Paper

Image-Based Crack Detection for Real Concrete Surfaces

Tomoyuki Yamaguchi*a, Non-member
Shingo Nakamura*, Non-member
Ryo Saegusa**, Non-member
Shuji Hashimoto*, Member

In this paper, we introduce a novel image-based approach to detect cracks in concrete surfaces. Crack detection is important for the inspection, diagnosis, and maintenance of concrete structures. However, conventional image-based approaches cannot achieve precise detection since the image of the concrete surface contains various types of noise due to different causes such as concrete blebs, stain, insufficient contrast, and shading. In order to detect the cracks with high fidelity, we assume that they are composed of thin interconnected textures and propose an image-based percolation model that extracts a continuous texture by referring to the connectivity of brightness and the shape of the percolated region, depending on the length criterion of the scalable local image processing techniques. Additionally, noise reduction based on the percolation model is proposed. We evaluated the validity of the proposed technique by using precision recall and receiver operating characteristic (ROC) analysis by means of some experiments with actual concrete surface images. © 2007 Institute of Electrical Engineers of Japan. Published by John Wiley & Sons, Inc.

Keywords: crack detection, percolation, noise reduction, real concrete surface

1. Introduction

Visual inspection has become increasingly important in civil and construction engineering. It is useful for the nondestructive testing and maintenance of architectural structures. Inspecting such structures in the early stages of their degradation is critical to their maintenance, since their damage induces further degradation under prolonged exposure to severe environments. The degradation of concrete—a commonly used building material—is caused by a variety of factors such as earthquakes, frost damage, salt erosion, rain water, and dry shrinkage. Cracks on the concrete surface are one of the earliest indications of degradation. The most popular method for crack inspection is to manually prepare a detailed sketch of the cracks and to—simultaneously measure the condition of the concrete. However, the manual approaches strongly depend on the specialist's knowledge and experience and lack objectivity in the quantitative analysis. Therefore, automatic image-based crack detection is proposed as an alternative to manually drawn sketches.

Recently, some methods for crack detection by means of visual inspection have been proposed [1–3]. Abdel-Qader *et al.* suggested a comparison of the effectiveness of crack detection in the images of a bridge surface by using the wavelet transform, Fourier transform, Sobel filter, and Canny filter [4]. They concluded that the wavelet transform is significantly more reliable than the other methods. Hutchinson *et al.* used a Canny filter and the wavelet transform for crack detection and estimated the parameters using the receiver operating characteristic (ROC) analysis [5]. We have also proposed an automatic visual inspection system using images captured by a digital camera [6,7]. This system can extract and analyze cracks on the concrete surface by combining several image processing techniques including the wavelet transform, shading correction, and binarization. Kawamura *et al.* proposed a method using a genetic algorithm for the semiautomatic optimization of image processing parameters for precise crack detection [8]. However, these methods do not consider the essential characteristics of cracks such as its connectivity. Also, these methods use global image processing methods such as the wavelet transform by focusing on the characteristics of the entire image.

On the other hand, some approaches employ local image processing for crack detection. Local image processing is necessary to extract such typical local characteristics of cracks as the direction and connectivity. Roli proposed a method utilizing conditional texture anisotropy for crack detection in granite slabs [9]. This method uses the orientation feature in the local window. Fujita *et al.* proposed two preprocessing methods using the subtraction method and the Hessian matrix [10]. Since the local window is fixed, these methods cannot be flexibly applied to different widths. Also, Miwa *et al.* used watershed segmentation to detect crack lines on a tunnel [11]. However, this method does not detect cracks with high precision since it mainly focuses on the watershed for the region of separation.

a Correspondence to: Tomoyuki Yamaguchi.
E-mail: yamaguchi@aoni.waseda.jp
* Department of Applied Physics, Waseda University, 3-4-1 Okubo, Shinjuku-ku, Tokyo 169-8555, Japan
** Department of Robotics Behavior and Cognitive Science, the Italian Institute of Technology, Via Morego 30, 16163 Genova, Italy

The images of real concrete surfaces contain some noises such as concrete blebs, stains, and shadings in several sizes. The crack is difficult to distinguish from the image with noises by the conventional methods which do not use the characteristics of cracks. Moreover, the methods whose window size is fixed are inadequate to extract accurate cracks, because the length and width of cracks are different on real concrete surfaces. Therefore, the above-mentioned conventional methods tend to miss cracks while they tend to regard noise as cracks.

In our previous study, we proposed an image processing technique based on the percolation model, which considers the relationship with a neighbor for crack detection [12]. This method is a type of scalable local processing method that considers the connectivity of brightness among neighboring pixels. This method was demonstrated to be more accurate than the conventional methods. However, when the contrast in the crack region is unclear as cracks, the previous method cannot accurately detect cracks since it is only based on the brightness information during the percolation process. It is important in the inspection to detect cracks without missing them (i.e. accuracy of crack detection), because inspection is done on the basis of the detected cracks.

In this study, we describe an appropriate crack detection method that includes an improved percolation processing technique based on the shapes and brightness with noise reduction and binarization. The improved percolation model can realize robust and highly accurate crack detection by introducing the circularity of the percolated region as a sort of shape information. Additionally, the percolation processing technique uses the length criterion for scalable local processing. The performance of the proposed method is investigated by performing precision-recall and ROC analysis using actual concrete surface images.

In Section 2, we describe the characteristics of cracks and the procedures in our method. A summary of percolation processing is described in Section 3. The proposed crack detection method including the improved percolation processing and noise reduction techniques is described in Section 4. Finally, in Section 5, we show experiments using actual concrete surface images, following which precision–recall and ROC analysis are performed to evaluate the proposed method. The conclusions are provided in Section 6.

2. Overview

2.1. Definition of a crack Figure 1 shows examples of cracks and noises on a concrete surface. Cracks appear in the area represented by the dashed line. The noise patterns are indicated by rectangular areas. In this study, we assume that the cracks possess the following two characteristics: (i) their shape is thinner than those of other textural patterns and (ii) their brightness is lower than that of the background.

Cracks with dark colors are easily detected, while cracks with unclear colors are difficult to detect since their brightness is similar to that of the background. Shape information is extremely effective for detecting the unclear cracks.

2.2. Crack detection procedure We focus on the two above-mentioned characteristics of cracks. In our previous percolation model, the central pixel in a local window is
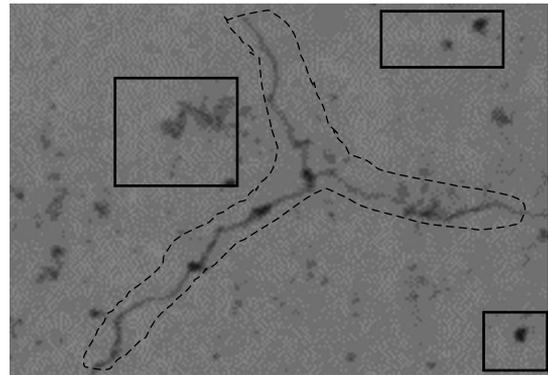

Fig. 1 Cracks and similar noises.
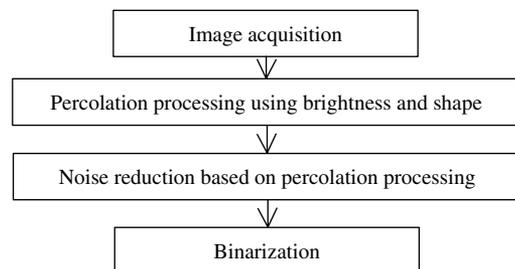(dashed line: cracked part; black rectangle: noisy part)

Fig. 2 Flowchart of the crack detection approach

evaluated according to a cluster formed by the percolation process using only the brightness criterion, which enables scalable window processing. In this study, we improve the percolation model by adding the shape criterion to the percolation process and we consider the length of the cracks to represent thin shapes. In a scalable window processing technique, the length criterion enables the detection of fine cracks even when the contrast of the crack region is unclear as cracks. Moreover, in order to reduce the noise sensitivity, we applied noise reduction, which is based on the shape of the percolated cluster. Finally, binarization is applied to the images after noise reduction. Figure 2 shows the proposed crack detection procedure.

3. Summary of Percolation Processing

Percolation [13,14] is a physical model based on the natural phenomenon of liquid permeation. This model is effective for describing various phenomena such as the spread of epidemics, fires in orchards, ferromagnetism, and disordered electrical networks.

In our previous study, we employed scalable local processing based on the percolation model to detect cracks [12]. The feature of this method is that it evaluates the central pixel in a local window (i.e. the focal pixel) according to a cluster formed using the percolation process. In our method, a variety of percolation processing tasks can be realized by arranging the cluster classification criterion. Scalable window processing is represented by changing the termination condition of the percolation process. For the brightness values, our method uses 256-level grayscale images. We set black to 0 and white to 255 for representation. The percolation process is described below.

1. At first, the size of the initial window is fixed as $N \times N$ and the maximum window size is defined as $M \times M$. The pixel located at the center of the local window is set as the initial pixel p_s for percolation and included in the percolated region D_p. Further, the percolation threshold T is set to the value of the initial pixel brightness $I(p_s)$.

2. The threshold T is updated as follows:

$$T = \max(\max_{p \in D_p}(I(p)), T) + w \tag{1}$$

   where w is an acceleration parameter used to accelerate the percolation.

3. The eight neighboring regions of D_p are defined as the candidate region D_c. In D_c, the pixels whose brightness is lower than the threshold T are percolated and included in D_p. If there are no such pixels, the darkest pixel in D_c is included in D_p.

4. When D_p reaches the boundary of the $N \times N$ window, the percolation process proceeds to step 5 and N is incremented to $N + 2$. Otherwise, the process goes back to step 2.

5. The threshold T is updated in the same manner as in step 2.

6. In the neighboring regions D_c of D_p, the pixels whose brightness is lower than T are included in D_p. If there are no such pixels, the percolation process is terminated. Otherwise, N is incremented to $N + 2$.

7. If N is larger than the maximum window M, the process is terminated. Otherwise, the process goes back to step 5.

Conclusively, we obtain the final D_p as the resultant region of the percolation process, as shown in Fig. 3. Then, we can evaluate the focal pixel to determine whether it belongs to a crack by characterizing D_p. We estimate the circularity F_c as a characteristic of D_p, and it is expressed by the following equation:

$$F_c = \frac{4 \cdot C_{count}}{\pi \cdot C_{max}^2} \tag{2}$$

where C_{count} is the number of pixels in D_p and C_{max} is the maximum length of D_p. The F_c value ranges from 0 to 1. For example, the F_c value of the image is close to 1 when the shape of D_p is nearly circular, as shown in Fig. 3(a). On the other hand, the F_c value of the image is close to 0 when the shape of D_p as a crack is linear, and it is completely different from a circle, as shown in Fig. 3(b).

Consequently, the brightness of the focal pixel in the output image is associated with the F_c value by setting it to $F_c \times 255$. Percolation processing is carried out for every pixel in the input image. We distinguish whether the pixels are included in the cracks or not by using the value of $F_c \times 255$.

4. Proposed Crack Detection Method

The previous percolation processing is a difficult problem in detecting the unclear cracks accurately. In the literature [12], we observed that this algorithm requires improvements to make the parameter w scalable. When the contrast in the cracked region is very low (i.e. unclear), this parameter is often too large to form an accurate crack cluster as the percolation is progressing, since the previous percolation model depended only on the brightness. Therefore, in this situation, the circularity becomes
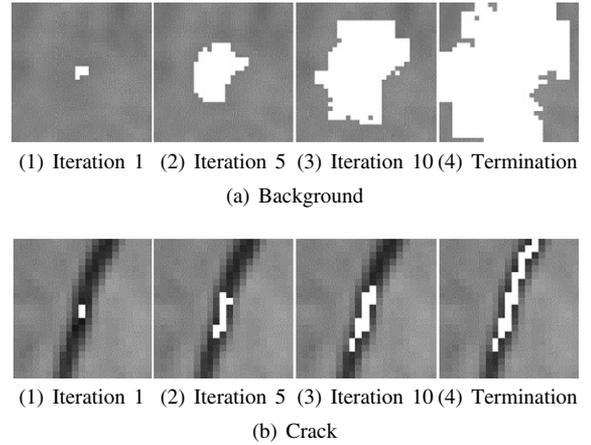

(1) Iteration 1 (2) Iteration 5 (3) Iteration 10 (4) Termination

(a) Background

(1) Iteration 1 (2) Iteration 5 (3) Iteration 10 (4) Termination

(b) Crack

Fig. 3 Examples of the percolation process
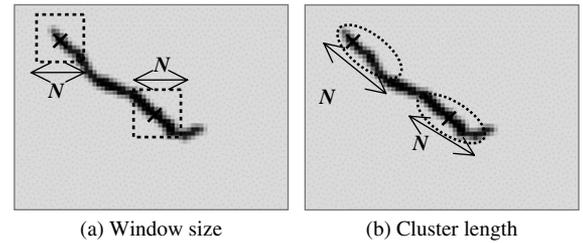

(a) Window size   (b) Cluster length

Fig. 4 Length criterion

close to 1, and then unclear cracks are regarded as backgrounds. This section describes an improved percolation processing technique for crack detection. The improved features of the percolation model are as follows: (i) the parameter w is modified in the percolation process and (ii) the criterion size for the percolation process is replaced by the length of the percolated cluster. Additionally, we introduce a noise reduction technique for cracks based on the percolation model.

4.1. Improved percolation process The parameter w can be modified according to the percolated cluster. In the improved process, we use F_c, which is calculated every time the percolation process executes step 2 described in Section 3. F_c is calculated by (2) using D_p of the immediate shape at each iteration. The improved parameter w' is obtained by using the following equation:

$$w' = F_c \cdot w \tag{3}$$

Then, Eqn (1) is improved as follows.

$$T = \max(\max_{p \in D_p}(I(p)), T) + w' \tag{1'}$$

where, when the percolated cluster forms a thin shape (i.e. F_c approaches 0), the parameter w' rapidly decreases. When the percolated cluster forms a circle (i.e. F_c approaches 1), the parameter w' is almost constant. Therefore, the improved percolation processing technique uses (4) instead of (1) in step 2 of Section 3.

A further improvement is that we employ the length of the percolated region and not the window size. This improvement is
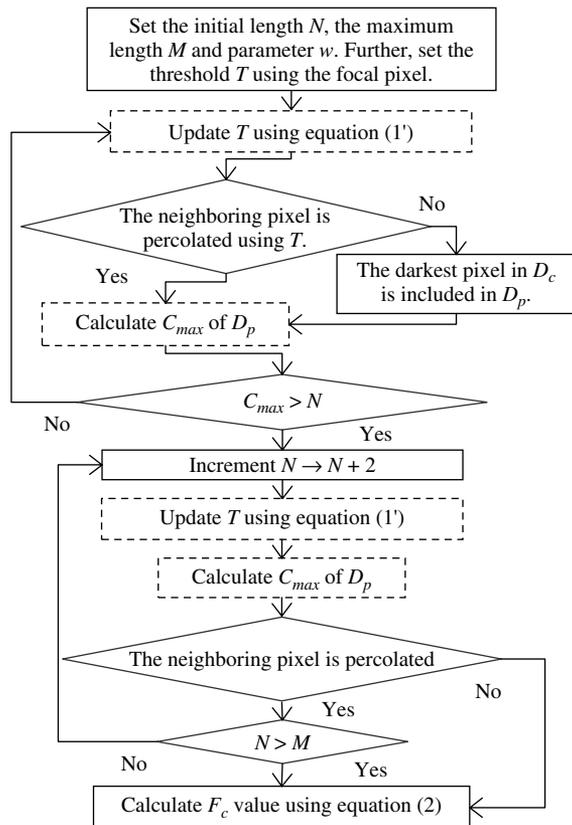
Fig. 5 Flowchart of the improved percolation process

Fig. 6 Differences in the percolated regions
($\times$: focal point for percolation; white part: percolated region)

simple, since the proposed percolation processing is a scalable window processing technique. In Fig. 4, the percolated cluster can be evaluated using the same criterion, even if the initial points are the front or intermediate points of the cracks.

The improved descriptions are modified as follows: the initial window size $N \times N$ is replaced with length N, the maximum window size $M \times M$ becomes length M, and step 4 described in Section 3 is modified as follows:

4'. When N is greater than $C_{\max}$, the percolation process proceeds to step 5, and N is incremented to $N + 2$. Otherwise, the process goes back to step 2.

Figure 5 shows the flowchart of the improved percolation processing. The improved parts are represented as the dashed boxes. The update process of T is improved. To use the length as criterion of the percolation process, the calculation process of $C_{\max}$ is added instead of the window size.

4.2. Verification   In this section, we verify the effectiveness of parameter w' compared with the previous parameter w. Figure 6 shows the original image and the results obtained by using the two methods for three types of cracks: 'dark crack,' 'background,' and 'unclear crack,' respectively. A dark crack is noticeably darker than the background. On the other hand, an unclear crack is brighter than ordinary cracks. The parameters of the improved percolation processing are set as follows: initial length $N = 21$, maximum length $M = 41$, and parameter $w = 1$.

In the dark crack and the background cases, the results reveal percolated regions having almost comparable shapes. On the
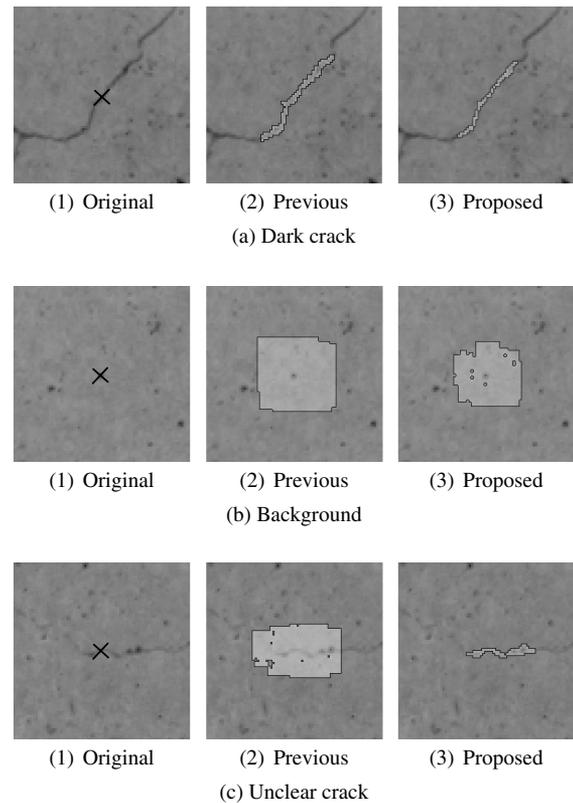
other hand, in the case of an unclear crack, the two methods yield percolated regions having different shapes. The previous method expands the shape of the percolated region; however, the improved method can maintain a percolated region with a thinner shape. The improved method thus detects unclear cracks in a manner similar to that for dark cracks. Moreover, Fig. 7(a) and (b) show the alteration of parameter w' and threshold value T in these situations. From Fig. 7(a), the parameter w' is close to 0 in the crack case, while the previous parameter w is fixed as 1. In the background case, the parameter w' is not close to 0. From Fig. 7(b), the improved method can be verified by the alteration of T. In the situation of a dark and unclear crack, T stays almost constant. On the other hands, in the previous method, T gradually increases in all situations. Therefore, the unclear cracks are regarded as background.

4.3. Noise reduction   In this section, a noise reduction method based on the percolation model is introduced. When noise reduction is performed after the binarization, the resultant noise reduction depends on the threshold value used in the binarization procedure. Therefore, we apply a noise reduction method that is based on the percolation model, before the binarization.

The image produced by the improved percolation processing is represented by $F_c \times 255$ (i.e. circularity). One of the features of the cracks is their thin shape. When the focal pixel is regarded as a crack, the F_c of the neighboring pixels has a value similar to the F_c of the focal pixel, and these pixels are interconnected by a 'constant length.' Therefore, the noise
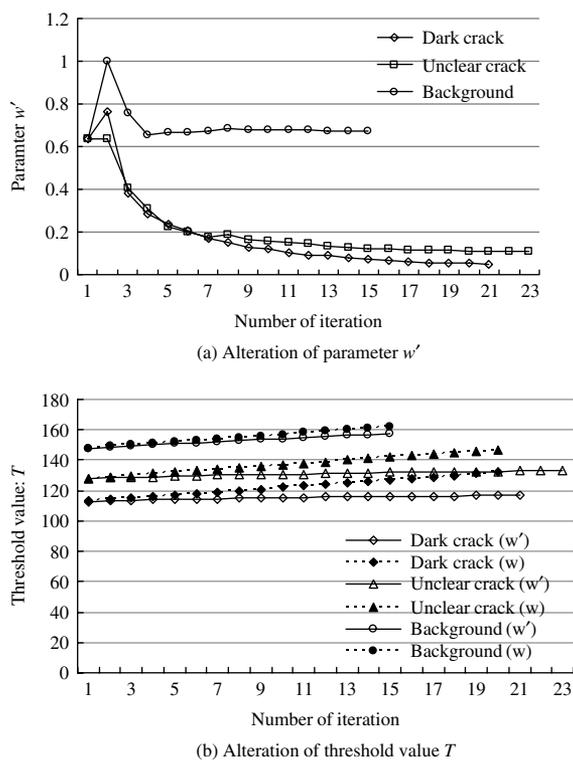
(a) Alteration of parameter w'

(b) Alteration of threshold value T

Fig. 7 The effectiveness of the improved method

(a) Image using $F_c \times 255$ (b) After noise reduction
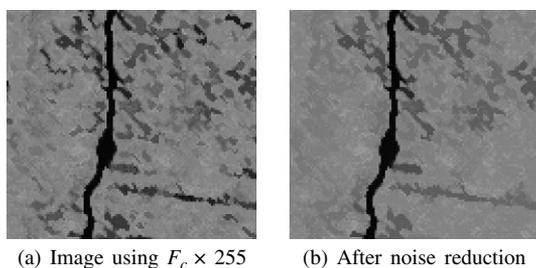
Fig. 8 Example of the proposed noise reduction technique

(1) Image 1   (2) Image 2

(a) Original images

(1) Image 1   (2) Image 2

(b) Results of improved percolation and noise reduction

(1) Image 1   (2) Image 2

(c) Results of the proposed method

Fig. 9 Results of the proposed crack detection method

reduction method is applied to the images using the F_c value. In this process, the constant length is then set to N. Consequently, the resultant value of F_c on the focal pixel after the noise reduction depends on its connective length among the neighborhoods.

The improved percolation method can use the length of the percolated cluster. Then, in the percolation processing, N, M, and w are set to N, N, and 0, respectively. Further, the feature of the noise reduction is extracted as the maximum value of F_m in the percolated region as follows:

$$F_m = \max_{p \in D_p} (I(p)) \qquad (4)$$

Figure 8 shows an example of the results of the noise reduction. In this example, the parameter N is set to 21.

5. Experiments

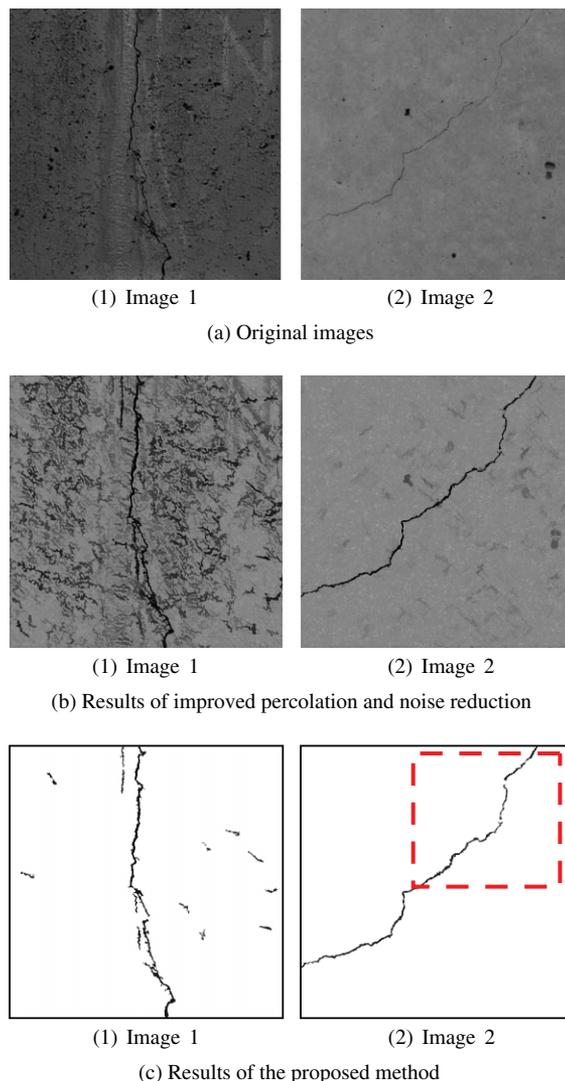**5.1. Crack detection**   We conducted experiments using images in different actual concrete surfaces in order to evaluate the proposed method. In the experiments, our method is applied to crack detection in 50 images of concrete surfaces having some noise and unclear cracks. Figure 9(a) shows examples of the original images used in this experiment. Image 1 includes many noise spots, and the brightness of the crack in image 2 is unclear.

The image resolution is 480×480 pixels, corresponding to an area of approximately 240×240 mm^2 in the actual surface scale.

First, we applied the improved percolation processing to the original images, as described in Section 4.1. The parameters of the percolation processing are configured as follows: initial length $N = 21$, maximum length $M = 41$, and parameter $w = 1$. These parameters are derived from Ref. 12. In order to determine the initial length, we performed a statistical analysis of the crack length in a digital image of the concrete surface by using our previous measurement system [7]. For the statistical analysis, we used an image with a resolution of 3040×2008 pixels, corresponding to an area of approximately 1.5×1.0 m

(1) Image 1   (2) Image 2

(a) Previous percolation method[12]

(1) Image 1   (2) Image 2

(b) Previous wavelet method[7]

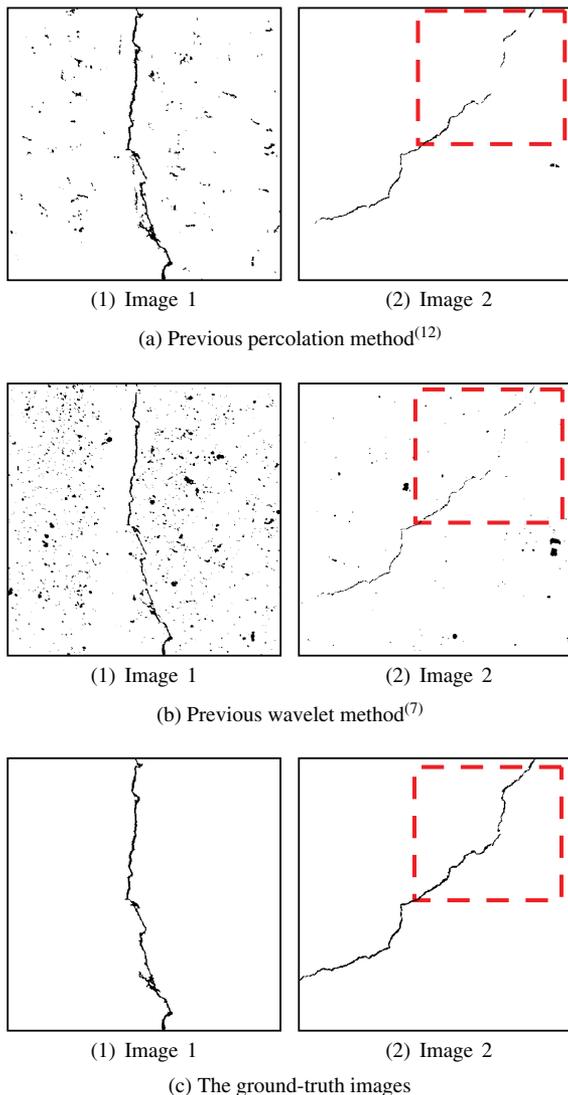(1) Image 1   (2) Image 2

(c) The ground-truth images

Fig. 10 Results of the comparison of images

[2]. Next, we measured 114 cracks and obtained an average length of 20.4 pixels. The maximum length M is determined as 41, which is twice the average length. The parameter w was experimentally determined. Next, we applied noise reduction based on the percolation model as described in Section 4.3. The parameter N was set to 21 in the same manner as the previous percolation processing technique, and the parameter w was set to 0. Figure 9(b) shows the results of the proposed method after the noise reduction. Figure 9(c) shows the binarization results of crack detection with an optimal threshold for the different images.

For comparison, we employed three approaches: (i) the previous percolation method [12] using the same parameters $N = 21$, $M = 41$, and $w = 1$; (ii) the previous wavelet-based method [7], which used a combination of several image processing techniques including wavelet transform, shading correction, and binarization; and (iii) the ground-truth image created manually by a human observer. Figure 10 shows the results of these comparison methods.

In Fig. 9, the proposed method can detect fine cracks in both the noisy and unclear crack cases. In the previous percolation method [12], the result in the noisy case is comparable with the proposed result. However, in the concrete surface image with unclear cracks, the previous percolation method cannot detect cracks with high accuracy and the connectivity is lost in the dashed rectangular area in Figs 9(c) and 10. The previous wavelet-based method [7] includes many background noises and the cracks are disconnected.

The proposed method detects fine cracks with connectivity, even if the brightness of the cracks is unclear. This is achieved by using the shape information of the percolated region in the expanding process. When images include some noise, the performance of the proposed method is comparable to the previous percolation methods.

5.2. Estimation of performance   In this section, we perform an ROC analysis [15,16], which is also used in Refs 5,10, and a precision-recall analysis [17] to evaluate the performance of our method. ROC analysis plots the sensitivity *versus* (1 − specificity) for each discriminant threshold of a binary classifier system. The precision-recall analysis is also similar to the ROC analysis; however, the calculation of the quantitative evaluation is different. Let G denote the ground-truth image and H denote the compared image produced by the crack detection method. The sensitivity, (1 − specificity), precision, and recall are represented as

$$Sensitivity = \frac{[Crack\ pixels\ in\ H \cap crack\ pixels\ in\ G]}{[Crack\ pixels\ in\ G]} \quad (5)$$

$$\begin{aligned} 1 &- specificity \\ &= \frac{[Crack\ pixels\ in\ H \cap background\ pixels\ in\ G]}{[Background\ pixels\ in\ G]} \end{aligned} \quad (6)$$

$$Precision = \frac{[Crack\ pixels\ in\ H \cap crack\ pixels\ in\ G]}{[Crack\ pixels\ in\ H]} \quad (7)$$

$$Recall = sensitivity \quad (8)$$

where [·] represents the number of pixels in the argument area and ∩ represents product set. These values are influenced by the selection of the threshold value in the binarization procedure. In the ROC analysis, the ROC curve can represent a reciprocal relationship by plotting the sensitivity values on the y-axis and the (1 − specificity) values on the x-axis. The precision—recall curve is also identified by plotting the precision values on the y-axis and recall values on the x-axis. We employed these curves for the quantitative evaluation. Figures 11 and 12 show the ROC curve and the precision—recall curve of each method.

For the method of evaluation, we used the area Az under the curve in the ROC analysis. We calculated Az by an approximation of the trapezoidal integration. Table I shows the average value of Az and the average precision values in each method for 50 images. Moreover, as is well known, there is a trade-off relationship between the precision and recall values. In concrete surface inspection, detecting noises is preferable to missing cracks from the view point of fail safety. We calculated the precision values at different recall values.

In the ROC analysis, the closer the curve is to the upper left corner, the better is the performance. On the other hand, in the precision-recall curve, the closer the curve is to the upper right corner, the better is the performance.

Table I. Quantitative results

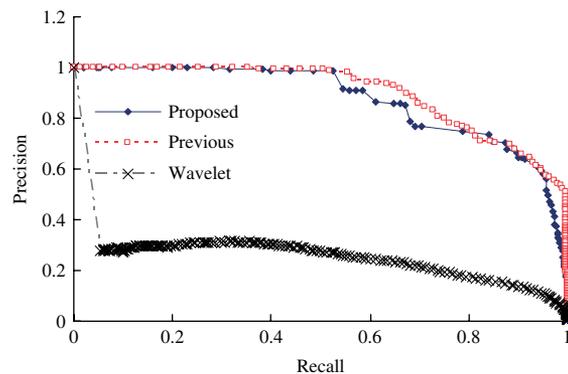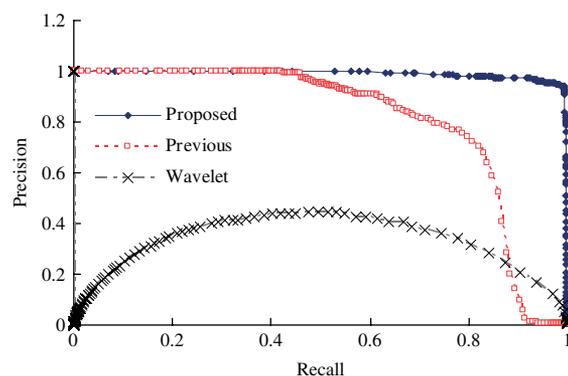| Method	$\overline{Az}$	Precision		
		Recall: 0.80	*Recall*: 0.85	*Recall*: 0.90
Proposed	0.995	0.788	0.753	0.701
Previous [12]	0.959	0.750	0.647	0.529
Wavelet [7]	0.981	0.324	0.265	0.201

(a) ROC curve in image 1
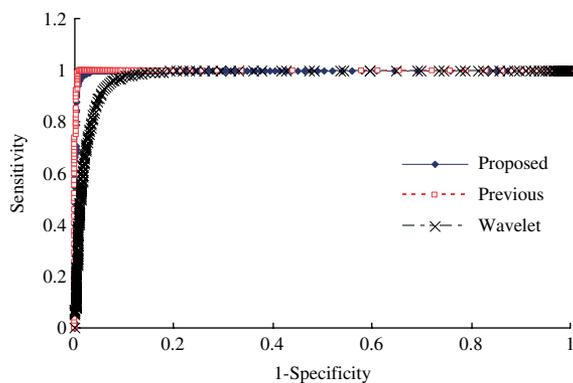
(a) Precision-recall curve in image 1

(b) ROC curve in image 2
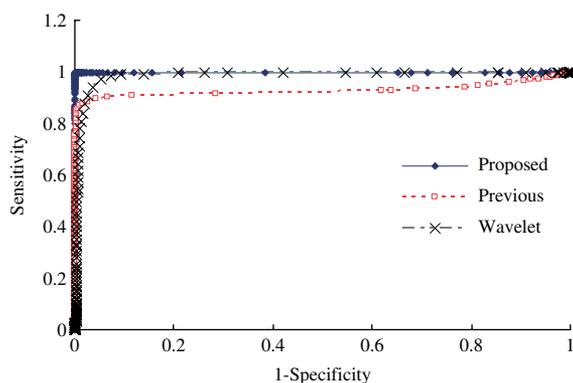
(b) Precision-recall curve in image 2

Fig. 11 ROC curves

Fig. 12 Precision-recall curves

In Fig. 11(a), the proposed method and the previous percolation method are closer to the upper left corner. In Fig. 11(b), the closest curve represents the proposed method. In this study, we focus on the accuracy of crack detection. The improved method works better in detecting unclear cracks accurately. We calculated the difference of average brightness between crack pixels and background pixels. The results of images 1 and image 2 are approximately 52 and 25, respectively. Although image 1 includes some noise, the cracks are relatively clear. Therefore, the proposed method is comparable to the previous percolation method. On the other hand, the proposed method can yield better performance even for unclear cracks. In Table I, the *Az* value of the proposed method is also high.

In Fig. 12, the results are similar to the ROC curve. The proposed and the previous methods are closer to the upper right corner in (a). Moreover, the proposed method is the closest in (b). Following the precision values in Table I, that of the

proposed method is higher than in the other methods, and then the precision value is realized 70% even when the recall value is 90%.

Thus, the proposed method has been experimentally proven to have better performance than the other previous methods in both the ROC analysis and precision-recall analysis.

6. Conclusions

In this paper, we proposed a crack detection method based on the percolation model. We improved the previous percolation model whose process depends on the shape and brightness in the percolated region. The proposed method is based on the length criterion and not the window criterion. Further, we introduced a noise detection technique based on the percolation model using the length criterion.

In the experiments, the effectiveness of the proposed method is investigated using actual concrete surfaces images, and we applied the ROC analysis and precision-recall analysis. In the case of the images with unclear cracks, the proposed method demonstrated the most effective capabilities. In noisy images, the proposed method exhibited a performance comparable to the previous percolation method. In the quantitative analysis, the proposed method had a performance superior to the other previous methods. Our method scored 70% as the precision value when the recall value was 90%.

With regard to practical use, methods described in the literature [6,7] (using the wavelet-based approach) have already been employed. As our method realizes a higher accuracy than these methods, we are planning to implement the proposed method in a crack measurement system to be used in practical applications.

References

(1) Architectural Institute of Japan. *Shrinkage Cracking in Reinforced Concrete Structures—Mechanisms and Practice of Crack Control*. AIJ: Tokyo, 2003.

(2) Dare PM, Hanley HB, Fraser CS, Riedel B, Niemeier W. An operational application of automatic feature extraction: the measurement of cracks in concrete structures. *Photogrammetric Record* 2002; **17(99)**:453–464.

(3) Chen LC, Shao YC, Jan HH, Huang CW, Tien YM. Measuring system for cracks in concrete using multitemporal images. *Journal of Surveying Engineering* 2006; **132(2)**:77–82.

(4) Abdel-Qader I, Abudayyeh O, Kelly ME. Analysis of edge detection techniques for crack identification in bridges. *Journal of Computing in Civil Engineering* 2003; **17(3)**:255–263.

(5) Hutchinson TC, Chen Z. Improved image analysis for evaluating concrete damage. *Journal of Computing in Civil Engineering* 2006; **20(3)**:210–216.

(6) Ito A, Aoki Y, Hashimoto S. Accurate extraction and measurement of fine cracks from concrete block surface image. *Proceedings IECON2002*, 2002; 77–82.

(7) Tanaka H, Yamada M, Sato R, Hashimoto S. Japanese Patent, P2006-162477A, 2006.

(8) Kawamura K, Miyamoto A, Nakamura H, Sato R. Proposal of a crack pattern extraction method from digital images using an interactive genetic algorithm. *Proceedings of Japan Society of Civil Engineers* 2003; **742**:115–131.

(9) Roli F. Measure of texture anisotropy for crack detection on textured surfaces. *Electronics Letters* 1996; **32(14)**:1274–1275.

(10) Fujita Y, Mitani Y, Hamamoto Y. A method for crack detection on a concrete structure. *Proceedings of International Conference on Pattern Recognition* 2006; **3**:901–904.

(11) Miwa M, Kobayashi T, Zhang X, Sato M. Detecting cracks on the tunnel wall using watershed and graph analysis. *ITE Technical Report* 2005; **29(59)**:11–14.

(12) Yamaguchi T, Hashimoto S. Image processing based on percolation model. *IEICE Transactions on Information and Systems* 2006; **E89-D(7)**:2044–2052.

(13) Stauffer D. *Introduction to Percolation Theory*, 2nd edn. CRC press: New York, 1994.

(14) Grimmett G. *Percolation*. Springer: Berlin, 1999.

(15) Masumoto J, Hori M, Sato Y, Murakami T, Johkoh T, Nakamura H, Tamura S. Automated detection of liver tumors in X-ray CT images. *IEICE Transactions on Information and Systems* 2000; **J83-D(1)**:219–227.

(16) van Erkel AR, Pattynama PMTH. Receiver operating characteristic (ROC) analysis: basic principles and applications in radiology. *European Journal of Radiology* 1998; **27(2)**:88–94.

(17) Van Rijsbergen CJ. *Information Retrieval*, 2nd edn. Butterworth-Heinemann: London, 1979.

Tomoyuki Yamaguchi (Non-member) received the BE and ME degrees in Mechanical Engineering from Waseda University in 2002 and 2004, respectively. Since October 2004, he has been a Research Associate at the Graduate School of Science and Engineering, Waseda University. His research interests include image processing, image recognition, robotics, and agriculture.

Shingo Nakamura (Non-member) received the BE and ME degrees in Applied Physics from Waseda University in 2000 and 2002, respectively. Since October 2005, he has been a Research Associate at the Graduate School of Science and Engineering, Waseda University. His research interests include machine learning, robotics and image processing.
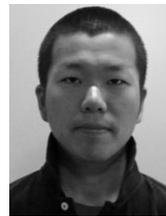
Ryo Saegusa (Non-member) received his DE degree in Applied Physics from Waseda University in 2005. He has been a Post-doctoral Researcher at the Italian Institute of Technology and a visiting researcher at the Humanoid Research Institute in Waseda University since 2007. His research interests include machine learning for robotics and data analysis. He is a regular member of IEEE and IEICE.

Shuji Hashimoto (Member) received the BS and DE degrees in Applied Physics from Waseda University in 1970 and 1977, respectively. He is currently a Professor at the Department of Applied Physics, Waseda University. His research interests include neural networks, image processing, humanoid robots, and Kansei information processing.