CAIT-UTC-009

Multi-resolution Information Mining and a Computer Vision Approach to Pavement Condition Distress Analysis

FINAL REPORT
July 2014

Submitted by:

Yaw Okyere Adu-Gyamfi
Graduate Research Assistant

Nii Attoh-Okine
Professor

Civil & Environmental Engineering Department
University of Delaware
Newark DE 19716

External Project Manager
Dr Alexander Appea,
Virginia DOT

In cooperation with

Rutgers, The State University of New Jersey
And
State of Delaware
Department of Transportation
And
U.S. Department of Transportation
Federal Highway Administration

Disclaimer Statement

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
CAIT-UTC-009		

4. Title and Subtitle	5. Report Date
Multi-resolution Information Mining and a Computer Vision Approach to Pavement Condition Distress Analysis	July 2014
	6. Performing Organization Code CAIT/University of Delaware

7. Author(s)	8. Performing Organization Report No.
Yaw Okyere Adu-Gyamfi N.O. Attoh-Okine	CAIT-UTC-009

9. Performing Organization Name and Address	10. Work Unit No.
Department of Civil & Environmental Engineering University of Delaware Newar006B DE 19716	11. Contract or Grant No. DTRT12-G-UTC16

12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
Center for Advanced Infrastructure and Transportation Rutgers, The State University of New Jersey 100 Brett Road Piscataway, NJ 08854	Final Report 9/1/12-1/31/14
	14. Sponsoring Agency Code

15. Supplementary Notes
U.S. Department of Transportation/Research and Innovative Technology Administration
1200 New Jersey Avenue, SE
Washington, DC 20590-0001

16. Abstract

Pavement Condition surveys are carried out periodically to gather information on pavement distresses that will guide decision-making for maintenance and preservation. Traditional methods involve manual pavement inspections which are time-consuming and subjective. In recent times, there has been a move towards automated methods of pavement condition assessment. The automated methods which comprise of acquiring pavement data with cameras and analyzing the images have several shortcomings, especially in the area of image analysis. A major problem is that most of the image processing algorithms are based on assumptions that may not work well under certain conditions. Therefore, there is a need for adaptive image processing methods that are robust under varying conditions. This study focused on the use of multi-resolution information-mining techniques with a computer vision approach to analyze pavement conditions. A vision-system which seeks to fully-automate the pavement condition survey process is also developed. With a user-friendly interface and GIS integration, the vision system comprising of three main components; image acquisition, image retrieval and the output analysis and visualization component, this system will serve as the foundation for the future of fully-automated pavement distress surveys.

17. Key Words	18. Distribution Statement
Pavement; Computer-Visions; GIS; Pavement Monitoring; Bidimensional Empirical Mode Decomposition	

19. Security Classification (of this report)	20. Security Classification (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	153	

Table of Contents

List of Figures

List of Tables

Chapter 1

REVIEW OF AUTOMATED DISTRESS DETECTION

1.0. Introduction

The culture of road asset management requires immediate protection or preservation of road infrastructures once they are constructed or rehabilitated. Therefore, agencies are usually required to design and implement programs for pavement evaluation and performance measurement, maintenance, rehabilitation and reconstruction of the pavement structure. Acquiring precise information about road conditions is critical to accomplishing these tasks.

Traditionally, distress surveys are carried out manually. Inspectors must look at all areas of the roadway to measure its distress elements. This method is expensive, however, and risky for personnel especially on roads with high traffic volume. It is also error-prone and inconsistent. Therefore, manual road surveys are not feasible especially for network–level pavement management. The introduction of an automated system to quantify the quality of road surfaces and assist in the prioritization and maintenance planning of road networks has therefore become essential. Over the years, enormous improvements have been made regarding digital crack detection systems. Beginning as a semi-automated system, an acquisition system was used to record or grab pavement distress images while distress identification and classification was posted to an offline process. This improved the safety of conducting surveys but was still dependent on manual distress detection (Kenneth, 2004). Later on, manual distress detection was almost completely eliminated by the introduction of digital image processing and analysis. In spite of this improvement, the automated systems were still plagued with challenges such as:

shadowing from road side objects, low crack resolution and low image quality. More rapid advancement has occurred within the past five years, resulting in the development of shadow-free image acquisition and real-time processing systems (Lee 2005, Wang 2007, and Kim 2009) which are capable of conducting road condition surveys at highway speeds. This system is therefore much safer and probably the most efficient, having the potential to provide higher objectivity, as well as improved data consistency and repeatability.

The basic outputs from a pavement image analysis system include the following: distress location, distress type, distress severity and distress extent. Table 1.1 describes sample information useful for developing a road management plan.

Table 1.1: Key outputs of a pavement image analysis system

Distress Location	Distress Type	Distress Severity	Distress Extent
Road edges, central	Longitudinal, Transverse, Alligator, block crack, etc.	Low, Medium and High	Occasional, Frequent and Extensive

Pavement Inspection system consists of two important steps: Pavement Image Acquisition (image sensors such as video cameras and photo multiplier tubes) and Pavement Image Analysis (involves developing algorithms for denoising, enhancement and crack feature extraction).

1.1 Pavement Distress Data Acquisition

Different methods exist for capturing pavement surface distresses, and can mainly be grouped into three: Manual, Analogue and Digital devices. The choice of an acquisition method depends on several factors such as: cost, safety, efficiency (image quality and storage flexibility), traffic volume, etc. Very few agencies employ manual approaches alone. Usually, they combine these with any of the analogue types. Automated analogue or digital approaches are used for high traffic volume roads where safety is an issue, while manual methods can be used for low traffic volume roads.

Analogue approaches refer to a process where images are physically imposed on a film through chemical, mechanical or magnetic changes in the surface of the film. Although high quality images can be obtained from analogue approaches, they may introduce electronic noise during signal distribution and transmission which may degrade the quality of images acquired. Also, it is difficult to manipulate or integrate analogue results with other data types such as text and graphics. Analogue systems are slowly becoming extinct due to the above mentioned challenges.

On the other hand, digital approaches capture images as streams of electronic bits storing them on an electronic medium. The digital bits can be read electronically for processing or reproduction purposes. Due to their ease of manipulation and image quality, the approach is fast becoming the most popular method. Survey vehicles using digital cameras can have one or two cameras capturing the pavement image while any number are used for other data required by the user. Special lighting may also be used to overcome shadowing problems. Digital acquisition methods can be grouped mainly in two categories: Area Scan and Line Scan.

Area Scan: This is when the captured image consisting of thousands of pixels depicts some defined pavement surface area. The area covered usually depends on factors such as: camera lens, angle, placement and vehicle speed. Image distortion can result if the camera is not perpendicular to the pavement surface.

Line Scan: Builds up a 2D image line by successive line; single line scans are produced while the object moves (perpendicularly) past the line of pixels in the image sensor. The scan lines are stitched together to form a continuous image. Shadowing is a key challenge to the line scan systems. They can render images useless (because it is impossible to extract meaningful crack information).

Laser Technology: The introduction of laser technology has improved the quality of images acquired during surveys. It achieves a more accurate and efficient measurement, detecting cracks of 1 mm width. The quality of images acquired is very high.

3D Laser Imaging: GIE Laser VISION: Uses four (4) laser sensors providing 3D measurements giving the system the potential for improved distress measurement. Nonetheless, it has too low resolution, 3 mm by 110 mm.

Various attempts (Wang and Gong, 2002) have been made to compare the efficiency of these systems. It is challenging to compare because detection performance for the systems is highly dependent on the type or sets of roads being assessed. Second, systems present different survey widths making it harder to compare. Lastly, the road condition itself, the presence of non-cracked elements and the different texture backgrounds faced in each case will be decisive.

It is worth noting that the key barriers to the use of automated systems today is not the collection of images, rather, the problem lies in reliable analysis, interpretation and evaluation of the data reflecting pavement condition.

1.2 Digital Image Processing and Analysis

In this section we discuss various attempts made by researchers to develop autonomous systems capable of producing a more consistent and accurate method for pavement distress detection using digital image processing and analysis. The authors' various contributions will also be outlined clearly. The different methods for image processing have been grouped into two main categories: Spatial-Domain and Frequency domain methods.

1.2.1 Spatial-Domain Processing Methods

These methods manipulate or change the image pixels in space to enhance the image for a given application. One way in which images are processed is to look at the neighborhood around the pixel that one wishes to manipulate. The neighborhood around a given pixel is made up of the surrounding pixels depending on how one defines the pixel's neighborhood. In general, the neighborhood of i is the set of all j such that j is a neighbor of i . The value of a pixel x_i, with a sum of all j such that j is in the neighborhood of i (N_i) is given by:

$$g(x_i) = \sum_j \alpha_j x_j$$

where α_j is the weighting of a particular neighboring pixel or voxel and x_j is the value of the neighboring pixel or voxel. Spatial-domain methods are commonly used for preprocessing and enhancement of the distress images.

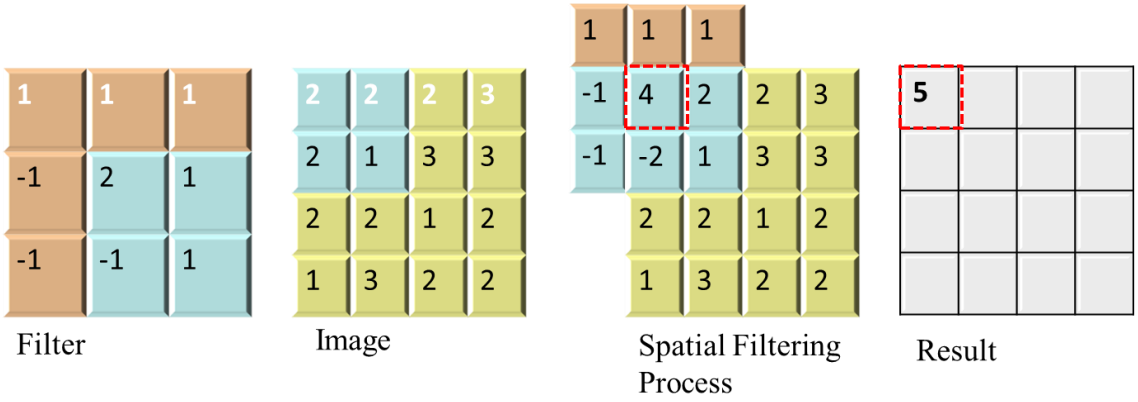
Step 1

Goal: filter the top left-most pixel

Filter: Brown regions are not used in the first step of filtering

Image: regions with aqua color shows pixels involved in the filtering process

Spatial Filtering and results: linear combination of filter weights and pixel values.



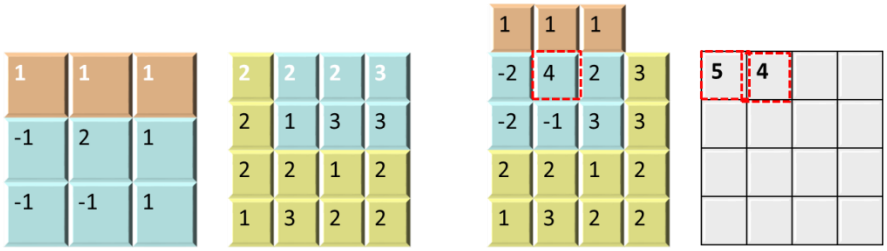
Step 2

Goal: Filter the second pixel in the first row.

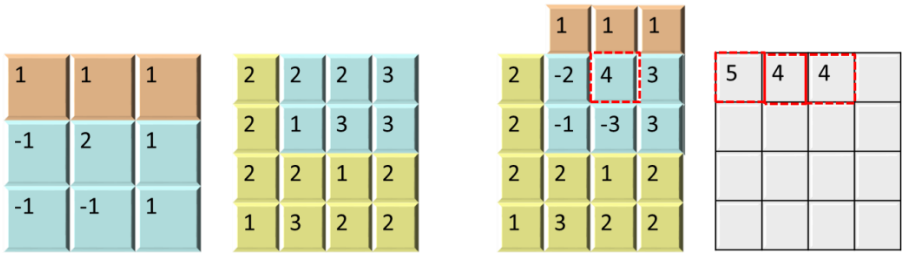
Filter: Brown regions are not used in the first step of filtering

Image: regions with aqua color shows pixels involved in the filtering process

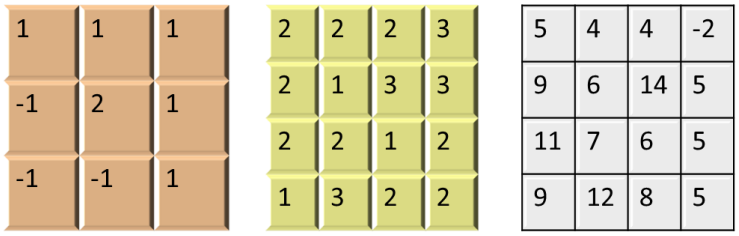
Spatial Filtering and results: linear combination of filter weights and pixel values.



Step 3



Final Result



1.2.1.1 Preprocessing and Enhancement

Image preprocessing and enhancement are very important steps to developing an efficient distress detection system. Its primary goal is to ease the cracking detection process by both reducing image noise and enhancing the visibility of linear crack features to improve the quality

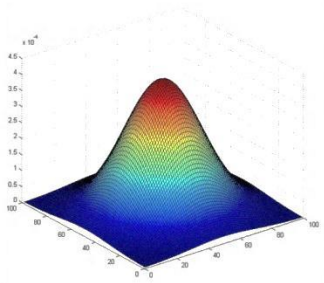
of information on the distress image. Pavement surface roughness and non-uniform illumination are the key contributors to noise in pavement distress images. Basic operations or techniques used belong to median filtering methods, neighborhood averaging and histogram equalization. Some algorithms can simultaneously enhance and preprocess the image, while others are designed for either preprocessing or enhancement.

Mraz et al. (2007) developed a statistical filtering method for enhancing crack images. A target grayscale image is captured under the same ambient temperature and lighting conditions as those of the crack imaging operation. The total noise in a given region of the images is calculated using the noise in the image of the grayscale wedge of corresponding intensity. This enhances the contrast between cracks and surrounding area (Mraz and Amarasin, 2007). In Subirat et al. (2004), a gray-scale morphological filter is used for the preprocessing and enhancement step. In grayscale morphology, a pixel is compared to those pixels surrounding it in order to keep the pixels whose values are the smallest (in the case of an erosion function) or the largest (in the case of a dilation function).These functions can be used to alter the shape of regions by expanding bright areas at the expense of dark areas and vice versa. They smoothen gradually varying patterns and increase the contrast in boundary areas. An alternative and probably more efficient method to gray-scale morphology is histogram or image equalization. This technique has been applied in Sy et al., 2008. It improves the contrast of the inspected image and helps thresholding. Histogram equalization might fail if the distress image is too noisy. This is because while it enhances the visibility of crack objects, it also amplifies image noise at the same time. Median approaches to preprocessing have also been introduced in Chambon et al., 2010.

1.2.1.2 Linear Spatial Filters

In general, linear filtering yields a new image whose pixels are a weighted sum of the original pixel values, using the same set of weights at each point. The output image is therefore a linear function of the input and also shift-invariant function of the input (i.e. shifting the image three pixels to the left, shifts the output also three pixels to the left). The averaging filter is a typical linear filter used mainly for image smoothing. It replaces the central pixel with the average pixel value of the neighborhood.

1/3	1/3	1/3	0.011	0.084	0.011
1/3	1/3	1/3	0.084	0.619	0.084
1/3	1/3	1/3	0.011	0.084	0.011



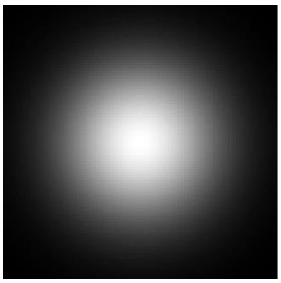


Figure 1.1: a). 3 by 3 averaging filter b). Gaussian filter. c & d). surface plot and image of Gaussian weighting filter function.

The Gaussian filter is another linear filter commonly used in practice. It uses a weighted average of pixels in the neighborhood to smoothen the image. Both the Gaussian and averaging filters are separable; hence it can be implemented with reduced computational cost. Separable filters can be generated using singular value decomposition of the 2D filter. For example:

- Let g be a 5 by 5 gaussian filter.
- $[\mathrm{U}, \mathrm{S}, \mathrm{V}] = \mathrm{svd}(\mathrm{g});$
- $\mathrm{G\_x} = -(\mathrm{U}(:,1) * \sqrt{(\mathrm{S}(1,1))});$

- $\mathrm{G\_y} = -(\mathrm{V}(:,1)' * \sqrt{(\mathrm{S}(1,1))})$;

The Gaussian filtered image is obtained by convolving image g with the 1D filter G_x in one direction and G_y in the other direction of the resulting image. Figure 1.2 shows results of smoothing pavement texture information with Gaussian and averaging filter. The averaging filter appears to saturate faster than the Gaussian. Also, both filters erase fine crack width information.

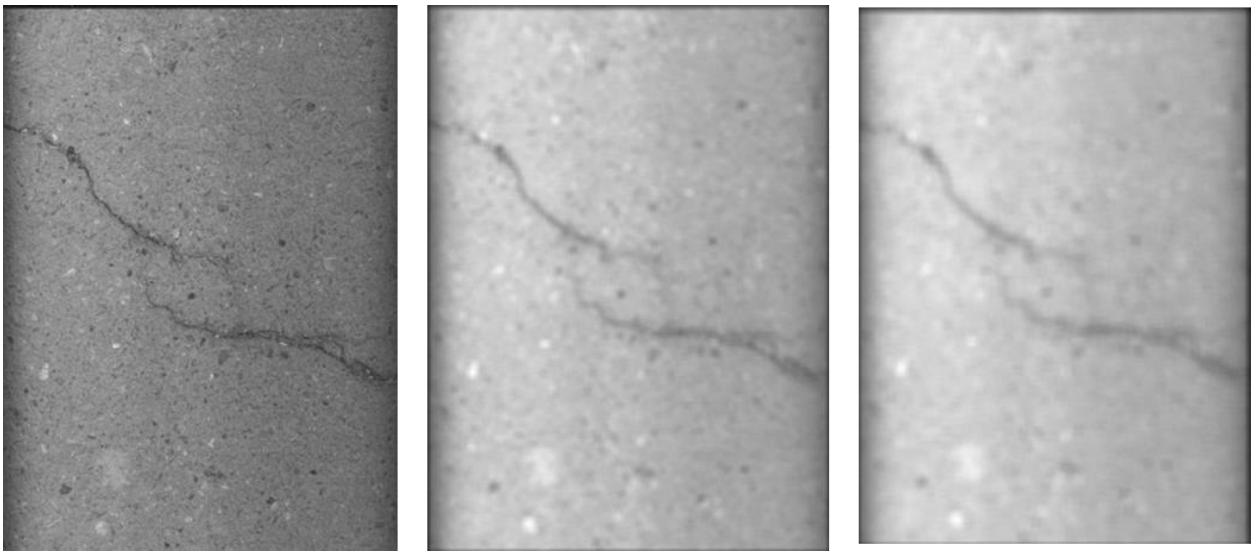


Figure 1.2: a). Original image b) filtered image with 5 by 5 Gaussian filter, $\boldsymbol{sigma} = \mathbf{5}$ c). Filtered image with a 5 by 5 averaging filter.

1.2.1.3 Median Filtering

It is well known that using linear filters such as ordinary mean or weighted averaging methods for smoothening rough textures and unwanted stains and particles on road surfaces could produce serious blurring of images and introduce false crack edges. This can affect successive post-processing such as crack width estimation, crack type classification, etc. Although they are simple and easy to implement, they is sensitive to extreme values and may render poor results for certain types of noise. As a result of the limitations of linear filters, nonlinear filters are widely

exploited due to their efficiency in noise attenuation and sharp edge detail preservation. Median filters remain the most popular non-linear low pass filter in pavement distress image analysis. The filter works by replacing the value of each pixel by the median of nine values of the adjacent eight pixels and the pixel itself (assuming a 3 by 3 window median filter). This reduces its sensitivity to extreme values and noise. However, compared to neighborhood averaging and max-min operations, the median filter is time consuming. This makes it unsatisfactory for many speed-critical applications. "Separable median" filters are often used to overcome this challenge. Separable median works by separating 2D median operations into two 1-D operations - a row operation followed by a column operation, or a column operation followed by a row operation or two orthogonal diagonal operations. Such operations are true for other methods but not the 2D-median algorithm. Let us consider for example the 3x3 matrix below;

$$A = \begin{bmatrix} 2 & 2 & 1 \\ 4 & 8 & 9 \\ 5 & 8 & 9 \end{bmatrix}$$

- Row subsets $(R_S) = [2\ 2\ 1]; [4\ 8\ 9];\ [5\ 8\ 9]$
- Median of Row subsets $(mR_S) = [2\ 8\ 8];$
- Median of Row subsets Median $(mR_s) = 8$
- $Median\ of\ A = [1\ 2\ 2\ 4\ 5\ 8\ 8\ 9\ 9] = 5;$

The results show that separable medians may not yield the same results as the 2D overall median. Conventional application of the median operation is based on two key assumptions:

1. A noise-free image consists of locally smoothly varying areas separated by edges.
2. A noisy pixel has a tendency for very high or very low gray value compared to its neighbors.

In most pavement distress images, cracked pixels have very low gray values. Assuming that very low pixel values are noise may lead to blurring or smoothening of uncorrupted cracked pixels. In view of this, we develop two important solutions by:

1. Considering noisy pixels to be of values x times higher than the median value of the neighborhood pixels. Then we separate low value crack information from noise with similar values using morphological sets.
2. Avoiding filtering of uncorrupted pixels through adaptive median filtering.

1.2.1.4 Histogram Equalization

Histogram equalization expands the pixel value distribution of an image so as to increase perceptional information. This increases the local contrast of pavement images, especially when the usable data of the image is represented by close contrast values. In distress image analysis, histogram equalization improves the efficiency of the crack detection algorithm by reducing its dependence on illumination condition and the type of textures on the road surface. While it may improve image contrast, it can also cause saturation of certain regions of the image. This has led to the introduction of adaptive algorithms discussed in the following section.

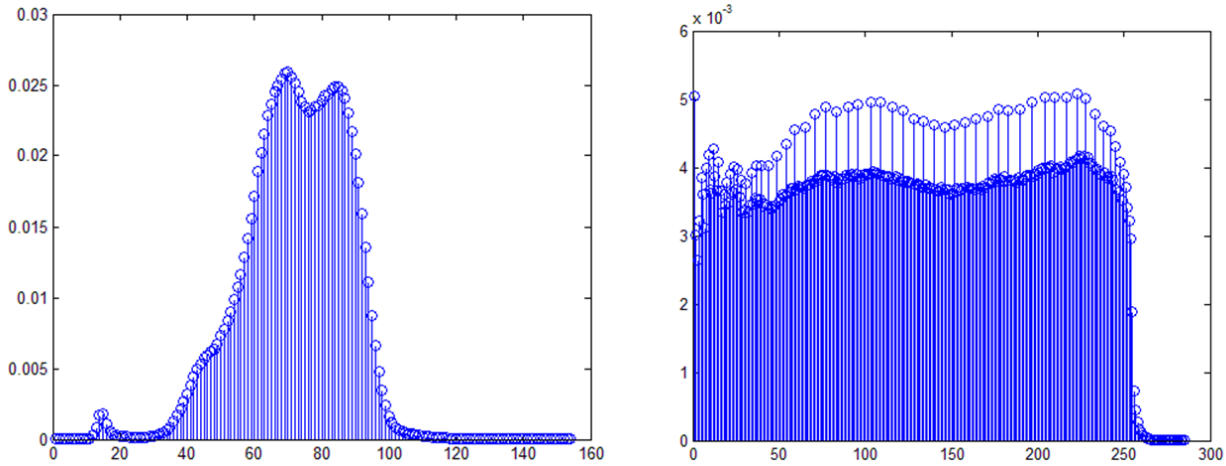


Figure 1.3: a) Original histogram of a pavement image, b) Histogram after the Histogram Equalization.

Histogram Equalization

- Compute the histogram $\boldsymbol{h(r_k)}$ of the image over the range [0 G]. Where G=255,

$$\boldsymbol{h(r_k) = n_k}$$

$\boldsymbol{r_k}$is the $\boldsymbol{kth}$ intensity level in the interval [0, G] and $\boldsymbol{n_k}$ is the number of pixels in the image whose intensity level is $\boldsymbol{r_k}$

- Compute the CDF of the resulting histogram and normalize it over [0 255] range.

$$\boldsymbol{h_{cdf} = cumsum[h(r_k)]}$$

- Finally compute the histogram equalized image using the formula

$$\boldsymbol{h(v) = round\left(\frac{cdf(v) - cdf_{min}}{(M * N) - cdf_{min}} * (L - 1)\right)}$$

where $\boldsymbol{cdf_{min}}$ is the minimum value of the cumulative distribution function and $\boldsymbol{L}$ is the number of grey levels used (256).

A simple variation of the algorithm above is to perform histogram equalization for local regions, dividing pixels in the neighborhood into low and high contrast pixels so as to avoid saturating pixels. This method is called Block-based adaptive histogram equalization.

Contrast Limited Adaptive Histogram Equalization(Zuiderveld, 1994) is a popular variation of adaptive histogram equalization for enhancing the local contrast of an image. It operates on small regions of the image rather than the whole image. The method has three main parameters: a block size is required to define the local region around a pixel for which the histogram is equalized. It is important that the block size is bigger than the size of the features to be preserved. On the other hand the number of histogram bins within each block must be less than the number of pixels in the block. A maximum slope value is used to limit the contrast stretch in the intensity transfer function.

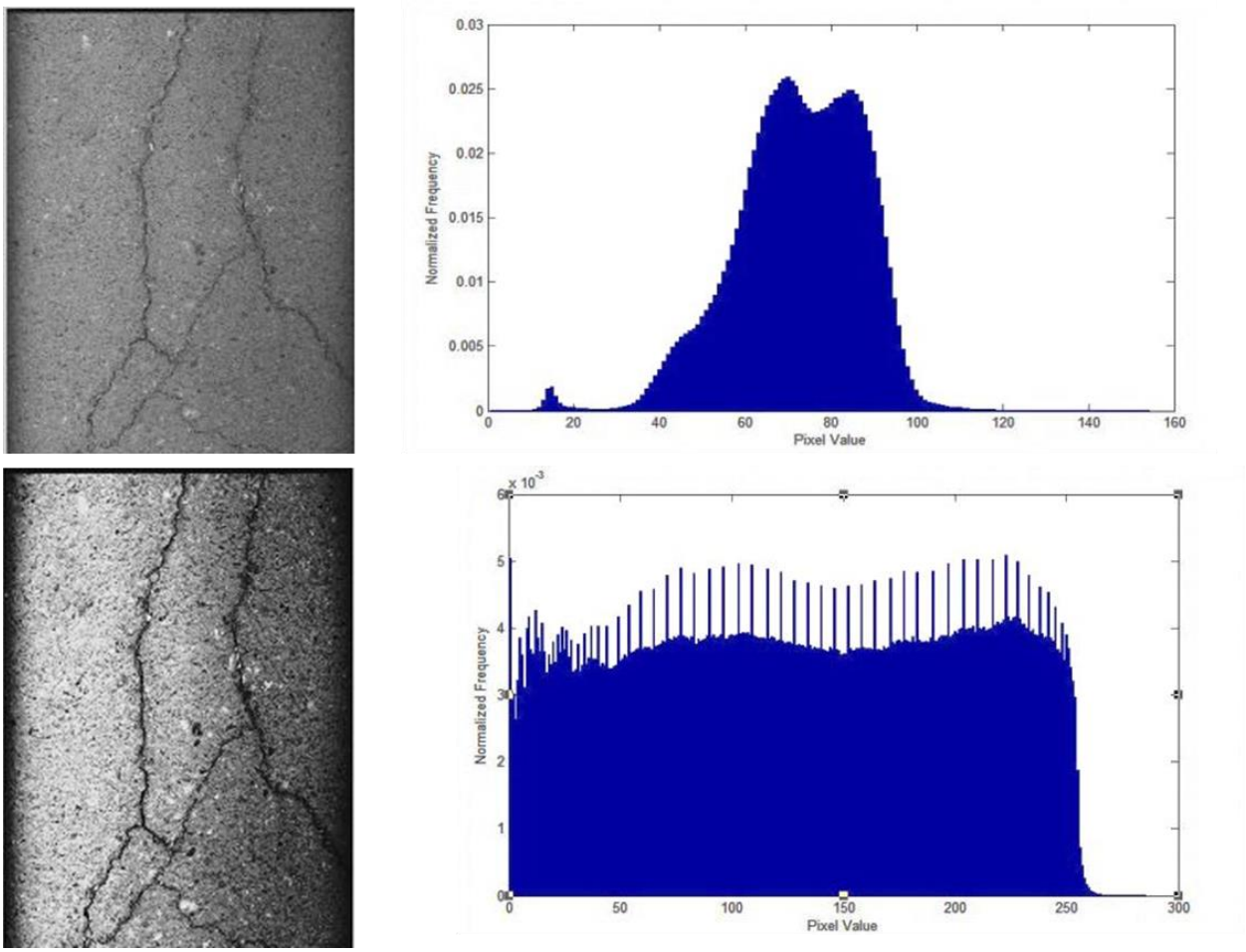


Figure1.4: First row is the original image and its histogram. The second row is the result after adaptive histogram equalization. Uniform distribution of pixel values in the histogram. Crack information is more pronounced and easily detectable.

Different variations of histogram equalization yield an appreciable enhancement result. The basic HE algorithm is not recommended in situations where specular reflections or dark stains exist on the pavement image. This is because of the problem of pixel value saturation. The choice of an adaptive algorithm should be dependent on the computation complexities involved.

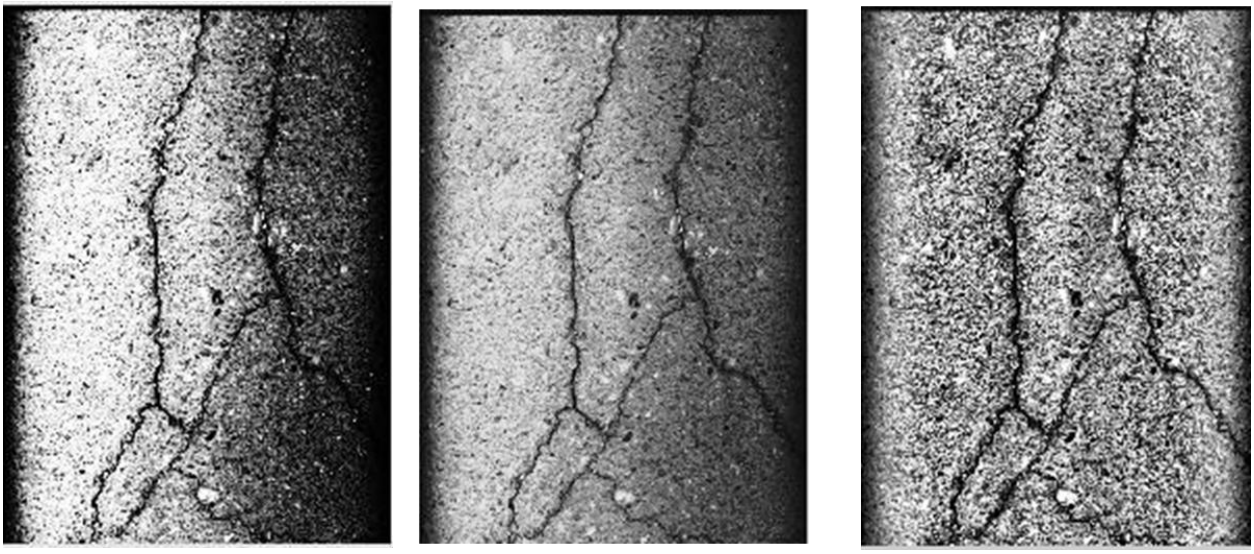


Figure 1.5: a). Normal histogram equalization. b). Block-based adaptive histogram equalization c). Contrast-limited adaptive histogram equalization.

1.2.1.5 Segmentation and Binarization

Traditional methods for extracting crack features can be grouped into two broad categories: Thresholding and Edge detection. They all detect distresses in the space-domain by a process that divides the original image into two main subgroups (cracked and non-cracked pixels). Extensive research has been carried out on this during the past two to three decades. In Chan et al., 1989), the proposed thresholding technique is based on statistical parameters of the image. Subsequent binarization is achieved by using the parameters to separate cracked regions from non-cracked ones. The algorithm has been successfully tested against linear and pattern cracks. Cheng et al., 2001 utilizes neural networks to determine the optimum threshold value for segmentation. The mean and standard deviation of the image are used as parameters to train neural networks for threshold selection. The algorithm proposed in Cheng et al., 1999, could be considered as a fuzzy set theory used to detect and segment cracks based on the fact that crack

pixels are always darker than their surroundings Cheng et al., 1999. Wang and Gong, 2005 first applied an adaptive thresholding method for segmentation. Then morphological closing was applied, followed by a noise reduction method applied to 5 by 5 pixel neighborhoods to discard small size noises from actual cracks and unconnected dots on the image. Other methods include the Otsu method, regression method, relaxation method, etc.

Thresholding is the most commonly used technique in distress image segmentation. Due to road conditions such as different distress types, different lighting or weather conditions and different external material on pavement surfaces, pavement images almost always have non-uniform backgrounds. Therefore thresholding algorithms should be preceded by an enhancement algorithm. A combination of space and frequency filtering techniques such as histogram equalization (or Median Filtering) and the Fast Fourier Transform could be used to standardize the background before segmentation is carried out.

1.2.1.6 Segmentation Algorithm Examples

Otsu Thresholding Method: This is an unsupervised method of thresholding using information from the gray-level histogram. It works well when the distress image histogram is bimodal. The algorithm searches for the threshold that minimizes the intra-class variance, defined as the weighted sum of variances of the two classes (background and object).

$$\sigma_B^2 = w_b(t)\sigma_b^2(t) + w_o(t)\sigma_o^2(t)$$

Regression Method: This method relates the optimal threshold for a given image to the mean and standard deviation of the gray level distribution through regression analysis. It is a trial and error approach which requires the user to manually set thresholds until a visually pleasing output is obtained. The process is applied to different images while taking note of the optimal threshold, mean and standard deviation. A final regression equation designed using the equation below will then be used to obtain optimal thresholds for segmenting the image database.

$$\bar{T} = k + a_1\mu + a_1\sigma^2$$

1.2.2 Frequency and or time-based Image Analysis methods

Information within a distress image could be overwhelming and complex. It may contain different types of distresses; surface texture could also be different at different locations; there could be combinations of non-uniform backgrounds, different lighting and weather conditions. This is likely to create problems for space-domain processing methods. Frequency-based techniques transform the different information in the image into narrow bands which makes these methods easy for isolating useful information for analysis.

A key foundation for linear filtering in both spatial and frequency domains is the convolution theorem. In Fourier analysis, the convolution of two functions $f(x, y)$ and $h(x, y)$ is equal to the inverse of the product of the transform of the two functions in frequency domain. That is:

$$f(x, y) * h(x, y) \ll=\gg H(u, v)F(u, v)$$

Conversely, the product of the two functions in the spatial domain is the same as the convolution in the frequency domain.

$$f(x, y)h(x, y) \ll=\gg H(u, v) * F(u, v)$$

$H(u,v)$is usually referred to as the filter transfer function. In spatial domain $h(x,y)$ is the filter mask. The basic idea of frequency domain filtering is to select a filter transfer function $H(u,v)$ that modifies the transformed image in a specified manner. Two main frequency and or time based methods will be described in the following sections: the Fourier Transform and the Wavelet Transform.

1.2.2.1 Fourier Transform

The Fourier Transform decomposes an image into its sine and cosine components. In the Fourier domain, each point represents a particular frequency contained in the spatial domain image. A 2-D discrete Fourier transform $F(u,v)$ is given by:

$$F(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)e^{-j2\pi(\frac{\mu x}{M}+\frac{vy}{N})}$$

For $u = 0,1,2,\dots\dots,M-1$ and $v = 0,1,2,\dots\dots\dots,N-1$

The inverse Fourier transform is given by:

$$F(x,y) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{j2\pi(\frac{\mu x}{M}+\frac{vy}{N})}$$

For $x = 0,1,2,\dots\dots,M-1$ and $y = 0,1,2,\dots\dots\dots,N-1,$ $F(u,v)$ is sometimes called the Fourier coefficients.

Pavement texture smoothening and feature extraction with Fourier Transform

- Subdivide the image into small processing blocks (64 by 64 pixels). This enhances processing speed of the system.
- Obtain the Fourier transform for each block with padding:

  $\boldsymbol{F = FFT\_2D(f(x,y), [P\ D])}$**:** Where $\boldsymbol{P}$ and $\boldsymbol{D}$ is the resulting row and column size of the processed image after padding.

  $\boldsymbol{P \geq row\_length(f) + row\_length(h)}$; $\boldsymbol{h}$ is a low pass transfer filter function.

  $$\boldsymbol{Q \geq column\_length(f) + column\_length(h)}$$

- Generate a low pass filter function,$\boldsymbol{H_{lp}}$, of size $\boldsymbol{P\ by\ Q}$ as shown in figure 1.6. Low pass filters are used for enhancement while high pass filters are for edge detection.
  - For crack feature extraction, the filter function should be high pass, $\boldsymbol{H_{hp}}$

  $$\boldsymbol{H_{hp} = 1 - H_{lp}}$$

- Multiply the transform $\boldsymbol{F}$ by the filter $\boldsymbol{H}$ to attenuate noise and very high frequencies. Obtain

  $$\boldsymbol{G = FFT\_2D\_INV(H.* F)}$$

  $\boldsymbol{FT_2D_INV}$is given by the equation above

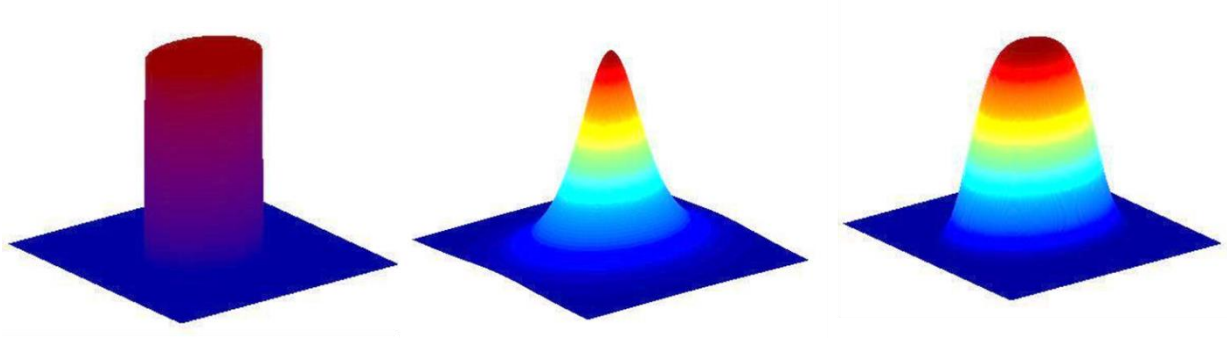


Figure 1.6: A plot of different transfer filter functions. A): Ideal low pass filter. B). Butterworth filter C). Gaussian filter.

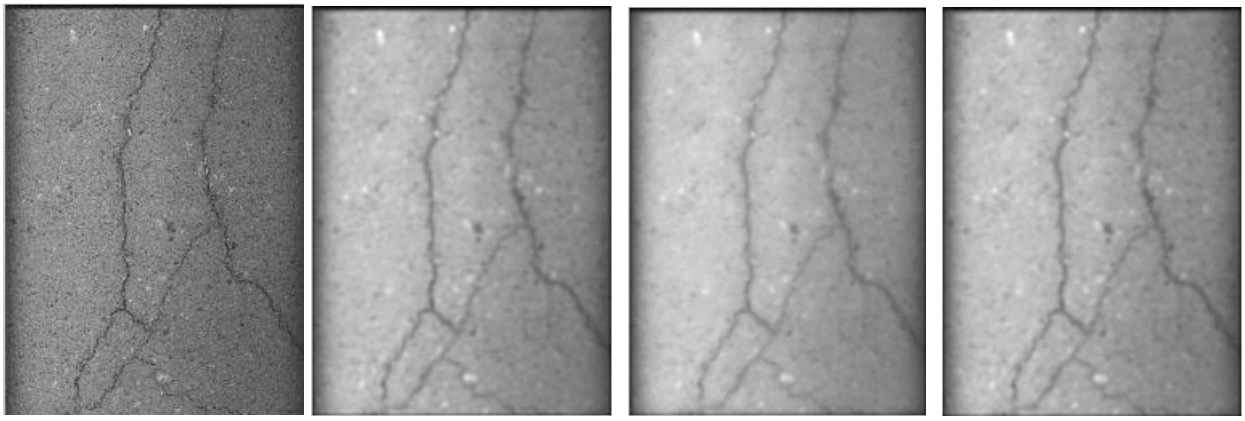


Figure 1.7: a). Original Image. The remaining images are results of frequency filtering using b).Ideal low pass (**cutoff frequency = 3% max size of image**). c).Butterworth filter (**cutoff frequency = 3% max size of image**) and d).Gaussian filter.

1.2.2.2 Feature Extraction

Crack features can be obtained by thresholding the high pass image. It should be noted that the high pass image is generated from the low pass image, not the original image. This is to reduce the effect of noise in our results. Results are shown in figure 1.8 below.

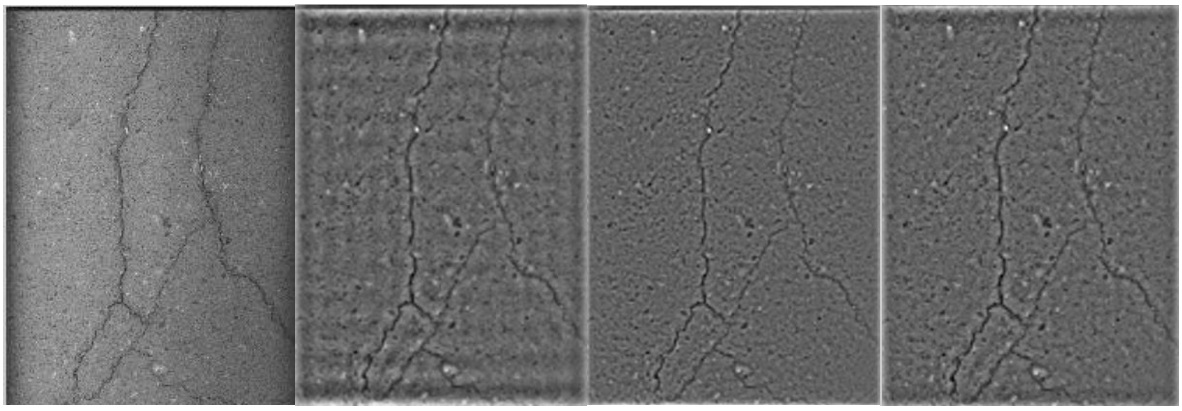
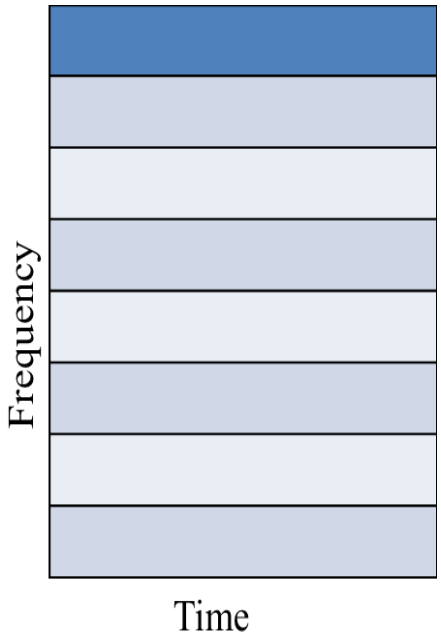


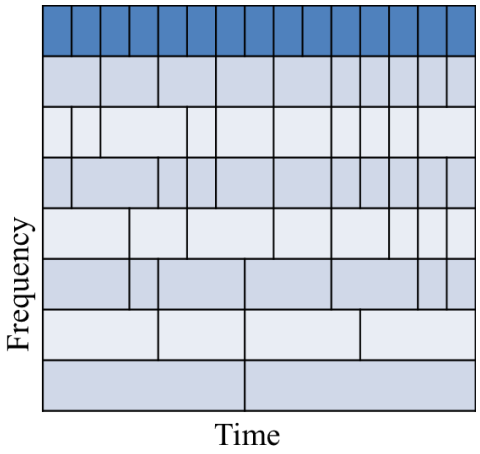
Figure 1.8: a). Original Image. The remaining images are results of frequency filtering using b).Ideal high pass (**cutoff frequency = 3% max size of image**) – introduces artifaces, high fidelity to crack widths, however, thresholding may be difficult. C). Butterworth filter (**cutoff frequency = 3% max size of image**) - reduced crack widths, noise well suppressed. d).Gaussian filter – high fidelity to crack widths.

1.2.2.3 Limitation of Fourier



The Fourier transform gives information only in the frequency or time domain, but not simultaneously. As an example, the diagram on the left shows different frequencies that may exist in a dataset, however, it is not possible to know where they occur in the time domain. As a result it is challenging to use when the local statistics of the image being processed are highly non-stationary. This results in the introduction of multiresolution decomposition methods which can give time-frequency information helpful for analysis of nonstationary images.

1.2.2.4 Wavelet Methods



The wavelet methods process distress images in both time and frequency domain. Comparing to the previous diagram, the wavelet, is able to resolve time and frequency information simultaneously.

The wavelet transform can be described as a standalone tool for pavement distress analysis. This is mainly because all the key parameters required for road condition assessment can be obtained in the wavelet domain. The following section briefly describes application of the wavelet transform for processing distress images.

Wavelet Image Processing Summary:

- Image is decomposed into k levels (3 levels usually used); $\boldsymbol{HL_k}$, $\boldsymbol{LH_k}$, $\boldsymbol{HH_k}$ $\boldsymbol{and}$ $\boldsymbol{LL_k}$. $\boldsymbol{HL_k}$, $\boldsymbol{LH_k}$, $\boldsymbol{HH_k}$ are called details in the high frequency horizontal, vertical and diagonal subbands at level k respectively. $\boldsymbol{LL_k}$denotes the approximation in the low frequency subband.

Assumptions

- Distresses which are high frequency components are most likely transformed into high amplitude wavelet coefficients.
- Noise will be transformed into low amplitude wavelet coefficients.
- Background information is low frequency, hence transformed into high amplitude wavelet coefficients in the low frequency subband.

Table 1.2: Comparison between Wavelet and Fourier Transform

Wavelet	**Fourier**
They both analyze a signal by decomposition into constituent parts	
Signal is decomposed into wavelets, scaled and shifted versions of the mother wavelet	Signal is decomposed into a series of sine waves with different frequencies
Irregular in shape and compactly supported. This makes it suitable for non-stationary signals with sharp changes or discontinuities. Compactly supported nature enables temporal localization of signal's features.	Sinusoidal wave is smooth and of infinite length.

1.2.2.5 Image Denoising and Distress Detection

Image Denoising or approximation in the wavelet domain is obtained by removing low amplitude coefficients (setting them to 0). Denoising in the wavelet domain involves applying hard or soft thresholds to the detail coefficients of the wavelet transform. An inverse transform of the result will yield the denoised image. Figure 1.9 below shows the result of denoising a pavement distress image using a hard threshold. Cracked pixels can easily be differentiated from the background. Texture information is, however, still persistent and could affect segmentation.

For distress detection and segmentation, the following procedure is followed:

- Compute the wavelet modulus$M_k(x,y)$: a combination of horizontal, vertical and diagonal details from the wavelet coefficients.

  $M_k(x,y) = \sqrt{HL_k^2(x,y) + LH_k^2(x,y) + HH_k^2(x,y)}$:

- Binarize the wavelet modulus image

$$D_1(x,y) = \begin{cases} 1 & if\ M_1(x,y) \geq k \\ 0 & if\ M_1(x,y) < k \end{cases}$$

  $D_1(x,y)$ is the binarized image, and $k = \sqrt{2\sigma^2 \log(n)}$; is the threshold for wavelet modulus. σ is the noise power and n is the size of the subband containing the contributed wavelet coefficient.

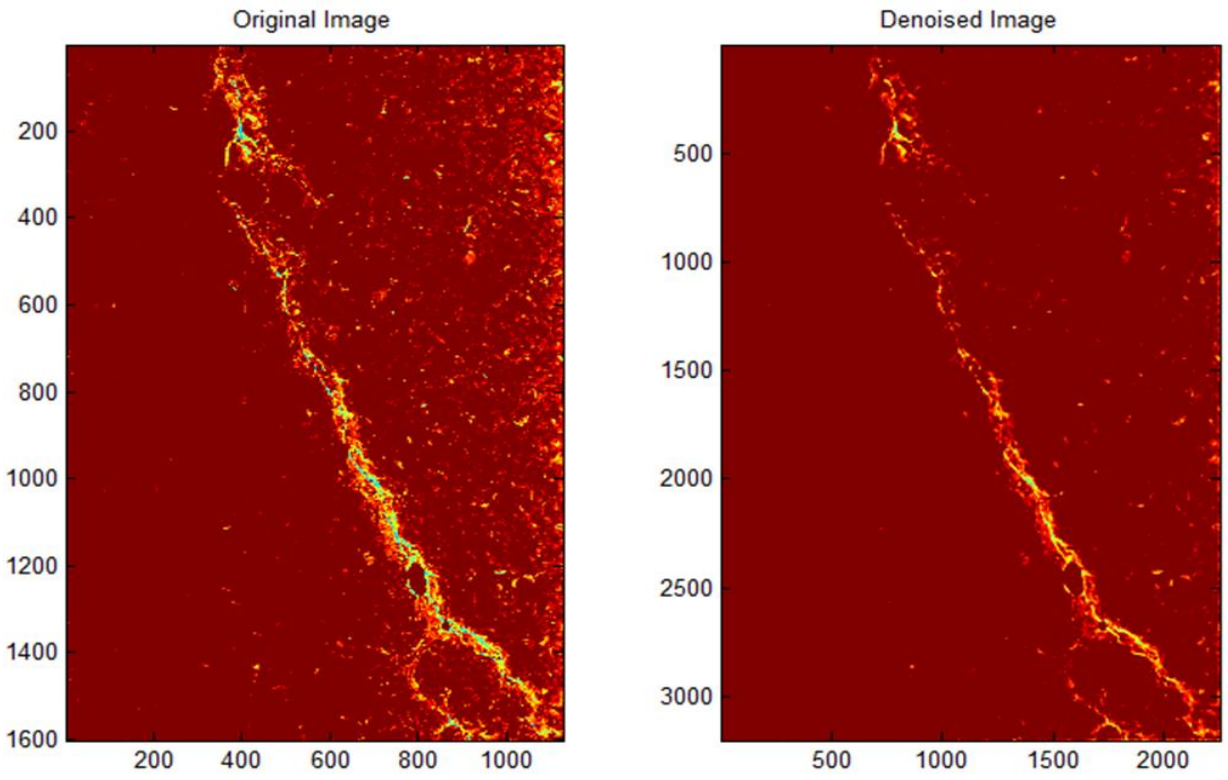


Figure 1.9: Results of image denoising based on hard thresholding of Haar wavelet transform coefficients.

Figure 1.10 Below shows the efficiency of the procedure described above. In the absence of severe scene illumination and rough texture, the performance of the wavelet is ideal although fine cracks could be compromised during the denoising process. The algorithm's performance in the presence of shadows is not very appreciable as shown in the last column of figure 1.10 below. The computational cost of using the wavelet for crack detection is relatively low as compared to other adaptive methods like empirical mode decomposition and adaptive median and thresholding algorithms.

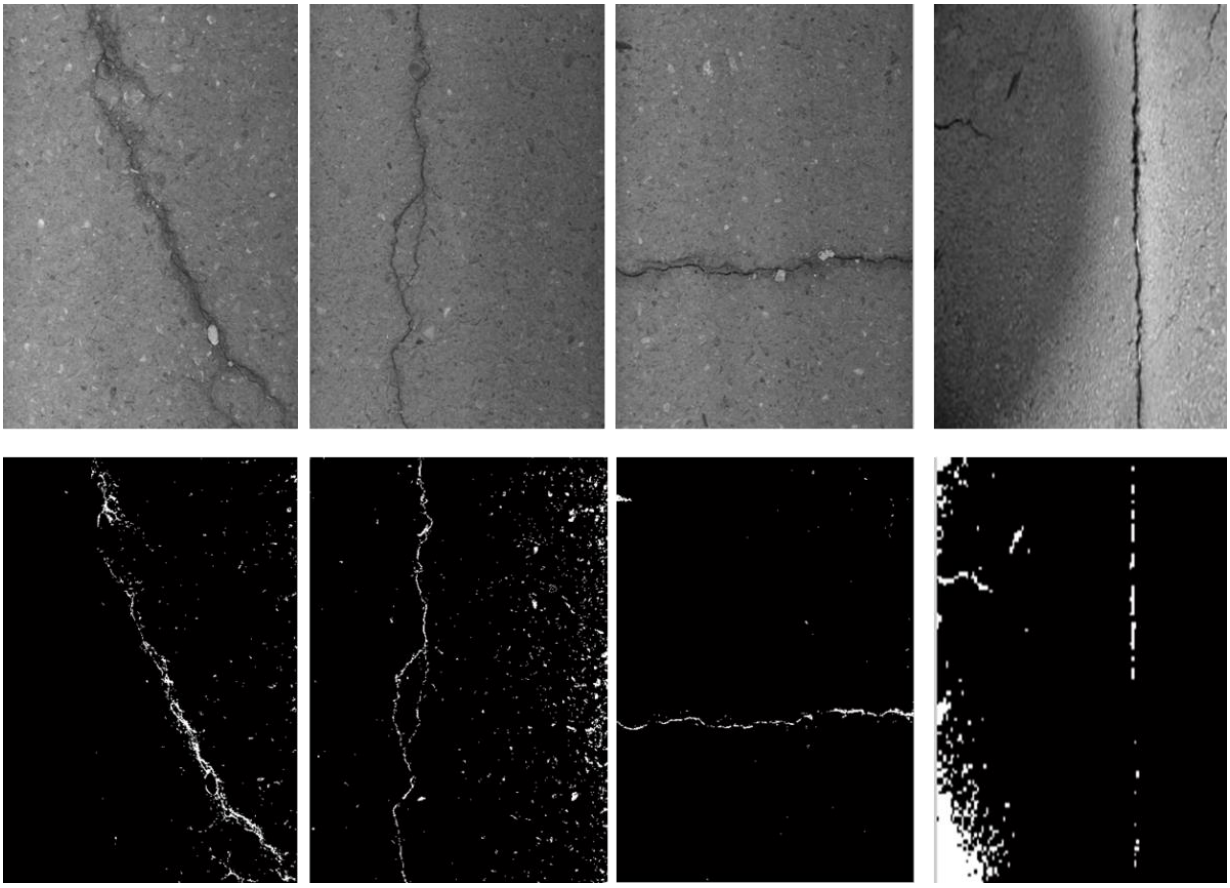


Figure 1.10: Distress detection in the wavelet domain.

1.3 Summary

The chapter presented a comprehensive review and advancements in current algorithms for automatic distress image processing and detection. Below is a list of key achievements and challenges of current detection systems.

Achievements

- In spite of the idealistic assumptions of traditional systems, they provide a safe, objective and relatively more effective means of conducting road condition surveys. Rapid advancements in pavement image data acquisition have resulted in the development of

high resolution cameras which are able to detect sub-millimeter crack widths and eliminate shadowing problems.

- Image enhancements through adaptive spatial filters are strongly recommended due to their easy manipulation. The use of adaptive median filter and histogram equalization reduces smoothing of uncorrupted pixels. This ensures that fine cracks are not erased during the preprocessing and enhancement process. Most importantly, combining spatial filters with frequency domain filtering improved the denoising process.

Challenges

- Feature Extraction: Firstly, traditional gradient based segmentation and thresholding algorithms are sensitive to noisy pixels and specular effects on the surfaces of the pavement image. As a result, false cracks could be detected on non-cracked images. Usually, robust post-processing algorithms are suggested which increases the complexity of the system. Secondly, during the crack detection process, pavement cracks are assumed to be the objects with lowest pixel values. This may not be necessarily true if the pavement image acquired is corrupted with noise. Also, thresholding algorithms assume slow pixel value variability in the pavement images.
- Frequency Based Processing: Although the application of the wavelet transform for pavement crack detection is efficient for images that are highly non-stationary, the choice of a basis function remains an important factor in the quality of the processed output. There exist no automated means for selecting basis functions for different types of images.

REFERENCES

Chambon, Sylvie., Gourraud, Christian ., Moliard, Jean-Marc., Nicolle, Philippe., (2010): Road Crack Extraction with Adapted Filtering and Markov Model-based Segmentation - Introduction and Validation. VISAPP (2) 2010: pp. 81- 90.

Chan, K. B., Soetandio, S., and Lytton, R. L. (1989). "Distress identification by an automatic thresholding technique," presented at the International Conference on Application of Advanced Technologies in Transportation Engineering, San Diego, Calif.

Cheng, H., Wang, J., Hu, Y., Glazier, C., Shi, X., Chen, X., (2001). "Novel Approach To Pavement Cracking Detection Based On Neural Network". Transportation Research Record, Vol. 1764, pp. 119-127.

Costa, (1995). "Algorithm for pavement distress classification by video image analysis", PHD Dissertation, Case Western University.

Cheng, D., Chen, J-R., Glazier, C., and Hu, Y. G., (1999). Journal of Computing in Civil Engineering, Vol. 13, No. 4, October 1999, pp. 270-280.

Kim, J. Y. (2009). "Development of New Automated Crack Measurement Algorithm using Laser Images of Pavement Surface." PHD Dissertation, 2009., University of Iowa.

Kirschke, K. R., Velinsky, S. A., (1992). "Histogram-Based Approach for Automated Pavement –Crack Sensing",Journal of Transportation Engineering, Vol. 118, No. 5, September/October 1992, pp. 700-710.

Lee, H.D. (2005). “Development of a Manual Crack Quantification and Automated Crack Measurement System.” Project TR-457, Public Center, Civil and Environmental Engineering, University of Iowa.

Lihao Hong; Salari, E.; Chou, E. (2010), "Pavement information system: Detection, classification and evaluation," Electro/Information Technology (EIT), 2010 IEEE International Conference on , vol., no., pp. 1-5.

McGhee, K. H., (2004). “Automated Pavement Distress Collection Techniques”, Transportation Research Board, NCHRP Synthesis 334, Washington D.C, USA, 2004.

Mraz, Alexander, S. Amarasin, G., Manjriker, N. Abdenour “Innovative methods for enhancing pavement crack images” Transportation Research Board, 2007, Washington, DC 20001 USA.

Ritchie, S G, Kaseko, M, Bavarian, B., (1992) “Development of an intelligent system for automated pavement evaluation”. Transportation Research Board, Washington, DC 20001 USA.

Subirats, P., Fabre, O., Dumoulin, J., Legeay , V., and Barba, D., (2006) “A combined wavelet-based image processing method for emergent crack detection on pavement surface images”, European Signal Processing Conference, Florence, Italy, September 4-8, 2006.

Salari, E.; Bao, G. (2010), "Pavement distress detection and classification using feature mapping," Electro/Information Technology (EIT), 2010 IEEE International Conference on , vol., no., pp. 1-5.

Sy, N.T.; Avila, M.; Begot, S.; Bardet, J.C.; , "Detection of defects in road surface by a vision system," Electrotechnical Conference, 2008. MELECON 2008. The 14th IEEE Mediterranean , vol., no., pp. 847-851, 5-7 May 2008.

Wang, K. C. P. (2004). “Challenges and Feasibility for Comprehensive Automated Survey of Pavement Conditions.” Proc., Int. Conf. on Applications of Advanced Technologies in Transportation Engineering., ASCE, Reston, Va., pp. 531–536.

Wang, K.C.P., Gong, W., (2005). “Real Time Automated Survey System”, ASCE Journal of Infrastructure Systems, Vol. 11, No. 3, pp. 154-164.

Wang, K.C.P., Gong, W., 2002 “Automated Pavement Distress Survey: A Review and A New Direction”, Pavement Evaluation Conference, 21-25, Roanoke, Virginia

Zuiderveld, Karel (1994), "Contrast limited adaptive histogram equalization", Graphics gems IV, Academic Press Professional, Inc., pp. 474– 485.

Chapter 2

PAVEMENT DISTRESS IMAGE ANALYSIS USING MULTI-RESOLUTION INFORMATION MINING

2.0. General Background

The consistency and accuracy of road condition rating and analysis has considerably improved with the introduction of Automated Digital Image Processing (ADIP) routines discussed in Chapter 1. The ADIP presents more objective, robust and faster procedures than traditional manual rating methods. The ADIP techniques are, however, confronted with a number of challenges, some of which are a result of idealistic assumptions used to develop the algorithms which run the systems. For example, in most ADIP systems, images are acquired with special lighting systems at night, off-peak hours or under very controlled environments. This is mainly to avoid the introduction of artifacts or noise from trees or objects by the road side. Such a well-controlled environment that could isolate noise during the acquisition process is very difficult to achieve in practice; hence, pavement images are almost always corrupted with background illumination variation from single or multiple sources. During the crack detection process, pavement cracks are assumed to be the objects with lowest pixel values. This may not be necessarily true if the pavement image acquired is corrupted with noise. Also, thresholding algorithms assume slow pixel value variability in the pavement images.

Such assumptions and restrictions could affect the ADIP's efficiency and overall, its ability to distinguish between the different types of cracks or image background information such as shadows, oil stains, pavement texture, etc. An improved ADIP should therefore be able to clearly distinguish between three key pavement image components; background, pavement texture and pavement cracks. This challenge calls for image analysis techniques which have no special requirements for the input pavement image; it should be data-dependent and adaptive.

In recent years, extensive research has been conducted into more efficient ADIP methods such as the use of Wavelet and Fourier transforms. These techniques are usually combined with edge detection and or thresholding techniques to extract cracks from images. Abdel-Qader et al., 2003 and Hutchinson and Chen 2006 present a comprehensive review and comparisons of the effectiveness of these transform techniques and the various edge detection techniques such as the Sobel filter and Canny filter. Despite the strength of these techniques, they are not fully adaptive. Fourier transforms and most traditional denoising or smoothening filters assume that the localized image statistics are stationary. They are therefore ineffective when applied to pavement images with a wide range of crack widths; smaller cracks will be erased. The choice of a pre-determined mother wavelet also renders Wavelet transforms not completely data driven and fully adaptive.

With regard to separating background and pavement texture information from the cracks, other improved methods such as the median filter-based approach developed in Fujita et al. (2009), Bayesian approaches, and multiscale space techniques (Xianglong and Qingquan 2006) are also being investigated. In Fujita et al. (2006), a median filtering approach is used to smoothen the pavement image texture while isolating pavement image background variations (especially shadows) simultaneously. This preprocessing technique, although effective, is limited to images

acquired in very simplistic or ideal environments. In complex scenarios where pavement texture is very rough with different sources of illumination, the median filter erases some crack information and also could lead to saturation of light areas. Several similar proposed methodologies (Liang and Salari 2009, Cheng and Miyojim1998) suffer the same challenges mainly because they assume that the background illumination may vary smoothly. A comprehensive survey of design and implementation issues related to pavement distress analysis systems is discussed by Wang et al., (2004).

Pavement images are acquired with a variety of devices under non-uniform distributed lighting (Cheng and Miyojim 1998)**.** The problem with varying illumination usually results from the use of one or more primary sources of light such as camera flashes, or special lights in current pavement distress detection systems. It is common to use a flash especially when there is shade on the pavement. The problem arises with uneven coating surfaces where using flash may create a number of specular reflections in the image acquired. As a result, most detection systems undergo very cumbersome and laborious pre- and post-processing routines which affect the integrity of the post-processed image. In this work, a more robust approach is developed to standardize corrupted images instead of imposing certain generic assumptions about the images.

Recent development of high resolution cameras and laser-based image acquisition techniques has improved pavement image quality and resolution of cracks in classical pavement crack detection systems (Lee 2005, Wang 2007, and Kim 2009). These advancements, however, introduce a major challenge in pavement image analysis. Improving the resolution of cracks also increases the amount of noise contributed from the pavement texture. A common tendency to reduce the effect of texture noise is to use low spatial resolution images. Apart from the fact that low spatial resolution images might erase crack edges, the noise in pavement images does not always reside

in a particular frequency; they are intermittent. For example, pavement textures (noise in this case) could be smaller, similar or wider than the crack itself. In this case, a normal convolution with certain pre-defined basis functions will not be able to filter out the noise in the image. To efficiently extract and classify pavement crack textures, it is necessary to emphasize the specific features of interest while other redundant information is de-emphasized. This results in the need for multi-resolution image information mining for optimal, fast and reliable pre-processing of images, which is particularly important for images recorded in real world situations.

Multiresolution analysis allows for the preservation of an image according to certain levels of resolution or blurring. Broadly speaking, multiresolution analysis allows for zooming in and out on the underlying texture structure (Alzu'bi and A. Amira, 2010). Therefore, the texture extraction is not affected by the size of the pixel neighborhood. In this chapter, we introduce the application of a unique multiresolution image analysis technique known as Bidimensional Empirical Mode Decomposition (BEMD). BEMD is a 2D extension of the EMD explained in chapter 2. BEMD can be seen as a filtering process which results in a set of narrow band components called Bidimensional Intrinsic Mode Functions (BIMFs). These BIMFs essentially reflect variations in the spatial frequency of the original pavement image. We therefore use this capability to extract various linear patterns of interest while isolating the slowly varying background pavement image information.

BEMD was first applied to pavement images in (Ayenu-Prah and Attoh-Okine 2008) mainly for the purpose of denoising. This work improves the denoising methodology proposed and explores new areas for the application of BEMD in pavement image analysis. One major challenge of using the BEMD technique has to do with the reconstruction of the BIMFs with a goal of obtaining the desired objective of decomposition. So far, reconstruction techniques employed in

the EMD could be optimal (Weng and Barner 2007), complete (Bhagavatula and Savvides 2007) and selective (Ayenu-Prah and Attoh-Okine 2008) procedures. Complete reconstruction techniques could produce redundant information in the composite image due to the inclusion of detailed components that might not offer good discrimination ability. This could also slow the system because it requires more computation time. Selective procedures lack optimality and are also susceptible to false edge generation. Selective procedures compromise either the frequency or spatial resolution of the selected image. A procedure that compromises between the two (spatial and frequency resolution) is required for accurate classification. Optimal reconstruction techniques remain the future direction for BIMF reconstruction. However, current methods such as Least Mean Square (LMS) technique require a "desired image" to generate reasonable weights for reconstruction; this may be impractical to obtain under real world situations.

The first goal of this chapter is to develop an image processing system based on the multiresolution analysis techniques, BEMD, that will simultaneously and adaptively denoise (smooth textures and isolate foreign objects) and standardize the background (remove background illumination) of a pavement image without any prior requirement for the image. The second goal is to reconstruct a composite image which records only salient information (crack features) from both fine and coarse scales of the BIMFs while refraining from combining complementary and redundant information.

2.1. Bidimensional Empirical Mode Decomposition

The Bidimensional Empirical Mode decomposition is a 2-dimensional extension of empirical mode decomposition described in earlier chapters. To avoid repetition, we summarize the methodology of the BEMD in figure 2.1 below.

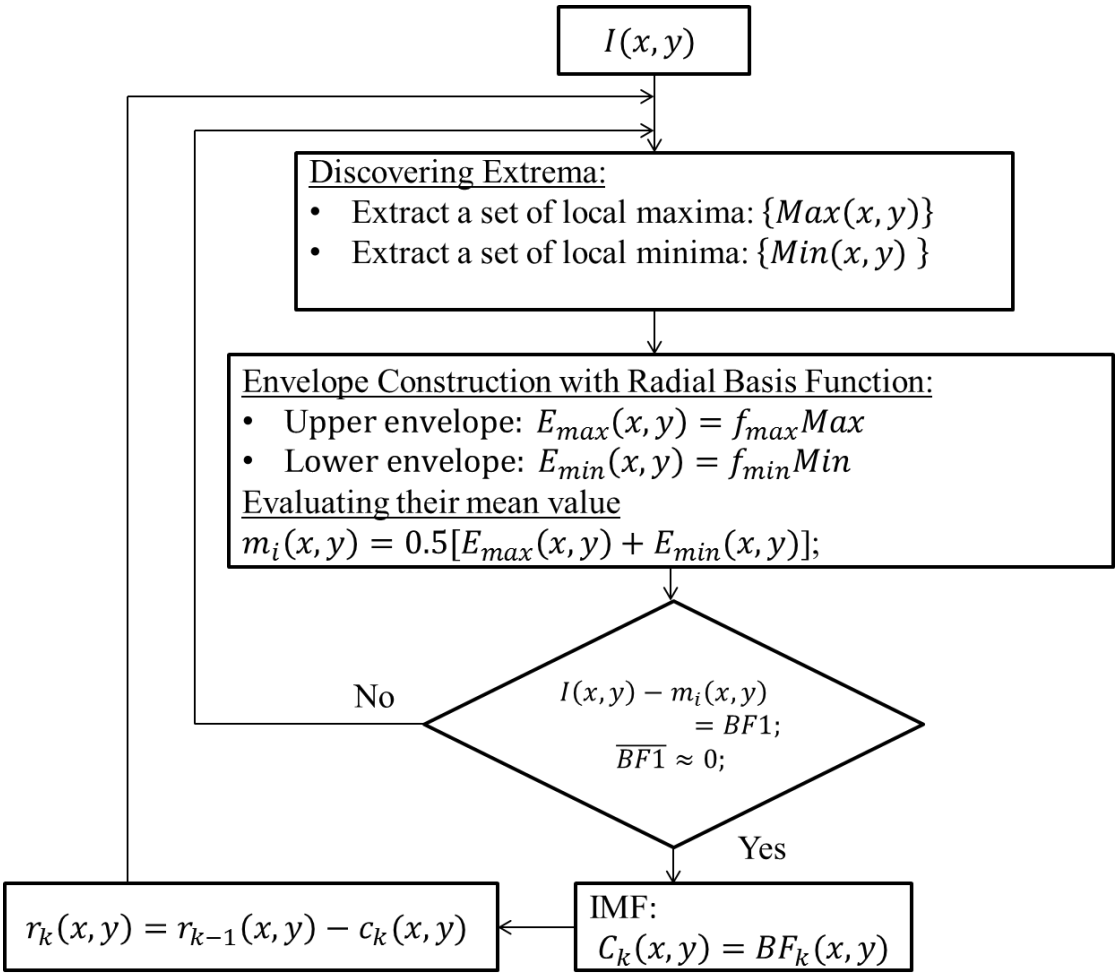


Figure 2.1: The sifting process (Klionski et al., 2008)

Extrema Detection and Interpolation: Identify local maxima, $\boldsymbol{Max}$ and minima, $\boldsymbol{Min}$ using a 3 by 3 neighborhood window. Using Radial Basis Function interpolation, construct a maxima and minima envelope from the extrema detected.

$$E_{max}(x,y) = f_{max}(Max,x,y) \quad (1)$$

$$E_{min}(x,y) = f_{min}(Max,x,y) \quad (2)$$

The remaining steps are similar to the ones discussed in chapter 2.

2.2. Pavement Distress Image Processing in BEMD Domain

In chapter 1, we discussed various techniques for preprocessing the distress images in order to enhance the efficiency of crack object detection algorithms. In the BEMD domain, these techniques can be implemented in a more efficient and simpler manner.

2.2.1. Image Enhancement

In spite of recent advances leading to the use of laser-based imaging to acquire quality (no shadows) images, low-cost pavement distress surveying (Ahmed and Haas 2010) still remains attractive as long as there are efficient algorithms to standardize the images acquired. The varying background brightness of a pavement distress image could be caused by shadows of road-side objects, uneven lightening from image acquisition equipment or even the type of material used in constructing the pavement. An important step in pavement image background removal is the characterization of the different variations in the image. An intensity matrix from a pavement distress image usually contains three types of variations: (1) Incident non-uniform illumination effects; (2) Pavement texture which usually consists of the pavement distresses and irregularities such as oil stains and other materials on the surface; (3) Image noise which is usually caused by poor quality data acquisition, heterogeneous materials and granularity. Hence, a generalized pavement image model is usually expressed as:

$$I(x,y) = I_{back}(x,y) + I_{crack}(x,y) + I_{noise}(x,y) \qquad (3)$$

$I_{back}$is the background illumination or variation; $I_{noise}$is the image noise and I_{crack}epresents the cracks in the image. In this section we assume that $I_{noise}$is the image noise which is introduced mainly by pavement texture (not from poor data acquisition) and other irregularities such as oil stains and other materials on the surface which might be falsely detected as cracks. We seek to

smoothen out as much as possible the irregularities introduced by the pavement's texture without erasing the cracks.

Usually, image noise is characterized as a high-frequency and low-to medium amplitude signal, however, the behavior of I_{noise}might be different for different distress images. These high frequency and low amplitude data are usually extracted into the low index Intrinsic Mode functions by BEMD. This is, however, dependent on the type of pavement being analyzed. Sometimes this component is mixed with fine crack features as shown in the figure 2.2 &2.3 below.

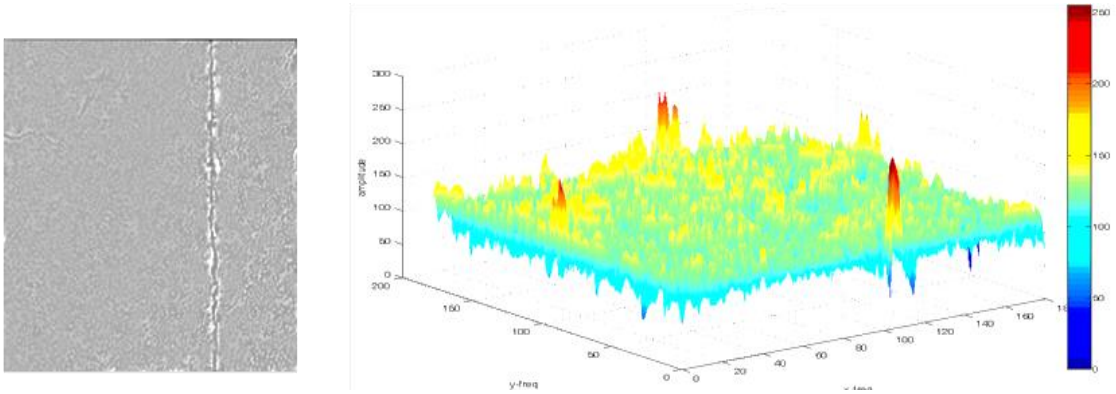


Figure 2.2: First intrinsic mode function and its mesh plot. The mode contains both texture features and fine edges of crack features.

Crack features $I_{crack}(x, y)$are usually collated in more than a single IMF. In later sections, we will describe how to combine crack information from the different scales to obtain just the salient crack information.

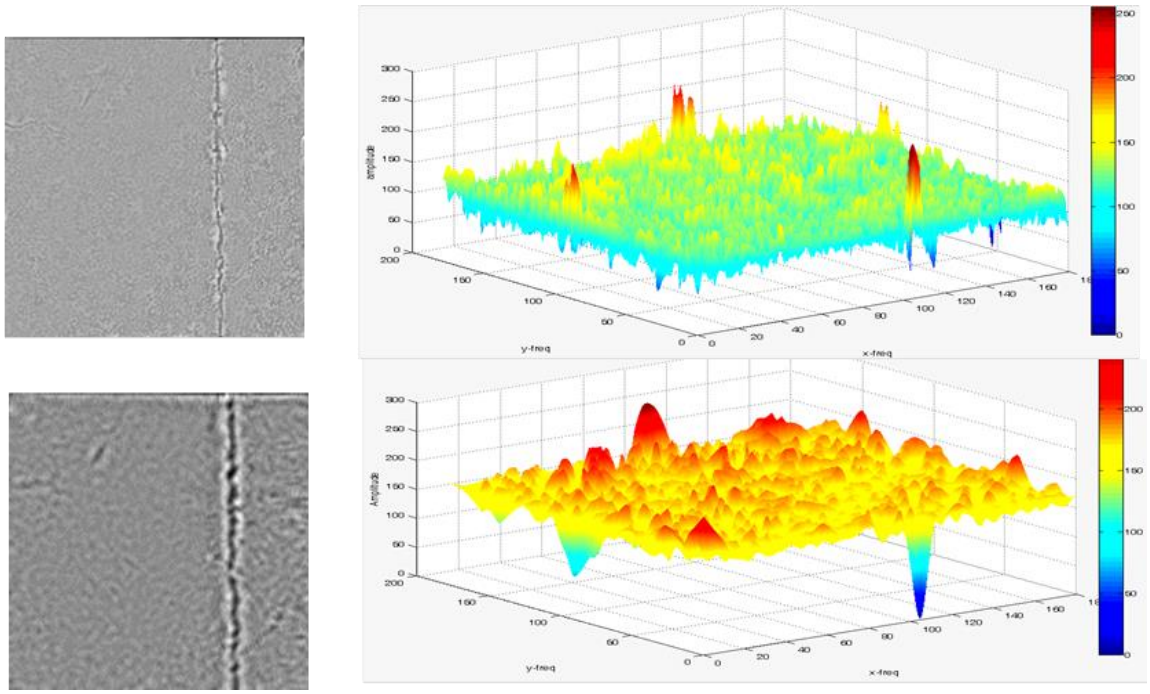


Figure 2.3: Modes containing Crack features

The pavement image is standardized by removing any variations in background illumination $I_{back}(x, y)$. The characterization of illumination in images is widely known in the literature. Illumination effects are usually partitioned into two types, shadowing and specular reflections. Since the shadowing darkens regions of an image, it creates low-valued regions while specular reflections create relatively high-valued regions. These are effectively the largest magnitude extrema in the images, but are also the most slowly changing (Bhagavatula and Savvides, M. 2007). Many solutions and improvements to image recognitions under illumination-variant conditions have been proposed (Belhumeur et al., 1997, Chen et al., 2002, Gross et al., 2002). Yet, the complexities and assumptions of idealities in many of these methods often limit their overall applicability, and hence are not ideal. A generic test of an ideal method for background standardization is one that preserves the crack widths, lengths and directions in

the original image. In the following experiment, we illustrate how BEMD provides a simple, yet efficient way of removing the backgrounds of pavement images acquired under different forms of illumination variation using low-cost cameras and laser devices.

2.2.1.1. Single Source Background Illumination Removal

For an image corrupted from single source illumination, the shadows and specular effects usually have similar intensities. This does not necessarily imply that the shadows are created from a single object; it could be from different objects, however, the intensity of the shadows they cast on the image is similar. Figure 2.4 and 6 shows examples of an image corrupted from a single and multiple sources. The single source clearly has a very simple bimodal histogram. This makes its characterizing or modeling more straightforward than from the multisource. As already explained, in the BEMD domain, each different IMF reveals the state of the image in varied scales. The highest-frequency components usually contain the noise and the fine edges in the image. The intermediate components extract the widths and generic shapes of linear patterns in the image, while the high-indexed IMFs collate effectively the slowly changing and largest magnitude extrema in the images. Hence, the low- and high-valued regions in the image will be collated in these IMFs. Figure 2.4b shows the background illumination component extracted by BEMD.

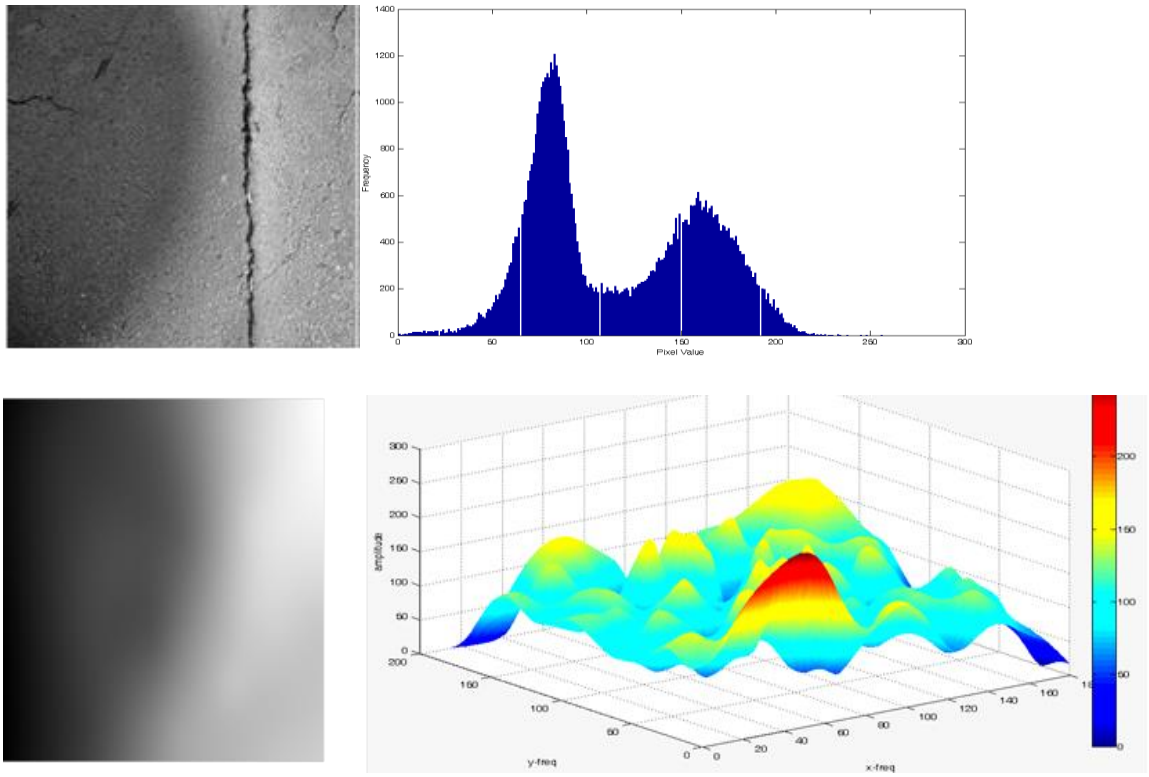


Figure 2.4: a). Pavement image corrupted by a single source illumination with its histogram. b). Illumination component extracted by BEMD.

From figure 2.5, the BEMD almost perfectly restores the corrupted image without compromising any other spatial features such as the width and shape of the cracks on the surface. Also, the implementation is very simple and straightforward; the restored image is obtained by a linear sum of IMF 1 and IMF 2. This is adaptive and data-dependent and can be applied for all images corrupted from a single source illumination variation.

Case 1: When image trend or illumination is separated into a single IMF.

$$BIMF_{restored} = \sum_{m=1}^{n-1} BIMF_m; \quad (4)$$

where $BIMF_{restored}$ is the restored image

Case 2: When image trend or illumination is separated into more than a single IMF.

$$BIMF_{restored} = \sum_{m=1}^{n-\sum_i residual} BIMF_m; \quad (5)$$

where $BIMF_{restored}$ is the restored image

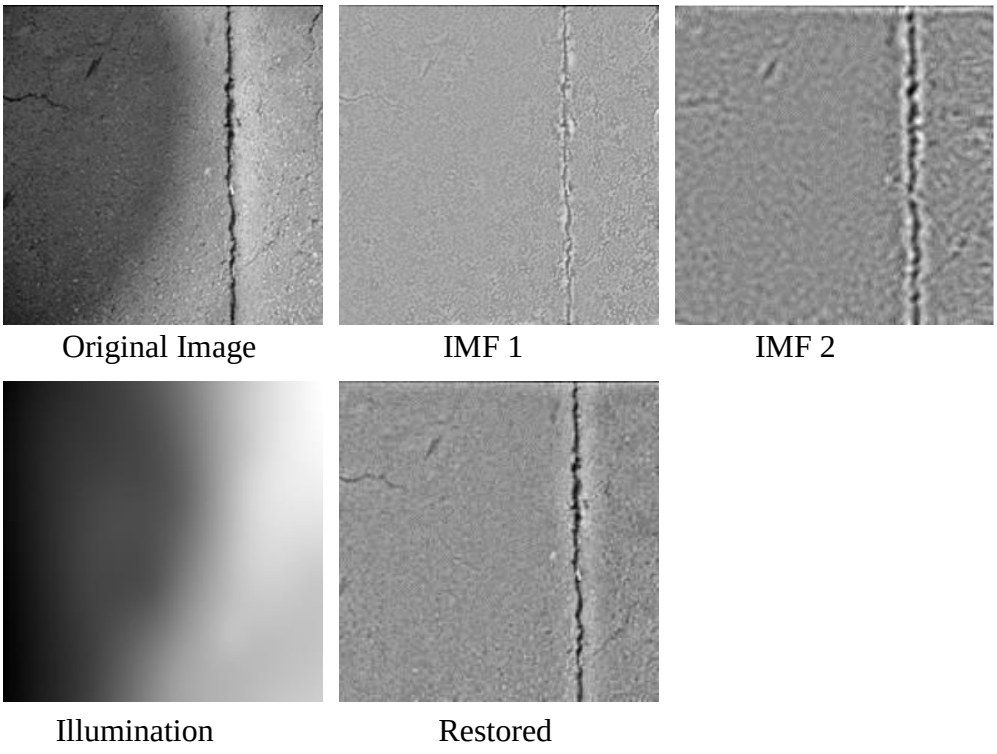


Figure 2.5: Original image, the IMFs, the illumination variation extracted and the restored image

2.2.1.2. Multisource Source Background Illumination

Images corrupted from multiple sources are more difficult to handle compared to those discussed in the previous section. In practice, a pavement image could also be corrupted from different sources creating sharp discontinuities between gray level pixels. In figure 2.6b, the histogram of the image corrupted from multisources is not completely bimodal. The shadowy regions appear

to have disparate intensity values. Note that the shades in the image have different intensities and hence don't have unique frequency characteristics. This is a challenge to the EMD because the different shades might not be collated into a single image. Initial decomposition results for the image are shown in figure 2.7. The restored image (figure 2.7f) still has some shadows on it. The difference in shadow intensities causes them to be collated in different IMFs, other than the residue. This will create difficulties in thresholding and segmentation (the shadow might appear as a crack). It is worth noting that this challenge defies our basic assumption that background variations are smooth variation high-low valued regions in the image.

b)

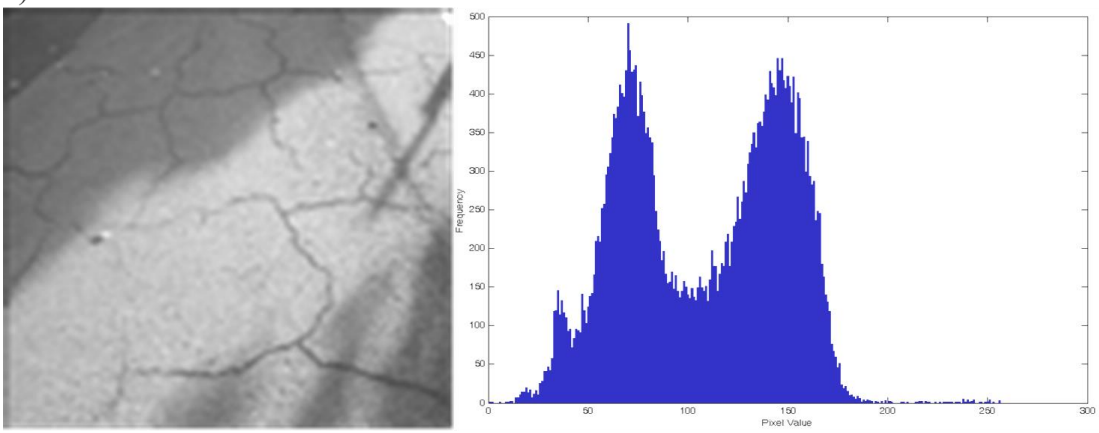
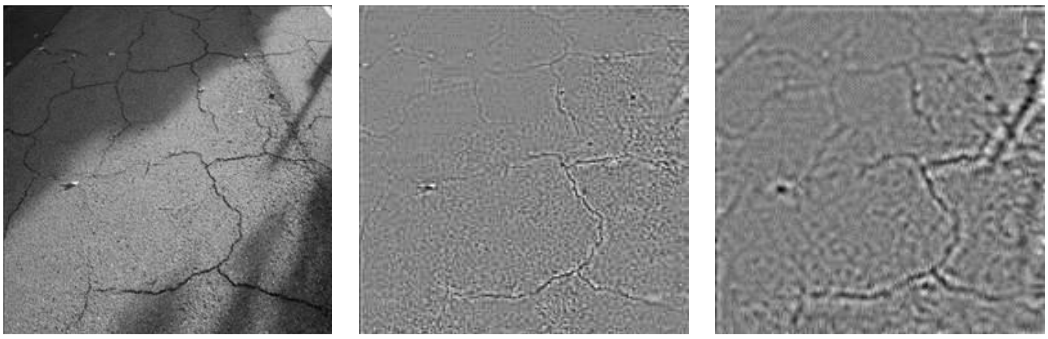
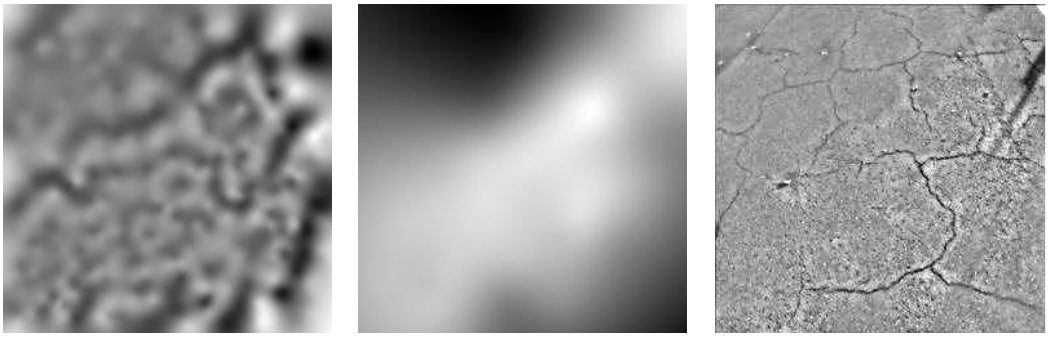


Figure 2.6: Pavement images corrupted by a) a single source illumination with its histogram and b) a multiple source illumination with its histogram.



a) Original b) IMF 1 c) IMF 2



d) IMF 3 e) Illumination f) Restored

Figure 2.7: Original image, the IMFs, the illumination variation extracted and the restored image.

To handle this challenge, we combine the EMD algorithm with a least mean square approximation approach to adaptively restore the corrupted image (see Table 4.1). The key objective is to approximate the BIMFs to a high-pass (contains only high frequency information, no shadows) of the original image. The result of applying the procedure outline in figure 2.9 is shown in figure 2.10.

Although weighting the low spatial BIMFs improved the results, the process is not perfectly automatic and adaptive. It requires working with extensive training data to estimate the optimal threshold value for the decomposition's stopping criterion. This defect was improved by restoring 17 pavement images with different levels of corruption. The histogram of the optimal threshold values used in each case is show in figure 2.8. From the plot, the threshold fluctuates;

however, it is predominantly between 0.6 and 0.8. The restored image also has a faint shadow on it. The reason is that because we are approximating the BIMFs to the high pass image, our resulting image will not be exactly the same as the high pass image. There may still be some light shadows present; however, these shadows are not a challenge as before, since they can easily be thresholded out.

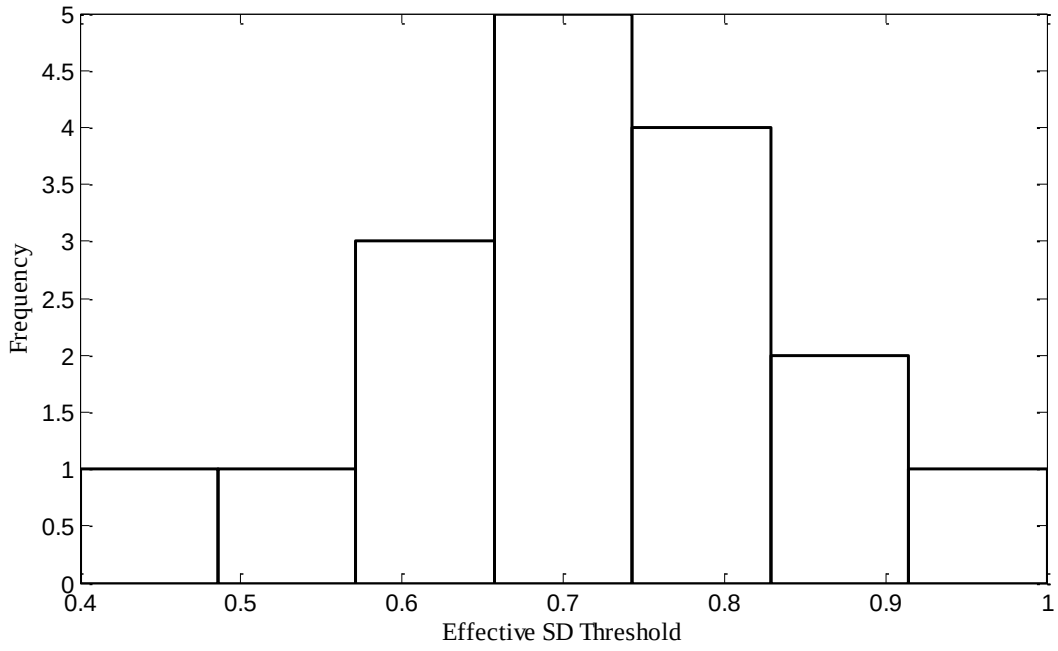


Figure 2.8: Histogram plot for the effective thresholds for SD criterion

Multisource Corrupted Image Normalization

Filtering:

1. Decompose the original image into a finite number of IMFs
2. High-pass filter the original image. This suppresses all shadows or low-pass information. Set this as the desired image, $\boldsymbol{d(x,y)}$.

Initialize a weight matrix? and determine the error of approximation:

3. Initialize $\boldsymbol{w_j(x,y) = 0; j = 1}$;
4. Compute the error of approximation

$$\boldsymbol{e(x,y) = d(x,y) - \sum_{i=2}^{n-1} w_j(x,y) * c_i(x,y)}$$

5. Compute pixel by pixel weights that minimizes the error of approximation $\boldsymbol{e(x,y)}$

$$\boldsymbol{w_{j+1}(x,y+1) = w_j(x,y) + \mu * e(x,y) * c_i(x,y)}$$

μ is a positive number controlling the convergence speed

6. Determine the restored image:

$$\boldsymbol{R(x,y) = \sum_{i=2}^{n-1} w_i(x,y) * c_i(x,y)}$$

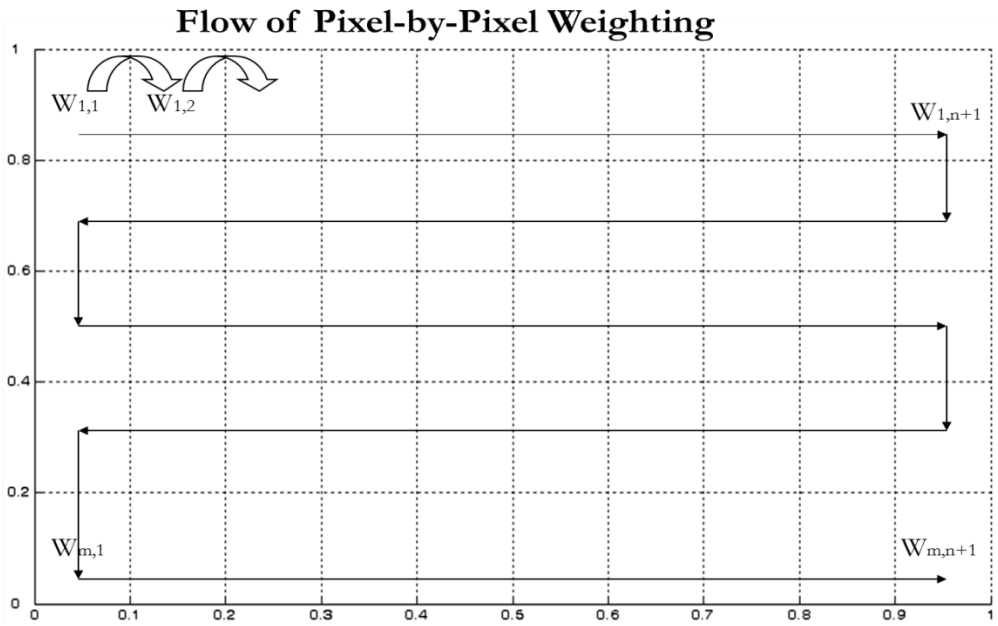
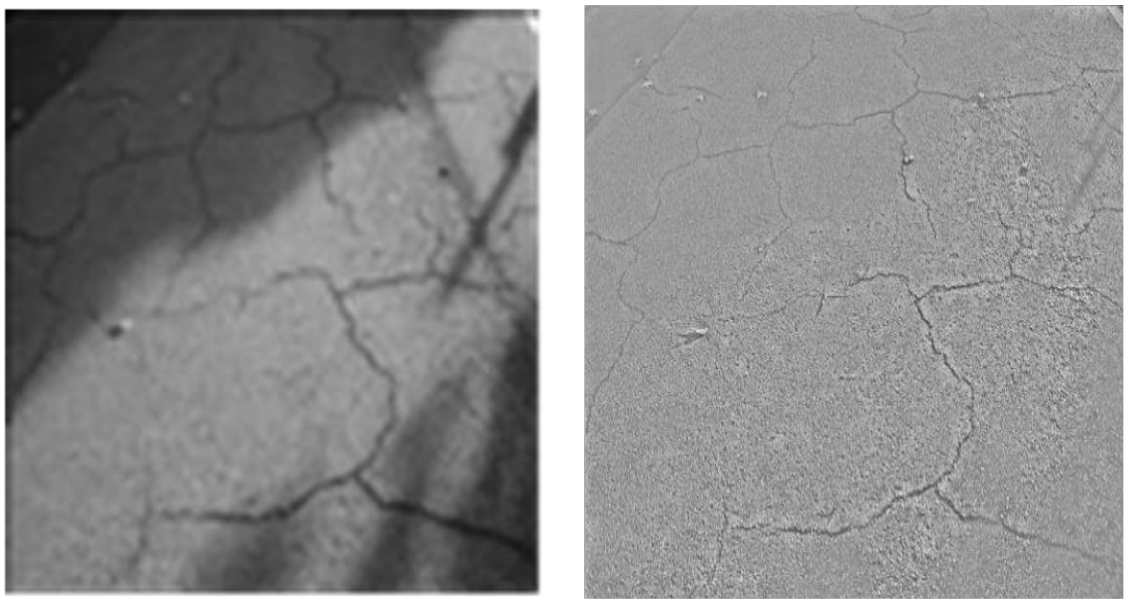


Figure 2.9: Flow of pixel-by-pixel weighting.



Original Restored

Figure 2.10: Original image, restored image.

2.2.2. **Image Denoising and Reconstruction for Crack Feature Extraction**

In this section we assume that $I_{noise}(x, y)$ is the image noise which is introduced mainly by pavement texture (not poor data acquisition) and other irregularities such as oil stains and other materials on the surface which might be falsely detected as cracks. We seek to smooth out as much as possible the irregularities introduced by the pavement's texture without erasing the cracks. Usually, image noise is characterized as a high-frequency and low- to medium-amplitude signal, however, the behavior of $I_{noise}(x, y)$might be different. Whereas texture information is predominant in the high frequency BIMFs, oil stains, paint markings and other objects will be collated in the medium to low scales. Therefore, pavement noise can be intermittent across the different image scale resolutions. So far, all BEMD denoising techniques can be generally grouped into two groups: Image Approximation (IA) and Selective or Complete Reconstruction. A brief description of these denoising techniques follows in the next section. It is important, however, to note that none of these techniques are able to deal with challenges posed by the pavement noise problem; hence, crack features extracted have spurious edge information.

2.2.2.1. Denoising and feature extraction by Image Approximation

This technique chooses an approximation of the original image (one BIMF) from an arbitrarily chosen decomposition level for feature extraction and segmentation. Usually, the low indexed BIMFs are used as the approximate image. This is because they have a very good characterization of the crack edges in the image. An arbitrary selection or combination of the first two BIMFs could extract all the edges in the image. However, these components are very sensitive to noise and as such could lead to false edges in the final image. Also, all the edges present in the composite image might not be needed for efficient extraction of crack patterns in

the pavement image. The widths of the edges are usually smaller than the actual width in the original image; this is because the low index BIMFs lose most of the low spatial information. Fidelity to the shape and width of cracks is very important for any pavement distress management system: An arbitrary selection, although it might be useful and simple in some special cases, generally will require laborious post-processing to remove spurious edges. Figure 2.11 illustrates some of the results obtained when this technique is employed.

2.2.2.2. Denoising and feature extraction by Complete or Arbitrary Selective Reconstruction

This technique aims at extracting a composite image while maintaining high fidelity to crack widths and shape in the original image. For complete reconstruction, the only pattern excluded from the original image is the image tendency or residue (last BIMF) which is usually the background illumination variation. Selective reconstruction combines BIMFs from arbitrary levels of decomposition. These procedures yield redundant information in the composite image due to the inclusion of detailed components (except the residue) that might not offer good discrimination ability. This could also lower the system's efficiency and would require more computation time. From Figure 2.11c and Figure 2.11e, the edge detector is unable to extract meaningful features from both criteria. Morphological techniques (Sun and Qiu 2007) were also used but the results were not appreciative. Similarly, edge detectors less sensitive to noise did not yield satisfactory results when applied. It should be noted that under simplistic cases, the techniques described above will work well. The key point here is that most traditional edge detectors are not necessarily good crack feature extractors. In this work, segmentation is used only to illustrate the accuracy of the crack feature extraction algorithm developed and to reduce

the storage requirements for each image. The next section introduces an improved reconstruction methodology to overcome some of the issues with traditional BIMF reconstruction and approximation. Later, an information mining technique that is capable of compromising between the two reconstruction techniques explained above will be introduced for extracting meaningful crack features from noisy images.

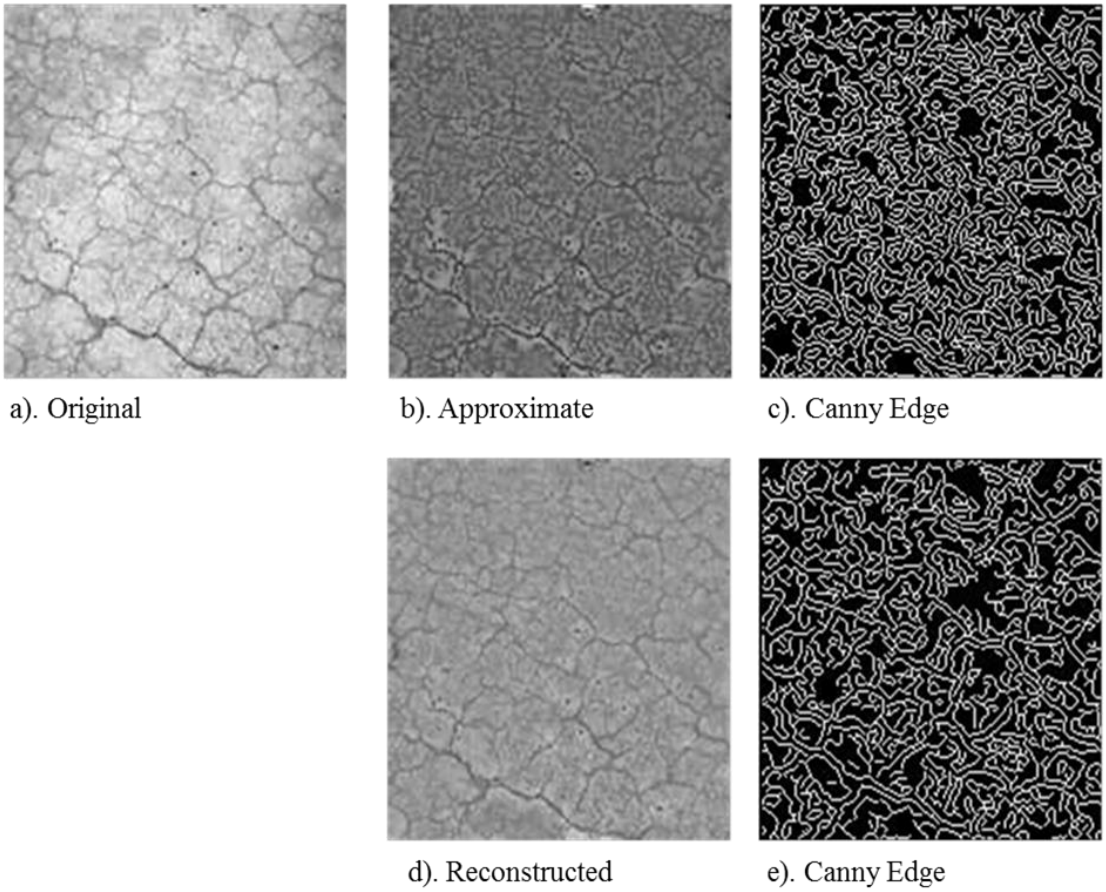


Figure 2.11: a).Original image. b). An approximation using the first BIMF. d). An image resulting from complete reconstruction. c) & e). A canny edge detector is used to detect the respective edges of the images.

2.2.2.3. Improved Multiscale Image Reconstruction and Approximation (IMIRA)

The key issue with the reconstruction and approximation methods described in the previous section is its inability to effectively separate noise from fine crack information. Figure 2.12 illustrates a proposed flow chart for improving the performance of the reconstruction algorithm. We suggest that before the image is fed into the BEMD algorithm, some preprocessing needs to be carried out using median filtering described in chapter 3. This will smoothen out some of the rough textures, reducing the noise in the first and second IMFs. Histogram equalization followed by the addition of white noise to the data aid the BEMD algorithm to generate IMFs with very narrow frequencies. Image approximation or reconstruction can then be used to reconstruct a composite image containing only salient crack information. If results are not satisfactory, the high frequency components of the composite image are decomposed again after white noise is added. The procedure is repeated until appreciable results are obtained.

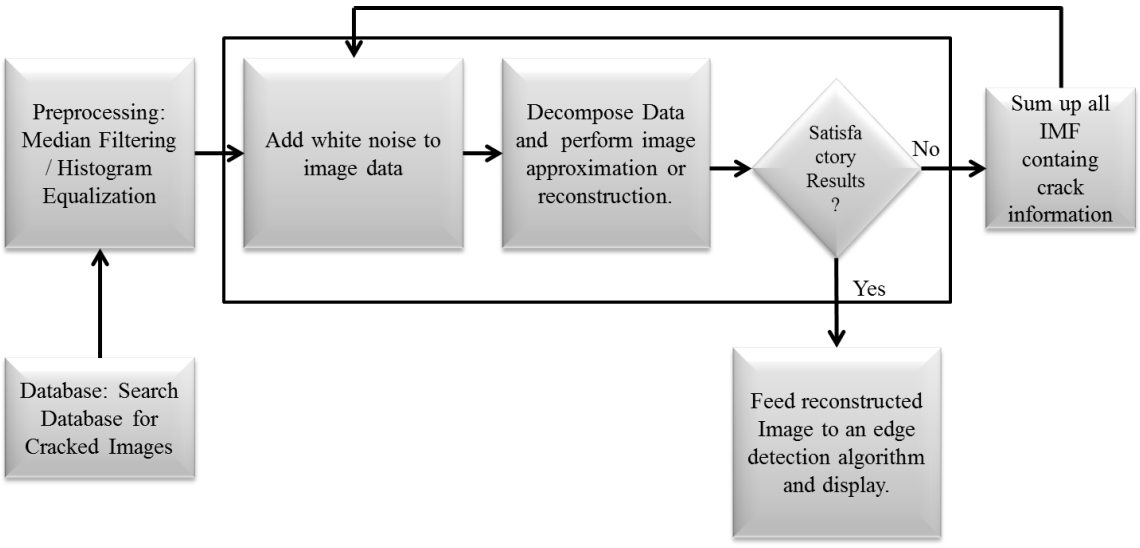


Figure 2.12: Flow diagram for Improved Multiscale Image Reconstruction and Approximation

The results of the proposed improved multiscale reconstruction and approximation is shown in figure 2.13. The discrimination between crack pixels and background objects is more pronounced and the canny edge detector does a better job than in the traditional methods. The algorithm, however, is unable to fully remove isolated pixels which consequently reduces the efficiency of the canny edge detector.

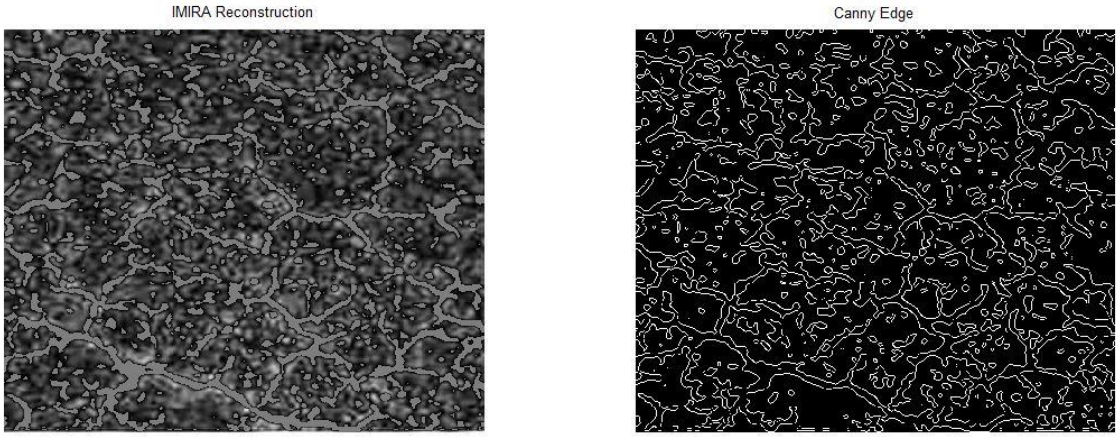


Figure 2.13: Results of proposed improved multiscale reconstruction and approximation

In spite of the improved results shown above, the process is not automatic and it can be time consuming since it may require a number of iterations to obtain desired results. In view of this, we introduce a more efficient reconstruction technique based on multiresolution information mining.

2.2.3. **Multiresolution Image Information Mining**

As explained earlier, in "complex" images, it may not be prudent to arbitrarily select BIMF for detection and classification of cracks on a pavement. The goal of multiresolution image mining is to reconstruct a composite image which records only salient information (crack features) from both fine and coarse scales of BIMFs while refraining from combining complementary, redundant information. So the term "multiresolution" stems from the fact that we are using different image scales or BIMFs whereas the information mining has to do with searching for crack features at the different resolutions.

To achieve this goal, we consider this issue as a separation problem. Each BIMF is considered as a superposition of a low-rank component and a sparse component as shown in equation 4. The sparse component captures the linear patterns in the foreground at each scale or BIMF, while the low-rank component corresponds to the background information.

$$C = L + S \tag{6}$$

Where $C$is a BIMF, $L$is the low-rank component and $S$is the sparse component.

Using the technique developed in Candes et al., 2009, we are able to recover the salient components (sparse components) efficiently without erasing crack information. This is achieved by using the Principal Component Pursuit (PCP) algorithm to solve the problem in equation 7.

$$\text{minimize} \|L\|_* + \lambda \|S\|_1 \tag{7}$$

$$\text{subject to } L + S = C \tag{8}$$

Where $\|L\|_*$ is the nuclear norm of the matrix L and $\|S\|_1$ is the l_1-norm of S.

For an exhaustive description of this technique, please refer to Candes et al., 2009. The general flow of the algorithm is as follows:

PCP Algorithm

Step 1: Initialize $S_0 = Y_0 = 0, \mu > 0$

Step 2: While not converged loop

Compute L_{k+1} via singular value decomposition

$L_{k+1} = D_\mu(M - S_k - \mu^{-1}Y_k)$.

Where $D_\mu(X) = US_\tau(\Sigma)V^*$ and $S_\tau(x) = \mathrm{sgn}(x)\max(abs(x) - \tau, 0)$,

$U\Sigma V^*$ is any singular value decomposition

Step 3: Compute $S_{k+1} = S_{\lambda\mu}(M - L_{k+1} + \mu^{-1}Y_k)$

Step 4: Compute $Y_{k+1} = Y_k + \mu(M - L_{k+1} - S_{k+1})$

end while

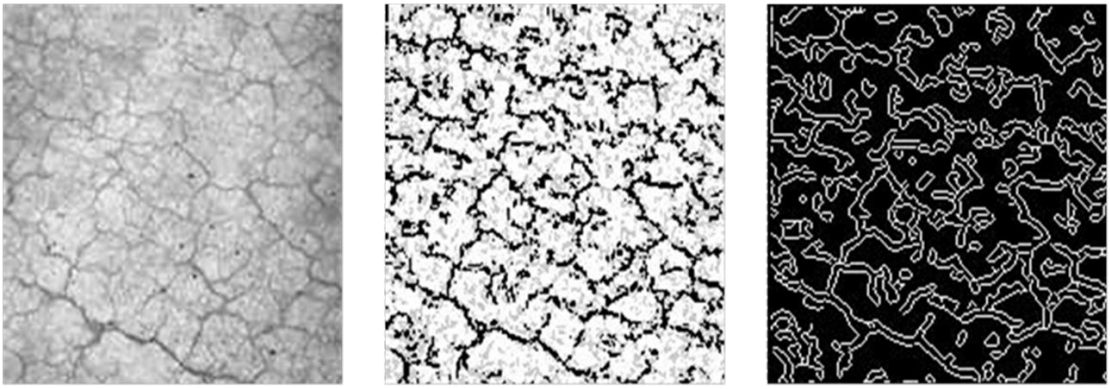
Step 5: output: L, S.

$\mu = M * N / 4\|M\|_1$ where M and N are the rows and columns of C. The algorithm is terminated when $\|M - L - S\|_F \le \delta\|M\|_F$ with δ =1e-7. The composite image is a sum of sparse representations at different scales.

$\vec{X} = \sum_{i=1}^{n} S_i$; where $\vec{X}$ is the composite image and S_i are the sparse representations at different resolutions.

The composite image in Figure 2.14b captures the salient information required for accurate and physically meaningful extraction of linear patterns in the pavement image. There are no spurious features in the composite image. Also, the widths of the cracks are well preserved. The segmented image is more physically meaningful than in Figure 2.11.

It should be noted that applying the PCP directly to the original image does not yield good results because of artifacts and background variations. It provides an improved result when there is minimum variation in the background. Hence, the PCP may not be efficient as a stand-alone methodology.



a). Original b). Composite c). Canny Edge

Figure 2.14: a) Original image; b) its approximation using the information mining technique; c) the segmented image by canny edge detector and the final edge after morphological closing.

2.3. Remarks

This study plays a vital role in pavement distress image analysis. The main focus of the chapter was to develop straightforward, adaptive and data driven procedures to extract salient

information from pavement distress images through a multiresolution information mining technique.

The EMD was used to decompose the distress image into components that accurately reflected the state of the image at different scales and resolutions in the spatial and frequency domains. A straightforward application of the EMD for image background standardization was demonstrated for single source illumination. For images corrupted from different sources, a weighted sum of the BIMFs was used to standardize the background of the image.

By combining the EMD algorithm and PCP, the authors achieve their ultimate aim of reconstructing a composite image that extracts only salient crack features from different resolutions while refraining from the inclusion of detailed components that may not provide discrimination ability. The fidelity of the composite image to the original image is very high in terms of the width and direction of cracks. The approach illustrated can be extended for analyzing images from different civil infrastructure systems.

REFERENCES

Abdel-Qader, I., Abudayyeh, O., Kelly, M.E. (2003). "Analysis of Edge Detection Techniques for Crack Identification in Bridges." *J. of Computing in Civil Engineerin*g, 17(3), pp. 255– 263.

Adu-Gyamfi, Y.O., Attoh-Okine, N.O., Ayenu-Prah, A.Y. (2010). "Critical Analysis of Different Hilbert Huang Algorithms for Pavement Profile Evaluation." *Journal of Computing in Civil Engineering*., 24 (6), pp. 514-524.

Ahmed, M.F.M., and Haas, C.T. (2010). "Potential of Low-Cost, Close-Range Photogrammetry Toward Unified Automatic Pavement Distress Surveying." Transportation Research Board Annual Meeting, CD-ROM.

Alzu'bi, S. and Amira, A. (2010) "3D Medical Volume Segmentation Using Hybrid Multiresolution Statistical Approaches," Advances in Artificial Intelligence Volume 2010, Article ID 520427.

Ayenu-Prah, A., and Attoh-Okine, N.O. (2008). "Evaluating Pavement Cracks with Bidimensional Empirical Mode Decomposition." *EURASIPJ. Adv. Signal Process.*, Volume 2008, Article ID 861701.

Belhumeur, P. N., Hespanha, J. P., & D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, pp. 711-720, 1997.

Bhagavatula, R., and Savvides, M. (2007). "Analyzing facial images using empirical mode decomposition for illumination artifact removal and improved face recognition." *IEEE InternationalConference on Acoustics, Speech and Signal Processing*., 1(15-20), pp. 501-508.

Candes, E., Li, X., Ma, Y., and Wright, J. (2009). "Robust principal component analysis?" http: // arxiv. org/ abs/ 0912. 3599.

Cheng, H. D., and Miyojim, M. **(**1998**). "**Automatic pavement distress detection system." *Information Sciences.*,108(1-4), pp. 219-240.

Chen, T., Hsu, Y. J., Liu, X., & Zhang, W., "Principle Component Analysis and its Variants for Biometrics," presented at Image Processing. 2002. Proceedings. 2002 International Conference on, 2002.

Fujita, Y., Mitani, Y., Hamamoto, Y. (2009). "A Robust Method for Automatically Detecting Cracks on Noisy Concrete Surfaces." *IEA/AIE 2009.*, LNAI 5579, pp. 76–85.

Gross, R., Matthews, I., & Baker, S., "Eigen Light-Fields and Face Recognition Across Pose," Proc. of FG02, 2002.

Hutchinson, T.C., and Chen, Z. (2006). "Improved image analysis for evaluating concrete damage." *J. of Computing in Civil Engineering.*, 20(3), pp. 210–216.

Huang, N.E., Shen, Z., Long, S.R., Wu, M.C., Shih, H.H., Zheng, Q., Yen, N.C., Tung C.C., and Liu, H. (1998). "The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis." *Proc. Royal Society London, Series A*, 454 (1971), pp. 903–995.

Kim, J. Y. (2009). "Development of New Automated Crack Measurement Algorithm using Laser Images of Pavement Surface." *PHD Dissertation*, 2009., University of Iowa.

Klionski, D.M., Oreshko, N.I., Geppener, V.V., and Vasiljev, A.V. (2008). "Applications of Empirical Mode Decomposition for Processing Nonstationary Signals." *Pattern Recognition and Image Analysis*, Vol.18 (3), pp. 390-399.

Lee, H.D. (2005). "Development of a Manual Crack Quantification and Automated Crack Measurement System." Project TR-457, Public Center, Civil and Environmental Engineering, University of Iowa.

Liang, Y., Salari, E. (2009). "Beamlet Transform Based Technique for Pavement Image Processing and Classification." *IEEE International Conference on Electro/Information Technology*,pp. 141 – 145.

Sun, B. -C., and Qiu, Y. -J. (2007). "Automatic Identification of Pavement Cracks using Mathematic Morphology," Proc., Int. Conf. on Transportation Engineering 2007., ASCE, Reston, Va., pp. 1783–1788.

Wang, K. C. P. (2004). "Challenges and Feasibility for Comprehensive Automated Survey of Pavement Conditions," Proc., Int. Conf. on Applications of Advanced Technologies in Transportation Engineering., ASCE, Reston, Va., pp. 531–536.

Wang, K.C.P., and Gong, W. (2007). "Automated Real-Time Pavement Crack Detection and Classification." Final Report for Highway IDEA Project 111.

Weng, B. and Barner, K. E. (2007). "Optimal Signal Reconstruction Using the Empirical Mode Decomposition." *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing* (ICASSP'07)., Volume III, pp. 1501-1504.

Xianglong, L., Qingquan, L. (2006). "An algorithm to pavement cracking detection based on multi-scale space." *Conference on Remotely Sensed Data and Information*, SPIE vol.6419.

CHAPTER 3

VISION SYSTEM FOR AUTOMATED PAVEMENT DISTRESS ANALYSIS

3.0. General Background

Vision is a startling but difficult feature of natural intelligence. It involves the use of sensory and brain parts of biological organisms. This makes it very difficult to explicitly mimic its functions using computers. Computer vision seeks to develop algorithms that replicate the capabilities of the human brain - inferring properties of the external world.It endows computers with information-processing capabilities which enable them to model and automate the process of visual recognition in a way comparable to those of biological organisms. The concept has a wide variety of applications in medical image analysis, mobile robot navigations, image retrieval in digital libraries, etc.

Previous chapters were focused on developing image processing algorithms whose main goal was to manipulate the image data and return another image as the output. In addition, a few concepts of vision such us detection or recognition and reconstruction were introduced. This chapter will focus on developing a robust and complete vision system capable of generating consistent and meaningful symbolic data (one which outputs decisions rather than a transformed image) from acquired pavement distress images. With computer vision, we are able to construct explicit and meaningful description of cracks, which is a prerequisite for distress quantification and classification.

The vision system will consist of two main components: first, an image retrieval component for separating cracked images from non-cracked ones. The goal here is to pass on only cracked images for edge detection. This will speed up the system considerably since non-cracked images will not be processed. The second component is for crack feature detection. Here, we propose a model-based approach known as active contour models. Edge detection systems are purely local; they have no concept of an edge (e.g. that edges are continuous; they tend to be smooth almost everywhere). They just detect points where the gradient is high, whether they are edges or noise (Waite and Welsh 1990). Built into the active contour models are various properties such as continuity and smoothness which are usually associated with both edge and human visual systems. Hence, they are more efficient for processing typical edge segments and extracting accurate geometric parameters of the crack boundary.

3.1.System Design

The design is expected to achieve three primary goals: First, the system should be able to take inputs from different camera types (laser, digital, etc.) and automatically identify the most efficient algorithm for processing their respective images. Secondly, the system uses adaptive and data-driven algorithms for feature extraction; this implies that we will also examine a parallel implementation of these algorithms in order to facilitate the speed of the system. The last goal is to provide the user with a database and output results that are easily understandable and helpful for project or network level road maintenance and repair programs. So the system will have three (3) main components:

- Image Acquisition Component
- Image Retrieval and Vision Component
- Output Analysis and Visualization Component.

Acquisition Devices: The system can process images from different cameras, mostly lasers or digital. The user has the option to choose either real-time or offline processing. For real-time processing, computers must be multi-core (at least eight cores) and have large memory and disk space. Acquisition vehicles cannot exceed 45 mph since that will affect the quality of images acquired. In view of current advancements in 3D crack depth perception, the geometry of cameras is designed to facilitate stereo vision.

3.1.1. Vision Component

This is the heart of the system. Figure 3.1 shows the key parts of the vision component. It involves four basic steps:

- Crack Image Retrieval System
- Pre-processing stage (enhancements and de-noising)
- Crack Feature Detection System

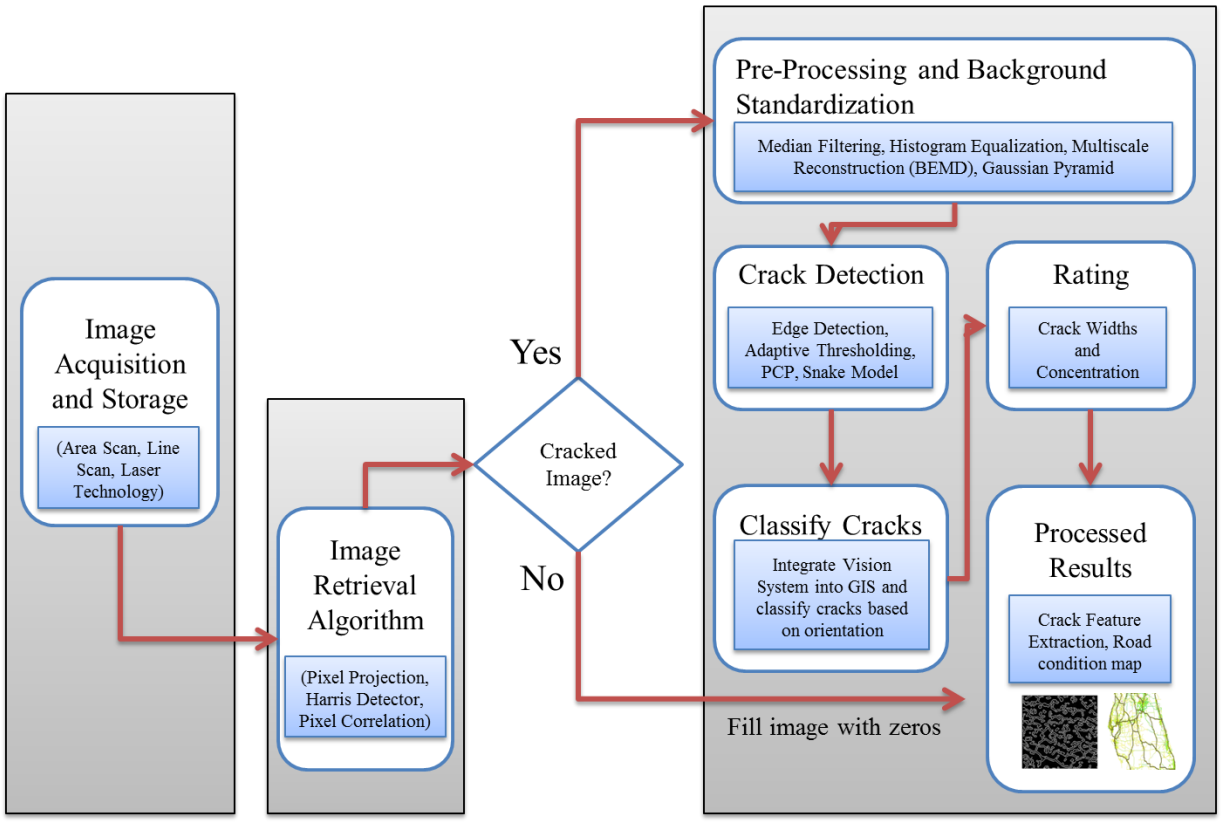


Figure 3.1: Complete Vision System Design

3.1.2. Crack Image Retrieval System

Image retrieval is a process for separating "cracked" image pixels from non-cracked images in the database. This process is an important step that improves the overall system's speed. Here, image retrieval algorithms will run through the database and inhibit all non-cracked images (see figure 3.2) while passing on cracked ones to the vision system for denoising, segmentation, feature extraction and classification. Three different procedures, a Harris detector-based method (Harris and Stephens, 1988), a row-column projection method and a correlation-based method were developed for crack interest point identification and isolation of cracked images from non-cracked ones. Their relative efficiency is also evaluated in this section.

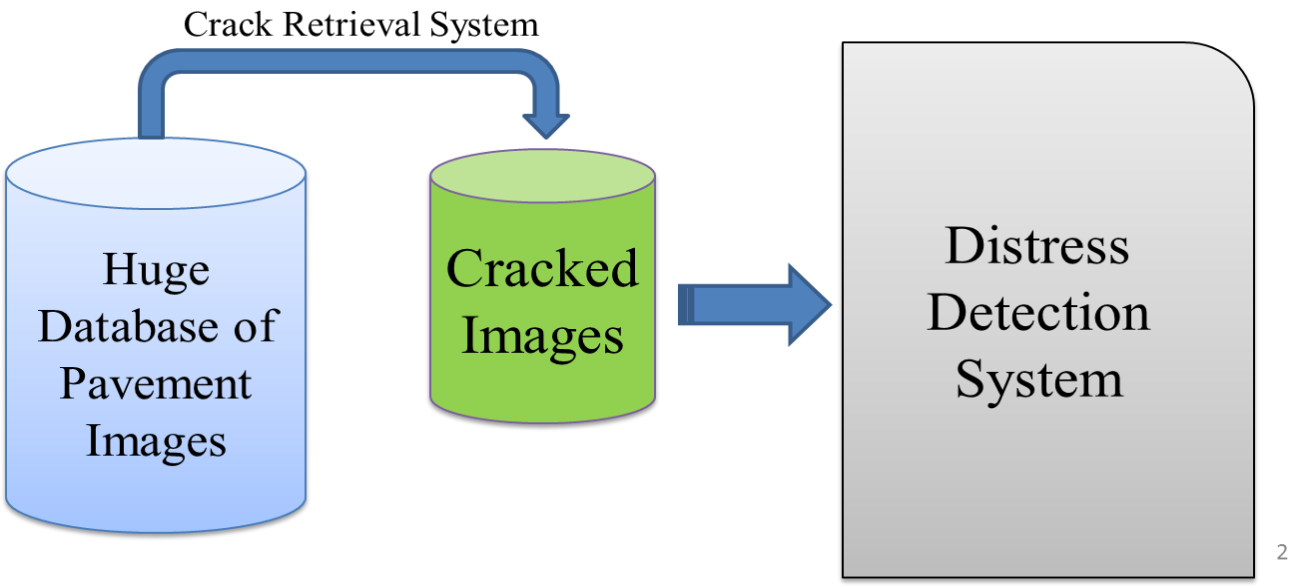


Figure 3.2: Illustrates the goal of an image retrieval system.

3.1.2.1.*Harris Detector Method:*

The Harris detection method uses descriptors that are rotation and partially scale invariant. It computes the change of intensity for a shift(u, v). That is:

$$E(u,v) = \sum_{x,y} w(x,y)[I(x+u, y+v) - I(x,y)]^2$$

Where $w(x,y)$ is a window function (e.g. Gaussian), $I(x+u, y+v)$ is the shifted intensity and $I(x,y)$ is the original intensity. For small shift(u, v), we have a bilinear approximation:

$$E(u,v) = [u \quad v] M \begin{bmatrix} u \\ v \end{bmatrix}$$

Where M is a 2 X 2 matrix computed from image derivatives

$$M = \sum_{x,y} w(x,y) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}$$

The response for each pixel is given by: $R = \det(M) - k(trace(M))^2$;

K is an empirical constant ranging from $0.04 - 0.06$. Table 3.1 describes the complete implementation of the algorithm.

Harris Interest Point Detector Algorithm

1. Compute image derivatives in $\boldsymbol{x}$ and $\boldsymbol{y}$ directions:
   a. $[\boldsymbol{I_x} \quad \boldsymbol{I_y}] = \boldsymbol{ImagDerivative(Image,' x-direction,' y-direction')}$
2. Convolve image derivatives with a Gaussian function to construct matrix M. Using a 1-D function for the convolution process speeds up the system.
   a. $\boldsymbol{I_x^2 = 1D\_gauss\_function(I_x^2,' x',' y')}$; **Similarly,** $\boldsymbol{I_y^2}$ **and** $\boldsymbol{I_x I_y}$
3. Compute Harris response for each pixel:
   a. $\boldsymbol{R = (I_x^2 .* I_y^2 - (I_x I_y).^2) - k * (I_x^2 + I_y^2).^2}$
4. Find points with large corner response function $\boldsymbol{R}$ $(\boldsymbol{R > threshold})$
5. Take the points of local maxima of R

The efficiency of this algorithm is dependent on the smoothing function used. However, materials such as loose rocks could be a big challenge, resulting in false detection of crack interest points. The results of using Harris detection for crack interest point locations are shown in figure 3.3. Shadows may not be an issue as long as the image is well smoothed. Clearly,

pavement texture remains an issue. Overall, the algorithm gives a good delineation of the cracks in the images.

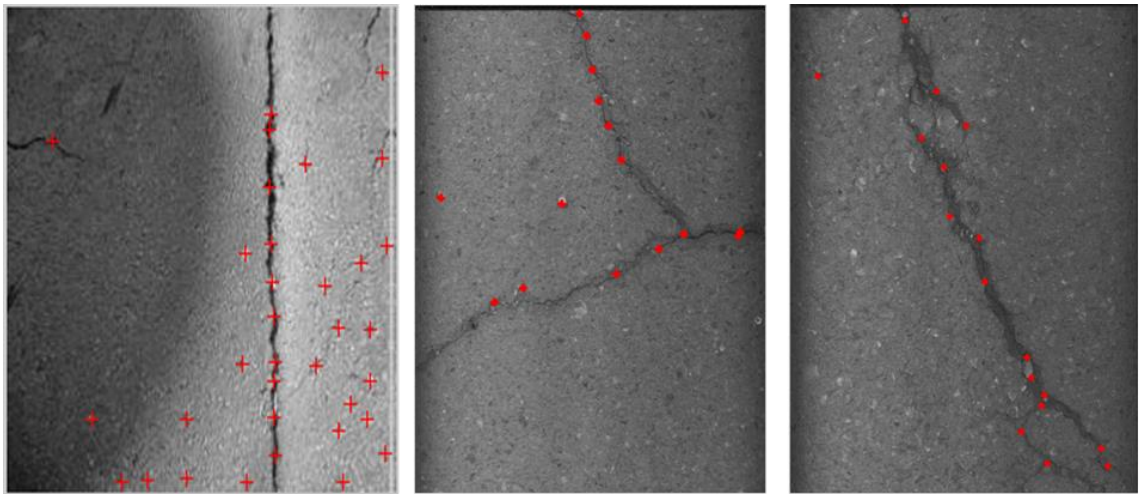
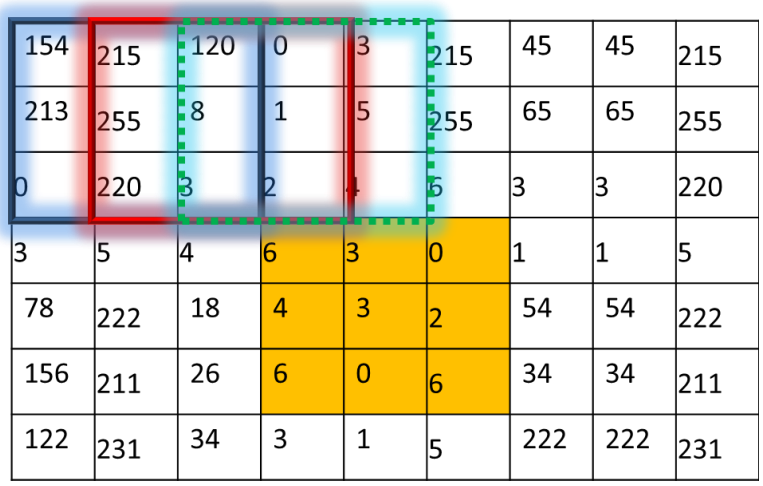


Figure 3.3: Interest point detection using Harris Detector

*3.1.2.2.Pixel Correlation Method**:***

This procedure assumes that the grey level of a point of interest must be low correlated to the grey levels of its neighbors.Sylvie and Moliard, 2011developed a correlation method that estimates the cost between a pixel and its neighborhood pixels. This technique could slow the system due to redundant pixel-to-pixel correlations and it also fails when an interest point covers a large image area. The method developed in this work uses a window matching technique described below. The algorithm is very effective; however, it is computationally expensive.



154	215	120	0	3	215	45	45	215
213	255	8	1	5	255	65	65	255
0	220	3	2	4	6	3	3	220
3	5	4	6	3	0	1	1	5
78	222	18	4	3	2	54	54	222
156	211	26	6	0	6	34	34	211
122	231	34	3	1	5	222	222	231

Figure 3.4: Correlation between a region of interest and its neighborhood

Pixel Correlation Algorithm

1. Select a pre-defined neighborhood window (say 3 by 3). This region could be any of the blue, red and green regions indicated in figure 3.4 above.
2. Compute correlation between center pixel $\boldsymbol{f_{ij}}$ (indicated by the yellow region in figure 3.4) and a neighborhood pixel $\boldsymbol{f_{kl}}$

$$\boldsymbol{R_{ij} = CC(f_{ij}, f_{kl})}$$

3. Final measure is $\mathbf{max}[\boldsymbol{CC(f_{ij}, f_{kl})}] - \boldsymbol{min}[\boldsymbol{CC(f_{ij}, f_{k'l'})}]$

From figure 3.5 below, the correlation method is able to locate interest points with few distractions from texture information. Compared to Harris, the number of false interest point detections is lower.

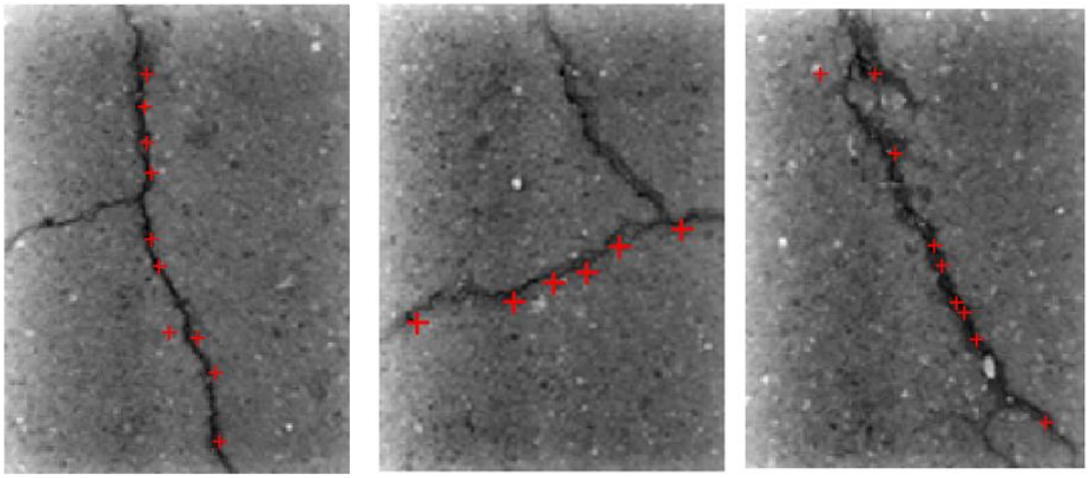


Figure 3.5: Interest point detection using Pixel Correlation Detector

3.1.2.3.Pixel Projection:

The projection function is summation. The direction of projection is vertical and horizontal. Alligator and block cracks may therefore be difficult to pick up. Cracks should lie in the direction of the projection (see figure 3.6).

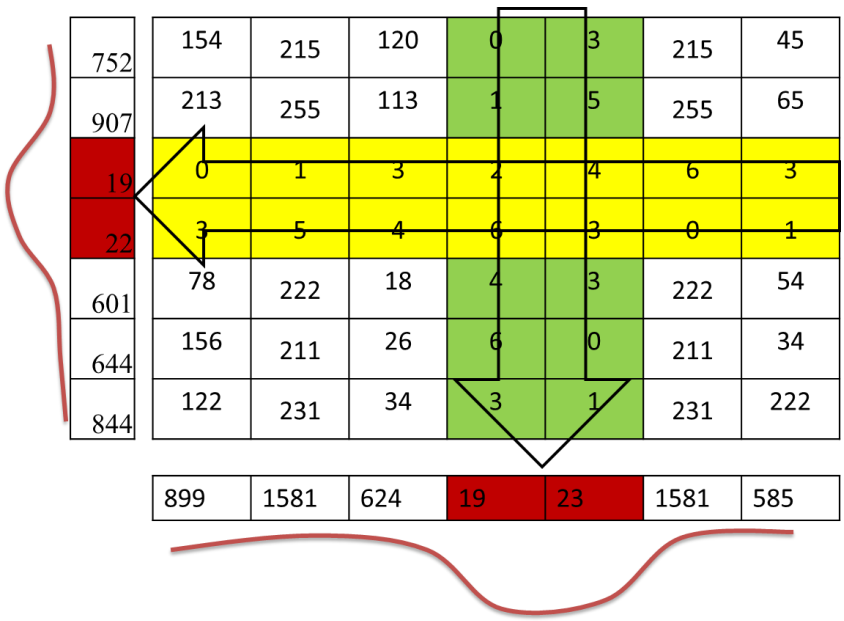


Figure 3.6: Illustrates pixel value projection in image rows and columns.

Pixel Projection Algorithm

1. Project image pixels $\boldsymbol{f(i,j)}$ column-wise and row-wise onto a horizontal and a vertical accumulator

$$\boldsymbol{H_{AC} = P[f(:,j)], and\ V_{AC} = P[f(i,:)];}$$

where P is the sum operator in the case and j is the column index and i is the row index

2. Take a derivative of $\mathbf{H_{AC}}$ $\boldsymbol{and}$ $\mathbf{V_{AC}}$ to obtain $\mathbf{H'_{AC}}$ $\boldsymbol{and}$ $\mathbf{V'_{AC}}$. Points of interest should be marked
3. Supposing that a crack POI has the highest local dissimilarity use $\boldsymbol{Max - k * Min}$ to estimate the threshold T, for finding the points of interests.

$$\boldsymbol{POI = f(i,j) > \max(H'_{AC} < 0) - k * \min(H'_{AC} < 0) = T}$$

If the image $\boldsymbol{f(i,j)}$, has sufficient POIs greater than the threshold, T, then the image is cracked; else it is a non-cracked image.

From figure 3.7 below, the projective method is easily distracted especially when the cracks in the image are not unidirectional. It is also very sensitive to texture and stones on the image.

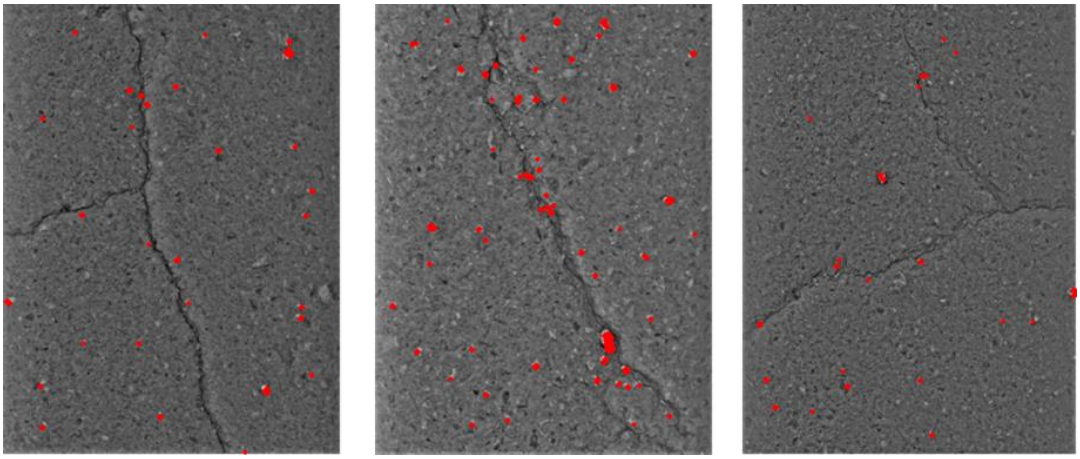


Figure 37: Interest point detection using Pixel Projection Detector.

3.1.2.4.Algorithm Efficiency

The efficiency of the algorithms described above is tested against three different image databases. The first and second database consists of only transverse and longitudinal cracked images respectively. The third database is a mixture of transverse, longitudinal, block and alligator cracked images. The test for algorithm efficiency is based on the equation below:

$$Efficiency = \frac{2TP}{FN + TP + P}$$

TP represents true positives, FN are the false negatives while P is the total number of positives.

An efficiency rate of 100% implies that there are no false negatives-FN (i.e. algorithm selects image as non-cracked when it is cracked) or the number of true positives (i.e. algorithm selects an image as cracked when it is cracked) corresponds to cracked pixels in the database.

From the results shown in Table 3.1, the Harris detector out-performs correlation and the projection based method. This is probably because it is sensitive to direction information. The

performance of correlation method is close to that of the Harris detector, however, it is computationally expensive.

Table 3.1: Evaluating the efficiency of image retrieval algorithms.

$Efficiency/Sensitivity = \frac{2TP}{FN+TP+P}$			
Algorithm/Database	Number of Images: 15 Crack Type: Transverse	Number of Images: 15 Crack Type: Longitudinal	Number of Images: 22 Crack Type: Transverse, Longitudinal and Pattern Cracks.
Harris	100%	100%	86.4%
Correlation	100%	93.3%	86.4%
Projection	100%	100%	77.3%
TP-True Positives, FN-False Negatives, P-Positives			

3.1.3. **Image Enhancement and Preprocessing**

In previous chapters, we developed several enhancement and denoising algorithms for preparing pavement image for crack detection and recognition. A new algorithm for preprocessing distress images is developed in this chapter due to its simplicity, efficiency and speed. This algorithm is used as the default for preprocessing in the complete system, except if the user selects a different algorithm suggested in chapters 3 and 4. The algorithm has two principal phases:

- Background normalization or standardization, and
- Image Smoothening

3.1.3.1.Gaussian Pyramid based background standardization

The algorithm decomposes the original image into different scales by sub-sampling and convolution with a pre-defined Gaussian function.

- ***Low-pass Filtering*:** the original image is smoothed by convolving it with Gaussian function illustrated in the figure 3.8 below. This is the first step of the multiscale background standardization approach. This step can be designated as: $g_i = (G_\sigma * g_{i-1})$

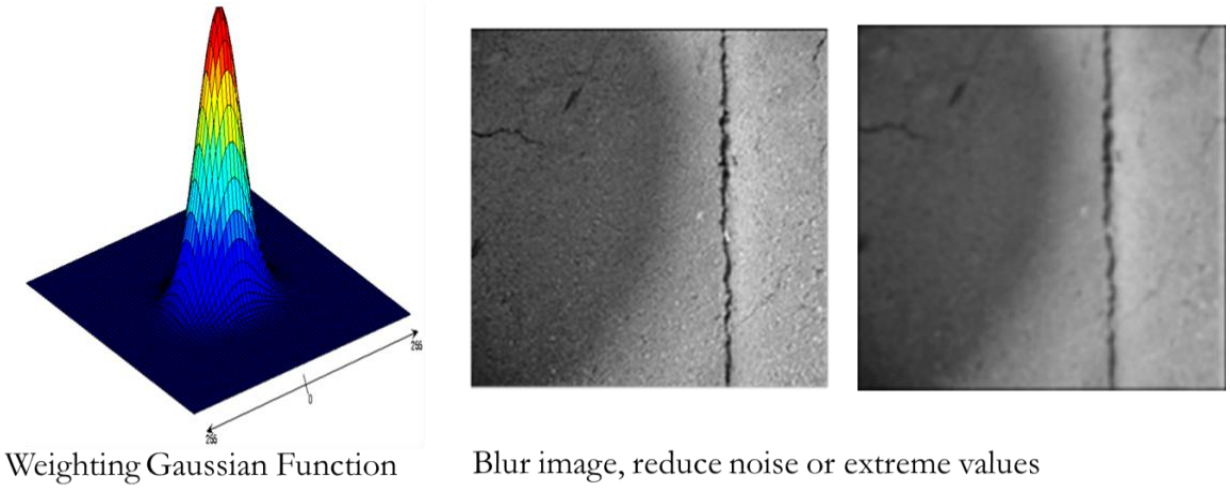


Figure 3.8: Illustrates low pass filter design and blurred images.

- ***Sub or Down Sampling*:** Involves cutting the width and height of the image into half at each iteration. It can be designated as $g_i = s^{\downarrow}(G_\sigma * g_{i-1})$. Where $s^{\downarrow}$ is the down sampling operation. An example is shown in figure 3.9 below.

154	215	120	0	3
213	255	113	1	5
0	1	3	2	4
3	5	4	6	3
78	222	18	4	3

154	120	3
0	3	4
78	18	3

Figure 3.9: Image down-sampling example.

Pyramid Levels: Iterative down-sampling generates a pyramid of images as shown in the figure below. The general trend in the image stays persistent while crack information is removed. Subsequent level g_{i+1} is obtained by convolving the preceding level with the Gaussian function followed by sub-sampling that is: $\boldsymbol{g_{i+1} = s^{\downarrow}(G_{\sigma} * g_i)}$**.** Figure 3.10 shows three pyramid levels generated from a distress image.

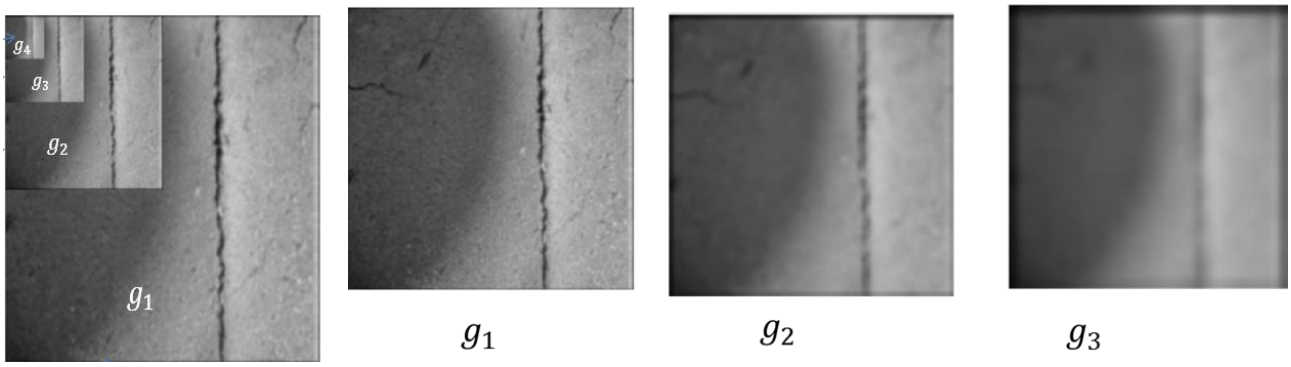


Figure 3.10: Left-pyramid levels after iterative sub-sampling. G1, G2 and G3 – images resized to original image after interpolation.

Image Restoration: The image is restored by subtracting the final level from the original to obtain the standardized image. From the figure below, the Gaussian pyramid algorithm almost perfectly restores the corrupted image without compromising any other spatial features such as the width and shape of the cracks on the surface. Thresholding at this stage is simple from the transverse profile shown in figure 3.11.

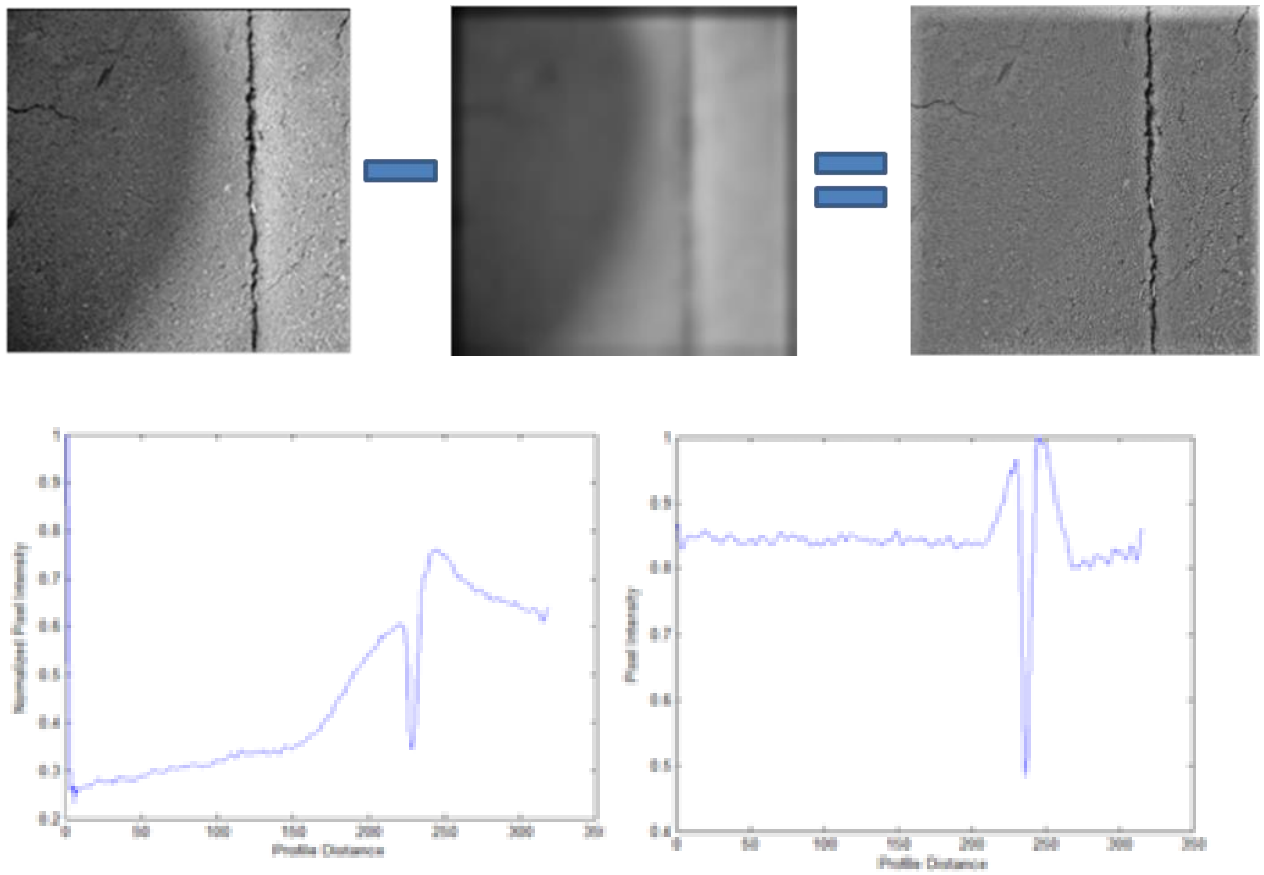


Figure 3.11: Top – image restoration using Gaussian pyramid. Bottom-transverse profile original (left) and restored (right) image.

Also, the implementation of the algorithm is very fast because of sub-sampling. However, intermediate levels are not useful. Since our ultimate goal is to standardize the image, and only the n^{th} level is useful, intermediate levels will not be needed. The algorithm steps are summarized in the table below.

Gaussian Pyramid Algorithm

1. Initialize the zero level of the Gaussian pyramid with an image array $\boldsymbol{g_0}$ containing C columns and R rows of pixels.
2. Low-pass filter the original image $\boldsymbol{g_0}$ and sub sample to obtain $\boldsymbol{g_1}$.

$$\boldsymbol{g_i = s^{\downarrow}(G_\sigma * g_{i-1})}$$

$\boldsymbol{G_\sigma}$is a 1D Gaussian filter used to convolve the image in both direction.

3. Subtract the $\boldsymbol{n^{th}}$ decomposition level from the original image to obtain the restored.

$$\boldsymbol{g_{restored} = g_0 - g_n}$$

In the preceding chapter, another multiscale technique called BEMD was used for background standardization. Below, we compare the strengths and weaknesses of both approaches. From figure 12, intermediate modes from BEMD are useful for crack detection. However, the Gaussian pyramid is five times faster than BEMD (see figure 3.13). On restoration quality, the Gaussian pyramid looks artificial as compared to BEMD.

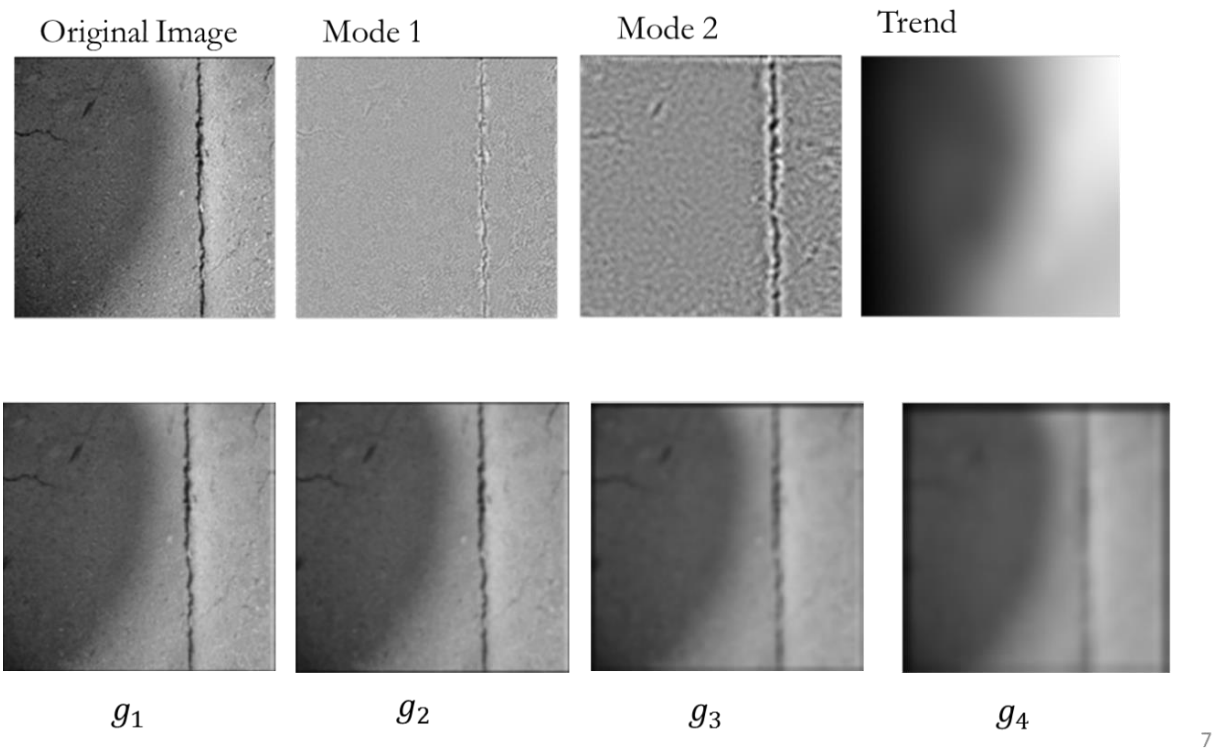


Figure 3.12: Top: BEMD decomposition levels. Bottom: Gaussian pyramid decomposition levels.

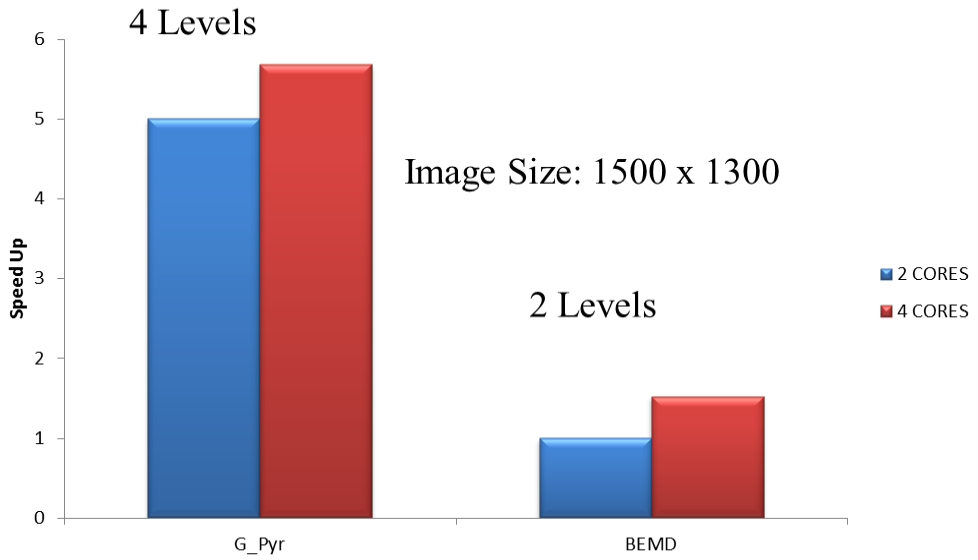


Figure 3.13: Computational time comparison.

3.1.4. **Crack Feature Extraction using Active Contour Models**

Traditional pavement detection systems (Lee 2005, Wang 2007, and Kim 2009) extract crack features by the use of edge detectors and thresholding algorithms, which generally work by setting the grey value of each pixel in the image to a value that is dependent on the magnitude of the gradient of the grey level at the corresponding point in the original image. The processing from this class of systems is purely local; they have no concept of an edge (e.g. that edges are continuous, and tend to be smooth almost everywhere) and just detects points where the gradient is high, whether they are edges or noise (Waite and Welsh 1990). Therefore such systems may fail in difficult conditions (rough textures, oil stains, etc.), creating spurious and discontinuous edges.

The concept of an edge is far more than the presence of a high gradient at a particular location; it highly depends on the spatial distribution of these high and low gradient points across the image. An edge detector, however, does not explicitly possess the capability of identifying the distribution of the gradients. In this section, instead of relying on tradition edge detection systems, we resort to investigating a model-based technique to detect the exact location of the crack, processing its edge segments and extracting accurate geometric parameters of the crack boundary.

Active contour models, also called snakes, appear to show promise in this direction. Snakes are energy-minimizing deformable splines, influenced by constraint and image forces that pull them toward object contours or boundaries. They are used to represent object boundaries or salient image features. Built into the snake models are various properties such as continuity and smoothness, which are usually associated with both edge and human visual systems. First introduced by Kass et al., the technique has gained much popularity since then. The snake can be

thought of as an elastic band of arbitrary shape, represented by a chain of points that wiggles in the image toward points of high image gradient. So the edge pixels must "pull" the snake points. The stronger the edge, the stronger its pull on the snake points.

Active contours are broadly classified into two main groups: parametric active contour models and geometric active contour models. Parametric models represent active contours explicitly as parameterized curves. In geometric models, however, they are represented as level sets of a two-dimensional function that evolves in a Eulerian framework. It is able to break or merge naturally during evolution. This helps it to handle topological changes very well. In the following sections, parametric contour models are used to explain the general properties and formulation of the snake algorithm. This is followed by the application of geometric models to different types of pavement distress images.

3.1.4.1.Snake Properties and formulation

Let us represent a snake by a parametric curve, $v(s)$

$$v(s) = [x(s), y(s)]; where\ 0 \leq s \leq 1$$

By this definition, the curve, $v(s)$ could be closed or open as shown below in figure 3.14

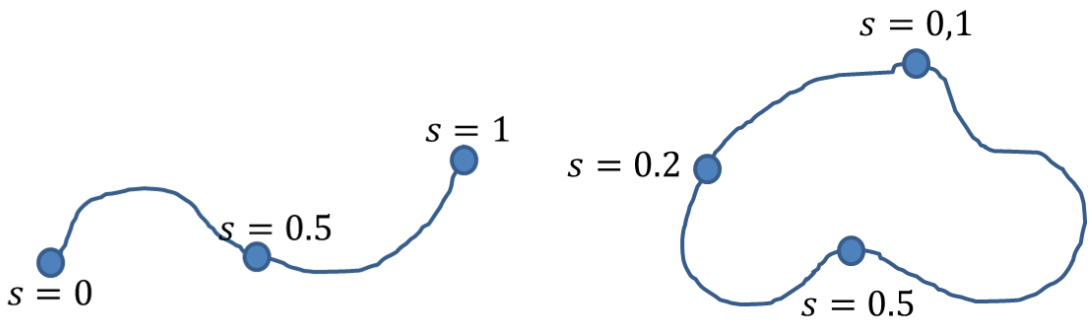


Figure 3.14: Two examples of parametric curves used to represent the snake.

***Property 1*:** The curve must be able to detect an edge in the image and move to align itself with it. In order to achieve this, the curve should be influenced and driven by the image edge strength $F[x(s), y(s)]$, until the point where F is maximized along the points of the snake.

$$\max \int_{s=0}^{s=1} F[x(s), y(s)]ds$$

Obviously the edge strength's influence on the snake will vary depending on the initial location of the curve. To prevent F from being zero at the initialization, it is advisable to blur the image so that it broadens the width of the edge. The energy/force contribution from the object edge strength is called the external energy. It is designated as:

$$E_{ext}[v(s)] = \left(|G_x[v(s)]|^2 + \left|G_y[v(s)]\right|^2\right)$$

The total external energy of the snake/curve is therefore:

$$E_{ext} = \int_0^1 E_{ext}[v(s)]ds$$

***Property 2*:** The curve must be able to contract or shrink as it is driven by the image forces. The curve is precluded from stretching by introducing this property. Without this, the curve cannot fit smoothly to the objects in the image. The amount of contraction is based on the amount of elastic energy at a point. For curve $v(s)$ its elastic energy is proportional to the square of how much it is being stretched at a point.

$$\left(\frac{dv}{ds}\right)^2 = \left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2$$

The elastic energy of the snake should be minimized when constrained by the edge of the object. This ensures that the snake does not contract through the object. Also, the elasticity of the snake should be controlled; too much of it will cause the snake to pull across the boundaries of the object.

Property 3: The last property is to enable the snake to continue smoothly in regions where crack edge is occluded by foreign objects, shadows, etc. For the snake to achieve this it must possess some stiffness together with the elasticity described under property 2. The stiffness of the snake is designed to be proportional to the curvature of the occluding object; this is also the second derivative of the curve:

$$\left(\frac{d^2v}{ds^2}\right)^2 = \left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2$$

Equation xxx must also be minimized when the curve is constrained by an object. Where there is too much stiffness the curve is unable to fit the object correctly. Combining the second and third property gives the bending energy of the snake or curve. It is also known as the internal energy of the curve. The internal energy is designated as:

$$E_{in}[v(s)] = \alpha(s) * \underbrace{\left\{\left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2\right\}}_{\text{Stiffness}} + \beta(s) * \underbrace{\left\{\left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2\right\}}_{\text{Elasticity}}$$

The more the curve bends, the larger the value of its internal energy. The total internal energy of the snake will therefore be:

$$E_{in}[v(s)] = \int_0^1 E_{in}[v(s)]ds$$

Based on the properties listed above, the parametric curve is chosen such that the following function is minimized:

$$I[x(s), y(s)] = \int_{s=0}^{s=1} \left\{ \alpha(s) * \left\{ \left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2 \right\} + \beta(s) * \left\{ \left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2 \right\} - F[x(s), y(s)] \right\} ds$$

$\alpha(s) > 0 \; and \; \beta(s) \geq 0$ represents the amount of stiffness and elasticity that the snake is to have. The negative sign in front of the $F[x(s), y(s)]$ is because equation xx is being minimized. Minimizing $-F[x(s), y(s)]$ is the same as maximizing $F[x(s), y(s)]$. Equation xxx is also known as the total energy of the snake. Similarly, combining property 1 and 2, the total energy of the snake can be defined as:

$$E_{total} = E_{in} + E_{ext}$$
$$= \int_{s=0}^{s=1} \left\{ \alpha(s) * \left\{ \left(\frac{d^2x}{ds^2}\right)^2 + \left(\frac{d^2y}{ds^2}\right)^2 \right\} + \beta(s) * \left\{ \left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2 \right\} - F[x(s), y(s)] \right\} ds$$

Let us represent the curve with a set of n points. That is:

$v_i = (x_i, y_i)$ For $i = 0 \ldots . n - 1$

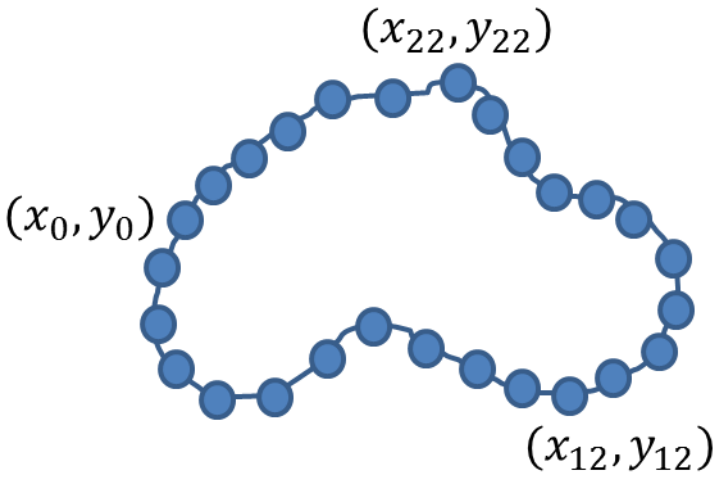


Figure 3.15: A parametric curve represented by a set of n points.

$$\frac{dv}{ds} = \frac{(x_{i+1} - x_i) + (y_{i+1} - y_i)}{2} = \frac{v_{i+1} - v_i}{2}$$

$$\frac{d^2v}{ds^2} = (y_{i+1} - 2y_i + y_{i-1}) + (x_{i+1} - 2x_i + x_{i-1}) = v_{i+1} - 2v_i + v_{i-1}$$

Therefore, the internal energy becomes:

$$E_{in} = \sum_{0}^{n-1} \alpha * |v_{i+1} - v_i|^2 + \beta * |v_{i+1} - 2v_i + v_{i-1}|^2$$

Parametric active contours have been applied successfully in a wide range of applications (Moller and Posch 2012, Bo et al. 2004). They have three key limitations however, due to parametric representation.

1. Difficult to adapt to different topology because it is unable to split and merge.
2. It is too sensitive to model parameters.

3. No external force acts on points which are far away from the edge boundary.

A more efficient model known as the level set active contour algorithm is reviewed and applied to detect edges in pavement distress images in the following section.

3.1.4.2.Level set Active Contour Algorithm

The level set algorithm developed in Li et al. (2005) is implemented in the work. The evolving level set function is approximated to a signed distance function. Equation 3 is used as a metric to characterize how close a function φ is to a signed distance function.

$$P(\varphi) = \int \frac{1}{2}(|\nabla\varphi| - 1)^2 dxdy \qquad (3)$$

The snake is influenced by two types of forces.

- The Internal Energy (E_{int}): Also called the bending energy of the curve. It preserves the snake (keeping it smooth) as it is being pulled by image forces.

  $$E_{int} = \mu P(\varphi) \qquad (4)$$

  Where $\mu > 0$ is a parameter controlling the effect of penalizing the deviation of φ from a signed distance function.

- The External Energy (E_{ext}): The force that moves the snake points or zero level curve. It is dependent on the edge strength at a point; the stronger the edge, the stronger the pull. The external energy is supposed to be minimal when the snake is at the object boundary position. We designate this energy as:

  $$E_{ext}(\varphi) = \lambda \int g\delta(\varphi)|\nabla\varphi|dxdy + \nu \int gH(-\varphi)dxdy \qquad (5)$$

$$\delta(\varphi) = \begin{cases} 0 & |x| > \varepsilon \\ \frac{1}{2}\left[1 + \cos\left(\frac{\pi x}{\varepsilon}\right)\right], & |x| \leq \varepsilon \end{cases} \quad (6)$$

$\lambda > 0$ and ν are constants, δ is the univariate Dirac function, and H is the Heaviside function.

The total energy E_{total} can therefore be defined as:

$$E_{total}(\varphi) = \mu P(\varphi) + \lambda \int g\delta(\varphi)|\nabla\varphi| dxdy + \nu \int gH(-\varphi) dxdy \quad (7)$$

3.1.4.3.Application and Discussion of Results

Modifications and different variations of the algorithm described above are applied to pavement images acquired under different environmental conditions, helping the authors to evaluate the merits and limitations of the algorithm.

The snake model should adapt to different environments and distress types. To achieve this, it is important for the model to have general information on the type of distress it is trying to detect before initializing the contour. This information (which is the type of crack) can be obtained from the crack retrieval algorithm.

Transverse and Longitudinal Cracks: Here, we consider one or a maximum of two cracks in the same direction or of the same type on one image. At most two contours are initialized and its orientation will be based on the type of distress in the image. From figure 3.16 above, it is realized that the algorithm is able to detect the accurate location and shape of the distress.

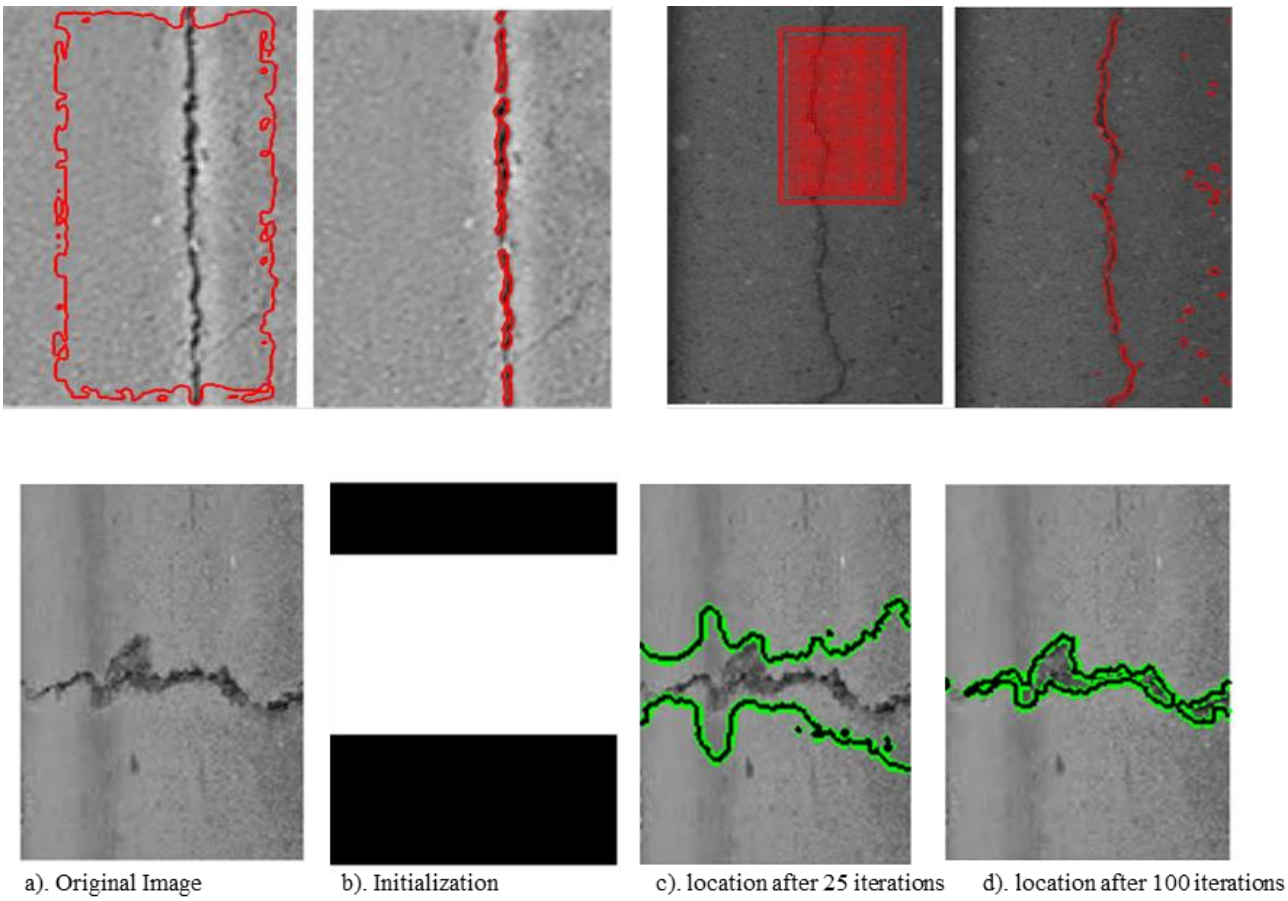


Figure 3.16: First row: the first and third images represent how the active contours were initialized. The second and fourth images show the identified crack after several iterations of the initial contour. Second Row: Initialization and segmentation of transverse cracks with the snake model.

Complex Cases: The complexities involved in crack detection increase under the following conditions:

- Pavement image texture is very rough.
- Pavement image surface has foreign objects such as oil stains, paint markings, etc.
- Image contains a mixture of distress types. For example, longitudinal and transverse cracks.

Under these conditions, it is almost impossible to use edge detection or thresholding in such environments. The snake model is robust in such conditions due to its unique interpretation of the edge detection concept. Under such conditions as listed above, the contours should be initialized as equally-spaced multiple circles or squares. The contours should have the capability to merge or split depending on local image conditions and also compete with each other to distinguish between the different crack severities. Figure 3.17 shows results obtained by using the snake algorithm. From the first column, the algorithm is able to accurately separate distresses (defects) from other objects such as gravels. In column two, paint markings are clearly differentiated from the pavement crack. Column three illustrates the algorithm's robustness against oil stains.

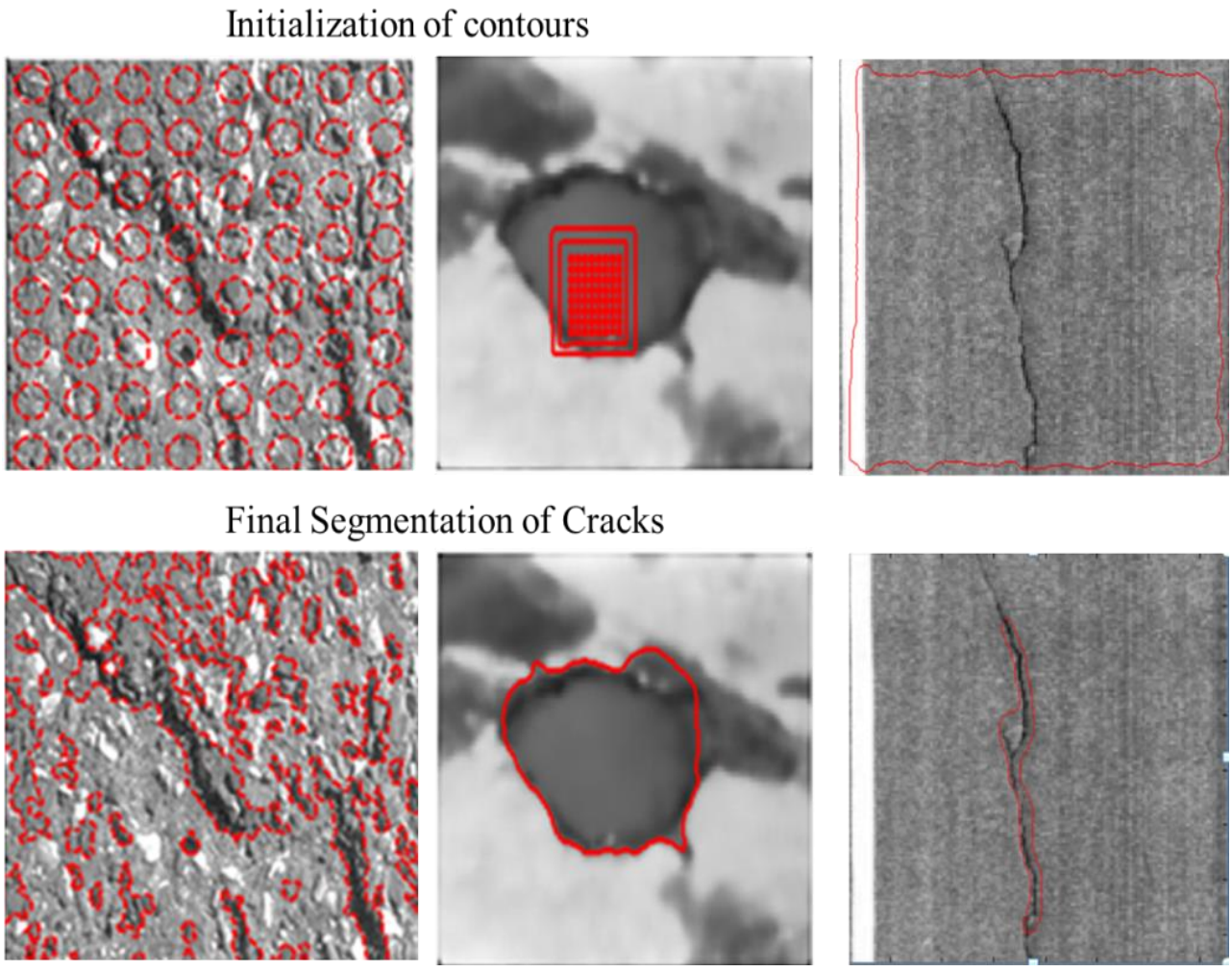


Figure 3.17: The first row represents how the active contours were initialized. The second row shows the identified crack after image smoothening and several iterations of the initial contour.

Pattern Cracks (Alligator and Block Cracks): Pattern cracks present a new set of challenges. First, the cracks are interconnected; that means a singular contour or equally spaced multiple contours cannot be used. Also, the severity of crack widths is disparate. Since active contours are designed to move towards features with broader widths, fine cracks may be missed leading to inaccurate detection results.

3.2.Concluding Remarks

This paper develops a complete vision system for pavement distress detection. Image retrieval algorithms such as Harris, projection or correlation based detectors are used to pre-screen acquired images. This will reduce to the image database to only cracked images. Consequently, the speed of the vision system is improved by inhibiting non cracked images from the feature extraction process. The use of a multiresolution image enhancement algorithm based on Gaussian pyramids is an efficient and fast method for background standardization. The resulting denoised image improves the efficiency of thresholding and edge detection algorithms.

Finally, the active contour models developed in this chapter are useful and objective tools for detecting different types of distresses in pavement images. They show promise in detecting pavement distress in very noisy environments. Their ability to split and match different topologies of the image data is essential for accurate crack location and shape detection. The algorithm can be extended to other problems in civil infrastructure systems such as void detection in bridge decks, traffic sign inventory, etc. The algorithm is unable to detect certain types of cracks such as alligator cracks. Also, very fine cracks can be missed. Lastly, the initial location of the active contour can affect the crack detection process. Future work should be geared toward an automated means of contour initialization.

REFERENCES

Bo Zhang; Zimmer, C.; Olivo-Marin, J.-C.;, "Tracking fluorescent cells with coupled geometric active contours," *Biomedical Imaging: Nano to Macro, 2004. IEEE International Symposium on* , vol 1, no., pp. 476- 479

Chambon, Sylvie., and Moliard, Jean-Marc. (2011). Automatic Road Pavement Assessment with Image Processing: Review and Comparison International Journal of Geophysics Volume 2011, Article ID 989354.

Harris, C., and Stephens, M.J., (1988).“A combined corner and edge detector”. *In Alvey Vision Conference*, pp. 147–152, 1988.

Kass, M., Witkin, A. and Terzopoulos, D., (1988). “Snakes: Active Contour Models.” *International Journal of Computer Vision*., Volume III, pp. 321-331.

Kim, J. Y. (2009). “Development of New Automated Crack Measurement Algorithm using Laser Images of Pavement Surface.” *PHD Dissertation,* 2009., University of Iowa.

Lee, H.D. (2005). “Development of a Manual Crack Quantification and Automated Crack Measurement System.” Project TR-457, Public Center, Civil and Environmental Engineering, University of Iowa.

Li, Chunming., Xu, Chenyang., Gui, Changfeng., and Fox, M. D., (2005). “Level Set Evolution Re-initialization: A New Variational Formulation”, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition.*

Min Li, Chandra Kambhamettu and Maureen Stone , "Automatic Contour Tracking in Ultrasound Images", International Journal of Clinical Linguistics and Phonetics., 19 (6-7): pp. 545-554, 2005.

Moller, B.; Posch, S.; (2012), "Comparing active contours for the segmentation of biomedical images," *Biomedical Imaging (ISBI), 2012 9th IEEE International Symposium on* , vol., no., pp.736-739

Waite J B and W J Welsh, "An Application of Active Contour Models to Head Boundary Location."

Wang, K.C.P., and Gong, W. (2007). "Automated Real-Time Pavement Crack Detection and Classification." Final Report for Highway IDEA Project 111.

Chapter 4

RESEACH IMPLEMENTATION AND SOFTWARE DESIGN

4.0. General Background

Within the past decade, several automated pavement distress image analysis systems have been developed (Lee 2005, Wang 2007, and Kim 2009). A number of these systems separate data processing from acquisition. This allows for experts to assist in the crack information extraction process. Inherent in such systems are some of the demerits of manual methods such as fatigue, inconsistencies, and cost (both money and time). These limitations are, however, minimized as compared to manual methods. There are also real time systems (Wang 2005) that are capable of accurately processing and analyzing image data at the speed of the image acquisition vehicle. Such incredible processing speed is achieved by taking advantage of current multicore CPUs or GPUs to process images in a parallel computing environment. In Wang (2005), a real-time system using parallel processing at the multi-CPU level is proposed. In their work, real time data collection and distress analysis is carried out by implementing two parallel features. The first one is based on parallelism within the CPU, usually called single instruction multiple data (SIMD) and the other is based on a dual processing system programmed with multithread techniques so that each CPU has its own resources to perform image analysis tasks. This work does not focus on parallel implementation of the image processing algorithms; rather, it focuses on parallelism within the hardware architecture. This level of parallelism improves processing rates mainly because the detection system employs algorithms with low computational cost such as image or histogram equalization, median, average, Gaussian filters and morphological operators such as thinning and closing. These algorithms are fast and easy to implement, however, they are not

very adaptive. Hence, they may not be able to detect all types of distress. Also, thresholds consistently need to be fine-tuned for different pavement types and environments.

Recently, very efficient image analysis techniques such as wavelets (Subirats et al., 2006; Nejad et al., 2011), empirical mode decomposition (Ayenu-Prah et al., 2008; Adu-Gyamfi et al., 2011), principal component analysis and its families have been developed. However, because they are very complex and computationally expensive, they are impractical for industries to implement. Some of the algorithms developed in this work fall into this category. Parallelism only at the hardware level will not be sufficient for faster implementation of this work. The first consideration to a successful implementation of this research work involves investigating parallelism within the algorithms and restructuring them to meet the demands for faster processing.

The next step of the implementation process involves integrating the algorithms developed onto a GIS platform to aid estimation of crack extent width and types and visualization for road condition. This will open the door to a fully automated pavement management system. New developments in spatial analysis will be employed for efficient documentation of surveyed results, condition visualization, data query, management and effectively planning maintenance and repair programs.

4.1. Toward Real-Time Implementation of Algorithms

The benefits of developing a real-time crack system when contrasted against an offline system may not be very significant. The reason is this: knowing that a section of a pavement is cracked in a second or in 30 minutes doesn't really make a difference, since response from pavement managers to fix the problems will take much longer. It is, however, unnecessary and a waste of time to sit behind a computer for hours before knowing how well or bad a section of road is. So, our goal is to develop a system that compromises between the two; one that is very fast, not necessarily real-time and also allows for human expert input into the analysis. In order to increase the computational speed of the system we adopt two procedures: Subsampling/Upsampling and parallel processing.

4.1.1. Image Subsampling and Upsampling

Image subsampling implies reducing the size of an image by a certain factor or frequency. That is to say, we are selecting one pixel to represent several pixels in its neighborhood. An example of subsampling by a factor of 0.5 is shown in figure 4.1 below. Obviously, from the figure, there are likely to be challenges if the variance within each neighborhood is high. It is therefore prudent to smoothen the image with preferably a Gaussian filter before subsampling. This will suppress extreme values in neighboring pixels, making subsampling more effective.

$$\begin{bmatrix} 14 & 14 & 31 & 19 \\ 13 & 16 & 18 & 21 \\ 18 & 19 & 11 & 12 \\ 21 & 20 & 10 & 15 \end{bmatrix} \xrightarrow{yields} \begin{bmatrix} 14 & 14 & 31 & 31 \\ 14 & 14 & 31 & 31 \\ 18 & 18 & 11 & 11 \\ 18 & 18 & 11 & 11 \end{bmatrix} \xrightarrow{yields} \begin{bmatrix} 14 & 31 \\ 18 & 11 \end{bmatrix}$$

Figure 4.1: Image subsampling example

Upsampling, on the other hand, is increasing the size of the image by a designated factor or frequency. Here, instead of replicating pixels (as done in the downsampling), missing pixels are found by interpolation. In this work, a bicubic interpolation approach is used. With bicubic interpolation, the output pixel is simply a weighted average of pixels in the nearest 4 by 4 neighborhood. The important steps to improving the speed of a crack detection algorithm simply by using image sub- and upsampling is illustrated below.

Fast crack image analysis using sub and upsampling
1. Apply a smoothening filter (Gaussian preferred) to the original image. 2. Subsample image to the desired frequency. The desired or optimum frequency is related to the crack widths by this equation: $\boldsymbol{W_{crack}} = \frac{\boldsymbol{2*f_{desired}}}{\sqrt{\boldsymbol{M*N}}}$. This equation is obtained by regression using 120 images of different sizes. 3. Apply crack detection algorithms to obtain the desired output. 4. Upsample the output to the original size of the image.

The experiment below shows the efficiency of using the procedure outlined above. Its limitations should not be overlooked, however. Figure 4.2 presents the results of crack detection using sub/up sampling as against a straightforward approach which does not require sub/up sampling. From the figure, the first column and the second column look similar in spite of different sampling frequencies. The processing speed, however, for column 2 is about 2.5 times faster than column 1 (see figure 64.2). For column 4 and 5, processing speed is incredibly high, but crack edge information is degraded.

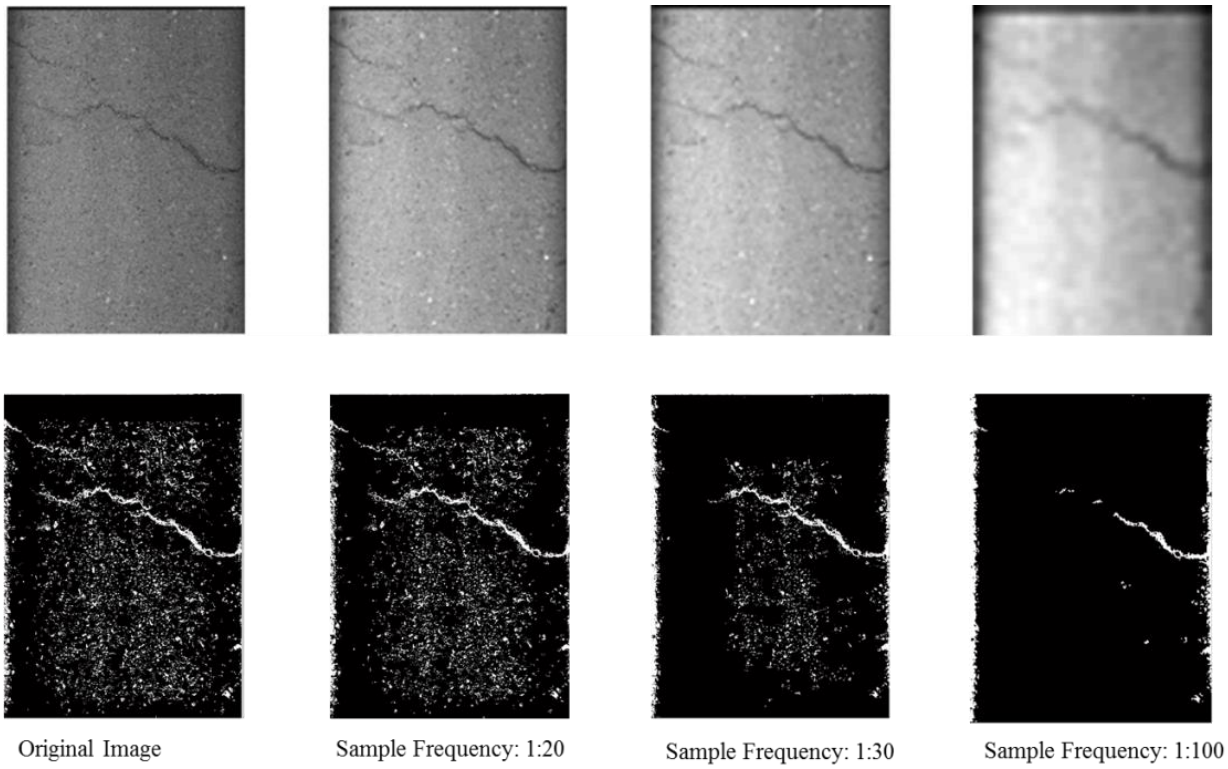
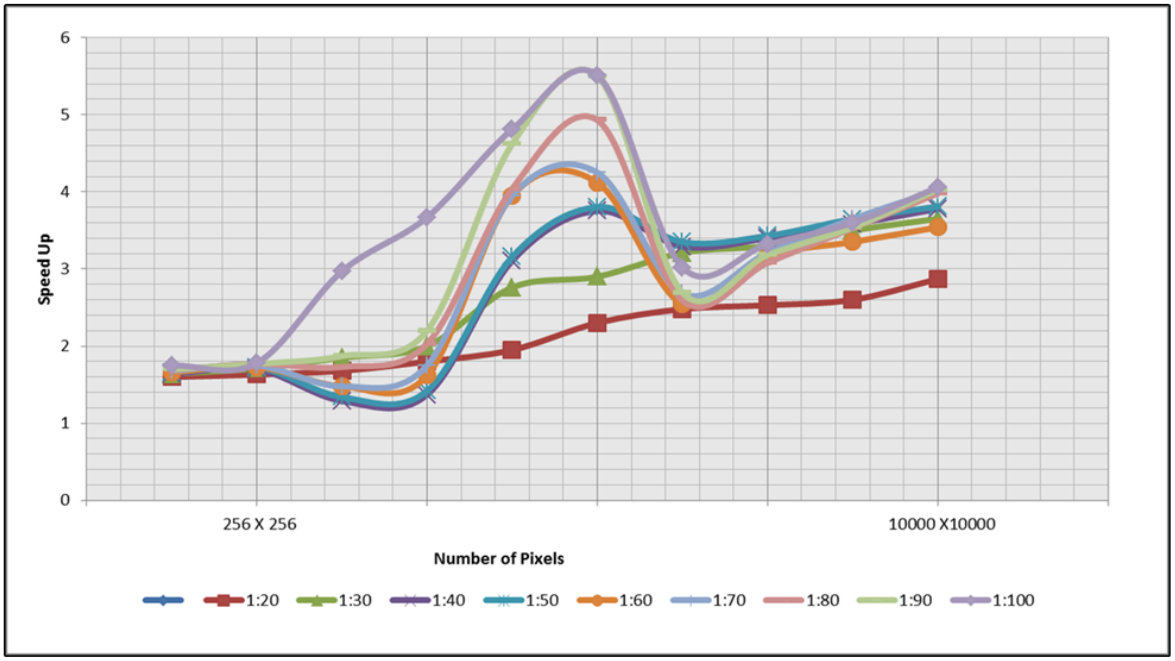


Figure 4.2: Crack detection at different sub-sampling frequency levels.

It is therefore very important to know the optimum frequency required for accurate feature extraction and improved speed. If it is too high, some cracks may be removed; if it is too low, the processing speed will be high. Another benefit of subsampling is denoising. As is portrayed in the figure above, increasing the sampling frequency reduces the number of isolated pixels and

rough textures. Lastly, figure 4.3 shows image sizes for which subsampling produces a more effective result and speed-up. For image size between 180 by 180 pixels and 512 by 512 pixels, speed-up is almost the same for different sampling frequencies. It will therefore be prudent not to subsample images with sizes less than or equal to 512 by 512 pixels. An impressive improvement in speed-up is realized when the images are greater than 512 by 512 pixels but less than or equal to 1500 by 1500 pixels. The speed-up could be as high as 5X. Beyond image size of 1500 by 1500 pixels, the speed-up reduces but is still significant between 2X to 4X.



	Speed Up								
Pix.No/Samp. Freq	1:20	1:30	1:40	1:50	1:60	1:70	1:80	1:90	1:100
180 X 180	1.60	1.64	1.64	1.66	1.66	1.68	1.69	1.70	1.75
256 X 256	1.63	1.71	1.70	1.73	1.73	1.75	1.77	1.77	1.78
512 X 512	1.68	1.85	1.29	1.34	1.48	1.48	1.73	1.87	2.97
1024 X 1024	1.80	2.00	1.37	1.42	1.62	1.75	2.03	2.20	3.67
1500 X 1500	1.95	2.76	3.11	3.16	3.95	3.95	4.04	4.62	4.81
2048 X 2048	2.30	2.90	3.75	3.80	4.12	4.25	4.93	5.50	5.51
4096 X 4096	2.48	3.21	3.29	3.35	2.55	2.68	2.59	2.69	3.02
5000 X 5000	2.53	3.30	3.40	3.43	3.16	3.21	3.08	3.18	3.32
8192 X 8192	2.60	3.50	3.58	3.65	3.35	3.65	3.53	3.54	3.6
10000 X 10000	2.87	3.65	3.78	3.81	3.54	4.02	3.98	4.03	4.06

Figure 4.3. Computational time gained by using subsampling/upsampling technique for different image sizes.

4.1.2. **Parallel Processing:**

A reason for looking at parallel processing is the issue of massive image data. Crack detection systems are expected to extract crack information from miles of road images. Each image contains approximately 1 million pixels. This will obviously increase computational time. To overcome this burden, most systems restrict the processing area to the resolution of the acquisition camera. However, analyzing pavement distress using restricted image sizes could lead to inaccurate classification of cracks. For example, a block crack could be misinterpreted as two longitudinal or transverse cracks depending on the camera view. To resolve this challenge, individual images could be stitched together to form one massive layer. This will improve results of crack classifications and condition ratings since analysis will be based on a network level. By re-structuring and re-thinking through the application of image analysis algorithms while taking advantage of current multicore computers, the authors will seek to simultaneously improve the accuracy and computational cost for processing massive distress image datasets.

4.1.3. **Parallel BEMD**

In chapter 4, BEMD was successfully used to denoise, enhance and detect distresses in images with the help of Principal Component Pursuit – PCP (Candes et al. 2009). The main disadvantage of this algorithm is its time consuming nature. The algorithm involves four main steps: Extrema detection, Interpolation, Sifting and Reconstruction. Extrema detection and

interpolation consumes about 80% of the computational time. The remainder of the time is used for the sifting process and reconstruction. The main focus for improving the speed of this algorithm is therefore on extrema detection and interpolation. In the following sections, a parallel implementation of extrema detection and interpolation is presented. It is important to note that sifting and reconstruction cannot be implemented in parallel due to data dependencies. The sifting process, however, is faster if an efficient interpolation technique is chosen. We take advantage of MATLAB Parallel Computing Toolbox for this implementation (Matlab, 2010)

4.1.3.1.Design Methodology

The design of an effective and fast algorithm is far more than just coding, but also intelligent manipulation of matrices. Our design of a parallel system uses two main approaches; Serial Parallelism and PARFOR/SPMD.

Serial Parallelism (SP): The goal of SP is to reconstruct a 2D image from its 1D components without losing spatial correlation between pixels (Smith, 2010). It also helps to improve the performance of non-serial parallelism. This technique is used in extrema detection and interpolation.

PARFOR & SPMD: "parfor" is a command used in MATLAB's parallel computing toolbox to run loops in parallel. A key limitation of this command is its inability to handle dual indexing such as $f(i,j)$; here let f be an arbitrary variable with i,j being the indexes. SP was used primarily to overcome this limitation. As explained, with SP, processing is one-dimensional. Hence, we are not bothered with dual indexes.

Combining SP and “parfor” tremendously improved speed for extrema detection and basis function’s distance matrices. In cases where SP cannot be implemented such as image reconstruction, feature extraction, etc., image arrays are co-distributed as shown in figure 4.4. Each local segment is processed in parallel using Single Program Multiple Data (SPMD) statements. The SPMD command is like a very simplified version of MPI. There is one client process, supervising workers who cooperate on a single program. Each worker (sometimes also called a \lab") has an identifier, knows how many workers there are in total, and can determine its behavior based on that ID. Each worker runs on a separate core (ideally) and workspace and they meet at synchronization points;

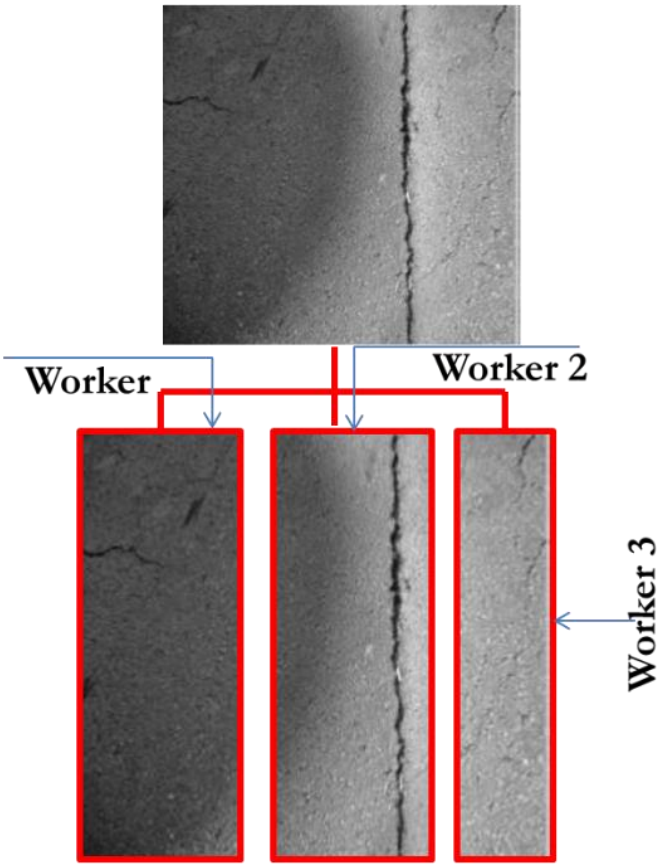


Figure 4.4: Distribution of image parts to different workers

Figure 4.5 Shows the methodology designed for implementing the decomposition algorithm in parallel. Two key factors are considered in this design.

Avoid interpolation of limited data: The parallel architecture will require partitioning and running image data on shared memories. This may lead to interpolation of limited data on the shared memory. The system is designed such that when the number of image extrema on the shared memory falls below a certain threshold, image interpolation will be performed on the global memory.

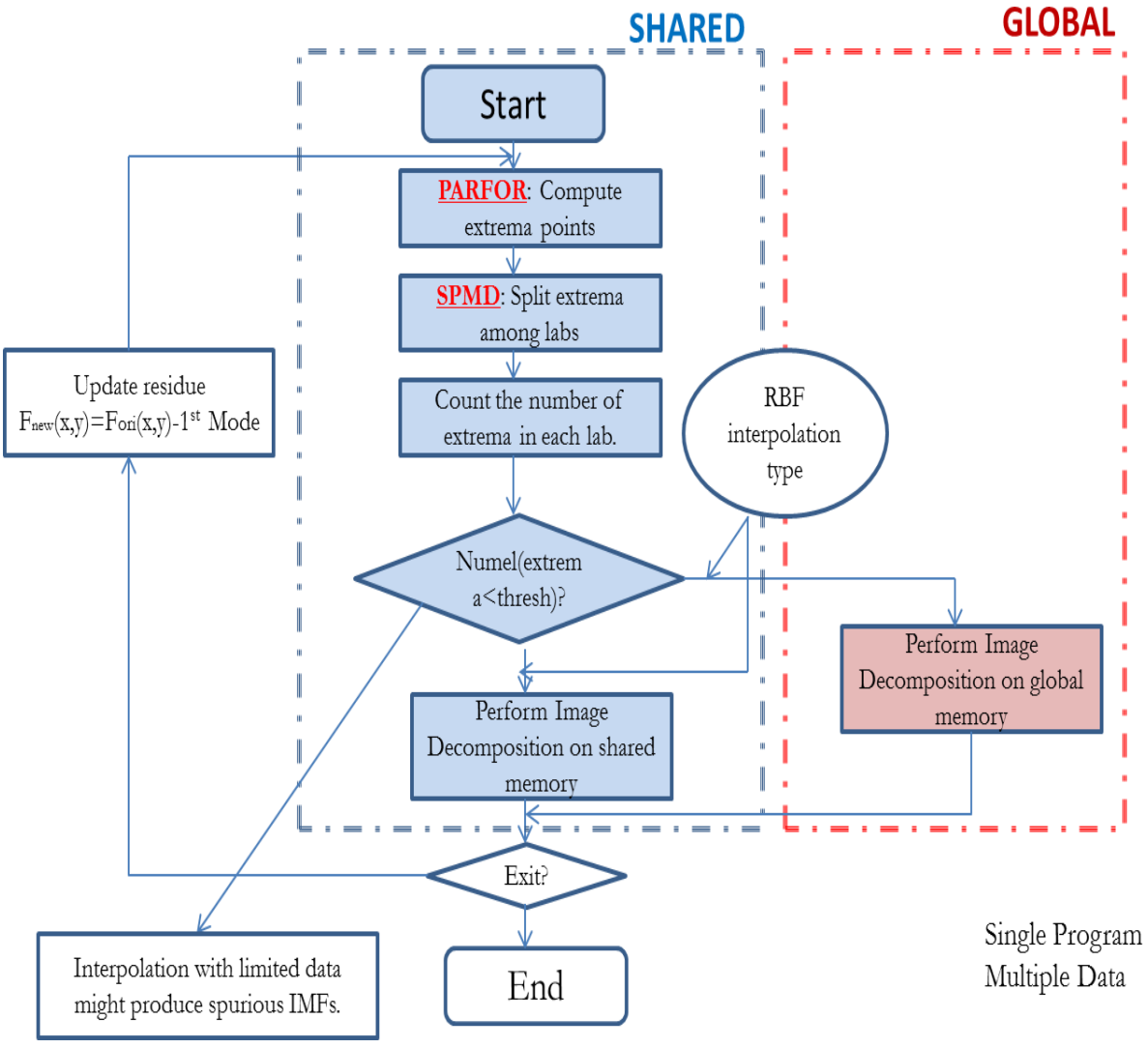
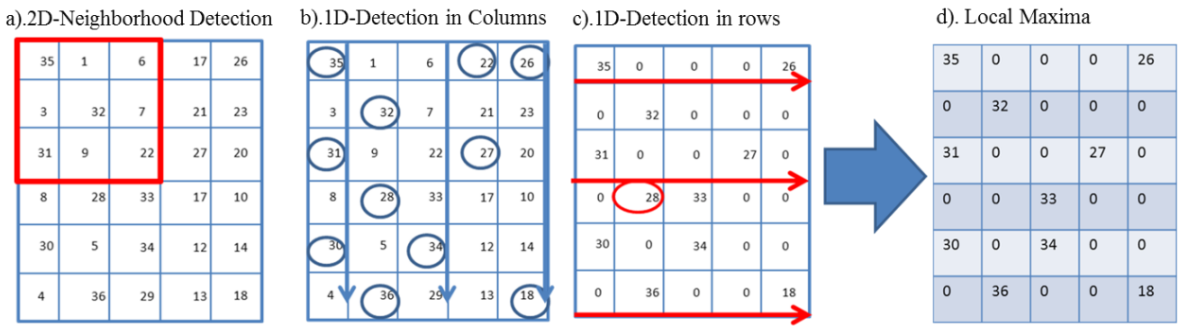


Figure 4.5: Design for fast computation of BEMD algorithm

Efficient choice of basis functions for interpolation: The Radial basis function (RBF) is used for interpolation during the sifting process. However, there are different variations of the RBF (linear, cubic, multiquadric, etc.) having different levels of accuracy depending on the size of data involved. The higher the accuracy, the higher the computational cost. In designing parallel architecture for data interpolation, the faster but less accurate basis function approximations are used when we have a lot of data points to interpolate. For limited data, multiquadric basis functions are used.

Extrema Detection

Extrema detection involves finding high and low points in a region. Usually, a neighborhood region analysis method is used to search for extrema. This ensures that spatial correlation between pixels is not lost. The choice of an efficient neighborhood region size is, however, uncertain and also not embarrassingly parallel (i.e. it is difficult to separate into a number parallel tasks). In order to take advantage of parallel processing technique, the 2D image is processed from its 1D components and extrema are searched in rows and columns (see figure 4.6). This helps remove the restrictions on neighborhood size, it is also easy to implement in parallel using MATLAB's “PARFOR”




```
function [maxima,minima]=parallel_extrema(x)
parfor i = 1:no_columns/no_rows
        y(:i) = findpeaksandlows[x(:,i)'];
        ["maxima,minima"] =
              findpeaksandlows[y(i,:)]
end
```

Figure 4.6: a). Neighborhood Extrema detection in series b) & c). Parallel Implementation of extrema detection. D). Extrema detected.

Extrema Interpolation

As explained in previous chapters, the Radial Basis Function is used as the interpolation technique for the BEMD decomposition since it imposes fewer restrictions on the geometry of the interpolation centers and is suited to problems where the interpolation centers do not form a regular grid as in the case of local maxima or minima maps of images or textures.

The radial basis interpolation function is designed as:

$$P_f(x) = \sum_{k=1}^{N} c_k(\psi\|x - x_k\|) + \sum_{l=1}^{M} d_l p_l$$

Parallelizing the Distance Matrix or Basis Function

Consider constructing a distance matrix, $\psi\|x - x_k\|$from a very simple square matrix of size 4 by 4 in parallel using just 2 workers or labs. The distance matrix can be implemented in parallel as follows (see figure 4.7):

Step 1: Reconstruct matrix into a 3-D matrix of 1 by number of rows by number of columns. For example if the matrix is as below, the new matrix is of size 1 by 4 by 4. Each face of the 3D matrix is a column of the original matrix.

$$Original\ Matrix = \begin{matrix} 16 & 2 & 3 & 13 \\ 5 & 11 & 10 & 8 \\ 9 & 7 & 6 & 12 \\ 4 & 14 & 15 & 1 \end{matrix}$$

Transpose the 3D matrix and assign it to a different worker (worker 2). Keep the original 3D matrix on worker 1.

Step 2: Replicate each face of the matrix such that each face has a dimension similar to the dimensions of the original matrix.

Final Step: Re-group results from different workers onto the global memory and calculate the distance matrix.

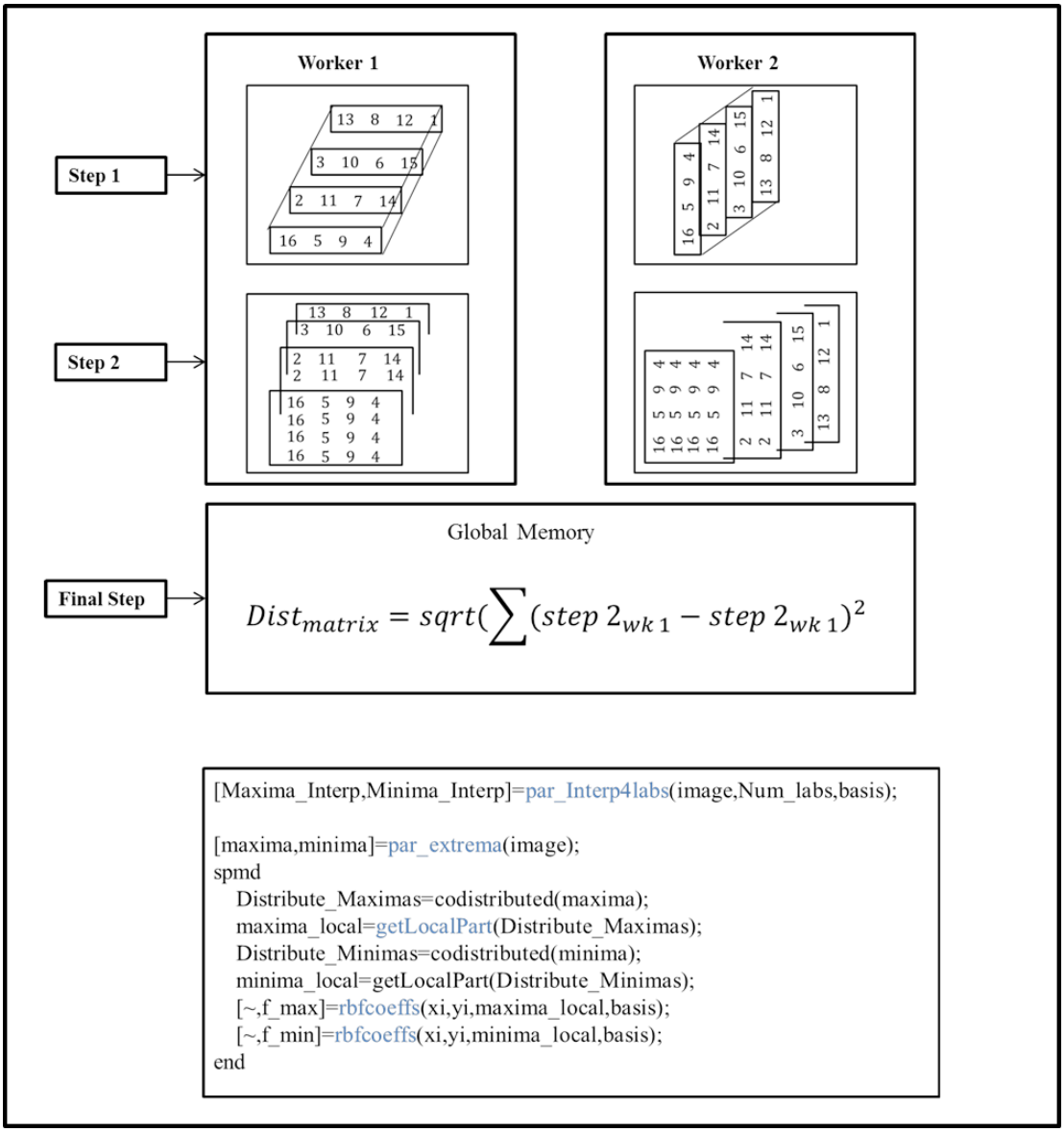


Figure 4.7: Splitting images over different workers or labs before running parallel extrema detection and interpolation.

Basis Functions Used

Three different types of basis functions were used in our analysis; linear, Gaussian and multiquadric basis. They are designed as follows:

$$\psi(r) = e^{-(\varepsilon r)^2}; r = \|x - x_i\| - Gaussian$$

$$\psi(r) = \sqrt{(1 + (\varepsilon r)^2} - Multiquadric$$

$$\psi(r) = r; \ -linear$$

The different basis function types pose two key issues: when to use a particular basis function and how should ε be selected.

4.1.3.2.Choosing a Basis Function:

The basis functions used for interpolation could affect the overall computational time and accuracy considerably. Whereas linear and Gaussian basis functions may not be as accurate as multiquadric, they are faster. The experiment in Table 4.1 serves as a guide to selecting the appropriate basis function for interpolation.

- Although multiquadrics are more accurate, its RMS error appears to converge (see Table 4.1) with linear basis functions as the number of interpolation points increase.
- Use linear interpolation for extracting high-frequency modes (edge-dominant modes) since they are faster, while multiquadrics are used when there are few extrema or when global memory is been used.

4.1.3.3.Choosing an Optimal Shape Parameter:

The optimal shape parameter is the one with:

- The least RMS error.
- No ill condition warning from MATLAB.

From Table 4.2, this value is 10.5 for multiquadric and 16.2 for Gaussian.

Table 4.1: Image Size and Interpolation Accuracy

	Gaussian	Multiquadrics	Linear
N	RMS Error	RMS Error	RMS Error
180 X180	0.218	0.235	0.221
256X256	0.203	0.135	0.141
350X350	0.108	0.023	0.029

Table 4.2: Optimal Shape Parameter value

	Gaussian		Multiquadrics	
N	S.P	RMS error	S.P.	RMS error
180 x 180	3	0.235	4.8	0.373
	5.4	0.135	9.2	0.315
	16.2	0.023	10.5	0.202
	20.4	0.422	18.5	0.225
	25	0.673	23.2	0.372

4.1.3.4.Speed-up (CPU)

Resources:

- Computers: All computations were done on two computers.
  - Intel (R) core (TM) 2 duo, 2.5GHz
  - Intel (R) core (TM) 2 quad, 2.5GHz
- Toolbox: MATLAB Parallel Computing Toolbox

Extrema Detection: A parallel implementation of the extrema detection algorithm increases the overall computation speed. In figure 4.8, the speed-up could be approximately 8x to10x for large image sizes. This is when 1D serial parallel is implemented. For 2D-parallel implementation, the speed-up is about 0.5x on average. The root mean square error for using the 1D serial is also shown in the figure below. RMSE is computed using the result of 2D-parallel as the reference.

$$rmse = \sum \|f_{2d} - f_{1d}\|^2$$

For small image sizes, errors for 1D parallel are relatively high. As image size increases, the error value becomes unstable, however. relatively low.

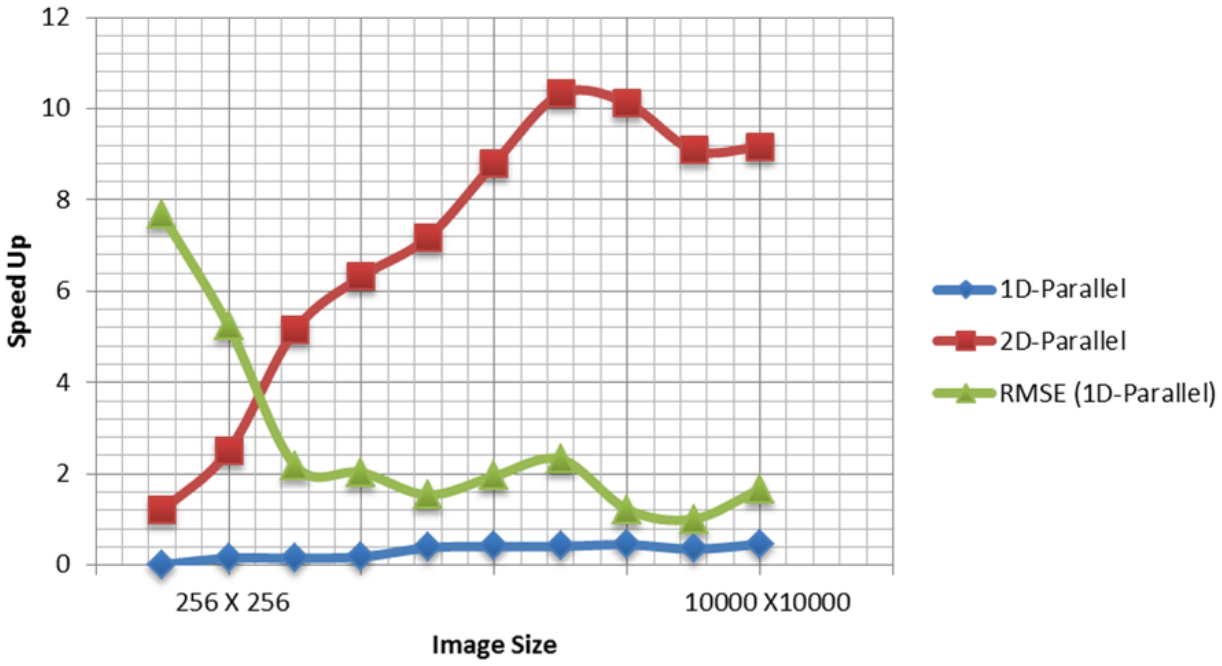
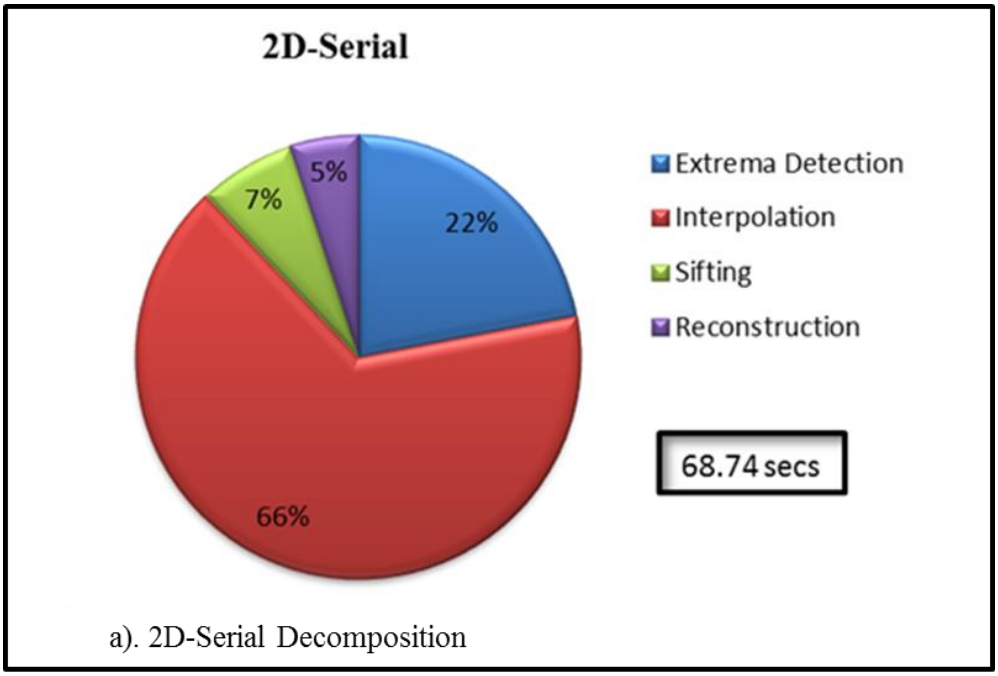


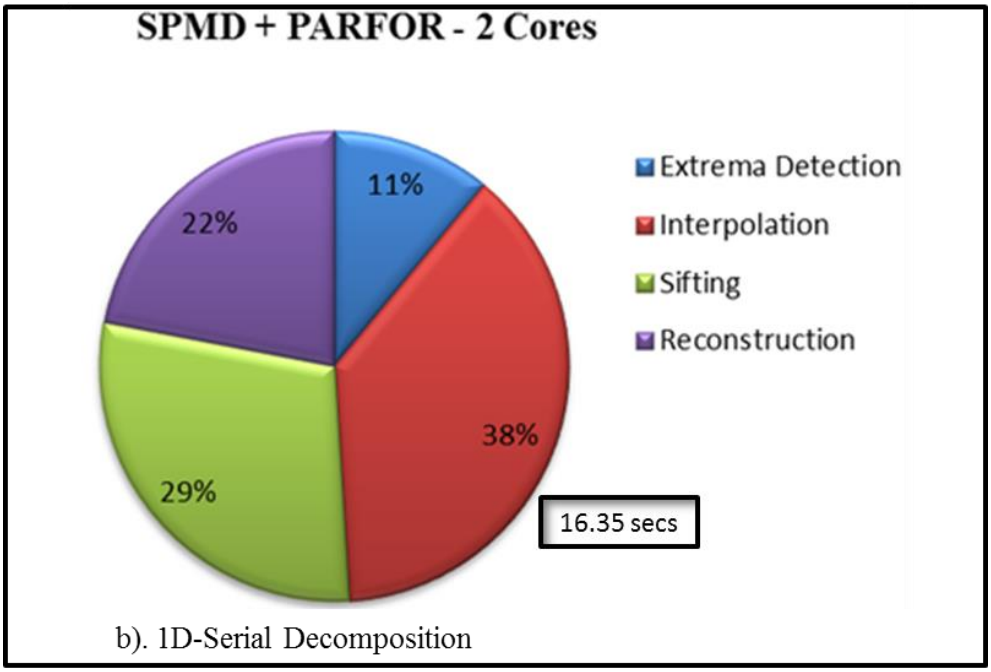
Figure 4.8: Extrema detection using parallel processing.

Table 4.3: Execution time (sec) and Speed using different fast approaches

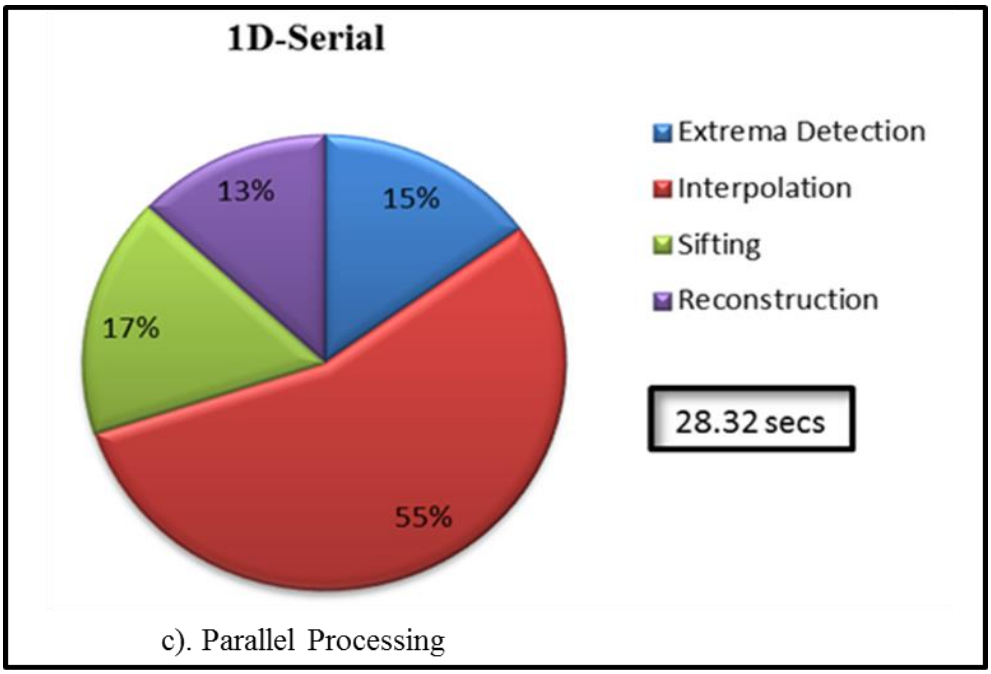
Image Size	Serial (2D)-Dual-Core	Serial (1D) – Dual-Core)	SPMD + PARFOR (Dual-Core)	SPMD + PARFOR (4-Cores)	Sub-Sampling+ quad-core parallel Proc.
180 X 180	23.26	12.49 (1.86x)	8.82 (2.64x)	4.15 (5.60)	1.36 (17.1x)
256 X 256	68.74	28.32 (2.43x)	16.35 (4.20x)	7.28 (9.44x)	3.61 (19.04x)
512 X 512	Out of Memory	Out of Memory	28.16	15.32	5.21
1024 X 1024	Out of Memory	Out of Memory	54.59	33.75	13.82



a). 2D-Serial Decomposition



b). 1D-Serial Decomposition



c). Parallel Processing

Figure 4.9: Overall speed contribution by extrema detection, extrema interpolation, sifting and the reconstruction process.

From figure 4.9, the 2D-BEMD takes 69secs to decompose an image of size 256 by 256 pixels. Of this time, 22% and 66% is used for extrema detection and interpolation respectively. This value decreases by half when the algorithm is implemented in parallel.

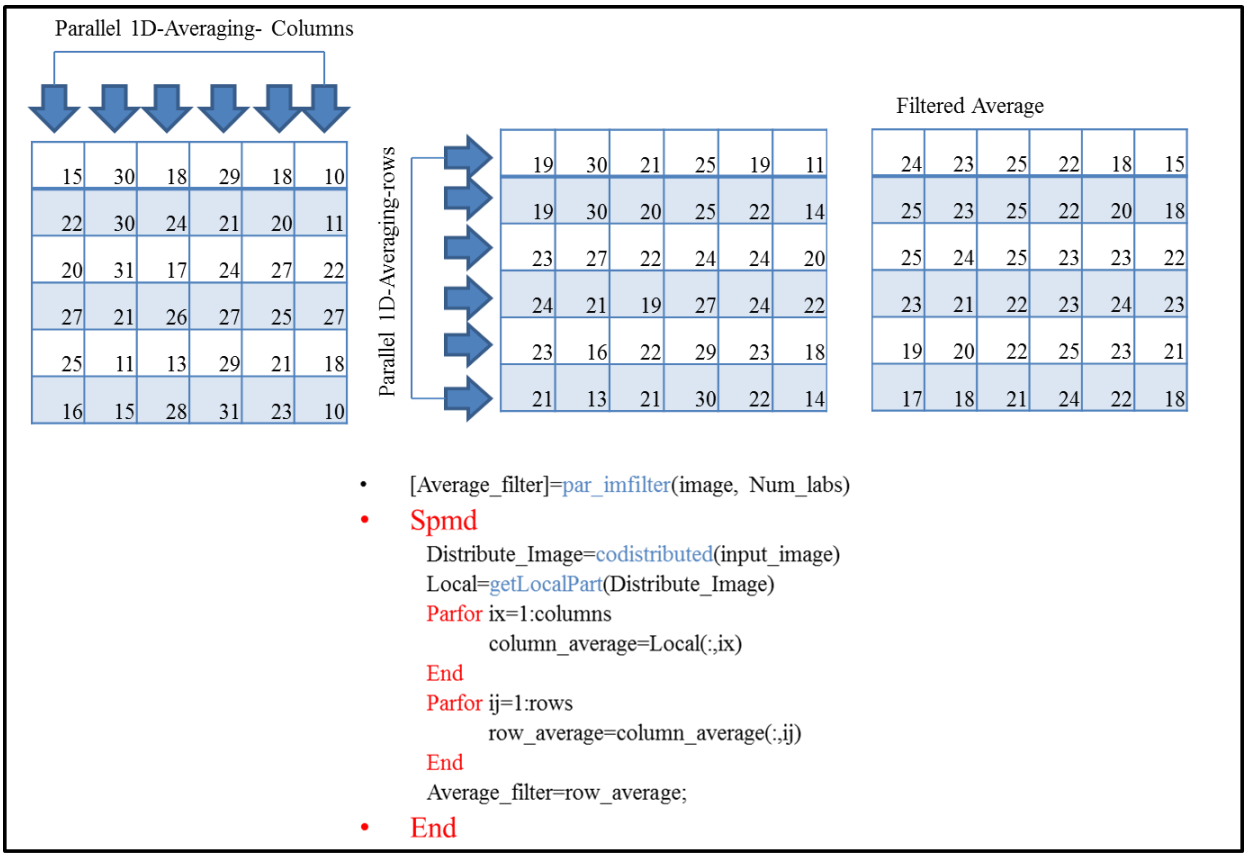


Figure 4.10: Parallel implementation of MATLAB's imfilter algorithms.

Computational speed comparison between the parallel BEMD developed, and a parallel implementation of MATLAB's averaging filtering algorithm is shown in Table 4.4 below. The parallel imfilter algorithm is implemented as shown in the steps described in figure 4.10. Although parallel BEMD is still slower than the averaging and median filters, the difference in speed-up is not very significant considering the merits of using the BEMD for denoising. It

should be noted that the computational times presented in table 4.3 above for parallel BEMD is different from that indicated in table 4.4. This is because for denoising, only two BIMFs are required, meaning higher speed-ups. That is why the speed-up in Table 4.4 is higher than in Table 4.3.

Table 4.4: Image De-noising Computation Speed

Image Size	Imfilter (Average, [3,3])	Imfilter (Median, [3,3])	Parallel BEMD
180 x 180	0.036	0.145	0.56
256 x 256	0.152	0.521	1.21
512 x 512	0.632	1.86	2.62
1500 x 1300	1.87	3.75	6.58

Figure 4.11 compares the speed gained by using the parallel implementation of the BEMD/PCP, Active Contours and Adaptive Thresholding. All the algorithms have common speed-up behavior for image sizes between 180 by 180 pixels and 1024 by 1024 pixels. Speed-up for adaptive thresholding, however, flattens out and even decreases for image sizes greater than 1500 by 1500. The reason for this is because the adaptive thresholding (parallel) is not embarrassingly parallel. Hence, it is heavily dependent on SPMD. The SPMD, however, has only 4 workers (since a quad core computer is used) available for processing images. As the data for each worker increases beyond a certain limit, their processing speed-up reduces. This will not happen for computers with more than four cores. The speed-up for BEMD-PCP and Adaptive

Thresholding, however, continues to increase because it is embarrassingly parallel. It is therefore able to successfully combine parfor and SPMD to execute loops in parallel.

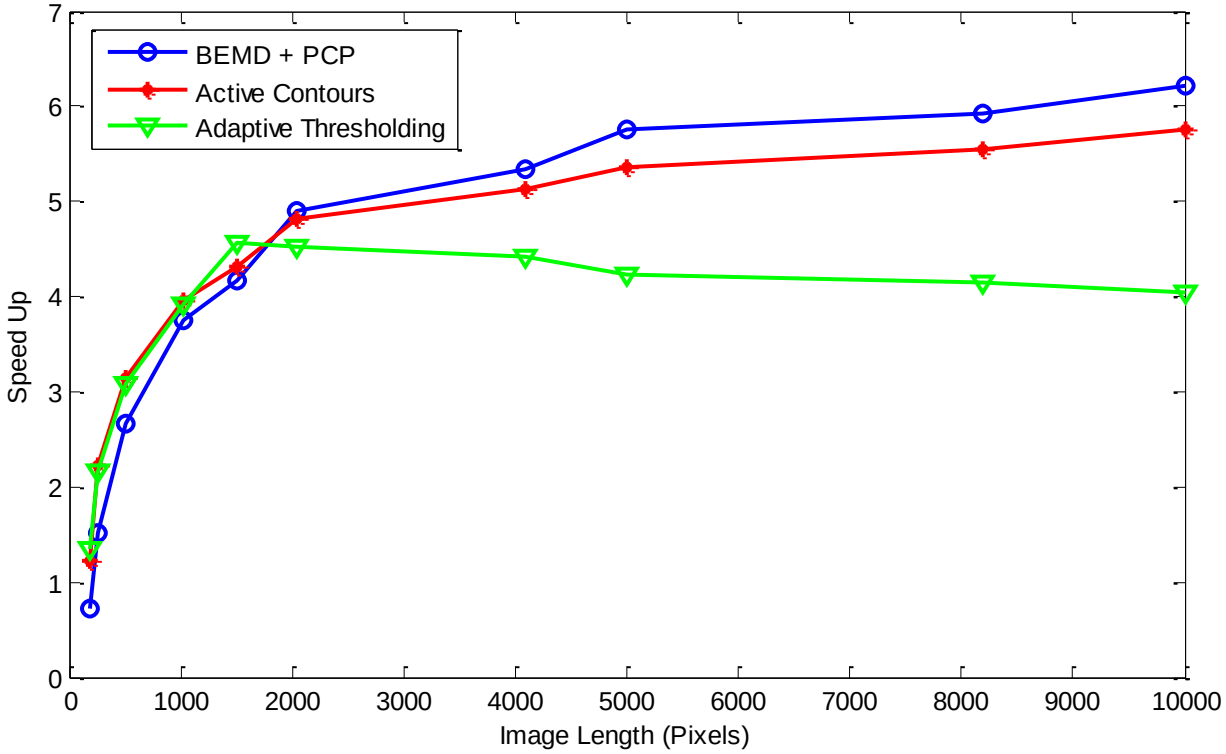


Figure 4.11: Speed improvements for different detection algorithms

4.2.Integrating Detection System onto GIS Platform

Crack detection systems provide inputs for Pavement Management Systems which helps personnel make cost-effective decisions with regard to pavement rehabilitation or re-construction. A Crack Detection System acquires the data and processes it to extract data regarding damages regions. The output is sent to Pavement Management System (PMS) which rates the degree of damage, displays a map of conditions along road segments and manages all

other documents created from the surveys such as prioritization and maintenance cost analysis. In recent years, there is a trend toward integrating crack detection systems (CDS) into Pavement Management Systems (PMS) (Wang et al., 2001, Obaidat and Sharaf, 2006) in order to enhance and support decisions for pavement or asset management. There is, therefore, a demand for efficient tools that can integrate, manage and analyze the information from CDS such that its output will be meaningful to the PMS for improved decision making. There are, however, three main challenges affecting the successful integration of the two systems:

- Platform Differences: A typical example is when algorithms in CDS are written in a programing language that is different from that of PMS.
- Full and Complete Automation: This is one of the biggest challenges. Most CDSs are fully automated; however, the PMS may require routines that require manual editing. Due to this, a fully automated system is difficult to find.
- Massive Data.

A common platform for integrating CDS into a PMS is usually through GIS. This is mainly because of the geographic nature of road networks and GIS's improved resources for spatial data management and analysis. Also, new developments in spatial analysis may be employed for efficient visualization and documentation of the surveyed results, query, manipulation, analysis and management in order to effectively plan maintenance and repair programs. Challenges such as automation and platform differences, however, still plague systems integrated using GIS.

The primary goal of this section is to integrate algorithms developed in previous chapters (Empirical Mode Decomposition/Principal Component Pursuit, Adaptive Thresholding, and Active Contour Models) onto a GIS platform. The steps used to overcome challenges such as platform differences, automation and massive data will also be illustrated. Overall, we seek to

provide a system which offers a safe data collection technique, fast, efficient processing, analysis and visualization of data. The integrated system will automatically provide the following:

- Inspect roads for cracks or damages.
- Identify the type of damage (longitudinal, transverse, block, alligator, etc.), its extent and location.
- Construct a map for visualizing conditions on the different segments of the road.
- Generate priority indexes for rehabilitation purposes.
- Perform maintenance cost analysis.

4.2.1. **Components of the Integrated System**

The platform will have four main components. A flowchart describing how they are integrated into the GIS system is presented in figure 4.12. See Appendix A for the complete code structure used to integrate the crack detections system into ArcGIS. The components are as follows:

- Network Identification and Database Development
- Image Processing System
- Pavement Condition Rating and Analysis System
- Data Management and Visualization
- Priority Indexing and Maintenance Cost Analysis.

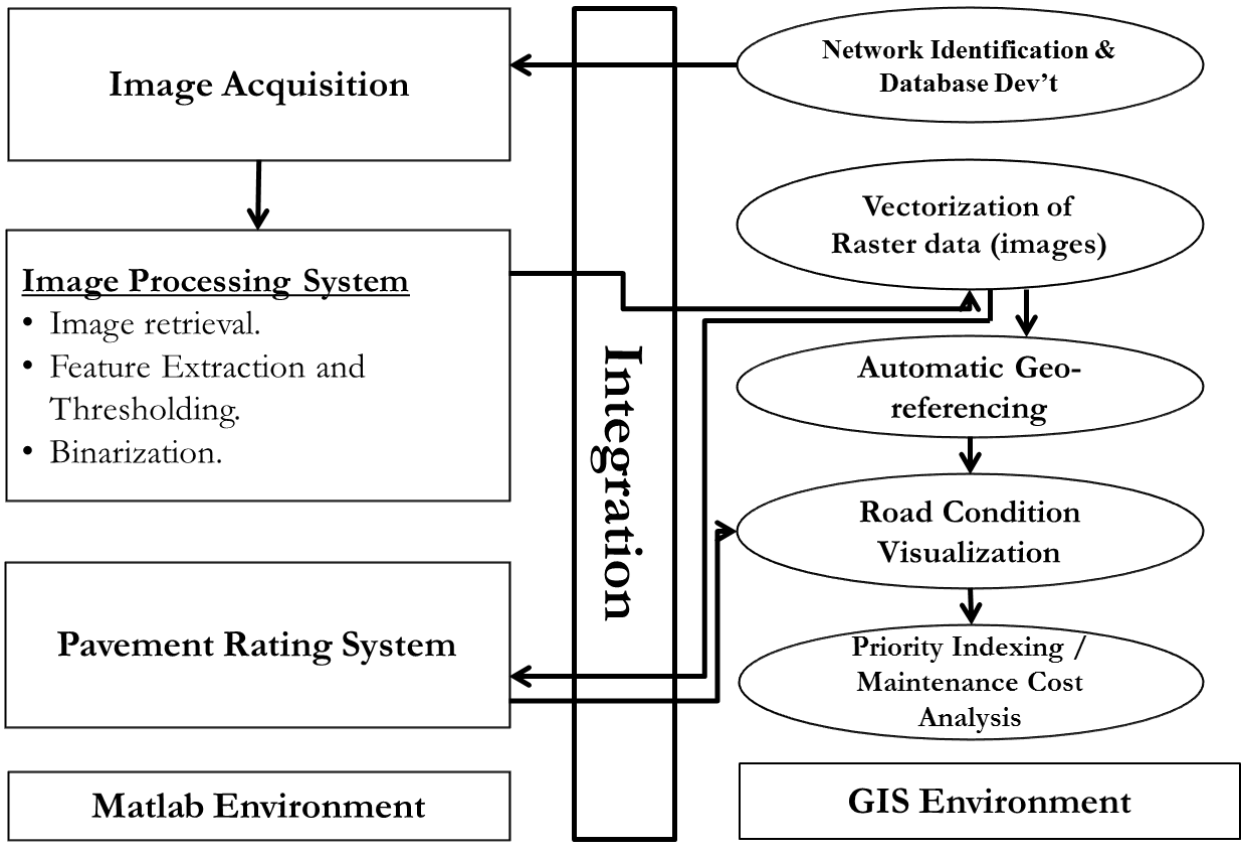


Figure 4.12: Key components of the Integrated System

4.2.1.1.Network Identification and Database Development (NIDD):

The primary goal of NIDD is to segment the entire road network system into manageable divisions for surveys. Also at this stage, database development begins through the acquisition of distress data and all other information such as street names, road type, road width, traffic volume, traffic speed, etc. Roadway information is obtained from individual states' department of transportion. After processing, the database is updated with other information such as damage or crack type and severity conditions and the priority index of each road segment. In modern systems, data acquisition vehicles are mounted with a GPS, hence network identification can be performed simultaneously with distress survey and database development. During this process,

the coordinate of each image is acquired. A potential problem may sometimes occur if the GPS's signal can be lost due to nearby trees or high rising buildings in an area. Also, the GPS instrument is unable to acquire accurate information within fractions of a second since data is being acquired at highway speeds.

For the system developed in this work, a mounted GPS is not required. Coordinates of each image are obtained through an automated georeferencing approach. The process requires coordinates of the first image taken at the beginning of each division in order to coordinate the remainder of the images taken. This work will seek to overcome some of the limitations of the current systems by preceding distress survey and database development with network identification. For our network identification, only six coordinates are required per street; three at the beginning and three at the end. This is shown below in figure 4.13.

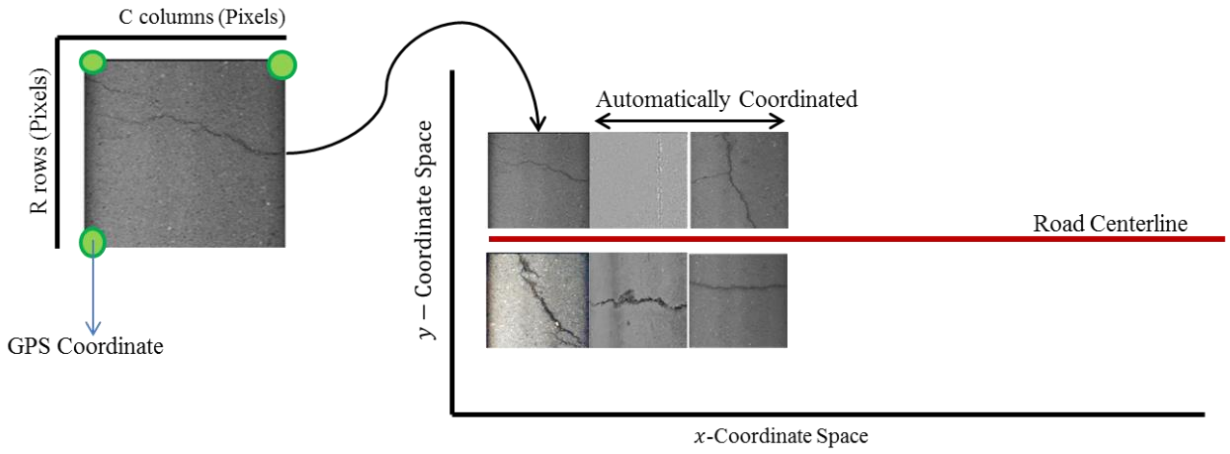


Figure 4.13: Network identification

4.2.1.2.Image Processing System (IPS):

The image processing algorithms used have been discussed exhaustively in chapters 3, 4 and 5. The IPS consists of three key components: Crack image retrieval, image pre-processing or enhancement, feature extraction or thresholding and binarization. At each stage, the system provides different alternatives for processing the image. The user can select the desired option depending on the level of accuracy and speed desired.

Crack Image Retrieval Algorithms

- Harris Detector Method
- Pixel Correlation Method
- Pixel Projection Method

Image Pre-Processing Algorithms

- Gaussian Pyramid based background standardization
- BEMD-based image restoration

Feature Extraction Algorithms

- BEMD and PCP algorithm
- Adaptive Thresholding algorithm
- Active Contour algorithm

4.2.2. System Integration onto GIS Platform

Integrating the vision system developed in previous sections into GIS improves the analysis, assessment and management of road condition information. The integrated system developed provides the following:

- Automates the process of converting binarized (raster) images into vectorized features for spatial analysis.
- Assigns coordinates to every pavement image acquired through automated geo-referencing.
- Provides an improved classification of cracked images and visualization of road condition.
- Improves database management of road condition information.

4.2.2.1.Vectorizing Raster Data:

Vectorization converts binarized (raster) images into geographical features or polygons. Raster cells that are interconnected will be grouped as one polygon. The key advantage of this step is its ability to detect different types of crack in an image. The assumption here is that different types of cracks in a single image will not be connected. Image classification algorithms can only detect one type of crack per image. This is usually not the case; a cracked image could consist of different types of cracks. An added advantage of the vectorization process is that the computation of crack lengths and orientations becomes easier since each crack polygon has its associated coordinates. Each vectorized image will be considered as a separate shapefile layer with its own attributes. Figure 4.14 shows an example of a vectorized cracked image with its attributes.

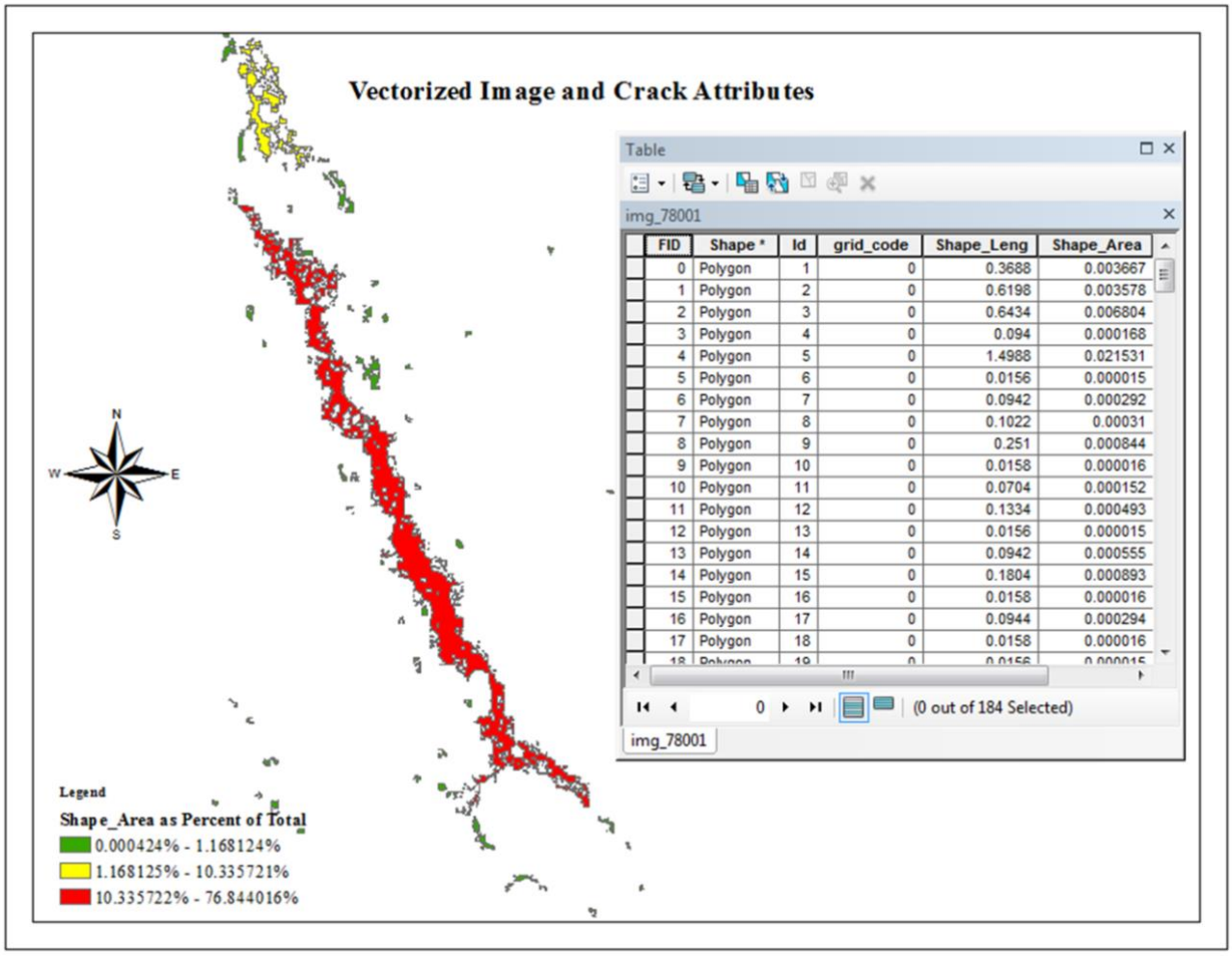


Figure 4.14: Vectorized Image and its crack attributes

4.2.2.2.Automated Geo-referencing:

This step assigns geographic coordinates to every pixel in the processed (vectorized) image. This can only be achieved by a process known as geo-referencing. Traditionally, geo-referencing is achieved only by manual procedures. Ground controls points (at least 4 or 5) are required to register each image. To geo-reference each image, they have to be resized (by stretching or shrinking) in order to maintain the pixels in their right geographical positions. This procedure is

impractical since we are dealing with a huge image database (about 3000 images per mile). The most feasible way will therefore require an automatic way of registering or geo-referencing.

General Idea

Auto geo-referencing is achieved by finding an approximate image location and orientation on a given road segment, which represents the start of imagery. Vertices of the road are read using ArcGIS *ArcPy* – a tool that provides access to geoprocessing tools as well as additional functions, classes, and modules that allow you to create simple or complex workflows quickly and easily. Using one ground image length, the snapshot location for every image is interpolated at the image ground distance using *Python shapely.geometryLineString*. A snapshot location (see figure 4.15) approximates the image center pixel coordinates. Using snapshot locations and the following image properties: upper left corner, pixel size in x, pixel size in y, the approximate ground coordinates are computed for upper left corner and upper right corner.

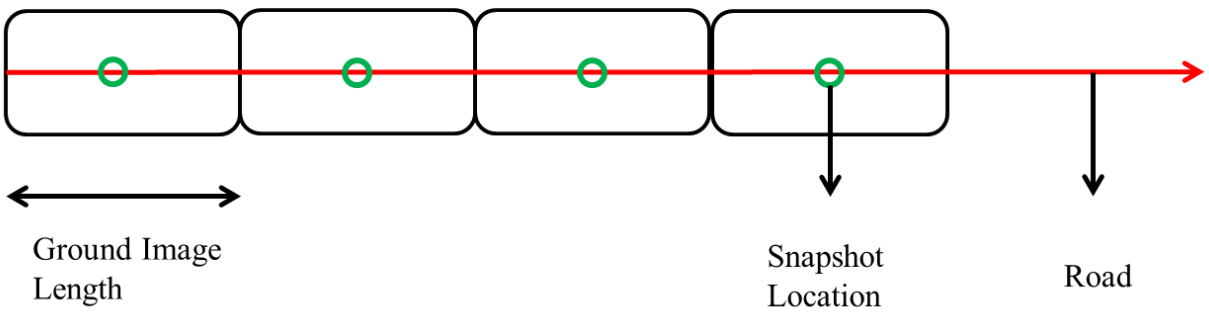


Figure 4.15: Snapshot locations of automatically geo-referenced image.

With at least three ground and image coordinates (center coordinates of the upper left corner and upper right corner), a six parameter image-to-world affine transformation can be used to obtain geo-referencing parameters for a raster *World File*. Each known coordinate generates two equations (x', y').

$$x' = Ax + By + C$$

$$y' = Dx + Ey + F$$

With three coordinates, six equations are generated between image and ground coordinates, and a solution for A, B, C, D, E, F are computed: Transpose $[A, B, C, D, E, F] = N * U$. Matrix calculations are achieved using *PythonNumpy*. A world file is generated containing the parameters needed to geo-reference the image. Where:

N = Transpose $[observation\ matrix] * observation\ matrix$ and

U = Transpose $[observation\ matrix]$*($Transpose\ [x', y'\ terms\ \ldots.]$)

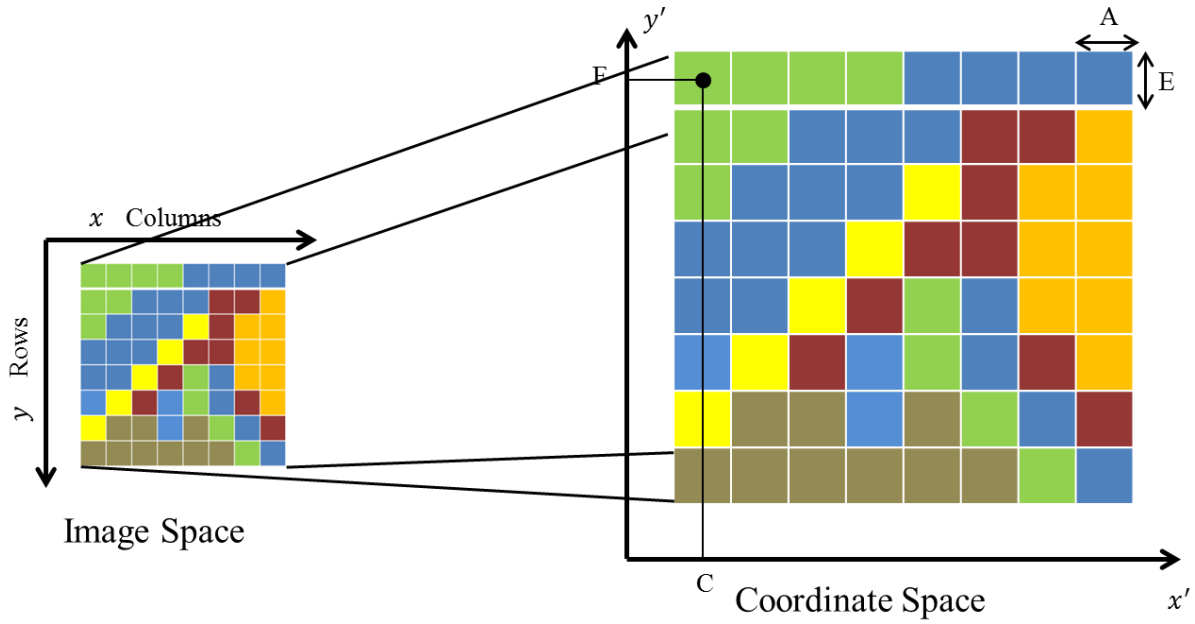


Figure 4.16: Transformation from image space (units in pixels) into geographic coordinate space (units in meters, feet, etc.). $\boldsymbol{x}$ and $\boldsymbol{y}$ are column and row count respectively in image space. $\boldsymbol{x'}$and $\boldsymbol{y'}$ are horizontal and vertical value respectively in coordinate space. A and E constitutes the size (width and height) of each cell in map units. C is the $\boldsymbol{x'}$ value of the center of the upper-left cell. F is the $\boldsymbol{y'}$ value of the center of the upper-left cell. Figure 4.17 shows an example of automatically georeferenced pavement crack images.

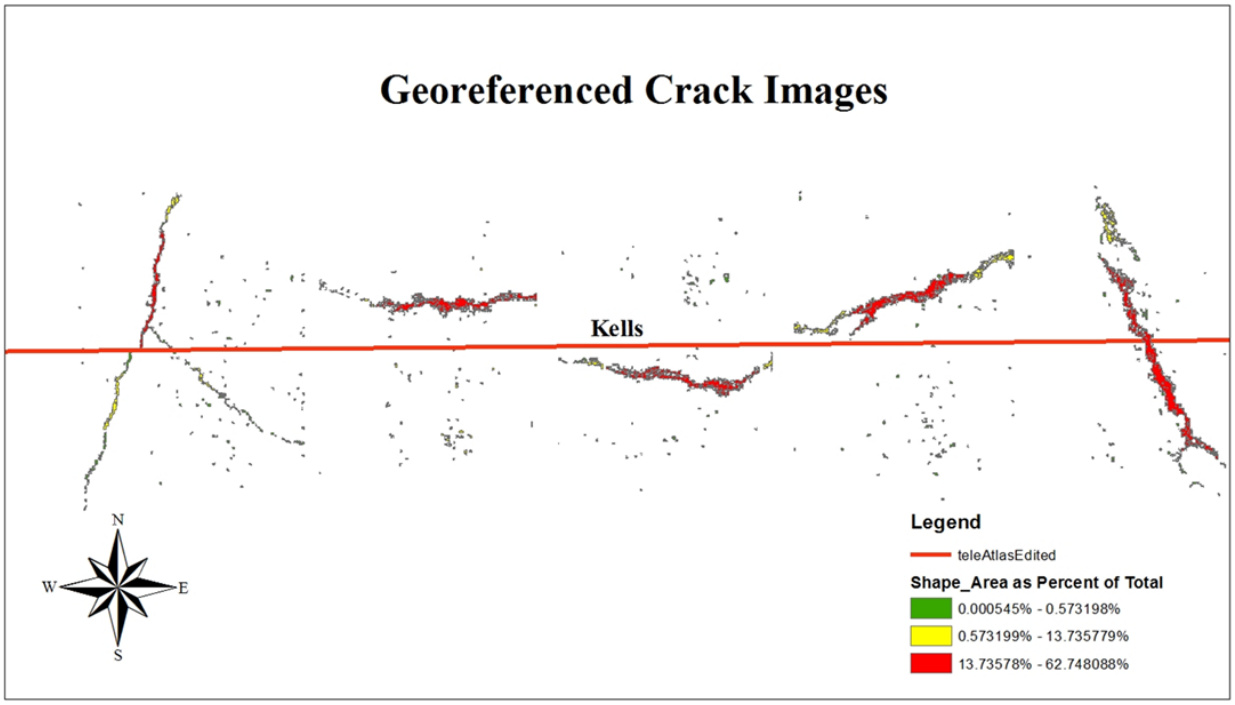


Figure 4.17: Automatically georeferenced images in ArcGIS.

Key Assumptions and Important notes:

- First image length starts from the first vertex of the road.
- Image acquisition is continued from first given image
- *Control Points*: During the network identification stage, four control points were obtained for each street. This will be all that is required for geo-referencing all the images on that road.
- *Image Size*: The image sizes for each road must be the same.

4.3.Case Study

The complete vision system developed was used to assess the condition of sections of roads on the University of Delaware campus. The study area is displayed in figure 4.18 below:

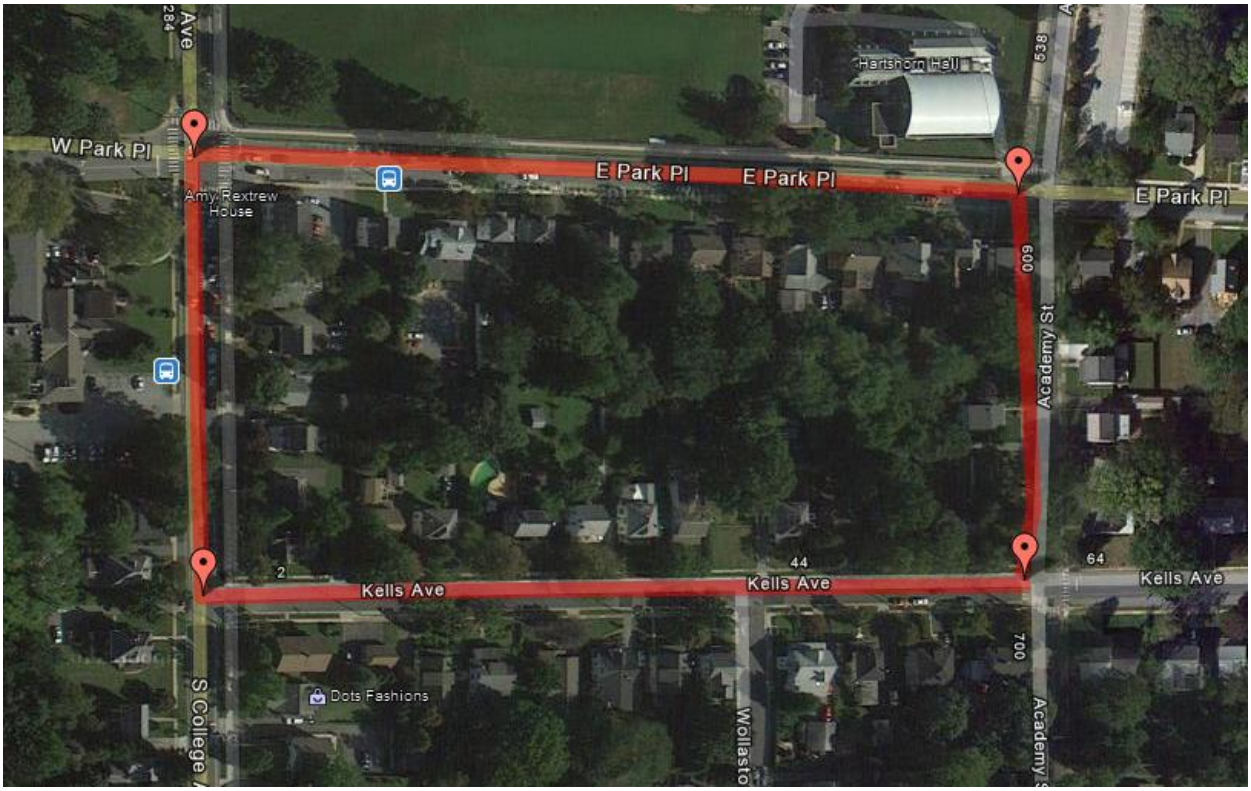


Figure 4.18: Map of Study Area

4.3.1. Database Development

- *Centerline data*: This database gives a spatial representation of the State road network. Variables such as road type (divided or not divided), name, length, width, median width, location (CBD or non-CBD), directions (one or two-way), elevation and traffic data (volumes and peak hour factor).
- *Pavement Condition data:* Still images were acquired instead of video images. The reason is due to the lack of an automobile equipped with an imaging system, so image acquisition was done manually. Consequently, the stream of images may not be continuous; however, considerable effort was made to capture most important distresses on the roads. In order to efficiently manage the vast number of pictures acquired, segment IDs were assigned based on picture names.

NB: The resolution of the camera determines the quality of survey and the minimum crack size that can be analyzed. For example, if an image with 640 x 480 pixels covers an area of 64 x 48 inches, then the survey represents an inch with 10 pixels or 2.5mm per pixel.

A graphic user interface (GUI) (shown in figure 4.19) application was developed in MATLAB for the implementation of the vision system developed. Sample outputs information from the GUI is shown in figures 6.20, 6.21 and 6.22. The following section will discuss and analyze the results obtained from the vision system.

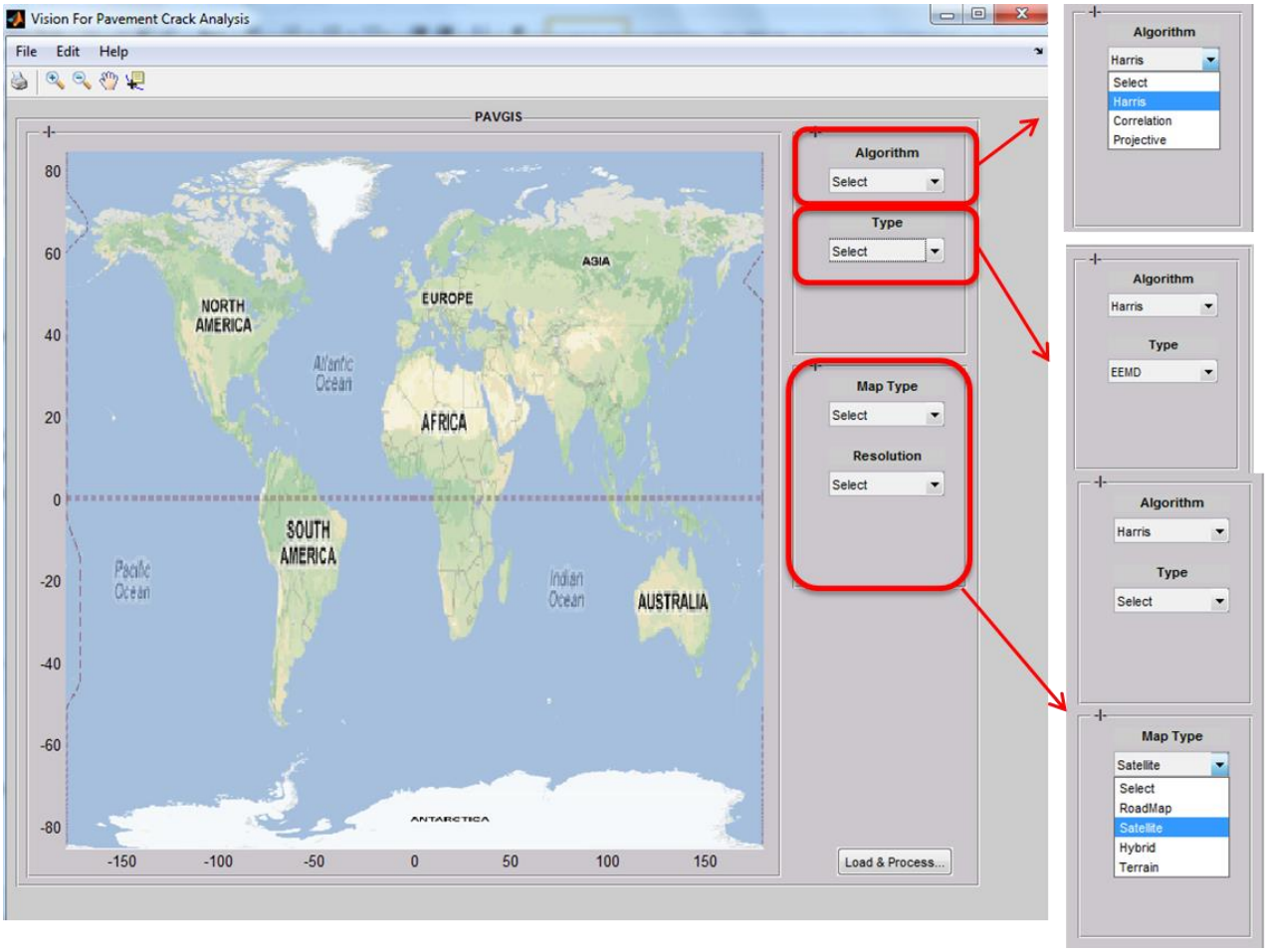


Figure 4.19: Graphic user interface developed for case study

4.3.2. Pavement Condition Rating and Analysis System

The condition of each road segment is rated based on the percentage of road surface cracked and the type of crack.

4.3.2.1.Damage Ratings:

A ratio of the area of cracks to the total image area is used to assess the distress condition of roads (rating index) and defined as:

$$PDI = \sum_{\forall i} \frac{(L_i * W_i)}{A_o}$$

where L_i is the length of the crack, W_i is the width of the crack and A_o is the area of the image.

4.3.2.2.Crack Type Classification:

The process of crack classification has varying degrees of complexity depending on the type of crack. For a simple transverse or longitudinal crack, a very simple algorithm based on pixel manipulations can be used to classify that image. However, classification can become much more complex. This has resulted in the development of very complex algorithms which are computationally expensive. In this system, crack classification is simpler because the coordinates of the cracks are known; hence their direction can easily be estimated. Also, if there is more than one crack in an image, conventional crack classification algorithms will fail. However, due to the vectorization process in GIS, different types of cracks can be detected.

Different types of cracks are assigned different ratings in order to calculate the overall condition rating for the road segment being analyzed. Table 4.5 provides a list of ratings from the most popular crack types. The pavement condition rating is a product of the crack type rating and the PDI. Figure 4.20 shows a map visualization of degree of damages of the pavement under study based on the PDI and crack type classification.

Table 4.5: Assignment of defect rating

Crack Type	Rating
Longitudinal Crack	0.10-0.35
Transverse	0.10-0.35
Pot holes	0.30-0.5
Pattern (block and alligator)	0.3-0.6

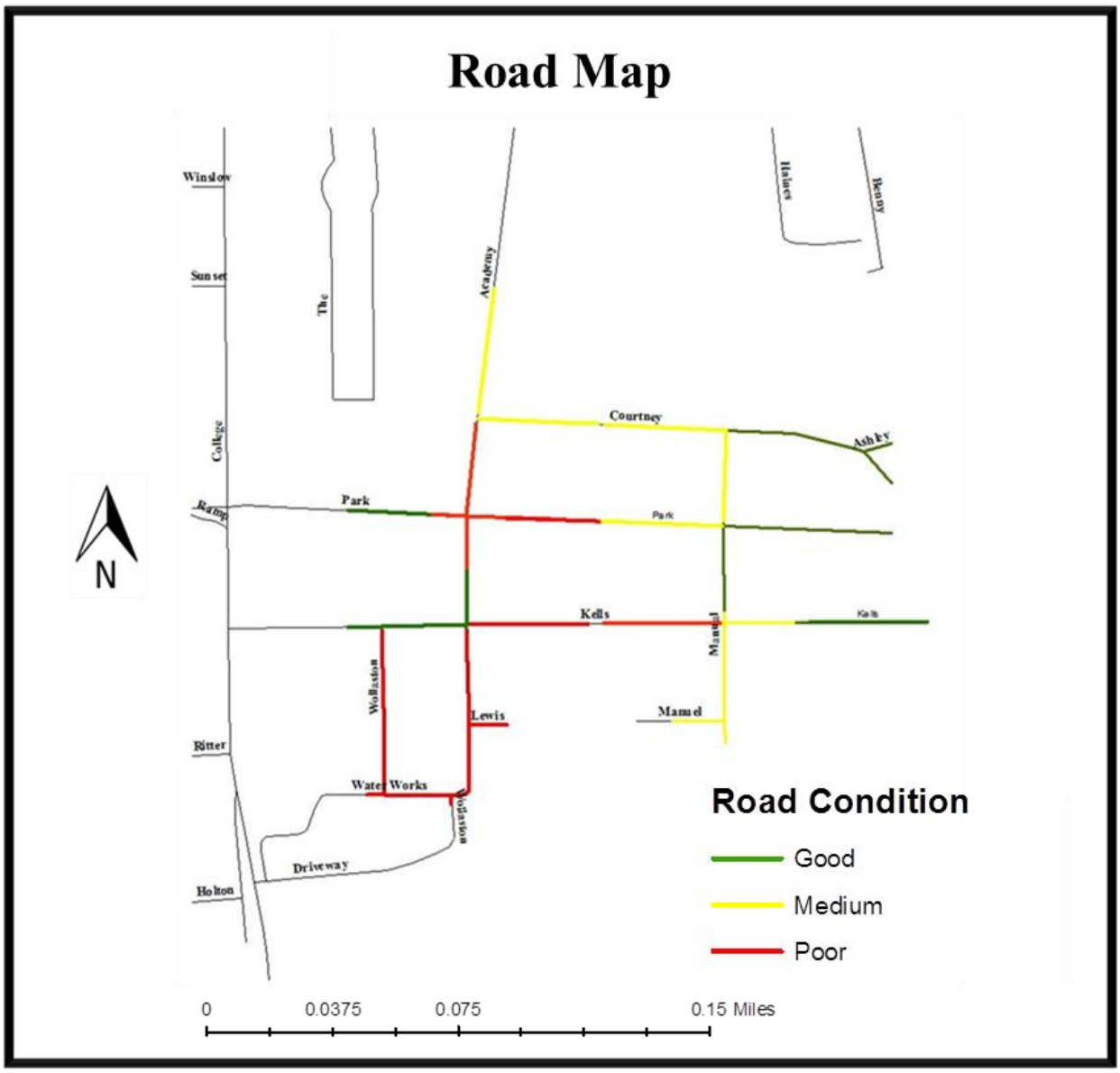


Figure 4.20: Visualization of road condition ratings

Also, in figure 4.21, a plot of average cracking per 0.05 mile is shown. This is helpful in describing the concentration of cracks along the road segment.

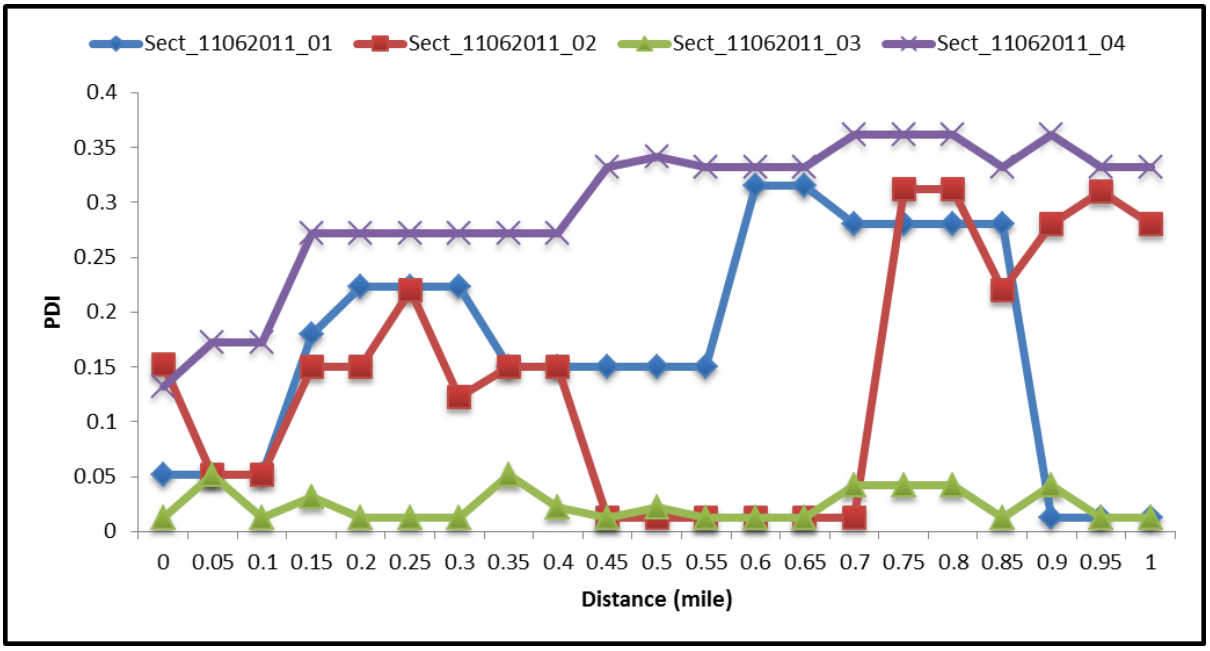


Figure 4.21: Average PDI per 0.05 of a mile

Table 4.6: Average PDI values for road sections

Section ID	**Average PDI per quarter mile**
Sect_11062011_01	0.168
Sect_11062011_02	0.142
Sect_11062011_03	0.023
Sect_11062011_04	0.296

4.3.3. **Maintenance Prioritization**

Maintenance prioritization groups identified damaged sections based on average traffic volume. This will help pavement managers to make decisions and request for appropriate interventions in the face of budget constraints.

$$P_r \propto (AADT, CIR\ and\ CTr)$$

where P_r is the section priority; CIR is the percentage of pavement image cracked. CTr is the rating assigned based on crack type (longitudinal, transverse, pothole, pattern cracks) and $AADT$ is the annual average daily traffic volume.

In figure 4.22, sections requiring immediate interventions are in red followed by yellow and green.

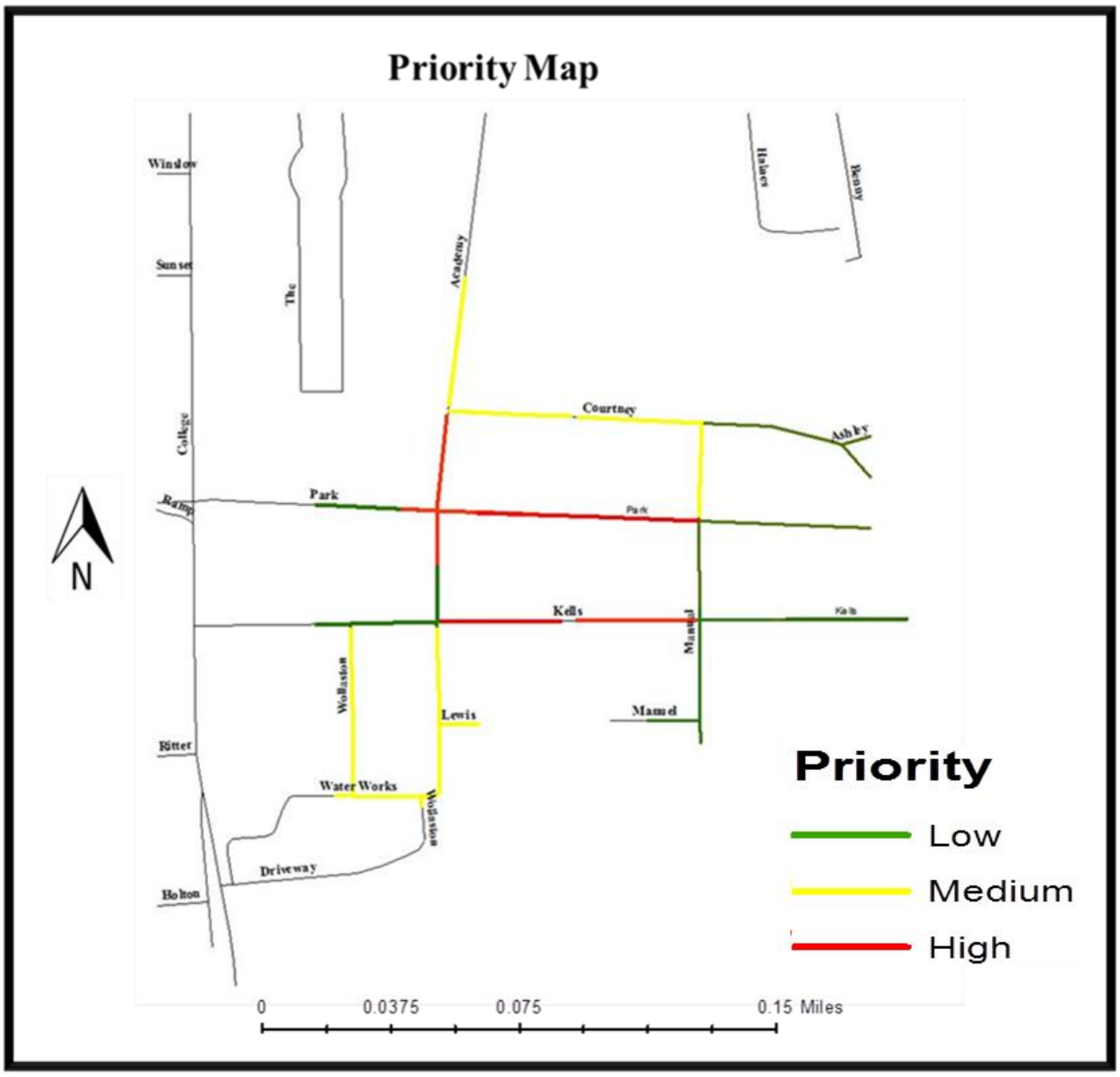


Figure 4.22: Maintenance priority maps.

4.3.4. **Summary of Invention**

In order to overcome the shortcomings of the state of the art, the current invention improves on automation of crack detection, assimilation (analysis) and visualization process. A framework for

developing adaptive algorithms with increased parallelism is also illustrated. Above all, the key components of the inventions are as follows:

- Crack Detection and analysis: Instead of using traditional image processing tools, in this invention, a bold attempt is made to use adaptive and data deriving methods, which outputs more accurate results. For the first time, active contour models are successfully used in place of edge detection methods. This provides much more continuous and smoother description for crack edges.
- Toward Real-Time Implementation: To improve the speed of the system, crack image retrieval is used to suppress non-cracked images from feature extraction. This saves the system significant time. Also, the invention provides a parallel implementation of computationally expensive algorithms, which dramatically reduces their computation time.

Ultimately, the pavement crack detection system must be:

Adaptive: Classical pavement distress detection systems usually extract desired features by convolving predefined basis functions with the original distress image. In this work, feature extraction is totally data driven through detection and interpolation of local extrema.

Partially Real-Time: Generating crack maps on-line while the image acquisition vehicle travels at 50mph. The results in this work cannot operate in real-time. The goal here is to explore the potential of a real-time application of the adaptive algorithm.

REFERENCES

Adu-Gyamfi Y.O., Attoh-Okine , N.O. Garateguy, Gonzalo., Carrillo Rafael and Gonzalo R. Arce, (2011) "Multi-resolution Information Mining for Pavement Crack Image Analysis," ASCE Journal of Computing.

Ayenu-Prah, A., and Attoh-Okine, N.O. (2008). "Evaluating Pavement Cracks with Bidimensional Empirical Mode Decomposition." *EURASIPJ. Adv. Signal Process.*, Volume 2008, Article ID 861701.

Candes, E., Li, X., Ma, Y., and Wright, J. (2009). "Robust principal component analysis?" http: // arxiv. org/ abs/ 0912. 3599.

Fereidoon Moghadas Nejad , Hamzeh Zakeri., "A comparison of multi-resolution methods for detection and isolation of pavement distress." *Expert Systems with Applications*, Volume 38, Issue 3, March 2011, pp. 2857–2872.

Kim, J. Y. (2009). "Development of New Automated Crack Measurement Algorithm using Laser Images of Pavement Surface." *PHD Dissertation,* 2009., University of Iowa.

Lee, H.D. (2005). "Development of a Manual Crack Quantification and Automated Crack Measurement System." Project TR-457, Public Center, Civil and Environmental Engineering, University of Iowa.

Matlab 2010, "Matlab Parallel Computing Toolbox": http://www.mathworks.com/products/parallel-computing/

Obaidat, Mohammed Taleb., Al-kheder , Sharaf A., (2006).,"Integration of geographic information systems and computer vision systems for pavement distress classification," Construction and Building MaterialsVolume 20, Issue 9, pp. 657–672.

Smith, William., (2010). "Matlab's Parallel Computation Toolbox And the Parallel Interpolation of Commodity Futures Curves"

http://commoditymodels.files.wordpress.com/2010/03/matlab-parallel- computing-toolbox-and-interpolating-futures-curves.pdf

Subirats, P.; Dumoulin, J.; Legeay, V.; Barba, D.; , "Automation of Pavement Surface Crack Detection using the Continuous Wavelet Transform," *Image Processing, 2006 IEEE International Conference on*, vol., no., pp. 3037-3040, 8-11 Oct. 2006.

Wang, Kelvin C. P., (2005), "Real-Time Automated Survey System of Pavement Cracking in Parallel Environment." *Journal of Infrastructure Systems*, Vol. 11, No. 3, pp. 154-164.

Wang, Kelvin C.P., Xuyang Li, and Gong, Weiguo., (2001) "An Integrated Information System for Highway Asset Management." 5th International Conference on Managing Pavements.

Chapter 5

CONCLUSIONS AND RECOMMENDATIONS

5.0. Introduction

This chapter provides a summary of the key findings and developments made with regards to the development and application of the multiresolution information mining and vision systems to pavement distress detection and analysis of condition information. Additionally, key challenges and their appropriate recommendations will be outlined to serve as a guide for future studies.

5.1. **Crack Detection**

Automated pavement crack detection has undoubtedly improved the pavement road condition assessment process. BEMD (Bidimensional Empirical Mode Decomposition) was introduced to improve the adaptation of current crack detection systems to different types of distresses. Decomposed results from BEMD were revealing, and separated cracks into different components based on their widths and intensity. In order to separate all the cracks from background information, an improved IMF reconstruction technique was developed. Although the results were better than traditional reconstruction techniques, the method was too slow and the discrimination between cracks and their background was not very clear. This resulted in combining BEMD with a multiresolution information mining algorithm known as Principal Component Pursuit (PCP) to develop straightforward, adaptive and data driven procedures to extract salient information from pavement distress images. Using this approach, image enhancement, denoising and reconstruction can be performed simultaneously without any a

priori assumptions. The fidelity of segmented images is very high in terms of the width and the direction of the cracks.

To enhance the practical application of BEMD, a real-time implementation of the algorithm was investigated through image subsampling and parallel processing. It was realized that extrema detection and interpolation alone contributed to about 80% of the computational time. Fortunately, these procedures are embarrassingly parallel. Hence, codes were restructured to take advantage of current generation of CPUs. Incredible improvement in speed is achieved during parallel extrema detection and interpolation.

The last contribution in this area is the introduction of a vision system into the crack detection process. As explained earlier, this development endows computers with information-processing capabilities enabling them to model and automate the process of crack detection in a way comparable to those of biological organisms. It involved the introduction of automatic crack image retrieval system for pre-screening the database, thus selecting only cracked images for processing. The main aim of the image retrieval system is to improve the speed of the system. Also, image denoising algorithm based on Gaussian-pyramid was explored.

In the area of edge detection, a model-based approach known as active contours or snakes are introduced. It is dependent on the distribution of high and low gradient points across the image rather than just the presence of a high gradient points; this is a concept that is predominantly used by traditional edge detection systems. The model-based approach therefore has a good perception of a crack edge. It is able to produce a continuous and smooth delineation of cracks in the image, detect its exact location and produce accurate geometric parameters of the crack boundary.

5.1.1. **Complete Pavement Management System**

A number of pavement management systems exist for assessing conditions, prioritizing and scheduling maintenance routines for road networks. Most often, condition assessments or ratings and database management are carried out on different platforms. As such data verification by decision makers can be very difficult. Much human intervention will then be required, which may reduce the system's efficiency and consistency. Recent works which seek to integrate these systems from different platforms have not been very successful as they are not fully automatic. The development carried out in this work is fully automatic; routines such as geo-referencing, vectorization of raster images, road condition visualization and prioritization are all performed without any interventions or manual editing procedures from the user.

The integrated system outputs a map visualization of the road condition based on the percentage of image area cracked. An important procedure to this step is the automated georeferencing of images that was introduced. This ensures that we have the accurate location of where the image was taken. From vectorized images, the type of crack in the image can be classified since each crack has its corresponding coordinates.

5.2.Recommendations for Future Research

In spite of the various contributions made in this study, there is still room for improvement in others to extend the methodology described to wider application areas.

5.2.1. **Crack Detection and Classification**

Recent advances in video imaging have improved the quality of images acquired by acquisition vehicles. Therefore, image enhancement algorithms may not be extremely important. The key challenge introduced by these advancements is that the current generation of video cameras reveals too much information which may not be helpful in analysis. This could affect the efficiency of edge detection algorithms. In the past, video cameras might not have been able to discern small objects or fine texture information, etc. hence, newer video equipment may not be hindered by rough textures and other small foreign objects on the road. Future algorithms should be geared toward intelligent feature extraction in the midst of the cloud of information in distress images.

5.2.2. **Robust Condition Monitoring**

One step that is conspicuously missing in this study is the analysis of crack depth information. This work assumes that there is a correlation between the width of cracks and its depth. This is not necessarily true. Pavements may look good on the surface however, due to water dissipating through small cracks into the core of the pavement; there could be cracks within the sub-base. There are two main ways to approach this: first, after the distress crack detection algorithm is run on the pavement, the pavement structural strength could be obtained using deflectographs or ground penetrating radars to assess what exactly is happening within the structure. A second technique could be developing devices that can simultaneously sample surface and depth information. In essence each pixel will have its relative depth. 3D algorithms will therefore be required to process such information. Although such advancement will be very hopeful, it may however increase computational time.

5.2.3. **Platform Speed and Database Management**

The current implementation of BEMD for crack detection is not real time. Usually, the crack detection algorithm is expected to assess the condition of the road while the vehicle acquires data at a travelling speed of approximately 50 to 60 mph. This will require the algorithm to be extremely fast. The current generation of computer memory and CPUs cannot provide this speed-up. Future research should explore the possibility of using GPU computing and MEX functions to improve the time for computations carried out on Global memory. MATLAB also slows computations since the programmer is unable to control memory allocations; therefore, a more efficient and fast programming language like Visual Studio C++ should be explored. Another area that requires attention is the platform for database management. In the present study, ESRI's ArcGIS is used for managing distress information. The software is very expensive so cheaper versions of GIS software could be explored.