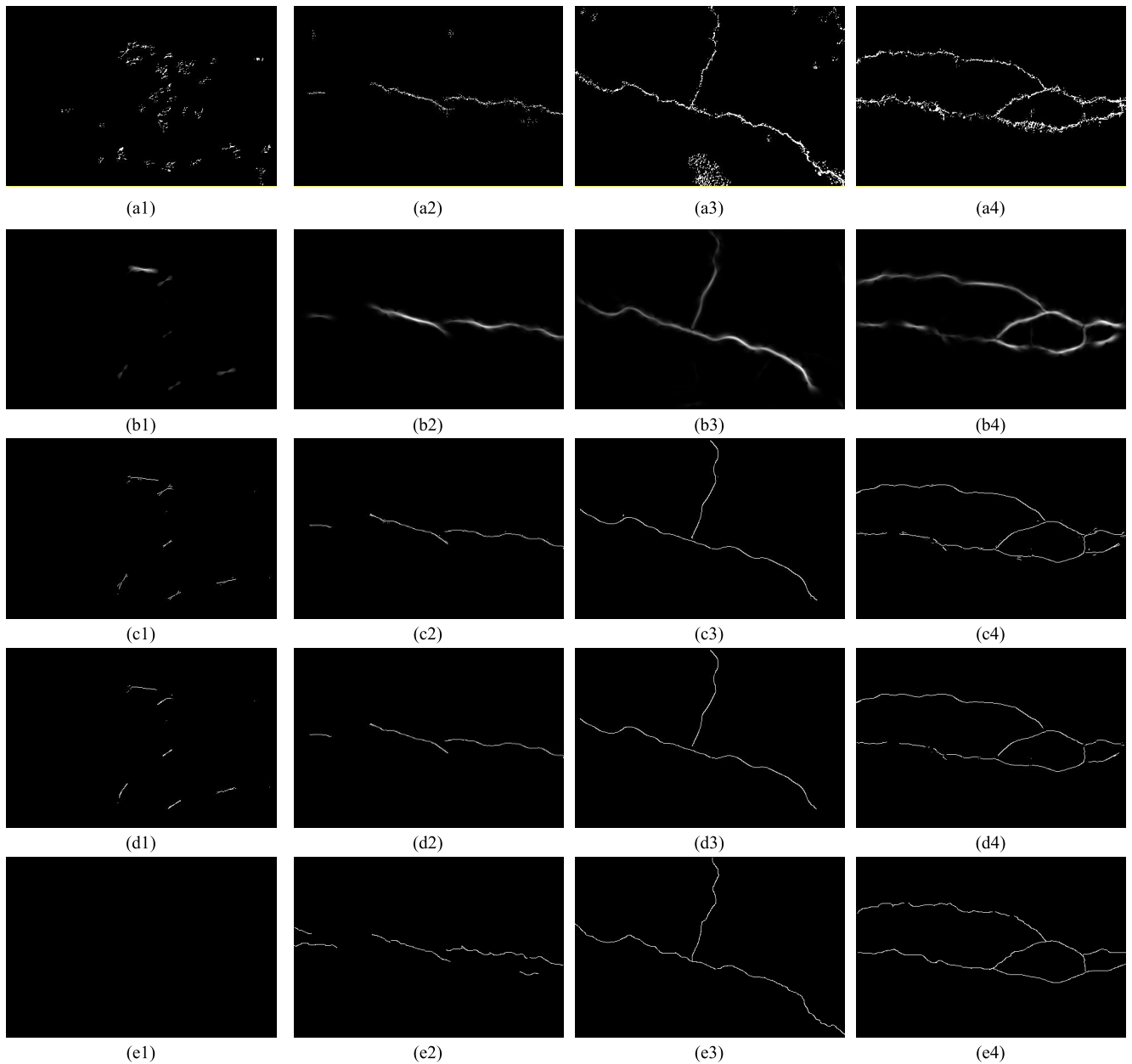
From Fig. 6, it can be observed that the non-maximum suppression algorithm proposed in this paper diminishes the occurrence of “spikes” and false crack recognition phenomena by expanding the search range to surrounding pixels.

The recall rate (abbreviated as  $Re$ ), precision rate (abbreviated as  $Pre$ ) and F1-score (abbreviated as  $F1$ ) are used as the quantitative evaluation metrics for crack skeleton segmentation performance [39].  $Re$  is defined as the ratio of correctly extracted crack pixels by the algorithm to all the manually labeled crack pixels.  $Pre$  is defined as ratio of correctly extracted crack pixels by the algorithm to all crack pixels extracted by the algorithm.  $F1$  represents the comprehensive evaluation of  $Re$  and  $Pre$ , and is calculated by (9).

$$F1 = 2 \frac{Re \cdot Pre}{Re + Pre} \quad (9)$$

These evaluation metrics are adopted to assess the skeleton segmentation results of 20 pavement crack images. Given the presence of errors in the manually labeled crack skeleton images, the morphological dilation operation with a  $5 \times 5$  square structuring element was conducted on the manual labeling and the algorithmically extracted crack skeleton images. The  $Re$ ,  $Pre$  and  $F1$  statistics for the crack skeleton segmentation algorithms from reference [38] and this paper are illustrated in Fig. 7.

The crack skeleton segmentation algorithm in this paper achieves an average  $Re$  of 0.7455, an average  $Pre$  of 0.8407, and an average  $F1$  of 0.7879. As can be seen from Fig. 7, the  $Re$  of the algorithm utilizing eight-neighborhood comparison non-maximum suppression in literature [38] is slightly higher than that of the method proposed in this paper, mainly because the crack skeleton segmented by the method in literature [38] contains more pixels. The  $Pre$  and  $F1$  of proposed algorithm



**FIGURE 6.** Comparison of pavement crack skeleton segmentation results. (a1), (a2) (a3), and (a4): Crack seed points. (b1), (b2) (b3), and (b4): Linear saliency maps. (c1), (c2), (c3) and (c4): Crack skeleton images using non-maximum suppression algorithm in literature [38]. (d1), (d2), (d3) and (d4): Crack skeleton images using non-maximum suppression algorithm proposed in this paper. (e1), (e2), (e3) and (e4): Crack skeleton images manually labeled.

in this paper demonstrate a substantial improvement over those reported in the literature [38]. The augmentation of the search range around each pixel within the non-maximum suppression algorithm contributes to the enhancement of segmentation precision.

Every pixel within the crack curve segmented by the tensor voting algorithm carries orientation information. Consequently, the crack skeleton map can be derived through a straightforward non-maximum suppression algorithm that integrates the pixel orientation information. The pixel orientation information of the crack is illustrated in Fig. 8. The

red arrows in Fig. 8 indicate the normal direction of the crack pixels, which can provide guidance for automatic repair work.

In the case of the semantic segmentation neural network, the resulting segmented crack only encompasses the crack curve without pixel orientation information. Consequently, additional image thinning algorithms are requisite to acquire the crack skeleton map, leading to an additional increase in algorithm's running time.

Each pixel in the crack curve segmented by tensor voting algorithm contains orientation information. Therefore, the crack skeleton map can be obtained by a simple

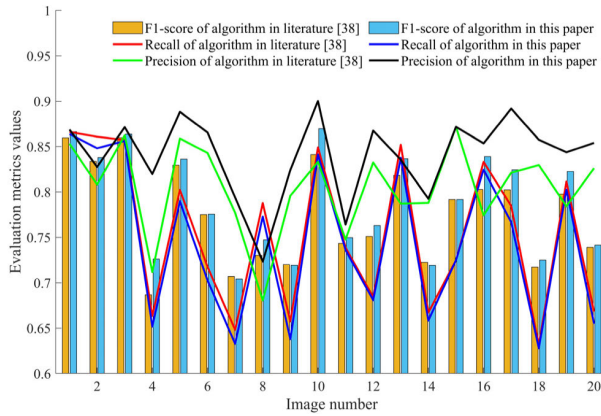


FIGURE 7. Evaluation metrics comparison of two non-maximum suppression algorithms.

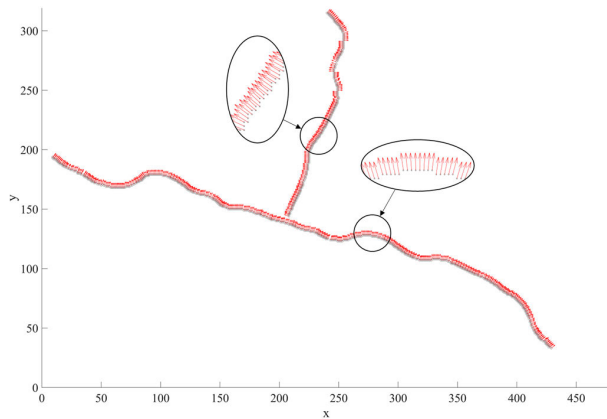
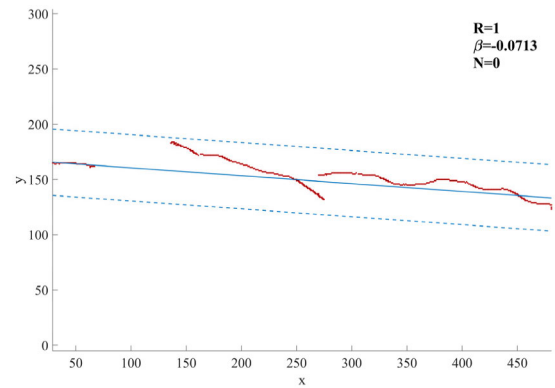


FIGURE 8. Diagram for the pixel orientation information of the crack in Fig. 6(d3).

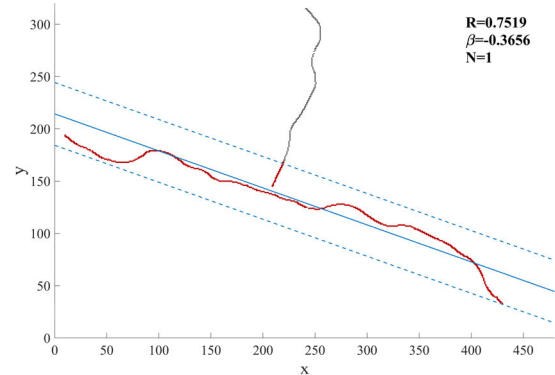
non-maximum suppression algorithm by combining the pixel orientation information. Upon acquiring the crack skeleton, the pixel count of the crack skeleton serves as a representation of the pixel length of the crack. Additionally, the pixel width of the crack can be denoted by the distance from the normal direction of the edge pixels to the crack skeleton curve, given the known pixel direction. However, obtaining pixel width information for the crack curve segmented by the semantic segmentation network proves challenging, as it lacks inherent direction information.

**D. FITTING RESULTS OF CRACK SKELETON CURVES**

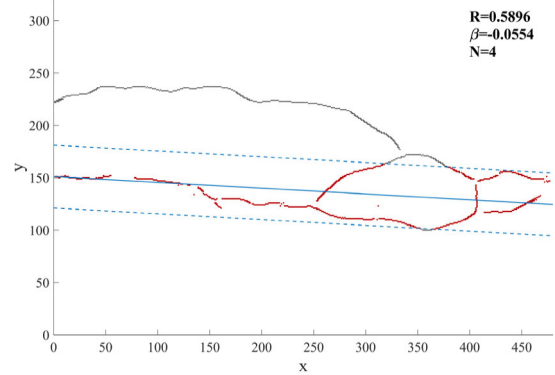
Following the generation of crack skeleton images through tensor voting and non-maximum suppression algorithms, the RANSAC algorithm is employed for linear fitting of the crack skeletons. The fitting outcomes are illustrated in Fig. 9, where (a), (b) and (c) correspond to the fitting outcomes for the three crack skeleton sets shown in Fig. 6(d2), (d3) and (d4). In the figures, the blue line represents the final fitted line, while the blue dashed lines indicate the two translation lines of the fitted line. Red dots denote crack pixels falling within the range of the two translation lines, while black dots



(a) Fitting results of crack skeleton in Fig. 6(d2).



(b) Fitting results of crack skeleton in Fig. 6(d3).



(c) Fitting results of crack skeleton in Fig. 6(d4).

FIGURE 9. Fitting results of crack skeleton curves using the RANSAC algorithm.

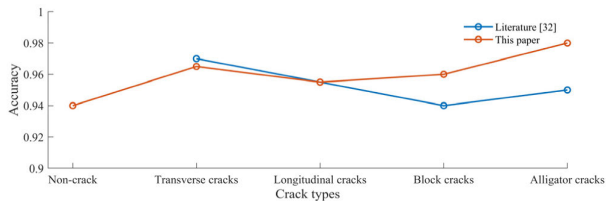
signify crack pixels not falling within this range. In accordance with the pavement crack classification criteria outlined in Table 3, the classification results for Fig. 9(a), (b), and (c) are respectively identified as transverse cracks, block cracks, and alligator cracks.

**E. ACCURACY COMPARISON AND DISCUSSION**

We present the results of our experiments in terms of crack classification and recognition accuracy, and comparison with existing methods. For five types of pavement cracks (non-crack, transverse cracks, longitudinal cracks, block cracks and alligator cracks), 40 images of each type, and a total of 200 images were selected. Initially, the crack

**TABLE 5.** Running times of various algorithms for different tasks.

	Crack segmentation	Crack thinning	Crack classification	Total running time
Algorithms in our method	Tensor voting (in Section II.B) 1.08s	Non-maximum suppression (in Section II.C) 0.17s	RANSAC (in Section II.D) 0.21s	1.46s
Classical algorithms	Semantic segmentation network (U-Net in literature [40]) 0.79s	Zhang-Suen thinning algorithm in literature [5] 0.32s	Classification network (VGG16 in literature [24]) 0.26s	1.37s

**FIGURE 10.** Comparison of crack classification and recognition accuracy using algorithm proposed in this paper and literature [32].

skeleton images were obtained using the methodologies outlined in Sections II-A-II-C of this paper. Subsequently, the classification and recognition effectiveness of the pavement crack classification algorithm based on RANSAC fitting, as described in Section II-D was compared with the square bounding box algorithm proposed in Literature [32]. The precise identification counts for each of the four crack types were individually tallied, and the recognition accuracy was computed. The results of this comparison are graphically presented in Fig. 10.

Fig. 10 reveals that the two methods demonstrate similar recognition accuracy for transverse and longitudinal cracks. However, for block cracks and alligator cracks, due to the accurate estimation of crack junctions enabled by the tensor voting algorithm, the recognition accuracy using the proposed method in this paper is notably higher. The comprehensive recognition precision of the proposed algorithm reaches 96%, effectively fulfilling essential usage requirements. The proposed method demonstrates its capability to accurately classify different types of pavement cracks and effectively distinguish them from other features in the images.

#### F. RUNNING TIME COMPARASION AND DISCUSSION

To gather additional details regarding direction and size, the use of supplementary algorithms becomes imperative for classification neural network. The running times of various algorithms are presented in Table 5. Notably, the scale determining parameter  $\delta$  in tensor voting plays a crucial role in determining the algorithm's running time, and in this particular implementation, the parameter  $\delta$  is set to 15 according to the processed image size.

As can be seen in Table 5, the tensor voting algorithm exhibits a slightly longer running time when compared to the classical semantic segmentation network U-Net. However, its subsequent non-maximum suppression algorithm requires only a single traversal of the image to obtain the

crack skeleton map, leveraging the known orientation information of the crack pixels. In contrast, other image refinement algorithms necessitate multiple traversals, resulting in longer running times. Concerning crack classification, the RANSAC algorithm does not differ significantly from the classical VGG16 classification network. The crack classification and recognition algorithm presented in this paper processes an image in approximately 1.5s on average, meeting the requirements of the practical project.

Compared with the classification recognition neural network, the proposed algorithm exhibits a notable advantage in acquiring not only crack type information but also crack direction and size details. This additional information can provide valuable guidance and recommendations for hierarchical management and automated repair of cracks. Furthermore, the proposed algorithm eliminates the need for extensive dataset creation and model training. It also demands less hardware resources, making it suitable for integration into embedded systems, thereby facilitating widespread practical application in engineering. All parameters in the algorithm possess explicit practical significance, providing a well-defined approach for parameter adjustment in practical applications.

#### G. LIMITATIONS AND FUTURE WORK

Despite the promising results, our method may have limitations when dealing with handling highly complex crack patterns or certain challenging lighting conditions, and the classification accuracy is slightly imperfect compared with deep learning methods. We discuss potential pathways for future research to further enhance the performance and applicability of the proposed approach. In practical application, the camera's shooting height, angle, and range can be customized based on the specific road surface conditions to capture more standardized images. Furthermore, adjustments to the algorithm's parameters can be made to align with the real-world context, thereby enhancing recognition accuracy.

#### IV. CONCLUSION

In this paper, we propose an integrated approach combining the tensor voting and RANSAC for pavement crack classification and recognition. The classification and recognition algorithm proposed in this paper is grounded on the accurate segmentation of crack skeletons, allowing simultaneous acquisition of crack type, orientation and size information. The feature information fusion of points and lines enables effective classification and recognition of cracks, with high



recognition accuracy that satisfies the road maintenance and management usage requirements. The method shows promising performance on automatic crack repair and provides a solid foundation for further research and practical application.

## V. DATA AVAILABILITY

The dataset used for comparison of crack classification and identification results is available at <https://github.com/0110TJY/Crack-Classification-Dataset>.

## REFERENCES

- [1] Z. Al-Huda, B. Peng, R. N. A. Algburi, M. A. Al-Antari, R. AL-Jarazi, and D. Zhai, "A hybrid deep learning pavement crack semantic segmentation," *Eng. Appl. Artif. Intell.*, vol. 122, Jun. 2023, Art. no. 106142, doi: 10.1016/j.engappai.2023.106142.
- [2] Z. Zhou, J. Zhang, and C. Gong, "Hybrid semantic segmentation for tunnel lining cracks based on Swin transformer and convolutional neural network," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 38, no. 17, pp. 2491–2510, Nov. 2023, doi: 10.1111/mice.13003.
- [3] M. A.-M. Khan, S.-H. Kee, and A.-A. Nahid, "Vision-based concrete-crack detection on railway sleepers using dense U-Net model," *Algorithms*, vol. 16, no. 12, p. 568, Dec. 2023, doi: 10.3390/a16120568.
- [4] A. Rezaie, R. Achanta, M. Godio, and K. Beyer, "Comparison of crack segmentation using digital image correlation measurements and deep learning," *Construct. Building Mater.*, vol. 261, Nov. 2020, Art. no. 120474, doi: 10.1016/j.conbuildmat.2020.120474.
- [5] Z. Li, C. Yin, and X. Zhang, "Crack segmentation extraction and parameter calculation of asphalt pavement based on image processing," *Sensors*, vol. 23, no. 22, p. 9161, Nov. 2023, doi: 10.3390/s23229161.
- [6] V. Vivekananthan, R. Vignesh, S. Vasanthaseelan, E. Joel, and K. S. Kumar, "Concrete bridge crack detection by image processing technique by using the improved Otsu method," *Mater. Today, Proc.*, vol. 74, pp. 1002–1007, Feb. 2023, doi: 10.1016/j.matpr.2022.11.356.
- [7] L. Xiong, Q. Qian, and S. Zhang, "Bridge crack detection method based on improved Otsu," *Semicond. Optoelectron.*, vol. 44, no. 6, pp. 965–971, Dec. 2023, doi: 10.16818/j.issn1001-5868.2023080101.
- [8] F. Zhu, T. Yao, Z. Zeng, P. Dong, and C. Miao, "Experimental on concrete beam crack identification based on digital image," *J. Nanjing Tech. Univ. Nat. Sci. Ed.*, vol. 45, no. 5, pp. 537–545, Sep. 2023, doi: 10.3969/j.issn.1671-7627.2023.05.008.
- [9] B. Hang, "A method for asphalt pavement crack segmentation based on improve K-means algorithm," *J. Highway Transp. Res. Develop.*, vol. 40, no. 4, pp. 1–8, Apr. 2023, doi: 10.3969/j.issn.1002-0268.2023.04.001.
- [10] M. Duan, Z. Li, and A. Cui, "Study on segmentation method of pavement crack image based on improved t- distribution mixture model," *J. Highway Transp. Res. Develop.*, vol. 39, no. 7, pp. 23–29, Jul. 2022, doi: 10.3969/j.issn.1002-0268.2022.07.004.
- [11] W. Chen, C. He, C. Ji, M. Zhang, and S. Chen, "An improved K-means algorithm for underwater image background segmentation," *Multimedia Tools Appl.*, vol. 80, no. 14, pp. 21059–21083, Jun. 2021, doi: 10.1007/s11042-021-10693-7.
- [12] G. Xu, T. Zhao, S. Jiang, and D. Gao, "Extraction method of structural surface cracks based on multiple connected domain features," *J. Huazhong Univ. Sci. Tech. Nat. Sci. Ed.*, vol. 47, no. 10, pp. 52–55+68, Oct. 2019, doi: 10.13245/j.hust.191010.
- [13] C. Peng, M. Yang, Q. Zheng, J. Zhang, D. Wang, R. Yan, J. Wang, and B. Li, "A triple-thresholds pavement crack detection method leveraging random structured forest," *Construct. Building Mater.*, vol. 263, Dec. 2020, Art. no. 120080, doi: 10.1016/j.conbuildmat.2020.120080.
- [14] K. Liu, H. Yan, K. Meng, H. Chen, and H. Sajid, "Iterating tensor voting: A perceptual grouping approach for crack detection on EL images," *IEEE Trans. Autom. Sci. Eng.*, vol. 18, no. 2, pp. 831–839, Apr. 2021, doi: 10.1109/TASE.2020.2988314.
- [15] G. Yin, J. Gao, J. Gao, C. Li, M. Jin, M. Shi, H. Tuo, and P. Wei, "Crack identification method of highway tunnel based on image processing," *J. Traffic Transp. Eng. English Ed.*, vol. 10, no. 3, pp. 469–484, Jun. 2023, doi: 10.1016/j.jtte.2022.06.006.
- [16] X. Zhang and T. Bao, "Identification and segmentation algorithm of small cracks on dam surface based on local Otsu threshold and regional growth," *Water Resour. Power*, vol. 40, no. 2, pp. 97–100, Feb. 2022.
- [17] P. Li, Q. Li, W. Ma, and W. Jiang, "Pavement crack segmentation based on K-means clustering," *Comput. Eng. Design*, vol. 41, no. 11, pp. 3143–3147, Nov. 2020, doi: 10.16208/j.issn1000-7024.2020.11.023.
- [18] W. Zheng, X. Jiang, Z. Fang, and Y. Gao, "TV-Net: A structure-level feature fusion network based on tensor voting for road crack segmentation," *IEEE Trans. Intell. Transp. Syst.*, early access, Jan. 25, 2024, doi: 10.1109/TITS.2023.3334266.
- [19] Y. Choi, H. W. Park, Y. Mi, and S. Song, "Crack detection and analysis of concrete structures based on neural network and clustering," *Sensors*, vol. 24, no. 6, p. 1725, Mar. 2024, doi: 10.3390/s24061725.
- [20] J. Luo, H. Lin, X. Wei, and Y. Wang, "Adaptive Canny and semantic segmentation networks based on feature fusion for road crack detection," *IEEE Access*, vol. 11, pp. 51740–51753, 2023, doi: 10.1109/ACCESS.2023.3279888.
- [21] M. He, Z. Liang, Y. Wang, J. Yan, X. Qing, and F. Wang, "Quantitative diagnosis of fatigue hole-edge crack in composite laminates using guided wave and optical fiber feature fusion," *Measurement*, vol. 225, Feb. 2024, Art. no. 114052, doi: 10.1016/j.measurement.2023.114052.
- [22] X. Liu, Z. Hong, W. Shi, and X. Guo, "Image-processing-based subway tunnel crack detection system," *Sensors*, vol. 23, no. 13, p. 6070, Jun. 2023, doi: 10.3390/s23136070.
- [23] R.-S. Rajadurai and S.-T. Kang, "Automated vision-based crack detection on concrete surfaces using deep learning," *Appl. Sci.*, vol. 11, no. 11, p. 5229, Jun. 2021, doi: 10.3390/app11115229.
- [24] Y. Que, Y. Dai, X. Ji, A. K. Leung, Z. Chen, Z. Jiang, and Y. Tang, "Automatic classification of asphalt pavement cracks using a novel integrated generative adversarial networks and improved VGG model," *Eng. Struct.*, vol. 277, Feb. 2023, Art. no. 115406, doi: 10.1016/j.engstruct.2022.115406.
- [25] S. T. Ramachandiraiah, P. Kumar, S. K. Pasupunuri, and J. R. Shinganmakki, "Evaluation of pavement surface distress using image processing and artificial neural network," *J. Test. Eval.*, vol. 51, no. 4, pp. 2041–2056, Jul. 2023, doi: 10.1520/jte20220259.
- [26] D. Li, Z. Duan, X. Hu, D. Zhang, and Y. Zhang, "Automated classification and detection of multiple pavement distress images based on deep learning," *J. Traffic Transp. Eng. English Ed.*, vol. 10, no. 2, pp. 276–290, Apr. 2023, doi: 10.1016/j.jtte.2021.04.008.
- [27] X. Li, X. Xu, X. He, X. Wei, and H. Yang, "Intelligent crack detection method based on GM-ResNet," *Sensors*, vol. 23, no. 20, p. 8369, Oct. 2023, doi: 10.3390/s23208369.
- [28] J. Zhang and T. Bao, "An improved ResNet-based algorithm for crack detection of concrete dams using dynamic knowledge distillation," *Water*, vol. 15, no. 15, p. 2839, Aug. 2023, doi: 10.3390/w15152839.
- [29] A. Dipankar and S. K. Suman, "Pavement crack detection based on a deep learning approach and visualisation by using GIS," *Int. J. Pavement Eng.*, vol. 24, no. 1, Feb. 2023, Art. no. 2173754, doi: 10.1080/10298436.2023.2173754.
- [30] F. J. Rodriguez-Lozano, J. C. Gámez-Granados, J. M. Palomares, and J. Olivares, "Efficient data dimensionality reduction method for improving road crack classification algorithms," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 38, no. 16, pp. 2339–2354, May 2023, doi: 10.1111/mice.13014.
- [31] S. Sghaier, M. Krichen, I. Ben Dhaou, H. Elmannai, and R. Alkanhel, "Identification, 3D-reconstruction, and classification of dangerous road cracks," *Sensors*, vol. 23, no. 7, p. 3578, Mar. 2023, doi: 10.3390/s23073578.
- [32] G. Qin, L. Huang, and S. Yang, "Classification of pavement crack types based on square bounding box diagonal matching method," *Neural Comput. Appl.*, vol. 34, no. 15, pp. 13125–13132, Aug. 2022, doi: 10.1007/s00521-020-04929-0.
- [33] J. Huyan, W. Li, S. Tighe, R. Deng, and S. Yan, "Illumination compensation model with k-means algorithm for detection of pavement surface cracks with shadow," *J. Comput. Civil Eng.*, vol. 34, no. 1, Jan. 2020, Art. no. 04019049, doi: 10.1061/(asce)cp.1943-5487.0000869.
- [34] W. Li, X. Geng, W. Sun, and Y. Li, "Single image shadow removal algorithm simulation based on hierarchical clustering," *Comput. Simul.*, vol. 37, no. 6, pp. 471–474, Jun. 2020.
- [35] H. Li, K. Zhang, B. Pan, P. Long, and J. Liu, "Automatic extraction of geometric information of tunnel surrounding rock and its engineering application," *J. Highway Transp. Res. Develop.*, vol. 40, no. 12, pp. 155–162, Dec. 2023, doi: 10.3969/j.issn.1002-0268.2023.12.018.

- [36] X. Cui, X. Huang, J. Yang, D. Zhang, X. Chen, and K. Li, "Seismic discontinuity enhancement method based on tensor voting," *Oil Geo. Prospect.*, vol. 56, no. 6, pp. 1351–1358, Dec. 2021, doi: [10.13810/j.cnki.issn.1000-7210.2021.06.018](https://doi.org/10.13810/j.cnki.issn.1000-7210.2021.06.018).
- [37] Y. Wang, Y. Huang, and W. Huang, "Crack junction detection in pavement image using correlation structure analysis and iterative tensor voting," *IEEE Access*, vol. 7, pp. 138094–138109, 2019, doi: [10.1109/ACCESS.2019.2942318](https://doi.org/10.1109/ACCESS.2019.2942318).
- [38] Y. Li, "Research and implementation of bridge crack detection technology based on tensor voting," M.S. thesis, Dept. School Telecommun. Eng., Xidian Univ., Xi'an, China, 2020.
- [39] L. Deng, H. Zhe, L. Long, W. Wang, X. Kong, and R. Cao, "Review of deep learning-based crack detection for civil infrastructures," *China J. Highway Transp.*, vol. 36, no. 2, pp. 1–21, Feb. 2023, doi: [10.19721/j.cnki.1001-7372.2023.02.001](https://doi.org/10.19721/j.cnki.1001-7372.2023.02.001).
- [40] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI*, Munich, Germany, 2015, pp. 234–241.



**HAIYING WANG** received the M.S. degree in computer-aided design and the Ph.D. degree in mechanical engineering from Chang'an University, Xi'an, China, in 1996 and 2003, respectively. She is currently an Associate Professor with the School of Construction Machinery, Chang'an University. Her research interests include mechatronics and intelligent control, machine vision, and engineering information management.



**JIANYU TAO** received the B.S. degree in mechanical electronics engineering from Chang'an University, Xi'an, China, in 2021, where he is currently pursuing the master's degree with the School of Construction Machinery.

His research interests include machine vision and engineering information management.



**HANSHENG LI** received the double B.S. degree from the Northwest University of Political Science and Law, Xi'an, China, and Fort Hays State University, Hays, KS, USA, in 2019.

He is currently an Assistant Engineer with Shaanxi Transportation Holding Group Company Ltd., Xi'an. He has published three journal articles in the related pavement materials performance and maintenance management fields. His research interests include procurement and management of highway materials, and engineering information management.

...