Smart Structures and Systems, *Vol. 14, No. 4 (2014) 719-741*
DOI: http://dx.doi.org/10.12989/sss.2014.14.4.719

Automated assessment of cracks on concrete surfaces using adaptive digital image processing

Yufei Liu[1], Soojin Cho[*2], Billie F. Spencer, Jr.[3] and Jiansheng Fan[1]

[1]*Key Laboratory of Civil Engineering Safety and Durability of Ministry of Education, Department of Civil Engineering, Tsinghua University, Beijing 100084, China*
[2]*School of Urban and Environmental Engineering, Ulsan National Institute of Science and Technology, Ulsan 689-798, Korea*
[3]*Department of Civil and Environmental Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA*

(Received May 30, 2014, Revised August 25, 2014, Accepted August 30, 2014)

Abstract. Monitoring surface cracks is important to ensure the health of concrete structures. However, traditional visual inspection to monitor the concrete cracks has disadvantages such as subjective inspection nature, associated time and cost, and possible danger to inspectors. To alter the visual inspection, a complete procedure for automated crack assessment based on adaptive digital image processing has been proposed in this study. Crack objects are extracted from the images using the subtraction with median filter and the local binarization using the Niblack's method. To adaptively determine the optimal window sizes for the median filter and the Niblack's method without distortion of crack object, an optimal filter size index (OFSI) is proposed. From the extracted crack objects using the optimal size of window, the crack objects are decomposed to the crack skeletons and edges, and the crack width is calculated using 4-connected normal line according to the orientation of the local skeleton line. For an image, a crack width nephogram is obtained to have an intuitive view of the crack distribution. The proposed procedure is verified from a test on a concrete reaction wall with various types of cracks. From the crack images with different crack widths and patterns, the widths of cracks in the order of submillimeters are calculated with high accuracy.

Keywords: crack width; digital image processing; adaptive image processing; window size; median filter

1. Introduction

Crack is one of the important features that need to be assessed in the monitoring of concrete structures. The observable cracks on the structure surface are caused by various factors: shrinkage, creep, corrosion of reinforcements, and the excessive loading by natural events (e.g., earthquakes, typhoons, floods, etc.). The detailed assessment of the cracks helps to identify the current conditions of the concrete structures. Currently, the visual inspection is carried out by structural specialists for the assessment of aged and possibly damaged structures. The specialists evaluate the structural integrity by looking at and manually measuring visual damage (e.g., cracks) on critical structural members.

*Corresponding author, Research Assistant Professor, E-mail: soojin@unist.ac.kr

http://www.techno-press.org/?journal=sss&subpage=8 ISSN: 1738-1584 (Print), 1738-1991 (Online)

Although the specialists are supposed to be well-trained for evaluating the safety of highly engineered environments (Prieto and Brinckerhoff 2002), the visual inspection technique has some disadvantages: subjective inspection nature, associated time and cost, and possible danger to the inspectors. Many have been studied to assess cracks using vibrations (Lu and Liu 2012, Xiang *et al*. 2012), smart materials (Kim *et al*. 2012), or NDT techniques (Shokri and Nanni 2014), which makes the inspection objective and accurate. However, the increasing usage of jumboized concrete structures and the acceleration of structural obsolescence demand alternative techniques to assess the cracks in the automated manner. Especially, the automacy is very demanding when natural hazards take place and a number of structures in a large region need to be evaluated as soon as possible. For example, it has been reported that the whole building safety evaluation in a city area took several weeks despite of the sufficient labor supplement (Johnson 2004, Chock 2007), and the automated technique is expected to reduce the inspection time, and the downtime of suspected buildings subsequently.

The digital image processing (DIP) has been actively studied as an alternative to the visual inspection, coupled with the fast development of digital imaging devices – cameras and camcorders. The DIP resembles the ocular nature of the visual inspection, and thus it has advanced by mimicking the cognitive principles of human vision (Chanda and Majumder 2004). Contrast to the human eyes with selective attention (Duncan 1984), the DIP can extract the information from the whole captured imaging area. The DIP can also be integrated into a robotic system, such as robots and quadcopters, to enable unmanned achievement of given tasks. For example, unmanned aerial vehicles are actively researched along with the implementation of the DIP (Rathinam *et al*. 2008, Torok *et al*. 2013). The unmanned system is expected to enable more frequent, safe, and objective inspection in a shorter time.

When assessing cracks using the DIP, the keys to the success are the enhancement of crack images with eliminating noises. Numerous preprocessing algorithms as well as their combinations have been introduced to unlock the keys; they are linear and non-linear spatial filters, frequency-domain filter, wavelet filter, morphological processing, and pattern recognition (Jahanshahi *et al*. 2009, Ehrig *et al*. 2011). Tanaka and Uematsu (1998) have used morphology-based techniques for detecting cracks of the pavement on concrete bridges. Abdel-Qader *et al*. (2003) have compared a wavelet method (fast Haar transform, FHT), a frequency-domain method (fast Fourier transform, FFT), and edge detection methods (Sobel and Canny edge detection methods) for the preprocessing of images to accurately assess cracks, and have resulted that edge detection methods are better in practice after the noises are properly eliminated. Yamaguchi and Hashimoto (2010) have proposed an image-based percolation model which could track continued cracks based on the connectivity of brightness and the shape of the percolated regions. Various DIP algorithms with its applications to alter the visual inspection can also be referred to Jahanshahi *et al*. (2009).

In particular, aiming at removing some complex background of the crack images, background subtraction techniques are adopted, including the subtraction with morphological opening, mean filter, Gaussian average, etc. Lee *et al.* (2013) have shown that the subtraction technique with morphological opening operation, sometimes called the Top-hat transform, is efficient to remove non-uniform backgrounds, such as smudges and shades, on the images of concrete. Fujita and Hamamoto (2011) have proposed a crack assessment method by two processing steps: subtraction with median filter to remove slight backgrounds and the Hessian line filter to emphasize cracks against blebs or stains.

Though aforementioned literatures have shown their effectiveness in finding the crack

information, they generally considered the cases when the images are in high resolution and cracks occupy many pixels in the images, or never mentioned about the image qualities. The preprocessing removes the noises as well as distorts the original objects, especially their boundaries, in the image. If the images are in low resolution (i.e., cracks occupy a few pixels), the preprocessing causing changes on the boundary pixels results in inaccurate assessment of cracks. This is very significant when using images taken distantly or videos that generally has lower resolution than still images. It is also applied when the crack is very narrow (in the order of a tenths mm) occupying a few pixels. Therefore, stable and accurate digital image processing methods, or validation procedures are urgently needed for accurate crack assessment.

In this study, an automated approach for assessment of crack on large civil structures is proposed employing the adaptive DIP techniques. The used DIP techniques are the subtraction with median filter, Niblack's binarization method (Niblack 1985), and the crack decomposition and associated width estimated approach. The subtraction with median filter is a nonlinear filter to remove the shades and smudge noise, and Niblack's binarization which is well-known with its performance on identifying line-shape objects (Trier and Jain 1995) is for both removing salt-and-pepper type noise and enhancing the crack shapes. To minimize the distortion with retained performance in noise removal, an optimal filter size index (OFSI) has been proposed to determine optimal filter size in the automated manner. A crack decomposition, and edge-based method is adopted to accurately assess the continuous cracks from the binary images. To validate the performance of the proposed approach, a validation test on a concrete reaction wall with different patterns of surface cracks is carried out. The distance between the camera lens and the wall is measured using a commercial laser distance meter, and the crack widths assessed using the proposed DIP approach are compared with those measured using a crack width gauge.

Note that all the images used in this paper are calibrated for the possible distortion of images by the used camera and lenses. To understand the image distortion, refer to the Camera Calibration Toolbox website (Bouguet 2013).

2. Adaptive digital image processing technique

In brief, the targets of techniques to pre-process crack images include two aspects: eliminate noises and enhance cracks. In this section, a two-step adaptive digital image processing technique is proposed including the subtraction with the median filter and Niblack's binarization method. Binary images with less noises and clearer crack objects are obtained from the proposed two-step approach.

2.1 Subtraction using median filter for gray-scale crack images

The median filter (Huang 1981) is an image blurring methods often employed for removing impulse noise, such as "salt (white) and pepper (black)" noise (Wang *et al*. 2010). The performance of median filter depends heavily on the chosen of filter window. Filter window refers to the pattern of neighbors where the filter runs each time. Two parameters should be set for the filter window: shape and size. The filter window for this paper is square, which is frequently the case. Therefore, the size of filter window becomes a significant parameter to have influence on the effect of the median filter. The window size will be described using the number of pixels in a size of the square window (i.e., 5-pixel window for the window sized 5×5 pixels).

Subsequently, the filtered image is subtracted from the original image. For a gray-scale image, the subtracted image can be expressed

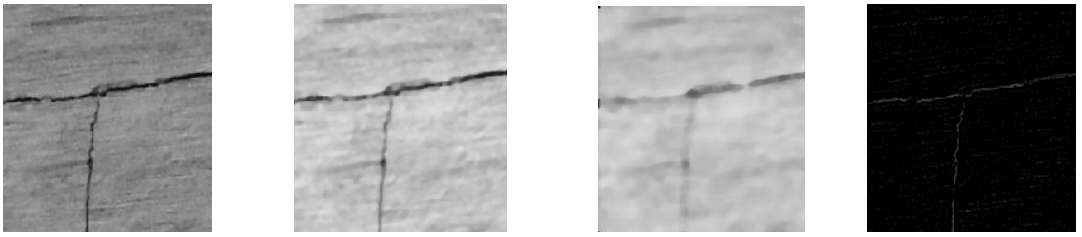
$$I_i^*(x,y) = \max\begin{cases} Median_i[I(x,y)] - I(x,y) \\ 0 \end{cases} \quad (1)$$

where I and I^* denote the original and subtracted images, respectively; x and y denote the vertical and horizontal pixel coordinates of the image, respectively; and the subscript i denotes the pixel size of the window for the median filter. The influence of subtraction includes: (i) the elimination of noise; (ii) the enhancement of the crack objects; (iii) possible changes in the crack boundary and the calculated crack width. Note that most reported research ignores this distortion to the crack width. This problem will be discussed in detail subsequently.

Blurring the image using the median filter does not change substantially the uniform distribution background noise in the image, such as large shadings; however, the crack will be significantly blurred. As a result, after the subtraction given in Eq. (1), distinct objects with sharp edges (e.g., cracks) will remain. Fig. 1 shows an example of such processing. Figs. 1(a) and 1(b) are the original image and the filtered image using median filter, respectively. Fig. 1(c) shows that using a larger window in a median filter will result in more averaging of the original image. Fig. 1(d) is the result of subtraction, which provides a uniform background with the crack clearly being shown, although some noise still remains.

When performing the subtraction, different window sizes for the median filter will result in varying effects. For example, smaller window size leads to more preservation of image details, while larger window size to more blurring and averaging. As a result, after the process of subtraction, images using smaller median filter window will have more offset of the image details.

Fig. 2 illustrates a removal of variations in the background (e.g., shadings) using the subtraction with varying window size. The intensities of the original image at the top of Fig. 2, the filtered image, and the subtracted image are plotted for small, optimal, and large window sizes. The subtraction using small window size results in heavy offset of the crack object (peak area of bold line), since the filtered image has intensity similar to the original image. When the window gets lager beyond the optimal size, crack width in the subtracted image trends to be normal while noise trends to grow. This phenomenon appears because the median filter averages the gray level too much and the subtracted image is similar to the original image. Therefore, improper window size may lead to the change of crack width or noisy result while optimal window size could obtain accurate crack width and less noise.



(a) Original image (b) Median filtered image using a 5-pixel window (c) Median filtered image using a 10-pixel window (d) Subtraction of (a) and (b)

Fig. 1 Examples of subtraction and the effect of median filter using different window sizes

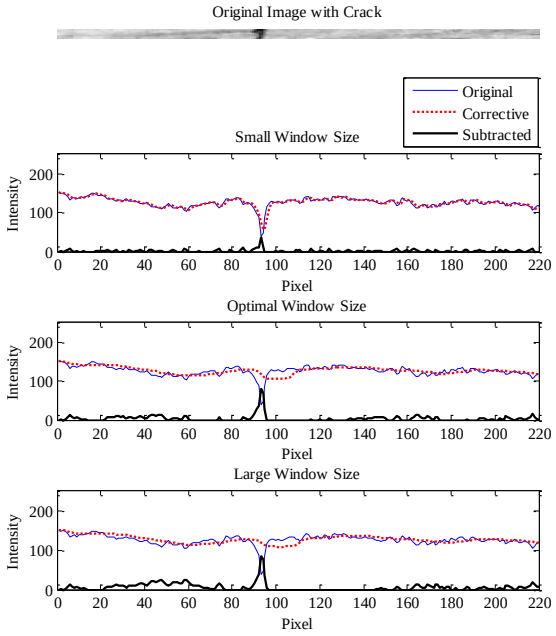


Fig. 2 Removal of variations in the background with varying window

2.2 Binarization of gray-scale images

Binarization is to convert a gray-scale image into a black and white image, which is a significant step in DIP. It aims to bisect the whole image to the crack image and the background image. In the binarization, the pixel is converted to black, representing crack object, if the intensity of a pixel in gray-scale is below a given threshold. Because the selection of threshold controls the performance of binarization, various algorithms have been proposed to find the optimal threshold.

Classified by how to determine the threshold, binarization algorithms could be divided into (i) global methods with a threshold for the whole image and (ii) local methods with different threshold for each pixel. Otsu's method (Otsu 1975) and Niblack's method (Niblack 1985) are the most effective methods in global and local method, respectively (Trier and Jain 1995). The Otsu's method has been cited as one of the most effective global techniques (Trier and Jain 1995), but it has low sensitivity on identifying objects with smaller proportion than the background (Zhang and Hu 2008). For Niblack's method, as mentioned in section 1, it works well in identifying line-shape objects. Therefore in this paper, Niblack's method is employed to identify cracks.

The main idea of Niblack's method is to obtain the threshold of every pixel from the mean and standard deviation of the intensities in a selected window around the pixel. The threshold is calculated as

$$T(x,y) = m(x,y) + k \cdot s(x,y) \tag{2}$$

where m and s are the mean and standard deviation of the intensities in the Niblack's window, and k is a correction factor varying from -0.1 to -0.2 (Kefali *et al*. 2010). Usually in a gray image such as 8-bit, white pixels are represented using "255" and black pixels are presented using "0". In binary images, white pixels are "1" and black pixels are "0". When k closes to -0.2, noises are mostly eliminated but cracks may be distorted a little; when k closes to -0.1, cracks may be unchanged but noises remain a lot (Khurshid *et al*. 2009). Usually k is selected -0.2 as a fixed parameter, as well as in this paper. Fig. 3 shows the situation of different k for a given image. The threshold is a little bit lower when k = -0.2 than k = -0.1, so less black object or noise left when k = -0.2.

Similar to the median filter, the shape and size of the window must be set first in the Niblack's method. For the Niblack's method, the window has a square shape, same as the median filter. Appropriate window should cover a local image region which contains enough information including both the target objects and the background (Peng and Hsu 2009). On the other hand, the window size shouldn't be too large for it may cause the loss of local image details. Note that Niblack's method with different window size will have different results of calculated crack width, for window sizes determine the gray scale standard deviations of the local area. For example, when the window size is too small, though more uncorrelated noises could be eliminated, the crack width of the result trends to be smaller than normal. Figs. 4 and 5 show the example of different window size for a given crack image. The image in Fig. 4(a) has a T-shape crack and the intersection of cracks has a complex shapes. Fig. 4(b) is the zoomed-in image with a 5-pixel window marked with a square. In the window, both the crack and background are not contained, and thus the crack information within the window will be lost. If an appropriate window is used as Fig. 4(c), then the crack information can be assessed properly. Fig. 5 is the binarized images using the windows shown in Figs. 4(b) and 4(c). Compared with Fig. 4(a), the crack width in Fig. 5(a) is estimated smaller, and some parts of cracks, including the intersection, are discontinuous. The Fig. 5(b) estimates cracks better in aspects of the width and the continuity. Due to some experiences (Tabatabaei and Bohlool 2010), the window size of Niblack's method should be set about 2~3 times wider than width of most appearing cracks on the crack image to get accurate binarization result. However, crack width is unknown before width calculation, and thus the window size of the Niblack's method is set to be equal to that of the median filter that will be addressed by the adaptive approach, which will be discussed in the subsequent section.

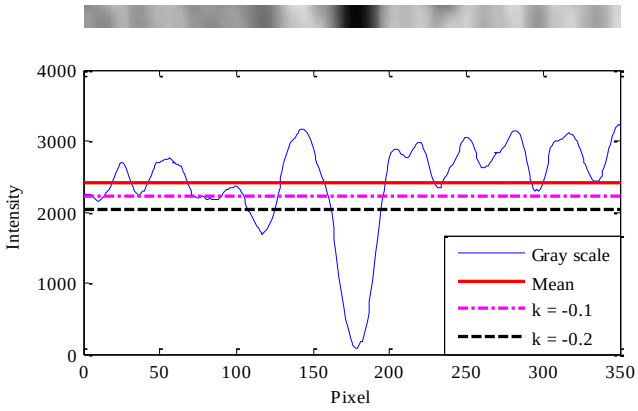
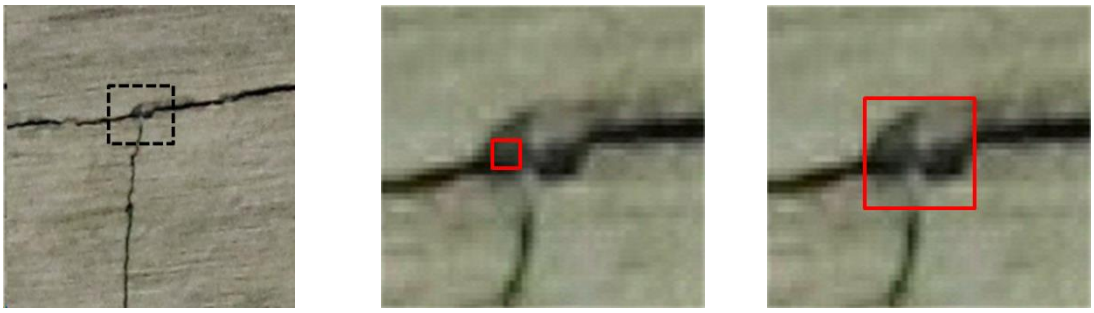
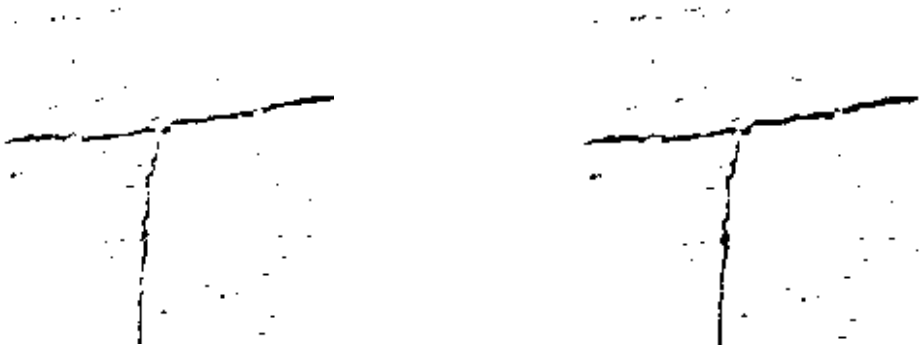


Fig. 3 Example image (top) and thresholds by Niblack's method with different k (bottom)



(a) Example Image (b) Zoomed-in image with a 5-pixel window (c) Zoom-in image with a 20-pixel window

Fig. 4 Example of appropriate window size for Niblack's method

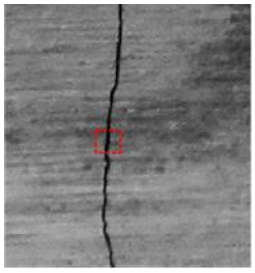


(a) Using a 5-pixel window (b) Using a 20-pixel windows

Fig. 5 Binarized images by Niblack's method

2.3 Optimal filter size index

As mentioned in section 2.1, the width of crack in the subtracted image may slightly change from their original value due to the different effects of offset. Here Fig. 6 is given to show the change. Fig. 6(a) is the original gray-scale image. Figs. 6(b) and 6(c) are the binary images by using the Niblack's method and the Otsu's method without any subtraction, respectively, that will show the effect of the subtraction. Effects of blurring and averaging when using median filter of different window sizes are shown in Figs. 6(d), 6(g), and 6(j), and corresponding results using subtraction are shown in Figs. 6(e), 6(h) and 6(k). In Fig. 6(e), when the window size is small, most of crack details are offset by subtraction. Figs. 6(f), 6(h), and 6(k) are the binarized images by the Niblack's method using the same window size and shape of the median filter. As a result, calculated crack width became smaller than accurate value for crack boundaries have been seriously offset.

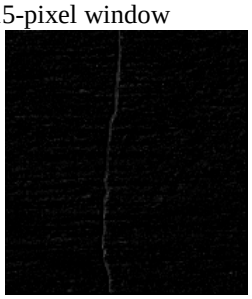


(a) Original Image in gray scale

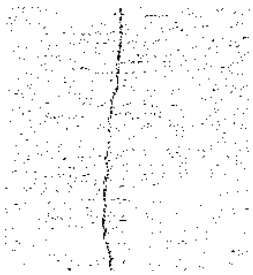
(b) Binarization of (a) using Niblack's method with a 15-pixel window

(c) Binarization of (a) using Otsu's method

(d) Using median filter of a 6-pixel window

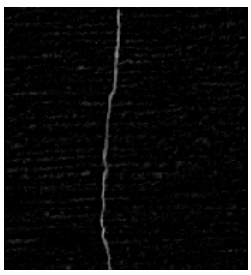


(e) Subtraction of (a) and (d)

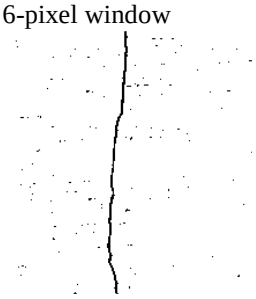


(f) Binarization of (d) using Niblack's method with a 6-pixel window

(g) Using median filter of a 12-pixel window

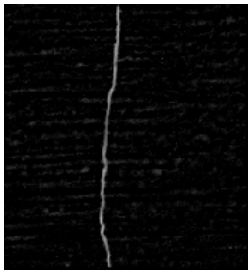


(h) Subtraction of (a) and (g)

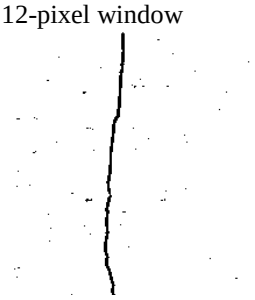


(i) Binarization of (g) using Niblack's method with a 12-pixel window

(j) Using median filter of a 18-pixel window



(k) Subtraction of (a) and (j)



(l) Binarization of (j) using Niblack's method with a 18-pixel window

Fig. 6 Example of subtraction and binarization using varying sizes of windows

When the same size of window is used for the median filter and the Niblack's method, the crack width is assessed more accurately as the window size gets larger: (i) the results of median filter (see Figs. 6(d), 6(g), and 6(j)) trend to be more and more fuzzy but the shadings and large area spots still exist; (ii) the results of subtraction (see Figs. 6(e), 6(h), and 6(k)) trend to be more and more clear; and (iii) cracks in binarization images (see Figs. 6(f), 6(i), and 6(l)) trend to be more and more normal, and noises reduce to a low level. In brief, the crack width trends to be accurate and stable after the initial stage, and the noises reduce firstly and may probably increase a little after the stable stage. Therefore, the optimum window size which produces clear image and accurate crack width appears at the beginning of stable stage.

An optimal filter size index (OFSI) is proposed to quantify the noise level and represent the possible change of crack width resulted by improper window size of median filter and binarization. For an image I whose width and height are w and h in pixels (in the image coordinate): define $N(i)$ the total pixel number of cracks as well as some noises after the subtraction and the binarization using window size of i; and define the OFSI $D_N(i)$ the normalized difference of $N(i+\Delta i)$ and $N(i)$ as

$$D_N(i) = \frac{\left|N(i+\Delta i) - N(i)\right|}{w \cdot h} \tag{3}$$

where Δi is the increment of the window size. With incremental window sizes: (i) when two neighboring processed images vary a lot, $D_N(i)$ will get a relative high value; (ii) when the two neighboring processed images appear to be similar, $D_N(i)$ will get a relative low value; and (iii) $D_N(i)$ will become stable after the stable stage begins. If $D_N(i)$ is large and unstable, it infers that the window size hasn't reach the optimal value. Note that Δi can be larger than 1 to reduce the computational time. Two illustrative examples for $D_N(i)$ are given in the following part of this section for the images with a single crack and multiple cracks.

Illustrative example (a): Image with a single crack

Crack width at the rectangular area of Fig. 6(a) is calculated under different window size of the median filter and the binarization. The real crack width is obtained from visual investigation with zooming in the rectangular area. The calculated crack widths according to the incremental window sizes are plotted in Fig. 7. Given with the real crack width 5 pixels, the calculated crack width increases as the window size gets larger, and it becomes stable at 5 pixels after the window size reaches 18 pixels. The result fits the display and comparison in Fig. 6 very well.

The OFSI $D_N(i)$ is also calculated as plotted in Fig. 8. The OFSI degrades from the maximum value and converges to be stable after the window size reaches 18 pixels. This phenomenon infers that with the incremental window sizes: (i) the identified crack width increases to accurate value and stay the same; (ii) noise reduce to low level stage. If the window size exceeds the optimal size, noise on the subtracted image may increase. Consequently, the accurate crack width with minimal noise level can be assessed by observing the curve of OFSI and taking the window size at the beginning of the stable stage. Therefore, the optimal window size is 18 pixels, which is identical to the result from Fig. 7.

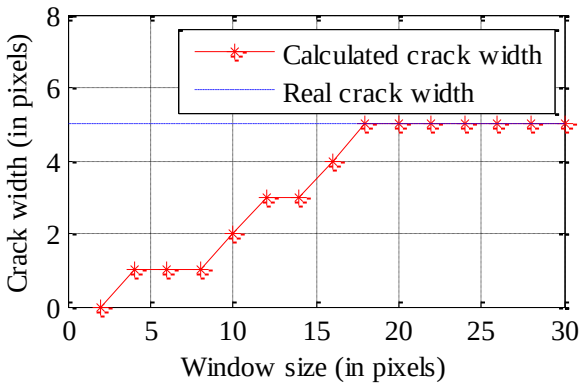


Fig. 7 Calculated crack width compared with real one

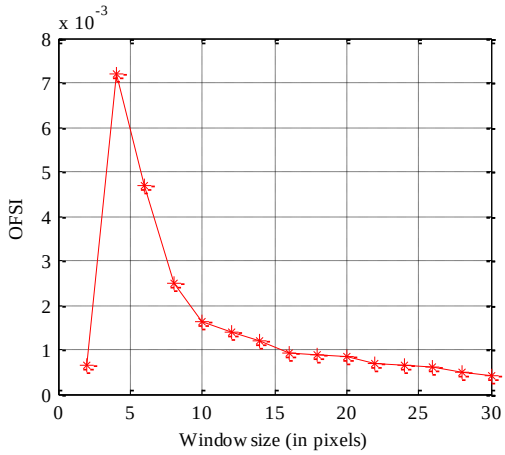


Fig. 8 OFSI plot of image with a single crack

Illustrative example (b): Image with multiple cracks

For images containing more than one crack or varying-width cracks, the proposed method is still effective, because the window size will be governed by the widest crack. Fig. 9 shows an example image that contains two cracks of different widths, whose real values are obtained from visual investigation with zooming in the rectangular areas. Fig. 10 is the calculated crack widths according to the incremental window sizes. The vertical crack is the first to be stable at around 12 pixels, and then the horizontal crack becomes stable at around 32 pixels.

The OFSI plotted in Fig. 11 shows two main peaks, which reveals the offsetting effect of crack pixels when doing subtraction with smaller window size than the optimum. The first peak at 6 pixels flattens out from 12 pixels, since the vertical crack becomes clear while the horizontal crack pixels are under the offsetting effect by the subtraction. Then the second peak at 24 pixels is

appeared as the horizontal crack becomes more and more normal, and the peak flattens out from 32 pixels and the OFSI becomes stable. Therefore, the optimal window size equals to 32 pixels with the governance of the widest crack in the window size.

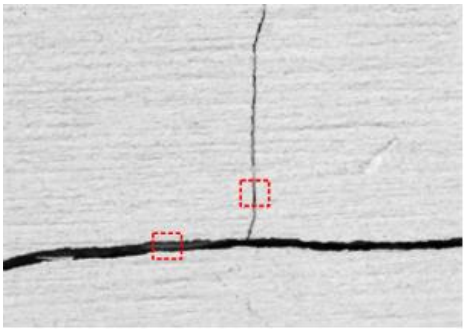


Fig. 9 Image with two cracks of different widths

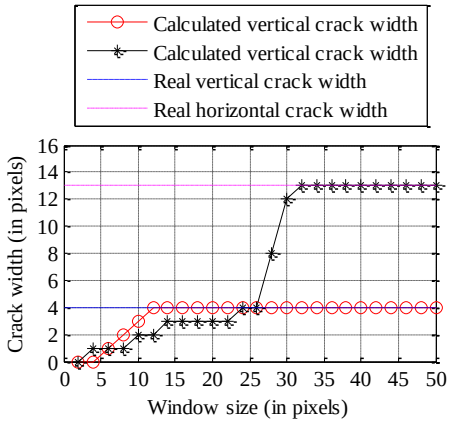


Fig. 10 Calculated crack widths compared with real ones

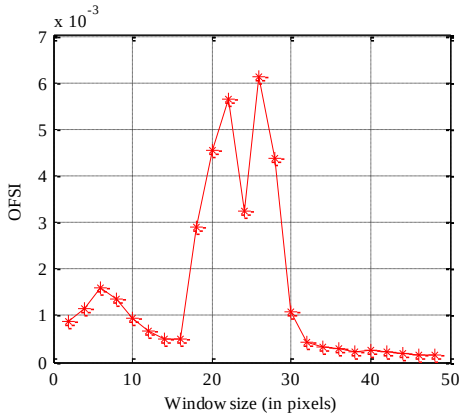


Fig. 11 OFSI plot of image with two cracks of different widths

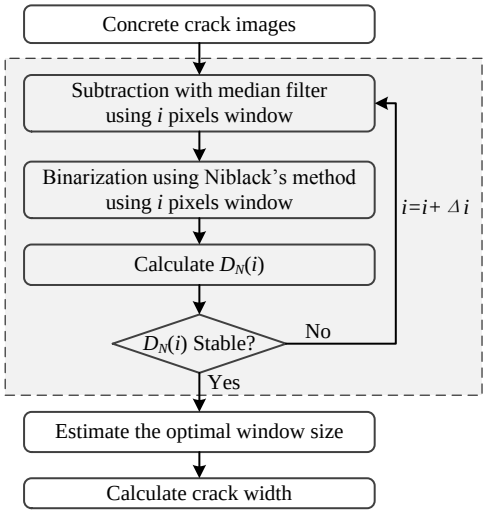


Fig. 12 Flow diagram of subtraction and binarization

Both of the two illustrative examples show that, the stable stage of the OFSI starts approximately when the window size is about 3 times of the widest crack width. This result is encouraging because binarization using Niblack's method requires the window size 2~3 times larger than crack width (see section 2.2). Therefore, the shared window size for both of the median filter and the binarization (i.e., Niblack's method) is proven to be appropriate.

In summary of this section, the process of the proposed approach combining subtraction and binarization using incremental window sizes is given in Fig. 12. In the process, whether $D_N(i)$ is stable or not could be judged using an empirical threshold which is given in section 4.2 later.

3. Algorithm for crack width calculation

In this section, a crack width calculation approach for complex crack patterns is proposed. Several researchers have reported similar methods for the crack width estimation (Lee *et al*. 2013, Nishikawa *et al*. 2012). The main difference of the proposed method from the previous researches is the way to search the intersection pixels, which will be introduced in detail in section 3.3. The proposed approach contains four main parts: (i) crack decomposition, (ii) edge detection for crack segments, (iii) calculation of crack widths in pixels, and (iv) calculation of real crack widths.

3.1 Crack decomposition

The first step of the proposed approach is to decompose the complex tree-shape or net-shape cracks into individual crack segments. To the end, a morphological operation on the binary images is performed to thin the crack objects into minimally connected lines, namely the crack skeletons (Lam *et al*. 1992). Here the term "skeleton" has been generally used to represent a pattern of a

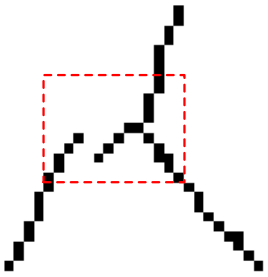
collection of thin arcs and curves. After obtaining the skeletons, very small spurs containing a few pixels on the skeletons which are resulted by the morphological operations are removed out. Figs. 13(a) and 13(b) show an example of a crack image and its skeletons, respectively. Then, the pixels of each skeleton are labelled by a unique number (from 1 to the number of total individual skeletons) as shown in Fig. 13(c), which is obtained from the dashed box of Fig. 13(b). Then, the crack graph (i.e., labelled skeleton) is decomposed into crack segments divided by the joints, which are the intersection points of three or more crack skeleton lines, as shown in Fig. 13(d).

3.2 Edge detection for crack segments

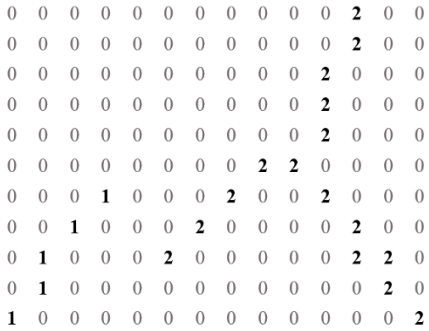
Edge detection algorithm is a classic image pattern recognition problem which has been widely developed for different purposes (Ziou and Tabbone 1998). In this paper, Canny edge detector (Canny 1986, Abdel-Qader *et al*. 2003) is adopted to detect the edge of cracks of the original image. After detecting the edges from the original image, the edges which do not correspond to the crack skeletons obtained in the previous step are considered as the small blob-shape edges, which are subsequently removed out.

In the 2D image whose pixels are aligned in a rectangular grid, the pixel connectivity can be defined in two ways: 4-connectivity and 8-connectivity. The 4-connected pixels share one of their edges with every neighboring pixel, while the 8-connected pixels share one of their edges or corners as shown in Fig. 14. The skeleton and the edges are obtained as 8-connected lines with unit pixel width.

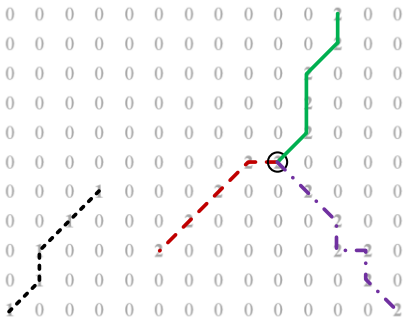
(a) Original crack image



(b) Crack skeleton



(c) Crack graph of dashed box in (b)



(d) Crack segments divided by joint point (marked with ○)

Fig. 13 Example of crack decomposition

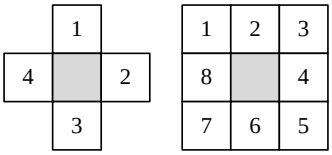


Fig. 14 4-connectivity (left) and 8-connectivity (right) in 2D image

3.3 Calculation of crack widths in pixels

Crack widths are calculated using the both of the crack segments and the corresponding edges obtained in the previous two steps. For a skeleton pixel where the width should be calculated, the crack width is calculated by following steps: (i) Find the two neighboring pixels in the skeleton; (ii) calculate the orientation of the pixel using two neighboring pixels in the skeleton, (iii) draw a normal line of the pixel according to the calculated orientation, (iv) find the intersection pixels with the both edges; and (v) calculate the pixel distance between two intersection pixels.

A search algorithm for the intersection pixels is proposed based on the understanding of the pixel connectivity. For a skeleton pixel, the orientation can be calculated by drawing a line connecting two neighboring pixels on the skeleton line, assuming the line approximately represents the tangent of the local skeleton line at the skeleton pixel. Then the possible orientations are 0°, 26.6°, 45°, 63.4°, 90°, 116.6°, 135°, and 153.4° as exampled in Fig. 15.

Subsequently, the intersection pixels could be obtained by drawing the normal line under such orientations until meeting the edges. Note that because the edges are 8-connectivity, the intersection of normal line and edge line may miss hitting the pixel points of edge line, if the 8-connected normal line is drawn as shown in Fig. 17(a). To address this problem, the 4-connected normal line is proposed as shown in Fig. 17(b). The 4-connected normal line guarantees to find two intersections on the edge lines. Examples of 4-connected normal lines when the skeleton has 26.6°, 63.4°, and 90°orientations are presented in in Fig. 17.

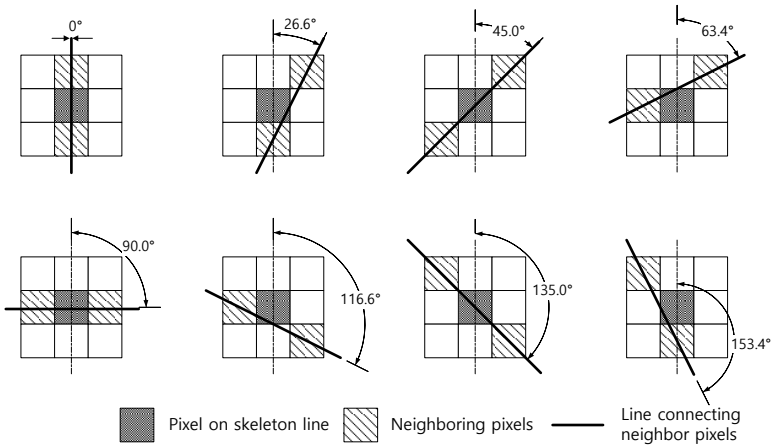
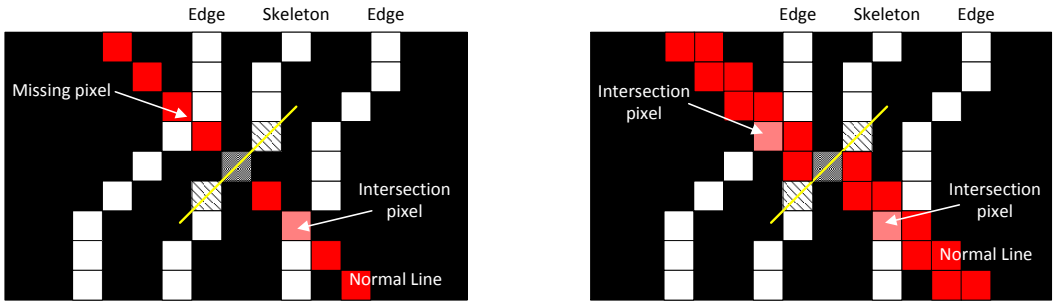


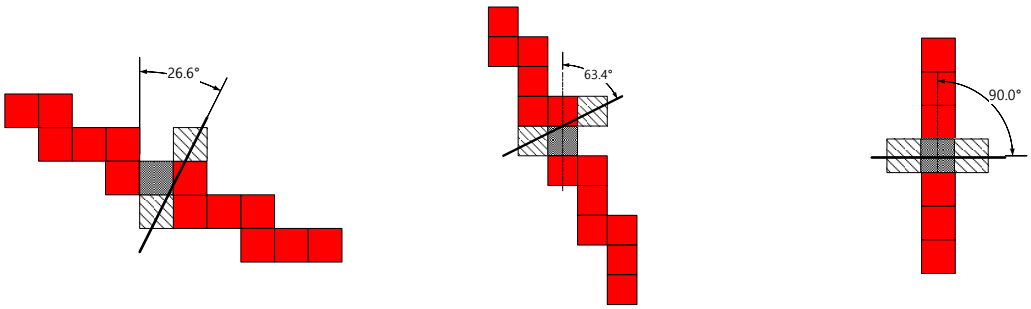
Fig. 15 Examples of eight orientations of local skeleton line

An example of finding the intersection pixels of skeleton's normal line and edge lines is given in Fig. 18, where the pixels on the skeleton line to draw the normal lines are selected every five pixels. In the proposed approach, the normal line could be drawn from every pixel on the skeleton line, or some longer intervals if concerning about the computational time.



(a) 8-connected normal line (b) 4-connected normal line

Fig. 16 Normal line of skeleton pixel



(a) 26.6° orientation (a) 63.4° orientation (a) 90.0° orientation

Fig. 17 4-connected normal lines for example orientations

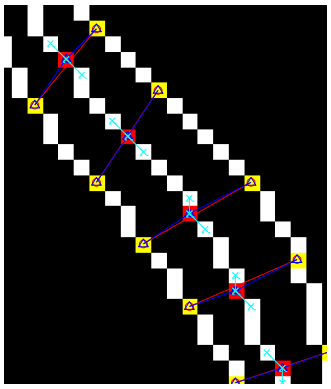


Fig. 18 Example of finding the intersection pixels

3.4 Calculation of real crack widths

With the crack widths in pixels known, the real crack width W_c can be calculated using the camera pin-hole model shown in Fig. 19 as

$$W_r = D_p W_p = \frac{D_w}{10 P_c L_f} W_p \tag{4}$$

where W_p is the crack width in pixels obtained in the section 3.3; P_c is the ppcm (pixel per centimeter) of the used camera sensor (e.g., CMOS and CCD); D_w is the working distance (i.e., the distance between concrete surface and camera sensor) in mm; and L_f is the focal length of the camera in mm; D_p is the resolution of the imaging system that describes that under such pinhole model, how long distance could be represented by one pixel on the image. Note that the primary optical axis of the camera should be vertical to the concrete surface.

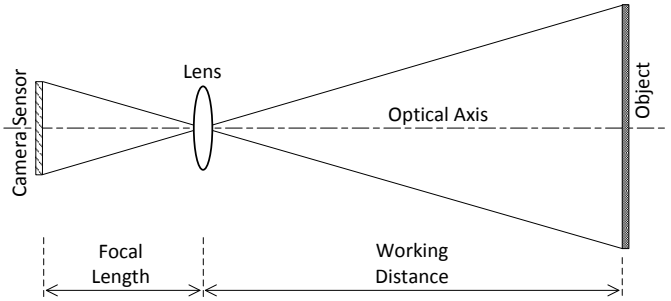


Fig. 19 Camera pinhole model

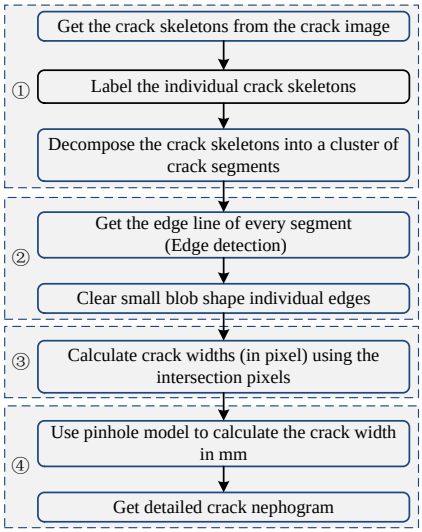


Fig. 20 Flow diagram of proposed crack width calculation method

The flow diagram for the proposed crack width calculation process discussed in this section is given in Fig. 20. By the proposed process, the crack widths can be calculated for the crack skeleton pixels. Then, the crack width nephogram can be obtained to display a whole view of the crack distribution and crack width information, as to be presented in section 4.

4. Validation tests on a concrete wall with cracks

To validate the proposed crack assessment method, experiments were conducted on the reaction wall with various types of cracks. The wall locates in the Newmark Structural Engineering Laboratory at the University of Illinois at Urbana-Champaign. This wall possesses lots of cracks caused by loading and shrinkage. An area shown in Fig. 21 was selected to be inspected. A Canon EOS 60D DSLR camera whose focal length is 55 mm was employed to take images of cracks. The CMOS of the Canon EOS 60D camera contains 5184(pixel)×3456(pixel) with the dimension of 22.3(mm)×14.9(mm). Therefore, 1 mm of the CMOS has about 232 pixels on the image (i.e., P_c = 2320). The working distances were measured using Bosch DLR130 laser displacement meter whose resolution is 1.5 mm.

4.1 Test images

Among various images taken against the inspected area, five images with vertical, horizontal, and combined cracks were selected as Fig. 22. The images were taken under different conditions (e.g., crack width, length, direction, background shadow, working distances, etc.). Note that Figs. 22(c) and 22(e) were taken for the same cracks shown in Figs. 22(b) and 22(d), respectively, with shorter working distances. The actual crack widths, which will be used as references, were measured using a conventional crack width gauge at the circled points in Fig. 22. The measured cracks have very narrow width less than 0.6 mm. The working distances and measured crack widths using crack gauge are tabulated in Table 1 in section 4.2.

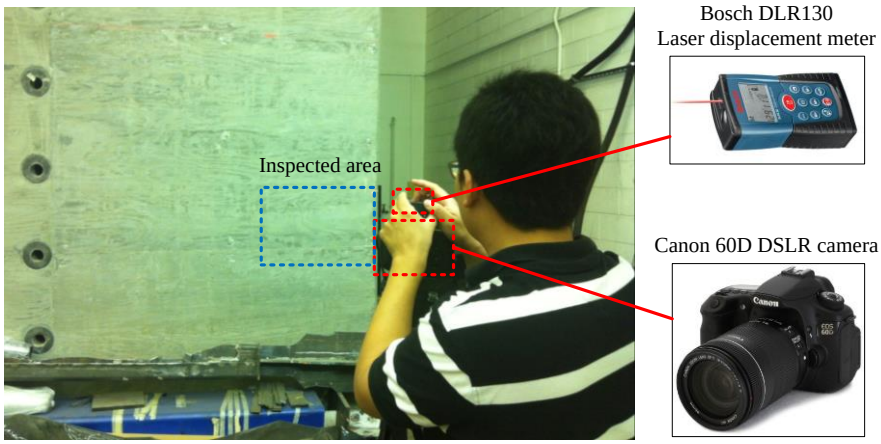
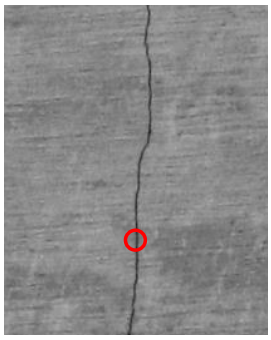
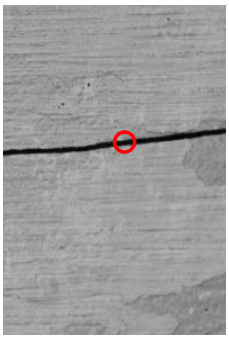


Fig. 21 Experiment setup

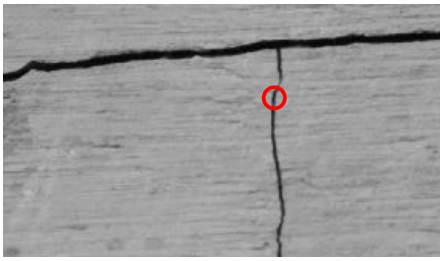


(a) A: image with vertical crack

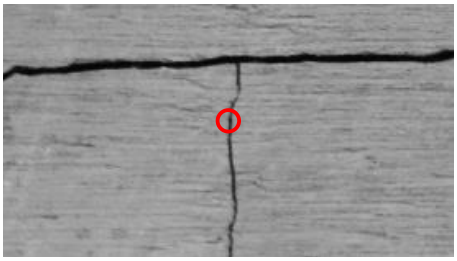
(b) B1: image with horizontal crack



(c) B2: same as B1 with shorter working distance



(d) C1: image with combined cracks



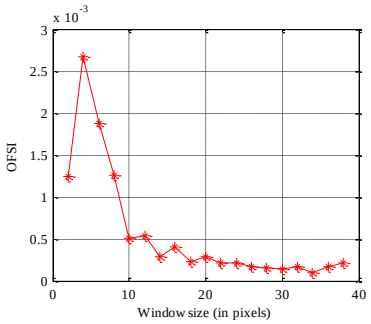
(e) C2: same as C1 with shorter working distance

Fig. 22 Gray-scale crack images used for crack width calculation

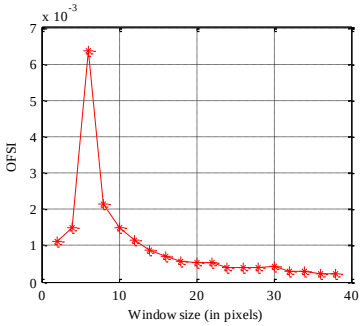
4.2 OFSI graphs

The OFSI graphs were obtained for the five images in Fig. 22 as shown in Fig. 23. The obtained OFSI graphs show similar trends to the graphs exampled in section 2.3, with the stable stages as the window size increases. The OFSI graphs show single peak patterns for the images A, B1, and B2 that contain cracks with nearly identical widths, while double peak patterns for the images C1 and C2 that contain combined cracks with 0different widths. From the OFSI graphs, the optimal window sizes for the five images were obtained at the beginning of the stable stage: they were 26, 16, 40, 38, 66 pixels, respectively.

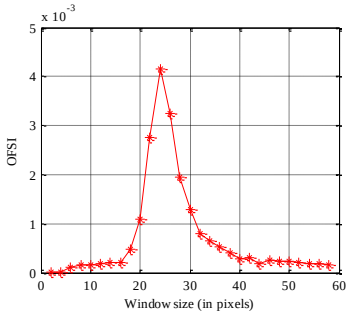
Two empirical conclusions could be obtained from the OFSI graphs: (i) Usually the OFSI graph will not become stable until it's less than 0.001; and (ii) The number of peaks of represents the number of cracks with distinct crack widths. As to the first conclusion, all the optimal window sizes were determined when the OFSI values less than 0.001 in Fig. 23. As to the second conclusion, the images A, B1, and B2 have only one main peak in the OFSI graph while the images C1 and C2 have two main peaks. For the images C1 and C2, the first peak is resulted in by the narrower crack, and the second peak is by the wider crack. Therefore, these two empirical conclusions could be applied to judge the beginning of stable stage, and to qualitatively learn the condition of crack distribution.



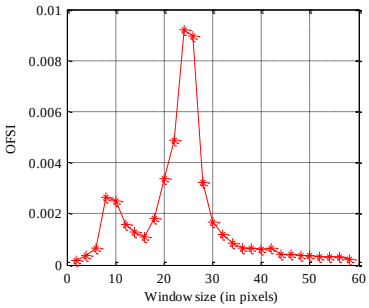
(a) A: image with vertical crack



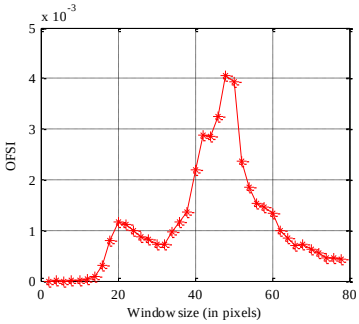
(b) B1: image with horizontal crack



(c) B2: same as B1 with shorter working distance



(d) C1: image with combined cracks



(e) C2: same as C1 with shorter working distance

Fig. 23 OFSI graphs for five example images

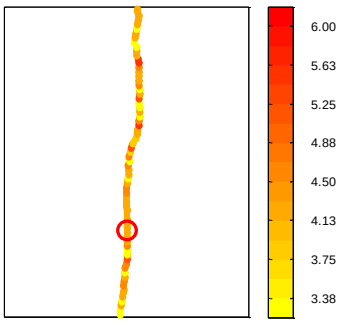
4.3 Crack width calculation

The five images were processed using the window size obtained by the OFSI, and the crack width nephograms of the five images were obtained as Fig. 24. The changing colors indicate the crack widths in pixels calculated for every five skeleton pixels, and the color bars are annexed to the figures to show the mapping of crack widths to color. For the five images, the crack widths were identified in the ranges of [3, 6], [3, 5], [12, 19], [4, 17], [3, 29] pixels, respectively, which means the shorter working distance resulted in the crack distribution with higher resolution. From the visual comparison with the original images shown in Fig. 22, the nephograms seem to have good agreement with the real crack width distribution.

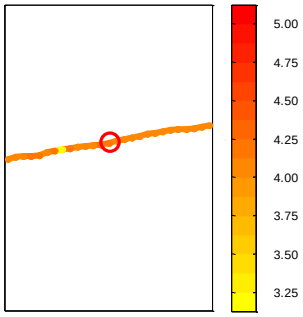
The crack widths at the circled points in Fig. 24 were calculated using the proposed approach as Table 1. In the case of images A, B1, and C1 with relatively further working distances, the crack widths in pixels are only 4, 5, and 6 pixels, and the relative errors are more than 10%. However, if the working distances become shorter as Images B2 and C2, more pixels are found for the crack widths and the errors are significantly reduced to less than 5%.

To make a quantitative study of calculation error, the resolutions of the imaging system D_p in Eq. (4) is also tabulated in Table 1. Since the D_p is the hardware error, the calculation errors smaller than D_p values are ignorable. In Table 1, the errors of the images B1, B2, C1, and C2 are smaller than the D_p values, which represent the accuracy of the proposed approach regardless of

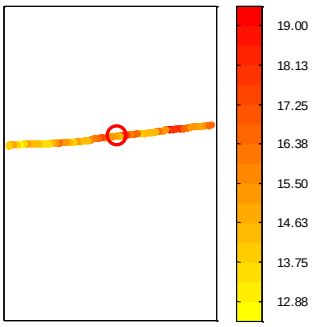
the working distances. Meanwhile, the image A1 has error a slightly larger than the D_p value, which is still acceptable considering very narrow crack width (i.e., 0.3 mm) and relatively long working distance. If many numbers of crack pixels can be identified by taking an image at the short working distance, the crack widths will be assessed with higher accuracy in essense.



(a) A: image with vertical crack



(b) B1: image with horizontal crack



(c) B2: same as B1 with shorter working distance

(d) C1: image with combined cracks

(e) C2: same as C1 with shorter working distance

Fig. 24 Crack width nephograms of five example images

Table 1 Calculated crack widths compared with measured ones

Image	A	B1	B2	C1	C2
Working distance	1384.3 mm	1733.55 mm	533.4 mm	596.9 mm	373 mm
Crack width	4 pixels	5 pixels	15 pixels	6 pixels	9 pixels
Calculated (=C)	0.43 mm	0.54 mm	0.63 mm	0.28 mm	0.26 mm
Measured (=M)	0.3 mm	0.6 mm	0.6 mm	0.25 mm	0.25 mm
D_p (mm/pixel)	0.11	0.11	0.04	0.05	0.03
Error (E=C-M)	0.13 mm	-0.06 mm	0.03 mm	0.03 mm	0.01 mm
Relative Error(R=E/M)	43.33%	10.00%	5.00%	12.00%	4.00%

5. Conclusions

This paper proposed a complete digital image processing (DIP) procedure for automated concrete crack assessment including crack object extraction, crack width calculation and display of crack distribution. To extract the crack object from the original image, the subtraction with median filter and local binarization using Niblack's method are combined together. An optimal filter size index (OFSI) is proposed to adaptively determine the optimal window sizes for the median filter and the Niblack's method, which could keep crack undistorted while eliminating the background shadings and noises. From the extracted crack object, the crack skeleton and edges are decomposed. For a skeleton pixel, the edge pixels to calculate the crack width are obtained using the proposed 4-connected normal line according to the orientation of the local skeleton line. The crack width nephogram is given at last, which could provide an intuitive view of crack distribution and information of crack widths.

To test and verify the proposed crack calculation procedure, a test on a reaction wall were conducted by taking five images with different crack patterns and working distances. The respective optimal window sizes are obtained for the five images using the shape of the OFSI graphs. For the five images in the test: (i) the crack objects were extracted very clearly using the subtraction and binarization techniques with the window sizes determined from the OFSI graphs; (ii) the crack width nephograms, which are visually similar to the original images but containing the quantitative crack distributions, were obtained; and (iii) the proposed approach provided the accurate crack widths for the five images, compared with the real ones measured using a crack width gauge, with ignorable errors nearly within the hardware error range. This work is supposed to be extended to an automated crack assessment system using an unmanned aerial vehicle equipped with imaging devices.

Acknowledgements

This research was supported by the Natural Science Foundation of China (No. 51222810). Yufei Liu was supported by the China Scholarship Council (No. 201206210142) for 1 year's study at the University of Illinois at Urbana-Champaign. The authors express their sincere appreciation to their supports.

References

Abdel-Qader, I., Abudayyeh, O. and Kelly, M.E. (2003), "Analysis of edge-detection techniques for crack identification in bridges", *J. Comput. Civil Eng. - ASCE*, **17**(4), 255-263.

Boughet, J.Y. (2013), Website: *Camera Calibration Toolbox for Matlab*: available at: http://www.vision.caltech.edu/bouguetj/calib_doc/.

Canny, J. (1986), "A computational approach to edge", *IEEE T. Pattern Anal.*, **8**(6), 679-698.

Chanda, B. and Majumder, D.D. (2004), *Digital Image Processing and Analysis*, PHI Learning Pvt. Ltd., New Delhi, India.

Chock, G. (2007), *ATC-20 Post-Earthquake Building Safety Evaluations Performed after the October 15, 2006 Hawaii Earthquakes Summary and Recommendations for Improvements (updated).* available at: https://www.eeri.org/wp-content/uploads/United%20States/SEAOH_EarthquakeSafetyEvalRec.pdf

Duncan, J. (1984), "Selective attention and the organization of visual information", *J. Exp. Psychol. Gen.*,

113(4), 501-517.
Ehrig, K., Goebbels, J., Meinel, D., Paetsch, O., Prohaska, S. and Zobel, V. (2011), "Comparison of crack detection methods for analyzing damage processes in concrete with computed tomography", *Proceedings of the International Symposium on Digital Industrial Radiology and Computed Tomography*, Berlin, Germany, June.
Fujita, Y. and Hamamoto, Y. (2011), "A robust automatic crack detection method from noisy concrete surfaces", *Mach. Vision Appl.*, **22**(2), 245-254.
Huang, T.S. (1981), *Two-dimensional Digital Signal Processing II: Transforms and Median Filters*, Springer-Verlag New York, Inc, New-York, USA.
Jahanshahi, M.R., Kelly, J.S., Masri, S.F. and Sukhatme, G.S. (2009), "A survey and evaluation of promising approaches for automatic image-based defect detection of bridge structures", *Struct. Infrasstruct. E.*, **5**(6), 455-486.
Johnson, K. (2004), *San Simeon Earthquake, City of Paso Robles Emergency Response Report*, available at: http://www. prcity. com/government/pdf/EQResponseRpt. pdf.
Kefali, A., Sari, T. and Sellami, M. (2010), "Evaluation of several binarization techniques for old Arabic documents images", *Proceedings of the 1st International Symposium on Modeling and Implementing Complex Systems*, Constantine, Algeria, May.
Khurshid, K., Siddiqi, I., Faure, C. and Vincent, N. (2009), "Comparison of Niblack inspired binarization methods for ancient documents", *IS&T/SPIE Electronic Imaging, International Society for Optics and Photonics*, San Francisco, CA, USA.
Kim, W.J., Lee, J.M., Kim, J.S. and Lee, C.J. (2012), "Measuring high speed crack propagation in concrete fracture test using mechanoluminescent material", *Smart Struct. Syst.*, **10**(6), 547-555.
Lam, L., Lee, S.W. and Suen, C.Y. (1992), "Thinning methodologies-a comprehensive survey", *IEEE T. Pattern Anal.*, **14**(9), 869-885.
Lee, B.Y., Kim, Y.Y., Yi, S.T. and Kim, J.K. (2013), "Automated image processing technique for detecting and analysing concrete surface cracks", *Struct. Infrastruct. E.*, **9**(6), 567-577.
Lu, Z.R. and Liu, J.K. (2012), "Vibration analysis of a cracked beam with axial force and crack identification", *Smart Struct. Syst.*, **9**(4), 355-371.
Niblack, W. (1985), *An introduction to digital image processing*, Strandberg Publishing Company, København, Denmark.
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.
Otsu, N. (1975), "A threshold selection method from gray-level histograms", *Automatica*, **11(**285-296), 23-27.
Peng, J.Y. and Hsu, C.N. (2009), *Adaptive local thresholding for fluorescence cell micrographs*, Inst. of Info. Sci., Academia Sinica, Tech. Rep. TR-IIS-09-008.
Prieto, R. and Brinckerhoff, P. (2002), *The 3Rs: Lessons Learned from September 11th*, Royal Academy of Engineering, London, UK.
Rathinam, S., Kim, Z.W. and Sengupta, R. (2008), "Vision-based monitoring of locally linear structures using an unmanned aerial vehicle 1", *J. Infrastruct. Syst. - ASCE*, **14**(1), 52-63.
Shokri, T. and Nanni, A. (2014), "Crack source location by acoustic emission monitoring method in RC strips during in-situ load test", *Smart Struct. Syst.*, **13**(1), 155-171.
Tabatabaei, S.A. and Bohlool, M. (2010), "A novel method for binarization of badly illuminated document images", *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP)*, Hong Kong, China, September.
Tanaka, N. and Uematsu, K. (1998), "A crack detection method in road surface images using morphology", *IAPR Workshop on Machine Vision Applications*, Makuhari, Chiba, Japan.
Torok, M.M., Golparvar-Fard, M. and Kochersberger, K.B. (2013), "Image-based automated 3D crack detection for post-disaster building assessment", *J. Comput. Civil Eng. - ASCE*, doi:10.1061/(ASCE)CP. 1943-5487.0000334.
Trier, O.D. and Jain, A.K. (1995), "Goal-directed evaluation of binarization methods", *IEEE T. Pattern Anal.*,

17(12), 1191-1201.
Wang, C., Chen, T. and Qu, Z. (2010), "A novel improved median filter for salt-and-pepper noise from highly corrupted images", *Proceedings of the 3rd IEEE International Symposium on Systems and Control in Aeronautics and Astronautics (ISSCAA)*, Harbin, China, June.
Xiang, J., Matsumoto, T., Long, J., Wang, Y. and Jiang, Z. (2012), "A simple method to detect cracks in beam-like structures", *Smart Struct. Syst.*, **9**(4), 335-353.
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.
Zhang, J. and Hu, J. (2008), "Image segmentation based on 2D Otsu method with histogram analysis", *Proceedings of the 2008 International IEEE Conference on Computer Science and Software Engineering*, Wuhan, Hubei, China, December.
Ziou, D. and Tabbone, S. (1998), "Edge detection techniques-an overview", *Int. J. Pattern Recogn*., **8,** 537-559.

CY