IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 26, 2018, accepted April 13, 2018, date of publication April 23, 2018, date of current version May 24, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2829347

Automatic Pixel-Level Pavement Crack Detection Using Information of Multi-Scale Neighborhoods

DIHAO AI[1,2], GUIYUAN JIANG[2], LAM SIEW KEI[2], AND CHENGWU LI[1]
[1]School of Resource and Safety Engineering, China University of Mining and Technology Beijing, Beijing 100083, China
[2]School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798

Corresponding author: Dihao Ai (aixi1020@163.com) and Guiyuan Jiang (jguiyuan@gmail.com)

This work was supported by the National Natural Science Foundation of China under Grant 51274206 and Grant 51404277.

ABSTRACT Robust automatic pavement crack detection is critical to automated road condition evaluation. However, research on crack detection is still limited and pixel-level automatic crack detection remains a challenging problem, due to heterogeneous pixel intensity, complex crack topology, poor illumination condition, and noisy texture background. In this paper, we propose a novel approach for automatically detecting pavement cracks at pixel level, leveraging on multi-scale neighborhood information, and pixel intensity. Using pixel intensity information, a probabilistic generative model (PGM) based method is developed to calculate the probability of a crack for each pixel. This produces a probability map consisting of the probability of each pixel being part of the crack. We demonstrate that the neighborhoods of each pixel contain critical information for crack detection, and propose a support vector machine (SVM) based method to calculate the probability maps using information of multi-scale neighborhoods. We develop a fusion algorithm to merge the multiple probability maps, obtained from both PGM and SVM approaches, into a fused map, which can detect cracks with accuracy higher than any of the original probability maps. We also propose a weighted dilation operation that relies on the fused probability map to enhance the recognition of borderline pixels and improve the crack continuity without increasing the crack width improperly. Experimental results demonstrate that our algorithm achieves better performance in terms of precision, recall, f1-score, and receiver operating characteristic, in comparison with the state-of-the-art pavement crack detection algorithms.

INDEX TERMS Pavement crack detection, probability map, multi-scale neighborhoods, probabilistic generative mode, support vector machine.

I. INTRODUCTION

Surveys and analysis of pavement distress is important for the maintenance and evaluation of pavement safety, which requires accurate information about the state of the pavement surface [1], [2]. The most direct manifestation and important indicator to evaluate the condition of the pavement distress is the production of cracks. Thus, accurately and timely detection and quantization of cracks, which usually have complex topology in physical form, can provide important information for quantifying the quality of pavement surfaces as well as preventing and controlling the pavement distress. As such, crack detection is an essential part of road maintenance systems and is attracting more and more attentions in recent years. Moreover, automatic crack detection will also benefit a wide range of applications in other industrial fields, including civil infrastructures like tunnels [3], bridges [4], and dams [5], as well as metallic surface [6] and rock surface [7].

Traditionally, a common method used in evaluating road surface distress is via human visual inspection, which has proven effective for new and variable pavement crack detection tasks. However, different people often produce different judgment results for the same pavement condition due to inherent human subjectivity. In addition, human visual inspection is often time-consuming, labor-intensive and cannot consistently produce results that are objective in quantitative aspect. Therefore, a robust automatic pavement crack detection method is crucial for evaluating the pavement condition automatically and conducting the necessary road maintenance activities.

To accelerate the detection speed as well as achieve reliable and consistent detections, many algorithms for automated pavement crack detection have been developed over the past several decades with the improvement of image processing capabilities. In the earlier works, the intensity-threshold

algorithms and filter-based methods are the most commonly adopted approaches.

The *intensity-threshold algorithm* [8]–[11] is based on the assumption that a crack has a lower intensity value than that of the background image. By setting different threshold values, this algorithm can quickly judge the results of input images. This assumption can be accepted in many general cases since cracks always absorb more light than other areas and often appear as dark curves or tapes in the image. However, when there exist a certain amount of noisy pixels whose intensities are lower than crack pixels, there will be a serious decline in detection performance.

Since the cracks and edges have many similarities in morphology, many researches [12], [13] apply *filter based methods* that have been developed for edge detection to detect cracks on pavement images. In recent years, some researches also employ *graph theory technique* to detect cracks, e.g. Minimal Path Selection (MPS) [14], [15] and Minimum Spanning Tree (MST) [16], [17]. They extract simple contour-like curves when given different endpoints in an image. These methods are robust to the background noises but cannot perform well when dealing with complex topology of cracks.

However, all the above mentioned approaches cannot achieve satisfactory results, as fully automatic pavement crack detection is inherently a challenging task. This is because 1) the pavement images always subject to a certain degree of noise in the process of acquisition, quantification as well as transmission; 2) the cracks contained in pavement distress images are of heterogeneous intensity, have complex topology in morphology, and are often under bad illumination condition; 3) due to the various illuminations, crack shape and noise, one method that works well for a particular situation of pavement cracks inspection often yields poor performance for other situations.

With the development of the computing capabilities and image acquisition equipment, many *machine learning algorithms* such as artificial neural network (ANN) [18], Random Forests (RF) [19], and Bayesian Data Fusion [6] have been introduced into pavement crack detection and achieved acceptable results. However, most of these algorithms detect cracks in block-level, i.e. they first partition images into pixel blocks and detect a block as crack block if it contains crack pixels. These approaches neglect the actual borders and the widths of the cracks to be detected, thus do not accurately measure cracks for subsequent applications.

In this paper, we focus on pavement crack detection at pixel-level, which produce the details about the borders and the widths of the cracks. It supports fine-grained characterization of the cracks, thus can provide more critical information for pavement distress analysis. In particular, with the accurate detection of pavement cracks, it is easy to achieve accurate result on various types of pavement cracks, e.g., longitudinal, transverse, diagonal, block, alligator [20], [21] and even some irregular shapes. Also, crack geometric features like width, length, and orientation are also important indicators to identify the road distress level.

In this work, we aim to design efficient methods to achieve pixel-level pavement crack detection. We propose novel probability based methods that jointly consider pixel intensity and multi-scale neighborhood information. That is, for each pixel, we estimate the probability of each pixel being a crack using different point of views, i.e. based on the pixel intensity, and based on information of neighborhoods with sizes of 7×7, 13×13, 19×19 and 25×25, respectively. The probability based on pixel intensity is calculated using Probabilistic Generative Model (PGM) based approach, while probabilities based on neighborhood information are calculated using Support Vector Machine (SVM) based method. The multiple probabilities of each pixel are further fused together to improve detection accuracy. In addition, the probability information is also used to enhance the recognition of borderline pixels through a weighted dilation operation.

The contributions of this paper can be summarized as follows: 1) we first develop a PGM based method to calculate the probability of a crack pixel for each pixel using intensity information, and illustrate and analyze the detection capability using only pixel information; 2) we design SVM based approach to calculate the probability of a crack pixel using neighborhood information, and then compare the detection capability using information of different scale neighborhoods; 3) by demonstrating and analyzing the results of different fusion strategies, we propose a novel method to fuse the multiple probabilities of each pixel, which can significantly improve the detection accuracy; 4) we propose a weighted dilation operation that relies on the fused probability to enhance the recognition of borderline pixels and improve the crack continuity without increasing the crack widths improperly; 5) Finally, we evaluate the performance of the proposed approach using public data set in comparison with two state-of-the-art crack detection algorithms.

The remainder of the paper is organized as follows. Related works on pavement crack detection are discussed in Section II. Section III introduces the problem description and motivational examples. Section IV discusses the details of the proposed method for pavement crack detection, including procedures of probability maps generation, fusion method and weighted dilation operation, respectively. We evaluate the performance of the proposed approach in Section V using public pavement image dataset, and Section VI concludes the paper.

II. RELATED WORKS

Over the past decade, the problem of pavement crack detection has received wide attention. In this section, we review related works and highlight their differences with ours. These related works can be categorized into the following categories.

Intensity threshold based algorithms are the most common way to identify pavement cracks by setting different local or global threshold values, based on the observation

that crack pixels are usually darker than the surroundings and thus have lower intensity [9]–[11]. These works segment the cracks from the background using intensity histogram and dynamic thresholding. The thresholding based algorithms are easy to implement and can obtain acceptable results on certain road images with good conditions. However, they failed to produce satisfactory results when the contrast of cracks and the background is not strong. Furthermore, the unevenly distributed illuminate conditions will have a large impact on the final detection results. On the other hand, when the background is noisy and has a complex texture, intensity-threshold methods will not be able to detect the complete cracks, and they also cannot successfully detect complete cracks that have complex topologies.

Filter-based methods have also been widely adopted in pavement crack detection. Based on the intuition that cracks and edges are highly similar in morphology, some edge detectors like the Sobel filter and Canny filter were employed by researchers [22]–[25]. The major problem of these algorithms is that they cannot identify complete crack profiles when the cracks are of complex texture backgrounds. In recent years, with the development of signal processing techniques, more sophisticated and robust filters such as Gabor filters were used for the pavement crack detection problem [12], [13]. Instead of using a single filter, the work in [26] uses many pre-designed filters that strongly match cracks in pixel intensity, shape, orientation, etc., and the cracks will be extracted from the background after several filtering processes using the pre-designed filters. This algorithm can well transform the pavement image into a new space such that the cracks will be preserved and background will be dropped. Nevertheless, these matched filters are sensitive to the crack width, so that they cannot correctly identify pixels at crack borders.

Minimal path selection techniques have drawn extensive attention and emerged as a new approach in pavement crack detection in recent years [14]–[17], [27]. The main idea of minimal path selection is to find the best paths between node pairs in a graph. In the context of pavement crack detection, it can be used to identify continuous cracks using the threshold of minimal distances, based on the assumption that cracks are continuous at the pixel-level. In [15], Amhaz *et al.* proposed a crack detection method that is based only on the photometry and developed an algorithm for minimal path searching that is faster than Dijkstra's algorithm. Besides minimal path selection, some graph model algorithms derive minimal spanning trees instead of a minimum path for recovering continuity in pavement crack detection. For example, in [16], a crack detection framework called CrackTree was designed by Zou et al., which addressed the shadows and discontinuity problems in pavement crack detection. However, these approaches assume that cracks fragments are connected trough minimum path or minimum spanning tree, which is not true for many real cases. In addition, they only try to recover the continuity of cracks based on physical distances without fully utilizing other information contained in the local neighborhood.

Since the automation of pavement crack detection generally requires robust algorithms of a high level of intelligence, many *machine learning algorithms* have been introduced and applied to pavement crack detection tasks. In these algorithms, pavement images are first divided into many sub-blocks and are manually labeled. Next, a crack detection classifier will be trained. The works in [18] and [28] employ artificial neural networks (ANNs) for crack detection, while in [29]–[32], other traditional classifiers such as KNN (K-Nearest Neighbor), RBF (Radial Basis Function), etc., are applied using features that were extracted from sub-blocks. Due to the challenging tasks, developing sophisticated feature descriptors is usually necessary to capture more information about cracks, which help to improve the detection accuracy. It has been shown that, the texture features [33], standard deviation and mean parameters [36] contain important distinctive information that distinguishes crack blocks from noncrack blocks.

In recent years, *deep learning* has attracted significant attention due to its outstanding performance, and was demonstrated to be useful for road crack detection. In [34] and [35], Deep Convolutional Neural Networks (DCNN) has been designed for pavement crack detection. These algorithms have achieved better performance both in accuracy and efficiency for pavement crack detection. Nevertheless, deep learning algorithms require a large number of labeled images for training which are not easy to obtain.

III. PROBLEM DESCRIPTION AND MOTIVATION

A. PROBLEM DESCRIPTION

Pixel-level Crack detection: Given an input image I, the aim is to associate a binary label $c \in \{0, 1\}$ (where 0 represents non-crack and 1 represents crack in this paper) with each pixel in the image, to indicate whether this pixel belongs to a crack or not, based on the observed data at each pixel.

B. MOTIVATION EXAMPLES

Figure 1 shows 4 pixels with neighborhood information of different scales, i.e. 7×7, 13×13, 19×19 and 25×25.

Pixel (1) is a crack pixel with very low intensity, thus intensity information is positive for detection. From the figure we can also observe that in the 7×7 neighborhood, the amount of crack pixels still form a relatively large ratio of the neighborhood. However, with the increase of neighborhood size, the ratio of crack pixels decrease significantly, meaning that larger neighborhood size will have a negative impact on the pixel.

Pixel (2) is a non-crack pixel but has low intensity, thus intensity information is negative for detection. Its 7×7 neighborhood still have a negative impact on the pixel due to the fact that relatively large amount of pixels within the neighborhood have lower intensity. However, with the increase in neighborhood size, the impact of neighborhoods on the pixel tends to become positive.

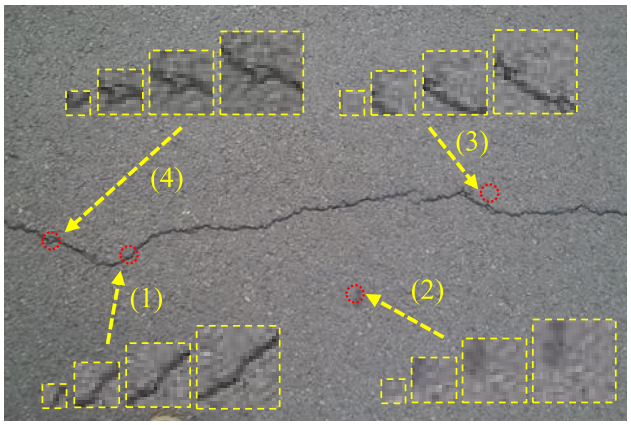


FIGURE 1. Intensity and neighborhoods information of crack and non-crack pixels.

Pixel (3) is a non-crack pixel and has high intensity, thus intensity information is positive for detection. Its 7×7 neighborhood also has strong positive impact on the pixel, as its close neighbors also have high intensities. However, since pixel (3) is not far away from the crack, the information of large size neighborhood will have negative impact on the detection.

Pixel (4) refers to the center point of the different scales' block and apparently has high intensity, but should be considered as a crack due to its location and the continuity property. In this case, intensity information is negative for detection, and information of small scale neighborhood has strong positive impact on the detection.

From the above analysis, it can be concluded that neighborhoods contain important information and neighborhoods of different sizes have different impacts on the detection for crack pixels. Thus in this paper, we will take the neighborhood information into consideration and use probabilities obtained based on the neighborhoods and pixel intensity as the descriptor of the pixel for crack detection.

IV. PROPOSED METHODS

A. OVERVIEW OF PROPOSED METHOD

In this paper, we propose novel probability based approach to achieve pixel-level pavement crack detection, by jointly considering pixel intensity and multi-scale neighborhood information. As shown in Figure 2, the proposed approach consists of three stages: *1) probability map construction*, *2) probability map fusion*, and *3) weighted dilation operation*. In the first stage, the probability map based on pixel intensity is constructed using a PGM method, while the probability maps based on multi-scale neighborhood information are constructed using an SVM method. In the second stage, the multiple probability maps are fused together and further used for crack detection. Then in the third stage, by relying on the fused probability map, a weighted dilation operation is performed on the cracks obtained from the second stage, to enhance the recognition of borderline pixels as well as

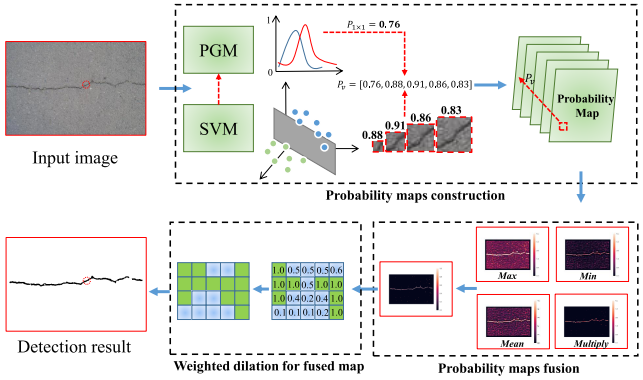


FIGURE 2. Framework of the proposed approach.

to improve the crack continuity without increasing the crack width improperly.

B. PROBABILITY MAP BASED ON PIXEL INTENSITY

Traditionally, intensity threshold based algorithms use the intensity to detect crack pixels because the crack pixels usually have lower intensity values than non-crack pixels. This is a fast and intuitive method, but often fails to obtain acceptable results in many situations. On the other hand, the pixel intensity provides important information that should not be neglected for deciding if the pixels are cracks or not. Thus in our approach, we extract the intensity information of all pixels and compute the probabilities that indicate the chances that each pixel should be regarded as a crack pixel, using probabilistic generative model. The probabilistic generative model is a method for generating all values for a phenomenon. And it has been shown to be able to obtain better performance than intensity threshold based algorithms in most binary classification problem.

We need to compute the posterior distribution $Pr(c|x)$ of pixel intensities, where x is the pixel intensity and $c \in \{0, 1\}$ is the label such that 0 stands for non-crack and 1 stands for crack. To this end, we first need to specify likelihood $Pr(x|c)$ and prior $Pr(c)$ using the training data-set which will be explained later, and then based on Bayes' rule we can obtain,

$$Pr(c|x) = \frac{Pr(x|c)Pr(c)}{\sum_{c=0}^{1} Pr(x|c)Pr(c)} \tag{1}$$

Since the data are univariate and continuous, we choose a univariate normal distribution and allow the variance σ^2 and the mean μ to be functions of the binary state c. Thus, we describe the likelihoods $Pr(x|c)$ as,

$$Pr(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2}$$

Explicitly,

$$Pr(x|\mu, \sigma^2) = Norm_x[\mu, \sigma^2] \tag{3}$$
$$Pr(x|c = 0) = Norm_x[\mu_0, \sigma_0^2] \tag{4}$$
$$Pr(x|c = 1) = Norm_x[\mu_1, \sigma_1^2] \tag{5}$$

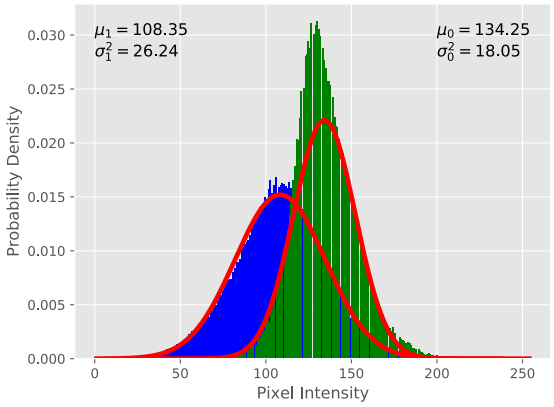


FIGURE 3. Intensity distribution fitting of crack and non-crack pixels.

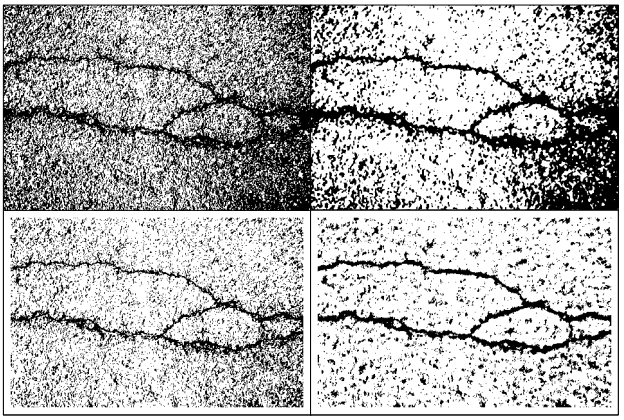


FIGURE 4. Detection results of threshold based methods and PGM based methods. (From left to right and top to bottom: threshold, threshold-OTSU, PGM using (6), PGM using (7).

In the learning procedure, the parameters $\mu_0, \sigma_0^2, \mu_1, \sigma_1^2$ are estimated based on the training data-set using Maximum-a-Posterior (MAP) fitting method. In particular, we obtain the μ_0, σ_0^2 from the subset of the training data-set where $c_i = 0$ and μ_1, σ_1^2 from the subset where $c_i = 1$. Figure 3 illustrates the fitting results for crack and non-crack data-set, from which it can be seen that the crack pixels has a lower mean and a higher deviation than non-crack pixels.

For prior $Pr(c)$, the most common prior probability model used in the binary classification problem is formulated as,

$$Pr(c) = \lambda^c (1-\lambda)^{1-c} \tag{6}$$

The prior parameter should be learned from the real states $\{c_i\}_{i=1}^m$ (m is the total number of samples) and a common method is to count the number of samples in each category.

With the likelihood $Pr(x|c)$ and prior $Pr(c)$ calculated using the above mentioned method, we can classify a new data point x as a crack or non-crack by Bayes' rule,

$$Pr(c=1|x) = \frac{Pr(x|c=1)Pr(c=1)}{\sum_{k=0}^{1} Pr(x|c=k)Pr(c=k)} \tag{7}$$

Specifically, it will classify the pixel as crack if $Pr(c = 1|x) \geq 0.5$. We conducted an experiment to demonstrate the classification capability using PGM method, as shown in Figure 4. Two existing intensity threshold based approaches, i.e. threshold, threshold-OTSU, are also implemented for comparison. In addition, we also implemented a second version of PGM method, by modifying the method for calculating the prior, as

$$Pr(c) = Median(P'_v) \tag{8}$$

where $P'_v = [P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$ is a vector of probabilities such that each of them indicates the probability of v to be a crack pixel obtained based on v's neighborhood information. The detailed calculation method of P'_v will be introduced in the next section.

Figure 4 shows the detection results obtained using four methods (two intensity threshold based methods and two PGM based methods), where 'threshold' method indicates the results obtained via setting an optimal global threshold value, 'threshold-OTSU' indicates the results obtained via the OTSU method [37], 'PGM using (6)' indicates the results obtained via the PGM method based on equation (6), while 'PGM using (8)' indicates the results obtained via the PGM method based on equation (8). From the figure, it is evident that PGM methods outperform intensity threshold based methods, and PGM using equation (8) performs better than that which uses (6). However, the result by PGM using (8) is still not acceptable for pavement crack detection.

This is because the intensity of pixels can only provide limited information on the associated labels, making it very difficult to detect cracks with high accuracy. On the other hand, the neighborhood of a pixel also contain important information that can be used to improve the crack detection accuracy. In the next section, we develop approaches to explore the information contained in the neighborhood. To distinguish with neighborhood information, we use $P_{1\times1}$ to indicate the probability map generated using intensity information without considering neighborhood information, i.e. $P_{1\times1}$ is the set of $Pr(c = 1|x)$ for each pixel x.

C. PROBABILITY MAPS BASED ON NEIGHBORHOOD INFORMATION

As mentioned before, the neighbors' information (information of neighborhood) of a pixel is also crucial for determining whether the pixel is a crack or not. This is because pixels of crack and non-crack have significant differences in neighborhood, as we have demonstrated in the motivation example in Section III(B). For each pixel, the surrounding information of different size blocks can be significantly different. Moreover, as we have demonstrated in Figure 4, the method 'PGM using (8)' obtained better results than 'PGM using (6)', which implies that neighborhood contain more distinctive information than the intensity of the pixel.

Therefore, it is essential to take neighborhood information into consideration. In this section, we develop methods to calculate the probability vector $P'_v = [P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$, where $P^v_{d\times d}$ indicates the

probability that the pixel should be regarded as a crack pixel based on v's neighborhood information with neighborhood size of $d \times d$. We next introduce the proposed method for calculating P'_v.

For each scale $d \times d \in \{7 \times 7, 13 \times 13, 19 \times 19, 25 \times 25\}$, we extract the neighborhood areas for each pixel to form a dataset Ω_d, formally,

$$\Omega_d = \{(X_i, y_i) \mid x_i = 1, 2, 3, \ldots, m\} \tag{9}$$

where m is the number of pixels, $X_i =< x_{i,1}, x_{i,2}, \ldots, x_{i,d^2} >$ is the set of pixels in x_i's neighborhood, and $y_i \in \{0, 1\}$ is the label of pixel x_i such that $y_i = 1$ if x_i is a crack pixel and $y_i = 0$ otherwise.

We develop an SVM method to calculate $P^v_{d\times d}$ using dataset Ω_d. SVM solve problems based on a straightforward idea: construct a hyperplane that sets apart the classes of data [38].

Different from the previous methods, which describe a block by extracting block features such as mean value, standard deviation matrix, histogram of oriented gradient (HOG), local binary pattern (LBP) etc., we rely on the original intensity of each pixel of the neighborhood block to characterize the neighborhood information. The training procedure can be formulated as the following optimization problem.

$$\begin{aligned} &\min_{w,b,\zeta} \frac{1}{2}||\omega||^2 + C\sum_{i=1}^{m}\zeta_i \\ &\text{s.t. } y_i(\omega \cdot x_i + b) \geq 1 - \zeta_i \\ &\quad \zeta_i \geq 0 \quad \text{for any i} = 1, \ldots, \text{m} \end{aligned} \tag{10}$$

where w, b, ζ are normal vector to the hyperplane, offset of the hyperplane, and slack variables, respectively.

Then the problem is further converted as an unconstrained optimization problem with loss function $\zeta(\omega; x_i, y_i)$,

$$\min_{\omega} \frac{1}{2}||\omega||^2 + C\sum_{i=1}^{m}\zeta(\omega; x_i, y_i) \tag{11}$$

where $C > 0$ is a penalty parameter, and $\zeta(\omega; x_i, y_i)$ is formulated as

$$\zeta(\omega; x_i, y_i) = (max(0, 1 - y_i\omega^T x_i))^2 \tag{12}$$

Standard SVM output a class label instead of a indicator of probability that the pixel should be detected to be a crack pixel. As such, we adopted the strategy introduced by John [39] to obtain the required probability indicator, by transforming SVM predictions to posterior probabilities using the following method.

$$P(y = 1|c) = \frac{1}{1 + e^{(Ac+B)}} \tag{13}$$

where c is the output of the standard SVM and A and B are parameters estimated using maximum likelihood estimation. The gradient descent approach is used to find A and B.

By applying the SVM method on dataset Ω_7, Ω_{13}, Ω_{19}, Ω_{25}, we can calculate a probability vector $P'_v = [P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$ for each pixel v. Each probability of the vector can be used to classify v to be either crack or non-crack pixel for pavement crack detection. Figure 5 shows the detection results based on the probabilities of neighborhood of 4 different scales. From the figure, it can be observed that using probability from large neighborhood, e.g. $P^v_{25\times25}$, can produce more cracks with less noises, however, the obtained crack width tend to be larger than the actual width, which will result in negative impact on road quality assessment. On the contrary, the detection using probability based on smaller neighborhood, e.g. $P^v_{7\times7}$, obtains more accurate result on cracks, but produces too many noisy results (i.e. non-crack pixels that are classified as cracks due to small intensity values).

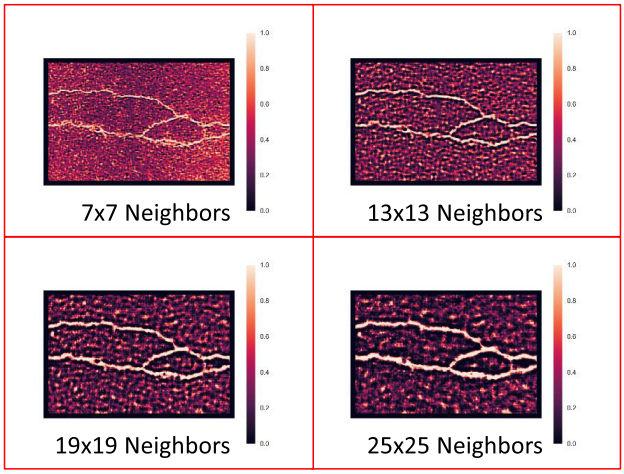


FIGURE 5. Probability maps obtained using information of different scale neighborhoods.

D. DATA FUSION METHOD

By combining the pixel intensity based probability $P^v_{1\times1}$ and the neighborhood information based probability vector $P'_v = [P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$, we obtain a probability vector $P_v = [P^v_{1\times1}, P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$ for the pixel v, and each of them can be used to classify v to be either crack or non-crack pixel but none of them can obtain very high accuracy separately. In this section, we develop a novel method to combine the five probabilities in order to achieve better detection accuracy.

Figure 6 illustrates 4 probability maps which are obtained by fusing the probability vector using Max operation, Min operation, Multiply operation and Mean operation respectively. Taking the first strategy, i.e. using Max operation as an example, the new probability for pixel v of the fused probability map is obtained by taking the maximum probability from the vector $P_v = [P^v_{1\times1}, P^v_{7\times7}, P^v_{13\times13}, P^v_{19\times19}, P^v_{25\times25}]$. Similarly, Min operation takes the minimum probability from the probability vector P_v to construct the fused probability map, Multiply operation calculates the fused probability as the product of the five elements of vector P_v, and Mean operation calculates the fused probability as the mean of

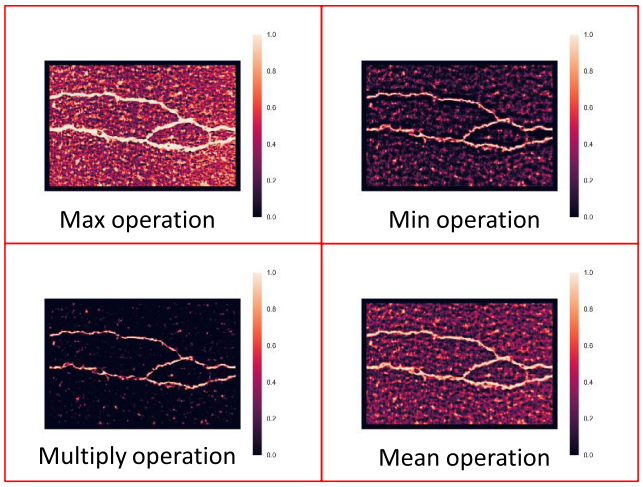


FIGURE 6. Fused probability map obtained using different fusion methods.

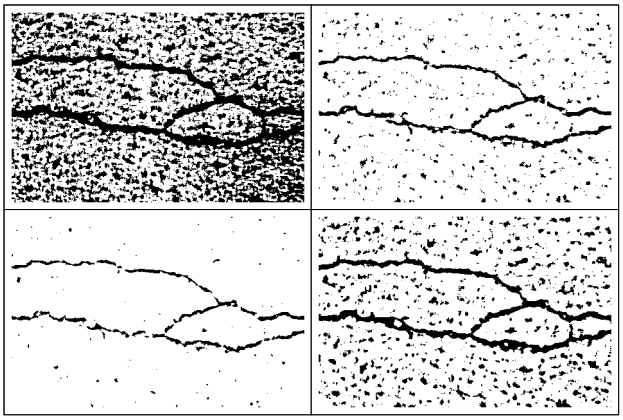


FIGURE 7. Detection results using the probability maps illustrated in Figure 6.

vector P_v. The results indicate that Max operation can detect the most number of cracks, as it tries to maximize the chance of classifying each pixel to cracks. As a result, a large number of non-crack pixels are classified as cracks. On the other hand, the Multiply operation only produces a small amount crack pixels leading to a certain amount of missed cracks. This is because a pixel, say v, can be classified as cracks only if the product of $P^v_{1\times1}$, $P^v_{7\times7}$, $P^v_{13\times13}$, $P^v_{19\times19}$ and $P^v_{25\times25}$ are large enough, which is a fairly strong constraint. The advantages of Multiply operation is that only a few noises are generated. The Min operation and Mean operation achieve results between Max operation and Multiply operation.

For ease of presentation, we define probability map as a matrix of all pixels probabilities. Formally, the probability map $P_{1\times1}$ consists of probability $P^v_{1\times1}$ for each pixel v a the input image. In this way, five probability maps can be constructed, i.e. $P_{1\times1}$, $P_{7\times7}$, $P_{13\times13}$, $P_{19\times19}$, and $P_{25\times25}$. We propose an efficient algorithm, as shown in Algorithm 1, to fuse the 5 probability maps to improve the crack detection accuracy.

Algorithm 1 Algorithm for Probability Maps Fusion

Input: Probability map of
$P_v = [P_{1\times1}, P_{7\times7}, P_{13\times13}, P_{19\times19}, P_{25\times25}]$.
Output: Fusion result P_{fused}.
begin
$T \leftarrow mean(P_v)$; /* Initialization */
foreach *pixel v in the input image* **do**
  $P_v \leftarrow \emptyset$;
  $vote \leftarrow 0$;
  foreach $d \in \{1, 7, 13, 19, 25\}$ **do**
    $P_v \leftarrow P_v \cup \{P^v_{d\times d}\}$;
    if $P^v_{d\times d} \geq T$ **then**
      $vote \leftarrow vote + 1$
  if $vote = 5$ **then**
    $P^v_{fused} \leftarrow MaxOperation(P_v)$
  else
    $P^v_{fused} \leftarrow MultiplyOperation(P_v)$
end

E. WEIGHTED-DILATION FOR CRACK CONTINUITY

Since we employ a strong constraint during the fusion procedure of probability maps, such that only pixels with high probability (in the fused probability map) can be classified as cracks. However, the pixels located at the borders of the cracks usually cannot obtain high probability, thus cannot be identified using the fused probability map. In this section, we develop a novel method to recover the border pixels as well as to improve the continuity of the cracks.

The dilation operation is a widely used approach for morphological image processing, which uses a structuring element for probing and expanding the shapes contained in the input image. In the pavement crack detection problem, binary dilation has been used to improve the connectivity of obtained cracks by changing the label of border pixels from non-cracks to cracks. However, the traditional binary dilation often wrongly increase the width of the cracks and thus decrease the detection precision.

In order to overcome this drawback as well as increase the crack continuity, in this paper, we extend the binary dilation operation to weighted-dilation operation which works on the probability maps obtained in the previous procedure.

After the fusion procedure of probability maps, we obtain a fused probability map in which crack pixels have probability values close to 1 and non-crack pixels have probability values close to 0. Since the neighborhood of pixel contain critical information, in weighted-dilation operation, we first compute the mean probability of the structuring element (covering the neighborhood of the current pixel), and check if it is greater than 0.5 as a condition to make a further decision on the dilation operation.

Figure 8 shows a sub-region that contains two separate cracks, one on the left and the other on the right. Light color indicates higher probability and darker color indicates

FIGURE 8. A sub-region of probability map where cracks are disconnected.

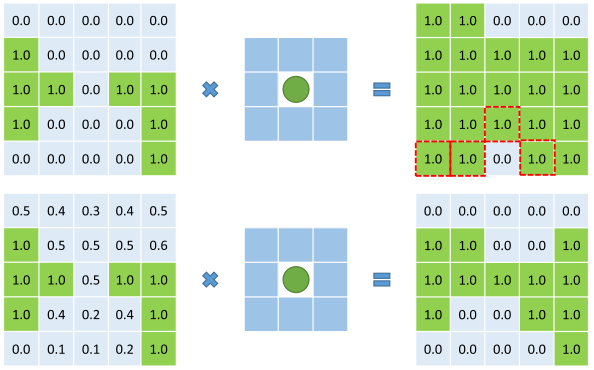


FIGURE 9. Comparison of results between traditional and weighted dilation operation (From top to bottom: traditional dilation result, weighted-dilation with mean kernel).

lower probability of being crack pixels. It can be seen that in the middle of the dotted box, the probability of being a crack pixel is 0.5, and thus it will not be considered as a crack. However, this should be regarded as a crack pixel due to the continuity property. Therefore, we develop the weighted-dilation operation to achieve this goal. In Figure 9, the first row is the result of the traditional dilation operation, the second is the result obtained by weighted dilations that use the mean value of the kernel (structuring element). It can be observed that traditional dilation operation has widened the cracks such that the pixels adjacent to crack borders are also marked as crack pixels. However, this should not be the case for many of them. The results of the second row show that our weighted dilations selectively recover pixels from non-crack to crack based on probability map. In this way, better accuracy at crack borders can be obtained, which not only enhances the continuity but also improves the detection precision.

V. RESULTS AND ANALYSIS

The proposed method as well as the baseline approaches are implemented using Python 3.5 and OpenCV, running on Intel(R) Xeon(R) CPU E5-1630v4@3.70GHz with 32GB RAM, on a server with an operating system of Windows 10 Pro. The performance of our method is evaluated by comparing it with two state-of-the-art algorithms using a public dataset.

A. DATASETS AND BASELINE APPROACHES

CFD data-set [19] is a public pavement image dataset that consists of 118 images with the resolution of 480x320 pixels.

TABLE 1. Confusion matrix of pavement crack detection.

	actual crack	actual non-crack
predicted crack	TP	FP
predicted non-crack	FN	TN

The images in this data set contain a significant amount of noisy pixels like oil spots and water stains, and some of them are under the poor illumination condition. Similar to [19] and [40], 60% data (70 figures) are used for training and 40% (48 figures) for testing.

The CrackForest [19] is the first method that has achieved relatively higher performance in CFD data set, and recently, WFCD and WMIP [40] have demonstrated better results than CrackForest on the CFD data set. In this section, we evaluate our approach by comparing it with two baseline approaches, i.e. WFCD and WMIP.

B. EVALUATION METRICS

Table 1 shows a table of confusion matrix that consists of two rows and two columns which reports the number of true positives(TP), true negatives(TN), false positives(FP) and false negatives(FN) that allows visualization of the performance of a classification algorithm. Based on the TP, FN and FP, precision, recall and F1-score can be defined as follows.

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{16}$$

In addition to the above indicators, we also compute the Area Under the Receiver Operating Characteristic Curve (ROC AUC), as shown in Figure 11. ROC curve is a commonly used graph that summarizes the performance of a classifier over all possible thresholds.

Since acquiring a high quality ground truth is difficult for real images, existing works (such as CrackForest, WFCD, WMIP, etc.) in [15], [19], and [40] allow a tolerance margin for measuring the coincidence between the detected cracks and the ground truth. They assume that TP pixels are included within a 2 and 5-pixel vicinity of the ground truth. That is the detected crack pixels which are no more than five pixels away from the manually labeled pixel are true positive pixels. In this paper, we also test the results with consideration of tolerance margin, as shown in Table 2. However, since '5 pixel vicinity' will hide important information of cracks with respect to crack widths, we limit the tolerance margin to no more than 2-pixel vicinity.

C. RESULTS AND ANALYSIS

The comparison of our proposed approach with the two baselines, in terms of Precision, Recall and F1-score is shown in Table 2. The notation '0-pixel' indicates that tolerance

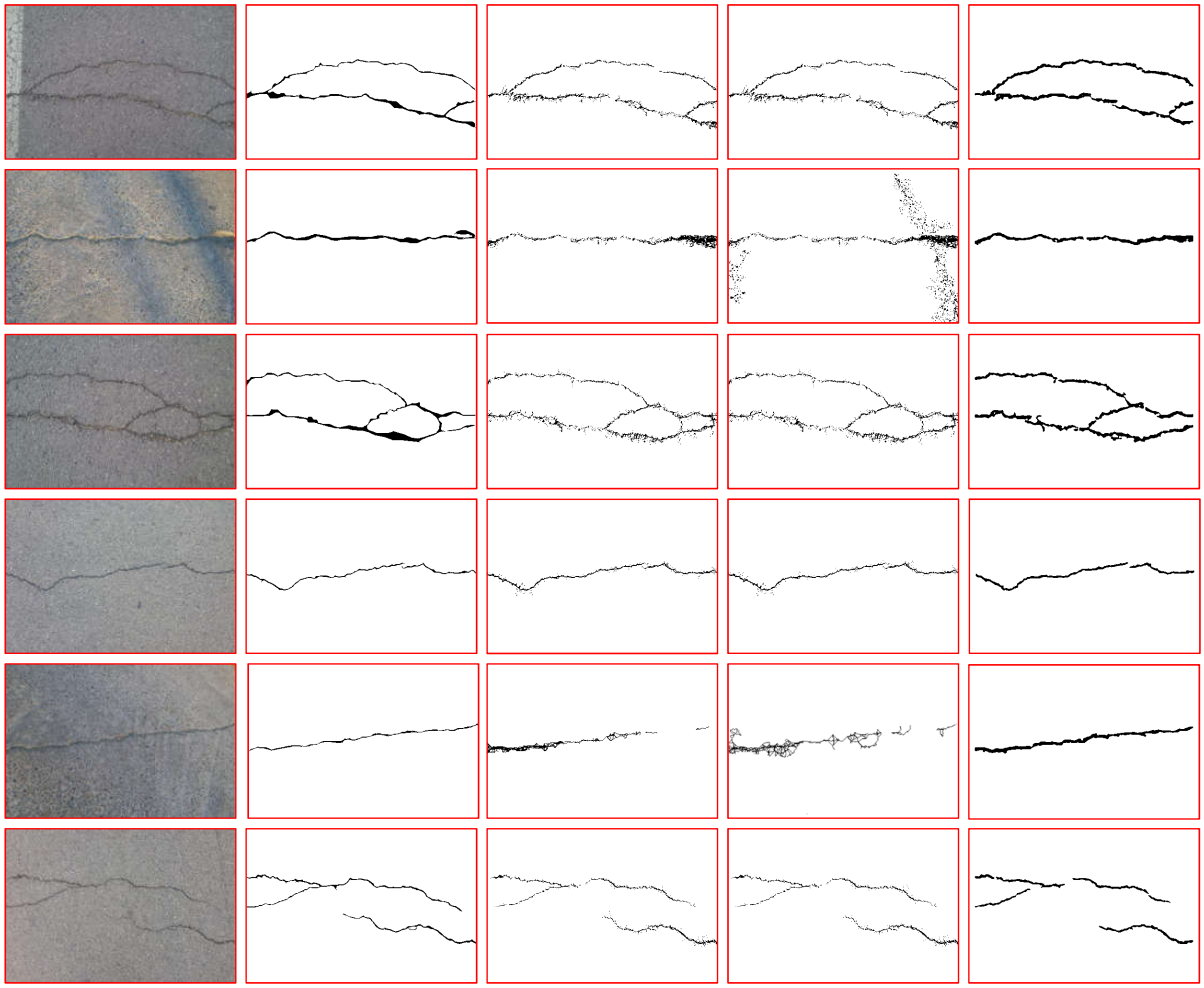


FIGURE 10. Example of crack detection results. (From left to right: original image, ground truth, WFCD, WMIP, ours).

TABLE 2. Comparison of precision, recall and F1-score.

	Precision	Recall	F1-score
ours with 2-pixels	**0.907**	**0.846**	**0.870**
MFCD with 2-pixels	0.858	0.597	0.689
WMIP with 2-pixels	0.773	0.581	0.643
ours with 1-pixel	**0.789**	**0.829**	**0.800**
MFCD with 1-pixel	0.788	0.567	0.650
WMIP with 1-pixel	0.712	0.560	0.606
ours with 0-pixel	0.471	**0.757**	**0.567**
MFCD with 0-pixel	**0.548**	0.489	0.500
WMIP with 0-pixel	0.501	0.474	0.467

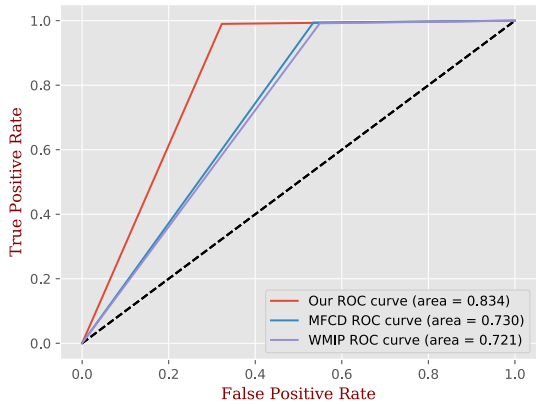


FIGURE 11. Comparison on AUC of the ROC curves.

margin is not utilized while '1-pixel' and '2-pixel' indicate that the tolerance margins of 1 and 2-pixel vicinity have been used for results collection. From the Table 2, it can be observed that, without using the tolerance margin, our method significantly outperform the two baselines in terms of Recall and F1-score while achieving Precision close to the baseline approaches. When considering the tolerance margin, our method performs even better. And when the tolerance margin is set to 2-pixel vicinity, our method can achieve Precision as high as 90.7%.

Figure 10 shows the visual comparison of results obtained using our method and the two baselines, as well as the original

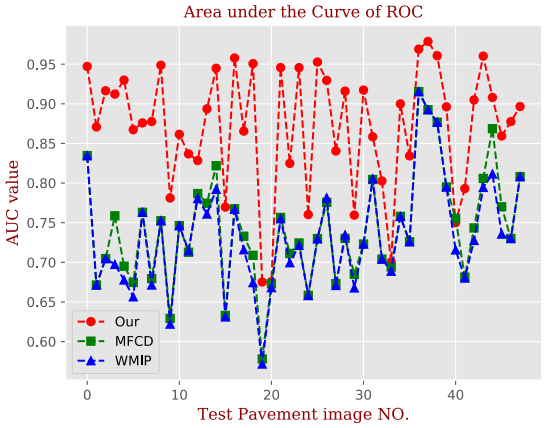


FIGURE 12. ROC-AUC results for each test image.

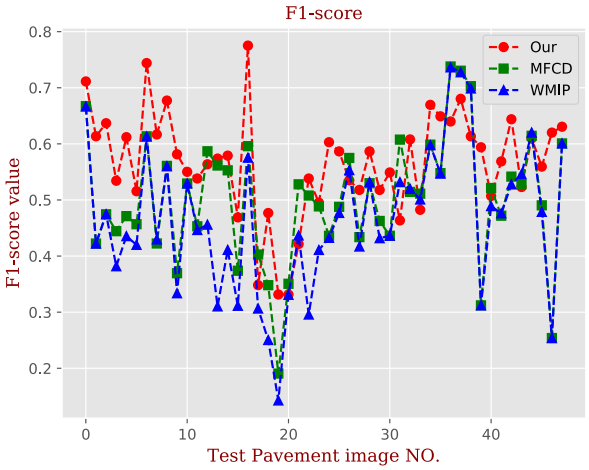


FIGURE 13. F1-score with 0-pixel vicinity of tolerance margin for each test image.

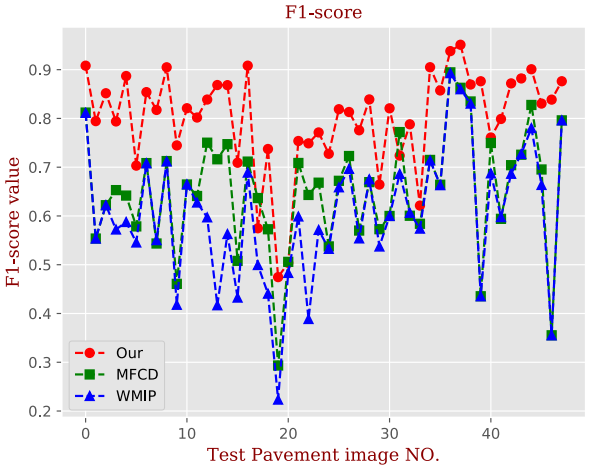


FIGURE 14. F1-score with 1-pixel vicinity of tolerance margin for each test image.

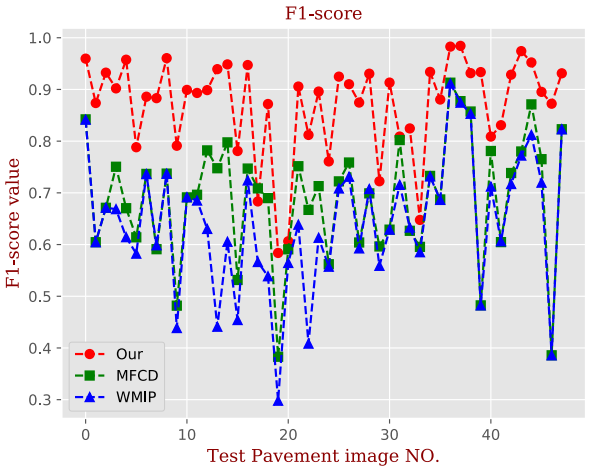


FIGURE 15. F1-score with 2-pixel vicinity of tolerance margin for each test image.

image and ground truth. We can observe intuitively that our method outperforms the alternatives, as it not only produces fewer noises but maintains the continuity of cracks. Both the method WFCD and WMIP cannot perform well on images with noisy pixels. As a result, they classify a lot of non-crack pixels as cracks due to the lower intensities in their neighborhood.

The ROC curves of the three methods are illustrated in Figure 11. Based on this, it can be concluded that our approach significantly outperforms the two baselines, as our ROC curve subsumes the curves of both WFCD and WMIP.

To show a more detailed comparison of our approach and the two baselines, we demonstrate the results of AUC value and F1-score of the three methods on each of the test image with the tolerance margin is set to 0-pixel, 1-pixel and 2-pixel as shown in Figure 14, Figure 13 and Figure 15, respectively. The results indicate that our method significantly outperforms the baselines in AUC and F1-score.

VI. CONCLUSION

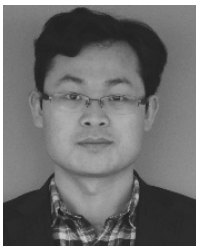
We address the pixel level pavement crack detection problem by leveraging on multi-scale neighborhood information as well as the pixel intensity, to meet the challenges posed by heterogeneous pixel intensity, complex crack topology, poor illumination condition, and noisy texture background such as oil spots, water stains, shadow casting etc. We have demonstrated that neighborhoods contain critical information for crack detection and different size of neighborhoods have significant impact on the detection results. We developed novel PGM based method to generate probability map based on pixel intensity information, and SVM based method to generate probability maps based on multi-scale neighborhood information. We also compared the results of probability fusion strategies and designed a novel fusion algorithm to merge the multiple probability maps into a fused map that can detect cracks with high accuracy. To enhance the recognition of borderline pixels as well as improve the crack continuity, we have proposed a weighted dilation operation to further optimize the detected cracks, which also relies on the fused probability maps. Experimental results have demonstrated

the superiority of our approach in terms of precision, recall, f1-score and ROC, over the state-of-the-art pavement crack detection algorithms. In future, we will continue to investigate how the neighborhoods will affect the detection results and determine features to extract from the neighborhoods. We also plan to accelerate the detection by using CUDA as well as further optimizing the detection algorithms.

REFERENCES

[1] K. C. P. Wang, "Designs and implementations of automated systems for pavement surface distress survey," *J. Infrastruct. Syst.*, vol. 6, no. 1, pp. 24–32, 2000.
[2] F. Kalim, J. P. Jeong, and M. U. Ilyas, "CRATER: A crowd sensing application to estimate road conditions," *IEEE Access*, vol. 4, pp. 8317–8326, 2016.
[3] W. Zhang, Z. Zhang, D. Qi, and Y. Liu, "Automatic crack detection and classification method for subway tunnel safety monitoring," *Sensors*, vol. 14, no. 10, pp. 19307–19328, 2014.
[4] G. Li, X. Zhao, K. Du, F. Ru, and Y. Zhang, "Recognition and evaluation of bridge cracks with modified active contour model and greedy search-based support vector machine," *Autom. Construction*, vol. 78, pp. 51–61, Jun. 2017.
[5] S. J. Schmugge *et al.*, "Detection of cracks in nuclear power plant using spatial-temporal grouping of local patches," in *Proc. IEEE WACV*, Lake Placid, NY, USA, Mar. 2016, pp. 1–7.
[6] F.-C. Chen, M. R. Jahanshahi, R.-T. Wu, and C. Joffe, "A texture-based video processing methodology using Bayesian data fusion for autonomous crack detection on metallic surfaces," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 4, pp. 271–287, 2017.
[7] Z. Luo, Z. Zhu, H. Ruan, and C. Shi, "Extraction of microcracks in rock images based on heuristic graph searching and application," *Comput. Geosci.*, vol. 85, pp. 22–35, Dec. 2015.
[8] D. S. Mahler, Z. B. Kharoufa, E. K. Wong, and L. G. Shaw, "Pavement distress analysis using image processing techniques," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 6, no. 1, pp. 1–14, 1991.
[9] H. D. Cheng, X. J. Shi, and C. Glazier, "Real-time image thresholding based on sample space reduction and interpolation approach," *J. Comput. Civil Eng.*, vol. 17, no. 4, pp. 264–272, 2003.
[10] H. Oliveira and P. L. Correia, "Automatic road crack segmentation using entropy and image dynamic thresholding," in *Proc. IEEE EUSIPCO*, Glasgow, U.K., Aug. 2009, pp. 622–626.
[11] Q. Li and X. Liu, "Novel approach to pavement image segmentation based on neighboring difference histogram method," in *Proc. IEEE CISP*, vol. 2. Tianjin, China, May 2009, pp. 792–796.
[12] E. Zalama, J. Gómez-García-Bermejo, R. Medina, and J. Llamas, "Road crack detection using visual features extracted by Gabor filters," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 29, no. 5, pp. 342–358, 2014.
[13] M. Salman, S. Mathavan, K. Kamal, and M. Rahman, "Pavement crack detection using the Gabor filter," in *Proc. IEEE ITSC*, The Hague, The Netherlands, Oct. 2013, pp. 2039–2044.
[14] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart, "A new minimal path selection algorithm for automatic crack detection on pavement images," in *Proc. IEEE ICIP*, Paris, France, Oct. 2014, pp. 788–792.
[15] R. Amhaz, S. Chambon, J. Idier, and V. Baltazart, "Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 10, pp. 2718–2729, Oct. 2016.
[16] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "CrackTree: Automatic crack detection from pavement images," *Pattern Recognit. Lett.*, vol. 33, no. 3, pp. 227–238, 2012.
[17] K. Fernandes and L. Ciobanu, "Pavement pathologies classification using graph-based features," in *Proc. IEEE ICIP*, Paris, France, Oct. 2014, pp. 793–797.
[18] L. Li, L. Sun, G. Ning, and S. Tan, "Automatic pavement crack recognition based on BP neural network," *Promet-Traffic Transp.*, vol. 26, no. 1, pp. 11–22, 2014.
[19] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 12, pp. 3434–3445, Dec. 2016.
[20] Y.-C. Tsai, V. Kaul, and R. M. Mersereau, "Critical assessment of pavement distress segmentation methods," *J. Transp. Eng.*, vol. 136, no. 1, pp. 11–19, 2010.
[21] B. J. Lee and H. D. Lee, "Position-invariant neural network for digital pavement crack analysis," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 19, no. 2, pp. 105–118, 2004.
[22] A. P. Albert and A. O. Nii, "Evaluating pavement cracks with bidimensional empirical mode decomposition," Ì *EURASIP J. Adv. Signal Process.*, vol. 2008, no. 1, pp. 1–7, 2008.
[23] O. A. Aiguo and W. Yaping, "Edge detection in pavement crack image with beamlet transform," in *Proc. IEEE EMEIT*, Liaoning, China, Sep. 2012, pp. 881–883.
[24] B. Santhi, G. Krishnamurthy, S. Siddharth, and P. K. Ramakrishnan, "Automatic detection of cracks in pavements using edge detection operator," *J. Theor. Appl. Inf. Technol.*, vol. 36, no. 2, pp. 199–205, 2012.
[25] A. Nisanth and A. Mathew, "Automated visual inspection on pavement crack detection and characterization," *Int. J. Technol. Eng. Syst.*, vol. 6, no. 1, pp. 14–20, 2014.
[26] A. Zhang, Q. Li, K. C. P. Wang, and S. Qiu, "Matched filtering algorithm for pavement cracking detection," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2367, no. 1, pp. 30–42, 2013.
[27] M. Avila, S. Begot, F. Duculty, and T. S. Nguyen, "2D image based road pavement crack detection by calculating minimal paths and dynamic programming," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2014, pp. 783–787.
[28] G. Xu, J. Ma, F. Liu, and X. Niu, "Automatic recognition of pavement surface crack based on BP neural network," in *Proc. IEEE ICCEE*, Phuket, Thailand, Dec. 2008, pp. 19–22.
[29] H. Oliveira and P. L. Correia, "Supervised strategies for cracks detection in images of road pavement flexible surfaces," in *Proc. 16th Eur. Signal Process. Conf. (EUSIPCO)*, Lausanne, Switzerland, Aug. 2008, pp. 1–5.
[30] X. Sun, J. Huang, W. Liu, and M. Xu, "Pavement crack characteristic detection based on sparse representation," *EURASIP J. Adv. Signal Process.*, vol. 2012, no. 1, p. 191, 2012.
[31] M. Gavilán *et al.*, "Adaptive road crack detection system by pavement classification," *Sensors*, vol. 11, no. 10, pp. 9628–9657, 2011.
[32] H. Oliveira and P. L. Correia, "CrackIT—An image processing toolbox for crack detection and characterization," in *Proc. IEEE ICIP*, Paris, France, Oct. 2014, pp. 798–802.
[33] A. Cord and S. Chambon, "Automatic road defect detection by textural pattern recognition based on AdaBoost," *Comput.-Aided Civ. Inf.*, vol. 27, no. 4, pp. 244–259, 2011.
[34] L. Zhang, F. Yang, Y. D. Zhang, and Y. J. Zhu, "Road crack detection using deep convolutional neural network," in *Proc. IEEE ICIP*, Phoenix, AZ, USA, Sep. 2016, pp. 3708–3712.
[35] A. Zhang *et al.*, "Automated pixel-level pavement crack detection on 3D asphalt surfaces using a deep-learning network," *Comput.-Aided Civil Infrastruct. Eng.*, vol. 32, no. 10, pp. 805–819, 2017.
[36] H. Oliveira and P. L. Correia, "Automatic road crack detection and characterization," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 1, pp. 155–168, Mar. 2013.
[37] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
[38] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
[39] J. C. Platt, "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods," *Adv. Large Margin Classifiers*, vol. 10, no. 3, pp. 61–74, 1999.
[40] H. Li, D. Song, Y. Liu, and B. Li. (2017). *Automatic Pavement Crack Detection by Multi-Scale Image Fusion*. [Online]. Available: https://pdfs.semanticscholar.org/75d1/07d77886b8a6ff86343cc72e905c47a366a1.pdf

DIHAO AI received the B.Eng. degree in electronics science and technology from Liaoning Technical University, Liaoning, China. He is currently pursuing the Ph.D. degree in safety science and engineering from the China University of Mining and Technology, Beijing, Beijing, China. He is currently a Visiting Student with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests include safety monitoring system design, disaster prediction using image processing techniques, and machine learning algorithms.



GUIYUAN JIANG received the B.S. degree from Northwest University for Nationalities, China, the M.Eng. degree from Tianjin Polytechnic University, China, and the Ph.D. degree from Tianjin University, all in computer science and technology. He is currently a Research Fellow with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests include data analytics for citywide transportation modeling and optimization, design methodologies for reconfigurable computing systems, and resource optimization for datacenter and sensor networks.

LAM SIEW KEI received the B.A.Sc., M.Eng., and Ph.D. degrees from the School of Computer Science and Engineering (SCSE), Nanyang Technological University. He was a Visiting Research Fellow with the Imperial College of London, University of Warwick, and RWTH Aachen, Germany. He is currently an Assistant Professor with SCSE focuses on devising custom computing solutions to meet the challenging demands of energy-efficiency, reliability, and security in embedded systems. His current projects include architecture-aware algorithms for vision-enabled sensing, design methodologies for secure and reliable embedded systems, and transportation analytics.

CHENGWU LI received the Ph.D. degree from the China University of Mining and Technology, Beijing (CUMTB), China, in 2005. He is currently a Professor with CUMTB, and also serves as a Director of the Safety Engineering Department. His research interests include safety evaluation and management, mine disaster monitoring, and safety production inspecting. He has conducted many NSFC-funded programs, include robust mine safety monitoring technologies, disaster accident investigation specification, and early warning information systems for mine disasters.