electronics

MDPI

Article

A Novel Path Voting Algorithm for Surface Crack Detection

Jianwei Yu [1,2,*], Zhipeng Chen [1] and Zhiming Xiong [1]

[1] School of Architecture and Urban Planning, Research Institute for Smart Cities, Shenzhen University & Key Laboratory of Urban Land Resources Monitoring and Simulation, Ministry of Natural Resources, Shenzhen 518060, China; chenzp@whu.edu.cn (Z.C.); xiongzm1987@hotmail.com (Z.X.)

[2] Wuhan Hirail Profiling Technology Co., Ltd., Wuhan 430060, China

* Correspondence: jwyu@szu.edu.cn

Abstract: Path voting is a widely used technique for line structure detection in images. Traditional path voting, based on minimal-path, is performed to track paths based on how seeds grow. The former requires to set a starting point and an end point. Thus, the performance of minimal-path path voting depends on the initialization. However, high-quality initialization often requires human interaction, which limits its applications in practice. In this paper, a fully automatic path voting method has been proposed and applied for crack detection. The proposed path voting is performed to segment images, which partitions an image patch along the potential crack path and integrates the path to form a crack probability map. After path voting, crack seeds are sampled and modeled into a graph, and the edge weights are assigned using an attraction field algorithm. Finally, cracks are extracted by using spanning tree and tree pruning algorithms. Experimental results demonstrate that the proposed path voting approach can effectively infer the cracks from 2D optic images and 3D depth images.

Keywords: crack detection; path voting; minimal-path; pavement crack; defect detection

check for updates

Citation: Yu, J.; Chen, Z.; Xiong, Z. A Novel Path Voting Algorithm for Surface Crack Detection. *Electronics* **2022**, *11*, 501. https://doi.org/10.3390/electronics11030501

Academic Editor: Jose Santamaria

Received: 22 December 2021
Accepted: 4 February 2022
Published: 8 February 2022

1. Introduction

Cracks are common defects that can be found on the surface of various types of physical structures such as metal surfaces, road pavement, and plastic shells. Although cracks are not troublesome defects, detection and reparation reduce costs, as a crack can quickly deteriorate into an important defect such as a hole. As a result, crack detection is generally a necessary and periodic operation for many engineering projects. On the other hand, crack detection is a time-consuming and labor-intensive task when performed by human staff.

In the past two decades, a large number of crack detection methods have been proposed. Pavement crack detection has attracted wide attention from both the academia and the industry [1,2] due to its importance and urgency. The goal of these methods/systems is to detect and locate pavement cracks automatically in the pavement images. For pavement data collection, techniques have evolved from 2D optic imaging [3–7] to 3D depth imaging [8–10]. Prior to 3D laser imaging technology, high speed, and high-resolution CCD cameras were used for real-time pavement image-based collection, which gave birth to a number of image-based crack detection methods [11–15]. Note that line scan imaging captures pavement accurately and is used by most of the methods for image collection. However, crack detection that uses 2D optic images has to deal with the following problems. First, the intensity of a crack pixel is very sensitive to the direction of the exposure light. Under some exposure directions, the shadow can be formed in the crack area which makes the crack appear darker than the background. While under other exposure directions, no discriminative shadow will be formed in the crack area, especially when the exposure light shoots directly inside the crack. As the camera is commonly mounted onto the vehicle rear with a fixed pose, the captured optic images often capture and represent pavement cracks with low contrast. Second, some cracks may have low continuity under frequent rolling of wheels with heavy loadings. The vehicles, especially the heavy truck, run on the road pavement day by day, which damages some parts of the crack walls, and makes the crack less distinctive under the light. An example image is shown in Figure 1a. Third, possible shadows have a similar intensity to cracks. The vehicle body, as well as the road facilities such as trees and light poles, cast shadows on the pavement. Cracks outside the shadow would have a similar intensity to the pavement in the shadow, and cracks inside the shadow will have decreased contrast to the undamaged pavement.

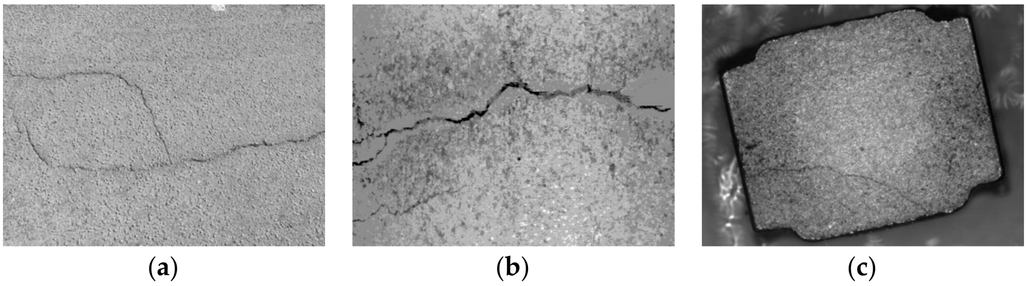

(**a**) (**b**) (**c**)

Figure 1. Three example cracks. (**a**) A crack with low continuity in a 2D optic pavement image (from CrackTree206). (**b**) A pavement crack in a 3D depth image (from CrackPV14). (**c**) A crack in a material profile image (from StoneCrack50).

In recent years, 3D laser imaging technology has become mature and has gradually been applied to collect pavement data, e.g., the 3D depth image [16–18]. Generally, the 3D laser imaging system is comprised of laser light, a highspeed camera, and a computing unit. The laser projects a beam on the ground to form a line, the camera captures the line of light, and the computing unit calculates the 3D depth based on the line structure theory. Three-dimensional laser imaging is widely used as it can reduce the influence of cast shadows when compared to optic imaging technologies. An example depth image captured by 3D laser imaging is shown in Figure 1b. Consequently, crack detection using 3D depth images is performed, and has become a popular research topic in recent years. Despite pavement maintenance and testing, crack detection has also been required in many other applications, e.g., the visual examinations of nuclear power plant components [19] and the inspection of crack defects on the material profile, as shown in Figure 1c.

Although cracks may show different appearances in different kinds of images, they share some characteristics which make them special and discriminative to other objects. One major characteristic of the crack is that a crack is a linear/curvilinear structure that has a relatively lower intensity than the background. Based on this observation, various crack detection methods have been proposed, such as the thresholding based methods [20], the edge detection based methods [14,21], and the minimal-path methods [22–24], delineation filtering based methods [25,26]. Among the various line structure detection methods, the minimal-path method and its extension path voting is an outstanding one which holds a stable performance in enhancing the line structures and has been applied to the detection many kinds of line structures other than pavement cracks, for example, the guide-wire segmentation from X-ray images [27], the road extraction from remote sensing images [28], etc. However, traditional path voting is usually dependent on the minimal-path search, which requires a set start point and end point for the path tracking. This requirement would make the minimal-path path voting less automatic. Meanwhile, the minimal-path searching would return the false path when the crack has sharp corners, as will be illustrated in Section 3.

In this study, we propose a novel path voting algorithm that yields the votes segmentate images. Based on the proposed path voting algorithm, we construct a two-level grouping approach for crack detection. It consists of a local grouping and a global grouping. In local grouping, potential crack curves are enhanced by the proposed path voting algorithm. This path voting adapts the standard normalized cut algorithm to partition an image patch along the potential crack path, which makes it fully automatic and avoids the limitation of the minimal-path path voting that requires the careful setting of tracking points. Through path voting on the whole image, the votes aggregated to form a crack probability map. Then, in the global grouping, potential crack seeds are sampled on the crack probability map and modeled into a graph. An attractive field-based algorithm is presented to calculate the weights for the edges in the graph, and the final crack curves are extracted using the minimum spanning tree and tree pruning algorithms. A flowchart of the proposed method is shown in Figure 2.

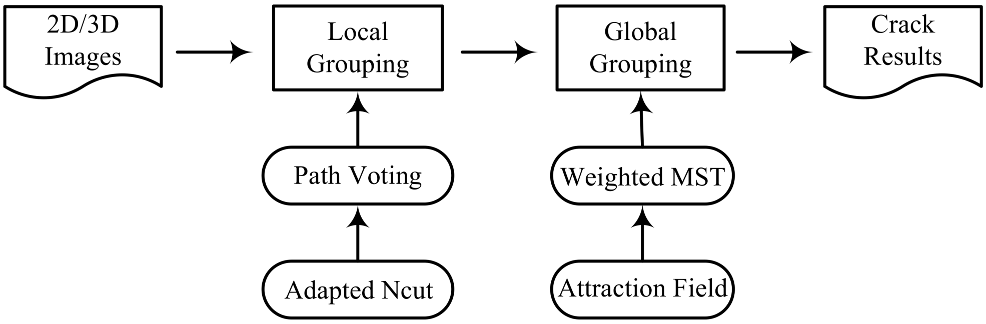


Figure 2. A flowchart of the proposed method.

The remainder of this paper is organized as follows. Section 2 introduces the related work. Section 3 presents the proposed crack detection method. Section 4 reports the experimental results on two optic image datasets and one depth image dataset, and Section 5 concludes our work.

2. Related Work

In the past two decades, crack detection has become a popular research topic, inspired by the urgent demands of industrial and infrastructure inspections. Crack detection is usually performed in two kinds of data, one is the 2D optic images, and the other is the 3D depth images. The former can be commonly found in the early research in the community, while the latter has gradually become a research focus since 2010 in an environment of rapid development of laser imaging technologies. In the following section, we briefly overview both of them.

2.1. Crack Detection Using 2D Images

In the early research of crack detection, the cracks are commonly captured using 2D optic images. An interesting study was performed in [29], where the characteristics of the cracks and their influence on the crack recognition were analyzed by using different illumination conditions and image acquisition distance. Under a normal illuminance, a crack is generally darker than the background and distributes into a linear or curvilinear structure. Thus, image thresholding is a straightforward way to detect cracks. In [22,30], the threshold value was figured out by examining the difference between the cracks and their neighboring non crack pixels. In [31], entropy is embedded into a two-level thresholding framework for pavement crack detection. In [20], the threshold value was calculated in a heuristic way and was used to extract the sealed cracks. However, pavement shadows and uneven illuminations would undermine the robustness of the thresholding methods [32]. As the crack is thin and displays an edge-like disturbance in intensity, many methods stemming from edge detection and wavelet transformation have been developed for crack detection. In [33], the Sobel edge detector was investigated for crack detection. In [14,21,34], wavelet transforms such as Gabor filters and anisotropic-based method were developed to examine the edge properties of the cracks. In [21], a 2D continuous wavelet transform is applied to create multiscale complex coefficient maps, on which the modulus and phase maps are constructed and a maxima location map is obtained for crack detection. In [14,35], the crack features were extracted by Gabor filters. In [34], cracks are examined by an anisotropic clustering method and were applied to extract surface cracks. However, the edge information would easily be tangled by sparkle noise. Minimal-path methods have also been studied. In [22], a seed growing method built on minimal-path searching was proposed for pavement crack detection. A similar method was presented in [24], where a set of minimal paths were selected by checking the intensities of the pixels in the image, and two postprocessing steps were introduced to guarantee the accuracy of the detected results. In [23], the minimal-path search was used to track cracks in the complex background in the 2D image, where the seed points for path tracking should be handily set in advance. As a self-adaptive strategy, machine learning-based methods were investigated for crack detection. In [36], a backpropagation neural network was employed to classify cracks based on the moment invariants features. In [37], deep convolutional neural networks were used to classify the image patches into crack blocks and non crack ones. In [38], the detection of bridge cracks was studied by using a modified active contour model and greedy search-based support vector machine. In [19], fully convolutional neural networks were studied to infer cracks of nuclear power plants using multi-view images. Many other methods were also proposed for crack detection, e.g., the saliency detection method [12], the structure analysis methods by using the minimal spanning tree [4] and the random structure forest [39], and the high efficient crack detection using GPUs [40].

2.2. Crack Detection Using 3D Images

With the development of sensing technology, especially the advancement of laser scanning sensors, accurate 3D measurement of the object's surface has become possible. As a result, several types of research have been conducted to use 3D depth images for crack detection. In [8], a mobile laser scanning system was developed to collect high-density point clouds of the pavement, and a framework named ITVCrack was introduced to handle the point clouds data. The road points are separated from the nonroad points, and then an iterative tensor voting algorithm was proposed to extract the cracks from the noisy background. The method was reported to be applicable for pavement cracks with low contrast, low signal-to-noise ratio, and bad continuity. In [9], the performance of crack segmentation was improved by an enhanced dynamic optimization algorithm using 3D laser imaging data. In this method, the data were preprocessed using a two-step Gaussian filter to obtain a smooth depth image avoiding the influence of cross-slope and ruts, and then a dynamic optimization algorithm was proposed to extract the final cracks from the crack candidates obtained by rough segmentation. In [41], a hybrid procedure was proposed for pavement crack detection, where the 3D pavement data was collected by 3D Ultra. The precision of the range value is 1 mm. In addition, the system developed is also equipped with optical cameras, which makes it possible to fuse the depth image with the intensity image. In crack detection, algorithms based on tensor voting and minimum spanning trees were employed. In [15], the 2D optic image and 3D depth images were also combined to detect the surface cracks, in which a 3D camera captures the laser line and calculates the 3D profile of the pavement every 10 mm along the road. The AdaBoost algorithm [42,43] is equipped with two strong classifiers where the first strong classifier composed of 56 Gabor filters is used for transverse crack detection, and the other strong classifier composed of 9 filters is used for longitudinal crack detection. The method was reported to obtain an accuracy of over 90%. In [44], crack classification was performed based on the 3D laser scanning data. In this method, the structure of the pavement surface was represented by the 3D depth image at an accuracy of 1 mm, and the crack segments extracted were modeled with bounding box-based technologies and used for classification and defect severity evaluation.

3. Crack Detection Method

In a 2D optic image or a 3D depth image, a crack is a linear/curvilinear structure that is represented by a spatially continuous gradient and intensity. To extract cracks from the images, we first propose a novel path voting local grouping algorithm to enhance the potential crack curves and then present a global grouping algorithm to extract the final cracks.

3.1. Local Grouping

Over the past two decades, path voting as an image processing tool has been used in several applications, e.g., map matching in GPS tracks [45], guidewire in fluoroscopy image [27], and road detection in remote sensing images [28], etc. In our work, a new crack detection method is proposed based on the path voting algorithm.

In curve enhancement/extraction research, minimal-path algorithms have been widely used [46–49]. However, the minimal-path algorithm is limited, as it requires a start point and an end point in advance. One possible way to loosen the constraint is using a seed growing strategy [22,23,50], where only one seed point is required for searching one minimal path. In particular, the minimal path algorithm can be performed in a voting way for curve structure enhancement [27,28]. In path voting, the number of times that a pixel is passed by all minimal paths is taken as the probability (normalized to [0 1]) that the pixel belongs to a curve. In this work, we perform path voting locally to produce a crack probability map.

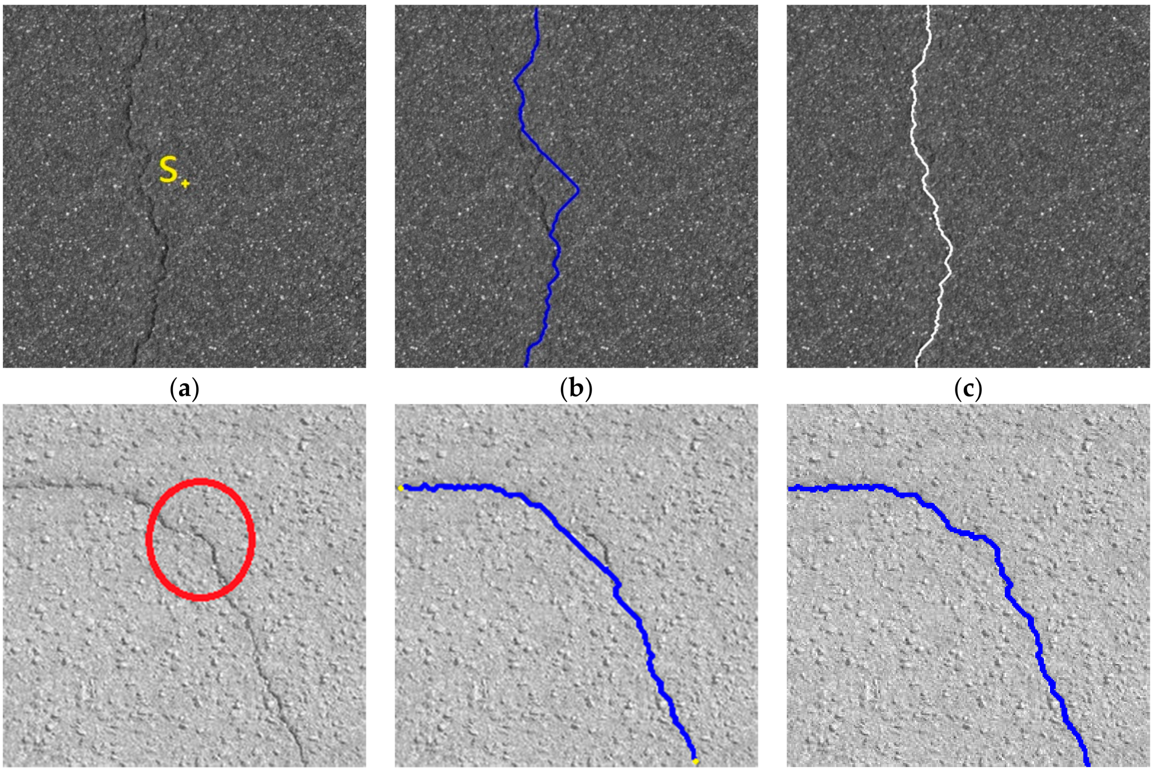
Let I be a gray image, q be a pixel in I, r be a searching radius, and I_q^r be a $(2r+1) \times (2r+1)$ sub-image centered at q. Then, the path locally found at point q w.r.t.r can be described as C_q,

$$C_q = \Psi\left(I_q^r\right) \tag{1}$$

where $\Psi()$ denotes the ways to find a path in the sub-image I_q^r. One popular way is a minimal-path seed growing, as introduced by Li and Zou [22], where the center of a sub-image is taken as a seed point for the seed growing algorithm. However, it would incur problems when the potential crack curve does not go across the center of the sub-image. Figure 3 gives an example to illustrate the limitation of the seed growing strategy. Figure 3a is a sub-image with a crack crossing it. Figure 3b shows the seed growing result, in which the center of the sub-image is the seed point. We can see that the seed growing path does not fully cover the target curve, as the seed points are not guaranteed to locate on the crack curve. Even when the start point and the end point are given for crack tracking, the minimal-path algorithm may have errors in tracking the true crack. For example, the result produced by the minimal-path tracking with the start and end points is shown in Figure 3e, where a tracking error occurs at the sharp section as having been annotated by the red region in Figure 3d. This is because the minimal-path algorithm is not robust in handling the grid graph [23]. In this work, we get the paths involved in voting by image segmentation strategy other than the seed growing, as described by Equation (2),

$$C_q = Ncut\left(I_q^r\right) \tag{2}$$

where $Ncut()$ denotes a popular image segmentation method, normalized cut [51]. In the following, we will show how to adapt it to meet requirements for the local grouping of crack points.



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 3. Illustration of the weakness of the minimal-path algorithm inline structure extraction in a local subitem age. In the top row, it illustrates the weakness of minimal-path seed growing, where (**a**) is an input image, (**b**) is the seed growing results based on a seed S located at the image center, and (**c**) is the proposed adaptive *NCut* result. In the bottom row, it illustrates the weakness of minimal-path tracking, where (**d**) is another input image, (**e**) is the minimal-path tracking results based on two end points marked in yellow, and (**f**) is the proposed adaptive *NCut* result. The crack in the red circle, which has a sharp curve, cannot be tracked in the minimal-path tracking.

Let $G = (V, E)$ be a graph, which can be partitioned into two disjoint sets, A and B, $A \cup B = V$, $A \cap B = \phi$, by removing edges connecting the two parts. Let $w(u, v)$ be the weight between nodes u and v, then the similarity of the two parts can be measured by the total weight of the edges that have been removed, which is defined as the graph-cut cost:

$$Cut(A,B) = \sum_{u \in A, v \in B} w[u,v] \tag{3}$$

In normalized cut (*Ncut* [51]), the cut cost is computed as a fraction of the total edge connections to all the nodes in the graph:

$$N\ cut(A,B) = \frac{cut(A,B)}{assos(A,V)} + \frac{cut(A,B)}{assos(B,V)} \tag{4}$$

where $assos(A,V) = \sum_{u \in A, t \in V} w(u,t)$ is the total connection from nodes in A to all nodes in the graph and $assos(B,V)$ is the total connection from nodes in B to all nodes in the graph. According to *Ncut* [51], the cut cost in Equation (4) can be minimized by an efficient eigenvalue-based technique. In the context of crack detection, we want to achieve a minimum cut along the crack curve. To achieve this goal, we adapt the weight function. Specifically, we assign lower weights to edges involved in the nodes on the crack curve, and higher weights to other edges. As a crack has a certain width, the weight should also be formulated to reach a local maximum at the center of the crack. Based on the discussion above, the weight between pixel i and j is defined by

$$w_{ij} = e^{\frac{-(Y_i+Y_o)^2}{\sigma^2}} \times \begin{cases} 1\ if \parallel X_i - X_j \parallel < r \\ 0 \text{ otherwise} \end{cases} \tag{5}$$

where

$$Y_o = \arg max\{(Y_i + Y_k)|\|X_i - X_k\| < r\} \tag{6}$$

X_i and Y_i denote the location and intensity of pixel i, respectively. σ tunes w_{ij} to a reasonable range and makes it discriminative to one another. As in the voting process, a segmentation curve will be generated in each patch. However, for the patches that do not contain any cracks, it is not desired to be predicted with any crack curve. In other words, the segmentation path generated in a non-crack patch should give less vote to the crack probability map. Thus, in our solution, we give an additional weight to relieve this concern. Specifically, we set it as a two-class segmentation problem as we suppose there is only one crack line in a small patch, for example, a 23 × 23 patch cropped from an original large image. Then, two eigenvalues will be calculated by the *NCut* solution. The two eigenvalues indicate the confidence of the two parts to be segmented. When there is a crack curve in the patch and the segmentation path goes on it, the two eigenvalues will be very close as the crack curve gives confidence to both parts in the segmentation, otherwise, the second eigenvalue will be much smaller than the first eigenvalue. On this point, we propose to highlight the crack curve, which is likely to be the segmentation path obtained by the adapted *NCut*, by placing additional weight on the segmentation path. Let I_q^r be a (2r + 1) × (2r + 1) subimgage centered at q, eig_1 and eig_2 be the two eigenvalues generated by the adapted *NCut*, and C_q be the segmentation path on I_q^r, then we give the segmentation path C_q a weight defined by Equation (7),

$$Z(C_q) = e^{-c\times(eig_1 - eig_2)} \tag{7}$$

where c is a constant to increase the contrast between the crack path and non-crack path, which further highlights the cracks. We empirically set c as 10. When a sub-image is cropped at a pixel and the adapted *Ncut* is applied to it, the segmentation path can generally cover a set of pixels in the corresponding sub-image. In our approach, path voting is achieved by aggregating the segmentation path generated by the adapted *Ncut* on a set of sampled pixels. Note that each pixel will be crossed through by multiple paths. Let N be the maximum number of crossing paths over all pixels; with N as a normalization base, then a crack probability at pixel i can be computed by

$$\mathcal{P}_i = \frac{1}{N} \sum_{I_q^r \in \Phi(I,r,s)} \xi^{(+,-)}(X_i, C_q) \tag{8}$$

where

$$\xi^+(X_i, C_q) = \{Z(C_q)|X_i \in C_q\} \tag{9}$$

and

$$\xi^-(X_i, C_q) = \{0|X_i \notin C_q\} \tag{10}$$

The X_i is the location of pixel i, $\Phi(I, r, s)$ denotes all the sub-images sampled on the original image I, with a searching radius r, and a marching step s. C_q is the path found on sub-image I_q^r by using the proposed adapted *NCut* algorithm.

Figure 4 shows the results of different voting algorithms on an image. In the section of experiments, we will examine the influence of the two parameters on the crack detection performance. Figure 5 illustrates the robustness of the proposed path voting algorithm on enhancing the crack curves under different widths and different noise rates. To facilitate the comparison, simulated cracks of different widths are used and are added noise at a different rate. It can be seen from Figure 5 that the proposed path voting holds a stable performance in enhancing the crack curves of different widths and different noise rates.

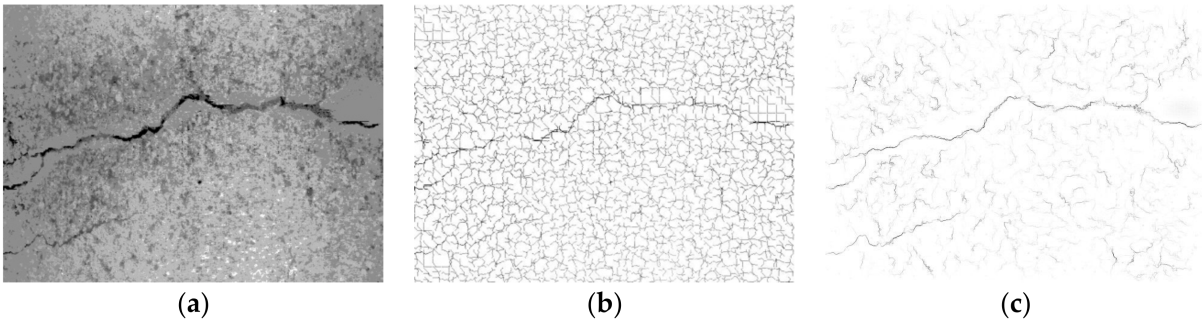

(**a**) (**b**) (**c**)

Figure 4. Results generated by different path voting algorithms. (**a**) An original image. (**b**) The result was produced by the minimal-path path voting. (**c**) The result was produced by the proposed path voting.

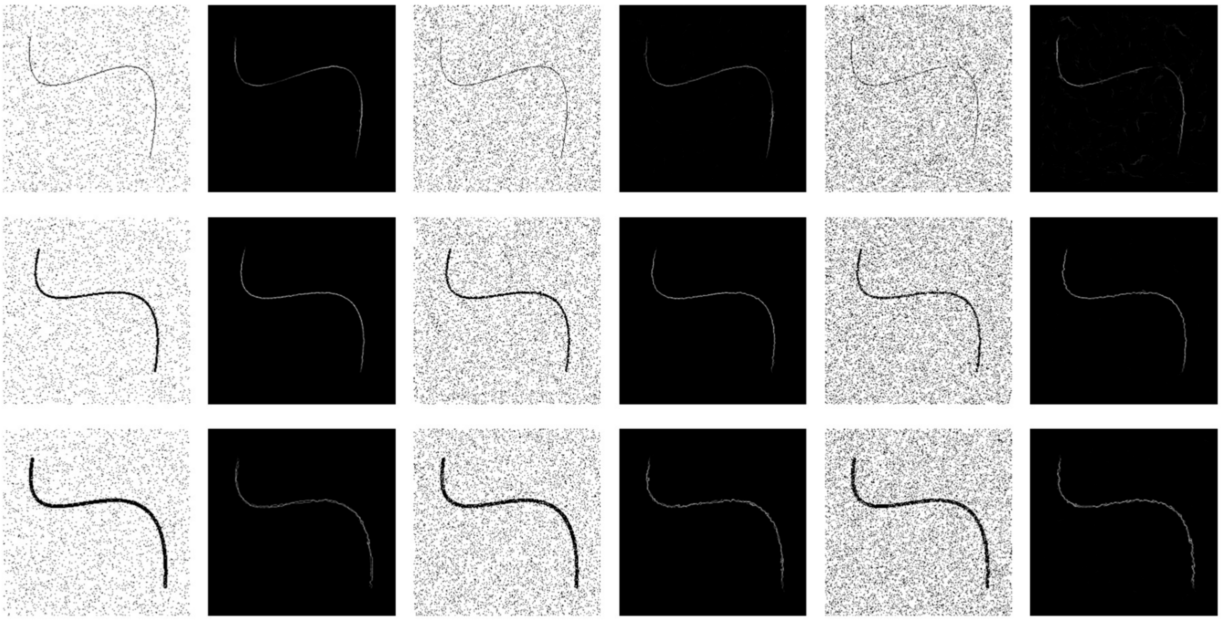


Figure 5. Results obtained by the proposed path voting method on simulated cracks of different widths and different noise rates. The cracks in the top row, middle row, and bottom row have the width of 1 pixel, 3 pixels, and 5 pixels, respectively. The simulated crack images have been added the 'pepper and salt' noise at an intensity of 0.1, 0.2, and 0.3 in the first, third, and fifth column, respectively.

3.2. Global Grouping

Taking the local grouping result as a crack probability map, we implement the crack seed sampling with a maximal value validation algorithm [4]. Based on the crack seeds, we construct an undirected graph G to model the possible connections among these crack seeds. Let each crack seed be a vertex v_i, and each pair of vertices $\left(v_i, v_j\right)$ be assigned with an edge $e_{i,j}$, then the graph can be represented by

$$G = (V, E) \tag{11}$$

where $V = \{v_i | i = 1, 2, \ldots N\}$ is the vertex set, and $E = \left\{e_{i,j} \middle| e_{ij} = \left(v_i, v_j\right)\right\}$ is the edge set. There are two major rules in assigning the weight to the edges.

First, for two vertices, the larger distance is between them, the lower weight is set to the corresponding edge. This is because a larger distance will decrease the continuity, and hence makes them less groupable.

Second, for two vertices, the larger values they have in the crack probability map, the higher weight it would set to the corresponding edge. This is because a pixel with a larger crack probability value will bring more influence to the surrounding area, and two vertices with larger crack probability values are likely to group together.

The above observations may easily remind us of the gravity field model. Similarly, we defined the weight of an edge $e_{i,j}$ in the graph by Equation (12), and we name it as attraction field,

$$w_{i,j} = \frac{I_i \cdot I_j}{D_{i,j}} \tag{12}$$

We find the minimum spanning tree (MST) from the constructed graph G to identify the desired edge connections among crack seeds. As MST is a spanning tree with the minimum total edge weight, the edges remaining in an MST connect the crack seeds with the best proximity. In addition, a recursive edge pruning algorithm [4] is applied to find long branches, and remove short branches with poor proximity and continuity in the resulting MSTs.

4. Experiments and Results

In this section, we will validate the effectiveness of the proposed method by evaluating and comparing the crack detection performances on several datasets. Specifically, we will first introduce the metrics for performance evaluation, and then present two 2D optic image datasets and one 3D depth image dataset used in the experiments, and give the experiment results.

4.1. Metrics

For each image, *Precision* and *Recall* were computed by comparing the detected cracks against the human-annotated ground truth. These two metrics are defined by

$$Precision = \frac{Ture\ Positives}{Ture\ Positives + False\ Positives} \tag{13}$$

$$Recall = \frac{Ture\ Positives}{Ture\ Positives + False\ Negatives} \tag{14}$$

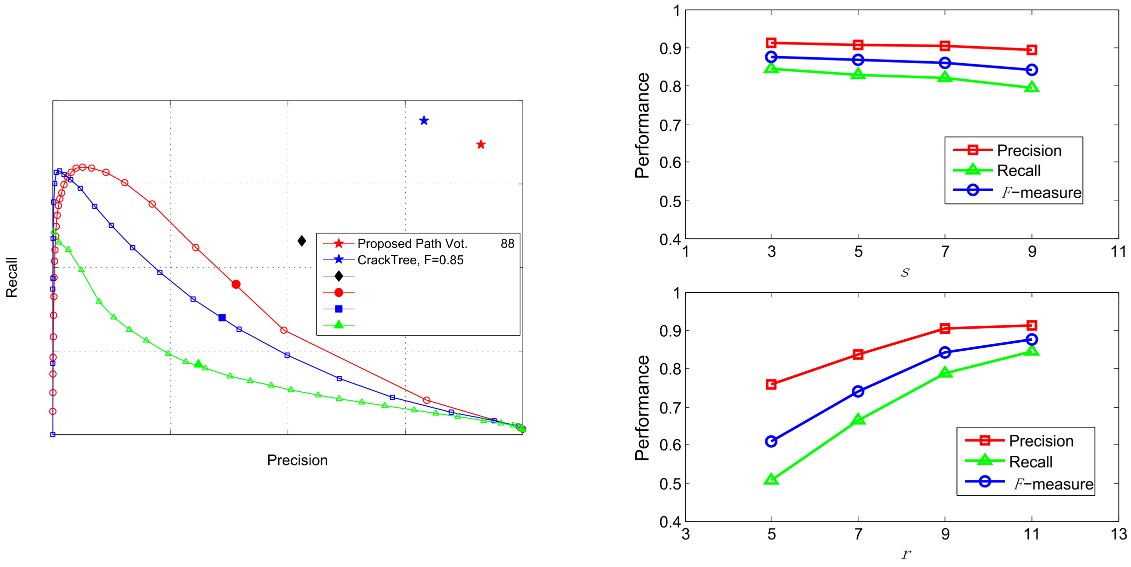
and an overall performance metric *F measure* is defined by

$$F - measure = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{15}$$

Considering that cracks have a certain width, a detected crack pixel is still taken as a true positive if it is no more than 2 pixels away from human-annotated crack curves.

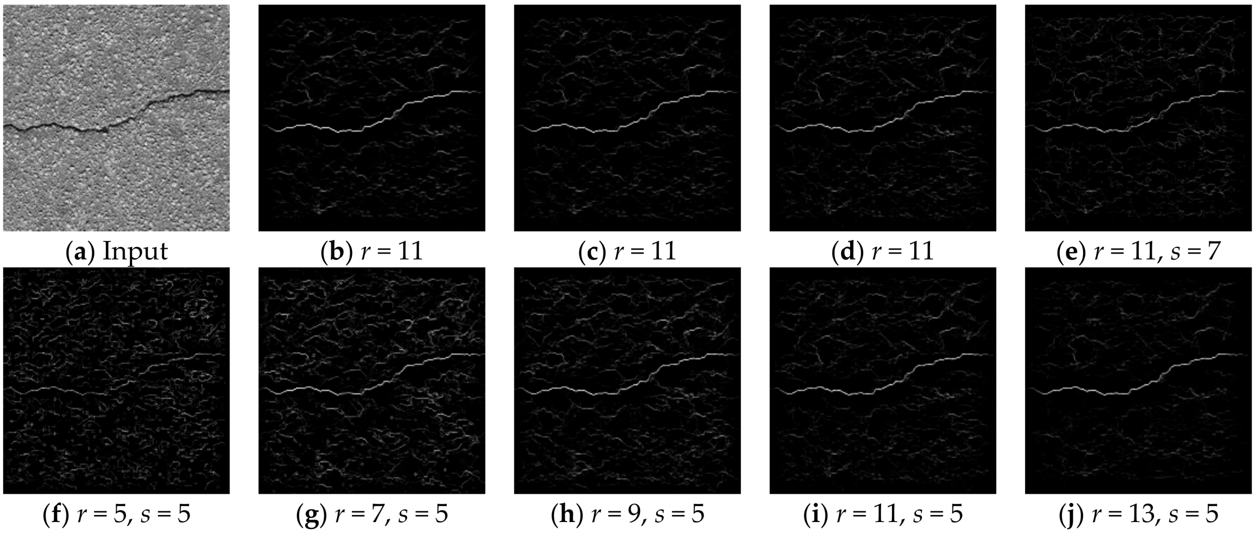
4.2. Results on CrackTree206 Dataset

The CrackTree206 dataset 1 contains 206 2D optic pavement images captured by using a CCD camera. All images share the same size of 800 × 600. The cracks in the images have been annotated using an interactive crack detection tool. About one-fourth of the images contain shadows that are cast by trees or light poles, which brings additional challenges to the crack detection problem. Several crack detection methods have been introduced for comparison, such as the segmentation extension method (Seg-ext) [52], the global Pb series [53], and the CrackTree [4]. The precision–recall curves of these methods have been shown in Figure 6a. It can be seen from Figure 6a that the proposed path voting method achieves an F-measure value of 0.88, which is the highest among all comparison methods. Figure 6b shows the results of the proposed path voting algorithm by using different radius r and different step length s. In the top figure of Figure 6b, the radius is fixed to r = 11, and the step length s is tuned from 3 to 9 at an interval of 2 pixels. It can be observed that the increase in step length s from 3 to 9 will slightly decrease the performance of the proposed method. In the bottom figure of Figure 6b, the step length s is fixed to 5, and the radius r is tuned from 5 to 11 at an interval of 2 pixels. It can be observed that the increase in radius valuer will dramatically decrease the performance of the proposed method. Figure 7 shows the results under different r and s values.



(**a**) (**b**)

Figure 6. Crack detection results on CrackTree206 dataset. (**a**) The precision–recall curves for the comparison methods. (**b**) The performance of the proposed path voting method running at different step s and different radius r.



(**a**) Input (**b**) $r = 11$ (**c**) $r = 11$ (**d**) $r = 11$ (**e**) $r = 11$, $s = 7$

(**f**) $r = 5$, $s = 5$ (**g**) $r = 7$, $s = 5$ (**h**) $r = 9$, $s = 5$ (**i**) $r = 11$, $s = 5$ (**j**) $r = 13$, $s = 5$

Figure 7. Path voting results by using different radius r and different step length s. In the top row, the radius is fixed to be $r = 11$, and the step length s is varied from 1 to 7 at an interval of 2. In the bottom row, the step length is fixed to be $s = 5$, and the radius r is varied from 5 to 13 at an interval of 2.

4.3. Results on StoneCrack50 Dataset

The StoneCrack50 dataset is a material crack dataset. In many stone material factories, stones have to be reformulated or cut into specific shapes. During the process, cracks may appear on the surface of the stone. We collect a set of 50 2D optic stone images. Each image has a size of 628 × 516. The cracks in the image are also annotated with human interactions. We compare the performance of the proposed method with the CrackTree method and the minimal-path path voting method on the StoneCrack50 dataset. The results have been given in Table 1. For the two path voting-based methods, they have the same parameter settings, that is, step length s = 5 and radius r = 11.

Table 1. Crack Detection on StoneCrack50 dataset.

Method	CrackTree	Minimal-Path Path Voting	The Proposed Path Voting
Precison	0.8725	0.8218	0.9182
Recall	0.5562	0.5236	0.5697
FMeasure	0.6793	0.6397	0.7031

We also examine the influence of the step length s and the radius r on the performance of the proposed path voting method. It can be seen from Table 2 that a larger radius r will decrease the performance of the proposed method. This is because the segmentation performed in too small a patch will easily bring poor results. It can also be observed from Table 2 that a larger marching step length s generally leads to a lower performance for the proposed method. This is because too large a step length will lead to inadequate voting for the discriminative curves.

Table 2. Performance of the proposed path voting method on StoneCrack50 at different radius r and different step length *s*.

Location		$s = 5$	$s = 7$	$s = 9$	$s = 11$
Precison		0.9522	0.9521	0.9490	0.9823
Recall	$r = 5$	0.0645	0.0663	0.0635	0.0607
FMeasure		0.1208	0.1240	0.1190	0.1144
Precison		0.9581	0.9668	0.9424	0.9112
Recall	$r = 7$	0.1782	0.1382	0.1026	0.0689
FMeasure		0.3005	0.2418	0.1851	0.1281
Precison		0.9247	0.9116	0.8751	0.9064
Recall	$r = 9$	0.4274	0.3517	0.2328	0.1517
FMeasure		0.5846	0.5076	0.3678	0.2599
Precison		0.9182	0.9284	0.8735	0.8612
Recall	$r = 11$	0.5696	0.4908	0.4403	0.2103
FMeasure		0.7031	0.6421	0.5855	0.3381

4.4. Results on CrackPV14 Dataset

The CrackPV14 dataset is collected using a 3D laser imaging system. The 3D laser imaging system consists of a line structure laser and a 3D camera mounted onto the rear of a vehicle, as illustrated in Figure 8. The laser projects a laser beam onto the pavement. The projected beam will be curved under a bumpy pavement surface. The 3D camera will recognize the beam curve in real-time and transform the beam curve into a number of 3D points by using the Laser Triangulation techniques [54]. When the vehicle moves along the road, the pavement surface will be measured with dense 3D points. In our system, the collected 3D points have the following attributes: (i) the resolution in *X*-axis (perpendicular to *Y*-axis on the road surface plan) is 1.80 mm; (ii) the resolution in *Y*-axis (along the road) is 1.00 mm; (iii) the resolution in *Z*-axis (perpendicular to the road surface plan) is 0.25 mm. Originally, the Z values of 3D points represent the relative range from the laser device to the pavement surface in the vertical direction. For convenience, we negate the range values and normalize them to [0 255]. Thus we can represent them with a gray image, namely the range image or the 3D depth image. A set of 14 laser range images are collected to validate the proposed approach. The ground truth crack curves were manually annotated on these images for objective performance evaluation.

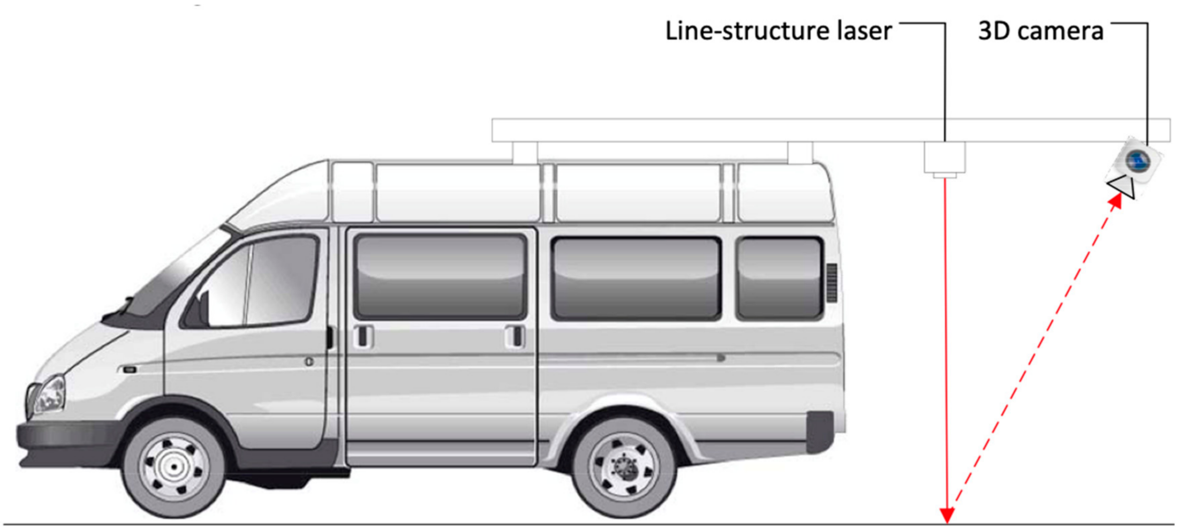


Figure 8. An illustration of the laser imaging system.

For comparison, the CrackTree [4] and FoSA [22] were tested on the same data, where the parameters were uniformly set as σ = 11, L_e = 10, and L_p = 100 for CrackTree, and seed growing radius = 28 for FoSA. For the proposed approach, in the local grouping step, the searching radius for path voting was set as r = 11, and the step length for sub-image sampling was set as s = 5. While in the global grouping step, parameters were set as same as that of the CrackTree.

Figure 9 displays the results of five images in the dataset. From Figure 9, we can see the crack maps (in row 3) produced by the proposed path voting algorithm have well grouped the potential crack points into salient crack curves. Compared with the crack map (in row 2) obtained from tensor voting, the path voting-based crack map shows more concentrated energy on the crack center. Note that, in tensor voting, a thresholding step is required before calculating the crack map. As improper thresholding would result in false crack seeds, the crack map produced by tensor voting is largely dependent on the thresholding quality. Unlike tensor voting, the proposed path voting algorithm does not require a thresholding step in crack map computation. Therefore, the proposed path voting algorithm is more reliable than tensor voting in crack map construction. Comparing results in rows 4, 5, and 6 in Figure 9, we can find that the proposed approach produced better crack results than CrackTree and FoSA.

Table 3. Crack detection performance on 14 laser range images (Prop: proposed approach, CrTr: CrackTree).

Method	Prop	CrTr	FoSA	Prop	CrTr	FoSA	Prop	CrTr	FoSA	Prop	CrTr	FoSA
		img. #1			img. #2			img. #3			img. #4	
Precison	0.872	0.821	0.845	0.842	0.625	0.733	0.846	0.885	0.897	0.793	0.753	0.756
Recall	0.965	0.691	0.628	0.904	0.605	0.568	0.905	0.713	0.612	0.903	0.776	0.691
FMeasure	0.916	0.751	0.721	0.872	0.614	0.640	0.874	0.790	0.728	0.845	0.764	0.722
		img. #5			img. #6			img. #7			img. #8	
Precison	0.949	0.845	0.860	0.671	0.780	0.836	0.960	0.698	0.716	0.846	0.696	0.749
Recall	0.939	0.600	0.557	0.843	0.649	0.647	0.915	0.605	0.552	0.929	0.668	0.654
FMeasure	0.994	0.700	0.691	0.747	0.708	0.729	0.937	0.648	0.623	0.886	0.682	0.698
		img. #9			img. #10			img. #11			img. #12	
Precison	0.767	0.722	0.779	0.833	0.927	0.811	0.833	0.839	0.792	0.997	0.847	0.868
Recall	0.996	0.669	0.636	0.961	0.860	0.805	0.993	0.967	0.937	0.823	0.923	0.805
FMeasure	0.867	0.695	0.700	0.892	0.892	0.808	0.906	0.898	0.858	0.893	0.883	0.835
		img. #13			img. #14			Average				
Precison	0.449	0.775	0.696	0.848	0.948	0.925						
Recall	0.890	0.706	0.663	0.988	0.985	0.880						
FMeasure	0.639	0.739	0.679	0.931	0.966	0.901	0.867	0.766	0.738			

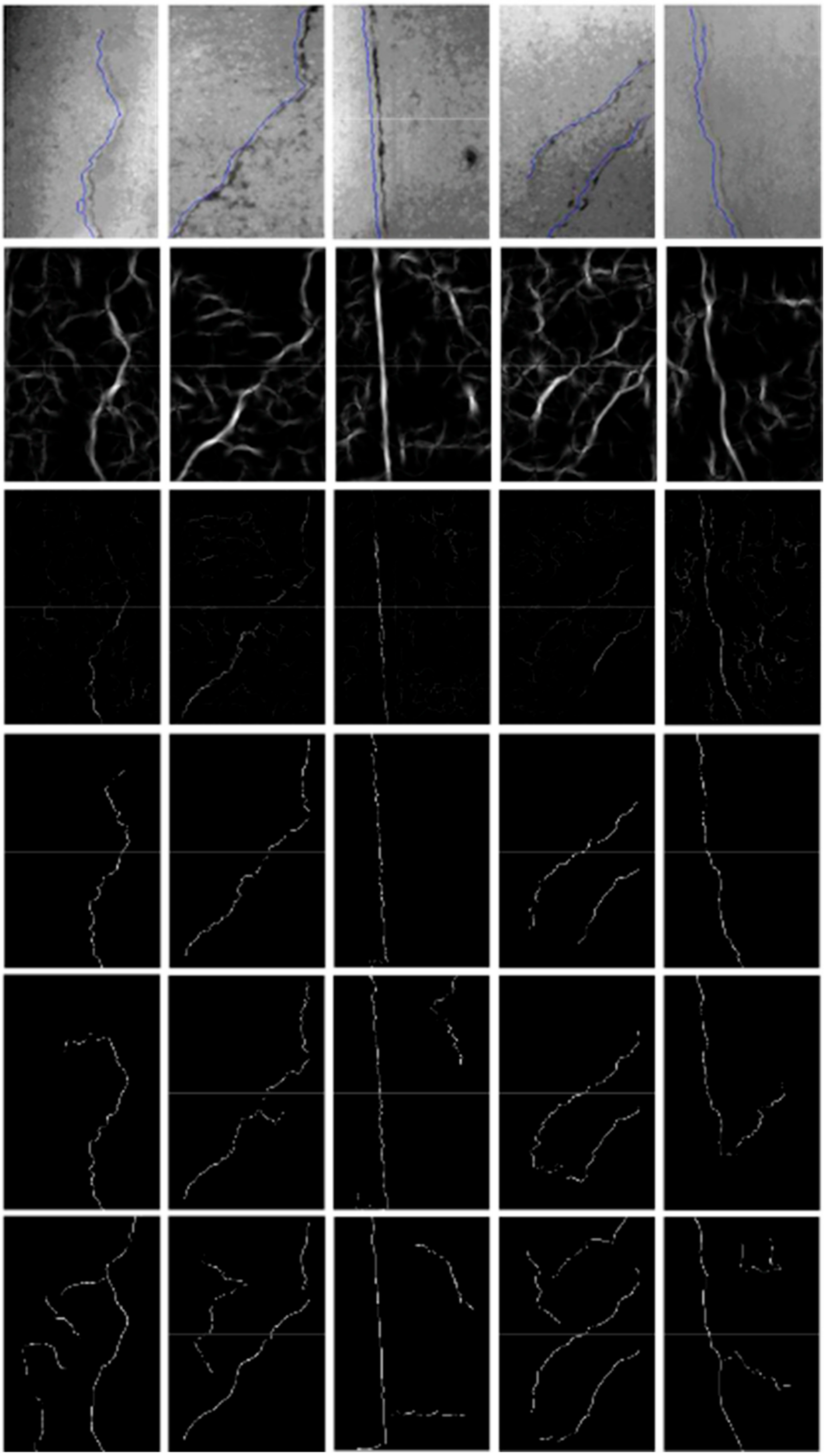


Figure 9. Experimental results. Row 1: five pavement images with ground truth cracks (in blue) shifted, corresponding to images No. 4, 2, 5, 6, and 11 in Table 3. Row 2: crack maps from tensor voting (in CrackTree). Row 3: crack maps from the proposed path voting. Row 4: results from the proposed approach. Row 5: CrackTree results. Row 6: FoSA results.

In our experiment, we evaluated and compared the running efficiency. CrTr, FoSA, and our method took an average of 2 s, 2.5 s, and 1.2 s to process an image, respectively. The hardware platform is configured with PC Operating System Windows 10, Intel CPU 2.6 G, and 32 G of DDR4 RAM.

Table 3 lists the results on fourteen laser range images. It can be seen from Table 3 that most of results from the proposed approach are better than that from CrackTree and FoSA. To make an overall comparison, the average Fmeasures over all the testing images were calculated. The proposed approach gained an average F measure much higher than that of CrackTree and FoSA, which demonstrated the advancement of the proposed approach.

5. Conclusions and Future Directions

In this paper, we have proposed a novel path voting algorithm that performed voting segment images. Based on this path voting, a two-level grouping framework has been present for detecting cracks from 2D optic images and 3D depth images. First, in the local grouping, the proposed path voting was applied to enhance the potential crack strings and produce a crack probability map. Then, in the global grouping, edge linking and pruning algorithms were performed to extract the desired cracks. In the experiments, two optic image datasets and one depth image dataset have been used for performance evaluation. The experimental results demonstrated that the proposed segmentation-based path voting could effectively enhance the crack curves at the different background and noise rates, and the proposed crack detection method outperformed several competing methods in crack detection from both optic images and depth images.

In our future work, we will study two aspects. One is to use more advanced sensors for pavement modeling, and the other is to use more advanced machine learning methods such as deep convolution neural networks for crack detection in [55–57].

Author Contributions: Conceptualization, J.Y.; Data curation, Z.C.; Formal analysis, Z.X.; Funding acquisition, J.Y.; Investigation, Z.C.; Methodology, J.Y.; Project administration, J.Y.; Resources, J.Y.; Software, Z.C. and Z.X.; Supervision, J.Y.; Validation, Z.C.; Writing—original draft, J.Y.; Writing—review and editing, J.Y., Z.C. and Z.X. All authors have read and agreed to the published version of the manuscript.

Funding: The research and publication of this article were funded by the open fund of key laboratory of urban land resources monitoring and simulation, ministry of natural resources (KF201904050).

Data Availability Statement: The raw data required to reproduce these findings cannot be shared at this time as the data also form part of an ongoing study.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Koch, C.; Georgieva, K.; Kasireddy, V.; Akinci, B.; Fieguth, P. A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Adv. Eng. Inform.* **2015**, *29*, 196–210. [CrossRef]
2. Qu, Z.; Bai, L.; An, S.Q.; Ju, F.R.; Liu, L. Lining seam elimination algorithm and surface crack detection in concrete tunnel lining. *J. Electron. Imaging* **2016**, *25*, 063004. [CrossRef]
3. Huang, Y.; Xu, B. Automatic inspection of pavement cracking distress. *J. Electron. Imaging* **2006**, *15*, 013017. [CrossRef]
4. Zou, Q.; Cao, Y.; Li, Q.; Mao, Q.; Wang, S. CrackTree: Automatic crack detection from pavement images. *Pattern Recognit. Lett.* **2012**, *33*, 227–238. [CrossRef]
5. Oliveira, H.; Correia, P. Automatic road crack detection and characterization. *IEEE Trans. Intell. Transp. Syst.* **2013**, *14*, 155–168. [CrossRef]
6. Aldea, E.; Le Hégarat-Mascle, S. Robust crack detection for unmanned aerial vehicles inspection in an a-contrario decision framework. *J. Electron. Imaging* **2015**, *24*, 061119. [CrossRef]
7. Zhang, D.; Li, Q.; Chen, Y.; Cao, M.; He, L.; Zhang, B. An efficient and reliable coarse-to-fine approach for asphalt pavement crack detection. *Image Vis. Comput.* **2017**, *57*, 130–146. [CrossRef]
8. Guan, H.; Li, J.; Yu, Y.; Chapman, M.; Wang, H.; Wang, C.; Zhai, R. Iterative tensor voting for pavement crack extraction using mobile laser scanning data. *IEEE Trans. Geosci. Remote Sens.* **2015**, *53*, 1527–1537. [CrossRef]
9. Jiang, C.; Tsai, Y.J. Enhanced crack segmentation algorithm using 3d pavement data. *J. Comput. Civ. Eng.* **2015**, *30*, 04015050. [CrossRef]

10. Zou, Q.; Li, Q.; Zhang, F.; Xiong, Z.; Wang, Q. Path voting based pavement crack detection from laser range images. In Proceedings of the 2016 IEEE International Conference on Digital Signal Processing, Beijing, China, 16–18 October 2016; pp. 211–215.
11. Nguyen, T.S.; Begot, F.; Duculty, F.; Avila, M. Freeform anisotropy: A new method for crack detection on pavement surface images. In Proceedings of the 2011 18th IEEE International Conference on Image Processing (ICIP), Brussels, Belgium, 11–14 September 2011.
12. Xu, W.; Tang, Z.; Zhou, J.; Ding, J. Pavement crack detection based on saliency and statistical features. In Proceedings of the IEEE International Conference on Image Processing, Melbourne, Australia, 15–18 September 2013; pp. 4093–4097.
13. Mathavan, S.; Rahman, M.; Kamal, K. Use of a self-organizing map for crack detection in highly textured pavement images. *J. Infrastruct. Syst.* **2015**, *21*, 04014052. [CrossRef]
14. Zalama, E.; Gómez-García-Bermejo, J.; Medina, R.; Llamas, J. Road crack detection using visual features extracted by gabor filters. *Comput. Aided Civ. Infrastruct. Eng.* **2014**, *29*, 342–358. [CrossRef]
15. Medina, R.; Llamas, J.; Zalama, E.; Gomez-Garcia-Bermejo, J.; Medina, R.; Llamas, J. Enhanced automatic detection of road surface cracks by combining 2d/3d image processing techniques. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Paris, France, 27–30 October 2014; pp. 778–782.
16. Jahanshahi, M.; Jazizadeh, F.; Masri, S.; Becerik-Gerber, B. Unsupervised approach for autonomous pavement-defect detection and quantification using an inexpensive depth sensor. *J. Comput. Civ. Eng.* **2013**, *27*, 743–754. [CrossRef]
17. Sun, X.; Huang, J.; Liu, W.; Xu, M. Pavement crack characteristic detection based on sparse representation. *EURASIP J. Adv. Signal Process.* **2012**, *2012*, 191. [CrossRef]
18. Ouyang, W.; Xu, B. Pavement cracking measurements using 3d laser scan images. *Meas. Sci. Technol.* **2013**, *24*, 113–119. [CrossRef]
19. Schmugge, S.J.; Rice, L.; Lindberg, J.; Grizziy, R.; Joffey, C.; Shin, M.C. Crack segmentation by leveraging multiple frames of varying illumination. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Santa Rosa, CA, USA, 24–31 March 2017; pp. 1045–1053.
20. Kamaliardakani, M.; Sun, L.; Ardakani, M.K. Sealed-crack detection algorithm using heuristic thresholding approach. *J. Comput. Civ. Eng.* **2016**, *30*, 04014110. [CrossRef]
21. Subirats, P.; Dumoulin, J.; Legeay, V.; Barba, D. Automation of pavement surface crack detection using the continuous wavelet transform. In Proceedings of the IEEE International Conference on Image Processing, Atlanta, GA, USA, 8–11 October 2006; pp. 3037–3040.
22. Li, Q.; Zou, Q.; Zhang, D.; Mao, Q. FoSA: F* seed-growing approach for crack-line detection from pavement images. *Image Vis. Comput.* **2011**, *29*, 861–872. [CrossRef]
23. Kaul, V.; Yezzi, A.; Tsai, Y. Detecting curves with unknown endpoints and arbitrary topology using minimal paths. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 1952–1965. [CrossRef]
24. Amhaz, R.; Chambon, S.; Idier, J.; Baltazart, V. Automatic crack detection on two-dimensional pavement images: An algorithm based on minimal path selection. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2718–2729. [CrossRef]
25. Strisciuglio, N.; Azzopardi, G.; Petkov, N. Robust inhibition augmented operator for delineation of curvilinear structures. *IEEE Trans. Image Process.* **2019**, *28*, 5852–5866. [CrossRef]
26. Strisciuglio, N.; Azzopardi, G.; Petkov, N. Detection of curved lines with b-cosfire filters: A case study on crack delineation. In Proceedings of the International Conference on Computer Analysis of Images and Patterns, Ystad, Sweden, 22–24 August 2017; pp. 108–120.
27. Bismuth, V.; Vaillant, R.; Talbot, H.; Najman, L. Curvilinear structure enhancement with the polygonal path image–application to guidewire segmentation in X-ray fluoroscopy. In Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Nice, France, 1–5 October 2012; pp. 9–16.
28. Carlotto, M. Enhancement of low-contrast curvilinear features in imagery. *IEEE Trans. Image Process.* **2007**, *16*, 221–228. [CrossRef]
29. Cho, H.W.; Yoon, H.J.; Yoon, J.C. Analysis of crack image recognition characteristics in concrete structures depending on the illumination and image acquisition distance through outdoor experiments. *Sensors* **2016**, *16*, 1646. [CrossRef]
30. Li, Q.; Liu, X. Novel approach to pavement image segmentation based on neighboring difference histogram method. In Proceedings of the International Congress on Image and Signal Processing, Sanya, China, 27–30 May 2008; pp. 792–796.
31. Oliveira, H.; Correia, P. Automatic road crack segmentation using entropy and image dynamic thresholding. In Proceedings of the European Signal Processing Conference (EUSIPCO'09), Glasgow, UK, 24–28 August 2009; pp. 622–626.
32. Zou, Q.; Hu, Z.; Chen, L.; Wang, Q.; Li, Q. Geodesic-based pavement shadow removal revisited. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Shanghai, China, 20–25 March 2016; pp. 1761–1765.
33. Ayenu-Prah, A.; Attoh-Okine, N. Evaluating pavement cracks with bidimensional empirical mode decomposition. *EURASIP J. Adv. Signal Process.* **2008**, *2008*, 861701. [CrossRef]
34. Zhao, G.; Wang, T.; Ye, J. Anisotropic clustering on surfaces for crack extraction. *Mach. Vis. Appl.* **2015**, *26*, 675–688. [CrossRef]
35. Salman, M.; Mathavan, S.; Kamal, K.; Rahman, M. Pavement crack detection using the gabor filter. In Proceedings of the IEEE Conference on Intelligent Transportation Systems, The Hague, The Netherlands, 6–9 October 2013; pp. 2039–2044.
36. Chou, J.; O'Neill, W.; Cheng, H. Pavement distress evaluation using fuzzy logic and moments invariants. *Transp. Res. Rec.* **1995**, *1505*, 39–46.

37. Zhang, L.; Yang, F.; Zhang, Y.D.; Zhu, Y.J. Road crack detection using deep convolutional neural network. In Proceedings of the IEEE International Conference on Image Processing, Phoenix, AZ, USA, 25–28 September 2016; pp. 3708–3712.
38. Li, G.; Zhao, X.; Du, K.; Ru, F.; Zhang, Y. Recognition and evaluation of bridge cracks with modified active contour model and greedy search based support vector machine. *Autom. Constr.* **2017**, *78*, 51–61. [CrossRef]
39. Shi, Y.; Cui, L.; Qi, Z.; Meng, F.; Chen, Z. Automatic road crack detection using random structured forests. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 3434–3445. [CrossRef]
40. Doycheva, K.; Koch, C.; König, M. Implementing textural features on GPUs for improved real-time pavement distress detection. *J. Real-Time Image Process.* **2019**, *16*, 1383–1394. [CrossRef]
41. Sollazzo, G.; Wang, K.; Bosurgi, G.; Li, J. Hybrid procedure for automated detection of cracking with 3d pavement data. *J. Comput. Civ. Eng.* **2016**, *30*, 04016032. [CrossRef]
42. Rätsch, G.; Onoda, T.; Müller, K.R. Soft margins for adaboost. *Mach. Learn.* **2001**, *42*, 287–320. [CrossRef]
43. Viola, P.; Jones, M.J. Robust real-time face detection. *Int. J. Comput. Vis.* **2004**, *57*, 137–154. [CrossRef]
44. Li, L.; Wang, K.C. Bounding box-based technique for pavement crack classification and measurement using 1 mm 3d laser data. *J. Comput. Civ. Eng.* **2016**, *30*, 04016011. [CrossRef]
45. Yuan, J.; Zheng, Y.; Zhang, C.; Xie, X.; Sun, G.Z. An interactive-voting based map matching algorithm. In Proceedings of the 2010 Eleventh International Conference on Mobile Data Management, Kansas City, MS, USA, 23–26 May 2010; pp. 43–52.
46. Cohen, L.; Kimmel, R. Global minimum for active contour models: A minimal path approach. *Int. J. Comput. Vis.* **1997**, *24*, 57–78. [CrossRef]
47. Vincent, L. Minimal path algorithms for the robust detection of linear features in gray images. In Proceedings of the International Symposium in Mathematic Morphology (ISMM), Amsterdam, The Netherlands, 3–5 June 1998; pp. 331–338.
48. Geusebroek, J.; Smeulders, A.; Geerts, H. A minimum cost approach for segmenting networks of lines. *Int. J. Comput. Vis.* **2001**, *43*, 99–111. [CrossRef]
49. Rouchdy, Y.; Cohen, L. The shading zone problem in geodesic voting and its solutions for the segmentation of tree structures. application to the seg-mentation of microglia extensions. In Proceedings of the IEEE CVPR Workshop, Miami, FL, USA, 20–25 June 2009; pp. 66–71.
50. Avila, M.; Begot, S.; Duculty, F.; Nguyen, T.S. 2d image based road pavement crack detection by calculating minimal paths and dynamic programming. In Proceedings of the IEEE International Conference on Image Processing, Paris, France, 27–30 October 2014; pp. 783–787.
51. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 888–905.
52. Liu, F.; Xu, G.; Yang, Y.; Niu, X.; Pan, Y. Novel approach to pavement cracking automatic detection based on segment extending. In Proceedings of the IEEE International Symposium on Knowledge Acquisition and Modeling, Wuhan, China, 21–22 December 2008; pp. 610–614.
53. Arbelaez, P.; Maire, M.; Fowlkes, C.; Malik, J. Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 898–916. [CrossRef]
54. Bradshaw, G. *Noncontact Surface Geometry Measurement Techniques*; Image Synthesis Group; Trinity College: Dublin, Ireland, 1999.
55. Li, Q.; Zhang, D.; Zou, Q.; Lin, H. 3D laser imaging and sparse points grouping for pavement crack detection. In Proceedings of the 2017 25th European Signal Processing Conference (EUSIPCO), Kos Island, Greece, 28 August–2 September 2017; p. 20362040.
56. Zou, Q.; Zhang, Z.; Li, Q.; Qi, X.; Wang, Q.; Wang, S. Deepcrack: Learning hierarchical convolutional features for crack detection. *IEEE Trans. Image Process.* **2018**, *28*, 14981512. [CrossRef]
57. Liao, J.; Yue, Y.; Zhang, D.; Tu, W.; Cao, R.; Zou, Q.; Li, Q. Automatic Tunnel Crack Inspection Using an Efficient Mobile Imaging Module and a Lightweight CNN. *IEEE Trans. Intell. Transp. Syst.* **2022**. [CrossRef]