Smart Structures and Systems, *Vol. 27, No. 4 (2021) 593-605*
DOI: https://doi.org/10.12989/sss.2021.27.4.593

A fast image-stitching algorithm for characterization of cracks in large-scale structures

Linlin Wang*[1,2], Billie F. Spencer, Jr.[2], Junjie Li[1] and Pan Hu[2,3]

[1] *Faculty of Infrastructure Engineering, Dalian University of Technology, Dalian 116024, China*
[2] *Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*
[3] *College of Automation Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing, 211106, China*

(*Received February 11, 2020, Revised November 5, 2020, Accepted January 3, 2021*)

Abstract. Visual inspection of concrete cracks has been widely used in structural health monitoring (SHM). Capturing high-resolution images is an effective method to visualize a complete crack, but it is difficult to show a whole crack from a single high-resolution image. One feasible method is using image stitching technique to stitch several images into a complete crack map. However, the current image stitching method is a computationally intensive process. Numerous images are required to cover large-scale structures with sufficient resolution, this can be computationally prohibitive. To address this problem, an improved image stitching method for crack damage evaluation is proposed, which can quickly stitch the crack images without affecting the quality of the stitching or the resulting images. Rather than first stitching the images together and then determining the crack maps, we propose to first develop the crack maps for the individual images and then stitch them together. The proposed method reduces the number of redundant matching points between the original images by combining their characteristics during image stitching, so it can reduce the calculation time without affecting the quality. Also, the results will not be influenced by the image stitching seam, which can reduce the complexity of the algorithm. Several experimental results are provided in this article to demonstrate that the proposed method can reduce the calculation time without affecting the quality of image stitching and have better robustness than the current method in use.

Keywords: visual inspection; concrete crack; image stitching; crack properties

1. Introduction

Cracks are the main form of damage that occur in concrete, and their detection is considered to be the basic process of maintaining civil infrastructure (Kim *et al*. 2017). Cracks on/in a concrete structure not only affect the visual appearance, but also affect the load carrying capacity, water tightness and durability (Nishikawa *et al*. 2012, Lins and Givigi 2016). Moreover, cracks are inevitable regardless of the type of concrete structure (Adhikari *et al*. 2014). For the analysis of cracks, crack quantification and identification of crack patterns are necessary to reveal the condition of concrete structures, such as bridges, buildings, and dams. Cracks are progressive in nature, and if they become too long, the concrete can become unstable, leading to catastrophic failure of the structure. Therefore, timely inspection and evaluation of cracks are necessary. Early detection allows preventive measures to be taken to prevent further damage and possible failure (Jang and An 2018). The Chinese code for design of concrete structures (GB50010-2015) assigns the permissible limit of crack width under various environmental categories for reinforced concrete structures. In general, the maximum allowable width of the crack cannot be over 0.3 mm.

Currently, most traditional visual assessment relies on manual inspectors. This detection method is inefficient and expensive (Graybeal *et al*. 2002, Liu *et al*. 2014), and the assessment results are subjective (Islam and Kim 2019). Due to the inevitable drawbacks of traditional visual assessment, there is an urgent need for efficient and economical methods to assess damage in concrete structures.

To achieve this goal, several mobile devices (e.g., cameras and unmanned aerial vehicles) can be used to obtain structural visual information. Through collecting image information, we can evaluate the condition of structures automatically, which saves time and cost. Many researchers have proposed crack detection methods based on image processing, such as detection methods based on percolation-based image processing (Yamaguchi and Hashimoto 2010, Qu *et al*. 2018), wavelet transformation (Hutchinson and Chen 2006, Andreaus *et al*. 2017), and convolutional neural networks (Cha *et al*. 2017, Kim and Cho 2019). However, the above methods are mostly applied for the image with only one single crack. In addition, to acquire high resolution images of the concrete surface, it is necessary to capture the image from the near field, which makes it only cover a small area at one time, and it is difficult to obtain a complete crack surface from a single image or a single frame. Identification and detection of individual crack are not sufficient for understanding the behavior of the structures. The engineer needs to know how

*Corresponding author, Ph.D. Student,
E-mail: llwang@mail.dlut.edu.cn

ISSN: 1738-1584 (Print), 1738-1991 (Online)

the crack patterns change with time and where they lie on structural members (Adhikari *et al*. 2014). For example, we assume that the length of a crack in a large-scale structure is 2 m. The image which can display the entire crack needs to be acquired at a distance of at least 3 m, if the image is taken by a normal camera (3000×4000 resolution). The captured image can be characterized by approximately 1 pixel by 0.8 mm. However, the maximum allowable crack width in general concrete structures is only 0.3 mm, which is not sufficient to characterize the crack feature. To get the high-resolution images using a normal camera, we need to capture images in a near field. If an image is taken at a distance of 0.5 m from the structure, the captured image can be characterized by 1 pixel by 0.1 mm. As a result, numerous images need be collected, when clear crack maps of concrete surfaces are needed. But detecting damages on all images one by one will take a lot of time and cannot represent the entire image.

To tackle this problem, all the images captured from the surface of the concrete structure can be stitched into a single wide view by image stitching techniques. Image stitching technology has been widely used in image processing (Szeliski 2006), photogrammetry (Attard *et al*. 2018) and has been applied in commercial applications (Agarwala *et al*. 2004). In the field of civil engineering, Zhu *et al*. (2010) created a panorama by automatically stitching pictures, depicting the bridge column area. A structural health monitoring software that can perform multi-image stitching using fixed cameras is conducted in the studies of Jahanshahi *et al*. (2011). La *et al*. (2015) provided a robotic system that integrated with several non-destructive evaluation sensors, using an image stitching algorithm to achieve simple bridge evaluation and condition monitoring. Whereas, Adhikari *et al*. (2014) offered a retrieval of concrete crack properties based on image processing techniques. They indeed provide powerful and accurate crack identification ability. However, although the above studies applied image stitching methods, they are inefficient because they did not focus on the cost of time.

It is well known that image stitching is computationally expensive. While the image stitching algorithm can find more matches in source images and synthesize higher quality images, it takes a lot of computation time. To make up for this, researchers either compromise on quality or invest more time optimizing code for specific hardware architectures (Wagner *et al*. 2010, Chia *et al*. 2012, Pulli *et al*. 2012, Du *et al*. 2019). Thus, it is critical to develop an image stitching method that can satisfy the stitching quality and take less calculation time.

In this study, an improved image stitching method for cracks of concrete structures is proposed. The goal of this research is to develop crack maps of the structure. Rather than first stitching the images together and then determining the crack maps, we propose to first develop the crack maps for the individual images and then stitch them together. Through such processing, the number of matching points between the original images is reduced, thereby reducing the stitching time without affecting the quality. Moreover, the direct image fusion method can be applied to eliminate the influence of the stitching seam. Results show the proposed approach can reduce the complexity of the algorithm and the associated computational time, and has better robustness.

The major contribution of the proposed technique is to propose a new approach to retrieve concrete crack properties based on the image stitching method. In the past, many scholars have paid attention to detect cracks for a single image or improve the image stitching speed alone. However, it is hard to characterize a complete crack image with a single image or a single frame. To confirm the high-resolution detection requirements, it is essential to combine the two and consider the cost of stitching. Instead, the proposed method begins from the crack maps and then crack characteristics are retrieved from the integrated stitching crack maps by image processing techniques. Through this method, the calculation time and algorithm complexity can be reduced without affecting the stitching quality.

The remainder of this article is organized as follows. Section 2 starts from a brief overview of the proposed method and provides technical detail about crack detection and image stitching. Experimental descriptions, results and discussion are presented in Section 3. Section 4 includes the conclusion and suggestions for future work.

2. Methodology

This section explains the overall diagram of the proposed technique, a novel method for acquiring the concrete crack properties by image stitching in large-scale structures. Fig. 1 shows the flow chart of the method, which is divided into two main stages: (1) crack detection, which produces a crack map for each structural element surface; and (2) image stitching, which stitches together the cracks distributed on different images into a complete crack map. Details about these two stages are explained in the following sub-sections.

2.1 Crack detection

The overview of crack detection using image processing is shown in this section. The method provides a simple and fast automatic crack detection algorithm on concrete structures that consists of four steps: image preprocessing,

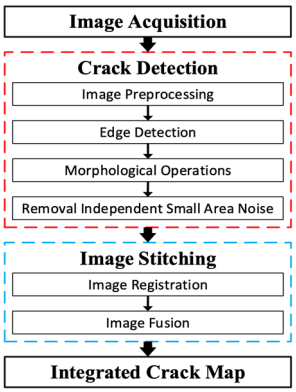


Fig. 1 Schematic overview of the proposed image stitching procedure

edge detection, morphological operations and removal independent small area noise.

2.1.1 Image preprocessing

The first step of image preprocessing includes linear stretching and converting the RGB image into the grayscale image. The acquired images usually appear with uneven brightness, because they are affected by the environment. The image preprocessing can increase the contrast between the background and the target cracks, so that the crack features are prominent. This prepares for subsequent edge detection.

First, the images are enhanced by linear stretching to increase the image contrast and brightness (Osman *et al*. 2009) and contribute to the visualization of the crack.

$$g(x,y) = k * f(x,y) + d \tag{1}$$

where $f(x,y)$ and $g(x,y)$ is the original image and the output image, respectively. The parameter k controls the contrast of the image, and the parameter d controls the brightness of the image.

Then, the color images can be converted into grayscale images by data conversion (0~255). Image graying can greatly reduce the necessary number of calculations.

2.1.2 Edge detection

The second step is edge detection, which is an image processing technique for finding the boundaries of objects where the brightness changes significantly. The intensity of a cracked pixel changes abruptly compared to its neighbor in a grayscale image. Therefore, using edge detection can locate cracks in the concrete surface present in each image. Many different edge detection methods have been proposed in the past few decades. Among them, the most commonly used is the Canny Edge Detector, which was developed by Canny (1986). The Canny operator detects image edges by performing the following steps sequentially (Cao *et al*. 2018).

Step a

Apply Gaussian Smoothing to Image Using Gaussian Convolution. The Canny operator applies the first derivative of the two-dimensional Gaussian function as a noise filter to eliminate image noise. It then performs convolution processing to smooth the image

$$G(x,y) = \frac{1}{2\pi\sigma^2}\exp\left[-\frac{x^2+y^2}{2\sigma^2}\right] \tag{2}$$

where σ represents the standard deviation of the Gaussian filter function, which is manually set and controls the image smoothness.

Step b

Filtering the Image Using the First Derivative of the Gaussian Operator to Obtain the Gradient Intensity and Direction of the Image. Suppose that $\mathrm{I(i,j)}$ is a smooth image. The first derivative of the image in the x and y directions is

$$\mathrm{P_x[i,j]} = \frac{1}{2}(\mathrm{I[i,j+1] - I[i,j] + I[i+1,j+1] - I[i+1,j]}) \tag{3}$$

$$\mathrm{P_y[i,j]} = \frac{1}{2}(\mathrm{I[i,j] - I[i+1,j] + I[i,j+1] - I[i+1,j+1]}) \tag{4}$$

Therefore, the gradient intensity $\mathrm{M[i,j]}$ and gradient direction $\mathrm{\theta[i,j]}$ are

$$\mathrm{M[i,j]} = \sqrt{\mathrm{P_x[i,j]^2 + P_y[i,j]^2}} \tag{5}$$

$$\mathrm{\theta[i,j]} = \arctan\left(\frac{\mathrm{P_y[i,j]}}{\mathrm{P_x[i,j]}}\right) \tag{6}$$

Step c

Non-maximum Suppression. The purpose of this step is to exclude non-edge pixels and leave only a few thin lines (candidate edges). The gradient directions can be divided into 8 directions. The value of the derivative of each pixel is compared with the modulus of the adjacent pixels in the images for the 8 directions along the edge detection points of the argument direction. Finally, the pixel with the largest partial derivative is used as the edge point.

Step d

Detecting and Connecting the Edge Points Using the Dual-Threshold Method. The Canny operator requires setting the high threshold T_{high} and low threshold T_{low}. Generally, $T_{high} = 3T_{low}$ or $T_{high} = 2T_{low}$. When the gradient intensity $\mathrm{M[i,j]}$ is greater than the T_{high} ,the point is marked as an edge point, and when the gradient intensity $\mathrm{M[i,j]}$ is less than the T_{low} , the pixel will be removed. When the gradient intensity $\mathrm{M[i,j]}$ is between T_{high} and T_{low} , the point is marked as a candidate edge point; further judgments are later made by combining each candidate point with its surrounding pixels.

2.1.3 Morphological operation

The third step is to apply the morphological method, a broad set of image processing operations that processes images based on shapes.

The basic morphological operators are dilation and erosion. The closing operation is the erosion of the dilating of a set $\boldsymbol{A}$ by a structuring element $\boldsymbol{S}$ (Haralick *et al*. 1987). $\boldsymbol{S}$ defines which neighboring pixels are included in the operation. The specific process is shown in the Eq. (7).

$$A \bullet S = (A \oplus S) \ominus S \tag{7}$$

where $\bullet$ is the opening operation, and $\oplus$ and $\ominus$ denote dilation and erosion, respectively.

The closing operation is employed to fill gaps between edges, connect narrow discontinues and not significantly change their area while smoothing the boundaries of larger objects.

2.1.4 Removal independent small area noise

As the final step, an area threshold is applied on the binary image to remove some small objects, which appear due to surface unevenness of the structure. The number of pixels is calculated for each connected component (object) and the area of each object is estimated. The objects with an area less than the area threshold T_A pixels will be added to the background pixels.

$$B(x,y) = \begin{cases} 1 & if\ AP < T_A \\ 0 & if\ AP > T_A \end{cases} \quad (8)$$

where $B(x,y)$ is the binary image, the T_A is the area threshold, and the $\boldsymbol{AP}$ is the number of the object area pixel.

According to the above steps, we can derive the crack map from different images, and then image stitch them together.

2.2 Image stitching

Image stitching is one of the most widely used algorithms in computer vision (Szeliski 2006). It can combine multiple images by overlapping them into one panoramic image. For large-scale structures, a defect may be distributed over several images. In order to obtain a panoramic image of damage, which is called a crack map, it is necessary for the images to be automatically stitched together to improve the accuracy of detection. An image stitching algorithm developed by Brown and Lowe (2007) has been adopted in this study. This method is based on extracting image invariant features and matching them to new images. It has several advantages over other available methods. This method does not require image initialization or fixed image ordering, because it is based on features for registration which are invariant to rotation, zooming, and illumination change in the input images (Brown and Lowe 2007). In this section, how to achieve this goal is introduced. The Figs. 2(a)-(d) illustrate the details of image stitching.

2.2.1 Image registration

Image registration is the process of transforming different sets of data into one coordinate system. The purpose of this process is to align two or more overlapping images (Chen and Klette 1999), Image registration is the core step of image stitching, which has a direct impact on stitching speed, accuracy and success rate. Image registration is mainly divided into two categories: pixel-based methods, and feature-based methods. At present, the feature-based approach is the most widely used image registration method, using the sparse feature descriptors and feature matching implemented sequentially (Lowe 2004). Different from the pixel-based methods, the feature-based methods construct different feature descriptors by selectively extracting sparse feature points and calculate the adjacent relationships between input images automatically. Therefore, the method has fewer calculations and is faster and more robust (Brown 2005). The proposed system in this study is essentially a feature-based image stitching algorithm. In the feature-based image registration process, key-point detection, which detects control points such as distinctive objects, edges, topographies, points, line intersections and corners is most important part (Jahanshahi *et al*. 2011). The Scale Invariant Feature Transform (SIFT) is a popular choice for key-point detection. The SIFT key-points are invariant to changes in scale and rotation. Moreover, the SIFT operator is also highly discriminative and robust to significant amounts of image noise.

When the feature points are extracted based on the SIFT algorithm, they need to be matched, as shown in Fig. 2(a). The match points obtained by the SIFT are the candidate match points. Then, eliminating mismatches of key-points can help ensure accuracy. The candidate match points are filtered by the following methods. After filtering, the good match points are obtained.

(1) Lowe (2004) introduced an effective matching strategy based on comparing the distance between the closest neighbor to that of the second closest neighbor. *EP* and *EP*' represent the distance to the selected feature point *E* in another matching image, respectively. If the ratio of *EP*/*EP*' is less than the threshold *T*, the matching points (*E*, *P*) are successfully matched in pairs. In this study, the threshold *T* is set to 0.6, that is, all matches in which the distance ratio is greater than 0.6 were rejected. By doing so, most of the mismatches were discarded.

(2) In addition, Random Sampling Consistency Algorithm (RANSAC) (Fischler and Bolles 1981) is used to compute the homography matrix between two images as well as to find outliers, which are the incorrect matching points, to improve the image key-points matching rate.

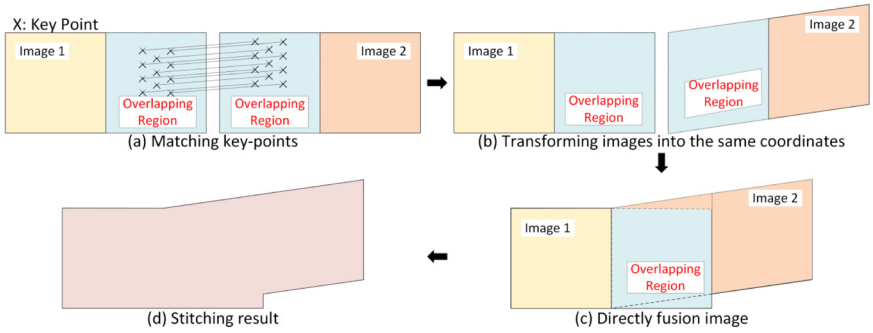


Fig. 2 The processing of image stitching

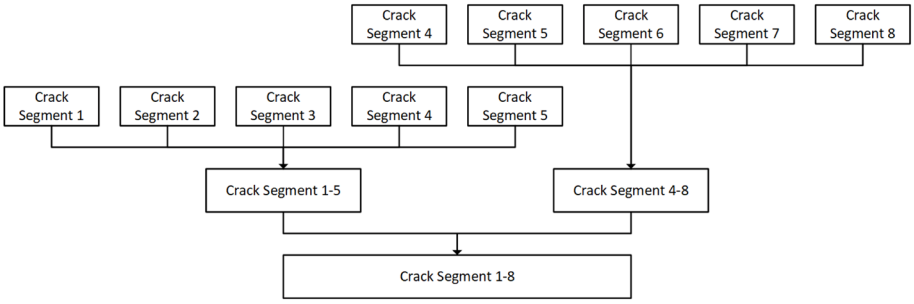


Fig. 3 The schematic diagram of stitching eight crack segments by grouping

2.2.2 Image fusion

The image stitching is completed by selecting an appropriate image fusion method, after the matching points of the crack image are extracted, and finally, an integrated crack map of the large-scale structures is generated.

The homography matrix can be calculated by the good match points obtained after filtering. It can convert all points in one view to another according to the projection transformation of Eq. (9), that is, the angles of view of the two images are the same, as shown in Fig. 2(b). Then, the image is copied to a specific position of another image, and image fusion can be completed to expand the view, as shown in Fig. 2(c).

$$X^{'} = HX \tag{9}$$

where $\boldsymbol{X}^{'}$and $\boldsymbol{X}$ is the transformed image and the original image, respectively. And $\boldsymbol{H}$ is the homography matrix.

However, due to the influence of natural factors or human factors, such as image collection equipment and the shooting angles, the acquisition images will have obvious stitching seams after stitching. Therefore, in most cases, the seams of the stitched images need to be eliminated. Since the binary image of the crack map is taken as the input image, the problem of generating the stitching seam is not involved in this study. Directly taking the data of one of the overlapping regions of the image as an image fusion algorithm can be used to achieve the image stitching, which reduces the complexity of the algorithm.

To reduce the cumulative error generated when stitching multiple images, the intermediate image is the best reference image for multiple stitching. Moreover, multi-image stitching can be better achieved by reasonable group stitching. For example, as shown in Fig. 3, when stitching 8 crack segments, the images can be divided into 2 groups, which are crack segment 1-5 and crack segment 4-8. The 2 groups of images are first stitched one by one, and then stitching them together to form crack segment 1-8.

3. Experimental validation and discussion

3.1 Laboratory validation

The proposed method was tested on a concrete wall with many cracks. The main goals of the laboratory test were: (1) to determine the effectiveness of the proposed method, and (2) to determine that the proposed method can reduce the calculation time without affecting the stitching quality.

3.1.1 Description of the experiment

A large scale, concrete wall with many cracks, as shown in Fig. 4, is used for validating the proposed technique. Test images are captured through a Sony X-T10 digital camera, with 55 mm lens, and the flash function is not utilized. Two test images having 4896 × 3264 resolution are sequentially taken at roughly 0.5 m working distance. Each of the images has at least 50% overlap with its neighboring images.

Image matching is a very time-consuming process. The feasibility of the proposed method can be quickly verified by this laboratory experiments. The calculation time in this study starts from the image pre-processing and ends with the successful image stitching.

All algorithms proposed in this study are implemented using Microsoft Visual Studio .NET. Intel® Open Source Computer Vision Library (OpenCV) were used as the prototype's main image processing toolbox. It is open source (Bradski and Kaehler 2008). In practice, the linear stretching parameters are 1.2 and 50 respectively. A low threshold for Canny edge detection is set to 80, and the high threshold is three times the low threshold. The area threshold T_A is set to 200 pixels. The objects with an area less than 200 pixels are added to the background pixels. Note that all parameters and thresholds are not automatically determined in this proposed method. Users should tune the parameters for different uses based on those suggested in the training steps and their own experience.

Fig. 4 A test large scale concrete structure including many cracks

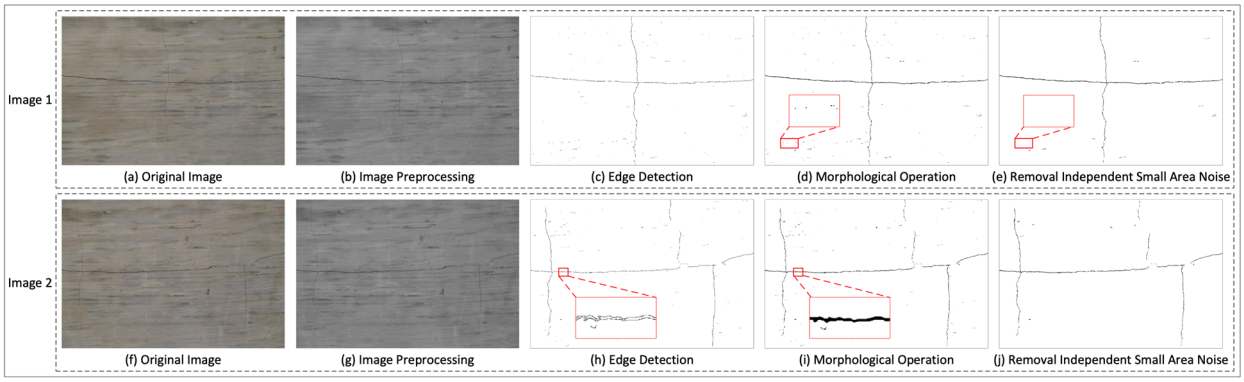


Fig. 5 Results of crack detection for Image 1 and Image 2.
(a) and (f) Original image; (b) and (g) Image processing results; (c) and (h) Edge detection results;
(d) and (i) Morphological operation results; and (e) and (j) Removal independent small area noise results

3.1.2 Experimental results

Figs. 5(a)-(j) shows the results of crack detection for Image 1 and Image 2 in the first stage of this study. Image 1 and Image 2 are images of two crack regions acquired continuously, with more than 50% overlap. Original image, image preprocessing, edge detection, morphological operation and removal independent small area noise for Images 1 and Image 2 are shown in Figs. 5(a)-(j), respectively. Fig. 5(b) and Fig. 5(g) show the image preprocessing results. Compared with the original image of Fig. 5(a) and Fig. 5(f), the contrast and brightness of the images increase to make the crack clearer, and the images are converted into the grayscale image. Fig. 5(c) and Fig. 5(h) show the edge detection results. The edges of the cracks and some noise can be presented after applying the Canny algorithm. An example of the edge is shown in the red rectangle in Fig. 5(h).

Fig. 5(d) and Fig. 5(i) show the morphological operation results. The crack edges are filled after the Closing operation is applied, and a filling effect example is shown in the red rectangle in Fig. 5(i). It can be seen from the two red rectangles in Fig. 5(h) and Fig. 5(i) that the crack regions are completely filled, which facilitates subsequent retrieval of crack properties. Fig. 5(e) and Fig. 5(j) show the removal independent small area noise results. The area threshold T_A is set to 200, and the objects with an area less than 200 pixels are added to the background pixels. Compared the Fig. 5(d) and Fig. 5(e), some small noise points are removed. Through the above processing, the cracks are detected and can be clearly seen in Figs. 5(a)-(j).

The results of image stitching for Image 1 and Image 2 in the second stage of this study are presented in Figs. 6(a)-(c). Fig. 6(a) shows the candidate match points of the two images, and the candidate match points are 870; Fig. 6(b) shows the match points after the filtering, and the good match points are 35; and Fig. 6(c) shows the final result of the stitched image. From Figs. 6(a)-(b), the number of incorrect matching points can be greatly eliminated by filtering, thereby improving the image key-points matching rate; and from Fig. 6(c), images can be fused by direct fusion from one of the image overlaps. There are no stitching seams in the final result of the stitched image. So, the proposed method can reduce the complexity of the

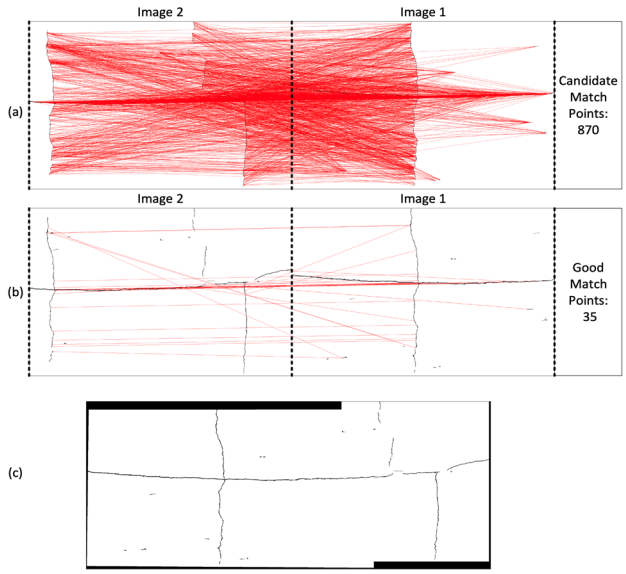


Fig. 6 Results of image stitching for Image 1 and Image 2.
(a) The candidate match points of the two images;
(b) The match points after the filtering of the two images; and (c) the final result of the stitched image

algorithm.

Table 1 describes the calculation time of the proposed method and the current method for Image 1 and Image 2 of the full size (4896 × 3264). Among them, the current method is to first use the image stitching algorithm to stitch several images with crack damage into a crack map, and then perform crack detection on the crack map. Both the proposed method and the current method are calculated from the input of the original crack images until the complete crack map is stitched. The calculation time of entire processing is including the crack detection process and image stitching process. For each method, the average of the five calculations is taken as the final calculation time, because the time for each program computation is different. By comparing the eighth column of Table 1, i.e., the final calculation time, it can be illustrated that the proposed method significantly reduces the computational time. Specifically, the reduction ratio is calculated and displayed

in the tenth column of Table 1. It can be seen from the results that the proposed method can reduce the calculation time by 31.95%. Furthermore, the computational time of each step is calculated. As shown in Table 1, the calculation time of crack detection obtained by the proposed method and the traditional method is roughly the same, almost 35 s, whereas the proposed method can greatly reduce the calculation time in the process of image stitching. It can be evidence that the proposed method mainly reduces the calculation time in the process of image stitching.

The main reason for the calculation time reduction of Table 1 is that the proposed method uses the crack map of each image as the input image of the image stitching by considering the characteristics of the damaged image itself, which results in a significant reduction in the number of matching points of the proposed method. In fact, matching the key-points is the most time-consuming part of the image stitching process. The proposed method is to reduce the computation time by reducing the number of image matching points Table 2 shows a comparison of the number of match points between the proposed method and the current method. As shown Table 2, the number of candidate match points and the good match points after filtering are obviously less than those of the current method. Taking the full-size images as an example, the number of candidate match points of the proposed method is reduced by 5771 compared with the current method, and the number of good match points is reduced by 765.

Crack skeletonization is an important step in retrieval of crack segments which show different paths at their branching points (Nishikawa *et al*. 2012). The skeleton of the crack can be quickly extracted by the most commonly used Zhang and Suen (1984) thinning algorithm. The skeleton image of the stitching crack map by Image 1 and

Table 1 The comparison of calculation time of the proposed method and current method

Method	Step	Calculation time (s)					Average time (s)	Method comparison	
		T1	T2	T3	T4	T5		Reduce time (s)	Reduction ratio (%)
New method	Entire processing	64.1163	64.2317	61.5151	64.7798	64.0426	64.3371	-30.2057	-31.95
	Crack detection	34.0885	35.1326	35.4817	35.0653	34.9366	34.9409		
	Image stitching	30.0278	29.0991	29.0334	29.7145	29.106	29.3962		
Current method	Entire processing	94.274	94.2025	94.3891	95.3839	94.4644	94.5428		
	Image stitching	59.6421	59.5359	59.776	60.85	59.6514	59.8911		
	Crack detection	34.6319	34.6666	34.6131	34.5339	34.813	34.6517		

*Note: '-' represents the time is reduced

Table 2 The comparison of match points of the proposed method and current method

Method	Match points		Method comparison	
	Candidate match points	Good match points	Candidate match points	Good match points
New method	870	35	-5771	-765
Current method	6641	800		

*Note: '-' represents the match points are reduced

Table 3 The AWP values for each crack segment (Segment A-F and Segment L1 and L2) of the proposed method and the current method

	New method			Current method			***AWP*** Error (pixel)
	AP (pixel)	***LP*** (pixel)	***AWP*** (pixel)	***AP*** (pixel)	***LP*** (pixel)	***AWP*** (pixel)	
Segment A	20974	1280	16.39	21467	1280	16.77	+0.39
Segment B	7755	926	8.37	7312	818	8.94	+0.56
Segment C	6454	696	9.27	7591	869	8.74	-0.54
Segment D	2480	292	8.49	2480	292	8.49	0.00
Segment E	8942	944	9.47	8828	931	9.48	+0.01
Segment F	4403	373	11.80	4301	331	12.99	+1.19
Segment L1	21484	1470	14.61	21934	1482	14.80	+0.19
Segment L2	3532	257	13.74	3540	260	13.62	-0.13

*Note: '+' and '-' represent ***AWP*** is added or reduced, respectively

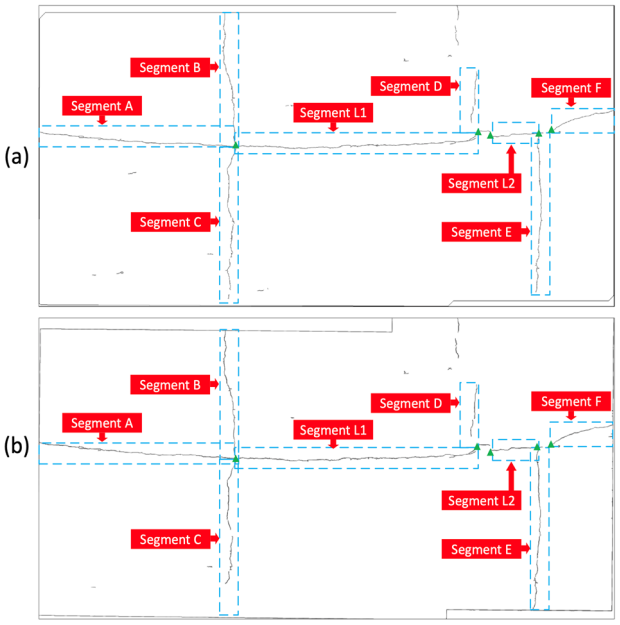
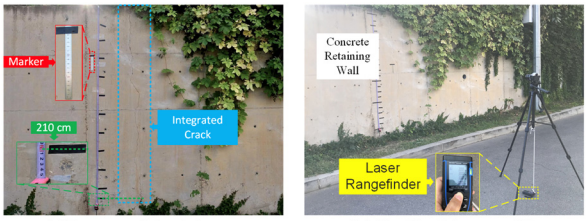


Fig. 7 The skeleton image of the stitching crack map by Image 1 and Image 2.
(a) The skeleton image of the crack map obtained by the proposed method; and (b) The skeleton image of the crack map obtained by the current method.

Image 2 is shown in Figs. 7(a)-(b), which can be obtained by the thinning algorithm. From the skeleton image, the crack is subdivided into 8 segments by 5 branching points, wherein the 5 branch points are as shown by the green triangle and the 8 crack segments (Segment A-F and Segment L1 and L2) are shown by the blue dotted rectangle in Figs. 7(a)-(b). To compare the quality of the complete crack map obtained by the proposed method with the quality of the complete crack map obtained by the current method, the average width pixel (***AWP***) of the crack is defined to assess the quality of the stitching image in this study. In fact, cracks in concrete structures have a smaller number of pixels than their background but a larger length to width ratio. Therefore, the number of pixels of the crack skeleton can be a good approximation of crack length, and the average width pixel (***AWP***) can be calculated by dividing total area pixel (***AP***) by length pixel (***LP***) of the segment. This indicator is easy to obtain and has a certain representativeness. This definition is shown in the Eq. (10).

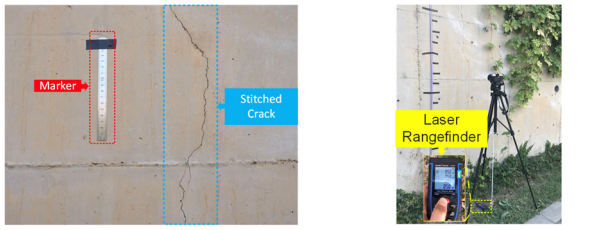
$$AWP = \frac{AP}{LP} \tag{10}$$

where ***AWP*** is the average width pixel, ***AP*** is the area pixel of the segment, and ***LP*** is the length pixel after thinning the segment, respectively.



(a) Integrated Crack Image (b) Working Distance of 3 m

Fig. 8 (a) The integrated crack in the concrete retaining wall; and (b) Picture of the working distance of 3 m



(a) Stitched Crack Image (b) Working Distance of 0.5 m

Fig. 9 (a) The stitched crack in the concrete retaining wall; and (b) Picture of the working distance of 0.5 m

Table 3 shows the ***AWP*** values for the crack segments (Segment A-F and Segment L1 and L2) of the proposed method and the current method. The difference in crack quality obtained by the two methods is small when comparing the ***AWP*** values. Specifically, the errors are all around 1 pixel value. Hence, the proposed method does not affect the image quality while reducing the stitching time.

3.2 Field validation

To verify the applicability of the proposed method in large-scale structures in outdoor conditions, a field test was undertaken on a concrete retaining wall with a longitudinal crack. The length of the crack is approximately 210 cm. Fig. 8(a) shows the integrated crack in the concrete retaining wall. Test images are captured by a normal camera (Sony X-T10 digital camera with 4896 × 3264 resolution). The entire crack image has to be acquired at a distance of at least 3 m, as shown in Fig. 8(b).

In this test, a steel ruler with a length of 150 mm was used as a marker to get the conversion relationship between the unit pixel and the actual unit (mm). The marker is shown in the red rectangle in Fig. 8(a) and Fig. 9(a).

Assume that the number of pixels of the marker is a pixel, and the size of the marker is b mm, the measurement accuracy u is calculated as follows

$$u = \frac{b}{a} \tag{11}$$

Table 4 The measurement accuracy of images acquired at different working distances

	Working distance (m)	ROI size	a (pixel)	b (mm)	u (mm/pixel)
Entire crack image	3.00	50 × 187	187	150	0.80
Stitched crack image	0.50	200 × 1195	1195	150	0.12

In this study, the actual length of the marker is 150 mm. By setting the region of interest (ROI), the measurement accuracy in the complete crack map and in the subsequent stitched image can be obtained. Table 4 shows the measurement accuracy of images acquired at different working distances. For entire crack image, 1 pixel roughly represents 0.8 mm. This accuracy cannot characterize the crack feature, because the maximum allowable width of the crack is usually only 0.3 mm. To improve the accuracy of images using a normal camera, it is necessary to capture images in a near field. Some stitched crack images were taken at a working distance of 0.5 m. To improve the interference of the useless area, this study sets the ROI of the images based on experience. The ROI size is 2000 ×3264. Fig. 10 shows the ROI of the total of 16 stitchedcrack segment images acquired in this field experiment. And each of the images has at least 60% overlap with its neighboring images. The accuracy can be obtained from the Table 4 as 0.12 mm, so the images can be used for visual inspection.

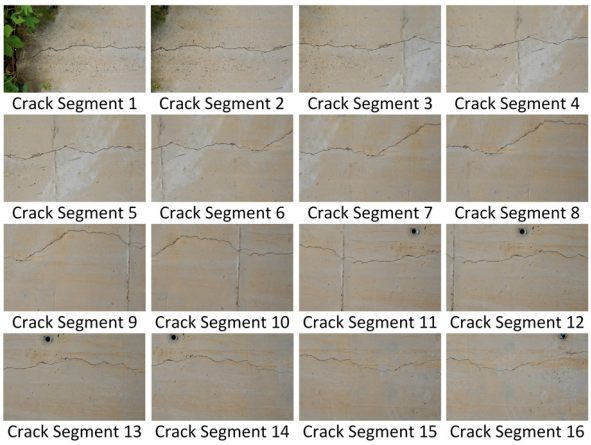


Fig. 10 The ROI of 16 images acquired in this field test

Figs. 11(a)-(b) shows the stitching results of 16 crack segments from field test using the proposed method and the current method. The results were all stitched by the same group stitching method. The group approach is first forming crack segment 1-10 by stitching crack segment 1-4, crack segment 4-8 and segment 7-10. Then, crack segment 1-10

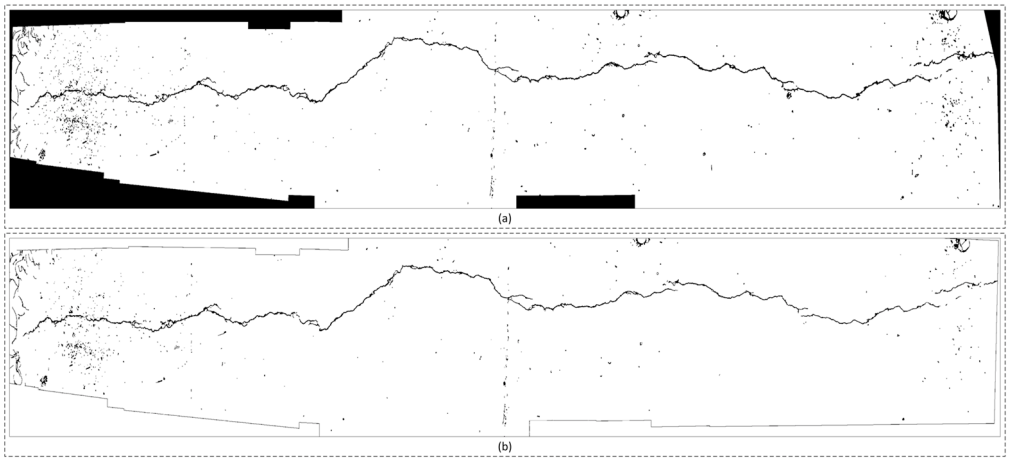


Fig. 11 (a) The stitching results of 16 crack segments obtained by the proposed method; and (b) The stitching results of 16 crack segments obtained by the current method

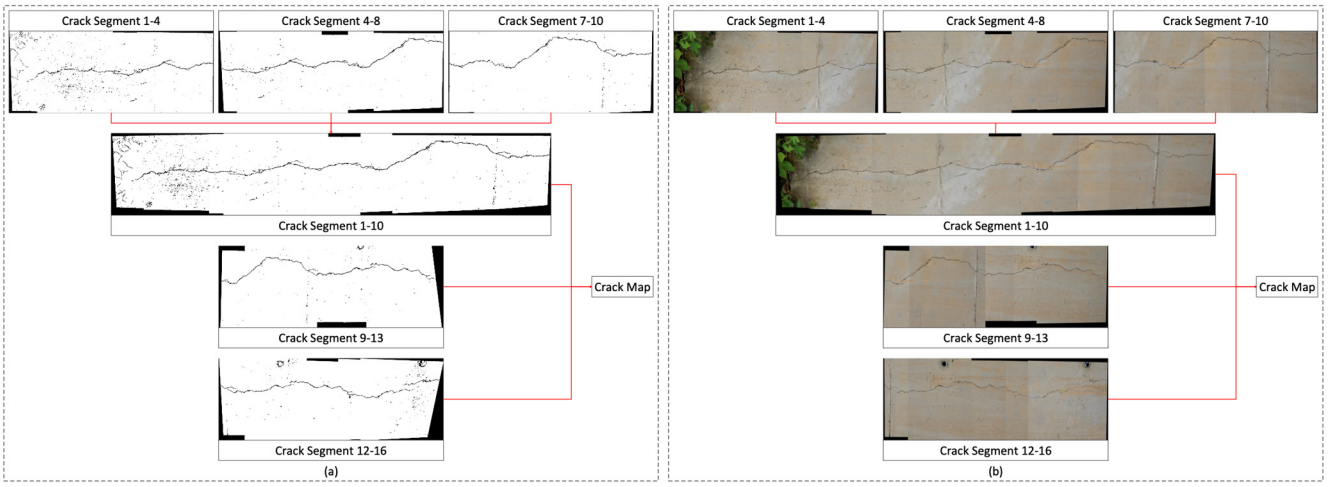


Fig. 12 (a) The crack segments for stitching by the proposed method; and (b) The crack segments for stitching by the current method

just obtained, and crack segment 9-13 and crack segment 12-16 are stitched together to gain an integrated crack map. Among them, the crack segment 9-13 is used as the stitching reference image. Each crack segment is shown in Figs. 12(a)-(b). By comparing the stitching crack map results, it can be seen that the results obtained by the two methods are roughly the same. However, the front part of both results has been deformed. It is because that there is no guarantee that all images are on the same plane during capturing pictures process. It is also the result of the accumulation of stitching deformation. Both of them are completely inevitable.

3.3 Discussion

The robustness and the efficiency analysis for stitching multiple images of the proposed method and current method are discussed in this section.

Figs. 13(a)-(b) shows that the number of good match points and the calculation time obtained by the proposed method and the current method varies with the number of candidate match points. From Fig. 13(a), the change in the number of good match points obtained by the proposed method and the current method can be observed by changing the number of candidate match points, which are the blue curve and the red curve, respectively. It can be seen from the blue curve in Fig. 13(a) that the number of good match points obtained by the proposed method is not greatly changed by the number of candidate match points. On the contrary, the number of good match points obtained by the current method increases significantly with the increase of the number of candidate match points, that is, the red curve shows a significant upward trend. In Fig. 13(b), the calculation time obtained by the proposed method and the current method is compared when changing the number of candidate match points. It can be seen from the blue curve in Fig. 13(b) that the calculation time of the proposed method is about 64.2839 s when the number of candidate match points is changed. In the same case, the calculation time obtained by the current method fluctuates greatly, i.e., the red curve, and most of the points are out of the average calculation time of 87.2348 s. Therefore, when comparing the proposed method and the current method in Figs. 13(a)-(b), the proposed method has a small change in the number of good match points and the calculation time after varying the candidate match points. In addition, the stitched result obtained by the current method is obviously deformed when the candidate match points are 100, but the stitched result of the proposed method is not affected by the candidate match points changing. The stitched results obtained by two methods are shown in Figs. 14(a)-(b). So, the proposed method has strong robustness.

To save space and further investigate the robustness for the field test, here only give the comparison results for crack segment 7 and crack segment 8. It can be seen from Figs. 15(a)-(b) that the good match points and the calculation time obtained by the proposed method smaller change compared to the current method, when the number of candidate match points is varying. Therefore, the same conclusion can be drawn that the proposed method is more robust.

Fig. 16 shows that the calculation time obtained by the proposed method and the current method varies with the number of stitched crack segments. It can be seen that the blue curve all under the red curve. That is, the calculation time of the proposed method is less than the calculation time of the current method, no matter how many images are stitched. Therefore, the proposed method can improve the

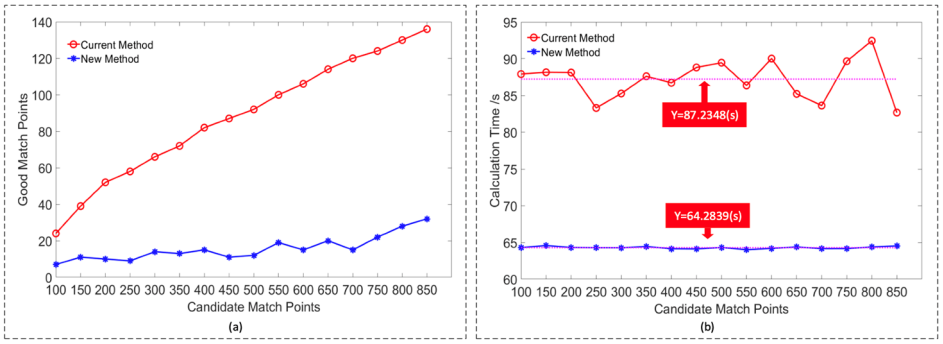


Fig. 13 (a) The number of good matching points obtained by the proposed method and the current method varies with the number of candidate match points for laboratory test images; and (b) The calculation time obtained by the proposed method and the current method varies with the number of candidate match points for laboratory test images

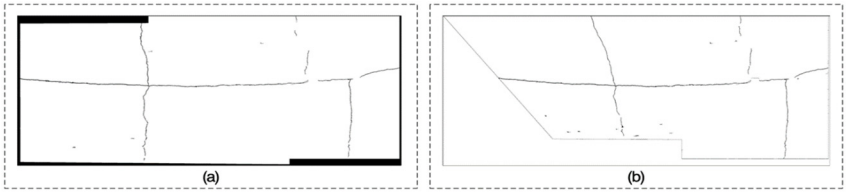


Fig. 14 (a) The stitched result obtained by the proposed method when the candidate match points are 100; and (b) The stitched result obtained by the current method when the candidate match points are 100

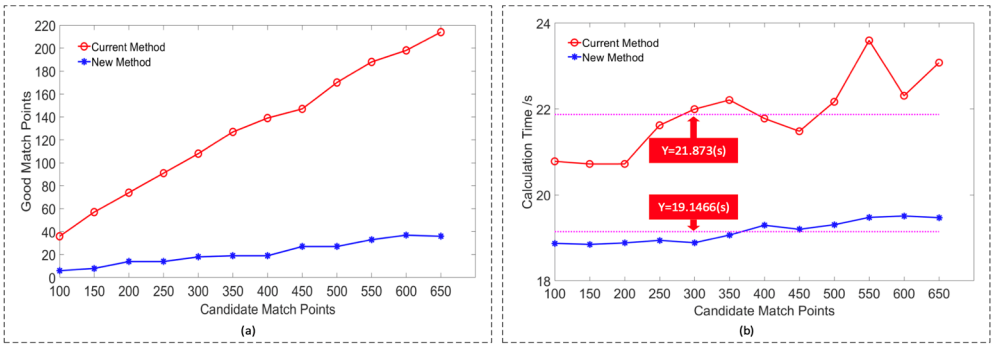


Fig. 15 (a) The number of good matching points obtained by the proposed method and the current method varies with the number of candidate match points for field test images; and (b) The calculation time obtained by the proposed method and the current method varies with the number of candidate match points for field test images

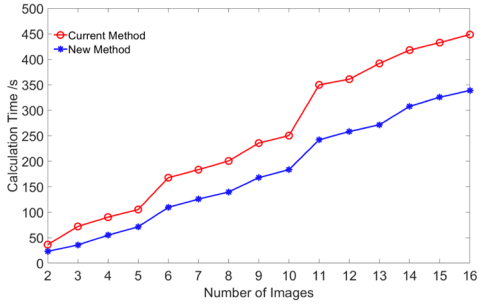


Fig. 16 The calculation time obtained by the proposed method and the current method for different numbers of stitched images

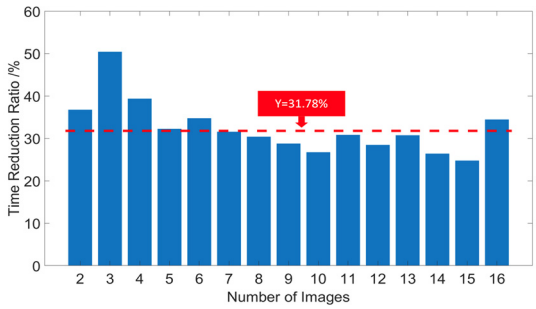


Fig. 17 The time reduction ratio of the proposed method for different number of stitched images comparing with the current method

computational efficiency for different number of stitched images. Furthermore, the reduction ratio for different number of stitched images is shown in Fig. 17. As the number of images changes, the rate of increase in efficiency is substantially stable, with fluctuating around 31.78%.

4. Conclusions

Visual inspection of cracks is frequently a concern for inspection engineers because cracks not only affect the visual appearance of the structure but also the integrity of the structural system. However, a single crack image is usually not enough to express the overall damage, and it is a time-consuming process to stitch these images into a complete crack map before crack property detection. In this study, a new method for fast retrieval of concrete crack properties based on image stitching technology is proposed. The study focuses on reducing the calculation time without affecting the quality of crack detection. The key idea is to start with the characteristics of the crack image and reduce the number of wrong matching points. To achieve this goal, cracks are detected from individual images first and then the integrated crack map is obtained by stitching crack images together. Using the presented method cannot only reduce the calculation time, but also reduce the complexity of the algorithm without affecting the quality of the stitching image. Moreover, this paper also compares robustness through varying the number of good match points and calculation time by the proposed method and the current method when the number of candidate match points is changing. It concludes that the proposed method is more robust. Finally, 16 crack segments of a whole crack are collected in field test. Comparing the computational time taken by the two methods to stitching different numbers of images, the proposed method can generally increase an efficiency of average 31.78% without affecting stitching quality.

In the future, the proposed method will try to adopt the image acquired in a more complicated environment. The contrast and brightness in different environments vary greatly. In a harsh environment, the image is seriously affected by the light and other conditions. The key points will be reduced, so follow-up research is still needed to improve the stability of the method. Furthermore, rather than stitching the entire structure images, only a certain part of the damage and its surroundings need to be stitched together in practical. However, it may also be necessary to process a large number of images. Therefore, the efficiency of the proposed method remains to be further studied when processing thousands of images.

Acknowledgments

The research is supported by the National Key R & D Program of China (2016YFC0401600 and 2017YFC0404900), the National Natural Science Foundation of China (51979027, 51769033 and 51779035), the Fundamental Research Funds for the Central

Universities (DUT17ZD205). We thank the China Scholarship Council for sponsoring Linlin Wang's visit to University of Illinois at Urbana-Champaign, USA. And the authors extend their sincere thanks to Yasutaka Narazaki and Vedhus Hoskere to propose some comments.

References

Adhikari, R., Moselhi, O. and Bagchi, A. (2014), "Image-based retrieval of concrete crack properties for bridge inspection", *Autom. Constr.*, **39**, 180-194. https://doi.org/10.1016/j.autcon.2013.06.011

Agarwala, A., Dontcheva, M., Agrawala, M., Drucker, S., Colburn, A., Curless, B., Salesin, D. and Cohen, M. (2004), "Interactive digital photomontage", In: *ACM SIGGRAPH 2004 Papers*, August. https://doi.org/10.1145/1186562.1015718

Andreaus, U., Baragatti, P., Casini, P. and Iacoviello, D. (2017), "Experimental damage evaluation of open and fatigue cracks of multi-cracked beams by using wavelet transform of static response via image analysis", *Struct. Control Health Monitor.*, **24**(4), e1902. https://doi.org/10.1002/stc.1902

Attard, L., Debono, C. J., Valentino, G. and Di Castro, M. (2018). "Tunnel inspection using photogrammetric techniques and image processing: A review", *ISPRS J. Photogramm. Remote Sens.*, **144**, 180-188. https://doi.org/10.1016/j.isprsjprs.2018.07.010

Bradski, G. and Kaehler, A. (2008), *Learning OpenCV: Computer vision with the OpenCV library*, O'Reilly Media, Inc., Sebastopol, CA, USA.

Brown, M.A. (2005), "Multi-image matching using invariant features", Ph.D. Dissertation, University of British Columbia, Vancouver, Canada.

Brown, M. and Lowe, D.G. (2007), "Automatic panoramic image stitching using invariant features", *Int. J. Comput. Vision*, **74**(1), 59-73. https://doi.org/10.1007/s11263-006-0002-3

Canny, J. (1986), "A computational approach to edge detection", *IEEE T. Pattern Anal.*, **8**(6), 679-698. https://doi.org/10.1109/TPAMI.1986.4767851

Cao, J., Chen, L., Wang, M. and Tian, Y. (2018), "Implementing a parallel image edge detection algorithm based on the Otsu-canny operator on the Hadoop platform", *Comput. Intell. Neurosci.*, 2018. https://doi.org/10.1155/2018/3598284

Cha, Y.J., Choi, W. and Büyüköztürk, O. (2017), "Deep learning-based crack damage detection using convolutional neural networks", *Comput-Aided Civ. Inf.*, **32**(5), 361-378. https://doi.org/10.1111/mice.12263

Chen, C.Y. and Klette, R. (1999), "Image stitching—Comparisons and new techniques", *Proceedings of International Conference on Computer Analysis of Images and Patterns*, Springer, Berlin, Heidelberg, Germany, September. https://doi.org/10.1007/3-540-48375-6_73

Chia, W.C., Chew, L.W., Ang, L.M. and Seng, K.P. (2012), "Low memory image stitching and compression for WMSN using strip-based processing", *Int. J. Sens. Networks*, **11**(1), 22-32. https://doi.org/10.1504/IJSNET.2012.045037

Du, C., Yuan, J., Dong, J., Li, L., Chen, M. and Li, T. (2019), "GPU based Parallel Optimization for Real Time Panoramic Video Stitching", *Pattern Recogn. Lett.* https://doi.org/10.1016/j.patrec.2019.06.018

Fischler, M.A. and Bolles, R.C. (1981), "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography", *Commun. ACM*, **24**(6), 381-395. https://doi.org/10.1145/358669.358692

GB50010-2015 (2015), Code for design of concrete structures, Standardization administration of the people's republic of China; Beijing, China.

Graybeal, B.A., Phares, B.M., Rolander, D.D., Moore, M. and Washer, G. (2002), "Visual inspection of highway bridges", *J. Nondestruct. Eval.*, **21**(3), 67-83. https://doi.org/10.1023/a:1022508121821

Haralick, R.M., Sternberg, S.R. and Zhuang, X. (1987), "Image analysis using mathematical morphology", *IEEE T. Pattern Anal.*, (4), 532-550. https://doi.org/10.1109/TPAMI.1987.4767941

Hutchinson, T.C. and Chen, Z. (2006), "Improved image analysis for evaluating concrete damage", *J. Comput. Civil Eng. - ASCE*, **20**(3), 210-216. https://doi.org/10.1061/(ASCE)0887-3801(2006)20:3(210)

Islam, M. and Kim, J.M. (2019), "Vision-based autonomous crack detection of concrete structures using a fully convolutional encoder–decoder network", *Sens.*, **19**(19), 4251. https://doi.org/10.3390/s19194251

Jahanshahi, M.R., Masri, S.F. and Sukhatme, G.S. (2011), "Multi-image stitching and scene reconstruction for evaluating defect evolution in structures", *Struct. Health Monit.*, **10**(6), 643-657. https://doi.org/10.1177/1475921710395809

Jang, K. and An, Y.K. (2018), "Multiple crack evaluation on concrete using a line laser thermography scanning system", *Smart Struct. Syst., Int. J.*, **22**(2), 201-207. https://doi.org/10.12989/sss.2018.22.2.201

Kim, B. and Cho, S. (2019), "Image-based concrete crack assessment using mask and region-based convolutional neural network", *Struct. Control Health Monit.*, e2381. https://doi.org/10.1002/stc.2381

Kim, H., Lee, J., Ahn, E., Cho, S., Shin, M. and Sim, S.H. (2017), "Concrete crack identification using a UAV incorporating hybrid image processing", *Sens.*, **17**(9), 2052. https://doi.org/10.3390/s17092052

La, H.M., Gucunski, N., Kee, S.H. and Van Nguyen, L. (2015), "Data analysis and visualization for the bridge deck inspection and evaluation robotic system", *Visualization Eng.*, **3**(1), 6. https://doi.org/10.1186/s40327-015-0017-3

Lins, R.G. and Givigi, S.N. (2016), "Automatic crack detection and measurement based on image analysis", *IEEE T. Instrum. Meas.*, **65**(3), 583-590. https://doi.org/10.1109/TIM.2015.2509278

Liu, Y., Cho, S., Spencer Jr, B.F. and Fan, J. (2014), "Automated assessment of cracks on concrete surfaces using adaptive digital image processing", *Smart Struct. Syst., Int. J.*, **14**(4), 719-741. ttp://doi.org/10.12989/sss.2014.14.4.719

Lowe, D.G. (2004), "Distinctive image features from scale-invariant keypoints", *Int. J. Comput. Vision*, **60**(2), 91-110. http://dx.doi.org/10.1023/b:visi.0000029664.99615.94

Nishikawa, T., Yoshida, J., Sugiyama, T. and Fujino, Y. (2012), "Concrete crack detection by multiple sequential image filtering", *Comput-Aided Civ. Inf.*, **27**(1), 29-47. http://dx.doi.org/10.1111/j.1467-8667.2011.00716.x

Osman, M.K., Mashor, M.Y., Saad, Z. and Jaafar, H. (2009), "Contrast enhancement for Ziehl-Neelsen tissue slide images using linear stretching and histogram equalization technique", *Proceedings of the 2009 IEEE Symposium on Industrial Electronics & Applications*, Kuala Lumpur, Malaysia, October. https://doi.org/10.1109/ISIEA.2009.5356411

Pulli, K., Baksheev, A., Kornyakov, K. and Eruhimov, V. (2012), "Real-Time Computer Vision with OpenCV", *Commun. ACM*, **55**(6), 61-69. http://dx.doi.org/10.1145/2184319.2184337

Qu, Z., Ju, F.R., Guo, Y., Bai, L. and Chen, K. (2018), "Concrete surface crack detection with the improved pre-extraction and the second percolation processing methods", *PLoS One*, **13**(7), e0201109. https://doi.org/10.1371/journal.pone.0201109

Szeliski, R. (2006), "Image Alignment and Stitching: A Tutorial", *Foundations Trends® Comput. Graphics Vision*, **2**(1), 1-104.

http://dx.doi.org/10.1561/0600000009
Wagner, D. Mulloni, A., Langlotz, T. and Schmalstieg, D. (2010), "Real-time panoramic mapping and tracking on mobile phones", *Proceedings of the 2010 IEEE Virtual Reality Conference*, Waltham, USA, March. https://doi.org/10.1109/VR.2010.5444786
Yamaguchi, T. and Hashimoto, S. (2010), "Fast crack detection method for large-size concrete surface images using percolation-based image processing", *Mach. Vision Appl.*, **21**(5), 797-809. https://doi.org/10.1007/s00138-009-0189-8
Zhang, T. and Suen, C.Y. (1984), "A fast parallel algorithm for thinning digital patterns", *Commun. ACM*, **27**(3), 236-239. https://doi.org/10.1145/357994.358023
Zhu, Z., German, S. and Brilakis, I. (2010), "Detection of large-scale concrete columns for automated bridge inspection", *Autom. Constr.*, **19**(8), 1047-1055. https://doi.org/10.1016/j.autcon.2010.07.016

HJ