Journal of the Society of Automotive Engineers Malaysia
Volume 7, Issue 2, pp 91-100, May 2023
e-ISSN 2550-2239

Automated Road Crack Detection: A Comparative Analysis of Edge Detection Techniques for Enhanced Accuracy

F. D. Hamzah[1], **Z. Othman***[2] and S. S. S. Ahmad[2]

[1] Infineon Technologies, Free Trade Zone Batu Berendam Melaka, 75350 Melaka, Malaysia
[2] Universiti Teknikal Malaysia Melaka (UTeM), 76100 Durian Tunggal, Melaka, Malaysia

*Corresponding author: zuraini@utem.edu.my

ORIGINAL ARTICLE *Open Access*

Article History:

Received
11 Nov 2022

Accepted
28 Dec 2022

Available online
1 May 2023

ABSTRACT – *In the modern era, road networks are vital conduits connecting diverse regions, thus necessitating continuous maintenance to ensure efficient transportation. Surface distresses, notably cracks, pose a ubiquitous challenge, affecting road safety and longevity. To address this, automated road crack detection has gained momentum, driven by image processing techniques and deep learning. This study investigates the impact of various image processing methods on road crack detection, aiming to augment conventional human visual inspection. Geometric model-based and deep learning approaches represent two prominent paradigms for crack detection. Geometric model-based techniques employ fundamental image processes, including segmentation and morphological operations, enhancing detection reliability through pre-processing. Thresholding and filtering-based methods, though widely used, tend to neglect critical geometric and photometric characteristics of cracks. However, when combined with geometric models, they yield more accurate results. Image preprocessing, involving grayscale conversion and Canny edge detection, is a critical initial step in the crack detection process. The embedded Gaussian filter mitigates Canny's noise sensitivity. Empirical analysis compared edge detection methods, with Roberts emerging as the most effective. Further improvements were proposed, including morphological operations. This study underscores the significance of image processing techniques and edge detection methods in automating road crack detection. The proposed modification offers a promising avenue for enhanced accuracy in detecting road cracks, contributing to developing efficient road monitoring systems. These advancements are vital for maintaining safer and more durable road networks, with potential applications in transportation infrastructure management.*

KEYWORDS: Road crack detection, image processing, edge detection, infrastructure maintenance, automated monitoring

1. INTRODUCTION

In contemporary times, the intricate network of roads serves as a vital conduit connecting diverse geographical regions, ranging from remote villages to entire nations. The maintenance of these roadways holds paramount importance in ensuring smooth and efficient transportation. However, the constant wear and tear inflicted by heavy vehicular traffic often give rise to surface deterioration referred to as distress. Among these, cracking emerges as the most prevalent issue plaguing main roads, while secondary roads contend with problems such as potholes, patches, and rutting. The responsibility of guaranteeing road safety and longevity falls squarely on the shoulders of transportation authorities. Nevertheless, the manual surveillance of road surface degradation proves to be a labor-intensive endeavor necessitating a high level of expertise. Consequently, there has been a burgeoning interest in harnessing image processing techniques and deep learning algorithms for automated road crack detection. This approach seeks to facilitate efficient road monitoring without the need for constant human intervention. The present study embarks on an exploration of the influence exerted by various image processing techniques on road crack detection. The overarching goal is to advance road quality assessment beyond the conventional method of human visual inspection.

As elucidated in a journal article by (Abdellatif et al., 2020), the realm of crack detection from 2D visual imagery can be broadly categorized into two distinct approaches: geometric model-based and deep learning methodologies. Geometric model-based approaches leverage fundamental image processes, including but not limited to edge detection, segmentation, morphological operations, and texture analysis, to extract pertinent crack features from the images. Several endeavors within the model-based paradigm have aimed at enhancing the reliability of crack detection by incorporating image pre-processing techniques and various forms of spatial smoothing. Such pre-processing measures may entail the application of median filters, as well as opening and closing morphological filters to unite crack segments.

Segmentation, a pivotal component of image processing, entails the assignment of predefined class labels to individual pixels within an image. The outcome of a segmentation algorithm often yields a binary image, wherein pixels bear a value of 0 for one class and 1 for the other. This resultant mask can subsequently be employed for further image analysis or potentially augmenting the input image, contingent upon the intended application, as expounded by (Jenkins et al., 2018).

Among the time-tested and widely embraced crack detection methods are thresholding and filtering-based approaches. Threshold-based methods hinge on histogram analysis, often underpinned by Gaussian assumptions with adaptive or local thresholding. For instance, (Al-Ghaili et al., 2020) employed average pixel intensity values from each row of the input image as threshold values, flagging those with significant deviations as potential crack pixels. However, these methods often fall short in accounting for the geometric and photometric characteristics unique to cracks. Moreover, they assume the separability of pixel intensity distributions between the background pavement and the crack area based on global-level statistics, occasionally resulting in inaccurate measurements.

Tanaka and Uematsu (1998) introduced a crack detection method based on identifying cracks as the progression of saddle points with linear features. The efficacy of this approach heavily relies on parameter selection, limiting its practical applicability. Nonetheless, when employed as an initial stage in conjunction with thresholding methods, it can substantially enhance accuracy. The work from He and Qiu (2012) presented an enhanced segmentation approach, founded on a blend of multidirectional and multi-threshold averaging morphologies, demonstrating marked improvements in the segmentation process, particularly in noisy environments.

In the domain of deep learning approaches, the capacity to learn the crack detection process end-to-end through neural networks, trained with extensive sets of labeled images, has garnered attention. Recent advancements, exemplified by U-net models, have exhibited promise in pixel-wise segmentation (Abdellatif et al., 2020). Architectures such as U-Net aim to address limitations by allowing for effective application to tasks with smaller training datasets. U-Net's unique architecture preserves fine image details, rendering it particularly well-suited for pavement and road crack segmentation, as posited by (Jenkins et al., 2018).

Crucially, before delving into the analytical phase, image processing stands as a pivotal step. A common starting point for many studies involves the conversion of images to grayscale. In grayscale crack surface images, crack pixels typically exhibit deeper gray values compared to non-crack pixels, and the distribution of gray values between cracks and pavement backgrounds remains distinct. Recent research has favored the adoption of the Canny edge detection method, a multi-stage process reliant on the gradient magnitude of a smoothed image to identify edges. This method is preferred due to its sensitivity to noise and its effectiveness in noise reduction, aided by the embedded Gaussian filter within the algorithm itself, as discussed by Öztürk and Akdemir (2015). Joshi and Vyas (2014) assert the superiority of Canny edge detection, especially when employing a Gaussian filter for smoothing, gradient computation, and a double threshold. Their findings demonstrate the superiority of Canny over other edge detection methods, including Sobel and Prewitt, in terms of both accuracy and computational efficiency.

This study endeavors to identify the most suitable method for road crack detection and evaluate the impact of various image processing techniques. The research methodology draws inspiration from prior studies. Consequently, the project will propose a novel framework founded on different edge detection methods and conduct a comprehensive performance analysis.

2. METHODOLOGY

In this section, each element in the methodology will be discussed. These elements are shown in Figure 1. Data collection for this project was obtained from the Crack Forest Dataset (CFD) website which contains 118 images (Shi et al., 2016). 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, an aperture of f/2.4, and an exposure time of 1/134 s. CFD contains many noise features such as shadows, oil spots, and water stains. These datasets also provide the ground truth for every image which can be used for comparison in the next step.

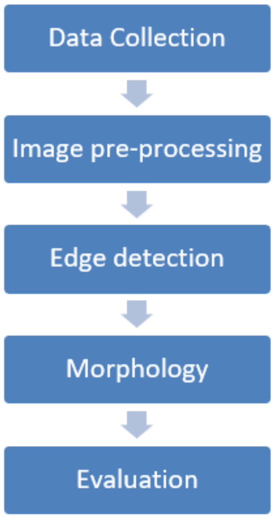

FIGURE 1: Data flow diagram of methodology

Next, the image obtained will go through the manual pre-processing procedure to enhance the image quality for best detection in MATLAB software and utilize the library provided. This project undergoes image pre-processing method which is noise removal, adjusting intensity value, segmented crack, and morphological operation as shown in Figures 2, 3, and 4.

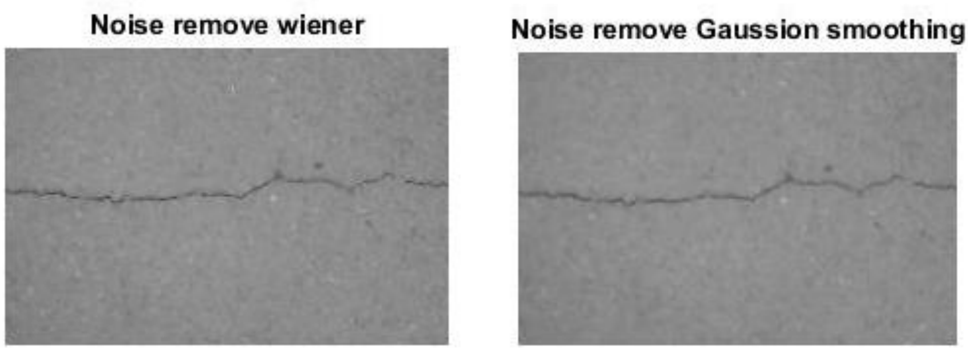

FIGURE 2: Noise removal

Wiener uses a lowpass filter to create an intensity image that has been degraded by constant power additive noise. WIENER2 uses a pixel-wise adaptive Wiener method based on statistics estimated from a local neighborhood of each pixel. It also estimates the additive noise power before filtering. Moreover, the Gaussian smoothing method is used to filter images after the Wiener method by using a 2-D Gaussian smoothing kernel with a standard deviation specified by sigma. Sigma can be a scalar or a 2-element vector with positive values. If sigma is a scalar, a square Gaussian kernel is used.

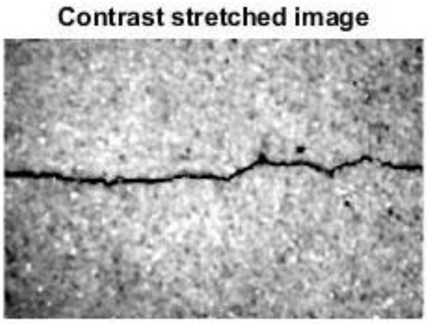

FIGURE 3: Adjusting intensity value

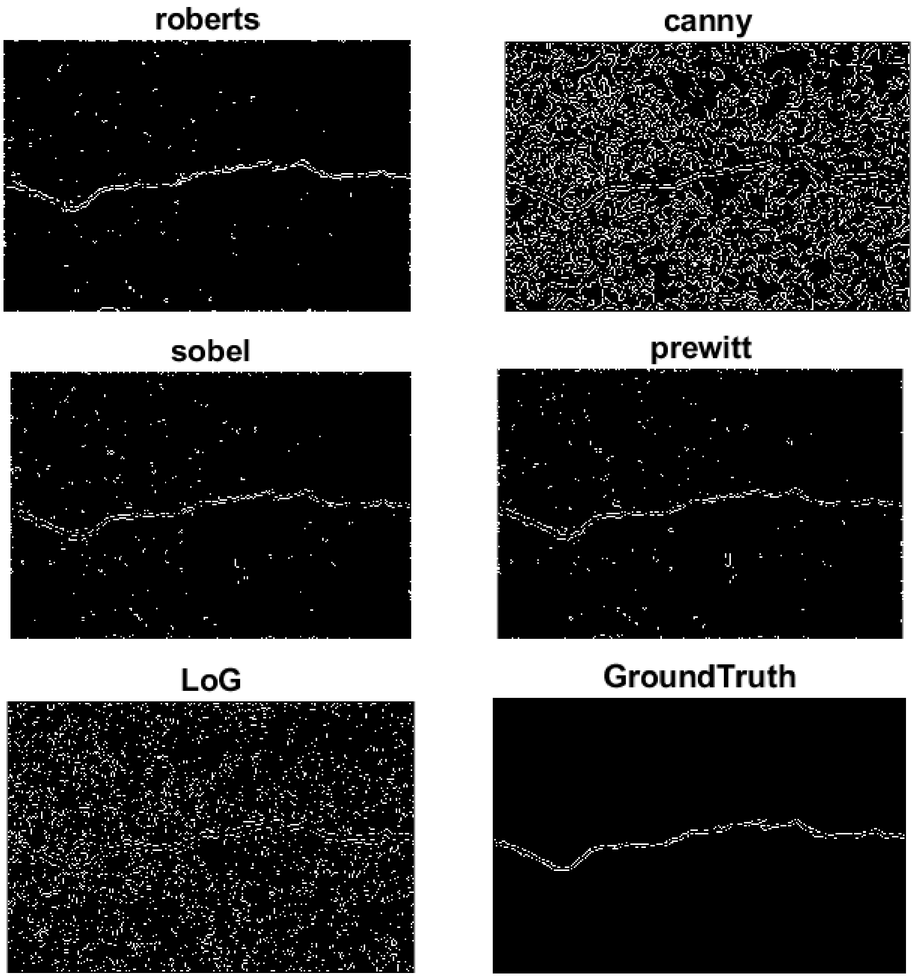

FIGURE 4: Roberts, Canny, Sobel, Prewitt, Log, and GroundTruth methods

Stretchlim is used to find limits to contrast the stretch of an image. It evaluates the lower and upper boundaries that can be applied to grayscale or RGB images from the dataset to increase contrast. The limits are displayed in low high order. The default setting for the limits specifies the bottom for 1% and the top 1% of all pixel values. Then the image undergoes with Imadjust method which is used to map the values in intensity image from the Stretchlim method to a new value such that values between LOW_IN and HIGH_IN map to values between LOW_OUT and HIGH_OUT.

In this step, edge detection is used to detect the crack in the segmented image. The methods used for the analysis are the canny method, Sobel method, Prewitt method, Robert's method, and Log method as shown in Figure 4. Robert, Sobel, and Prewitt's method used the first-order derivative method which is the gradient method. While Laplacian of Gaussian (Log) used second order derivative method and the canny detection method used optimal edge detection. Each of these detections has its own pros and cons which we used to compare and do analysis on which method is better to use.

The gradient approach finds the edges by examining the highest and minimum values in the image's first derivative. It is locating the gradient's magnitude at each pixel in an image and comparing it to a threshold to determine whether it is an edge pixel or not. Some of the operators that used gradient-based are Roberts, Sobel, and Prewitt.

Robert works by providing a simple and quick approximation of the gradient’s magnitude. The operator performs 2-D spatial gradient measurement on an image. It thus highlights regions of high spatial frequency which often correspond to edges. The reason the Roberts operator is so widely used is because of how quickly it operates. In terms of its disadvantages, this operator is highly noise-sensitive and fails to identify edges with subtle intensity jumps.

Next, the Sobel operator functionality is close to the Robert operator. Nevertheless, it uses larger convolution kernels. The Sobel operator is low in terms of calculations since it works by picture convolution with a tiny, separable, integer-valued filter in both the horizontal and vertical directions. However, it creates a somewhat weak gradient approximation, especially for high-frequency variations in the picture.

Finally, the Prewitt operator is likewise identical to the Sobel operator with the exception that the Prewitt does not offer the pixels that are near the centers of the masks for any emphasis whereas the Sobel does.

The zero crossing detector searches for regions of an image’s Laplacian where its value crosses zero points when the Laplacian's sign changes. These points frequently appear near the 'edges' of pictures, which are areas where the intensity of the image varies quickly, but they can also appear elsewhere that is less obvious to correlate with edges. Instead of focusing on the zero-crossing detector as a specific edge detector, it is preferable to see it as a kind of feature detector. Since zero crossings only occur on closed contours, the output of the zero-crossing detector is often a binary picture with lines of a single pixel thickness indicating the locations of the zero-crossing sites. As a result, the alternative to finding the location of an edge is to locate the zeros in the second derivative. This method is known as the Laplacian.

The Log operator, also known as the Laplacian of Gaussian operator, is a unique class of Zero-Crossing operators that performs filtering or smoothing using a Gaussian filter and enhancement using its second derivative. Both operations can be carried out directly by passing the image through a linear filter, which is the Laplacian of the Gaussian filter.

The Canny approach locates edges by scanning for local gradient maxima in the picture. The derivative of a Gaussian filter is used to compute the gradient. The technique employs two thresholds to distinguish between strong and weak edges and only outputs weak edges when they are related to strong edges. So, compared to other methods, this one is more likely to identify genuine weak edges and less likely to be “fooled” by noise.

In addition, the image is then analyzed by using morphological operations on binary images. The operation performs morphological area opening (erosion followed by dilation) and performs morphological closing (dilation followed by erosion). Basically, area opening helps to remove from a binary image all connected components which are objects that have fewer than P pixels, then it produces another binary image. It has been used to clean up flaws in many kinds of shapes, like object boundaries. Erosion and dilation are typically the two operators in this operation. Opening is the initial operation, which helps smooth the contour object, splits thin strips, and removes small protrusions. The second step, known as closing, likewise smoothest contours but, unlike opening, merges fine discontinuities, fills in small holes, and fills contour gaps as shown in Figure 5.

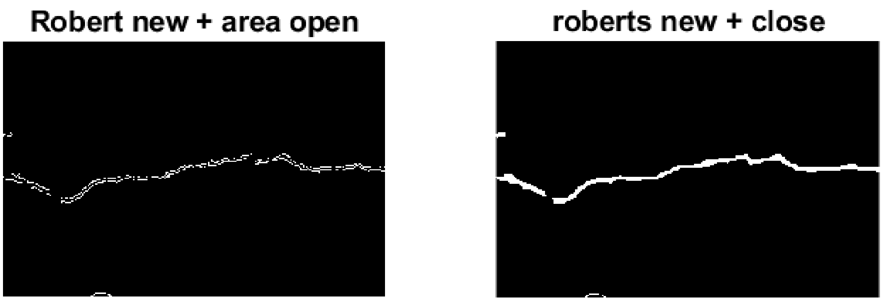

FIGURE 5: Result area open and close

Next step of the methodology, we will evaluate each of the images with the ground truth to compare the precision, recall, and F-measure. The results will be recorded and put on a table to compare their performance in detecting the crack. To determine the best detection, their performance must achieve high-value accuracy of precision, recall, and F-measure. The metric measurements used in this study are recall, precision, and F-Measure. A precision value is the number of documents retrieved with a relevant index and several retrievals. Data Recall equals the total number of files or the total number of relevant documents in a database that are retrieved that are relevant - the total number of relevant documents in the database. The F-measure is a way of combining the precision and recall of the model, and it is defined as the harmonic mean of the model's precision and recall. It is calculated as shown below:

$$\text{precision} = \frac{TP}{(TP + FP)},$$

$$\text{recall} = \frac{TP}{(TP + FN)},$$

$$F - \text{measure} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}.$$

3. RESULTS AND DISCUSSION

Result comparison is a phase where all the averages of evaluation methods will be compared, and the maximum average will be chosen as the best method for edge detection. As the result after comparing all the edge detection methods. We find that the Roberts method has had more impact in detecting the road crack by using the CFD. So, this method will undergo further analysis to propose a new framework for modification of edge detection.

TABLE 1: Result comparison

Method	Precision	Recall	Fmeasure
canny	0.043647458	0.27084661	0.059786441
sobel	0.116728814	0.192966949	0.137982203
prewitt	0.116438983	0.193569492	0.138144915
robert	0.155834746	0.239045763	0.181516949
LOG	0.04175339	0.139370339	0.053923729

The implementation between processing codes and the GUI was created by using MATLAB software. To do this, App Designer is used which enables users to construct professional apps in MATLAB without the need to be a professional software developer. App Designer aids in designing the user interface, i.e., the GUI. The tool is very user-friendly where it can create precise layout, drag, and drop visual elements onto the design canvas, and make use of alignment hints. The object-oriented code that specifies the app's layout and design it automatically generated by the App Designer as shown in Figure 6.

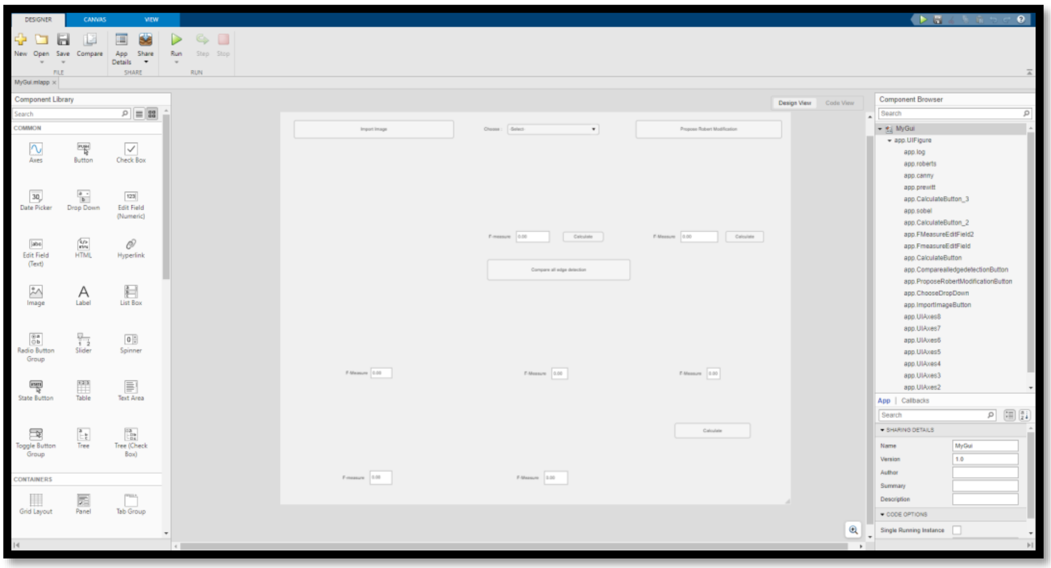

FIGURE 6: App Designer

To implement the processing code and the GUI, the canvas needs to be filled with all the components needed such as a button, edit field, axes, and drop-down button. When the designing part is done, the component needs to define their behavior which will make the system more interactive. Figure 7 shows the code for the callback button that loads the desired image from the file. It also automatically generates the pre-processing image that has the same code as the MATLAB file for further processing. The callback for the dropdown button is created for the user to choose which edge detection type to analyze. This code uses a switch case program to choose the desired edge detection. Furthermore, the code for the callback button and edit fieldwork to generate a measurement of the image. The calculate button will load the ground truth image then it will perform the calculation based on the comparison of edge detection and the ground truth image. The measurement will show in the edit file which only receives numbers. All the buttons used will repeat the same step as the above explanation. Figure 8 shows the full interface of the system after the code is fully finished.

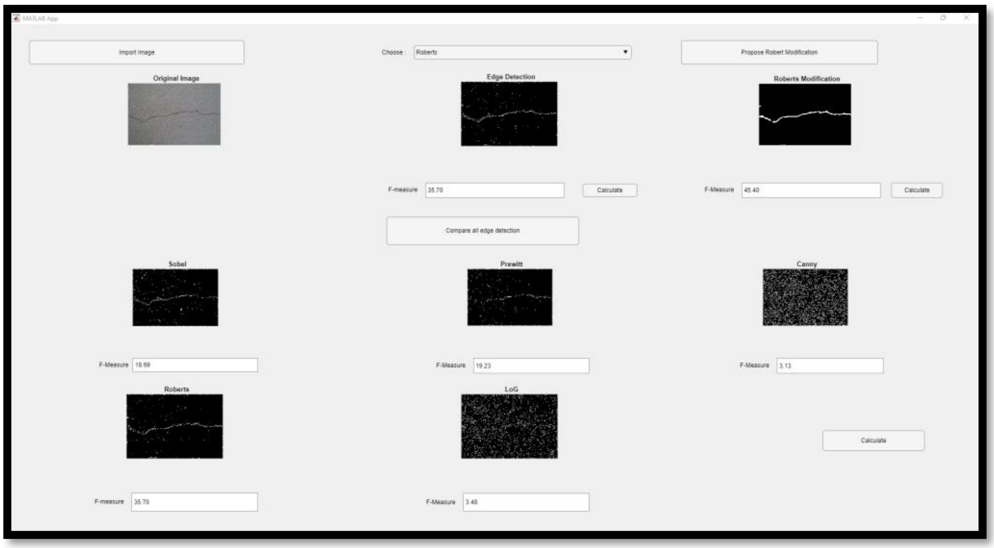

FIGURE 7: Interface of the system

The term "histogram of an image" often refers to a histogram of the pixel intensity values in the context of image processing. A graph called a histogram displays the number of pixels in an image at each possible intensity value that may be found in that picture. The histogram will visually display 256 numbers illustrating the distribution of pixels among those grayscale values as there are 256 distinct potential intensities for an 8-bit grayscale picture. While threshold is to distinguish between the image areas that correspond to the items we are interested in and the image regions that relate to the background in many vision applications. When segmenting a picture based on the varying intensities or colors in the foreground and background, thresholding frequently offers a quick and practical method of doing so. The ability to determine which portions of an image are made up of pixels whose values fall within a given range, or band of intensities, is also frequently helpful (or colors).

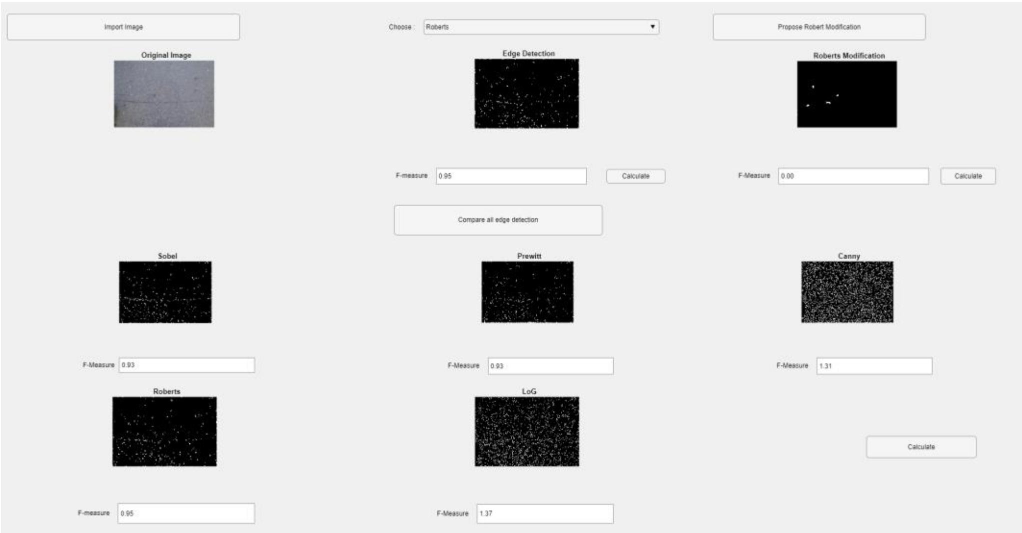

FIGURE 8: Example result for an image (file 039.jpg)

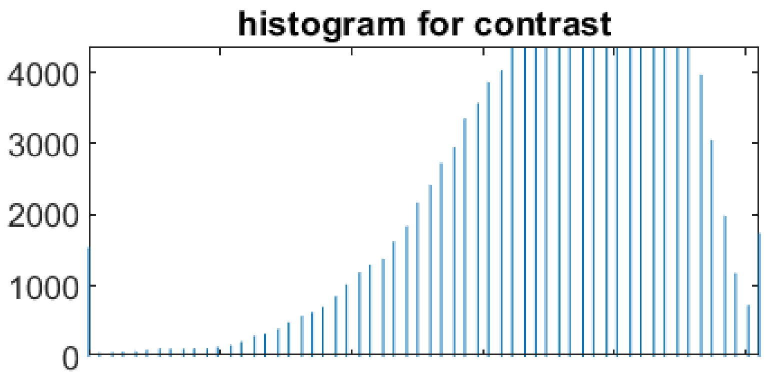

FIGURE 9: Histogram for image of file 039.jpg

Based on Figures 8 and 9, the histogram shows that the image is barely able to segment the foreground and background which affects the edge detection. It also gives a lower F-measure as the image cannot separate the foreground and background image. The histogram shows that most pixels have higher intensity values after the pre-processing.

Therefore, to enhance the image, this project proposes to perform morphology on the selected edge detection which is Roberts. This is because after comparing all the averages of the edge detection method, we find that Roberts gives more impact. Thus, the analysis has been performed for the purpose of edge detection and we need to vary the parameters to get better results.

Based on Figure 10 and Table 2, it shows that Robert proposed 2 give enhancement from the Robert before. The optimum parameter for bwareopen is 25 with strel of 10. In addition, the parameter used in Robert proposed 1 and 3 is represented to analyze whether the parameter is higher or lower than Robert proposed 2. In this case, this proposed method is not suitable for higher or lower than the parameter used in Robert's proposed 2.

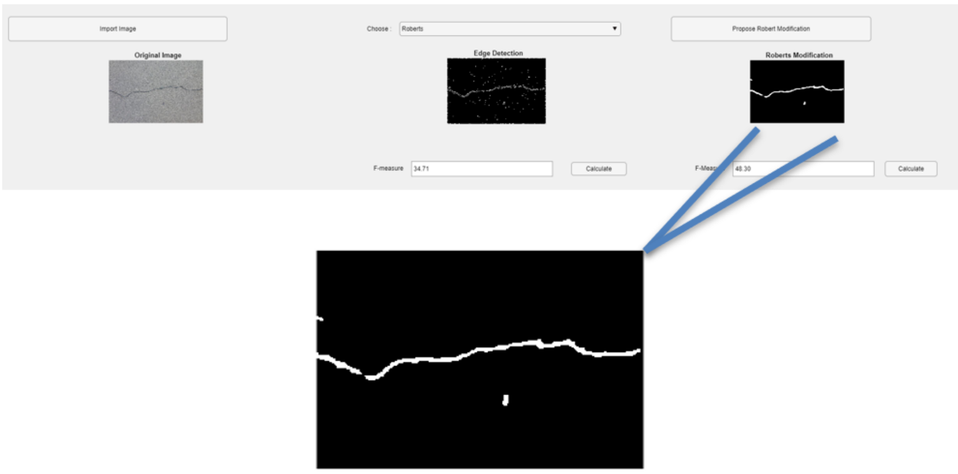

FIGURE 10: Result proposed of Robert's modification

TABLE 2: Analysis of average proposed Robert modification

Method	Precision	Recall	F-Measure
Robert (before)	0.155834746	0.239045763	0.181516949
Robert Proposed 1	0.287836441	0.150965	0.177839
Robert Proposed 2	0.227401695	0.238028	0.200105
Robert Proposed 3	0.207445763	0.260831	0.199129

4. CONCLUSION

In this study, we embarked on an exploration of various image processing techniques and their impact on road crack detection. The objective was to advance road quality assessment beyond traditional human visual inspection by leveraging geometric model-based and deep-learning approaches.

Our examination of geometric model-based techniques revealed the significance of image preprocessing and spatial smoothing in enhancing crack detection. Methods such as median filters, opening, and closing morphological filters played a vital role in reducing image noise and improving detection results. Additionally, segmentation techniques facilitated the assignment of class labels to pixels, enabling further analysis and augmentation of images. Thresholding and filtering-based methods, while widely used, exhibited limitations in considering the geometric and photometric characteristics of cracks, often leading to inaccurate measurements. Nonetheless, they proved valuable when integrated with geometric model-based approaches.

Image preprocessing, including grayscale conversion and edge detection using the Canny method, played a crucial role in preparing images for crack detection. Canny edge detection, known for its noise sensitivity, effectively reduced noise using the embedded Gaussian filter. Furthermore, we conducted an empirical analysis of various edge detection methods, including Roberts, Sobel, Prewitt, Log, and Canny, on the Crack Forest dataset. Through meticulous evaluation, we found that the Roberts method exhibited the most promising results.

To further improve the Roberts method, we proposed modifications involving morphological operations. After extensive parameter tuning and analysis, the proposed modification, labeled Robert Proposed 2, outperformed the original Roberts method in terms of precision, recall, and F-measure. This optimization involved setting the bwareopen parameter to 25 and employing a strel value of 10.

In conclusion, this study highlights the critical role of image processing techniques and edge detection methods in road crack detection. The proposed modification to the Roberts method represents a promising avenue for enhancing crack detection accuracy. As road infrastructure maintenance remains a vital concern, the findings from this study contribute to the ongoing efforts to develop efficient and automated road monitoring systems, ultimately ensuring safer and more durable road networks. Future research may explore additional enhancements and real-world applications of these techniques in the field of transportation infrastructure management.

ACKNOWLEDGEMENT

The authors express their deepest gratitude to ASEAN NCAP Collaborative Holistic Research (ANCHOR) III (No. ANTARABANGSA-ANCHOR/2020/FTMK-CACT/A00032) for supporting this research.

REFERENCES

Abdellatif, M., Peel, H., Cohn, A. G., & Fuentes, R. (2020). Pavement crack detection from hyperspectral images using a novel asphalt crack index. Remote Sensing, 12(18), 1-20.

Al-Ghaili, A. M., Kasim, H., Al-Hada, N. M., Othman, M., Saleh, M. A., & Jihua, W. (2020). Pixel Intensity-Based Contrast Algorithm (PICA) for Image Edges Extraction (IEE). IEEE Access, 8, 119200-119220.

He, Y., & Qiu, H. (2012). A method of cracks image segmentation based on the means of multiple thresholds. Journal of Communication and Computer, 9(10), 1147-1151.

Jenkins, M. D., Carr, T. A., Iglesias, M. I., Buggy, T., & Morison, G. (2018). A deep convolutional neural network for semantic pixel-wise segmentation of road and pavement surface cracks. European Signal Processing Conference, 2120-2124.

Joshi, M., & Vyas, A. (2014). Comparison of Canny edge detector with Sobel and Prewitt edge detector using different image formats. International Journal of Engineering Research & Technology, 1(1), 133-137.

Öztürk, Ş., & Akdemir, B. (2015). Comparison of edge detection algorithms for texture analysis on glass production. Procedia - Social and Behavioral Sciences, 195, 2675-2682.

Shi, Y., Cui, L., Qi, Z., Meng, F., & Chen, Z. (2016). Automatic road crack detection using random structured forests. IEEE Transactions on Intelligent Transportation Systems, 17(12), 3434-3445.

Tanaka, N., & Uematsu, K. (1998). Crack detection method in road surface images using morphology. IAPR Workshop on Machine Vision Applications, 154-157.