A Novel Adaptive Pixels Segmentation Algorithm for Pavement Crack Detection

Nima Safaei*
Graduate Student
Department of Civil, Construction and Environmental Engineering
Iowa State University, Institute for Transportation
2711 S Loop Dr.
Ames, IA 50010
Phone: (515) 715-3804
Email: nsafaei@uiowa.edu

Omar Smadi
Associate Professor
Department of Civil, Construction and Environmental Engineering
Iowa State University, Institute for Transportation
2711 S Loop Dr.
Ames, IA 50010
Phone: 515-294-7110
Email: smadi@iastate.edu

Babak Safaei
Graduate Student
Department of Civil and Environmental Engineering
Michigan State University
474 S Shaw Ln, East Lansing, MI 48824
Phone: 517-505-7215
Email: safaeiba@msu.edu

Arezoo Masoud
Graduate Student
Tippie College of Business
University of Iowa
21 E Market St, Iowa City, IA 52242
Phone: 515-708-6253
Email: amasoud@uiowa.edu

Word Count: 6,318 words + 2 table (250 words per table) = 6,818 words

Submitted [12/01/2020]

*The author is currently a Ph.D. student in Information Systems at the University of Iowa.

Declarations

Funding

The research was conducted at the Institute for Transportation of Iowa State University but was not directly funded.

Conflicts of interest/Competing interests

There is no conflict of interest associated with this study.

Availability of data and material

The data and pre-print version of this study is available and can be accessed through:
https://drive.google.com/file/d/1U8Ie-H-JYZUU5lSN0IXEiNf_S64g4kv8/view?usp=sharing (*1*)

Code availability

The custom code for this study is written in Matlab and will be provided upon request.

ABSTRACT

Cracks considerably reduce the life span of pavement surfaces. Currently, there is a need for the development of robust automated distress evaluation systems that comprise a low-cost crack detection method for performing fast and cost-effective roadway health monitoring practices. Most of the current methods are costly and have labor-intensive learning processes, so they are not suitable for small local-level projects with limited resources or are only usable for specific pavement types.

This paper proposes a new method that uses an improved version of the weighted neighborhood pixels segmentation algorithm to detect cracks in 2-D pavement images. This method uses the Gaussian cumulative density function as the adaptive threshold to overcome the drawback of fixed thresholds in noisy environments. The proposed algorithm was tested on 300 images containing a wide range of noise representative of different noise conditions. This method proved to be time and cost-efficient as it took less than 3.15 seconds per 320 × 480 pixels image for a Xeon (R) 3.70 GHz CPU processor to determine the detection results. This makes the model a perfect choice for county-level pavement maintenance projects requiring cost-effective pavement crack detection systems. The validation results were promising for the detection of low to severe-level cracks (Accuracy = 97.3%, Precision = 79.21%, Recall= 89.18% and F_1 score = 83.9%).

Keywords: Pavement Crack Detection, Pixel Segmentation, Pavement Management System

INTRODUCTION

Roads provide an effective way of transportation between different locations for various vehicle types. Modern societies are heavily dependent on transporting people and cargo through roads; this imposes a continuous degradation to pavement surfaces and necessitates establishing a robust pavement maintenance system. Maintaining pavement surface conditions in high quality will mitigate the dangers associated with poor pavement conditions and improve road safety, among the other factors (*2–4*). In general, pavement surfaces are classified into several groups based on ingredient materials and texture composition. Two primary groups of pavements are Asphalt Concrete (AC) and Portland Cement Concrete (PCC) pavements. They consist of mineral aggregates bound together with either asphalt or cement water past respectively, which provides each type of pavements with unique characteristics.

Pavement surfaces are often subjected to increasing challenges of direct vehicle forces (vertical, shear, and radial), deterioration, and fatigue stresses, leading to the formation of cracks on their surface. Also, factors such as aging and environmental conditions, temperature, wind, relative humidity, and sunlight are effective in the formation of cracks (*5*). Cracks make the water penetrate the surface and damage it by the process of freezing and thawing. Roads that approach the end of their service lives are frequently rehabilitated by removing the old surface and replacing it with new materials (*6*). If cracks that appear in the pavement surface are not treated, they spread and get worse with excessive traffic growth, pavement aging, road widening, extreme events, and debonding, which endanger the road's life performance (*7–9*). The pavement deterioration problem worsens with insufficient maintenance budgets. Globally, every year more than $400 billion is spent on pavement construction and maintenance costs; thus, reducing the inspection-related expenditures and maintenance costs is the primary goal of researchers working in this field (*10*, *11*). Early detection of cracks can prevent potential damage and failure (*12*). Building a fast, robust, and cost-effective image processing algorithm for detecting the pavement surface cracks is one of the most critical priorities for building a robust pavement management system (*13–17*).

Pavement distress surveys typically are conducted with one of the three approaches: manual, semi-automated, or fully automated. In manual distress surveys, raters perform either a visual study of pavement distresses by walking along the pavement surface or conducting a windshield survey from a slow-moving vehicle. In semi-automated evaluations, raters use a point and trace manual method at an office workstation to evaluate the type, extent, and severity of pavement surface distresses one image at a time. Previously, most state highway agencies (SHAs) used the semi-automated approach (*18*, *19*); however, as the method was time-consuming, especially for massive projects and network-level evaluations (*20*), it is not used anymore. Fully automated distress evaluations are conducted with image processing and pattern recognition software for distress identification and quantification. Raters conduct quality assurance testing of the software functionality and perform quality control of the distress rating output (*21*). Automated pavement assessment provides a fast, accurate, and non-subjective alternative to manual inspection surveys. They can deliver a quantitative analysis of the pavement condition compared with the traditional manual methods, which were mostly qualitative. Automated pavement assessment requires an automatic data collection system. Vavrik et al. indicated that as of 2012, most state agencies (more than 35 of them) used fully automated image analysis methods for pavement distress data collection systems (*22*).

State highway agencies routinely employ high-speed data collection vehicles for capturing the network level pavement images. The process's challenges are the random and irregular size of cracks, noise caused by variable lighting conditions, shadows, stains, and other external objects present on the surface. The following general steps are typically followed to conceptualize an algorithm for automated pavement distress surveys (*23–25*):

1- Initial processing: it improves and enhances the image quality and makes it ready for the next step. This step is mainly based on contrast stretching and histogram equalization techniques, and it is used to reduce the effects of shadows, equalize pavement texture variations, and reduce the contrast between different pavement areas.

2- Pre-processing: it removes noise that otherwise might be misunderstood as cracks. This step reduces false positive occurrences in the final segmentation results.

3- Pre-filtering: it identifies parts of the image with high chances of crack occurrence and limits the detection procedure for those areas; doing this reduces the computational expenses and makes the process more efficient.

4- Crack detection: it uses a combination of image processing techniques to separate the crack information from the rest of the image (*26*).

5- Post-processing: it is based on morphological and connectivity searching operations. Its goal is to reduce the number of false cracks detected previously and join fragmented crack regions to make groups of connected pixels with darker intensities than their neighboring areas.

The rest of this paper is organized as follows: "Literature Review" section represents a brief review of the past and current approaches. "Methodology" describes the proposed framework, along with presenting a brief description of the method. "Numerical Results" presents the detection results and compares them with some other studies' detection power in the literature. Finally, the "Conclusion" wraps up the finding and makes some suggestions for future studies.

LITERATURE REVIEW

Studies reported in the literature have focused on the development of automated image-based pavement crack detection methods, which could broadly be classified into four groups (*27*):

1- Thresholding and filtering-based methods are among the oldest and most popular methods used for crack detection. A threshold-based method can be based on a histogram analysis with the Gaussian hypothesis with adaptive or local thresholding. Ayaho et al. used the average pixel intensity values in each row of the input image as a threshold value for comparing it with other pixels. They labeled the ones with huge differences as potential crack pixels (*28*). Wang and Tang proposed a new pixel thresholding approach and compared it with the global and local threshold value algorithms (*29*). The main problem of thresholding-based methods is that they do not consider the cracks' geometric and photometric characteristics. This method also assumes that the pixel intensity distributions for the background pavement and the crack area can be separated based on global level statistics, which sometimes results in inaccurate measurements.

2- Tanaka and Uematsu introduced mathematical morphological operations in 1998 (*30*). They defined cracks as the progression of saddle points with linear features. The quality of the processes in this method is highly dependent on parameter selection, which limits their practical application; however, if used as the preliminary stage for the thresholding methods, the accuracy would improve considerably. He and Qiu presented an enhanced segmentation approach based on a combination of multidirectional and multi-threshold averaging morphologies (*31*). They showed that their proposed algorithm significantly improves the segmentation process and works effectively in noisy environments.

3- Model analysis can result in accurate crack modeling. Kass et al. proposed the minimal path selection model in 1988; they provided both the endpoints of a curve to the model and extracted simple open curves (cracks) from the image (*32*). This algorithm was then improved by Kaul et al. and Amhaz et al. in 2012, 2014; they reduced the need for prior knowledge about the topology and the desired curve endpoints (*33, 34*). In 2016, Amhaz et al. proposed an improved training-based minimal path selection method for asphalt crack detection. The idea was based on the fact that the lowest cost function belongs to a minimal path within a crack (*35*). P. Delagnes used Markov random field (MRF) model to detect poorly contrasted cracks in textured areas (*36*). Svenson et al. used finite mixture models to identify pavement segments with the most rapid deterioration rate and implemented them for making more efficient maintenance decisions (*37*). According to Shi et al. (*38*), the model-based methods ignore the local neighbor information, and they cannot precisely detect cracks with intensity inhomogeneity.

4- Learning-based methods have contributed significantly to roadway crack detection in recent years (*39, 40*). Recently, the image data size has increased considerably, and concurrently the computation power of computers has soared up. In 1998, Cheng and Miyojim used a neural network to select the appropriate threshold for distress detection. They used the standard deviation as a parameter in the training

of the neural network (*41*). Liu et al. pre-defined potential crack features based on the pixel intensity values and used a support vector machine (SVM) classifier (*42*) for detecting cracks. Later in 2006, Abdelqader et al. used an unsupervised Principal Component Analysis (PCA) to reduce the feature vector dimensions based on region values and extract cracks from PCC images (*43*). Deep learning is a promising branch of artificial intelligence that is widely used in different engineering disciplines. Deep learning techniques can be successfully implemented in crack detection problems. Deep convolutional neural networks (CNNs), which are extensively used for image classification (*44*), can be implemented in crack detection problems on pavement surfaces (*45*, *46*).

In recent years, crack detection algorithms were trained either by using object detection methods such as OverFeat, SSD, Yolo, Faster-RCNN, Mask-RCNN (=Faster-RCNN + FCN) or segmentation methods such as SegNet (*47*). In 2017, Gopalakrishnan et al. proposed a deep CNN and trained it on the massive ImageNet database for automatically detecting cracks in AC and PCC surfaces (*48*). The main drawback of CNNs is the vast amount of data required for training the model (*49*). In 2014, Wu et al. proposed a novel crack-defragmentation technique and named it MorphLink-C. The method's primary objective was to connect crack fragments; the method consisted of two sub-processes, including fragment grouping using the dilation transform and fragment connection using the thinning transform by an artificial neural network (ANN) (*50*). Learning-based methods have proven to have substantially better accuracy than the previously-mentioned algorithms; however, the main disadvantage of learning-based methods is that labeling the data to train the algorithm is a labor-intensive and error-prone procedure. Most of these methods are computationally expensive and require a vast amount of resources and budgets; these problems necessitate the development of more cost-effective methods, especially for smaller and county-level pavement crack detection systems, which is the primary goal of this study.

METHODOLOGY

This study aims to detect pavement cracks using an improved version of the weighted neighborhood pixels segmentation method based on the Gaussian cumulative density function as an adaptive threshold. First, using the initial processing (image enhancement) and pre-processing techniques, the noise was removed, and the luminosity condition was unified throughout the input image surface. Second, using the proposed segmentation algorithm, the image was binarized, and the probable crack regions were extracted. Next, by applying crack region property features, non-crack areas that resulted in false positive detections were removed. Finally, by taking advantage of morphological operations, a closing operator was used at the post-processing stage to connect the fragmented pieces of cracks (**Figure 1**).

There are two groups of datasets used in this study. The first dataset is the Crack Forest Dataset (CFD), which comprises 118 publicly available images. The dataset can generally reflect urban road surface conditions in Beijing, China. The CFD data set images were taken by an iPhone 5 with a focus of 4 mm, the aperture of f/2.4, and exposure time of 1/134 s. The CFD contains many noise features such as shadows, oil spots, and water stains. The second dataset is the MTC dataset owned by Midwest Transportation Center (MTC). MTC is one of the leading regional University Transportation Centers and is sponsored by the US Department of Transportation Office of the Assistant Secretary for Research and Technology (USDOT/OST-R). MTC is funded by the Moving Ahead for Progress in the 21st Century Act (MAP-21) federal transportation bill. One of the critical focus areas of the MTC's research is data-driven performance measures of transportation infrastructure, including pavement monitoring and maintenance.

The MTC dataset images are captured by a Laser Road Inspection System (LRIS) composed of two high-resolution line scan cameras and high-power lasers from the highway infrastructures at Illinois and Iowa states. A subset of 80 images from the CFD dataset combined with 220 images from the MTC dataset was used in the study. The reason for selecting a combination of two data sets was to assess the algorithm's strength in detecting cracks from low-cost smart phone-captured images (CFD) in addition to the network-level pavement images (MTC) that are costly and taken by special data collection vehicles. Also, the CFD dataset consists of a wider variety of crack types and represents the method's capability in detecting a more

comprehensive range of pavement cracks. MTC's original image dimensions were 3072 × 2048 pixels, so they were divided into 320 × 480 pixels to have a uniform size with the CFD images. Each image in the CFD has a manually labeled ground truth contour produced by Matlab (*38*, *51*). For the polygonal annotation of cracks in this study, Labelme, a graphical image annotation tool inspired by the Massachusetts Institute of Technology, was used. Labelme is coded in Python and uses Qt for its graphical interface.

Image Enhancement and Pre-processing

Since pavement images are captured under non-uniform illumination distribution, image enhancement is an essential step of image analysis. The pre-processing technique is also implemented to reduce the noise and enhance the desired image features like dark linear features. First, it is necessary to convert all input images to the same lighting condition. The histogram equalization technique was used to compensate for the luminosity problem during the capturing process. The existing differences in illumination are corrected using **Equations 1** and **2**.

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1 \tag{1}$$

$$p_{in}(r_k) = \frac{n_k}{n} \tag{2}$$

Where 's_k' is the image's accumulated normalized histogram. P_{in} (r_k) is the image's histogram for the pixel value 'r_k.' 'L' is the total number of gray level pixels in the image. 'n' is the total number of pixels in the image, and 'n_k' is the number of pixels with grayscale value 'r_k.'

After luminosity correction, the bottom 1% and the top 1% of all the pixel values were saturated, and the image contrast was increased. A sample image is illustrated in **Figure 2** with its enhanced version. Based on **Figure 2**, cracks are more discernible in (b) than (a).

An adaptive median filtering was implemented for the first step of the pre-processing stage. It works by changing the neighborhood's size during the operation and, consequently, reduces distortions such as excessive thinning of object boundaries. The adaptive median filter has some benefits over the standard median filter; it preserves detail while smoothing non-impulse noise (*52*), and an unrepresentative pixel value cannot divert the filter values (*53*). Through this method, it is determined that which pixels are affected by the impulse noise. Each pixel is compared to the surrounding pixels; if it is different from most of them, or if it is not structurally aligned with the similar surrounding pixels, that pixel is classified as impulse noise. The impulse noise is replaced by the median pixel value of the surrounding pixels. The algorithm for selecting the appropriate threshold value is shown in **Figure 3** (The algorithm was proposed by Peng (*52*)).

Z_{min} is the minimum pixel value in S_{xy}, Z_{max} is the maximum pixel value in S_{xy}, Z_{med} is the median of pixel values in S_{xy}, Z_{xy} is the pixel value at (x, y) coordinates, and S_{max} is the size of S_{xy}.

Based on **Figure 3**, in level 'A,' if $Z_{min} < Z_{med} < Z_{max}$ is true, then Z_{med} is not the impulse noise, and the algorithm moves to level 'B' to test whether Z_{xy} is an impulse noise. If $Z_{min} < Z_{med} < Z_{max}$ is not valid, then Z_{med} is impulse noise, so the size of the window is increased and level 'A' is repeated until Z_{med} is not an impulse noise and the algorithm moves to level 'B,' or the window size is reached to the maximum value, and the algorithm output is Z_{xy}. In level 'B,' if $Z_{min} < Z_{xy} < Z_{max}$ is true, then Z_{xy} is not an impulse noise, and the algorithm's output is Z_{xy}. If $Z_{min} < Z_{xy} < Z_{max}$ is not valid, then Z_{med} is not an impulse noise, and the algorithm's output is Z_{med}.

For further pre-processing, the K-SVD image denoising method was applied to the image. K-SVD is a signal representation method that utilizes a sparse combination of atoms and derives a dictionary from a group of signals to approximate each of them. The efficiency of the dictionary is optimized by taking advantage of the non-local similarities of the image. The K-SVD algorithm consists of three main steps:

1- Sparse coding is computing the sparse approximation of all image patches with a fixed size using the initial dictionary.
2- Dictionary Update is done to improve the quality of the sparse approximations.
3- Reconstruction Step builds the denoised image from the denoised patches (*54*, *55*).

Figure 4 shows the structure of the implemented K-SVD approximate algorithm. **Figure 5** shows the image output after enhancing the image and removing noise using the adaptive median filtering and K-SVD denoising algorithm.

Feature Extraction and Image Segmentation

Image segmentation is a technique for extracting a target object (e.g., cracks) from the background. As mentioned earlier, an improved version of the weighted neighborhood pixels segmentation method was used to enhance crack detection accuracy in noisy environments. Generally, cracks have a series of unique structural properties that differ from noise. Based on this idea, a novel crack descriptor was proposed by using the Gaussian adaptive threshold in the weighted neighborhood pixels segmentation method.

The neighborhood pixel orientation of the weighted neighborhood algorithm is shown in **Figure 6**, where 'C' represents the central pixel. A pixel can have 8, 16, and 24 neighbors in the first, second, and third circles surrounding it, respectively, represented by stars, dashes, and cross symbols in **Figure 6**. The pixel's attributed weight was calculated based on the pixel values around the pixel (**Equation 3**).

$$w = \frac{G_{total}}{G_{max}} \tag{3}$$

Where 'w' is the weight of the neighborhood pixels; 'G_{total}' represents the sum of all pixel values in three surrounding loops, and 'G_{max}' is the maximum achievable pixel value of all pixels among the three surrounding loops.

Different pavement surface distresses have different patterns with specific characteristics; so, a robust feature extraction algorithm needs to be calibrated for specific types of pavement cracking. There are different patterns selected for assigning neighborhood pixel weights in the literature such as 'four main far pixels', 'four partial far pixels', 'eight-direction far point pixels', 'four main direction pixels', 'four partial-direction pixels', 'Kumar pattern' (*56*) and 'eight-direction pixels'. In this paper, the eight-direction pattern was used as the main pattern of the algorithm. In the original method proposed by Sun et al. (*57*), a local sliding window was applied to all pixel locations to generate the averaged pixel value of the pre-processed image (I_{avg}) (**Equation 5**). Using **Equation 4**, the average pixel value was compared with the weighted pixel value (W_c). If the weighted pixel satisfied **Equation 4**, the pixel was labeled with 1, and it was identified as a crack. Otherwise, it was labeled with 0 and identified as a background pixel.

$$w_c \leq 0.85 \times I_{avg} \tag{4}$$

Where

$$I_{avg} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} I(i,j) \tag{5}$$

I (i, j) is the pixel value of the pre-processed image at row 'i' and column 'j,' 'M' is the number of rows, and 'N' is the number of columns in the sliding window.

The above approach's main drawback is using a fixed threshold (0.85), which is not appropriate for all the images. In this paper, a new adaptive threshold function was proposed, which was adapted based on local and global image statistics. In this method, there is no individual band for each image. Instead, the threshold value is changing for each window. The proposed threshold function is based on the cumulative density function (CDF) of the Gaussian distribution function (**Equation 6**) and leads to a fast and robust weighted neighborhood feature extraction algorithm:

$$f_i = \frac{1}{2}\left[1 + \text{erf}\left(\frac{\sigma_i}{\sigma|\mu - \mu_i|}\right)\right] \qquad , i = 1, \dots, N \tag{6}$$

Where 'N' is the number of window blocks, 'μ' and 'σ' are global mean and standard deviation respectively, and 'μ_i' and 'σ_i' are the mean and standard deviation of an image block, and erf (x) is the error function defined in **Equation 7**:

$$\text{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}\, dt \tag{7}$$

Finally, the binary image is obtained using **Equation 8**.

$$I_{binary} = \begin{cases} 1 & if \quad w_{ci} < f_i \times I_{avg} \\ 0 & if \quad w_{ci} \geq f_i \times I_{avg} \end{cases} \tag{8}$$

The method was named the Adaptive-Threshold Weighted Neighborhood Pixel method (AT-WNP). The threshold function is based on noise behavior. The threshold value is proportionate to the image noise density value, so when the noise density increases, the value of the threshold increases. This adaptive threshold made the feature selection work more robust. So far, the binary image results contain some false positive occurrences that consist of non-crack pixels that are wrongly identified as crack pixels (**Figure 7 (C)**). The reason for these false positives is the presence of external objects on the pavement surface. These irregular shapes need to be filtered out, but first, a closing operation is used to connect the small fragmented cracks around the main crack line. A closing operation consists of a dilation followed by an erosion, using the same structural element for both operations. The structural element for this closing operation was selected as a circle with a radius of 5 pixels (**Figure 7 (D)** and **(E)**). The characteristic feature of the false positive noise is their irregular shapes. The irregular-shaped noise can be filtered out effectively by measuring the crack regions' properties, such as the area and eccentricity of the connected crack areas. The eccentricity of the ellipse (identified crack regions) is defined as the distance between the ellipse's foci over the length of its major axis. The range for the eccentricity value is between 0 and 1. An ellipse with the eccentricity of 0 is a circle, and an ellipse with an eccentricity of 1 is a line segment. A filtering method is built to remove objects whose eccentricity value is less than 0.8. This value was chosen after trying and testing some extracted regions. Next, a filling operator was used to fill isolated pixels (individual 1s surrounded by 0s). The results of this step are shown in **Figure 7 (F)**. Based on **Figure 7 (F)**, segmented cracks so far are disjointed and need to be connected to build a united crack entity.

Figure 7 (A) shows the original image; **Figure 7 (B)** indicates the corresponding enhanced image; **Figure 7 (C)** shows the preliminary detection results after applying the improved method and obtaining the binary image; **Figure 7 (D)** shows the image after being dilated; **Figure 7 (E)** shows the image after being eroded, and **Figure 7 (F)** shows the image after filtering out the irregular shapes and filling isolated pixels.

Post-processing

For connecting crack fragments, a closing operator was applied to the image. The structural element (SE) size and shape are based on the characteristics of the actual crack features. Due to this metric's importance, two structural elements were built and tested on 50 random samples. By measuring the Intersection over Union (IoU) metric between the ground truth and the segmented image, the best SE was selected. If the SE size is too large, it might merge the noise, and if it is too small, it will fail to connect the actual crack fragments properly. First, assuming that the gaps are typically at the distance of less than 10 or 15 pixels, a line with a SE length of 10 and 15 pixels was selected. The line's orientation was between -90 and 90 degrees, with 35 degrees as the increment step. This value was chosen after trying and testing some extracted regions. Another morphological closing operator was used to fill gaps using a disk-shaped

structural element with a radius of 10 and 15 pixels. **Table 1** shows the IoU results of different parameter selections.

Based on **Table 1**, the line with a distance length of 15 pixels led to slightly better results and, therefore, was selected as the SE for the closing operator. After connecting the fragmented crack pixels, a filtering operation based on region properties was used to remove noise and connect fragmented pixels. The reason for this step is that once the cracks are connected, new non-crack regions might be created. A filtering algorithm is used to filter out the crack regions less than 75 pixels, whose eccentricity value is less than 0.8, and the length is less than 25 mm. These criteria were selected based on the AASHTO Cracking Protocol PP44-01 (*58*), which does not assume areas smaller than 75 mm^2 and shorter than 25 mm as cracks. By doing so, all the isolated pixels and probable noise created by the closing operator were removed. **Figure 7 (G)** shows the final image after applying the mentioned operations.

NUMERICAL RESULTS AND DISCUSSION

In this section, the performance of the proposed method was investigated and compared with Canny (*59*), CrackTree (*60*), CrackIT (*61*), and CrackForest (*38*), which were tested on the CFD dataset. The evaluation of the proposed crack detection system was performed using precision, recall, and F_1 score.

Figure 8 shows the performance of the proposed method on three sample images. It is observed that the proposed method is effective in eliminating the noise and identifying the cracks. The sample images were selected in a way to represent specific types of surface conditions. **Figure 8 (A)** was chosen to represent a sample of PCC surfaces with tining; **Figure 8 (B)** and **(C)** were selected as the representatives of the AC surfaces with different levels of external noise.

Table 2 compares the results of the proposed method with some other methods in the state-of-the-art. It is observed from **Table 2** that the proposed method shows better or almost equal performance in all four metrics. The evaluation of the mentioned methods on the CFD was provided by Shi et al. (*57*).

The average manual segmentation time for building the ground truth image dataset was 115 seconds per image. The average time for processing the 300 images was 3.15 seconds per image using a Xeon (R) CPU E3-1240 v6 @ 3.70 GHz, which is better than the fixed threshold weighted neighborhood pixels segmentation method, which was reported to be 4.67 seconds (*57*).

CONCLUSIONS

This paper proposes a novel image processing algorithm for the automated detection and segmentation of pavement cracks. For the image enhancement stage, the histogram equalization technique and contrast stretching were used to compensate for the luminosity problem during the image capturing process. The new enhanced image was then pre-processed using adaptive median filtering and K-SVD denoising algorithms. A corresponding binary image was then created by applying an improved version of the weighted neighborhood pixel segmentation algorithm based on cumulative density function as an adaptive threshold, and the image was then segmented using the eight-direction pattern. The threshold value is changing based on the noise present in the neighboring regions, and it makes the algorithm have a robust and accurate detection process. Next, through a disk-shaped dilation and erosion operator, some crack fragments were connected. Also, the irregular-shaped regions and isolated pixels were removed through closing and filling operations. Finally, in the post-processing stage, the crack fragments were connected through the linear closing operator, and linear regions shorter than 25 mm were filtered out. The method is extremely cost and time-efficient compared to deep learning-based methods widely used by the state and federal highway agencies. Generally, it is too expensive to use deep learning-based methods on local roads and small projects, where this algorithm could be implemented and save lots of resources. Like most of the comparable methods in the literature, the major drawback of this method is that the results' quality is mostly

dependent on the image quality. This method is not effective when there are severe anomalies and surface irregularities on the pavement surface or when the images are captured under poor lighting conditions and extreme shadow existence. Further research is necessary to enhance the accuracy and robustness of the method in the presence of surface anomalies. Although there is no perfect solution for accurate crack detection, and there is always a trade-off between accuracy and speed, further research could result in higher computational speed and F_1 scores and, consequently, better results.

AUTHOR CONTRIBUTIONS

The authors confirm contribution to the paper as follows: study conception and design: Smadi. O, Safaei. N; data collection: Smadi. O, Safaei. B; analysis and interpretation of results: Safaei. N, Masoud. A, Safaei. B, Smadi. O; draft manuscript preparation: Safaei. N, Masoud. A. All authors reviewed the results and approved the final version of the manuscript.

REFERENCES

1. Safaei, N., O. Smadi, B. Safaei, and A. Masoud. A Novel Adaptive Pixels Segmentation Algorithm for Pavement Crack Detection. 2021. https://doi.org/10.31124/advance.13601339.v1.

2. Safaei, N., and C. Zhou. Gasoline Pricing Policies for Transportation Safety. arXiv:2001.02734 [stat], 2020. https://arxiv.org/abs/2001.02734

3. Mahmud, M. S., N. Gupta, B. Safaei, H. Jashami, T. J. Gates, P. T. Savolainen, and E. Kassens-Noor. Evaluating the Impacts of Speed Limit Increases on Rural Two-Lane Highways Using Quantile Regression. https://engrxiv.org/afxsb/. Accessed Jan. 15, 2021.

4. Safaei, B., N. Safaei, A. Masoud, and S. Seyedekrami. Prioritizing Strategies to Reduce Motorcycle Crashes Using Fuzzy TOPSIS and AHP Methods: A Human Factors Perspective. Advances in Transportation Studies, No. In Press, 2021.

5. Safaei, N., O. Smadi, A. Masoud, and B. Safaei. An Automatic Image Processing Algorithm Based on Crack Pixel Density for Pavement Crack Detection and Classification. International Journal of Pavement Research and Technology, No. In Press, 2021.

6. Han, J., and J. K. Thakur. Sustainable Roadway Construction Using Recycled Aggregates with Geosynthetics. Sustainable Cities and Society, Vol. 14, 2015, pp. 342–350. https://doi.org/10.1016/j.scs.2013.11.011.

7. Park, H. J., and Y. R. Kim. Primary Causes of Cracking of Asphalt Pavement in North Carolina: Field Study. International Journal of Pavement Engineering, Vol. 16, No. 8, 2015, pp. 684–698. https://doi.org/10.1080/10298436.2014.943220.

8. Malakooti, A., W. S. Theh, S. M. S. Sadati, H. Ceylan, S. Kim, M. Mina, K. Cetin, and P. C. Taylor. Design and Full-Scale Implementation of the Largest Operational Electrically Conductive Concrete Heated Pavement System. Construction and Building Materials, Vol. 255, 2020, p. 119229. https://doi.org/10.1016/j.conbuildmat.2020.119229.

9. Hamedi, G. H., A. Sahraei, and M. R. Esmaeeli. Investigate the Effect of Using Polymeric Anti-Stripping Additives on Moisture Damage of Hot Mix Asphalt. European Journal of Environmental and Civil Engineering, Vol. 25, No. 1, 2018, pp. 90–103. https://doi.org/10.1080/19648189.2018.1517697.

10. Torres-Machí, C., A. Chamorro, E. Pellicer, V. Yepes, and C. Videla. Sustainable Pavement Management: Integrating Economic, Technical, and Environmental Aspects in Decision Making. Transportation Research Record, Vol. 2523, No. 1, 2015, pp. 56–63. https://doi.org/10.3141/2523-07.

11. Daghighi, A., A. Nahvi, S. Nazif, and U. Kim. Seeking Substantiality: Evaluation of Public Attitudes toward Resilient Wastewater Reuse Management. Journal of Water Management Modeling, 2020. https://doi.org/10.14796/JWMM.C470.

12. Moghadas Nejad, F., F. Zaremotekhases, H. Zakeri, L.-J. Wang, and J. Monaghan. An Image Processing Approach to Asphalt Concrete Feature Extraction. Journal of Industrial and Intelligent Information, Vol. 3, 2015. https://doi.org/10.12720/jiii.3.1.54-60.

13. Hosseini, S. A., and O. Smadi. How Prediction Accuracy Can Affect the Decision-Making Process in Pavement Management System. https://engrxiv.org/t28ue/. Accessed Dec. 15, 2020.

14. Hosseini, S. A. Data-Driven Framework for Modeling Deterioration of Pavements in the State of Iowa. Graduate Theses and Dissertations, 2020. https://doi.org/10.31274/etd-20200624-235.

15. Sadati, S. M. S., A. Malakooti, K. S. Cetin, H. Ceylan, and S. Kim. Proposed Improvements to the Construction of Electrically Conductive Concrete Pavement System Based on Lessons Learned. 2020, pp. 1049–1056. https://doi.org/10.1061/9780784482889.111.

16. Abukhalil, Y. Cross Asset Resource Allocation Framework for Pavement and Bridges in Iowa. Graduate Theses and Dissertations, 2019.

17. Daghighi, A. Full-Scale Field Implementation of Internally Cured Concrete Pavement Data Analysis for Iowa Pavement Systems. Creative Components, 2020.

18. McGhee, K. H. Automated Pavement Distress Collection Techniques. Transportation Research Board, 2004.

19. Pierce, L. M., G. McGovern, and K. A. Zimmerman. Practical Guide for Quality Management of Pavement Condition Data Collection. Presented at the 92th annual meeting of Transportation Research Board, 2013.

20. Offrell, P., and R. Magnusson. In Situ Photographic Survey of Crack Propagation in Flexible Pavements. International Journal of Pavement Engineering, Vol. 5, No. 2, 2004, pp. 91–102. https://doi.org/10.1080/10298430412331285752.

21. Kargah-Ostadi, N., A. Nazef, J. Daleiden, and Y. Zhou. Evaluation Framework for Automated Pavement Distress Identification and Quantification Applications. Transportation Research Record, Vol. 2639, No. 1, 2017, pp. 46–54. https://doi.org/10.3141/2639-06.

22. Vavrik, W. R., L. D. Evans, J. A. Stefanski, and S. Sargand. PCR Evaluation – Considering Transition from Manual to Semi-Automated Pavement Distress Collection

and Analysis. Presented at the 92th annual meeting of Transportation Research Board, 2013.

23. Nguyen, T. S., M. Avila, and S. Begot. Automatic Detection and Classification of Defect on Road Pavement Using Anisotropy Measure. Presented at the 17th European Signal Processing Conference, 2009.

24. Safaei, N., and O. Smadi. A Tile-Based Image Processing Method Based on Crack Density for Detection and Classification of Pavement Cracks. Presented at the 98th Annual Meeting of Transportation Research Board, 2019. https://trid.trb.org/view/1595322

25. Safaei, N. Pixel and Region-Based Image Processing Algorithms for Detection and Classification of Pavement Cracks. Graduate Theses and Dissertations, 2019. 17555. https://lib.dr.iastate.edu/etd/17555

26. Karballaeezadeh, N., F. Zaremotekhases, S. Shamshirband, A. Mosavi, N. Nabipour, P. Csiba, and A. R. Várkonyi-Kóczy. Intelligent Road Inspection with Advanced Machine Learning; Hybrid Prediction Models for Smart Mobility and Transportation Maintenance Systems. Energies, Vol. 13, No. 7, 2020, p. 1718. https://doi.org/10.3390/en13071718.

27. Chambon, S., and J.-M. Moliard. Automatic Road Pavement Assessment with Image Processing: Review and Comparison. International Journal of Geophysics. Volume 2011, e989354. https://www.hindawi.com/journals/ijge/2011/989354/.

28. Ayaho, M., K. Masa-Aki, and B. Eugen. Automatic Crack Recognition System for Concrete Structures Using Image Processing Approach. Asian Journal of Information Technology, 2007.

29. Wang, S., and W. Tang. Pavement Crack Segmentation Algorithm Based on Local Optimal Threshold of Cracks Density Distribution. Berlin, Heidelberg, 2012.

30. Tanaka, N. A Crack Detection Mcthod in Road Surfacc Images Using Morphology. Machine Vision and Applications, 1998, p. 4.

31. He, Y., and H. Qiu. A Method of Cracks Image Segmentation Based on the Means of Multiple Thresholds. Journal of Communication and Computer, 2012.

32. Kass, M., A. Witkin, and D. Terzopoulos. Snakes: Active Contour Models. International journal of computer vision, 1988.

33. Kaul, V., A. Yezzi, and Y. Tsai. Detecting Curves with Unknown Endpoints and Arbitrary Topology Using Minimal Paths. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 34, No. 10, 2012, pp. 1952–1965. https://doi.org/10.1109/TPAMI.2011.267.

34. Amhaz, R., S. Chambon, J. Idier, and V. Baltazart. A New Minimal Path Selection Algorithm for Automatic Crack Detection on Pavement Images. Presented at the 2014 IEEE International Conference on Image Processing (ICIP), 2014.

35. Amhaz, R., S. Chambon, J. Idier, and V. Baltazart. Automatic Crack Detection on Two-Dimensional Pavement Images: An Algorithm Based on Minimal Path Selection. IEEE Transactions on Intelligent Transportation Systems, Vol. 17, No. 10, 2016, pp. 2718–2729. https://doi.org/10.1109/TITS.2015.2477675.

36. Delagnes, P., and D. Barba. A Markov Random Field for Rectilinear Structure Extraction in Pavement Distress Image Analysis. In Proceedings., International Conference on Image Processing, No. 1, 1995, pp. 446–449 vol.1.

37. Svenson, K., S. McRobbie, and M. Alam. Detecting Road Pavement Deterioration with Finite Mixture Models. International Journal of Pavement Engineering, Vol. 20, No. 4, 2019, pp. 458–465. https://doi.org/10.1080/10298436.2017.1309193.

38. Shi, Y., 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, 2016, pp. 3434–3445. https://doi.org/10.1109/TITS.2016.2552248.

39. Hosseini, S. A., A. Alhasan, and O. Smadi. Use of Deep Learning to Study Modeling Deterioration of Pavements a Case Study in Iowa. Infrastructures, Vol. 5, No. 11, 2020, p. 95. https://doi.org/10.3390/infrastructures5110095.

40. Ghosh, R., and O. Smadi. Automated Detection and Classification of Pavement Distresses Using 3D Pavement Surface Images and Deep Learning. Presented at the 100th Annual Meeting of Transportation Research Board, 2021.

41. Cheng, H. D., and M. Miyojim. Novel System for Automatic Pavement Distress Detection. Journal of Computing in Civil Engineering, Vol. 12, No. 3, 1998, pp. 145–152. https://doi.org/10.1061/(ASCE)0887-3801(1998)12:3(145).

42. Liu, Z., S. A. Suandi, T. Ohashi, and T. Ejima. Tunnel Crack Detection and Classification System Based on Image Processing. In Machine Vision Applications in Industrial Inspection X, No. 4664, 2002, pp. 145–152.

43. Abdel-Qader, I., S. Pashaie-Rad, O. Abudayyeh, and S. Yehia. PCA-Based Algorithm for Unsupervised Bridge Crack Detection. Advances in Engineering Software, Vol. 37, No. 12, 2006, pp. 771–778. https://doi.org/10.1016/j.advengsoft.2006.06.002.

44. Li, Z., X. Yang, J. Song, K. Liu, Z. Wang, and W. Wu. Improving Resolution of 3D Surface With Convolutional Neural Networks. Sustainable Cities and Society, Vol. 42, 2018, pp. 127–138. https://doi.org/10.1016/j.scs.2018.06.028.

45. Li, B., K. C. P. Wang, A. Zhang, E. Yang, and G. Wang. Automatic Classification of Pavement Crack Using Deep Convolutional Neural Network. International Journal of Pavement Engineering, Vol. 21, No. 4, 2020, pp. 457–463. https://doi.org/10.1080/10298436.2018.1485917.

46. Tong, Z., J. Gao, Z. Han, and Z. Wang. Recognition of Asphalt Pavement Crack Length Using Deep Convolutional Neural Networks. Road Materials and Pavement Design, Vol. 19, No. 6, 2018, pp. 1334–1349. https://doi.org/10.1080/14680629.2017.1308265.

47. Badrinarayanan, V., A. Kendall, and R. Cipolla. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, No. 12, 2017, pp. 2481–2495. https://doi.org/10.1109/TPAMI.2016.2644615.

48. Gopalakrishnan, K., S. K. Khaitan, A. Choudhary, and A. Agrawal. Deep Convolutional Neural Networks with Transfer Learning for Computer Vision-Based Data-Driven Pavement Distress Detection. Construction and Building Materials, Vol. 157, 2017, pp. 322–330. https://doi.org/10.1016/j.conbuildmat.2017.09.110.

49. Wang, K. C. P., A. Zhang, J. Q. Li, Y. Fei, C. Chen, and B. Li. Deep Learning for Asphalt Pavement Cracking Recognition Using Convolutional Neural Network. 2017, pp. 166–177. https://doi.org/10.1061/9780784480922.015.

50. Wu, L., S. Mokhtari, A. Nazef, B. Nam, and H.-B. Yun. Improvement of Crack-Detection Accuracy Using a Novel Crack Defragmentation Technique in Image-Based Road Assessment. Journal of Computing in Civil Engineering, Vol. 30, No. 1, 2016, p. 04014118. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000451.

51. Cui, L., Z. Qi, Z. Chen, F. Meng, and Y. Shi. Pavement Distress Detection Using Random Decision Forests. Cham, 2015.

52. Lei, P. Adaptive Median Filtering. Publication 140.429. 2004.

53. Verma, K., B. Kumar Singh, and A. S. Thokec. An Enhancement in Adaptive Median Filter for Edge Preservation. Procedia Computer Science, Vol. 48, 2015, pp. 29–36. https://doi.org/10.1016/j.procs.2015.04.106.

54. Aharon, M., M. Elad, and A. Bruckstein. K-SVD: An Algorithm for Designing Overcomplete Dictionaries for Sparse Representation. IEEE Transactions on Signal Processing, Vol. 54, No. 11, 2006, pp. 4311–4322. https://doi.org/10.1109/TSP.2006.881199.

55. Lebrun, M., and A. Leclaire. An Implementation and Detailed Analysis of the K-SVD Image Denoising Algorithm. Image Processing On Line, Vol. 2, 2012, pp. 96–133. https://doi.org/10.5201/ipol.2012.llm-ksvd.

56. Kumar, S. Neighborhood Pixels Weights-A New Feature Extractor. International Journal of Computer Theory and Engineering, 2009, pp. 69–77. https://doi.org/10.7763/IJCTE.2010.V2.119.

57. Sun, L., M. Kamaliardakani, and Y. Zhang. Weighted Neighborhood Pixels Segmentation Method for Automated Detection of Cracks on Pavement Surface Images. Journal of Computing in Civil Engineering, Vol. 30, No. 2, 2016, p. 04015021. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000488.

58. AASHTO. Standard Practice for Quantifying Cracks in Asphalt Pavement Surface. 2005.

59. Canny, J. A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-8, No. 6, 1986, pp. 679–698. https://doi.org/10.1109/TPAMI.1986.4767851.

60. Zou, Q., Y. Cao, Q. Li, Q. Mao, and S. Wang. CrackTree: Automatic Crack Detection from Pavement Images. Pattern Recognition Letters, Vol. 33, No. 3, 2012, pp. 227–238. https://doi.org/10.1016/j.patrec.2011.11.004.

61. Oliveira, H., and P. L. Correia. CrackIT — An Image Processing Toolbox for Crack Detection and Characterization. Presented at the 2014 IEEE International Conference on Image Processing (ICIP), 2014.

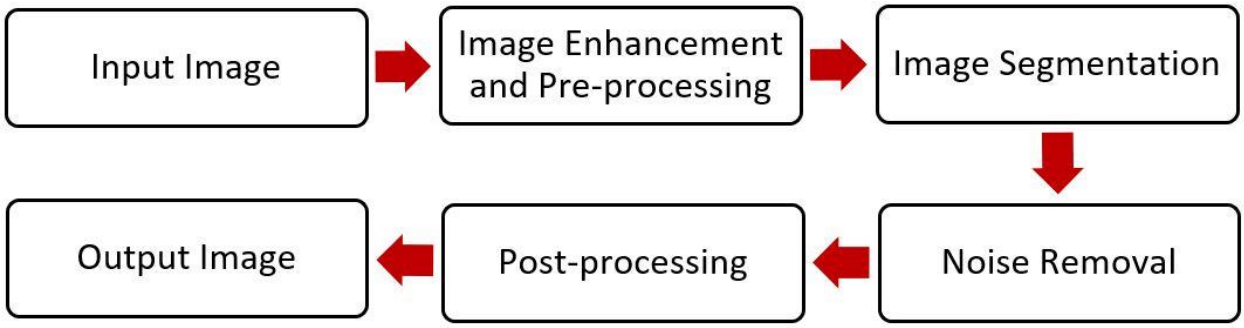

Figure 1 Proposed method procedure

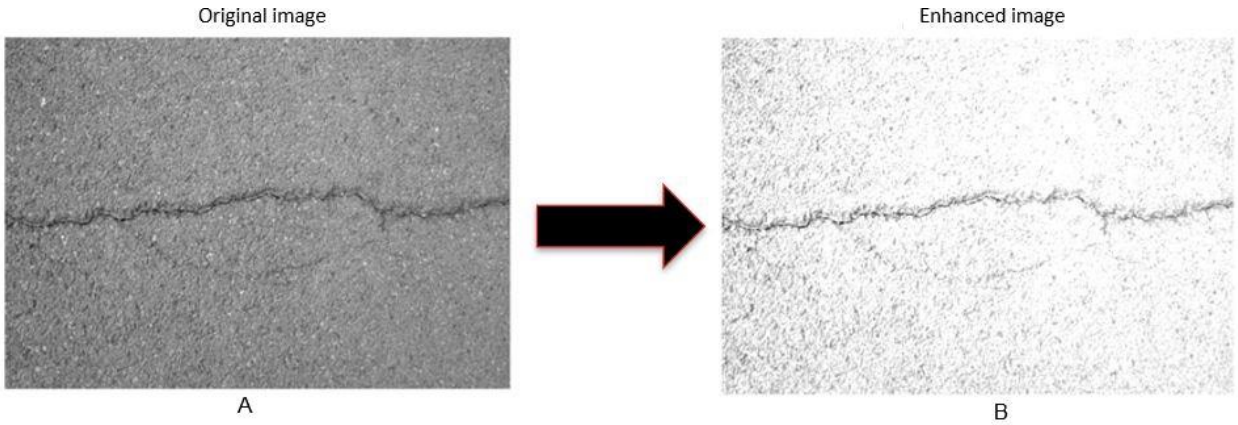

Figure 2 Image enhancement (A) original image (B) enhanced image after using histogram equalization technique

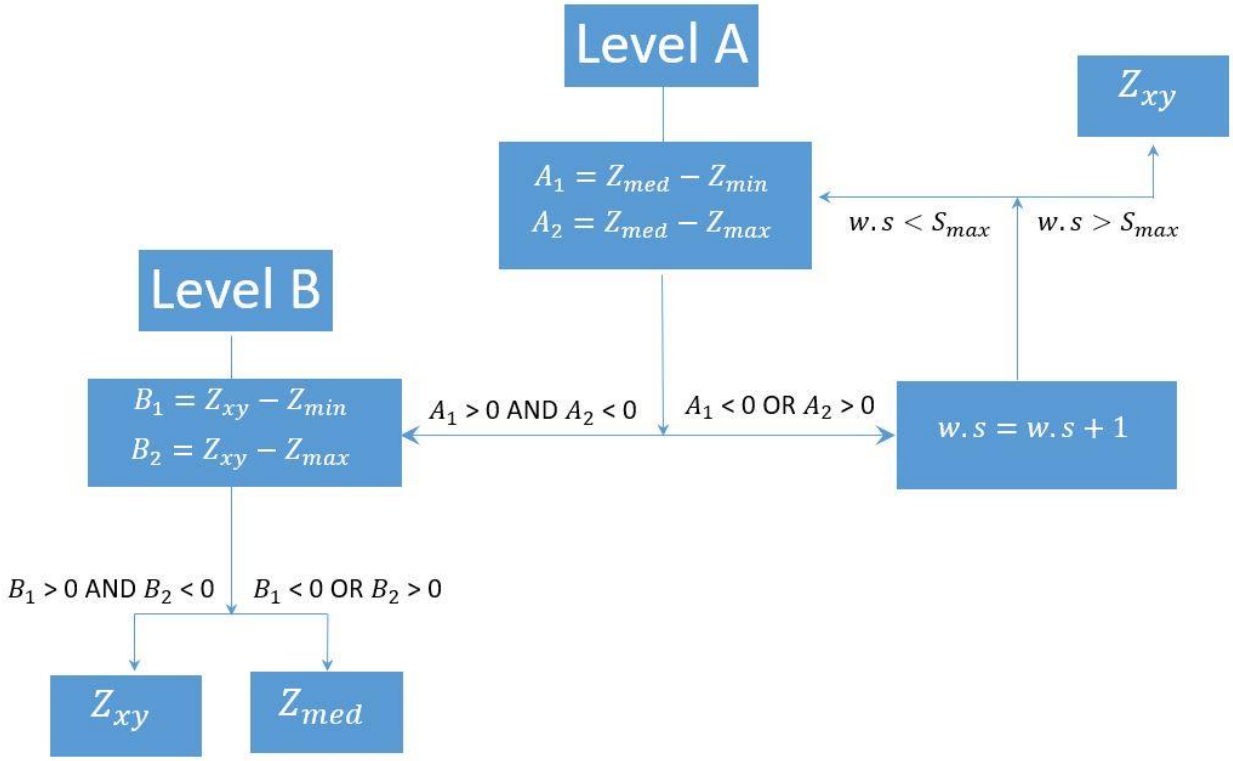

Figure 3 Algorithm of adaptive median filtering

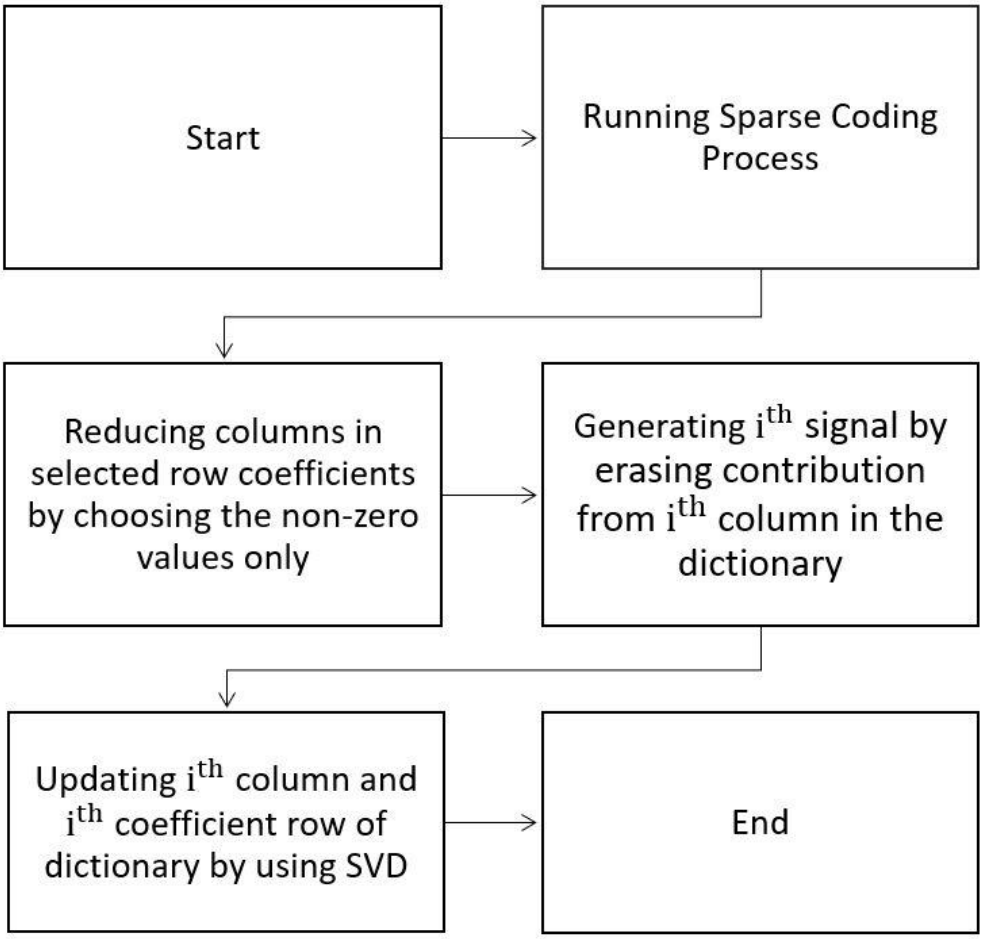

Figure 4 K-SVD approximate algorithm flowchart

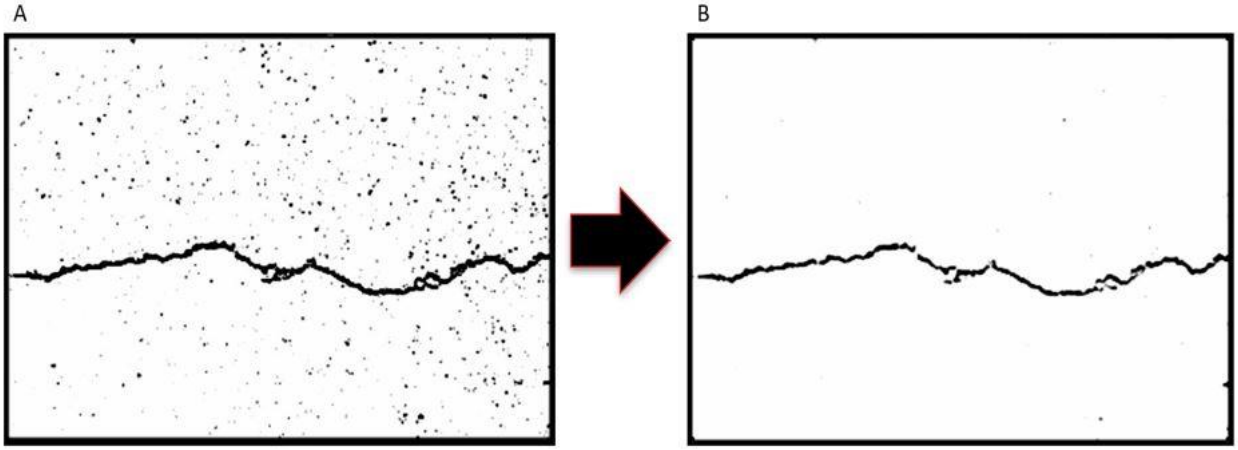

Figure 5 (A) enhanced image (B) enhanced image after applying adaptive median filtering and K-SVD denoising algorithm

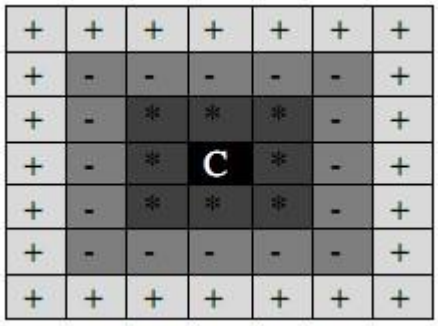

Figure 6 Neighborhood pixels of a three-loop window (reprinted from (*56*))

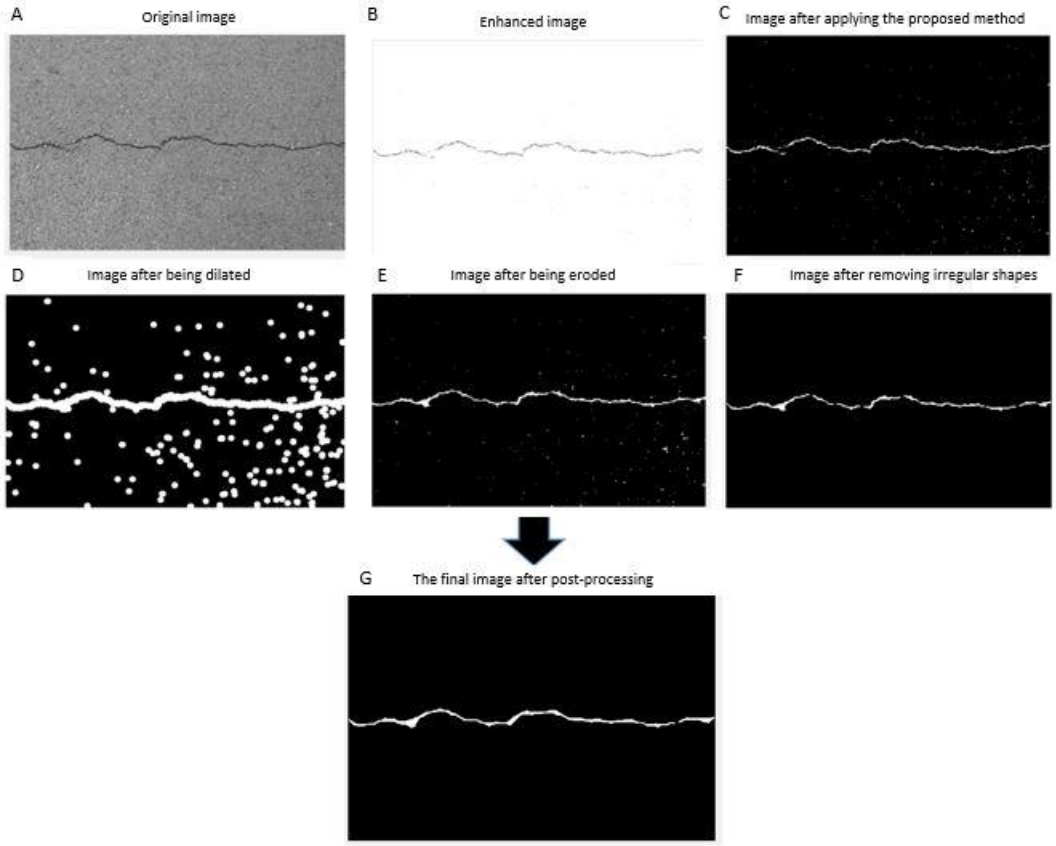

Figure 7 (A) original image (B) enhanced image (C) segmented cracks after (AT-WNP) being applied (D) image after dilation of crack regions (E) image after the erosion of crack regions (F) image after removing irregular shapes (G) final image after post-processing

TABLE 1 IoU results based on the SE and distance length selection

SE\distance length	10 pixels	15 pixels
Disk	94.2	93.1
Line	94.3	95.2

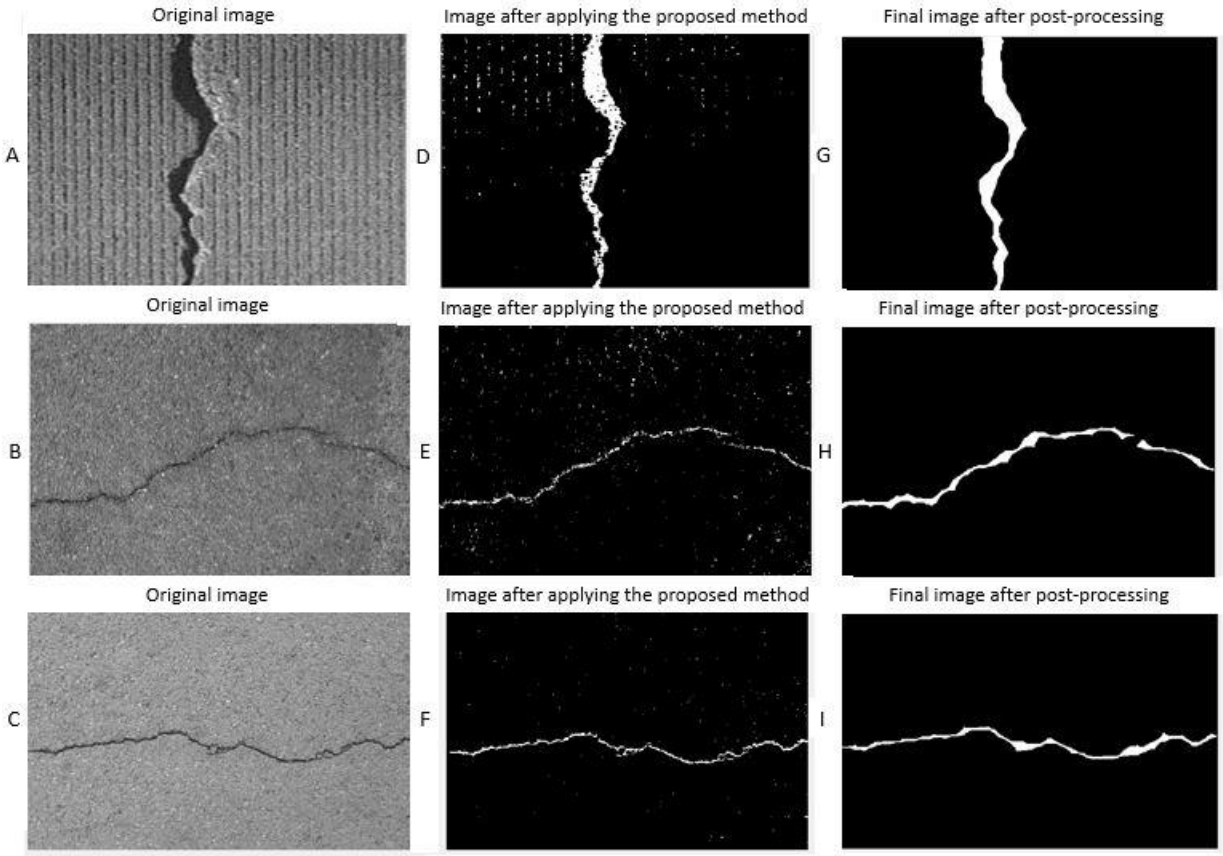

Figure 8 (A) original image of a PCC surface with longitudinal tining (B) the original image of an AC surface with severe noise (C) the original image of an AC surface with moderate noise (D), (E) and (F) segmented cracks after (AT-WNP) was applied on (A), (B) and (C) respectively (G), (H) and (I) final image results after post-processing

TABLE 2 Crack detection evaluation

Method	Precision	Recall	F_1 Score
Canny	12.23%	22.15%	15.76%
CrackIT	67.23%	76.69%	71.64%
CrackTree	73.22%	76.45%	70.80%
CrackForest	80.77%	78.15%	79.44%
Proposed Method (AT-WNP)	79.21%	89.18%	83.90%