Hindawi
Mathematical Problems in Engineering
Volume 2019, Article ID 8157205, 16 pages
https://doi.org/10.1155/2019/8157205

Research Article

Partial Differential Equation-Based Enhancement and Crack Detection

Uche A. Nnolim

Department of Electronic Engineering, University of Nigeria, Nsukka 410001, Enugu, Nigeria

Correspondence should be addressed to Uche A. Nnolim; uche.nnolim@unn.edu.ng

Received 25 November 2018; Accepted 10 March 2019; Published 26 March 2019

Academic Editor: Pablo Lopez-Crespo

This paper presents an effective partial differential equation- (PDE-) based preprocessing algorithm for automated image-based crack detection. The proposed formulation combines various relevant and multiple processes such as contrast and selective edge enhancement in addition to edge-preserving smoothing to enhance the image prior to detection. The approach is adaptive and controlled by reliable image metrics to determine the stopping time of the PDE ensuring optimum results for various images. Additionally, a simplified thresholding algorithm based on local global maximum gradient matching is used to extract the crack features from the image. The proposed scheme does not require arbitrary or manually tuned parameters nor a large dataset for training to obtain good results. Experiments indicate that the proposed approach performs better when compared to several other algorithms in the literature.

1. Introduction

There are several crack detection methods for concrete structures and surfaces. These are usually nondestructive methods, which include radiographic, X-ray, laser, ultrasonic, infrared, and thermal imaging [1]. Techniques involving image acquisition and software-based image processing are increasingly becoming an attractive option. This is due to low-cost and ease of operation and automation, reducing inspection time and manpower demands [1, 2]. The image acquisition system may employ ground or aerial unmanned autonomous vehicles or aircraft (UAV/UAA), popularly known as drones, which are sent into the target area. The image processing component combines a number of standard, modified, or newer algorithms to achieve the crack detection process.

Earlier works utilized thresholding, edge detection, morphological operations, genetic algorithms, Gabor filters, support vector machines (SVM), and other artificial (and recently convolutional) neural networks (ANN/CNN) [1, 2]. The simpler algorithms are not as effective and may register a considerable number of spurious edges or features as cracks, while the more complex methods require extensive training with a large dataset of crack images.

The inherent challenges of image-based detection include irregular geometry of cracks, noise effects, uneven illumination/shadows, texture roughness of concrete surface, and concrete spall addressed by some these algorithms [1]. The shadows generated by rough texture and the similar brightness of foreground and background features in such images affect segmentation results; thus illumination normalization may be required in some cases [1].

PDE-based techniques have proven their effectiveness in solving image processing problems. They provide an attractive alternative to closed form solutions. Multiple processes occurring simultaneously are combined in a weighted form, enabling the control of the individual contributions (of each of these processes) to the overall flow [3]. PDE-based algorithms have been proposed for smoothing, segmentation, thresholding, and morphological operations with interesting results. Thus we extend PDE-based formulations to crack image enhancement and detection using an adaptive optimization scheme to guide its evolution. We achieve this by exploiting gradient-based measures to obtain an automated and guided PDE for preprocessing the crack images prior to detection. We assume that the cracks represent components with the highest entropy/detail/edge information. Thus by maximizing the entropy or clarity, we minimize the energy

in the process and it is expected that the system converges to a steady state when the gradient is at a maximum.

The rest of the paper is as follows. Section 2 presents the methods used in addition to key contributions and features of the proposed algorithm. Section 3 presents the proposed algorithm and preliminary analysis of results. Section 4 provides extensive experimental results and comparative analysis. The final section presents the discussions and conclusion.

2. Materials and Methods

The various aspects of image processing employed in the proposed crack detection algorithm include illumination correction, selective edge enhancement and edge preserving smoothing, morphological operators, and thresholding functions.

Illumination correction algorithms employ the illumination-reflectance model to normalize illumination in addition to dynamic range compression and contrast enhancement [4]. The classical examples include Homomorphic and Retinex filters, though other algorithms have been proposed in the literature.

Edge detection and enhancement filters are classified as high pass/high frequency/gradient-based filters, which preserve edges or regions with high frequency content while removing low frequency content [4]. The edge detection filters such as Sobel, Prewitt, and Canny are high-pass filters, which preserve only edge information in images [4]. The edge enhancement/sharpening filters such as Laplacian or unsharp masking algorithms augment the edge information or high frequency content and are termed high-frequency emphasis/high-boost filters [4]. They yield sharper images after filtering, which can be observed visually and verified using image gradient-based measures.

Nonlinear, edge-preserving smoothing filters are useful in image filtering since they can suppress noise and unwanted high frequency content/component without smoothing out desired edge details. They are usually based on rank order statistics, nonlocal means, or tensor-based anisotropic diffusion (AD) PDE [5] or total variation regularization- (TVR-) [6] based algorithms. The classic examples include the median filter and its numerous variations, bilateral, trilateral, and guided filters, AD and TVR based approaches, and variants.

Morphological and thresholding operators employ Boolean logic in their operation and include dilation, opening, erosion, and closing processes, which are invaluable tools in image segmentation and object identification by shape and size analysis [4]. The thresholding functions are used for removing extraneous image information, leaving only the essential intensities required for the object recognition from the image. Thus, the resulting output image has much fewer gray level intensities than the input image. Classical algorithm examples include Otsu's method [7], entropy-based schemes, etc.

2.1. Key Contributions and Features. The proposed scheme utilizes terms for selective smoothing, edge, and contrast enhancement in its formulation. The key and novel contributions of the work include the following:

(i) Simultaneous smoothing, contrast, and selective edge enhancement.

(ii) Adaptive optimization using reliable image metrics such as edge/gradient information.

(iii) Automated stopping time of PDE due to guided evolution using the aforementioned processes.

(iv) Simplified thresholding scheme for extraction of crack pixels based on local-global maximum gradient matching.

3. Proposed Algorithms and Modifications

Cracks observed in acquired images can be characterized as being composed of high frequency components; thus they can be detected by employing edge enhancement filters. However, rough textures are also composed of high frequency components and will also be detected as edge detail. Thus, the proposed algorithm augments and/or preserves the crack details, while smoothing background texture, enabling improved edge detection, morphological filtering, and thresholding results. The algorithm is built on a closed form approach to realize an iterative PDE-based framework. The algorithms can be used for instances where the input crack image is degraded by uneven illumination and/or is a colour image. The basic steps for the proposed approaches (PA) are shown in Figure 1.

3.1. Preliminary Analysis. Previous algorithms work well for images with highly distinct crack features, where the crack pixel intensities have the lowest pixel values. Additionally, images with mainly horizontal or vertically oriented crack patterns are generally easier to process. Thus, straightforward approaches such as smoothing, edge detection, and thresholding yield almost completely segmented images, indicating prominent crack features. However, more complex cracks are highly irregular and are not completely horizontal, vertical, straight, or curved and will have smaller extensions emanating from the main crack region. Additionally, the non-ideal conditions such as uneven illumination/shadows/shading, noise, texture roughness of concrete surface, irregular shapes/sizes and spots, and blemishes complicate the image-based crack detection process. Usually, machine learning and neural network-based techniques are used to address such issues. However, such schemes require a large amount of image data to adequately train the system for crack detection and analysis. We address each of these problems using a variety of techniques. Additionally, some images display cracks, whose brightness or intensity closely resembles the background intensity, making it even more difficult to extract such features. This can be mitigated using tonal mapping operators such as Homomorphic filter (HF) and Retinex. We use a previously devised fuzzy Homomorphic enhancement (FHE) algorithm [13] to normalize the image intensities where necessary. For the problem of similarity of crack foreground and background intensities,

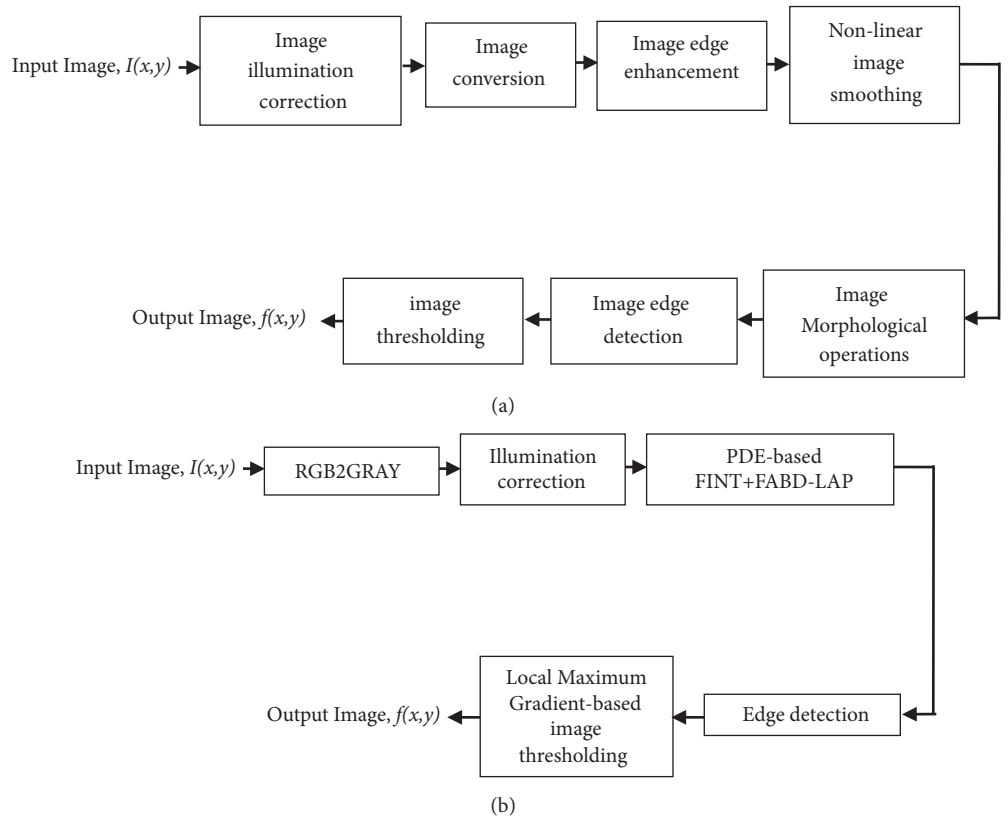

Figure 1: Basic flowchart of proposed image-based crack detection algorithms: (a) PA-1 [8] and (b) PA-2.

we first analyze a row profile of the input image, which is the usual practice in the field [12]. Figure 2 shows the plot of intensities against the number of pixels for a single row of the crack image for two different images from the literature [9, 14, 15].

In Figure 2, we observe that the lowest intensity value is likely to be a crack region pixel candidate. For images where both foreground and background intensities are relatively close, the variance or deviation is low. Thus we expect low contrast, entropy, or gradient values. This is easily observable in the images where the crack in Figure 2(a) is easily discernible compared to the one in Figure 2(b). Thus using illumination correction makes the case worse for images similar to (b) as shown in Figure 3. The lowest intensities have been considerably increased, while the highest intensities have obtained a modest increase in (b) compared to (a) where the gap is more or less maintained in spite of overall increase of all intensities and the crack is still clearly distinguishable in (a) compared to (b).

For the ideal crack images, the gap between the lowest pixel intensity and the others is much higher. We measure the entire original and contrast enhanced crack images and compute their relative values of standard deviation (RSD), entropy (RE), and average gradient (RAG) measures to compare and prove our assumptions. The dataset [14, 15] used consists of 155 images and evaluation results are shown in Figure 4.

The plots show which images benefit most from contrast enhancement based on the RS and RAG values, which yield nearly identical results. The former are the most reliable unlike the RE, which shows that entropy is unchanged or degraded for a majority of the crack images.

3.2. Proposed PDE-Based Fuzzy INT-Boosted Anisotropic Diffusion (PDE-FINT-AD) Algorithm. Based on the images and results in Figures 1–3, we require an algorithm that darkens already dark pixels while brightening currently bright pixels. Such an algorithm that easily performs this task is the fuzzy intensification operator (F-INT) [16]. Repeated passes of the F-INT lead to an almost thresholded or segmented image. However, we wish to perform this operation gradually and in a controlled fashion. Thus, we reformulate the algorithm in the form of a partial differential equation based on the framework by Shapiro et al. [3] as

$$\frac{\partial I(x,y,t)}{\partial t} = f(x,y,t) - I(x,y,t) \quad (1)$$

In (1), $I(x,y,t)$ is the continuous image while $f(x,y,t) = f\{I(x,y,t)\} = FINT\{I(x,y,t)\}$ and using finite difference method (FDM), we obtain the following expression in

$$I^{t+1}(x,y) = I^{t}(x,y) + [f\{I(x,y,t)\} - I(x,y,t)]\,\Delta t \quad (2)$$

The results are shown in Figure 5 indicating improvements.

However, we would also like to smoothen out the noise and rough background texture in the process. Thus we add

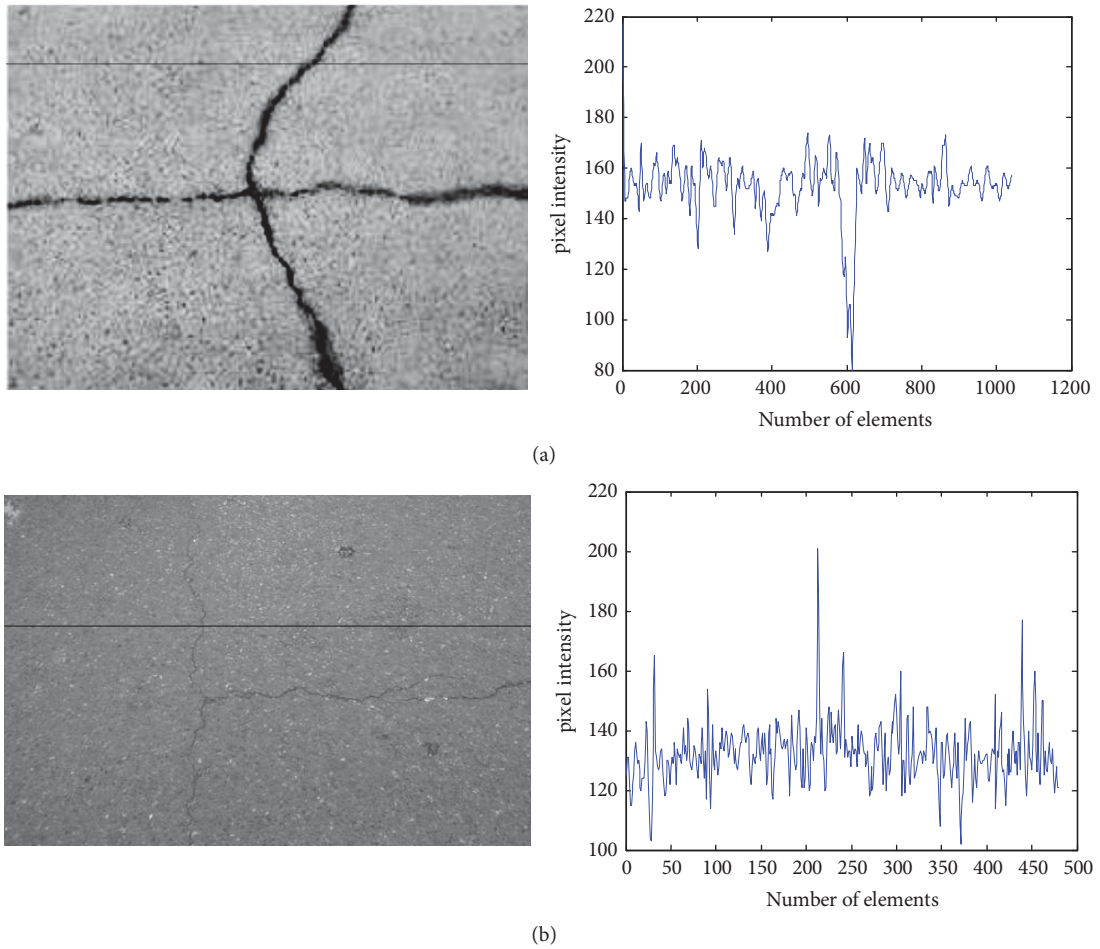

Figure 2: Crack images [9] with (a) distinct and (b) similar intensities between background and foreground and corresponding row pixel profile plots.

the anisotropic diffusion (AD) flow, resulting in the modified expression [3]:

$$\frac{\partial I(x,y,t)}{\partial t} = \lambda c_{AD}\left(\left\|\nabla I(x,y,t)\right\|\right) \operatorname{div}\left(\frac{\nabla I(x,y,t)}{\left\|\nabla I(x,y,t)\right\|}\right) + f(x,y,t) - I(x,y,t) \quad (3)$$

where λ is a weighting/balance factor that controls the degree of smoothing, $\nabla I(x,y,t)$ is the gradient of the image, $\|\,\|$ is the norm, while 'div' is the divergence operator, and t is the time scale. The term $c_{AD}(\|\nabla I(x,y,t)\|)$ is the AD diffusion coefficient (or edge stopping function in this case), given as

$$c_{AD}\left(\left\|\nabla I(x,y,t)\right\|\right) = \frac{1}{1+\left[\left\|\nabla I(x,y,t)\right\|/\kappa\right]^2} \quad (4)$$

The term κ controls the degree of smoothing in the function such that a larger value leads to a greater degree of smoothing while a lower value leads to less smoothing along edges.

3.3. Gradient Optimized Forward and Backward Diffusion with Laplacian (PDE-FINT-FAB-LAP). Though the AD addition is an improvement, it will also smoothen or smudge the small gradient magnitudes around the crack edges, leading to discontinuities when edge detection is performed. This loss of edges/details causes breakages as formerly linked edge features are disconnected. Thus, there is the need to strengthen edges prior to smoothing. However, linear edge enhancement operators such as the Laplacian or unsharp masking result in noise amplification, complicating the edge detection process by introducing spurious or false edges and spots. Thus we would like to selectively sharpen the edges while smoothing the background to prevent noise enhancement. We also wish to avoid the explosive instability of the reverse heat diffusion process [17]. Thus we utilize a previously proposed modification known as the forward and backward (FAB) diffusion coefficient [17], which is given as

$$c_{FAB}\left(\left\|\nabla I(x,y,t)\right\|\right) = \frac{1}{1+\left[\left\|\nabla I(x,y,t)\right\|/\kappa_f\right]^n} - \frac{\alpha}{1+\left[\left(\left\|\nabla I(x,y,t)\right\|-\kappa_b\right)/w\right]^{2m}} \quad (5)$$

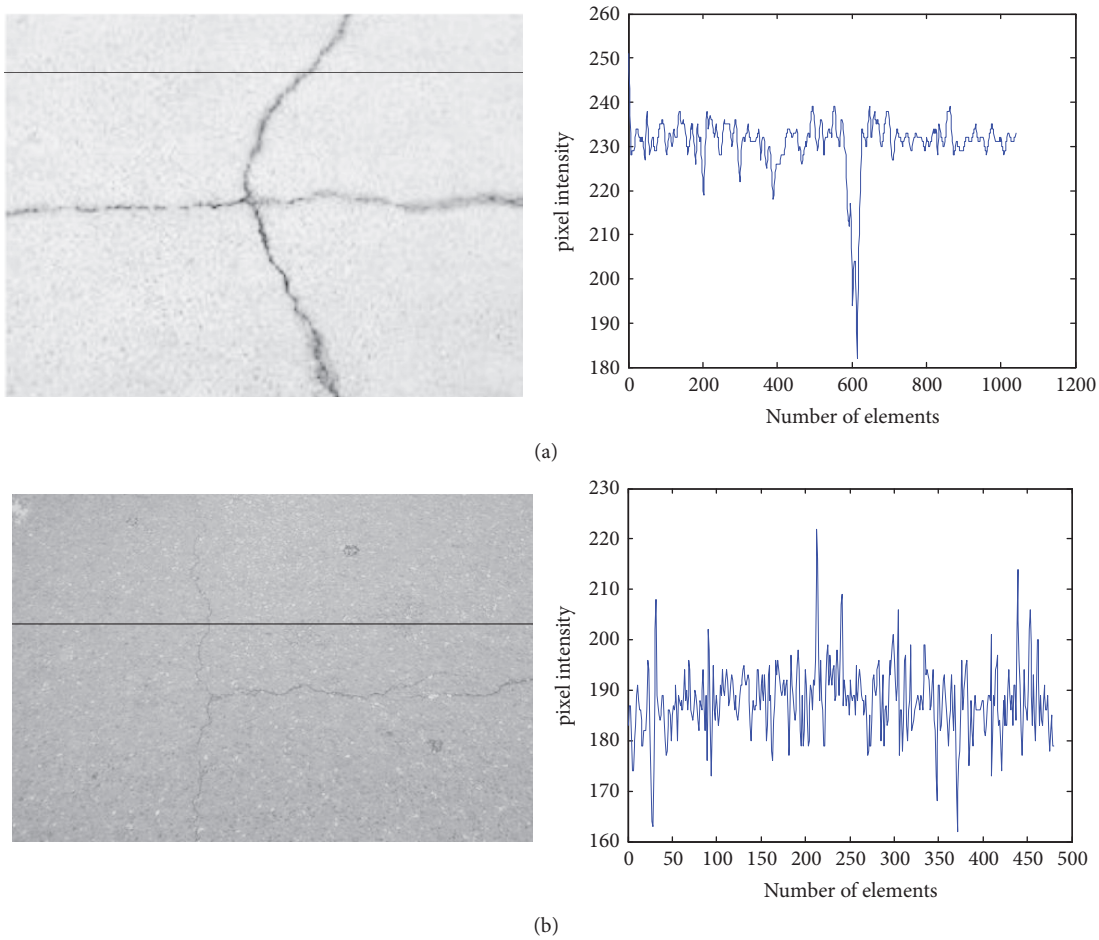

Figure 3: Illumination normalized crack images with (a) distinct and (b) similar intensities between background and foreground with corresponding row pixel profile plots.

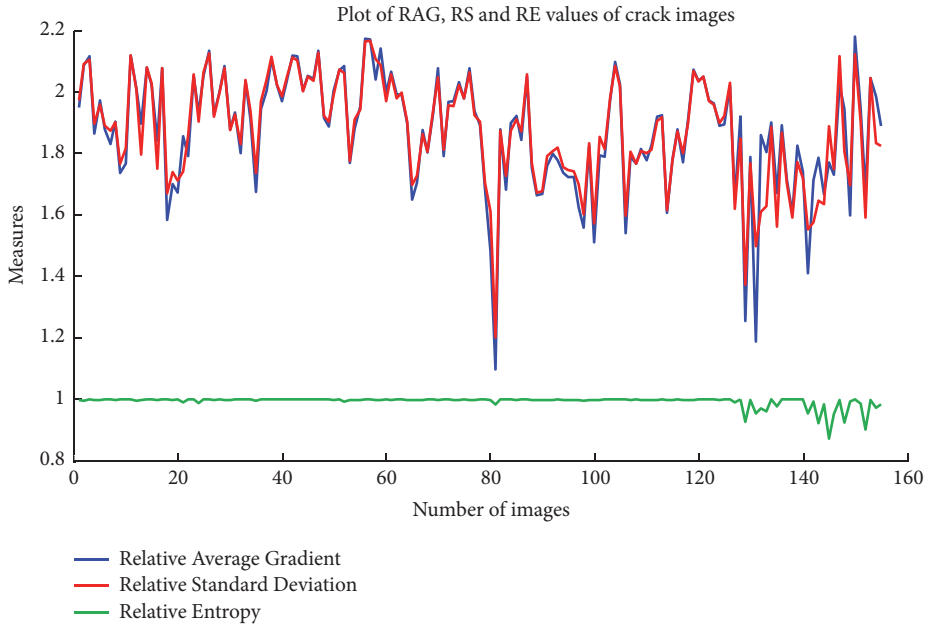

Figure 4: Plots of RAG, RS, and RE of the crack images from dataset.

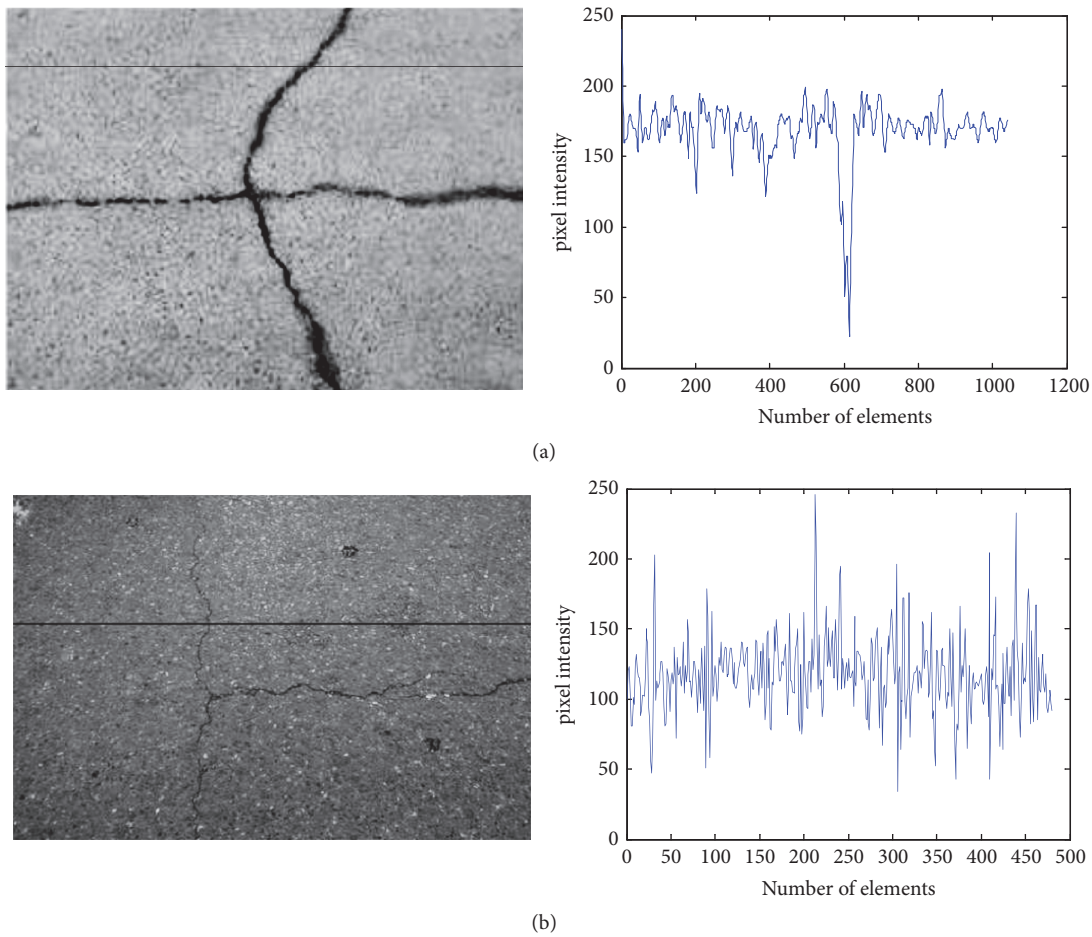

Figure 5: Contrast enhanced crack images with (a) distinct and (b) similar intensities between background and foreground with corresponding row pixel profile plots.

In (5), κ_f is the forward force parameter (FFP), while κ_b and w form the backward force parameter (BFP) and α is the ratio between forward (FDF) and backward diffusion force (BDF) [17]. If BDF > FDF, stabilizing forward force is not robust to mitigate oscillations [17]. The FAB process is unsuitable for texture preservation, making it suitable for crack image enhancement. The cracks in the images can be considered as regions of high gradients to be preserved or enhanced. Conversely, the background texture can be generally considered as regions of low gradients or small scale features. This scheme works well but does not enhance edges as fast as we desire. Thus we need to continue the smoothing while sharpening at a reasonable rate without disrupting the former. Thus we modify the equation by adding the Laplacian, resulting in the modified expression:

$$\begin{aligned}\frac{\partial I(x,y,t)}{\partial t} &= \lambda c_{FAB}\left(\left\|\nabla I(x,y,t)\right\|\right)\operatorname{div}\left(\frac{\nabla I(x,y,t)}{\left\|\nabla I(x,y,t)\right\|}\right)\\ &\quad+\left[f(x,y,t)-I(x,y,t)\right]-\gamma\nabla^2 I(x,y,t)\end{aligned} \tag{6}$$

This yields the generalized PDE-based formulation of smoothing/sharpening and enhancement of a continuous initial image field $I(x,y,t)$ in the form

$$\begin{aligned}\frac{\partial I(x,y,t)}{\partial t} &= \lambda G_{FAB}\left(I(x,y,t)\right)+G_e\left(I(x,y,t)\right)\\ &\quad+\gamma G_s\left(I(x,y,t)\right)\end{aligned} \tag{7}$$

In (7), $G_{FAB}(I(x,y,t))$, $G_s(I(x,y,t))$, and $G_e(I(x,y,t))$ are the simultaneous selective sharpening/smoothing and sharpening and enhancement functions, respectively:

$$\begin{aligned}&G_{FAB}\left(I(x,y,t)\right)\\ &\quad= c_{FAB}\left(\left\|\nabla I(x,y,t)\right\|\right)\operatorname{div}\left(\frac{\nabla I(x,y,t)}{\left\|\nabla I(x,y,t)\right\|}\right)\end{aligned} \tag{8}$$

The enhancement term, $G_e(I(x,y,t))$, is expressed in the PDE form as shown:

$$G_e\left(I(x,y,t)\right)=f\left(I(x,y)\right)-I(x,y,t) \tag{9}$$

The sharpening function, $G_s(I(x,y,t))$ is given as

$$G_s\left(I(x,y,t)\right)=-\nabla^2 I(x,y,t) \tag{10}$$

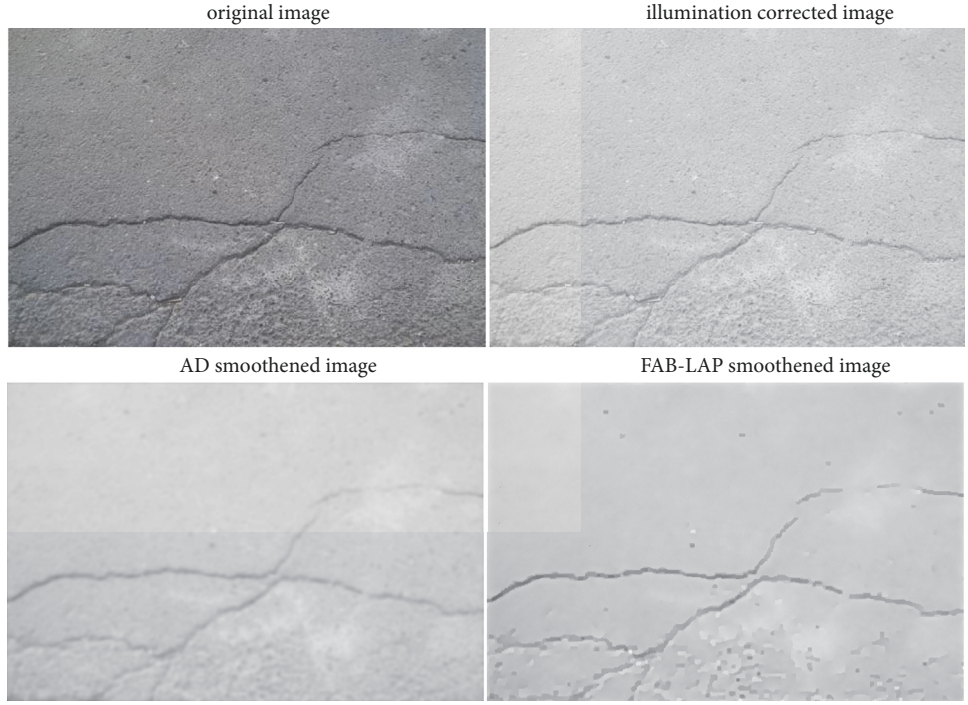

Figure 6: Comparison of AD and PDE-based FINT-FAB-LAP for smoothing and sharpening of crack image features.

The relationship between the enhancement ($G_e(I(x,y,t))$) and edge sharpening ($G_s(I(x,y,t))$) terms is that the former focuses primarily on contrast features while the latter on edge features, improving the edge sharpening result without dominating the selective simultaneous smoothing/sharpening operator, $G_{FAB}(I(x,y,t))$.

The coefficient, γ, is used to reduce or control the dominance of the sharpening term (Laplacian) in the process and, using finite difference method (FDM), we obtain the following;

$$I^{t+1}(x,y) = I^{t}(x,y) + \left[\lambda c_{FAB}\left(\left\|\nabla I(x,y,t)\right\|\right)\operatorname{div}\left(\frac{\nabla I(x,y,t)}{\left\|\nabla I(x,y,t)\right\|}\right) + \left[f(x,y,t) - I(x,y,t)\right] - \gamma\nabla^2 I(x,y,t)\right]\Delta t \quad (11)$$

This expression enables the selective smoothing, sharpening, and contrast enhancement on the crack image. However, our problem is not solved completely since we wish to automate the process for best results. Thus, we require a reliable parameter to help track the changes in contrast and edge modification and select the average gradient (AG) [18], which is computed as

$$AG = \frac{1}{MN} \cdot \sum_{x=1}^{M}\sum_{y=1}^{N}\sqrt{\frac{\left[I(x+1,y) - I(x,y)\right]^2 + \left[I(x,y+1) - I(x,y)\right]^2}{2}} \quad (12)$$

In (12), I is the processed image while M and N are the image dimensions for rows and columns, respectively.

The improvements with the FAB-LAP relative to AD can be seen in Figure 6, where simultaneous sharpening and smoothing are occurring to yield a selectively sharpened image. In the crack image processed with AD, relevant edges are also smoothed out along with the rough texture of the concrete surface. This is not the case with the FAB-LAP algorithm, where the rough background texture is smoothed out leaving the relevant and desired crack features. The illumination normalization also ensures that dark shadows will not be detected as crack features.

Since the AG tracks contrast enhancement much more accurately than most other contrast measures, it allows the adaptive and automatic determination of optimal stopping time for the algorithm without specifying a set of number of iterations. Thus we determine the optimum PDE stopping time by maximizing the average gradient (AG) metric as

$$\textit{While } \frac{\partial AG(I)}{\partial t} \geq 0, \quad \textit{compute the evolving image, } I^{t+1}(x,y) \quad (13)$$

This enables the algorithm to process any image adaptively. However, this means that runtime will vary for vastly different images.

Results of the automated optimization approach are shown in Figure 7, by the maximization of average gradient in the enhanced image. The AG measures clarity and increase in contrast (variance) between background and foreground elements. Thus, continuous enhancement of the image leads

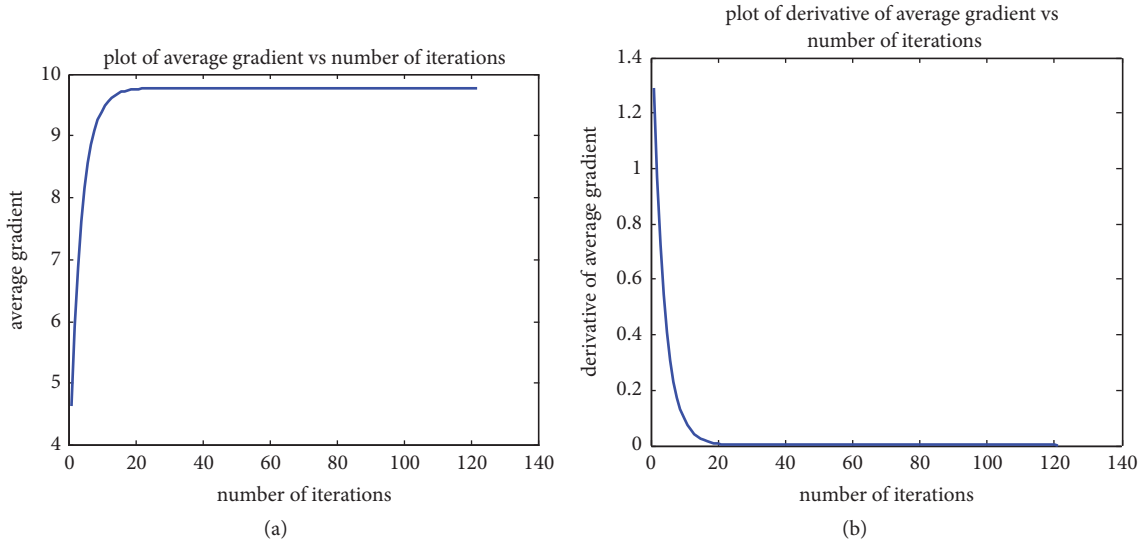

Figure 7: PDE-based FINT-FAB-LAP enhanced crack image gradient versus number of iterations for PA.

to increasing AG as seen in Figure 7(a) until it reaches maximum value (i.e., when background is completely smoothened while foreground and crack features are completely sharpened). However, we also wish to avoid noisy artefacts due to edge overenhancement, thus the need for the sharpness control factor, γ. We also wish to measure this rate of increase (derivative) of AG with respect to time, which should be zero at maximum AG ($\partial AG(I)/\partial t = 0$). Thus the rate of change in AG with respect to number of iterations is shown in Figure 7(b). The values for the parameters are given as $\lambda = 0.5$, $n = 2$, $m = 1$, $\kappa_f = 10$, $\kappa_b = 20$, $w = 20$, and $\alpha = \kappa_f/2(\kappa_b + w)$ [17] and $\gamma = 0.45$ for generating the plots in Figure 7.

Experiments involving the increase in values such as γ led to fast convergence but with oversharpening, while very low values of γ led to slower convergence, increasing the number of iterations.

Reduction in the values of κ_f, κ_b, w, and α led to milder effect of the sharpening/smoothing while increase led to oversmoothing and mild sharpening. Moreover, as noted earlier, the BDF and FDF must be carefully balanced to achieve results and this varies from image to image. Thus, we chose to maintain the aforementioned values and adjust only the γ factor to reduce uncertainties and run-times for various images.

Unfortunately, the runtime increases with the Laplacian, which we may omit for speed and quick convergence. Thus, three aspects that the formulation addresses are the following:

(i) The intensity contrast problem

(ii) The noise and texture problem

(iii) The crack edge enhancement problem

3.4. Local Maximum Gradient- (LMG-) Based Thresholding. We utilize the edge image and by-pass conventional thresholding stage by exploiting the information obtained from the edge pixel profile of the crack images. We outline the simple thresholding method as follows.

(i) *Step 1*: compute global maximum of the edge image.

(ii) *Step 2*: loop the image, using a local defined window, w of a known size.

(iii) *Step 3*: if a pixel in the local region matches the global maximum gradient, set the pixel to 1 (or 255 for display purposes).

(iv) *Step 4*: else set all the nonmatching pixel values in the local region to 0.

The scheme is similar to the black pixel extraction method by Tanaka and Uematsu [19] but without numerous manually set parameters or steps. The results of this simple approach are shown in Figure 8 for a sample of crack images where pixel intensities are distinct. However, it fails for some of the more difficult and intricate crack patterns as mentioned earlier. Thus, we compute the edge image prior to performing this step to improve results. Since the illumination correction algorithm does not generally improve results for such problem images, it is omitted here. However, we can also automate this process by analyzing the contrast, variance, or gradient of the crack images and process accordingly based on a threshold-based decision. Alternatively, we use edge detection filters such as Sobel and Prewitt.

4. Experiments and Results

We present a comparative analysis of the proposed algorithm with other algorithms from the literature to test the efficacy of the approach. The algorithms were executed on PC system running MATLAB for several crack images from the chosen database [14, 15].

4.1. Batch Processing Using Image Dataset. We test the algorithms using the image dataset [14, 15] consisting of 155

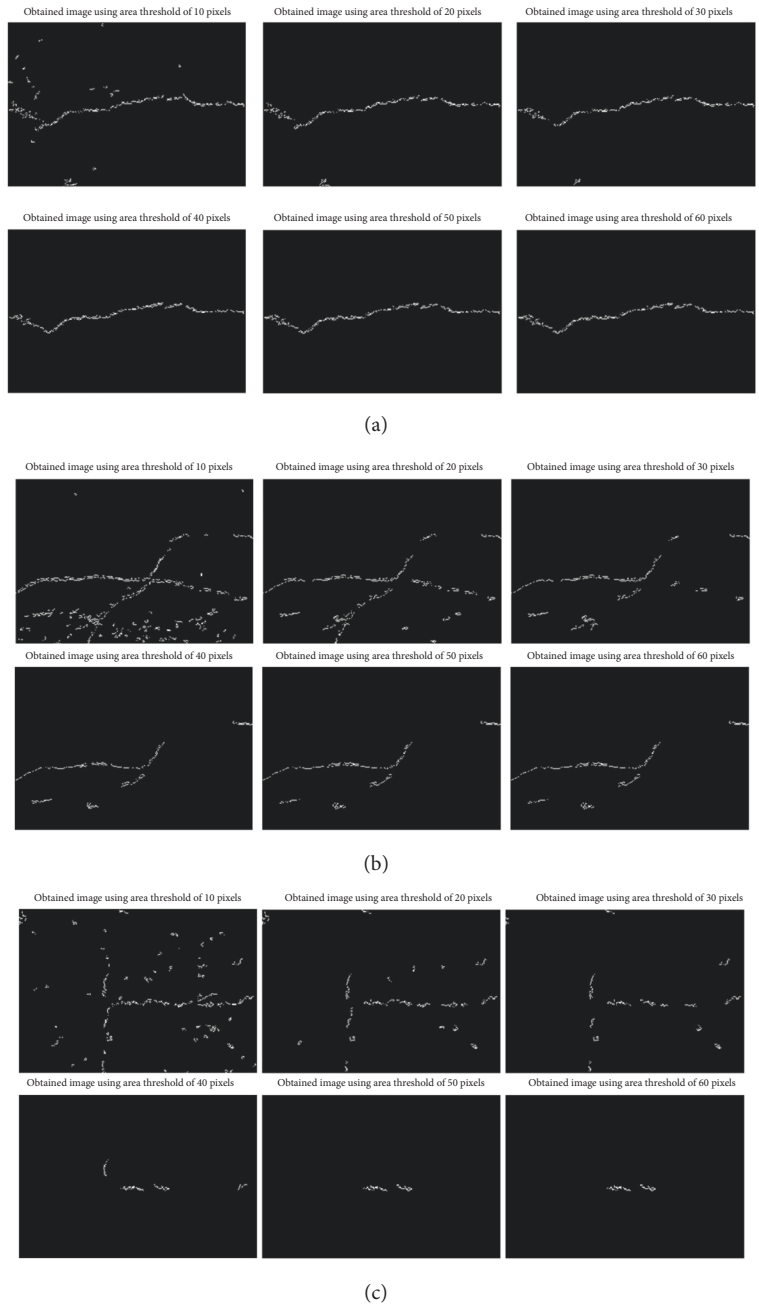

Figure 8: Detected crack regions using varying thresholds using the simple thresholding scheme for (a) clear (b) moderate and (c) difficult crack images.

images shown in Figure 9(a) while the processed images for PA-1 and PA-2 are shown in Figures 9(b) and 9(c), respectively. Based on visual evaluation, we can see that PA-2, though relatively simpler, actually detects most of the cracks in the images from the dataset compared to PA-1. However, both contain unwanted information in some cases and this can be improved by additional morphological postprocessing operations. However, the key aspect of the proposed scheme is the enhancement of crack regions. Another issue is the one of image resolution, where large crack images have to be reduced in size to enable faster processing, leading to the need to recalibrate the size of filtering neighbourhood for local operations. The resizing appears to affect the results in terms of very fine crack features.

4.2. Comparison and Discussion. We compare image results from [9–12] with the proposed algorithms. The first and second columns of images in Figures 10 and 11 are from both [10, 11] (darkest image) while the third and fourth columns represent those of PA-1 and PA-2. The proposed approaches achieve crack detection in most cases even without additional processing unlike the other more involved methods from

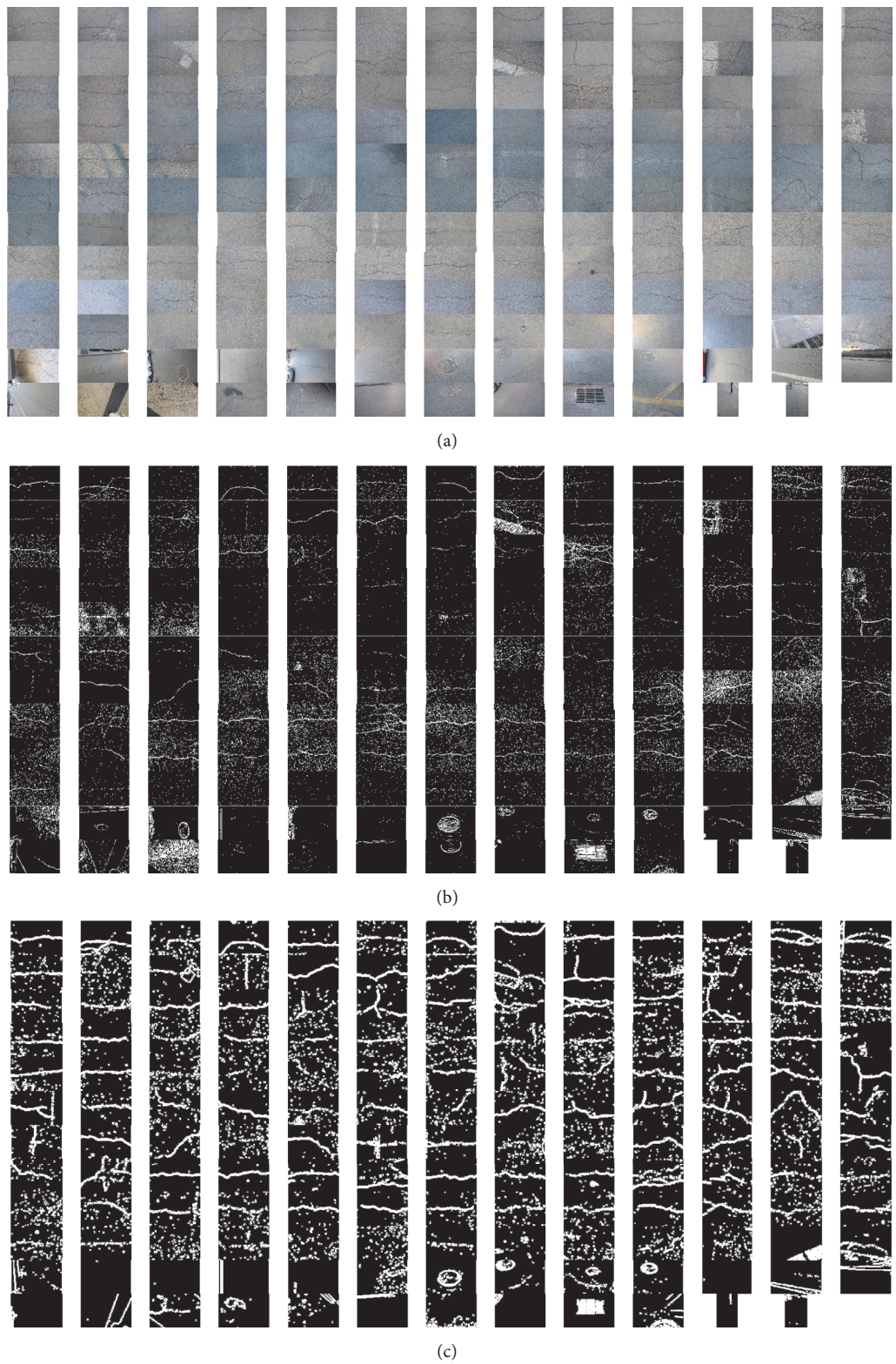

Figure 9: (a) Original crack images from database processed using (b) PA-1 and (c) PA-2.

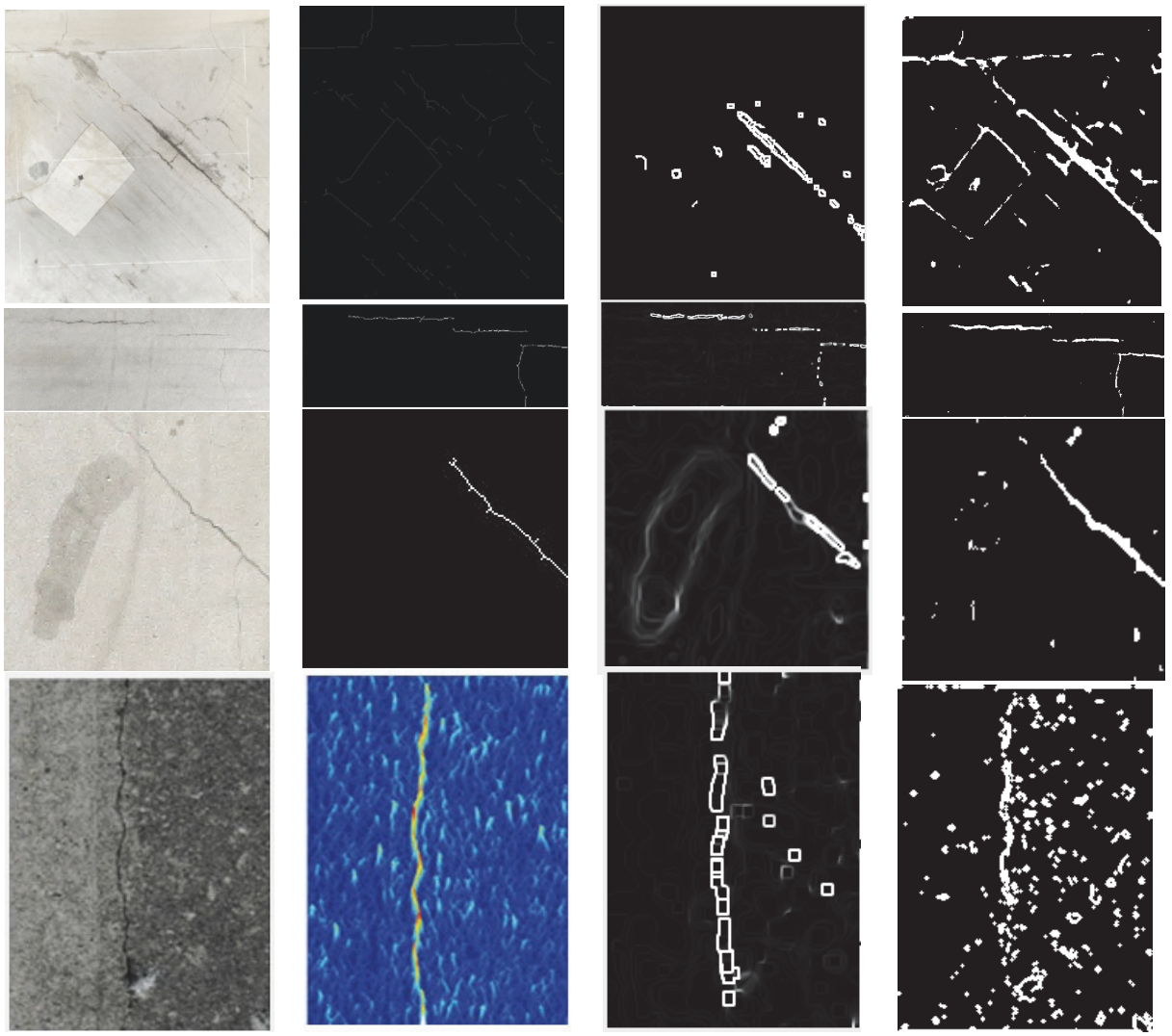

Figure 10: (a) Original category 1 images (b) results from methods by [10, 11]: (c) PA-1 and (d) PA-2.

[10, 11]. Additional processing improves the highlighting of the prominent crack region while eliminating spurious information.

In Figure 12, PA-2 is compared to the methods by Oliveria [12], while Figure 13 shows comparisons with algorithms by method by Talab et al. [9, 20] and Dorafshan [9]. In Figure 12, we show the results from Oliveira prior to the crack refinement algorithm for a fairer comparison. The images show that PA-2 yields cracks with the most similar width to that of the cracks in the original image. We also show the results after some additional postprocessing and compare results with the Oliveira's crack refinement algorithm [12] and method by Talab et al. [20] and Dorafshan et al [9], whose algorithm is based on an improvement of algorithm by Talab et al. The improved PA-2 result is shown in Figure 13(e).

In Figure 14, results are shown for methods by Talab and Dorashan et al. and PA-2 and visual analysis indicates that PA-2 yields the best crack detection result, though there is added noise. However, this is impressive considering that the method by Dorafshan et al. involves additional morphological postprocessing steps to obtain the final result. Additionally, PA-2 does not employ Otsu thresholding unlike the other algorithms.

The local maximum gradient is a relatively simple technique, showcasing the effectiveness and influence of the enhancement stage on the final results. Furthermore, the test images used in these other algorithms are relatively ideal compared to those from the utilized dataset, which have more intricate crack patterns and rougher textures. The backgrounds of the images tested with these algorithms also do not possess uneven illumination or shadows and are mostly ideal crack images with cracks clearly distinct from the background.

In Figure 14, we extensively compare the method by Dorafshan et al., which is claimed to be superior to the

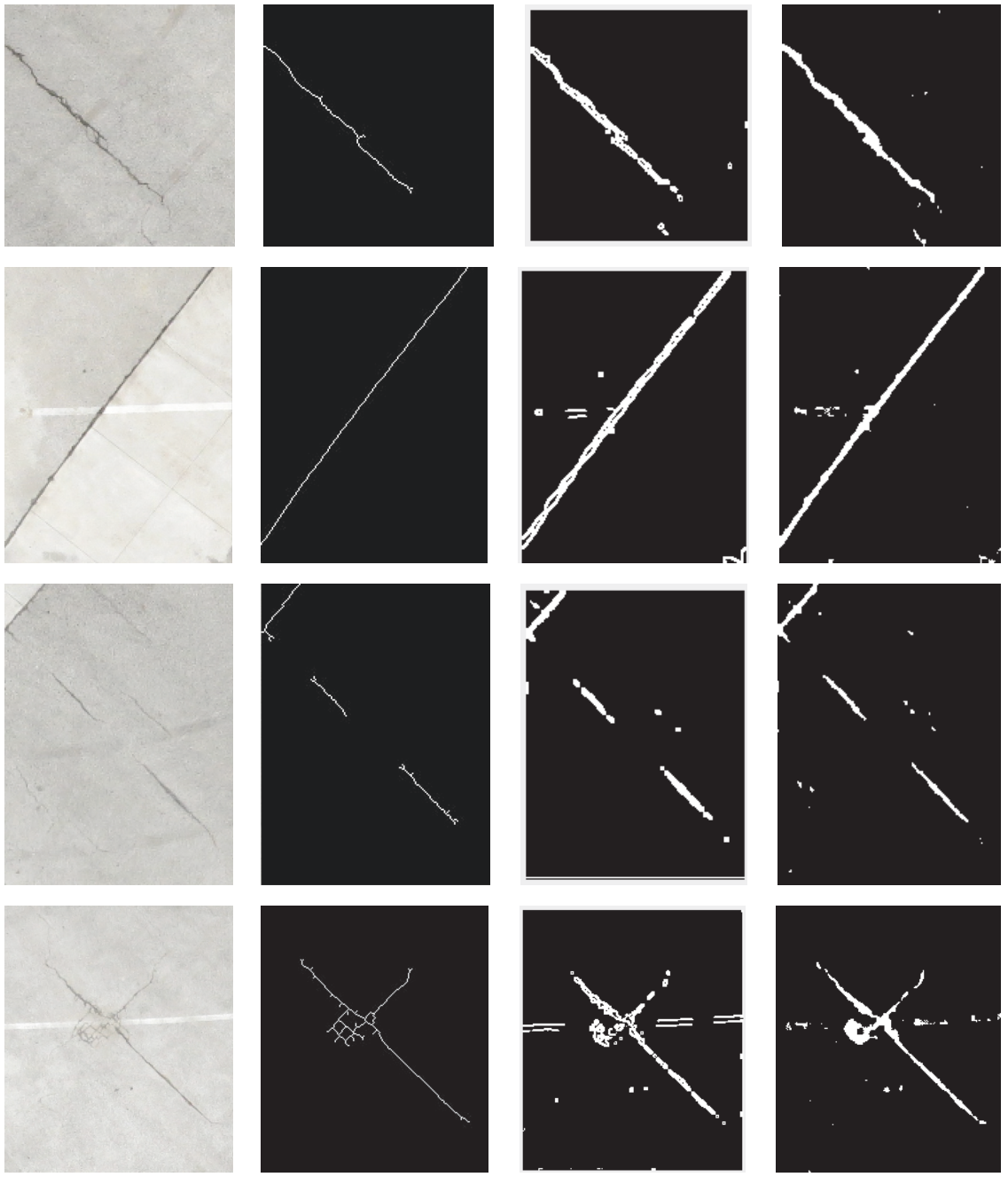

Figure 11: (a) Original category 2 images (b) results from method by [10]: (c) PA-1 and (d) PA-2.

method by Talab et al. The results are compared with PA-2 with and without additional morphological operations. Results indicate that PA-2 always yields a much more complete crack profile from the original image than the method by Dorafshan et al. This is due to the simultaneous edge sharpening and smoothing by the FAB diffusion. Note that the method by Dorafshan et al. results in breakages along the line of the crack pattern, whereas PA-2 links and joins edges due to edge enhancement and smoothening of the background to reduce interference in the tracing of the crack path during thresholding.

For the images with very rough textures, there is more noise after processing but still not usually enough to obscure the crack pattern. These noise points can be removed by additional morphological processes. However, the emphasis on the level of success obtained due to the proposed pre-processing scheme is clearly observed in most of the image results.

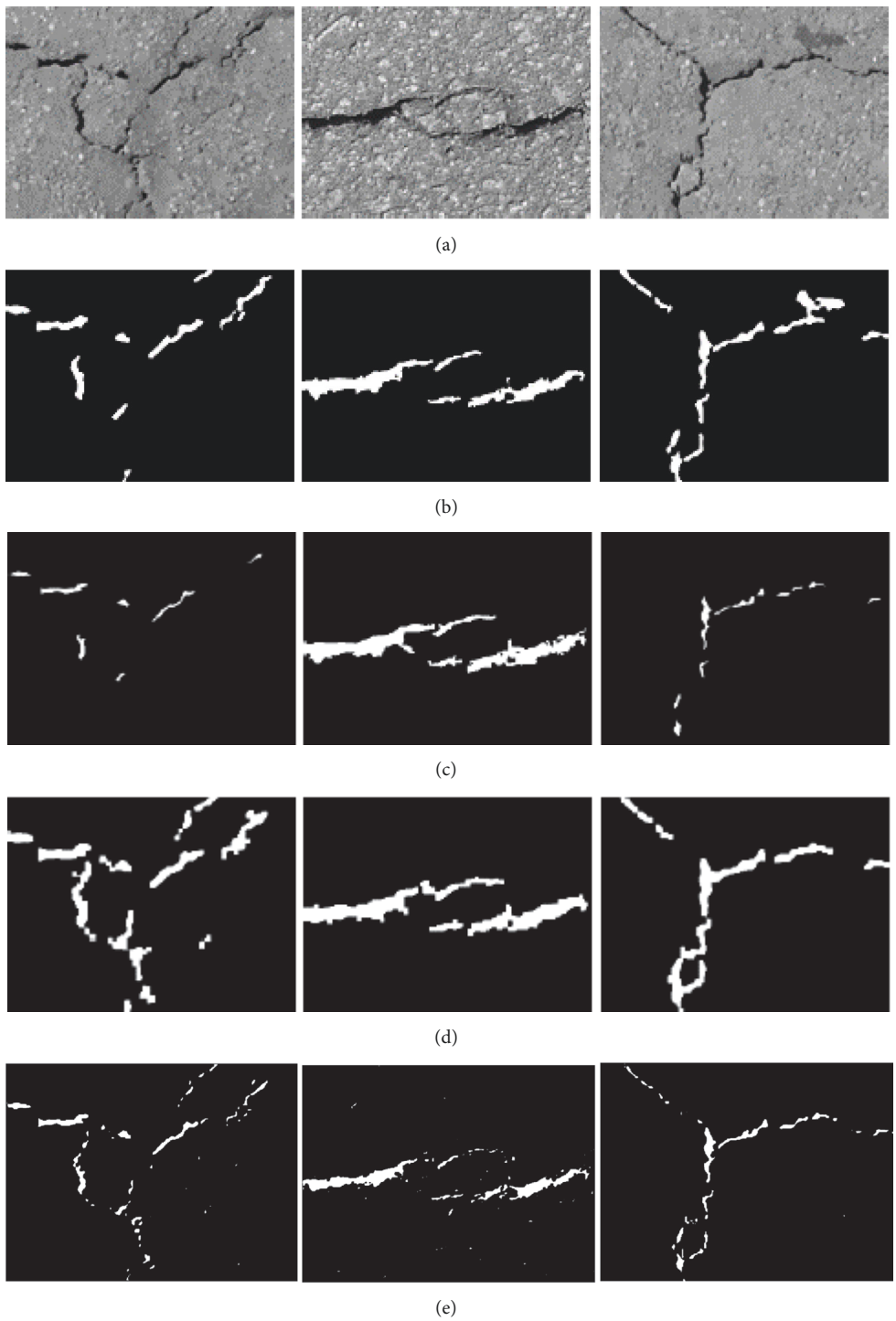

Figure 12: (a) Original crack images from [12] processed using (b) AD [12], (c) wavelets [12], (d) morphological operators [12], and (e) PA-2.

In future work, we hope to improve the results of the proposed algorithm for processing difficult crack images with multiple disjointed localized cracks. Also, we will explore nonlinear edge detectors and compare results. At the moment, the proposed algorithm has a reasonable execution time in spite of the iterative process, though image size and resolution are also a vital factor. However, utilization of nonlinear, highly computationally involved edge and morphological operators will considerably increase the runtime regardless of image size or resolution. We experimented with resizing images prior to crack detection and, though results are good, this affects accuracy in the case of ground truth images unless calibration is performed with a particular image size and resolution. Furthermore, given that the

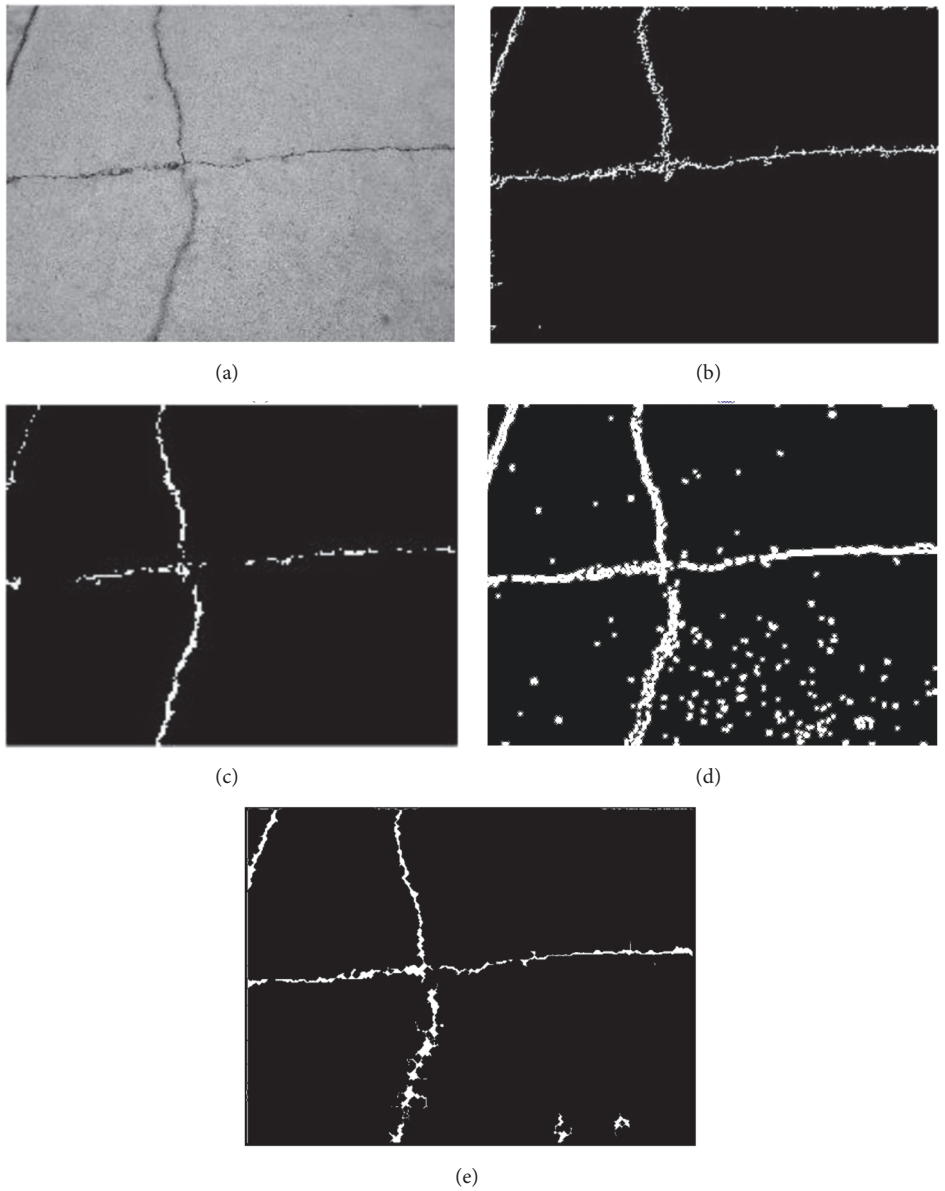

Figure 13: (a) Original crack image from [9] processed using algorithms by (b) Talab, (c) Dorafshan et al. [9], (d) PA-2, and (e) improved PA-2.

various datasets are of varying image dimensions and resolutions and the unavailability of the software implementation of several of these algorithms, it is difficult to perform a universal comparison. However, image preprocessing, which is the main thrust of the algorithm in this paper is important in the final result of crack detection algorithms. Thus, the processing of these images to remove spurious features, while enhancing the desired features has been achieved.

5. Conclusions

In this work, an adaptive PDE-based preprocessing algorithm for crack images was proposed, implemented, and combined with a simple, local maximum-based thresholding scheme. Experimental results indicated that automated operation, selective sharpening, and smoothing were achieved using the proposed approach, leading to better crack detection. The adaptive nature of the scheme ensures practical operation when processing a large batch of images as observed in experiments using a dataset. This would enable the system to be easily adopted in the inspection of concrete structures for continuous and fast operation. Future work will involve improving the enhancement and segmentation results, exploration of better PDE-based formulations, and additional morphological operations to achieve these objectives.

Data Availability

The data used for experiments were obtained from the following sources [14, 15].

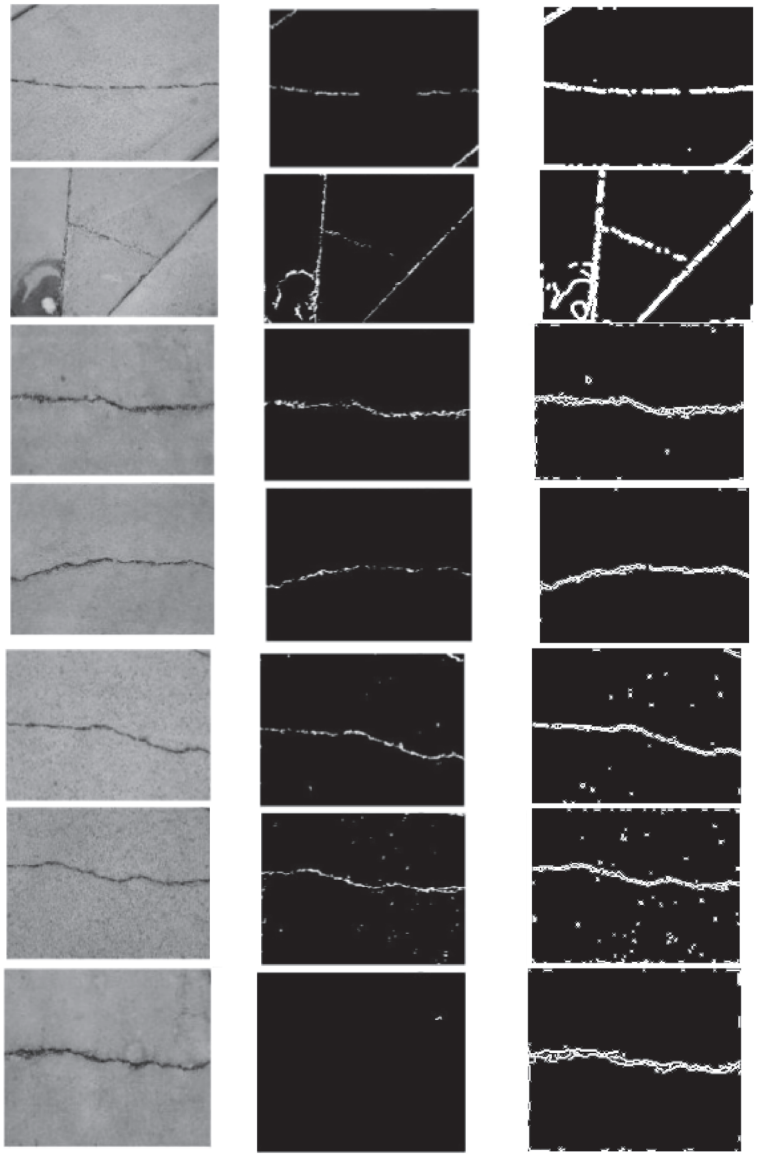

Figure 14: (a) Original crack images from [9] processed using algorithms by (b) Dorafshan et al. [9] and (c) PA-2.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding the publication of this paper.

References

[1] A. Mohan and S. Poobal, "Crack detection using image processing: a critical review and analysis," *Alexandria Engineering Journal*, pp. 1–12, 2017.

[2] R. Medina, J. Llamas, J. Gómez-García-Bermejo, E. Zalama, and M. Segarra, "Crack detection in concrete tunnels using a gabor filter invariant to rotation," *Sensors*, vol. 17, no. 7, article no 1670, 2017.

[3] G. Sapiro and V. Caselles, "Histogram modification via differential equations," *Journal of Differential Equations*, vol. 135, no. 2, pp. 238–268, 1997.

[4] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, *Digital Image Processing Using MATLAB*, Prentice Hall, Upper Saddle River, NJ, USA, 2004.

[5] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.

[6] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: Nonlinear Phenomena*, vol. 60, no. 1–4, pp. 259–268, 1992.

[7] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[8] U. A. Nnolim, "Implementation of a Java-based software application of a proposed crack detection algorithm for concrete surfaces," in *Proceedings of the 1st International Conference Faculty of Engineering University of Nigeria*, pp. 1–10, Nsukka, Enugu, Nigeria, 2018.

[9] S. Dorafshan, M. Maguire, and Q. Xiaojun, "Automatic surface crack detection in concrete structures using otsu thresholding and morphological operations," Technical Report, Utah State University, 2016.

[10] DroneYard-Experiments in UAS-Remote-Sensing, "Automated Optical Crack Detection in Pavement. Website," 2016, http://www.droneyard.com/2016/07/22/automated-optical-crack-detection-in-pavement/.

[11] S. Kaneko, S. Oka, and N. Matsumiya, "Detection of cracks in concrete structures from digital camera images," *NTT Technical Review: Imaging and Sensing Technologies for Safety and Security*, vol. 10, no. 2, pp. 1–5, 2012.

[12] H. Oliveira and P. L. Correia, "Road surface crack detection: improved segmentation with pixel-based refinement," in *Proceedings of the 25th European Signal Processing Conference (EUSIPCO)*, pp. 2080–2084, Greece, September 2017.

[13] U. Nnolim, "A fuzzy homomorphic algorithm for image enhancement," *Nigerian Journal of Technology*, vol. 34, no. 1, pp. 156–163, 2014.

[14] L. Cui, Z. Qi, Z. Chen, F. Meng, and Y. Shi, "Pavement distress detection using random decision forests," in *Proceedings of the International Conference on Data Science*, vol. 9208, pp. 95–102, 2015.

[15] Y. Shi, L. Cui, Z. Qi, F. Meng, and Z. Chen, "Automatic road crack detection using random structured forests," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 12, pp. 3434–3445, 2016.

[16] E. Etienne, M. Kerre, and M. Nachtegal, *Fuzzy Techniques in Image Processing, Studies in Fuzziness and Soft Computing*, Springer-Verlag, Berlin, Germany, 2000.

[17] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Real and complex pde-based schemes for image sharpening and enhancement," *Advances in Imaging and Electron Physics*, vol. 136, pp. 1–109, 2005.

[18] X. Shen, Q. Li, Y. Tian, and L. Shen, "An uneven illumination correction algorithm for optical remote sensing images covered with thin clouds," *Remote Sensing*, vol. 7, no. 9, pp. 11848–11862, 2015.

[19] N. Tanaka and K. Uematsu, "A crack detection method in road surface images using morphology," in *IAPR Workshop on Machine Visual Applications*, pp. 154–157, Makuhari, Chiba, Japan, 1998.

[20] A. M. A. Talab, Z. Huang, F. Xi, and L. Haiming, "Detection crack in image using otsu method and multiple filtering in image processing techniques," *Optik - International Journal for Light and Electron Optics*, vol. 127, no. 3, pp. 1030–1033, 2016.