

## ROAD CRACK DETECTION USING GAUSSIAN/PREWITT FILTER

Seyed Arvin Fakhri <sup>1\*</sup>, Seyed Arya Fakhri <sup>2</sup>, Mohammad Saadatesresht <sup>2</sup>

<sup>1</sup> Department of Photogrammetry and Remote Sensing, Faculty of Geodesy and Geomatics Engineering, K. N. Toosi University of Technology, Tehran, Iran – arvin.fakhri@email.kntu.ac.ir

<sup>2</sup> School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran

### Commission VI, WG VI/4

**KEY WORDS:** Road Crack, road quality, random forest, close range, image processing, Noise reduction

### ABSTRACT:

Crack is one of the main problems of roads that can reduce the quality of the road or put it in danger in terms of safety. These cracks are needed to be identified first in order to be investigated and followed up. So detecting cracks is one of the most important parts of the road maintenance procedures, which has been considered in recent years. On the other hand, traditional manual methods of crack detection are very time-consuming and dangerous. It is timely because human resources must have thorough and accurate road visits and examine their qualitative status. For this reason, the automatic methods are utilized instead of these methods to increase the speed and reliability of the crack analysis in intelligent transport systems. A simple method is used in this research to detect the crack. Based on the proposed method, a softening filter is applied first on the image to reduce the noise, and then an edge detection filter is applied to the image. Generally, the noise still exists in the image after applying these filters. A window is used here that scans all the image and calculates the average standard deviation for all the pixels in each window, and removes the noise by considering a range. Then the process of removing noise is done with more stringency by reducing the search window in each iteration. Finally, this method was compared with one of the most prominent and modern methods of detecting cracks using a random forest method and the results indicated that despite the simplicity and the speed of the existing method in this study, it has an acceptable performance compared to the manual and random forest methods.

### 1. INTRODUCTION

Pavement assessment plays a vital role in the pavement management system. One of the key elements of pavement assessment is the investigation and scrutiny of road surface damage. The simplest way to evaluate pavement failures is to get road experts a qualitative inspection of the road surface (Zakeri, Nejad, & Fahimifar, 2017). Initially, road engineers used ophthalmic methods to record road crashes to capture the information needed to estimate maintenance and repair costs. Regulations have also been published to help experts identify the type of breakdown and its severity (Miller & Bellinger, 2014). One of the major breakdowns in asphalt is cracks. It should be noted that cracks only represent one important aspect of pavement failures. And other breakdowns, such as pits, patches, bitumen, and stripping, such as cracking, play an important role in lowering the quality index of the pavement. Given these limitations, it is imperative to employ a precision and performance-based approach to automatically identify and process all types of pavement surface failures (Oliveira & Correia, 2012; Zhang, Zhang, Qi, & Liu, 2014). crack is one of the problems on the road that may reduce the quality of a road or compromise its safety. There have been many efforts to build a high-quality road network. Therefore, first it is necessary to identify the cracks. So, identifying Cracks is one of the most important parts of road maintenance that has received much attention in recent years. On the other hand, traditional manual methods of detection of cracks are very time consuming and dangerous. It is time-consuming because the staff needs to have a thorough and thorough inspection of the roads and assess their quality. And this factor may also carry risks. For this reason, automated methods have been replaced to increase the speed

and reliability of crack analysis in intelligent transportation systems. Automatic track detection systems can measure road surface quality and help to prioritize the maintenance of the road network, which increases the lifespan of the roads. With the development of image processing techniques, road cracks detection has been widely discussed in recent decades (Jahangiri, Rakha, & Dingus, 2015; Oliveira & Correia, 2014; Oliveira & Correia, 2012; Salman, Mathavan, Kamal, & Rahman, 2013; Zou, Cao, Li, Mao, & Wang, 2012). In the early methods, the researchers used threshold-based methods to find the cracking area on the basis that the cracking pixels were darker than around (Nguyen, Avila, & Begot, 2009). One of the reasons these methods are problematic is that relying on the feature only provides gray-level information. However, cracks include some other unique features. Most patches have known patterns Therefore, structure-based learning approaches have been introduced to solve such problems. Therefore, in order to counter the above-mentioned weaknesses, Shi et al in their article, have investigated a new method called (CrackForest) based on random forest algorithm (Shi, Cui, Qi, Meng, & Chen, 2016).

#### 1.1 Random forest

Humans use the decision tree every day to make their own choices. Even if they do not know what they are using, it is a kind of machine learning algorithm. Trees are widely used in computer science and software engineering. These trees also have a special place in data mining and classification and many classification algorithms are built on these trees. They are called decision trees because they can make a particular decision based

\* Corresponding author

on past information. A decision tree typically starts with an initial node. After that, the possible consequences are split into branches. And each of those consequences has led to other nodes, which in turn create branches of other possibilities. This branch structure eventually becomes a tree. There are three different types of nodes in the decision tree:

- 1- Random nodes
- 2- Decision nodes
- 3- End nodes

The random node, represented by a circle, indicates the probability that some particular results will occur. The decision node, represented by a square, represents the decision that can be made. And the end node will represent the final outcome of a decision path.

paper must be compiled in one column for the Title and Abstract and in two columns for all subsequent text. All text should be single-spaced, unless otherwise stated. Left and right justified typing is preferred. Random forest learns as an unsupervised machine learning model (Liaw & Wiener, 2002). The forest built, in fact, is a group of decision trees. The classifier for each input sample is  $h(x, \theta)$ . Where  $x$  is an input sample and  $t$  is the training set for the  $k$ th tree. For each sample  $x$ , each tree provides a prediction for sample  $x$ , and finally a category with the highest number of votes on the  $x$  input is selected. This process is called random forest (Breiman, 2001). In short, how the procedure is as follows (Shi et al., 2016):

- 1- Expanding the crack detection methods by introducing the feature-channel integral algorithm
- 2- Introducing Random Forest Algorithm to Extract Structural Information
- 3- Provide a new descriptor using statistical characters

## 2. RELATED WORKS

Chua and Zhu used momentum stability to extract the features of pavement cracks. In this paper, Euclidean minimization is used to classify images of several different types of cracks. The use of textural features compared to other features such as color and instantaneous consistency has provided more detailed information on patterns in areas of the image (Chua & Xu, 1994). Nalamut and Wang used the correlation function to describe the texture of 4 different types of asphalt cracks. Measuring the alternation of intervals between patterns is the basis of this approach. In this paper, the KNN method was used to separate the cracks in the images. And the classification accuracy was reported to be about 56% on average (Nallamothu & Wang, 1996). Cheng et al. Used 11 texture features derived from the coincidence matrix to analyze texture cracking images of asphalt pavement. The results of classification of these images based on decision tree method resulted in approximately 35% error (Cheng, Chen, Glazier, & Hu, 1999). Zou et al. Used image enhancement techniques such as histogram enhancement and Fourier transform to pre-process images and artificial neural network to classify them and reported about 25% error in image classification (Zou et al., 2012). In 2011, a 3D-capable device was launched by a team of researchers. They had already developed a method that could detect and process cracks up to 1 mm wide. A later version of the ADA device was developed to measure 3D images of the road. For this purpose, they used a laser detector next to a digital camera that captures two-dimensional images of the road (Wang, 2011). In 2012, a study was conducted to identify the damage caused by asphalt pavement repair using a histogram filtering and parity method to increase image resolution. To determine the type of damage,

the area constraints were used along with the gray level difference in the pixels (Radopoulou, Jog, & Brilakis, 2013). In 2014, an article examines existing methods for analyzing and identifying pits. In this paper, the identification methods are divided into three categories. Methods based on vibration. Three-dimensional reconstruction methods. And observation-based methods. In vibrational analysis, accelerometers are used that consider holes as failures based on their driving quality (Kim, Kim, Ryu, & Engineers, 2014).

## 3. MATERIALS AND METHODS

### 3.1 Study Images

In this research, the images used in article (Shi et al., 2016) are used. These images were taken from the cracked asphalt surface and were used for training and testing data in the article (Shi et al., 2016). The images have a resolution of 420 x 380 pixels, which is about 1.5 meters from the asphalt surface.

### 3.2 Crack detection by random forest algorithm

For this purpose, a dataset must first be used to train the random forest algorithm. Then a dataset similar to training data is computed to evaluate the above algorithm to determine its performance.

**3.2.1 Training data preparation:** To do this, first select a few images from the cracked area. Then separate the crack area from the non-crack area in a reliable manner, including the manual method. For this purpose, a binary matrix is required where the pixels with the number 1 belong to the crack and the pixels that the number 0 belongs to do not include the crack. Then, for each pixel, specify a window of size  $k \times k$ . So that the desired pixel is located in the center of that window. Now for this window, the average value of gray is calculated. Then the gray difference of each pixel and the average value is calculated. This will give us a vector for each pixel as shown below.

$K_1$	$K_2$	...	$K_{k*k}$	M	C
-------	-------	-----	-----------	---	---

Table 1. The vector for each pixel

Where  $K_i$  is the second power value of the mean difference matrix of gray pixels  $i$  and  $M$  is the mean gray degree of the desired window. And  $C$  denotes a class of pixels that can have a value of 0 or 1, with a value of 1 indicating that the pixel is in the crack region.

### 3.3 Crack detection by image processing methods

Here, crack detection is performed by common image processing algorithms. Since edge detection methods are generally performed by deriving the image, it causes the noise itself to show its effect. Therefore, the image should first be smoothed with smoothing algorithms and then edge detection. Here the simplest smoothing filter, the Average filter, can be used to smooth the edge.

**3.3.1 Average filter:** In the average filter, a window is inserted, which is usually a window of odd dimensions (He, Sun, & Tang, 2010). Like a  $3 \times 3$  window. That is, in the center, there are eight neighborhoods and we have to navigate this window on all matrices. Each pixel should be centered once in the window and decide on the value of its pixels based on the values of its neighbors. For every  $k \times k$  window we can greatly improve the noise effect by convolving this filter to the image (Lee & processing, 1983).

$$\frac{1}{k \times k} \begin{bmatrix} 1 & \dots & 1 \\ \vdots & \ddots & \vdots \\ 1 & \dots & 1 \end{bmatrix} \quad (1)$$

**3.3.2 Gaussian filter:** The Gaussian filter is obtained by converging between each pixel of the input and the Gaussian filter kernel and summing them all (Deng & Cahill, 1993). Assuming the image is one-dimensional, the pixels in the middle have the highest weight, and as we move from the centre to the margin, the pixel weight decreases. The two-dimensional Gaussian filter shows the following formula:

$$G_0(x, y) = A e^{-\frac{(x-\mu_x)^2}{2\sigma_x^2} - \frac{(y-\mu_y)^2}{2\sigma_y^2}} \quad (2)$$

Where  $\mu$  represents the peak and  $\sigma$  represents the variance. Next, a sharpening filter should be used. Experience has shown that edges are areas of the image that have a sudden change in brightness (Catté, Lions, Morel, & Coll, 1992). One of the most famous types of these filters is the Prewitt filter (Muthukrishnan, Radha, & Technology, 2011).

**3.3.3 Prewitt filter:** This filter was introduced by Prewitt. Prewitt was able to identify the edges of the image with almost high accuracy using two 3-by-3 matrices, which are convolved to pixels of image and known as gradient action. The Prewitt matrices are as follows (Yang, Wu, Zhao, Li, & Zhai, 2011).

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -1 & 0 & +1 \\ -1 & 0 & +1 \end{bmatrix} * I \quad (3)$$

$$G_y = \begin{bmatrix} +1 & +1 & +1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} * I \quad (4)$$

Where  $I$  is the image overlay and  $G$  is the image gradient in two directions. Generally, by applying these algorithms, noise is still observed.

### 3.4 Processing

Here, a  $k$ -window is used to scroll through the entire image and calculate the average and standard deviation of all pixels for each window. Initially for training, this is calculated on the parts that include cracks and the parts that are outside. By getting a threshold for this criterion, one can scroll the window in the image until all the pixels in that image are lower than the threshold value. This feature should then be applied to the image and gradually increased in severity. To do this, we start by eliminating noise by manipulating two values of the window dimensions and the threshold. Since increasing the threshold value increases the window hardening in crack detection, so it controls the rate of crack tightening by considering a parameter reducing the rate of increase in more iterations.

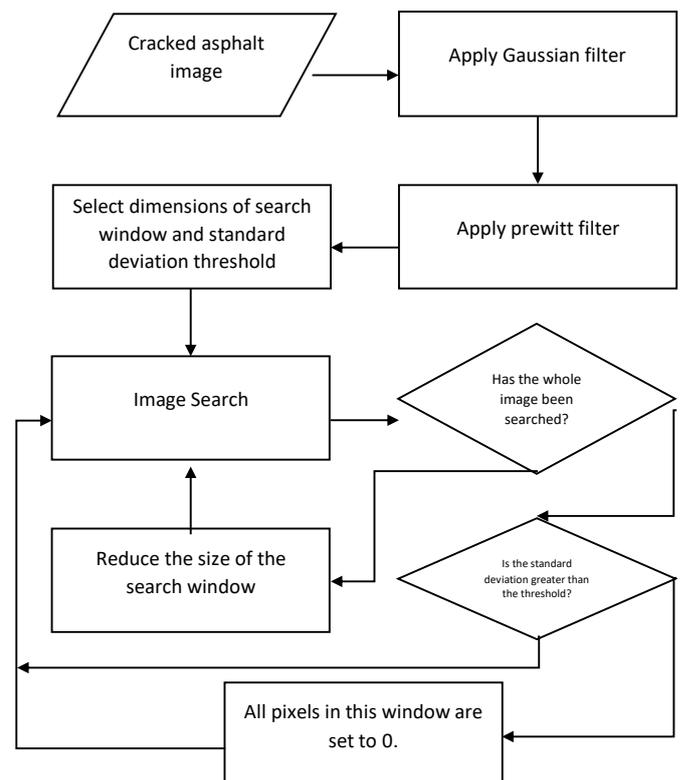


Figure 1. The flowchart of method

### 4. RESULTS

Here, 3 images with different cracks were used and the results of cracking detection at each step were shown.

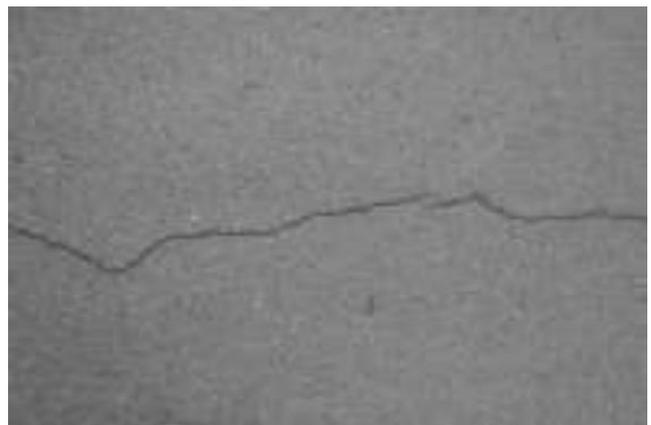


Figure 1. Original image 1

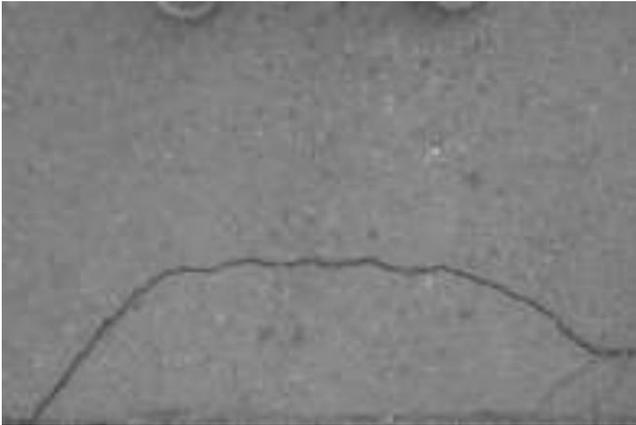


Figure 2. Original image 2



Figure 3. Original image 3

Then, the mean and Gaussian filter were used to smooth the images. Their parameters for the mean filter are the only filter dimensions that were chosen as number 3 and for Gaussian filter  $\mu = 5$  and  $\sigma = 4$  were used.



Figure 4. Smoothed image 1

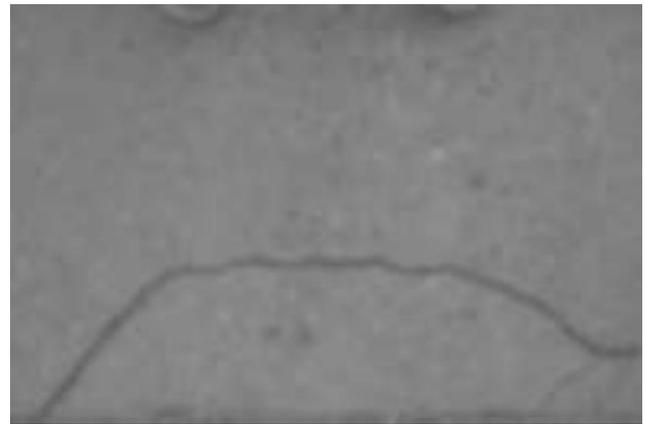


Figure 5. Smoothed image 2

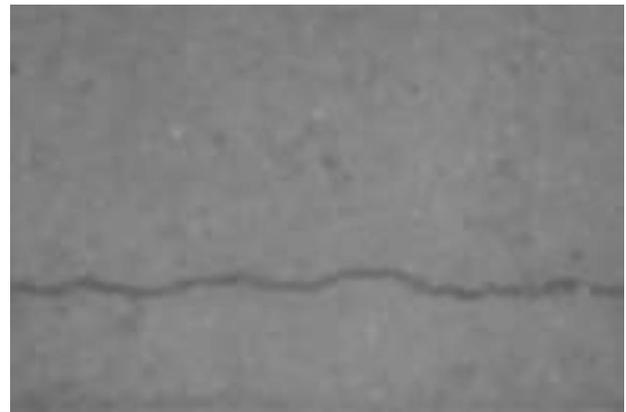


Figure 6. Smoothed image 3

Then, the Prewitt filter is applied to the image. Next, the noise should be removed from the image in a few steps. Adjustment parameters of this step were initially selected for maximum noise search window dimensions of 61 and 35 for the minimum noise window. Twenty repetitions were considered for the search and each window was reduced to 2 units. The target threshold value was initially selected as 1 and the deceleration parameter for increasing the threshold changes was 0.9. At each step, a value of 0.5 is added to the threshold.



Figure 7. Image 1 after applying Prewitt filter

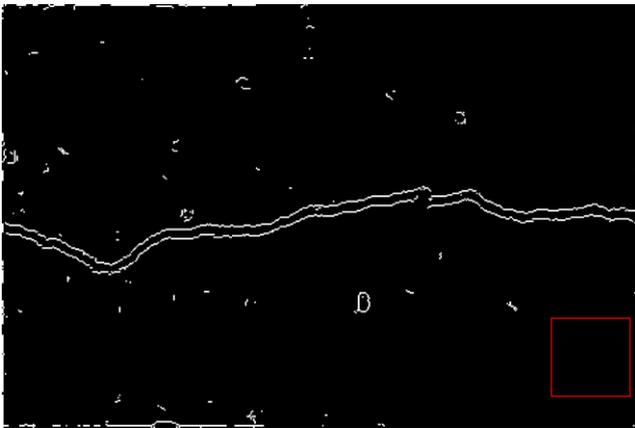


Figure 8. Image 1 after applying noise reduction, step 1

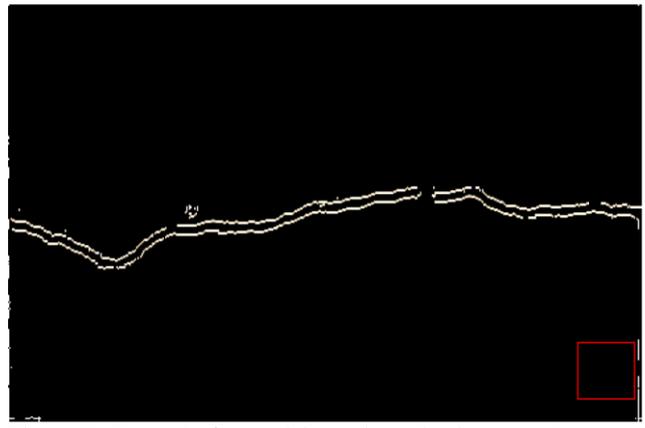


Figure 11. Image 1 after applying noise reduction, step 4

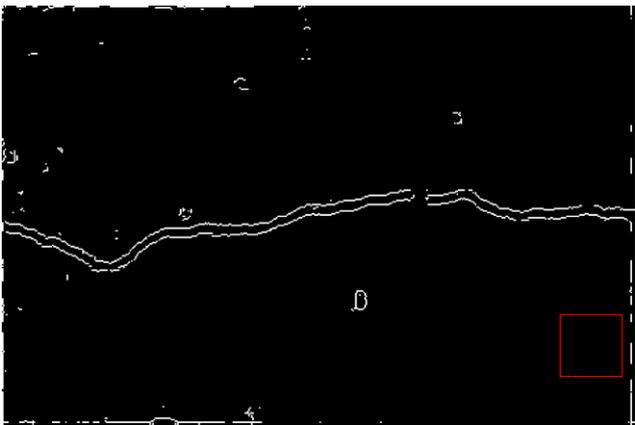


Figure 9. Image 1 after applying noise reduction, step 2

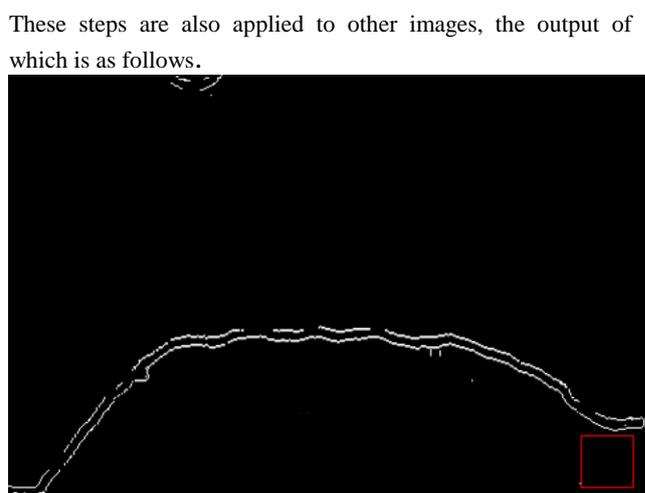


Figure 12. The result of crack detection on image 2

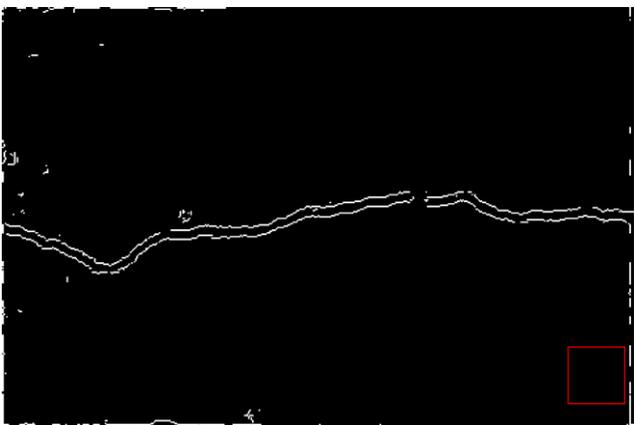


Figure 10. Image 1 after applying noise reduction, step 3

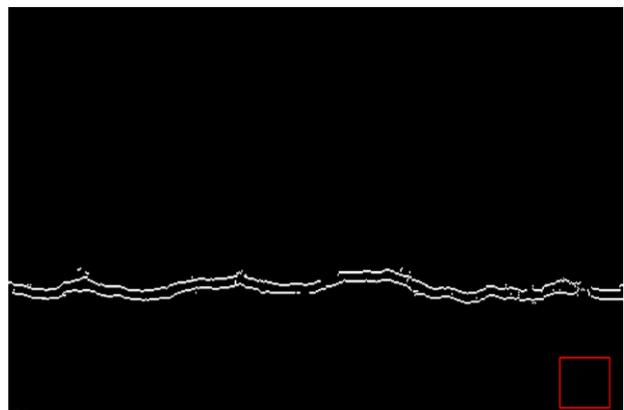


Figure 13. The result of crack detection on image 3

These steps are also applied to other images, the output of which is as follows.

Finally, using the random forest method, the crack detection was calculated for these 3 images as shown in the figure.

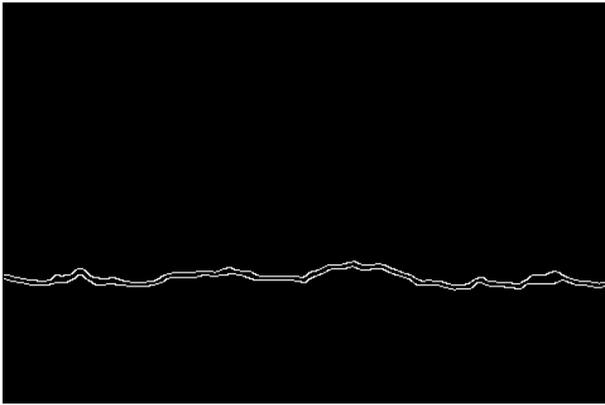


Figure 14. The result of crack detection using RF on image 1

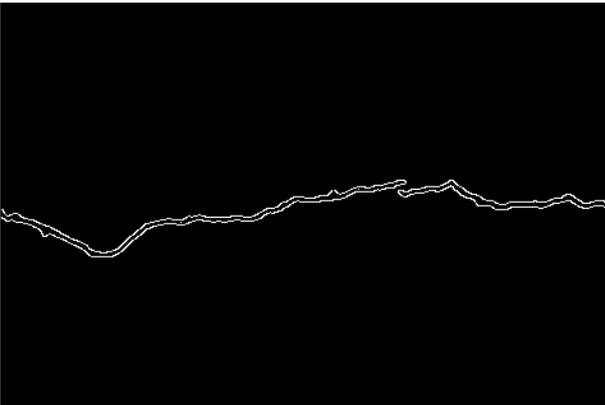


Figure 15. The result of crack detection using RF on image 2

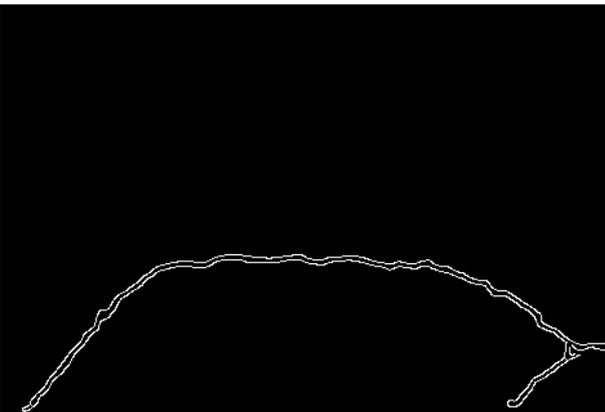


Figure 16. The result of crack detection using RF on image 3

## 5. DISCUSSION

In this paper, a simple image processing method was used to identify the cracks in the road. That was compared with more sophisticated methods of identifying road Cracks including the use of random forest algorithms. The advantage of the approach discussed in this paper is that there is no need for training that make it a simpler problem. High speed in detecting cracks, its simplicity, converting binary image (cracking and non-cracking) are among the advantages of this method. On the other hand, the accuracy of this method is only suitable for some applications, such as the initial identification of road cracks and because the shape and direction of the crack may be affected during noise removal from the image, it is best to develop this

method for specialized investigations on the crack. One of the challenges of this method is to select the correct threshold value for the standard deviation and to select the dimensions of the initial window. Here, the optimal value was chosen considering the image resolution and image dimensions and how many times this method was implemented on similar images. Therefore, this method cannot be used in all types of images.

## REFERENCES

- Breiman, L. J. M. I. (2001). Random forests. 45(1), 5-32.
- Catté, F., Lions, P.-L., Morel, J.-M., & Coll, T. J. S. J. o. N. a. (1992). Image selective smoothing and edge detection by nonlinear diffusion. 29(1), 182-193.
- Cheng, H., Chen, J.-R., Glazier, C., & Hu, Y. J. J. o. C. i. C. E. (1999). Novel approach to pavement cracking detection based on fuzzy set theory. 13(4), 270-280.
- Chua, K. M., & Xu, L. J. J. o. t. e. (1994). Simple procedure for identifying pavement distresses from video images. 120(3), 412-431.
- Deng, G., & Cahill, L. (1993). An adaptive Gaussian filter for noise reduction and edge detection. Paper presented at the 1993 IEEE Conference Record Nuclear Science Symposium and Medical Imaging Conference.
- He, K., Sun, J., & Tang, X. (2010). Guided image filtering. Paper presented at the European conference on computer vision.
- Jahangiri, A., Rakha, H. A., & Dingus, T. A. (2015). Adopting machine learning methods to predict red-light running violations. Paper presented at the 2015 IEEE 18th International Conference on Intelligent Transportation Systems.
- Kim, Y.-R., Kim, T., Ryu, S. J. J. o. T. I. o. E., & Engineers, I. (2014). Pothole Detection Method in Asphalt Pavement. 51(10), 248-255.
- Lee, J.-S. J. C. v., graphics,, & processing, i. (1983). Digital image smoothing and the sigma filter. 24(2), 255-269.
- Liaw, A., & Wiener, M. J. R. n. (2002). Classification and regression by randomForest. 2(3), 18-22.
- Miller, J. S., & Bellinger, W. Y. (2014). Distress identification manual for the long-term pavement performance program. Retrieved from
- Muthukrishnan, R., Radha, M. J. I. J. o. C. S., & Technology, I. (2011). Edge detection techniques for image segmentation. 3(6), 259.
- Nallamothe, S., & Wang, K. C. J. T. R. R. (1996). Experimenting with recognition accelerator for pavement distress identification. 1536(1), 130-135.
- Nguyen, T. S., Avila, M., & Begot, S. (2009). Automatic detection and classification of defect on road pavement using anisotropy measure. Paper presented at the 2009 17th European Signal Processing Conference.

- Oliveira, H., & Correia, P. L. (2014). CrackIT—An image processing toolbox for crack detection and characterization. Paper presented at the 2014 IEEE international conference on image processing (ICIP).
- Oliveira, H., & Correia, P. L. J. I. T. o. I. T. S. (2012). Automatic road crack detection and characterization. 14(1), 155-168.
- Radopoulou, S., Jog, G., & Brilakis, I. (2013). Patch distress detection in asphalt pavement images. Paper presented at the ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction.
- Salman, M., Mathavan, S., Kamal, K., & Rahman, M. (2013). Pavement crack detection using the Gabor filter. Paper presented at the 16th international IEEE conference on intelligent transportation systems (ITSC 2013).
- Shi, Y., Cui, L., Qi, Z., Meng, F., & Chen, Z. J. I. T. o. I. T. S. (2016). Automatic road crack detection using random structured forests. 17(12), 3434-3445.
- Tsai, Y.-C., Jiang, C., & Huang, Y. J. J. o. C. i. C. E. (2012). Multiscale crack fundamental element model for real-world pavement crack classification. 28(4), 04014012.
- Tsai, Y. J., Jiang, C., & Wang, Z. (2014). Implementation of automatic crack evaluation using crack fundamental element. Paper presented at the 2014 IEEE International Conference on Image Processing (ICIP).
- Wang, K. C. (2011). Automated survey of pavement distress based on 2D and 3D laser images. Retrieved from
- Yang, L., Wu, X., Zhao, D., Li, H., & Zhai, J. (2011). An improved Prewitt algorithm for edge detection based on noised image. Paper presented at the 2011 4th International Congress on Image and Signal Processing.
- Zakeri, H., Nejad, F. M., & Fahimifar, A. J. A. o. C. M. i. E. (2017). Image based techniques for crack detection, classification and quantification in asphalt pavement: a review. 24(4), 935-977.
- Zhang, W., Zhang, Z., Qi, D., & Liu, Y. J. S. (2014). Automatic crack detection and classification method for subway tunnel safety monitoring. 14(10), 19307-19328.
- Zou, Q., Cao, Y., Li, Q., Mao, Q., & Wang, S. J. P. R. L. (2012). CrackTree: Automatic crack detection from pavement images. 33(3), 227-238.
- Gago-Silva, A., 2016. GRASS GIS in Grid Environment. doi.org/10.6084/m9.figshare.3188950.
- GRASS Development Team, 2015. Geographic Resources Analysis Support System (GRASS) Software, Version 6.4. Open Source Geospatial Foundation. grass.osgeo.org (1 June 2017).
- GRASS Development Team, 2017. Geographic Resources Analysis Support System (GRASS) Software. Open Source Geospatial Foundation. grass.osgeo.org (20 September 2017).
- Lennert, M. and GRASS Development Team, 2017. Addon i.segment.stats. Geographic Resources Analysis Support System (GRASS) Software, Version 7.2, Open Source Geospatial Foundation. grass.osgeo.org/grass7/manuals/addons/i.segment.stats (1 June 2017).
- Maas, A., Rottensteiner, F., Heipke, C., 2017. Classification under label noise using outdated maps. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-1/W1, 215-222. doi.org/10.5194/isprs-annals-IV-1-W1-215-2017.
- Smith, J., 1987a. Close range photogrammetry for analyzing distressed trees. *Photogrammetria*, 42(1), 47-56.
- Smith, J., 1987b. Economic printing of color orthophotos. Report KRL-01234, Kennedy Research Laboratories, Arlington, VA, USA.
- Smith, J., 1989. *Space Data from Earth Sciences*. Elsevier, Amsterdam, 321-332.
- Smith, J., 2000. Remote sensing to predict volcano outbursts. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.*, XXVII-B1, 456-469.