



# An Automatic Image Processing Algorithm Based on Crack Pixel Density for Pavement Crack Detection and Classification

Nima Safaei<sup>1,4</sup> · Omar Smadi<sup>1</sup> · Arezoo Masoud<sup>2</sup> · Babak Safaei<sup>3</sup>

Received: 6 April 2020 / Revised: 28 December 2020 / Accepted: 24 January 2021 / Published online: 2 June 2021  
© The Author(s), under exclusive licence to Chinese Society of Pavement Engineering 2021

## Abstract

Nowadays, there is a massive necessity to develop fully automated and efficient distress assessment systems to evaluate pavement conditions with the minimum cost. Due to having complex training processes, most of the current supervised learning-based practices in this area are not suitable for smaller, local-level projects with limited resources. This paper aims to develop an automatic crack assessment method to detect and classify cracks from 2-D and 3-D pavement images. A tile-based image processing method was proposed to apply a localized thresholding technique on each tile and detect the cracked ones (tiles containing cracks) based on crack pixels' spatial distribution. For longitudinal and transverse cracking, a curve is then fitted on the cracked tiles to connect them. Next, cracks are classified, and their lengths are measured based on the orientation axes and length of the crack curves. This method is not limited to the pavement texture type, and it is cost-efficient as it takes less than 20 s per image for a commodity computer to generate results. The method was tested on 130 images of Portland Cement Concrete (PCC) and Asphalt Concrete (AC) surfaces; test results were found to be promising (Precision = 0.89, Recall = 0.83,  $F_1$  score = 0.86, and Crack length measurement accuracy = 80%).

**Keywords** Pavement management system · Crack detection and classification · Image processing · Adaptive thresholding

## 1 Introduction

Roads play a critical role in economic boom and expansion, and they have significant societal benefits. Road networks facilitate mobility and connectivity; they provide people with easy access to employment, social, health, and education services [1]. Thus, road infrastructure is considered one of the most crucial parts of all public assets. Pavement surfaces are subjected to deterioration and fatigue stresses, which lead to pavement surface distress [2]. Factors such as aging, traffic loads, environmental conditions, material properties [3], pavement thickness, and pavement strength

[4] play a significant role in the deterioration of the pavements [5]. Cracks that appear on the pavement surface are one of the early signs of pavement distress issues. Due to the rapid propagation of cracks, they lead to most of the pavement failure issues. Cracks reduce the local hardness and cause blockage of the material [6]. Untreated pavement cracks rapidly spread with aging, traffic growth, and severe environmental conditions and threaten the road's life-cycle performance [7–9]. Pavement performance directly impacts the maintenance costs [10] and safety; pavement repair and maintenance strategies are crucial steps for ensuring the health and safety of roadway systems and reducing traffic noise [11–14]. Due to the significant amount of resources and budget spent annually on pavement repair and maintenance projects, reducing the inspection-related expenditures and maintenance costs is a high priority for highway agencies. If cracks are detected early during the deterioration process, it can prevent further damage and failure [15] and reduce maintenance costs. Overall, having a fast, robust, and cost-effective algorithm for detecting pavement surface defects is one of the most critical priorities for having a robust pavement management system [8, 16–18].

✉ Nima Safaei  
nima-safaei@uiowa.edu

<sup>1</sup> Institute for Transportation, Iowa State University, Ames, IA, USA

<sup>2</sup> Tippie College of Business, University of Iowa, Iowa City, IA, USA

<sup>3</sup> Civil and Environmental Engineering Department, Michigan State University, East Lansing, MI, USA

<sup>4</sup> Present Address: Tippie College of Business, University of Iowa, Iowa City, IA, USA

Using digital image processing algorithms for image characterization and quantification is the current state-of-art in many transportation areas [19–21]. More specifically, in pavement research, numerous papers have introduced different methods to detect and analyze surface defects with the help of digital images [22–25]. A digital image is composed of numerous picture elements (pixels) that create a holistic representation of an object. The area of each pixel is equivalent to about  $\frac{1}{96}$  inch ( $\sim 0.26$  mm). Pixels are represented by discrete numeric values for their intensity. The resolution of an image is described in the pixels displayed in each inch of the image.

In general, there are five categories of distresses defined by The Distress Identification Manual for the Long-term Pavement Performance Program (LTPP): 1—Cracking; 2—Patching and potholes; 3—Surface deformations; 4—Surface defects; 5—Miscellaneous distresses [26]. In detecting pavement surface anomalies, cracks are highly important because they are the pioneer signs of deterioration trends. Pavement distress assessments are usually performed with manual, semi-automated, or fully automated approaches. In manual distress assessments, surveyors conduct a visual examination of pavement distresses by merely observing the pavement surface from the windshield of a moving vehicle or walking along the pavement surface and recording any surface defects. In semi-automated assessments, a point-and-trace manual method is used at the workstation to analyze the distress images individually, determine the type, and quantify the cracks' extent and severity [27]. The mentioned approach is not time-efficient for network-level evaluation and massive projects.

On the other hand, fully automatic distress assessments use image processing tools to identify and quantify the pavement surface distresses. After applying the method to the input images, raters conduct quality control surveys to test the software functionality and perform quality assurance [27]. Two primary factors that affect pavement image processing results are the calibration of distress identification procedures and the type of the recorded distress [28]. Generally, the automatic detection approaches can be categorized into four major groups [29–31]:

1. Filtering and thresholding-based algorithms are among the most popular crack detection methods due to their simplicity and efficiency [32]. Histogram-based methods use a thresholding algorithm on a histogram analysis either with the Gaussian hypothesis or adaptive or local thresholding. This method's basis is that the background pavement pixels can be separated from the cracks based on global level statistics; however, this idea does not seem to be promising as it is not always true. Edge extraction is a filter-based method that can operate by filtering either with a fixed or adaptive scale.

Using fixed scales for the edge extraction is not suitable for pavement crack detection because the crack width changes along the crack length. Researchers use wavelet detections with adaptive filterings like contourlets, finite impulse response filter (FIR), Gabor's filters, or algorithms based on partial differential equation (PDE) models for solving the problem. Safaei et al. introduced the use of Gaussian cumulative density function as an adaptive threshold in the weighted neighborhood pixels segmentation algorithm to obviate the issue of using fixed thresholds in noisy environments [31]. There are also other methods used by researchers that are based on autocorrelation filtering. In this method, targets of the original image are compared with the targets that represent cracks. Next, a similarity score is calculated, which acts as a criterion for deciding about the crack existence inside an image [29, 33]. Some other algorithms are based on texture analysis. In these methods, cracks are considered as noises inside the image texture. F. Roli defined Conditional Texture Anisotropy (CTA) in 1996 as a measure to distinguish the crack orientation from other directions [34]. In 2009, Nguyen et al. calculated CTA values by taking pixel intensity's mean and standard deviation into account to separate crack pixels from non-crack pixels [35]. The primary disadvantage of using the threshold and filter-based methods is the dilemma of how to select an appropriate threshold value for feature extraction. Secondly, the results produced by such methods have low accuracy due to shadows; shadow pixels have similar brightness to crack pixels, so they are usually mistaken as cracks [36].

2. Approaches based on morphological operations were introduced by N. Tanaka et al. [37] in 1998; they described cracks based on mathematical morphology and stated that cracks could be represented as progressions of linear saddle points. Although this definition is partially true, it is not completely clear. In most cases, there is a prerequisite of an initial thresholding method for implementing the mathematical morphological algorithm. In comparison with histogram-based analyses, this method has a better performance in avoiding false-positive detections. However, these algorithms' primary caveat is that the detection quality significantly depends on the parameter choice, so they are considerably limited in the application phase [29].
3. Machine learning-based approaches have caught lots of attention during recent years as the image data size has increased significantly. Machine learning-based methods [38–40] have extensively contributed to the research in pavement crack detection. Neural networks are the core of trainable algorithms for crack detection; they have proven to have superiorities over the threshold-based techniques and morphological tools. As mentioned pre-

viously, threshold-based methods try to identify crack pixels from their background pixels by choosing appropriate thresholds. However, there is not such a requirement for learning-based methods. In 2002, Liu et al. [41] applied a Support Vector Machine (SVM) algorithm for detecting pavement cracks. Making robust classifiers by manipulating the training data is the core burden of ML-based crack detection methods. Manual labeling is a core part of data training and validation; it is an error-prone procedure that must be handled with extreme diligence [36]. The main disadvantage of learning-based methods is that building the learning step is usually accompanied by a thorny manual labeling step, which does not provide a fast and fully automatic analysis. In addition, the algorithm typically needs to be trained on each sub-image, and similar to the non-training-based methods, they have difficulties detecting complete crack curves over the entire image [32].

4. The purpose of model-based approaches is to incorporate the local and global properties of a crack into the algorithm. It is accomplished by either performing a multi-scale texture analysis combined with a minimal path algorithm or combining geodesic contours to detect interesting local points. The minimal path selection method was first proposed by Kass et al. [42] in 1988. The algorithm extracts simple open curves from the image using both endpoints of the curve. This method is utterly reliant on the pre-knowledge about the curve endpoints. Kaul et al. [43] proposed a novel approach in 2012 that could detect the same image structures without the need for comprehensive prior information about the topology of contours and curve endpoints. Later in 2014 and 2015, R. Amhaz et al. proposed an improved algorithm that selects curve endpoints and minimal paths at the local and global scale, respectively [44, 45]; this prevented false detection problems associated with assimilating loops. In 2006, Paar et al. [46] proposed a novel model-based crack detection technique using the idea of the line tracing algorithm; this algorithm premises that cracks are progressions of smaller lines that are linked together. In 2010, Yamaguchi et al. [47] modeled cracks using the fundamentals of the percolation phenomenon. Percolation is a physical model that describes the liquid permeation phenomenon. First, the method makes a seed region; next, based on the percolation process, the adjacent areas are labeled as crack regions. According to K. Chaiyasarn [48], model-based methods are strongly reliant on the operator input for building seed pixels, so a considerable amount of prior knowledge is required for crack detection. As operators may not be able to identify the seed pixels, hairline cracks might remain unrecognized.

Pavement analysis has passed a vast advancement in recent years due to the deep convolutional neural network (CNN) techniques [25]. Most recently, CNN is used for image clustering at the pixel level, and it is used for image segmentation purposes [49, 50]. In 2017, Badrinarayanan et al. presented SegNet, a deep convolutional network architecture for semantic segmentation [49]. SegNet consists of a deep 13-layer convolutional encoder-decoder architecture and is used for producing pixel-level labeling. It has proven to be efficient, based on memory usage and computational time [49].

In the area of non-crack distress detection and classification, Hoang et al. proposed an automatic method for raveling detection in asphalt pavement [51]. They extracted 34 features from pavement images using the statistical properties of image samples. They used a stochastic gradient descent logistic regression method for classifying images into two classes of raveling and non-raveling. Hadjimetriou et al. proposed an algorithm based on entropy texture descriptor. Their algorithm detects 12 pavement distress types, including potholes, patches, shoving, rutting, distortion, raveling, and bleeding from the pavement background using an adaptive entropy threshold value [52]. Also, Yousaf et al. developed a top-down strategy for detecting potholes from 2-D pavement images. They categorized pavement images into pothole/non-pothole classes using a bag of words technique. They used a scale-invariant feature transform method to develop the visual representation of pavement surfaces. Finally, they used a combination of support vector machine and graph cut segmentation method to train the words of pavement images and localize potholes, respectively [53]. Although many studies have been conducted on detecting pavement surface anomalies such as cracks, a consensus is missing on executing the best method for distress analysis of AC and PCC surfaces.

In the crack classification field, Cheng et al. [54] and Ying et al. [55] used direction indices of pixels and each direction property for characterizing cracks. By taking advantage of chromosome representation to encode different directions, they represented cracks as binary sequences [32]. Tsai et al. [56] and Lee et al. [57] classified cracks into five groups of longitudinal, transverse, diagonal, alligator, and block; they took advantage of neural network techniques and used buffers to identify crack regions. Xu et al. used crack seeds and clusters to label cracks as either longitudinal or transverse [58]. After normalizing pixel-intensity values, H. Oliveira et al. used block features to classify cracks into three different groups of longitudinal, transverse, and miscellaneous [59, 60]. The main drawback of these measurements is poor performance in complex textures.

This study aimed to detect and classify pavement cracks by building upon a tile-based localized thresholding

algorithm that belongs to the first group of the described methods. The main difference of the proposed method from the methods described in the first category (thresholding) is that in this study, thresholding is used as a preliminary step to focus the main analysis on the areas with higher probabilities of crack existence. This study's main analysis is based on crack pixels density and distribution inside the defined tiles, which despite having similarities with the model-based analysis, is original and does not directly belong to any groups of the described methods. Similar algorithms were previously used in [61, 62]. First, the original image was divided into tiles with pre-determined dimensions. A localized thresholding method was applied to binarize the pixel intensity values inside each tile and filter out probable non-crack pixels based on each tile's average pixel intensity value. Next, the connected components in a tile were determined, and the small objects were removed to reduce the number of non-crack pixels. The adaptive median filtering was used for the pre-processing stage; while maintaining minimal degradation of sharp crack edges, it removes background noise caused by the pavement's rough texture. Based on the size of the most massive connected pixels, polynomial evaluation and curve fitting were utilized to determine the crack existence in each tile based on the crack pixels' distribution density. Finally, section joints of the PCC surfaces were filtered out from the detected cracks using the Radon Transform algorithm. After completing the crack detection process for numerous crack types, the crack curves were drawn for longitudinal and transverse cracks, and their lengths were measured. Based on the slopes of the crack curves, they were classified as longitudinal or transverse. Finally, to validate the results, the accuracy of the crack detection and length measurement was evaluated by calculating the precision, recall, and  $F_1$  score. Finally, the validation results of the proposed method were compared with some other state-of-the-art algorithms. This method is built upon the localized thresholding algorithms and is similar to model-based techniques; it implements a tile-based analysis and proposes shape metrics to detect and classify pavement cracks. Unlike many of the methods studied in the literature, this method is not limited to the texture type.

In contrast to the methods solely based on thresholding, this study's proposed approach is less susceptible to threshold value choices because thresholding is used merely as a preliminary step. Furthermore, this study's pixel density-based algorithm can successfully distinguish between crack and non-crack objects such as shadows. In addition, this method is incredibly cost-effective as no advanced computational system is required for the detection process. It takes less than 20 s per image for a commodity computer to generate results. A summary of the described approach is provided in Fig. 1.

## 2 Method Implementation

### 2.1 Localized Thresholding, Morphological Operations, and Pre-Processing

Localized thresholding identifies thresholds local to a specific area and identifies potential crack pixels by utilizing the classical thresholding method [62]. The original image was divided into "n" by "n" pixels, which were called tiles. "n" value is selected as 50 pixels by a trial and error process. This approach is discussed in the tile dimension optimization subsection of this paper.

The average intensity value of each tile was considered the threshold value. The threshold values were stored separately in an array and further processed for crack extraction. For each tile of the ath row and bth column, the threshold value was calculated using Eq. (1):

$$T_{(a,b)} = \frac{\sum_{i=1}^{50} \sum_{j=1}^{50} \text{Im}(((50 \times a) + i), ((50 \times b) + j))}{50 \times 50}, \quad (1)$$

Im is the image matrix,  $T$  is the threshold value,  $a, b$  is the tile row/column number,  $i, j$  is the pixel elements of the tile matrix.

Overall, this step was taken to facilitate the pixel segmentation by distinguishing the intensity values of possible crack pixels from the rest of the image. Localized thresholding reduces the inconsistency of the results to some extent. After binarizing the tile, morphological operations were applied to the binary image to fill isolated pixels (individual 1 s surrounded by 0 s). Many problems are associated with the popular average filtering method, such as the vulnerability of the average pixel values in the neighborhood against a single outlier, interpolation of new pixel values on the edges, and reduced edge sharpness. Due to the mentioned problems, an adaptive median filtering technique was implemented for the pre-processing stage. This technique preserves the image details while smoothing non-impulse noise and enhances the desired image features like dark linear features [63]. The adaptive median filtering moves pixel by pixel, and each pixel value is replaced by the median value of the adjacent pixels. It works by changing the neighborhood size during the operation and reduces malformations such as undue thinning of object boundaries [63, 64].

### 2.2 Crack Detection

In this stage, minor objects with fewer than 15 connected pixels were removed from the binary image. Typically, the crack area has the largest number of connected pixels. Using this concept, the greatest connected pixels in each tile were selected as the crack, and the rest of the pixels were filtered

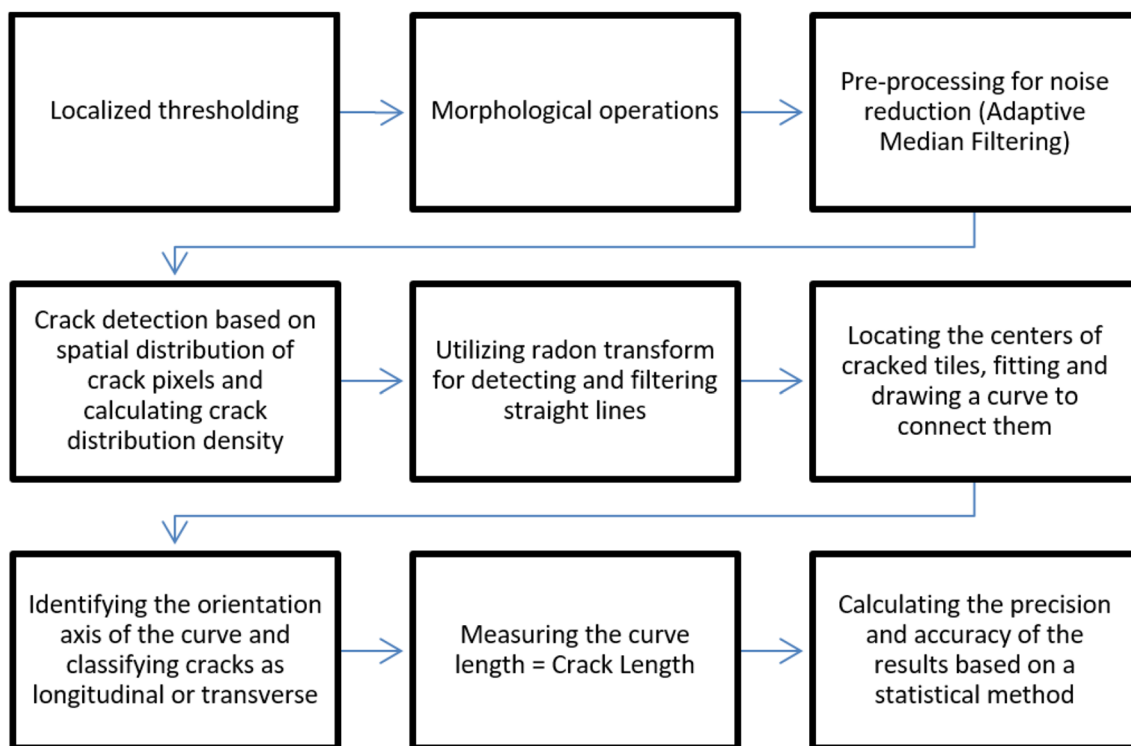


Fig. 1 Summary of approach

out. Through testing this method on 40 tile samples extracted from different AC and PCC surfaces (three examples are shown in Fig. 2), it was concluded that the method does not work well on PCC surfaces with longitudinal or transverse tining which do not contain any cracks (Fig. 2a). If these tiles had any cracks (Fig. 2d), those crack pixels would be identified as the largest connected pixels and taken as a real crack, leaving the tining regions with the smaller connected areas filtered out. However, when the tile does not contain any cracks, the method falsely selects the largest connected tining area as a crack (Fig. 2c). To solve this problem, if the largest extracted connected pixels in a tile is smaller than 175 pixels (which is usually the case for these surface types), the largest extracted component would be ignored, and the output of the previous stage would be used as the input for the future process.

In the next step, to calculate the crack distribution density, a polynomial curve was fitted on all crack pixels within a tile (Eq. (2)), and the fitting error was calculated accordingly (Eq. (3)).

$$f(x) = p_1x^n + p_2x^{n-1} + \dots + p_nx + p_{n+1} \tag{2}$$

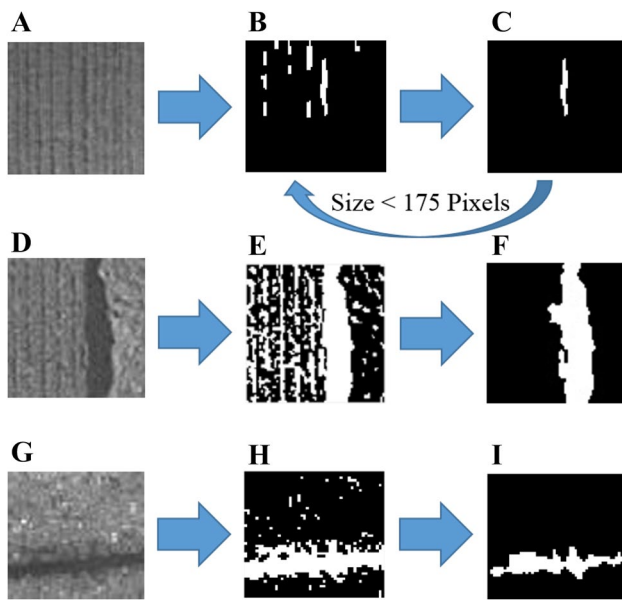
$$\text{Fitting Error} = \sum [y_i - f(x_i, a_1, a_2, \dots, a_n)]^2 \tag{3}$$

$f(x)$ : the polynomial value of the degree “ $n$ ” evaluated at  $x$ ,  $p_i$ : input coefficients,  $n$ : polynomial curve degree,  $y_i$ : the crack pixel location.

A third-degree (cubic) polynomial was implemented in this step. As no coordinate system was defined for the image, the vertical offsets were calculated for both  $x$  and  $y$  directions (Eq. (3)). The minimum value calculated for both directions divided by the total number of crack pixels results in the average polynomial fitting error (calibrated error). The calibrated error shows the crack distribution density within each tile. If the probable crack pixels are dispersed within the tile (Fig. 2b), it shows that crack pixels’ distribution has a low density. In this case, the polynomial fit would have high error values, and the tile would be detected as non-cracked.

On the other hand, if the probable crack pixels are amassed inside the tile with a size larger than 175 pixels (Fig. 2f and i), it shows that crack pixels’ distribution has a high density. In this case, the polynomial fit would have low error values, and the tile would be detected as cracked. The threshold value for the calibrated error was set to 0.05 using the concept of devised shape metric [61]. Thus, if the calibrated error is larger than 0.05, it shows that the identified crack pixels are dispersed within the tile, and they should not be considered as cracks; otherwise, if the calibrated error is lower than 0.05, the tile is deemed to be cracked.

The polynomial fit diagram for the cracks identified in Fig. 2b, f, and i are shown in Fig. 3a–c, respectively. Based



**Fig. 2** **a** PCC surface with longitudinal tining and without any cracks, **b** Output before applying the largest connected component extraction function on **(a)**, **c** Output after applying the largest connected component extraction function on **(b)**, **d** PCC surface with longitudinal tining and a crack, **e** Output before applying the largest connected component extraction function on **(d)**, **f** Output after applying the largest connected component extraction function on **(e)**, **g** AC surface with a crack, **h** Output before applying the largest connected component extraction function on **(g)**, **i** Output after applying the largest connected component extraction function on **(h)**

on Eq. (3), the calibrated errors (fitting error divided by the total number of crack pixels) were calculated; the results are shown in Fig. 3.

The algorithm moves from each tile to the other in a horizontal order until it reaches the end of the row (from left to right). After getting to the edge, the algorithm jumps to the next row. Fifteen pixels were considered for zero-padding around the image.

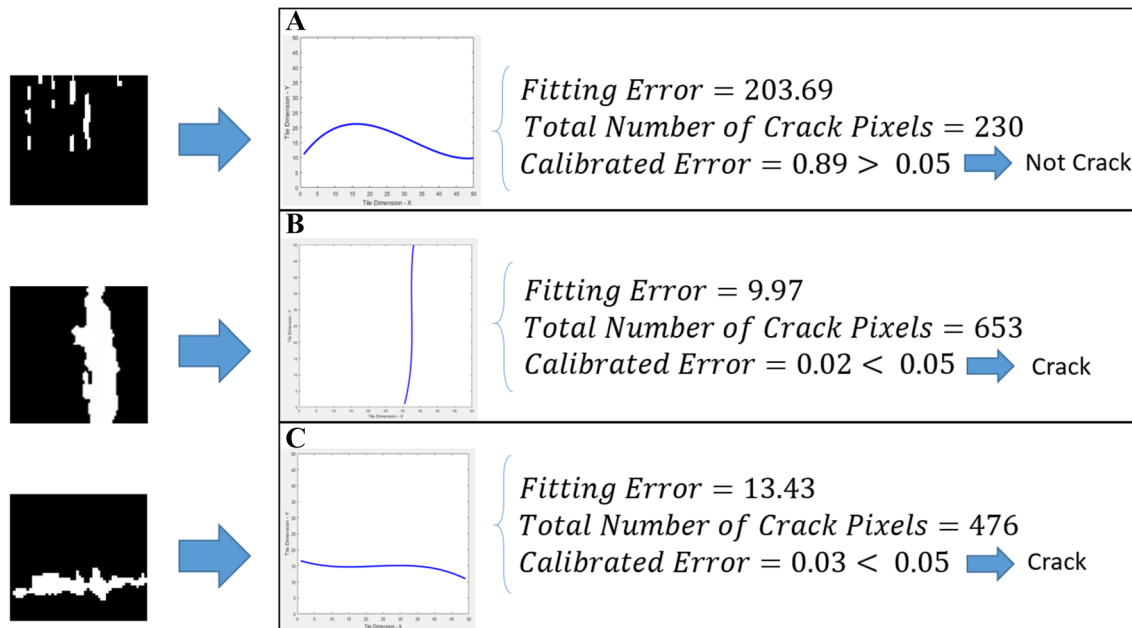
### 2.3 Filtering-Out Joints

If the section joints of a PCC surface are not removed, the algorithm would wrongly identify them as cracks. Fortunately, there is a common feature among these false positive detections: they are straight lines and can be filtered out using methods like Radon transform, Random Sample Consensus, and Hough transform. In this study, the Radon transform was used for the detection of straight lines. Radon transform implements a form of Hough transform, represented in the form of Eq. (4). The shortest distance from the line to the origin is calculated as [62, 65]:

$$r(\theta) = x_0 \cos(\theta) + y_0 \sin(\theta) \tag{4}$$

$r$ : shortest distance between the line and the origin point,  $\theta$ : the vector angle that joins the closest line point to the origin point,  $x_0, y_0$ : the  $x$  and  $y$  coordinates of points in the plane.

Radon transform shows the image as some blurred sine waves with various phases and amplitudes. For this study, the Radon transform was used to identify the number and



**Fig. 3** **a** Polynomial curve fit on probable crack pixels of Fig. 2b, **b** Polynomial curve fit on probable crack pixels of Fig. 2f, **c** Polynomial curve fit on probable crack pixels of Fig. 2i

slope of the straight lines (section joints and lane markings) by observing the number of nodes in the result. White nodes in the image represent the presence of straight lines (section joints and lane markings); the location and orientation of section joints can be determined by the position of the white nodes within the image. The algorithm for finding the straight line in this study consists of applying a 2-D Gaussian smoothing kernel combined with the Canny edge detection. The derivative of the Gaussian filter is used for calculating the gradient. After implementing the Radon transform, straight lines (section joints and lane markings) were extracted and filtered out. Figure 4a shows a sample PCC surface with two transverse joints.

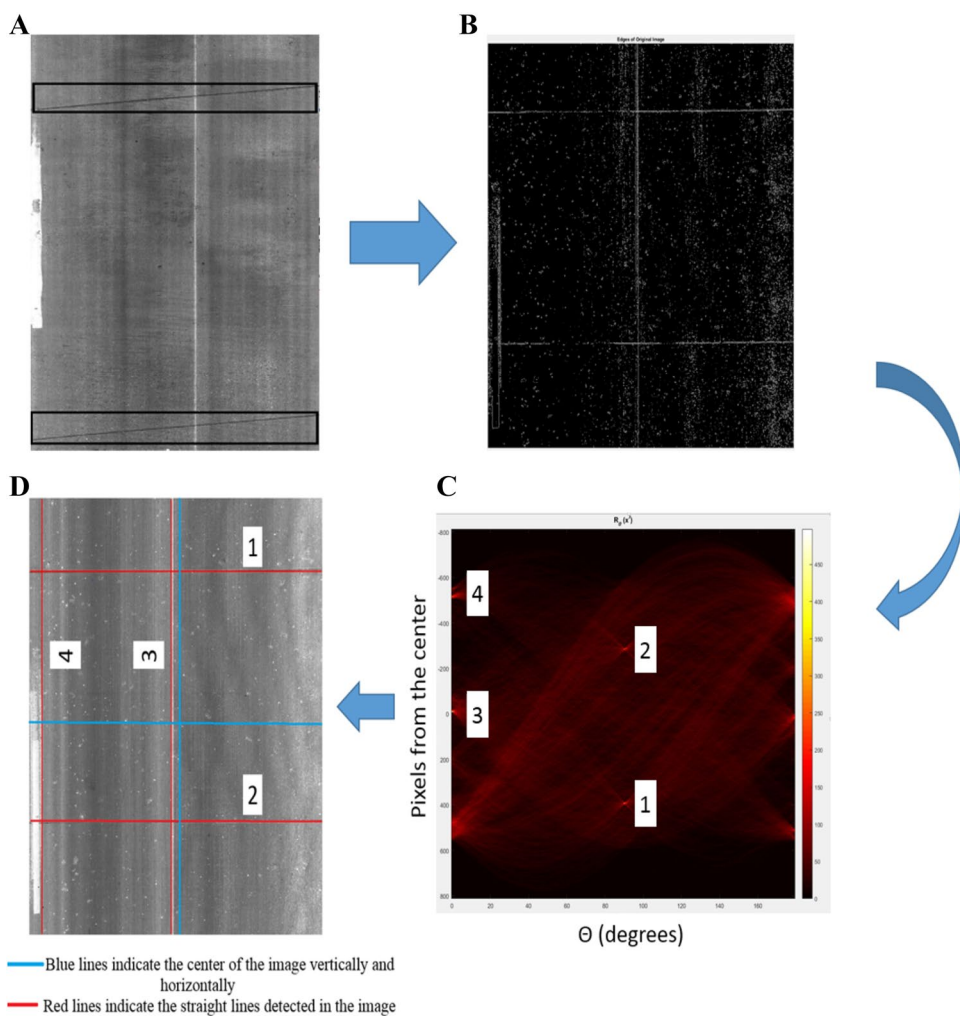
The blue horizontal line is the radial line. It passes through the center at  $\theta \approx 0$  degrees. The radial line intersects two solid red lines at  $-20$  and  $-510$  pixels from the center to the left. These solid red lines indicate the straight lines with the signal labeled 3 and 4 (Fig. 4c). Line 1 and 2 with  $\theta \approx 90$  degrees and a distance of 390 and  $-300$  pixels from the center correspond to signal 1 and 2 (Fig. 4c and d). As shown in Fig. 4, the transverse joints (lines 1 and 2) were

successfully filtered out. The other detected straight lines (lines 3 and 4) show the lane markings in the center and the edge of the roadway.

### 2.4 Tile Dimension Optimization

Due to the high importance of economic evaluation in pavement management and maintenance projects [66], in this section, the process for optimizing the algorithm and finding the best tile dimension is described. For this purpose, five different tile dimensions (“ $n \times n$ ”) were tested on 20 pavement image samples using the algorithm described in the previous sections. The time it took for the algorithm to detect the cracks was recorded. Also, the method’s accuracy with varying dimension sizes was calculated using the  $F_1$  score metric.  $F_1$  score is a robust and widely used metric for measuring the accuracy of a classification problem (crack versus non-crack). It is calculated by taking the harmonic mean between recall and precision [67, 68]. Recall shows the method’s ability to detect all relevant instances (cracks) in the dataset. It is defined as the ratio

**Fig. 4** a PCC surface with transverse joints, b Result of applying the Edge Detection on (a), c Result of applying Radon transform on (b), d detected straight lines



of the number of cracks classified correctly to the total number of cracks in the ground truth image dataset; it is calculated using Eq. (5):

$$\text{Recall} = \frac{T_P}{T_P + F_N} \quad (5)$$

Precision shows the proportion of correct detections to the total number of detected cracks; it is calculated using Eq. (6):

$$\text{Precision} = \frac{T_P}{T_P + F_P} \quad (6)$$

$F_1$  score is calculated using Eq. (7):

$$F_1 \text{ score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (7)$$

The validation results are summarized in Table 1.

As shown in Table 1, by reducing the tile dimension, the  $F_1$  score improves, and the time spent for detecting cracks increases.  $F_1$  score for  $25 \times 25$  tile was slightly higher than  $50 \times 50$  tile, but on the other hand, the time spent on detecting cracks using  $25 \times 25$  tiles was 4.9 times the time spent using  $50 \times 50$  tiles.  $F_1$  score is 13 percent lower using  $75 \times 75$  tiles than  $50 \times 50$  tiles, but it could only save 4 s compared to using  $50 \times 50$  tiles. Although the time spent using  $100 \times 100$  and  $125 \times 125$  tiles was significantly lower than the smaller tiles, their  $F_1$  score was considerably worse. The reason for the mentioned observations is that as the tile size increases, the number of crack pixels represented by a single label of cracked or non-cracked tiles increases; this leads to the reduction in accuracy ( $F_1$  score). The goal is to select a tile size that results in the highest  $F_1$  score in the shortest time, so  $50 \times 50$  tile is chosen as the best tile dimension and used in this paper.

## 2.5 Crack Drawing

After successfully identifying all the cracked tiles and filtering out any probable section joints, the tile centers were determined. Once again, the polynomial fitting curve was

fitted on the centers of the detected cracked tiles. The fitted curve was drawn as an overlay to the cracked area as long as the distance between the cracked tile centers was not over 50 pixels (the tile dimension). The adjacent tiles that the distance between their centers were more than 50 pixels were considered under separate crack curves. Curves that had less than three connected centers were filtered out because they were too small to be measured, and they would have caused some false alarms if they were close to the other curves. Finally, the crack lengths were measured by calculating the integration of the curves.

## 2.6 Crack Classification

Although this paper's primary purpose is to detect cracks, the algorithm can successfully classify two crack types into their respective groups. Longitudinal and transverse cracking are the most prevalent crack types in small pavement maintenance projects, which are the targets of this study. For longitudinal and transverse cracks, based on the orientation axes (slopes) of the crack curves, they were classified into longitudinal and transverse groups. First, the curves' gradients were calculated by measuring the curve orientation's degree from the horizontal line and computing the differential. As all the images were in the same direction, a fixed coordinate system was set for measuring the curve orientation. Second, the maximum value of the slope along each direction was selected as the curve's slope; if the slope of the curve was greater than 0.75, it was labeled as "Longitudinal Cracking," and if it was smaller than 0.75, it was classified as "Transverse Cracking."

## 3 Results

The dataset for this study was provided by the Midwest Transportation Center (MTC). About 60 percent (78 images) of the images used in this study were captured using a 35 mm camera mounted on a boom on top of a vehicle (at the height of around 10 feet from the ground). The data collection was performed as part of the LTPP's data collection from highways in Illinois and Iowa states in 2003. LTPP program monitors more than 2,500 pavement test sections around the United States and Canada. The captured distances are equivalent to hundreds of miles of AC and PCC pavement surfaces in these areas. The camera lens was placed parallel to the pavement surface, and the vehicle was moving with the highway speed. Each image covered 60 ft. ( $\approx 18.29$  m.) and 20 ft. ( $\approx 6.1$  m.) of roadway length and width, respectively. The rest of the images were captured using a 3-D laser-based high-resolution video log imaging system in 2017. The vehicle captured a 100 mm sample of data every meter at 70 mph ( $\approx 112.66$  km/h) with a 6-megapixel camera

**Table 1** Validation metrics based on the tile dimension selection

Tile dimensions ( $n \times n$ )	$F_1$ score	Time spent on crack detection (seconds)
$25 \times 25$	0.89	132
$50 \times 50$	0.88	27
$75 \times 75$	0.77	23
$100 \times 100$	0.51	11
$125 \times 125$	0.38	6



located around 6 ft. ( $\approx 1.83$  m.) from the ground level. Each image covered 52.8 ft. ( $\approx 16.09$  m.) and 20 ft. ( $\approx 6.1$  m.) of roadway length and width, respectively. The collective length of the highway test sections used in this study was about 1.5 miles (equivalent to  $14,716$  m<sup>2</sup>). This study's test sections were selected from a wide range of pavement conditions (poor to good) containing cracks with various severity levels (moderately low level to high level).

The reason for using two different image datasets in this study is to show the method's suitability for detecting cracks from images captured using various imaging systems. The second dataset consists of a wider variety of crack types and fully represents the method's ability to detect a wide range of pavement cracks. The developed method was tested on these two datasets that consist of 130 images in total, 85 images of AC surfaces and 45 images of PCC surfaces with several pavement crack types and defects including longitudinal, transverse, alligator cracking, potholes, surface failure, edge, and lane joint cracking and PCC joint spalling. The proposed method could successfully detect most of the cracks. Twenty-five of the tested images did not have any types of cracks. The results of applying the proposed method on four sample pavement surfaces with longitudinal and transverse cracks are shown in Fig. 5.

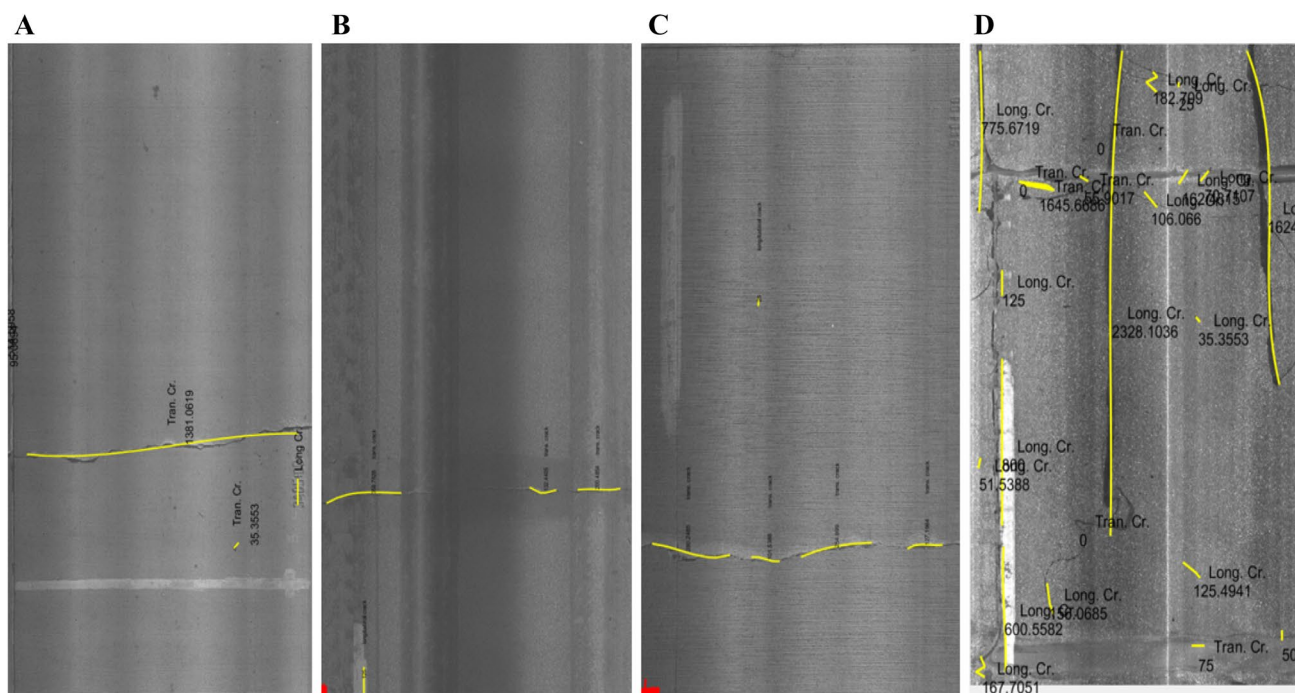
Figure 5a–c shows that most of the longitudinal and transverse cracks were correctly detected and drawn;

false-negative detections were rare, and only a few numbers of false-positive detections were observed. False-positive detections were mostly due to pavement carving or other superficial objects that were misunderstood as cracks. External (superficial) objects such as oil spills, manholes (Fig. 6b), jagged lane markings (Fig. 5d), or even pavement discoloration can make parts of the affected pavement surface have similar pixel intensity values to cracks. Fortunately, these instances are not regularly present on pavement surfaces, and most of them are filtered out during different stages of the proposed algorithm. Due to the small width of low-level and hairy cracks, this method was unsuccessful in detecting them.

Figure 6 shows the crack detection results in surfaces with more complex crack types and patterns.

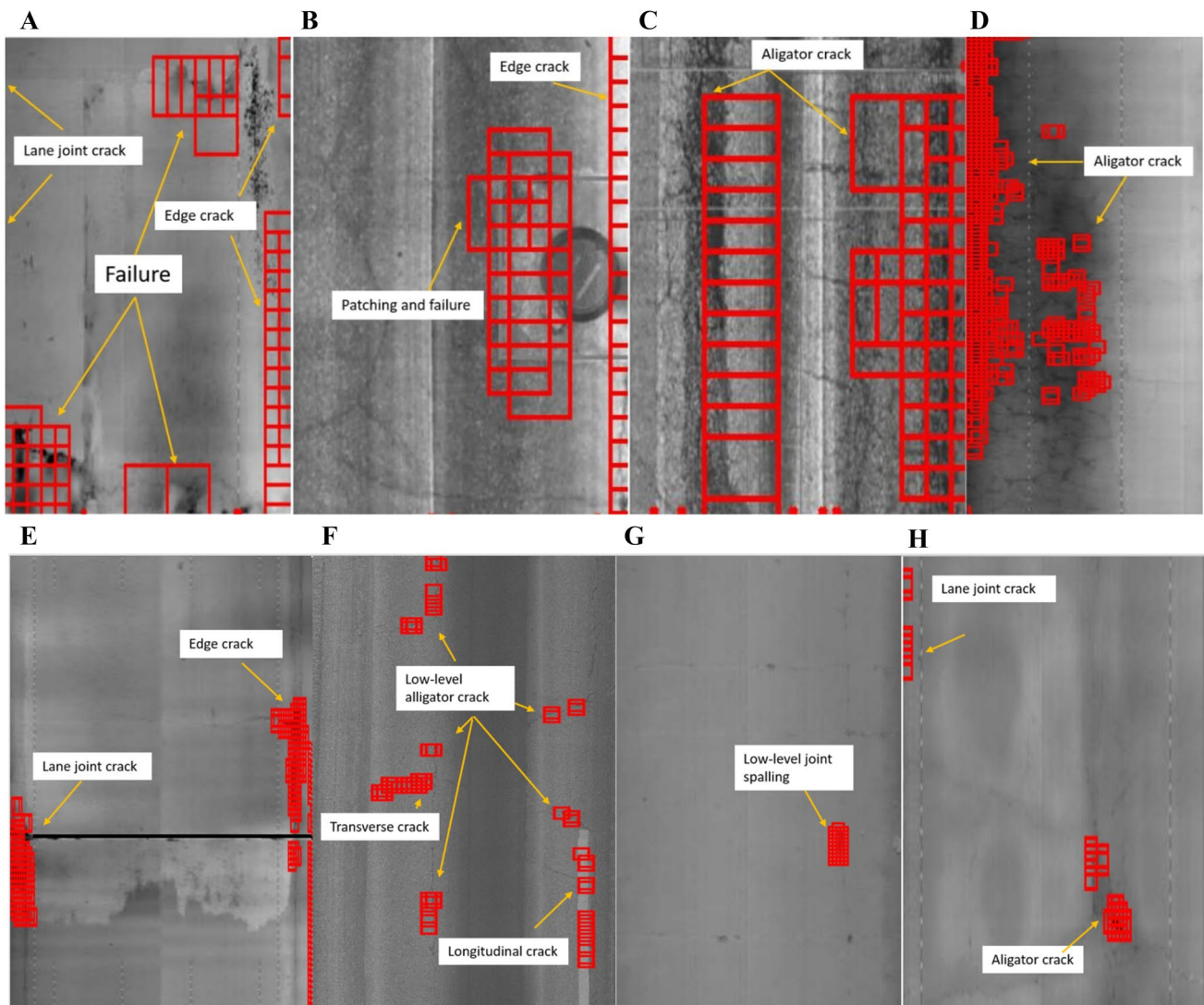
Figure 6a–h shows that most of the pavement crack types (longitudinal cracking, transverse cracking, alligator cracking, edge, and lane joint cracking) and defects (potholes, patching, failure, joint spalling) were successfully detected using the proposed algorithm.

For validating the proposed method, the detection and length measurement accuracies were calculated based on the precision and recall scores. The number of true/false positives/negative detections was determined by comparing the method results with the human experts' manual labeling results. Validation results for the detection and length



**Fig. 5** **a** 2-D image of a PCC surface with transverse tining and severe-level transverse cracking, **b** 2-D image of an AC surface with medium-level transverse and longitudinal cracking, **c** 2-D image of

a PCC surface with transverse tining and medium-level transverse cracking, **d** 2-D image of an AC surface with longitudinal and transverse sealings



**Fig. 6** **a** 3-D image of an AC surface with failure, edge, and lane joint crack, **b** 2-D image of an AC surface with patching, failure and edge crack, **c** 2-D image of an AC surface with alligator cracking, **d** 3-D image of a PCC surface with alligator cracking, **e** 3-D image of a

PCC surface with edge and lane joint cracking, **f** 2-D image of an AC surface with low-level alligator cracking, **g** 3D image of a PCC surface with low-level joint spalling, **h** 3-D image of an AC surface with alligator and lane joint cracking

measurement of medium to severe-level cracks are shown in Table 2.

The validation power of the proposed method is compared with some state-of-the-art crack detection algorithms in Table 3.

Table 3 shows the superiority of the proposed algorithm compared to the others (CrackIT [69], CrackForest [32], CrackTree [70]). It should be noted that the validation results of some of these studies are based on the boundary and pixel-level ground truth analyses, which are more accurate than the method used in this study for building the ground truth. However, the network-level pavement crack detection surveys mainly care about the existence of cracks, lengths,

and their types; more sensitive metrics may not be required for this purpose.

## 4 Conclusion

In this study, a tile-based image processing algorithm was developed to detect pavement cracks and classify the longitudinal and transverse cracks into their respective groups. The main stages of the method are categorized into the following steps: 1—localized thresholding (pre-filtering) 2—morphological operations, 3—pre-processing for noise reduction using the adaptive median filtering, 4—detecting cracks based on the spatial distribution of crack pixels,

**Table 2** Validation results

$N$	158
$t_p$	131
$f_p$	16
$f_n$	27
Precision	0.89
Recall	0.83
$F_1$ score	0.86
Crack length measurement accuracy	80%

$N$  total number of cracks,  $t_p$  number of true positives (number of detected cracks that are real cracks),  $f_p$  number of false positives (number of detected cracks that are not real cracks),  $f_n$  number of false negatives (number of undetected cracks that are real cracks).

**Table 3** Comparison of the validation results of the proposed method with some other methods

Method	Precision	Recall	$F_1$ score
Canny	0.17	0.26	0.21
CrackIT	0.69	0.78	0.73
CrackTree	0.71	0.77	0.74
FFA	0.74	0.65	0.69
CrackForest (KNN)	0.84	0.81	0.82
Proposed method	0.89	0.83	0.86

5—Radon transform for detecting and removing PCC pavement joints, 6—curve-fitting on cracked tiles and identifying the orientation axes of the fitted curves for classifying cracks and measuring their lengths. For validation, the precision, recall, and  $F_1$  score were used to compare the study results with the human experts' manual labeling results. Finally, the validation results of the proposed method were compared with some other state-of-the-art algorithms. The results showed that the detection method is highly accurate for detecting the existence of medium to high-level cracks and measuring their lengths (Precision = 0.89, recall = 0.83,  $F_1$  score = 0.86 and crack length measurement accuracy = 80%). The method successfully detected most types of pavement cracks (longitudinal cracking, transverse cracking, alligator cracking, edge, and lane joint cracking) and other pavement surface defects (potholes, patching, failure, joint spalling). This method is built upon thresholding algorithms and is similar to model-based techniques; it implements a tile-based analysis and proposes shape metrics to detect and classify pavement cracks. Unlike most of the previous methods, this method is not limited to the texture type.

In contrast to the methods that are solely based on thresholding, this study's proposed approach is less susceptible to the choice of threshold values because thresholding is used merely as a preliminary step. Furthermore, this study's pixel density-based algorithm can successfully distinguish between the crack properties and non-crack objects such as shadows. Besides, this method is incredibly cost-effective as no advanced computational system is required for the detection and classification process. The crack detection method is fast; it took less than 20 s for each image to be processed using an Intel® Xeon® CPU 3.7 GHz processor and an academic version of Matlab 2018a. To further improve this method's economic aspect, a high-performance computing system could be used to accelerate the detection process. However, the main reason for using a commodity processor in this study is to minimize the cost and required resources and make it an ideal method for smaller, county-level crack detection projects that need a fast, on-site, and low-cost method with an acceptable detection power. The classification and crack length measurement took less than 10 s per image. This method could help engineers and decision-makers to determine the best fiscally constrained action to be taken on pavement maintenance programs and long-term capital investment plans. It would allow local agencies to perform low-cost crack detection studies on their limited number of roads. It is not economically efficient for local highway agencies to use giant corporations' high-cost crack detection systems. One of the study's significant limitations is that the method cannot detect low-level cracks; thus, it cannot measure their lengths. The developed method also showed some inconsistencies when applied on the 3-D laser-scanned images due to the particular contrast levels of those images. The method showed some problems of falsely identifying parts of the external objects on the pavement surface as cracks. For future study, it is recommended to do additional research to improve the method's strength in dealing with external objects on the highway surface (such as number curving, oil spillage, etc.). It is suggested to combine the proposed method with a subsidiary deep neural network-based model such as various architectures of convolutional neural networks (CNNs) to automatically learn some salient image features. Artificial intelligence-based (AI) techniques would improve the algorithm's accuracy and efficiency. The potential improvements include detecting low-level cracks in complex patterns, improving noise removal, and measuring crack width.

**Funding** The research was conducted at the Institute for Transportation of Iowa State University and did not receive external funding directly.

**Availability of data and material** The data for this study is available and can be accessed through: [https://drive.google.com/file/d/1U8le-HJYZU5ISN0IXEiNf\\_S64g4kv8/view?usp=sharing](https://drive.google.com/file/d/1U8le-HJYZU5ISN0IXEiNf_S64g4kv8/view?usp=sharing).

**Code availability** The custom code for this study is written in Matlab and will be provided upon request.

## Declarations

**Conflict of interest** There is no conflict of interest associated with this study.

## References

- Lekshmipathy, J., Samuel, N. M., & Velayudhan, S. (2020). Vibration vs vision: best approach for automated pavement distress detection. *International Journal of Pavement Research and Technology*. <https://doi.org/10.1007/s42947-020-0302-y>.
- Arezoumand, S., Mahmoudzadeh, A., Golroo, A., & Mojaradi, B. (2021). Automatic pavement rutting measurement by fusing a high speed-shot camera and a linear laser. *Construction and Building Materials*, 283, 122668.
- Kazemian, M., Sedighi, S., Ramezaniapour, A. A., Bahmanzadeh, F., & Ramezaniapour, A. M. (2021). Effects of cyclic carbonation and chloride ingress on durability properties of mortars containing Trass and Pumice natural pozzolans. *Structural Concrete*.
- Barri, K., Jahangiri, B., Davami, O., Buttler, W. G., & Alavi, A. H. (2020). Smartphone-based molecular sensing for advanced characterization of asphalt concrete materials. *Measurement*, 151, 107212.
- Adlinge, S. S., & Gupta, A. K. (2013). Pavement deterioration and its causes. *International Journal of Innovative Research and Development*, 2(4), 437–450.
- Aboudi, J. (1987). Stiffness reduction of cracked solids. *Engineering Fracture Mechanics*, 26(5), 637–650.
- Hamed, G. H., Sahraei, A., & Esmaeeli, M. R. (2018). Investigate the effect of using polymeric anti-stripping additives on moisture damage of hot mix asphalt. *European Journal of Environmental and Civil Engineering*, 25, 1–14.
- Daghighi, A. (2020). Full-Scale Field Implementation of Internally Cured Concrete Pavement Data Analysis for Iowa Pavement Systems. *Creative Components*. 638. <https://lib.dr.iastate.edu/creativecomponents/638>.
- Malakooti, A., Theh, W. S., Sadati, S. S., Ceylan, H., Kim, S., Mina, M., & Taylor, P. C. (2020). Design and full-scale implementation of the largest operational electrically conductive concrete heated pavement system. *Construction and Building Materials*, 255, 119229.
- Farhadmanesh, M., Cross, C., Mashhadi, A. H., Rashidi, A., & Wempen, J. (2021). Highway Asset and Pavement Condition Management using Mobile Photogrammetry. *Transportation Research Record*, 03611981211001855.
- Majidifard, H., Adu-Gyamfi, Y., & Buttler, W. G. (2020). Deep machine learning approach to develop a new asphalt pavement condition index. *Construction and building materials*, 247, 118513.
- Justo-Silva, R., & Ferreira, A. (2019). Pavement maintenance considering traffic accident costs. *International Journal of Pavement Research and Technology*, 12, 562–573. <https://doi.org/10.1007/s42947-019-0067-3>.
- Cross, C., Farhadmanesh, M., & Rashidi, A. (2020). Assessing Close-Range Photogrammetry as an Alternative for LiDAR Technology at UDOT Divisions (No. UT-20.18). Utah. Dept. of Transportation. Division of Research.
- Khajehvand, M., Rassafi, A. A., & Mirbaha, B. (2021). Modeling traffic noise level near at-grade junctions: Roundabouts, T and cross intersections. *Transportation Research Part D: Transport and Environment*, 93, 102752.
- Dhital, D., & Lee, J. R. (2012). A fully non-contact ultrasonic propagation imaging system for closed surface crack evaluation. *Experimental Mechanics*, 52(8), 1111–1122.
- Hosseini, S., & Smadi, O. (2020). How prediction accuracy can affect the decision-making process in pavement management system. *Infrastructures*. <https://doi.org/10.31224/osf.io/t28ue>.
- Mahmoudzadeh, A., Golroo, A., Jahanshahi, M. R., & Firoozi Yeganeh, S. (2019). Estimating pavement roughness by fusing color and depth data obtained from an inexpensive RGB-D sensor. *Sensors*, 19(7), 1655.
- Abukhalil, Y. B. (2019). Cross asset resource allocation framework for pavement and bridges in Iowa. *Graduate Theses and Dissertations*. 16951. <https://lib.dr.iastate.edu/etd/16951>.
- Zahedian, S., Sadabadi, K. F., & Nohekhan, A. (2021). Localization of autonomous vehicles: Proof of concept for a computer vision approach. arXiv preprint [arXiv:2104.02785](https://arxiv.org/abs/2104.02785).
- Aboah, A., Shoman, M., Mandal, V., Davami, S., Adu-Gyamfi, Y., & Sharma, A. (2021). A vision-based system for traffic anomaly detection using deep learning and decision trees. arXiv preprint [arXiv:2104.06856](https://arxiv.org/abs/2104.06856).
- Zahedian, S., Sekula, P., Nohekhan, A., & Vander Laan, Z. (2020). Estimating hourly traffic volumes using artificial neural network with additional inputs from automatic traffic recorders. *Transportation Research Record*, 2674(3), 272–282.
- Nejad, F. M., Motekhasas, F. Z., Zakeri, H., & Mehrabi, A. (2015). An image processing approach to asphalt concrete feature extraction. *Journal of Industrial and Intelligent Information*, 3(1).
- Tsai, Y.-C.J., & Chatterjee, A. (2018). Pothole detection and classification using 3D technology and watershed method. *Journal of Computing in Civil Engineering*, 32(2), 04017078.
- Safaei, N., & Smadi, O., (2019). A tile-based image processing method based on crack density for detection and classification of pavement cracks. In: *98th Annual Meeting of Transportation Research Board*. TRID. <https://trid.trb.org/view/1595322>.
- Hosseini, S. A., Ahmad, A., & Smadi, O. (2020). Use of deep learning to study modeling deterioration of pavements a case study in Iowa. *Infrastructures*, 5(11), 95.
- Federal Highway Administration Research and Technology, (2003). Distress Identification Manual for The LTPP. FHWA-RD-03-031, (May).
- Rgah-Ostadi, N., Nazef, A., Daleiden, J., & Zhou, Y. (2017). Evaluation framework for automated pavement distress identification and quantification applications. *Transportation Research Record: Journal of the Transportation Research Board*, 2639, 46–54.
- Albitres, C. M. C., Smith, R. E., & Pendleton, O. J. (2007). Comparison of automated pavement distress data collection procedures for local agencies in San Francisco Bay Area, California. *Transportation Research Record: Journal of the Transportation Research Board*, 1990(1), 119–126.
- Chambon, S., & Moliard, J.-M. (2011). Automatic road pavement assessment with image processing: review and comparison. *International Journal of Geophysics*, 2011, 1–20.
- Safaei, N. (2019). Pixel and region-based image processing algorithms for detection and classification of pavement cracks. *Graduate Theses and Dissertations*. 17555. <https://lib.dr.iastate.edu/etd/17555>.
- Safaei, N., Smadi, O., Safaei, B., & Masoud, A. (2021). Efficient road crack detection based on an adaptive pixel-level segmentation algorithm. *Transportation Research Record*, 03611981211002203.
- Shi, Y., Cui, L., Qi, Z., Meng, F., & Chen, Z. (2016). Automatic road crack detection using random structured forests. *IEEE Transactions on Intelligent Transportation Systems*, 17(12), 3434–3445.

33. Oliveira, H. J. M. (2013). Crack detection and characterization in flexible road pavements using digital image processing. *Ph.D. Dissertation*, Universidade de Lisboa-Instituto Superior Técnico.
34. Roli, F. (1996). Measure of texture anisotropy for crack detection on textured surfaces. *Electronics Letters*, 32(14), 1274–1275.
35. Nguyen, T. S., Avila, M., & Begot, S. (2009). Automatic detection and classification of defect on road pavement using anisotropy measure. *17th European Signal Processing Conference*, 2009, pp. 617–621.
36. Koch, C., Doycheva, K., Kasireddy, V., Akinci, B., & Fieguth, P. (2016). A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Advanced Engineering Informatics*, 30(2), 208–210.
37. Tanaka, N., & Uematsu, K. (1998). A crack detection method in road surface images using morphology. *MVA*, 98, 17–19.
38. Dung, C. V. (2019). Autonomous concrete crack detection using deep fully convolutional neural network. *Automation in Construction*, 99, 52–58.
39. Cord, A., & Chambon, S. (2011). Automatic road defect detection by textural pattern recognition based on AdaBoost. *Computer-Aided Civil and Infrastructure Engineering*, 27(4), 244–259.
40. Karballaezadeh, N., Zaremotekhas, F., Shamshirband, S., Mosavi, A., Nabipour, N., Csiba, P., & Várkonyi-Kóczy, A. R. (2020). Intelligent road inspection with advanced machine learning; hybrid prediction models for smart mobility and transportation maintenance systems. *Energies*, 13(7), 1718.
41. Liu, Z., Azmin, S., Ohashi, T., & Ejima, T. (2002). A tunnel crack detection and classification systems based on image processing. *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, 4664, 145–152.
42. Kass, M., Witkin, A., & Terzopoulos, D. (1988). Snakes: Active contour models. *International Journal of Computer Vision*, 1(4), 321–331.
43. Kaul, V., Yezzi, A., & Tsai, Y. (2012). Detecting curves with unknown endpoints and arbitrary topology using minimal paths. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(10), 1952–1965.
44. Amhaz, R., Chambon, S., Idier, J., & Baltazart, V. (2014). A new minimal path selection algorithm for automatic crack detection on pavement images. *IEEE International Conference on Image Processing (ICIP)*, 2014, 788–792.
45. Amhaz, R., Chambon, S., Idier, J., & Baltazart, V. (2015). Automatic crack detection on 2D pavement images: An algorithm based on minimal path selection. *IEEE Transactions on Intelligent Transportation Systems*.
46. Paar, H. K., & Sidla, O. (2006). Optical crack following on tunnel surfaces. *Proceedings of the SPIE International Society for Optical Engineering* 6382.
47. Yamaguchi, T., & Hashimoto, S. (2009). Fast crack detection method for large-size concrete surface images using percolation-based image processing. *Machine Vision and Applications*, 21(5), 797–809.
48. Chaiyasarn, K. (2011). Damage detection and monitoring for tunnel inspection based on computer vision. *Ph.D. Dissertation*, Christ's College, University of Cambridge, UK.
49. Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(12), 2481–2495.
50. Ghosh, R., & Smadi, O. (2021). Automated Detection and Classification of Pavement Distresses Using 3D Pavement Surface Images and Deep Learning. *Transportation Research Record*, 036119812111007481.
51. Hoang, N.-D. (2019). Automatic detection of asphalt pavement raveling using image texture based feature extraction and stochastic gradient descent logistic regression. *Automation in Construction*, 105, 102843.
52. Hadjidemetriou, G. M., & Christodoulou, S. E. (2019). Vision- and entropy-based detection of distressed areas for integrated pavement condition assessment. *Journal of Computing in Civil Engineering*, 33(3), 04019020.
53. Yousaf, M. H., et al. (2018). Visual analysis of asphalt pavement for detection and localization of potholes. *Advanced Engineering Informatics*, 38, 527–537.
54. Cheng, H. D., Chen, J.-R., Glazier, C., & Hu, Y. G. (1999). Novel approach to pavement cracking detection based on fuzzy set theory. *Journal of Computing in Civil Engineering*, 13(4), 270–280.
55. Ying, L., & Salari, E. (2010). Beamlet transform-based technique for pavement crack detection and classification. *Computer-Aided Civil and Infrastructure Engineering*, 25(8), 572–580.
56. Tsai, Y.-C., Kaul, V., & Mersereau, R. M. (2010). Critical assessment of pavement distress segmentation methods. *Journal of Transportation Engineering*, 136(1), 11–19.
57. Lee, B. J., & David Lee, H. (2004). Position-invariant neural network for digital pavement crack analysis. *Computer-Aided Civil and Infrastructure Engineering*, 19(2), 105–118.
58. Xu, B., & Huang, Y. (2005). Automatic inspection of pavement cracking distress. *Applications of Digital Image Processing XXVIII*.
59. Oliveira, H., & Lobato, P. (2009). Supervised crack detection and classification in images of road pavement flexible surfaces. *Recent Advances in Signal Processing*.
60. Oliveira, H., & Correia, P. L. (2013). Automatic road crack detection and characterization. *IEEE Transactions on Intelligent Transportation Systems*, 14(1), 155–168.
61. Wang, T., Gopalakrishnan, K., Somani, A., Smadi, O., & Ceylan, H. (2016). Machine-Vision-Based Roadway Health Monitoring and Assessment: Development of a Shape-Based Pavement-Crack-Detection Approach. Final Report. Institute for Transportation, Iowa State University.
62. Katakam, N. (2009). Pavement Crack Detection System Through Localized Thresholding. *Master's Thesis*. The University of Toledo, The University of Toledo Digital Respiratory, pp. 36–36.
63. Peng, L. (2004). Adaptive Median Filtering. Seminar Report, Machine Vision 140.429 Digital Image Processing.
64. Image Filtering. Patrice's Lectures. Department of Computer Science, University of Auckland, New Zealand. [www.cs.auckland.ac.nz/courses/compsci373s1c/PatricesLectures/Image%20Filtering\\_2up.pdf](http://www.cs.auckland.ac.nz/courses/compsci373s1c/PatricesLectures/Image%20Filtering_2up.pdf). Accessed 10 Mar 2018.
65. Detect Lines Using the Random Transform. Documentation in MathWorks. [www.mathworks.com/help/images/detect-lines-using-the-radon-transform.html](http://www.mathworks.com/help/images/detect-lines-using-the-radon-transform.html). Accessed 29 July 2018.
66. Babashamsi, P., Yusoff, N. I. M., Ceylan, H., Nor, N. G. M., & Jenatabadi, H. S. (2016). Evaluation of pavement life cycle cost analysis: Review and analysis. *International Journal of Pavement Research and Technology*, 9(4), 241–254.
67. Sasaki, Y. (2007) 'The Truth of the F-measure', Teach Tutor mater, 1–5.
68. Derczynski, L. (2016). Complementarity, F-score, and NLP Evaluation. In: *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*. 2016.
69. Oliveira, H., & Correia, P. L. (2014). CrackIT—An image processing toolbox for crack detection and characterization. In: *2014 IEEE International Conference on Image Processing (ICIP)*.
70. Zou, Q., et al. (2012). CrackTree: automatic crack detection from pavement images. *Pattern Recognition Letters*, 33(3), 227–238.



**Nima Safaei** is a PhD student in Information Systems at the Tippie College of Business of the University of Iowa. He received his master's degree in Transportation Engineering from Iowa State University in 2019. Nima's research interests include data science, machine learning, computer vision, and asset management. Nima holds a bachelor's degree in Civil Engineering from Amirkabir University of Technology.

projects for the Iowa Department of Transportation, the Iowa Highway Research Board, the federally funded Midwest Transportation Center, National Cooperative Highway Research Program, Second Highway Research Program, and Federal Highway Administration dealing with roadway data, safety, and asset management.

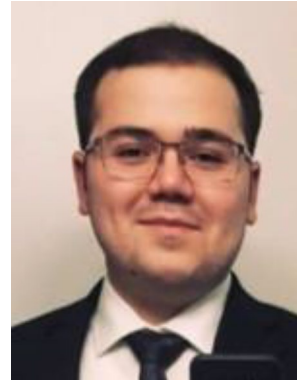


**Arezoo Masoud** received her B.Sc. degree in Civil Engineering from Amirkabir University of Technology in 2016. She is currently a master's student in Business Analytics at the Tippie College of Business of the University of Iowa. She also works as a data research assistant at the National Advanced Driving Simulator (NADS). Arezoo's research interests include data science, business analytics, quantitative finance, and computer vision.



**Dr. Omar Smadi** received his PhD degree in Transportation Engineering from Iowa State University (ISU) in 2000. He is currently an associate professor at the Department of Civil, Construction and Environmental Engineering of Iowa State University. Omar is the director of the Center for Transportation Research and Education (CTRE) at Institute for Transportation (InTrans) since 2016. Omar has over 25 years of experience in the area of infrastructure and asset management ranging from

pavements, bridges, safety, pavement marking, signs, and other infrastructure assets. He is currently serving as PI for several research



**Babak Safaei** is a Master's student in Civil Engineering with a focus on Transportation at Michigan State University (MSU). He is serving as a transportation research assistant at the Mobility Center of Michigan State University. His contribution involves transportation network analytics, and data analysis for projects in areas of traffic operations, safety, and planning. Babak also holds a BSc degree in Civil Engineering from Sharif University of Technology.