

IMAGE-BASED “D”-CRACK DETECTION IN PAVEMENTS

by

ALLISON DAY

B.S., Kansas State University, 2007

A THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Electrical and Computer Engineering
College of Engineering

KANSAS STATE UNIVERSITY

Manhattan, Kansas

2011

Approved by:

Major Professor
Balasubramaniam Natarajan

Copyright

Allison Day

2011

Abstract

This thesis proposes an automated crack detection and classification algorithm to detect durability cracking (“D”-cracking) in pavement by using image processing and pattern recognition techniques. For the Departments of Transportation across the country, efficient and effective crack detection is vital to maintaining quality roadways. Manual inspection of roadways is tedious and cumbersome. Previous research has focus on distinct transverse and longitudinal cracks. However, “D”-cracking presents a unique challenge since the cracks are fine and have a distinctive shape surrounding the intersection of the transverse and longitudinal joints. This thesis presents an automated crack detection and classification system using several known image processing techniques.

The algorithm consists of four sections: 1) lighting correction, 2) subimage processing, 3) postprocessing and 4) classification. Some images contain uneven lighting, which are corrected based on a model of the lighting system. The region of interest is identified by locating the lateral joints. These regions are then divided into overlapping subimages, which are then divided into cracked and noncracked pixels using thresholds on the residual error. Postprocessing includes a row/column sum filter and morphological open operation to reduce noise. Finally, metrics are calculated from the final crack map to classify each section as cracked or noncracked using the Mahalanobis distance from the noncracked distribution.

Table of Contents

| | |
|---|-------------|
| Table of Contents | iv |
| List of Figures | vi |
| List of Tables | viii |
| Acknowledgements | ix |
| 1 Introduction | 1 |
| 1.1 Background | 1 |
| 1.2 Motivation for Research | 2 |
| 1.3 Key Contributions | 3 |
| 1.4 Organization | 4 |
| 2 Pavement Distress | 6 |
| 2.1 Cracking Phenomena | 6 |
| 2.2 Durability Cracking | 8 |
| 2.3 Measurement System | 9 |
| 2.4 Summary | 11 |
| 3 Lighting Correction | 13 |
| 3.1 Lighting System | 13 |
| 3.2 Histogram Equalization | 14 |
| 3.3 Column Average Scaling | 15 |
| 3.4 Model-based Lighting Correction | 17 |
| 3.4.1 Light System Model | 17 |
| 3.4.2 Global Model Correction | 19 |
| 3.4.3 Single Light Model Correction | 19 |
| 3.5 Summary | 22 |
| 4 Crack Detection | 24 |
| 4.1 Region of Interest | 24 |
| 4.2 Artifact Identification | 24 |
| 4.3 Subimage Processing | 25 |
| 4.3.1 Residual Error | 26 |
| 4.3.2 Probabilistic Relaxation | 28 |
| 4.4 Connectivity | 31 |
| 4.5 Summary | 32 |

| | | |
|----------|--|-----------|
| 5 | Postprocessing and Classification | 33 |
| 5.1 | Postprocessing | 33 |
| 5.2 | Classification | 34 |
| 5.3 | Summary | 39 |
| 6 | Visualization and Results | 40 |
| 6.1 | Visualization | 40 |
| 6.2 | Large-scale Testing | 41 |
| 6.3 | Summary | 43 |
| 7 | Conclusion | 44 |
| 7.1 | Summary of Key Contributions | 44 |
| 7.2 | Future Work | 45 |
| | Bibliography | 46 |

List of Figures

| | | |
|------|--|----|
| 1.1 | Flowchart of Proposed Algorithm | 4 |
| 2.1 | Examples of Longitudinal and Transverse Cracks | 7 |
| | (a) Longitudinal Crack | 7 |
| | (b) Transverse Crack | 7 |
| 2.2 | Example of Block Cracking | 7 |
| 2.3 | Example of Fatigue Cracking | 8 |
| 2.4 | Example of a Corner Break | 9 |
| 2.5 | Example of High Severity “D”-cracking | 10 |
| 2.6 | Example of Low Severity “D”-cracking | 10 |
| 2.7 | ICC Imaging Vehicle | 11 |
| 3.1 | Original image | 14 |
| 3.2 | Relative light positions on imaging vehicle | 15 |
| 3.3 | Histogram equalization | 16 |
| 3.4 | Resulting image after column average scaling | 18 |
| 3.5 | Column vs column average | 19 |
| 3.6 | Resulting image from global model correction | 20 |
| 3.7 | Column mean with section means | 21 |
| 3.8 | Final image after light correction | 22 |
| 4.1 | Row sums | 25 |
| 4.2 | Column averages for image section with joint | 26 |
| 4.3 | Subimage of a crack | 27 |
| | (a) Original subimage | 27 |
| | (b) Residual error | 27 |
| 4.4 | Subimage After Thresholding | 27 |
| 4.5 | Probabilistic setup | 28 |
| 4.6 | Initial probabilities | 29 |
| 4.7 | Probabilities after five iterations | 30 |
| 4.8 | Final crack map for subimage | 30 |
| 4.9 | Probabilistic relaxation of subimage | 30 |
| | (a) Original subimage | 30 |
| | (b) Result | 30 |
| 4.10 | 20-Connectivity | 31 |
| 4.11 | Subimage after connectivity applied | 31 |
| 5.1 | Structure Elements | 34 |

| | | |
|-----|--|----|
| (a) | 20° | 34 |
| (b) | 45° | 34 |
| (c) | 70° | 34 |
| 5.2 | Crack Map After Postprocessing | 35 |
| 5.3 | Quadrants of region to be analyzed | 35 |
| 5.4 | Cracks along 45° line | 36 |
| 5.5 | Angles of crack segments | 36 |
| 5.6 | Metric histograms | 37 |
| (a) | Number of cracks on 45° line | 37 |
| (b) | Percent cracked | 37 |
| (c) | Angles of crack segments | 37 |
| (d) | Average crack size | 37 |
| 5.7 | Percent cracked vs. average set size | 38 |
| 5.8 | Thresholds for classification | 39 |
| 6.1 | Final Crack Map | 41 |
| 6.2 | Probability of missed crack and false alarm for varying thresholds | 42 |

List of Tables

| | | |
|-----|--------------------------------------|----|
| 6.1 | Classification of test set | 41 |
| 6.2 | Performance Statistics | 42 |

Acknowledgments

This research was funded in part by the Kansas Department of Transportation (KDOT). We gratefully acknowledge KDOT's support and guidance.

Chapter 1

Introduction

This chapter introduces the problem of durability cracking (“D”-cracking), the motivation for this research, and the key contributions of this thesis.

1.1 Background

The Departments of Transportation (DOT) across the country are tasked with monitoring the roadways for wear and distress. If untreated, pavement distress can cause major problems, which in turn affect ride quality as well as the structural integrity of the roadway. DOTs have employed various strategies to monitor pavement conditions in a timely and efficient manner. In the past, manual inspection was used to analyze sections of highway. This tedious process required trained experts to walk or drive sections of highway. The data collected was then entered into a computer log. Overall this process was time consuming and difficult, allowing only a limited amount of highway to be analyzed. Many DOTs have moved to using vehicles with imaging equipment to record pavement condition. A human operator then analyzes these images by visual observation to determine if there is a defect. One-hundred miles of highway can produce more than 50,000 images, making manual analysis time consuming and in many cases impractical. Automated and semi-automated image processing methods have been used to assist in analyzing data in an efficient manner.

Pavement distresses cover a broad range of effects that can occur and affect ride quality and structural integrity of the pavement. These include cracking, joint deficiencies, surface

defects and miscellaneous distresses. The focus of this thesis is a specific type of crack known as durability cracking, or “D”-cracking. The Federal Highway Administration defines these type of phenomena in the Distress Identification Manual [6]. A detailed definition of “D”-cracking is provided in chapter 2. The next section discusses the previous research done in the area of crack detection and the motivation for this thesis.

1.2 Motivation for Research

To improve the detection process, automated detection systems are necessary for efficient and effective management of pavement distress. Thus many researchers have focused on developing image processing algorithms that require little to no human interaction. Most of this research has been on longitudinal and transverse cracks.

For example, [9] focused specifically on an automated system to detect transverse and longitudinal cracking using Principal Component Analysis to connect areas of dark pixels. A similar thresholding technique was used in [12]. Here bilevel thresholding was used to spatially connect very dark pixels with less dark but nearby pixels. To classify images, the thresholded image was projected onto different axes. While this works well for large distinct cracks, “D”-cracks can be only a few pixels wide and vary in angle.

Many transformations have also been used to detect longitudinal and transverse cracks. Reference [4] uses a radon transformation to detect the number and location of cracks in an images. However, from the test images shown it is unclear if this would work on smaller, less distinct cracks. Reference [11] uses a continuous wavelet transform to segment the image and create a binary image of the cracked pixels. This technique does not provide a method of reducing noise and could miss small cracks with the amount of noise present.

After detection, many different methods of classification have been used. Neural networks have been used extensively for classification in crack detection [1] [10]. Reference [10] used a neural network with 7 inputs and 4 outputs, and was able to detect transverse, longitudinal, and fatigue cracks. Researchers at Utah State University also used a back-

propagation neural network for classification in [1]. After detecting cracks using fuzzy logic, the neural network uses 18 input nodes and has 7 outputs, which correspond to 7 types of cracking that can be detected. The test set of 42 images does not provide a large enough sample for a reliable classifier. In general, the performance of a neural network is dependent on the quality of the training data. If the test data deviates from the training data, performance will be poor.

While these phenomena present similar challenges, “D”-cracking is a specialized problem that consists of very fine cracks concentrated around the intersections of concrete pavement joints. Very little research has been done to date on the detection and classification of this type of cracking, so it is still an open problem. The next section describes the key contributions of this thesis and introduces the algorithm used to solve the problem.

1.3 Key Contributions

In this thesis, we propose a novel image processing algorithm for automated detection of “D”-cracking. The goal of the algorithm is to detect “D” type cracks in pavement images, and ultimately classifying images into cracked and noncracked. The final output is a crack map for visual inspection in analysis of pavement condition. The algorithm can be described in four stages as shown in figure 1.1: 1) lighting correction, 2) crack detection, 3) postprocessing, and 4) classification. Each image is processed through these stages individually. Overall, the process will reduce the time spend combing through thousands of images of pavement.

Below summarizes the key contribution of this thesis:

- The lighting correction outlined in chapter 3 is adaptable to various lighting effects and conditions. It also corrects for lighting without destroying information.
- The use of the residual error as defined in section 4.3.1 is useful for identifying changes in intensity and accounts for changes in background intensity.

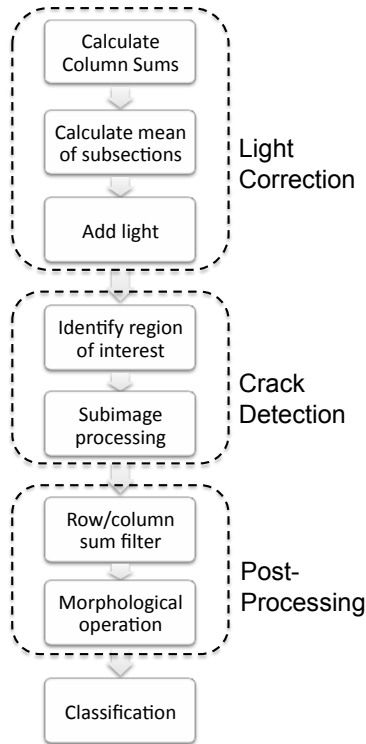


Figure 1.1: Flowchart of Proposed Algorithm

- In section 4.4, connectivity is used to eliminate isolated pixels.
- Noise from isolated dark pixels due to image artifacts are reduced using morphological operations.
- Section 5.2 shows the metrics used to classify images as cracked or noncracked. The Mahalanobis distance is used to determine the classification, and can also be used to determine the severity of “D”-cracking.
- In large scale testing, the algorithm has a 67% probability for correct classification of images as cracked and noncracked, and has a probability of 10.9% for missed cracks.

1.4 Organization

This thesis is organized as follows: Chapter 2 introduces the various cracking phenomena including “D”-cracking and the measurement system; Chapter 3 describes methods used

to correct for lighting artifacts in an image; Chapter 4 presents the approach for crack detection including identifying the region of interest; Chapter 5 describes the postprocessing techniques used to reduce noise and classification of images; Chapter 6 is the results from large scale testing; and finally Chapter 7 is the conclusion of the thesis including key contributions and future research.

Chapter 2

Pavement Distress

This chapter discusses the types of cracking that can occur in pavement, specifically “D”-cracking and its characteristics, and the measurement system used to collect pavement images. The types of pavement distresses that can occur vary with pavement type. In the Federal Highway Administration’s Distress Identification Manual [6], they cover distresses found in asphalt, jointed portland cement, and continuously reinforced concrete surfaces. There are many different distress phenomena including: cracking, patching and potholes, joint deficiencies, surface deformations, surface defects, and other miscellaneous defects. In this thesis we are primarily concerned with cracking, specifically “D”-cracking.

2.1 Cracking Phenomena

While there are a range of pavement distresses that can be found, our focus is primarily cracking. This section describes the types of cracks found in the three types of pavement: asphalt, jointed portland cement and continuously reinforced concrete.

Longitudinal and transverse cracks are found in all three pavement types. Longitudinal cracks are predominantly parallel to the pavement centerline, such as the one shown in figure 2.1a [5]. These cracks are typically caused by improper construction techniques or a poorly paved joint. Similar in characteristics to longitudinal cracks, transverse cracks run perpendicular to the pavement centerline. A transverse crack is shown in figure 2.1b [7]. The cause of these cracks is shrinkage from temperature changes or hardening of the asphalt,

which makes them much more common than longitudinal cracks. In both cases, however, severity is determined by the width of the crack. Cracks are then categorized by the highest severity present for at least 10 percent of the crack.

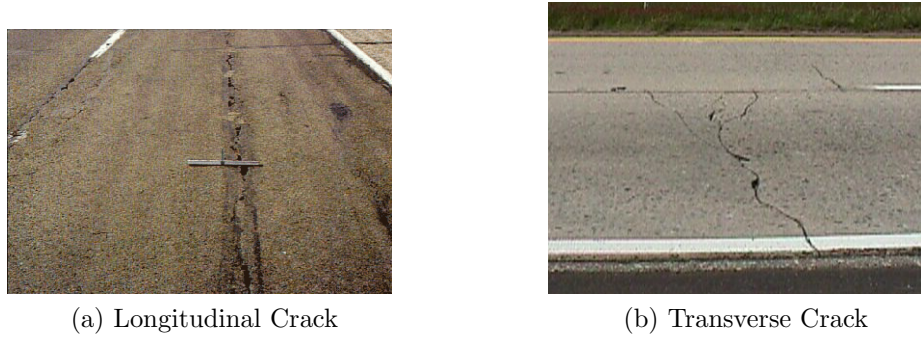


Figure 2.1: Examples of Longitudinal and Transverse Cracks

Block cracks consist of networks of longitudinal and transverse cracks which create rectangular sections, which is caused by shrinking of pavement due to temperature changes. Here severity is measured by the width of the crack as well as by the amount of secondary cracking surrounding each transverse/longitudinal crack. Figure 2.2 shows an example of block cracking in asphalt [5].



Figure 2.2: Example of Block Cracking

Another common type of crack is fatigue cracking. Since these cracks form a pattern resembling alligator hide, these cracks are also referred to as alligator cracks. Fatigue cracks are generally located along the wheel path where heavy loads are repeatedly impacting the

surface. The impacts can cause a weakening of the support structure beneath the pavement, which in turn causes the surface to deteriorate. They can also form as secondary cracks around longitudinal/transverse cracks. An example is shown in figure 2.3.



Figure 2.3: Example of Fatigue Cracking

In jointed portland cement concrete, corner breaks occur when a crack intersects both the transverse and longitudinal joints at approximately a 45 degree angle and separates the corner of a slab shown in figure 2.4. This is usually due to poor load transfer across the joint. Since “D”-cracking also occurs at the intersection of joints, this could pose difficulties when trying to differentiate the two types. The main difference is that “D”-cracks consist of multiple small, closely spaced cracks, whereas corner breaks are large single cracks. The next section discusses “D”-cracking in further detail as it is the main focus of this thesis.

2.2 Durability Cracking

Durability cracking (“D”-cracking) is a difficult type of cracking to identify, especially in the early stages. It begins as small crescent shaped cracks near the intersection of the transverse and longitudinal joints. An example of a high severity case is shown in figure 2.5. “D”-cracking begins when moisture seeps into the aggregate particles of the concrete, then repeated freezing and thawing of the pavement begins to cause structural damage. As



Figure 2.4: Example of a Corner Break

it progresses small cracks begin to form parallel to the transverse joint [2]. The severity is determined by the number of cracks as well as how tightly spaced they are. This can be seen in the low severity example in figure 2.6, where there are multiple cracks but they do not propagate far out from the joint intersection. “D”-cracking is only found in concrete pavements, and is more likely to occur in certain types of aggregates. Certain states are particularly susceptible to “D”-cracking due to the types of aggregates that are available in that area, as well as wet freeze-thaw cycles. Because of this, Kansas highways are particularly prone to “D”-cracking.

The process of “D”-crack detection is complex due to the nature of these cracks. Since “D”-cracking is specifically located at the joint intersection, processing can be limited to the region immediately surrounding the longitudinal joints. To classify these cracks, several criteria should be taken into consideration such as size, shape and location. The metrics that were used to classify images are detailed in section 5.2. The next section discusses the details of the measurement system used to identify these cracks.

2.3 Measurement System

Since the hairline cracks seen in “D”-cracking are so fine, the measurement system plays a key role in the analysis since factors such as lighting and camera position can affect the

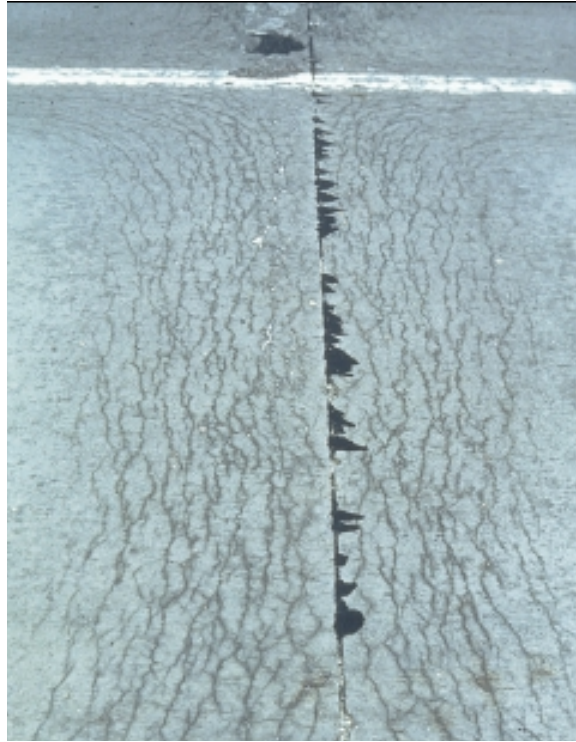


Figure 2.5: Example of High Severity “D”-cracking

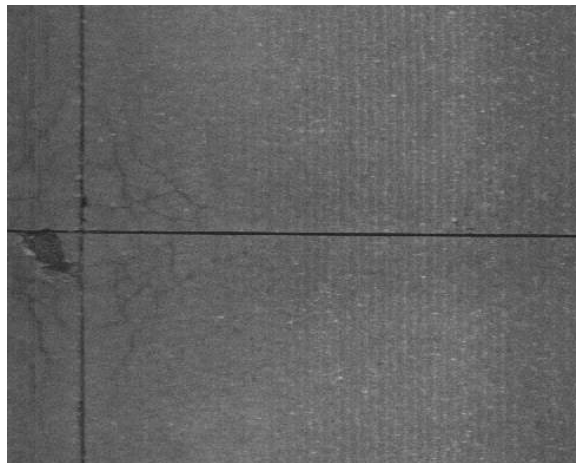


Figure 2.6: Example of Low Severity “D”-cracking

images collected. The artifacts that may be produced must be accounted for in image preprocessing. This section briefly describes the vehicle used to collect pavement images.

The measurement system consists of several components contained within a vehicle designed to be driven at highway speeds. KDOT owns such a vehicle made by International

Cybernetics Corporation (ICC), which was designed to record real-time high-resolution pavement images in various lighting conditions and at speeds up to 70 mph. The components of the system include: a pavement camera system, rack mounted computers, laser sensors, pavement lighting systems, and a Global Positioning System (GPS). The pavement camera system consists of a high-resolution Basler L103 line-scan cameras focused downward on the road surface. These images are used in our analysis. To run the system, the computers use a 3.0 GHz Pentium IV processors with Windows 2000. Also, pavement elevation data is recorded by laser elevation sensors mounted on the front bumper. All the data is recorded to removable hard drives so that it is easily accessible for further analysis. The lighting system, which will be described in greater detail in section 3.1, consists of ten 150-watt lamps mounted to the rear of the vehicle.



Figure 2.7: ICC Imaging Vehicle

2.4 Summary

This chapter described many of the types of crack phenomena that can be found in pavements, from longitudinal and transverse cracks to “D”-cracking. Since the focus of this

thesis is the detection of “D”-cracking, how these cracks develop and can be detected was described in detail. Also the measurement system was introduced. In the next chapter, the lighting correction algorithm is presented.

Chapter 3

Lighting Correction

Although the imaging system used by KDOT includes a lighting system that is intended to prevent the effects of shadows and low ambient light, in certain situations it is not sufficient. In this chapter, the lighting correction portion of the algorithm is explained. In addition to ambient lighting, wear also affects the reflectivity of the pavement causing portions of pavement to appear darker, as shown in figure 3.1. All of these differences in lighting make it difficult to detect “D”-cracks. Since our goal is not only to detect cracks but also create a crack map for visual inspection, our light correction must improve the appearance of the image to the human eye.

3.1 Lighting System

As briefly described in section 2.3, the imaging vehicle lighting system consists of ten 150-watt lamps. Six lamps are centered on the rear of the vehicle, equally spaced and focused directly toward the pavement, with two on each side angled outward to help light the edges of the pavement. The relative positions as measured from the center of the lamps are shown in figure 3.2. Although the light is dispersed using lenses, the intensity peaks at the center focal point of each lamp. In low light conditions this can create uneven lighting in images. Several approaches were tried with mixed results as explained in the following section.

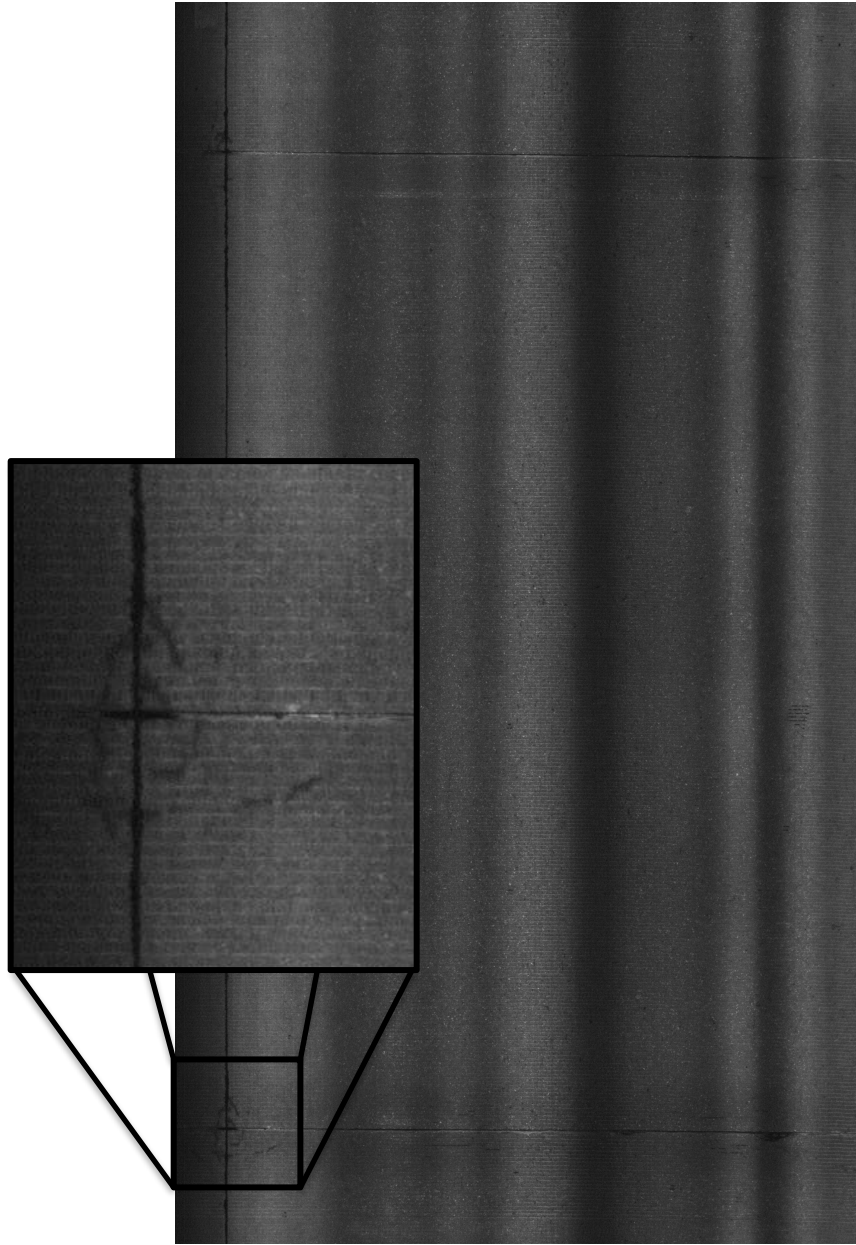


Figure 3.1: Original image

3.2 Histogram Equalization

Uneven lighting has been a common problem in crack detection. For example, [12] used histogram equalization to correct for uneven lighting. This approach is a well-known way to increase image contrast. Histogram equalization works by mapping the intensities of a

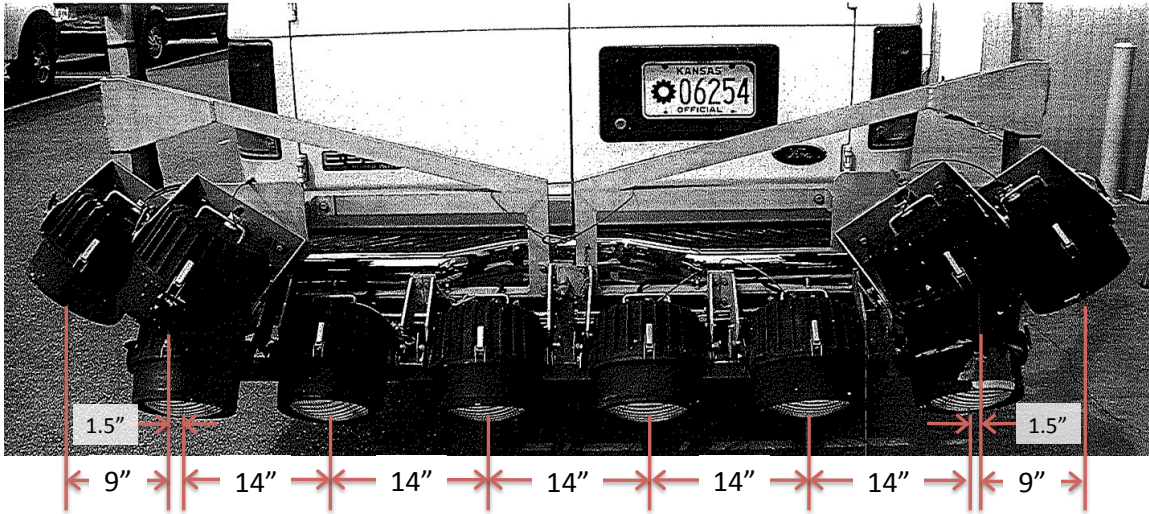


Figure 3.2: Relative light positions on imaging vehicle

grayscale image to a new distribution that is approximately uniform. By stretching the histogram, the contrast of the image is increased making the details of the image more visible. This is useful in many cases of unequal lighting since it works to equalize the distribution of pixels.

However in the case of “D”-cracking, the cracks are extremely fine and the difference in intensity is minimal. Histogram equalization can cause that information to be lost. Also it may accentuate any lighting artifacts in the image such as bright areas as shown in figure 3.3. The next section describes the next approach that was tried using column sums to detect trends in the lighting.

3.3 Column Average Scaling

Histogram equalization proved to be too global a transformation and did not respond well to local lighting changes. Instead we wanted an approach that would adapt to the image lighting. Since the lighting along the image columns is consistent, the average of the columns can be used to detect trends in lighting.

After calculating the average values for each column, the mean of these values is calcu-

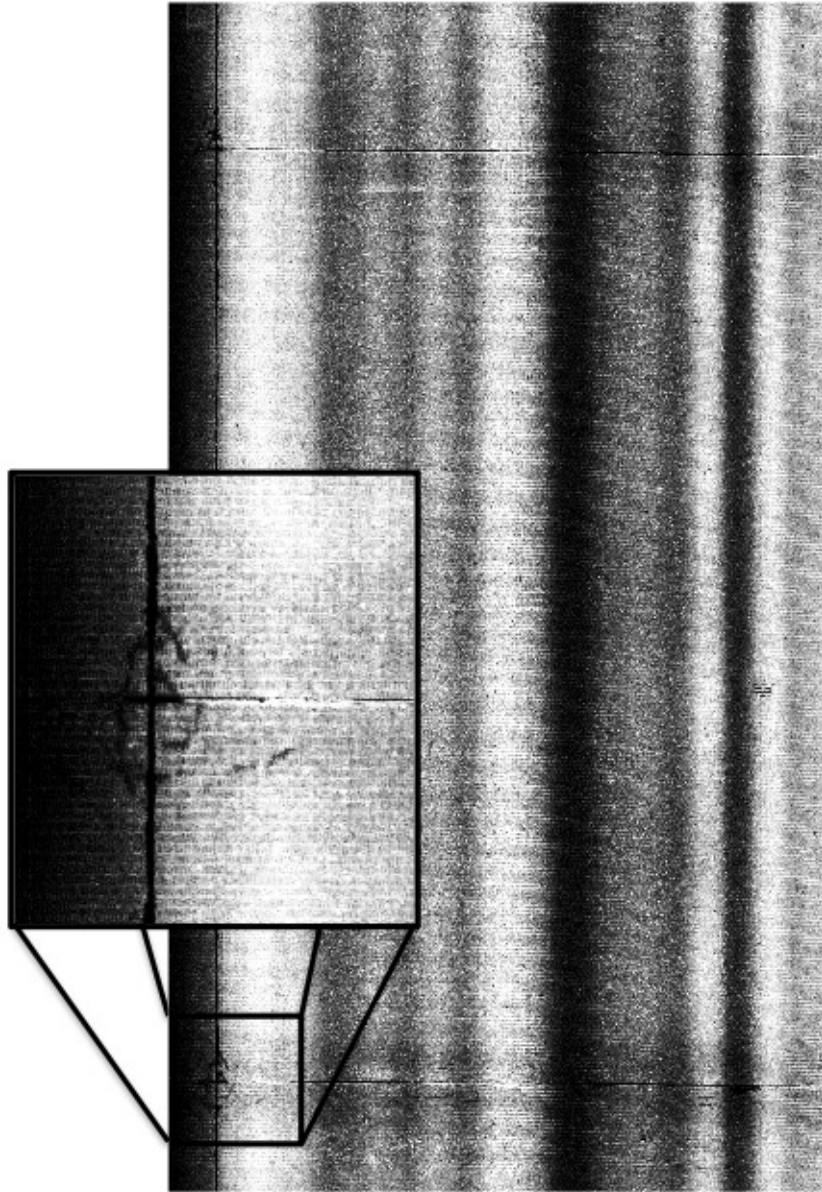


Figure 3.3: Histogram equalization

lated. This mean is the desired average for each column and gains are calculated based on the ratio between the global mean and the column mean as shown in equation 3.1, where $p(x, y)$ is the pixel value at (x, y) , M is the number of columns and N is the number of rows in the image. Each column is then multiplied by the corresponding gain, resulting in an equalized image. An example is shown in figure 3.4.

$$G_y = \frac{\frac{1}{M} \frac{1}{N} \sum_x^N \sum_y^M p(x, y)}{\frac{1}{N} \sum_x^N p(x, y)} \quad (3.1)$$

The resulting image looks much improved over the original image, but there are still a few problems with this approach. First, if the overall image is dark then the resulting image may be too dark to distinguish cracks effectively. Also local changes in column intensity, such as that due to the longitudinal joint, are essentially lost. This can be seen in figure 3.5. Instead of scaling to the average of the columns, an empirically determined value could be used so that all resulting images would have approximately the same level of brightness, however this would not resolve the second issue. Instead, we propose an algorithm based on a model of the lighting system as described in the following section.

3.4 Model-based Lighting Correction

Due to the nature of the system as described previously, there are peaks and valleys in the intensity of the light being reflected by the pavement. By measuring and modeling these intensities, a transformation could then be applied to equalize the intensities across the image. This section describes the approaches used based on our model of the lighting system.

3.4.1 Light System Model

As described in section 3.1, the lighting system is made up of multiple lamps at different positions and angles. To model this system, we used three components: a single center lamp, a single side lamp, and the overall lighting system. We measured the intensities by drawing a line under the center of the lamps, then using a light meter to determine the intensity at one inch intervals. To measure the single lamps, the others were blocked or turned off and then the intensities measured. By combining these values along with their position, a

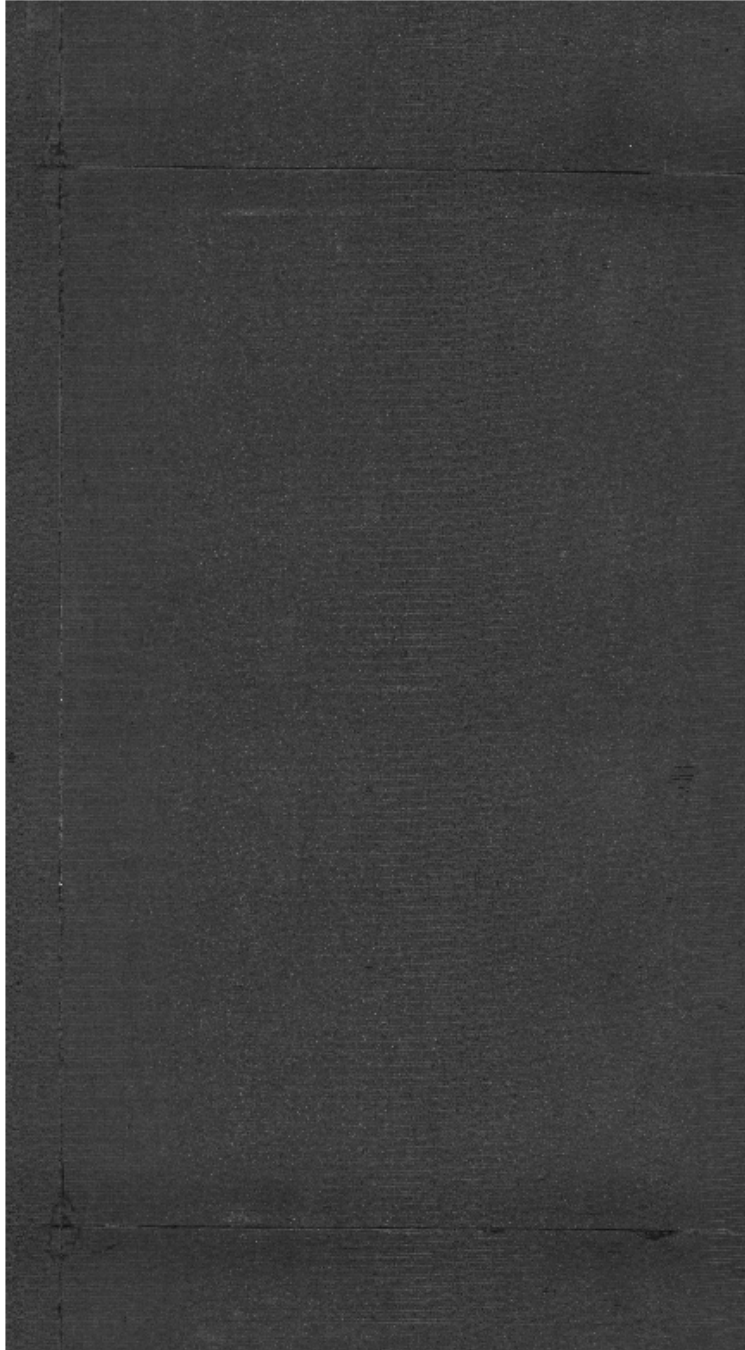


Figure 3.4: Resulting image after column average scaling

representation closely matching that of the overall lighting system could be calculated.

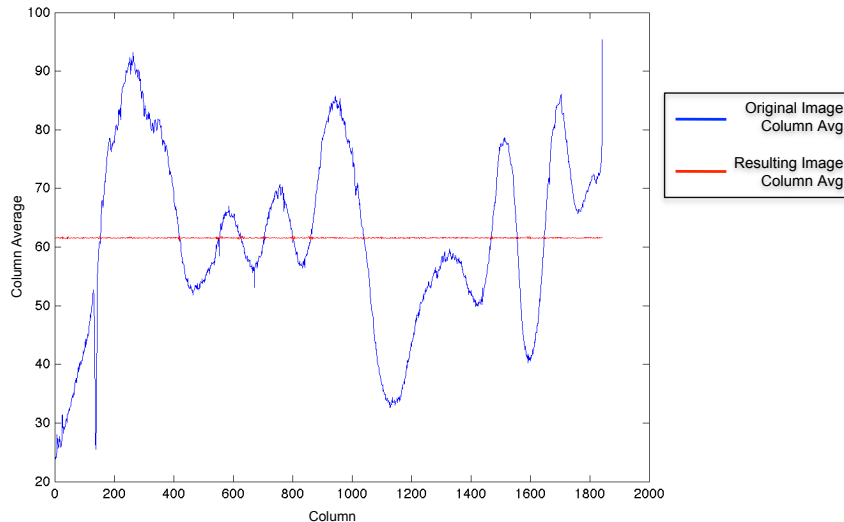


Figure 3.5: Column vs column average

3.4.2 Global Model Correction

After determining the global model as described previously, a set of gains could be calculated to compensate for the peaks and valleys in intensities. The gain was calculated similarly to the column gains in section 3.3, but the mean of the model is used instead of the mean of the column averages. Since these gains are not dependent on the image, they can uniformly be applied to any image.

A resulting image is shown in figure 3.6. Since the position of the lights in relation to the lane can change, it was difficult to match the portion of the gains to be applied. This caused some areas of the image to be brighter where higher gains than necessary were used.

3.4.3 Single Light Model Correction

In order to be more adaptive than the global model, the single light model was used alone. A cosine function was used to create a model of the single light, which allowed more controllability to change the width of the light as needed. To determine which areas were too dark, the column averages were again used. These averages were then separated into sections and the mean of each section is calculated. The sections and their mean are shown in figure 3.7,

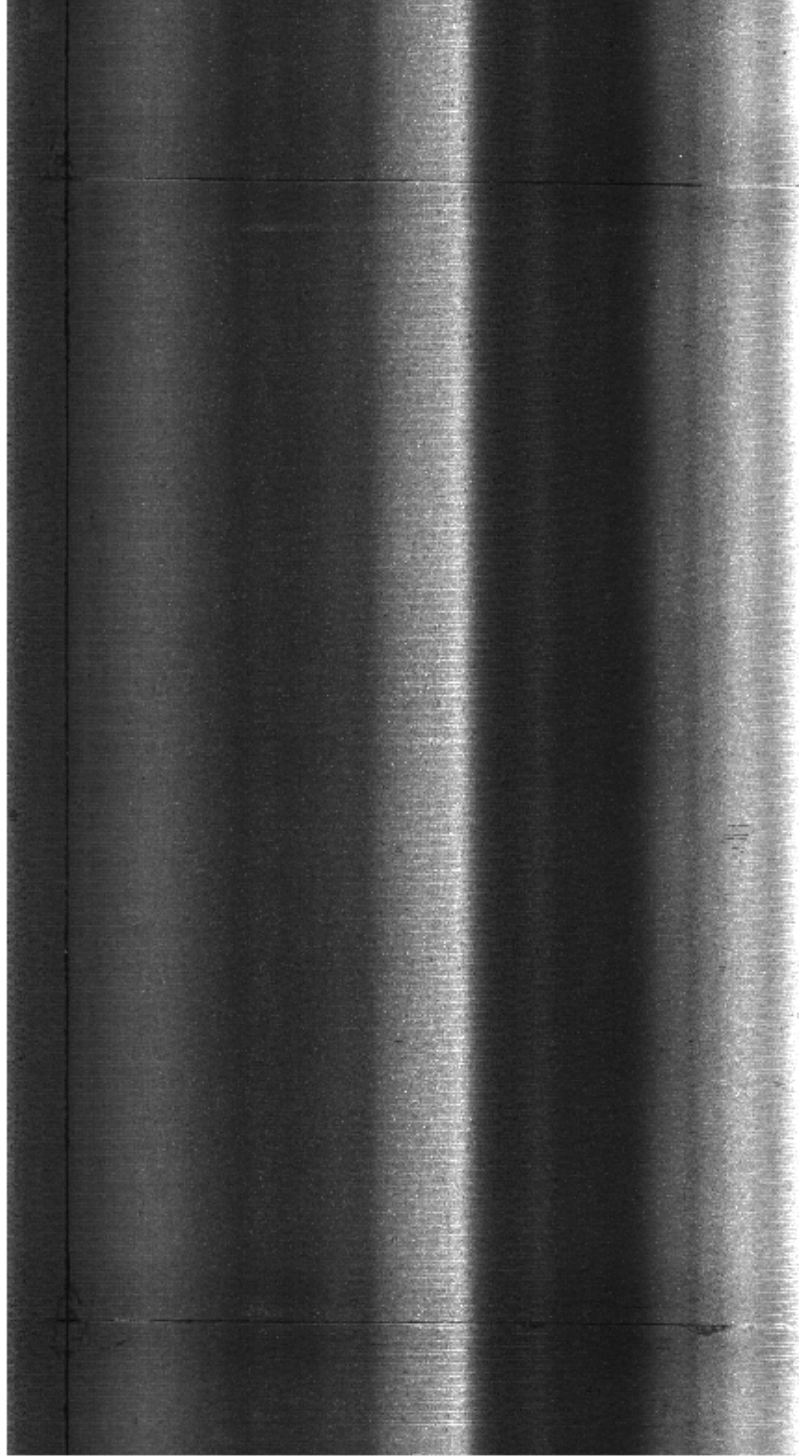


Figure 3.6: Resulting image from global model correction

where the column averages are in blue, the section boundaries are in red, and the section means are in green.

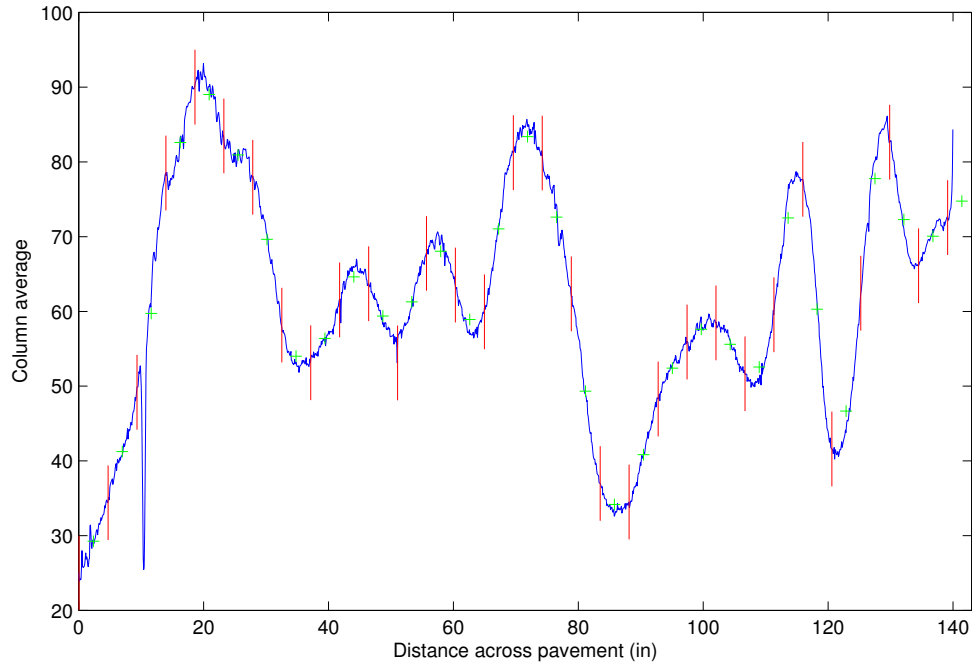


Figure 3.7: Column mean with section means

Each section's mean is compared to the maximum mean of the neighboring sections. If the mean is lower than either of the neighbors, a light is added. This threshold was empirically determined to be 7 percent. A light is essentially a set of gains that are applied to a portion of the columns. As a light is added, it is scaled by the ratio of the section mean and the neighbor in order to make it more adaptive to the change in lighting needed. The resulting image can be seen in figure 3.8. This process is done iteratively until all of the sections are within the threshold, or the maximum number of iterations is reached.

This approach allows us to get a more even appearance in the images while maintaining local information. While the resulting image is not perfectly even, the appearance is enough that it is visually easy to identify cracks.

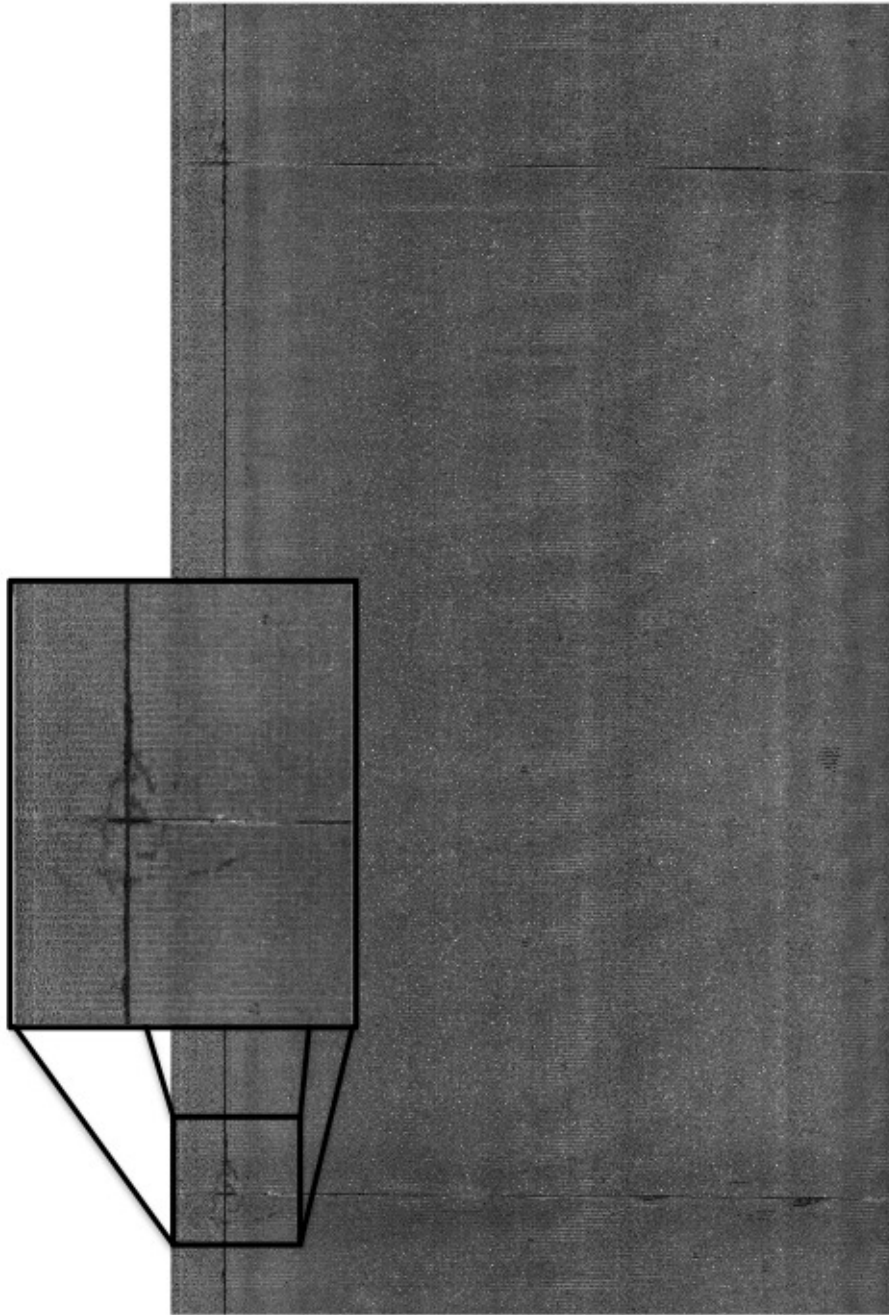


Figure 3.8: Final image after light correction

3.5 Summary

While the lighting system on the imaging vehicle helps prevent effects of low lighting, a method for improving the lighting in some situations is necessary. Traditional approaches

to light correction such as histogram equalization were unsuccessful in correcting for lighting artifacts in pavement images. Other methods of column scaling and the global model-based correction, while moderately effective, were not successful in every facet. However, the single light model-based system presented here corrects for lighting artifacts while preserving local cracking phenomena. This allows us to have effective crack detection, as described in the next section.

Chapter 4

Crack Detection

Cracks are defined by a set of pixels which are darker than the pixels in the surrounding neighborhood. “D”-cracks are especially fine, usually only 2-3 pixels in width. As defined in chapter 2, the cracks first appear most prominently surrounding the intersection of the longitudinal and transverse joints.

4.1 Region of Interest

To avoid additional noise and reduce processing time, crack detection is done only on the area surrounding the transverse joint. We identify the transverse joints as the minimums of the row sums. The row sums of an image are the sum of the pixel values across the rows, as shown below in figure 4.1. After identifying the center of the transverse joint, a one foot area on each side of the joint is taken as the region of interest. The region is then processed at the pixel level.

4.2 Artifact Identification

After identifying the region of interest around the transverse joint, the paint stripe and longitudinal joints must be identified before further processing. If not, during the sub image processing stage they can be identified as cracks, which can be not only visually distracting but also cause improper classification.

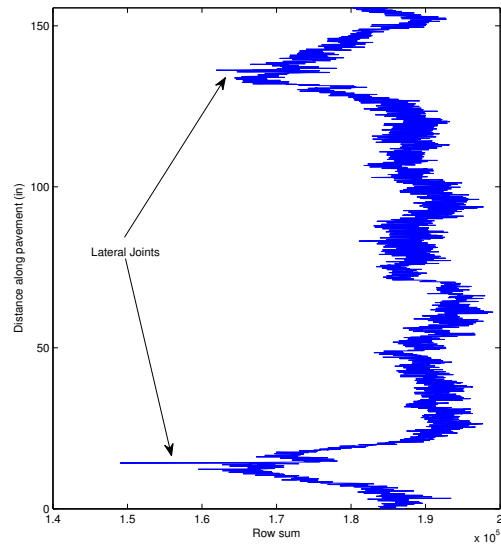


Figure 4.1: Row sums

We begin by limiting the search to the outer quarters of the image. First the paint stripe is identified by looking across the image for consecutive pixels above the mean. After a stripe is identified, the area to the left and right of the stripe is searched for a longitudinal joint. This is done by calculating the column sums for each section then the mean is subtracted and absolute value taken so that the dark joints appear as spikes. Then it searches from the left and right to find the edges of the spike. The zero-mean column averages for an example image section is shown in figure 4.2 with the threshold.

4.3 Subimage Processing

After identifying and removing the artifacts, processing the subimages for cracks can begin. In order to identify cracked and noncracked pixels, we implemented multiple approaches. Each region is divided into overlapping 64 x 64 pixel subimages for further processing.

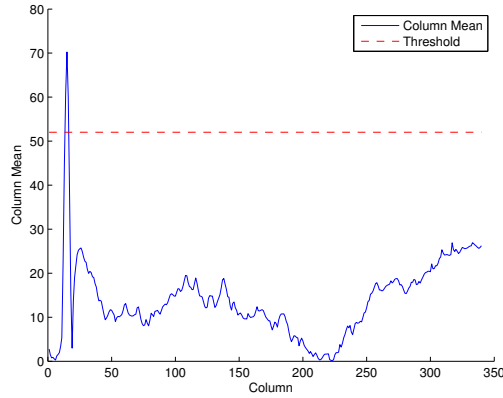


Figure 4.2: Column averages for image section with joint

4.3.1 Residual Error

Using the principle that cracks are darker than surrounding pixels, the first technique was a simple thresholding technique using the residual error. On each subimage, the residual error is given by:

$$r = (AA^+ - I) * b \quad A = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 2 & 1 \\ & \vdots & \\ 0 & N-1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \\ & \vdots & \\ 1 & N-1 & 1 \\ 2 & 0 & 1 \\ & \vdots & \\ N-1 & N-1 & 1 \end{bmatrix} \quad (4.1)$$

where A is a matrix of the subimage pixels indices, A^+ is its pseudoinverse, and b is the subimage pixel intensity values that has been reshaped into a vector. The original subimage and the resulting residual error is shown in figure 4.3b. In the residual, cracks are shown as brighter pixels.

We determine the threshold by creating the histogram and calculating the value for

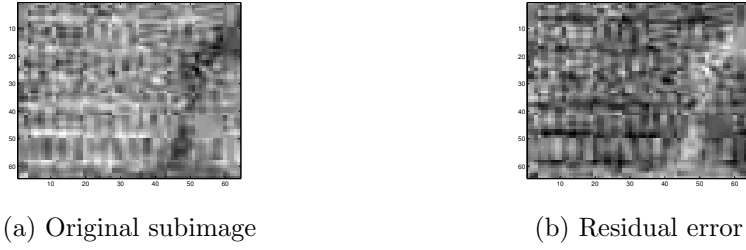


Figure 4.3: Subimage of a crack

which a percentage of the subimage is above the threshold. This threshold is then used to separate the subimage into cracked and noncracked pixels as shown in figure 4.4. Denoting the pixel value at location (x, y) in the image as $p(x, y)$:

$$p(x, y) = \begin{cases} 1 & \text{cracked if } r(x, y) \geq T \\ 0 & \text{noncracked if } r(x, y) < T \end{cases} \quad (4.2)$$

where T is the threshold as determined by the histogram. The threshold is changed depending on the location within the section. Since “D”-cracking appears more heavily at the intersection of the joints, the threshold is lower in these areas so that more pixels are detected as cracked. The center area has a higher threshold so that less noise is detected.



Figure 4.4: Subimage After Thresholding

This method is simple yet effective at identifying possible cracked pixels. Using the residual highlights the differences between background pixels and those which are cracks. For comparison, probabilistic relaxation was implemented as described in the next section.

4.3.2 Probabilistic Relaxation

Another approach to detect cracked pixels is probabilistic relaxation. Probabilistic relaxation is a proven iterative technique used to identify features such as ridges or edges in images [8]. We set up the probabilities in 4 directions as labeled in figure 4.5, as well as a probability of no crack.

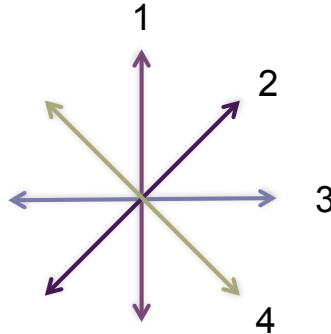


Figure 4.5: Probabilistic setup

After creating subimages as described previously, a negative image is produced so that higher pixel values correspond to cracks. To determine the initial probabilities, zero-mean spatial filters are applied to the images for each crack direction, then two normalizing functions, shown in equations 4.3 and 4.4, are applied to the output of the filters. The output from each function is then multiplied together. P_c is the probability and σ_c is the standard deviation corresponding to each direction of cracks. The probability of no cracking is normalized by the g function alone. The output of these function are normalized to probabilities by dividing each pixel by the sum of the probabilities, so that for each pixel the sum of the probabilities is 1. The normalized initial probabilities for a subimage are displayed in figure 4.6 below.

$$f = \frac{1}{1 + e^{P_c/\sigma_c}} \quad (4.3)$$

$$g = e^{\frac{F_c^2}{a}}, \quad \text{where } a = \frac{\sigma_c^2}{\ln(0.5)} \quad (4.4)$$

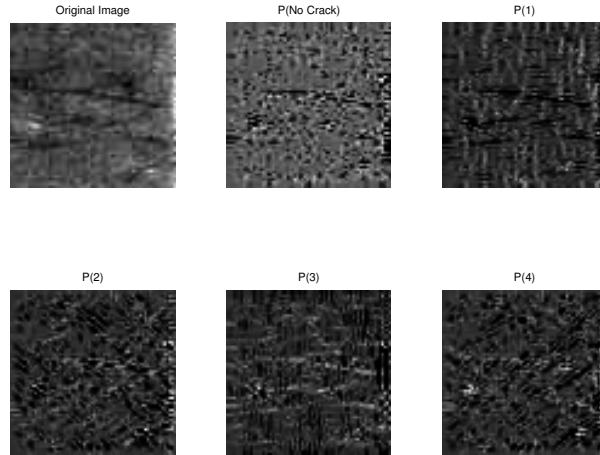


Figure 4.6: Initial probabilities

In every iteration, evidence is calculated for each pixel by the probabilities of the neighboring pixels then normalized to be between 0 and 2. A value of 2 would indicate strong evidence, whereas a value of 0 would indicate opposing evidence. Any evidence over 1 would increase the probability for that pixel, and evidence of less than 1 decreases that probability. The probabilities are then multiplied by the evidence and normalized. Figure 4.7 shows the resulting probabilities in each iteration. Using these probabilities, the algorithm iterates until the change in value is below a given threshold or the maximum number of iterations is reached.

After the iterations are complete, the final probabilities are used to determine which pixels are cracked. If a pixel has a probability in any direction (1, 2, 3, or 4) above 50%, it is considered cracked. An example of a final crack map for a subimage is shown below in figure 4.8, with the cracked pixels shown in blue.

While this method was effective on some images, it was not repeatable in all cases. This is due to the different nature of the cracks, which can vary in size and shape. The filter

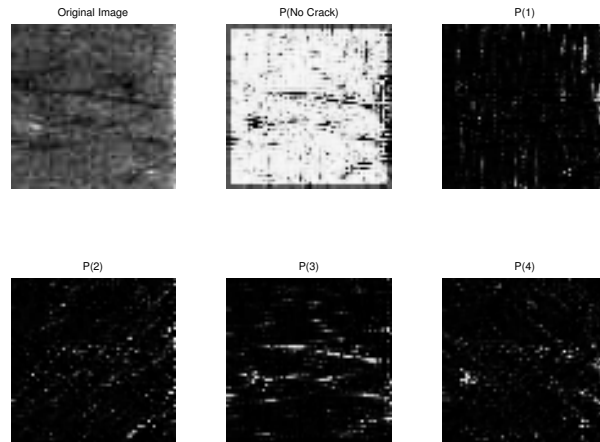


Figure 4.7: Probabilities after five iterations

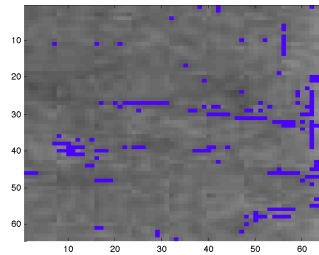
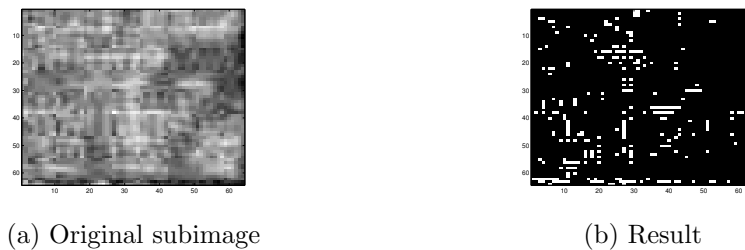


Figure 4.8: Final crack map for subimage

setup is greatly dependent on being able to identify characteristics which would be common to all the images. Figure 4.9 shows one such image where the cracks were not identified by this approach. Instead the residual thresholding was used along with connectivity.



(a) Original subimage

(b) Result

Figure 4.9: Probabilistic relaxation of subimage

4.4 Connectivity

At the subimage level, further processing is needed to help eliminate noise from other isolated dark pixels. Connectivity of pixels is an often used image processing technique which groups pixels into sets based on their proximity. For example, in 8-connected sets any neighboring cracked pixel in a 3 x 3 neighborhood is considered to be in the set. Figure 4.10 shows the neighborhood used to determine 20-connectivity, which was used to determine our sets of cracked pixels at the joint intersections. Just as the threshold was adjusted based on location in section 4.3.1, the type of connectivity used was also changed. In the center of the pavement lane, 8-connectivity was used to reduce the amount of noise detected.

| | | | | |
|---|---|---|---|---|
| 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 |

Figure 4.10: 20-Connectivity

Sets can range in size from 1 pixel to the size of the subimage. These sets are thresholded to be larger than a percentage of the subimage size. The threshold was empirically determined to be 0.5% of the subimage size, which for a 64 x 64 pixel subimage is 20 pixels.



Figure 4.11: Subimage after connectivity applied

4.5 Summary

Using the image processing techniques above, the crack detection algorithm is able to identify candidate cracks with accuracy. The residual error proved to be more effective than probabilistic relaxation at identifying all possible types of cracks. Connectivity was useful for identifying and removing isolated dark pixels. Finally, the overlapping subimages are combined using a logical “or” operation to create a crack map for the region of interest. If a pixel in a subimage is determined to be cracked, it is considered cracked for the region as well. In the next chapter, we discuss morphological methods used in postprocessing to reduce noise and classification to identify those regions which demonstrate “D”-cracking.

Chapter 5

Postprocessing and Classification

After crack detection has been performed, many artifacts may remain in the image as isolated cracked pixels. In an effort to reduce this noise, morphological operations were used to clean up the crack maps. Next the classification algorithm attempts to differentiate those images which exhibit “D”-cracking using several metrics that indicate the type and severity of cracking. The following section describes the postprocessing done before classification is performed.

5.1 Postprocessing

The pavement images contain many artifacts, such as tinning, that may be picked up during crack detection. To remove these and any other noise that may have been detected, we used 2 approaches: row/column sums and morphological methods. The row/column sums are a simple sum of the crack map across the row or column. If any row or column is cracked for over two-thirds of the subimage length, it is determined to be an image artifact and the entire row or column is set to noncracked.

Morphological methods, on the other hand, are useful for detecting phenomena which occur in a specific pattern. This pattern is a simple, pre-defined shape called a structure element, such as defined in figure 5.1. Two types of operations can be performed: erosion and dilation. Erosion accentuates gaps whereas dilation fills them. Also, these operations can be combined for an open or close operation. In the open operation, for example, erosion

is executed first followed by dilation, which causes anything smaller than the structuring element to disappear and anything larger to be smoothed. A close is the inverse operation: dilation followed by erosion. Here an open operation was applied in order to allow larger cracks to remain, while smaller isolated cracks were excluded.

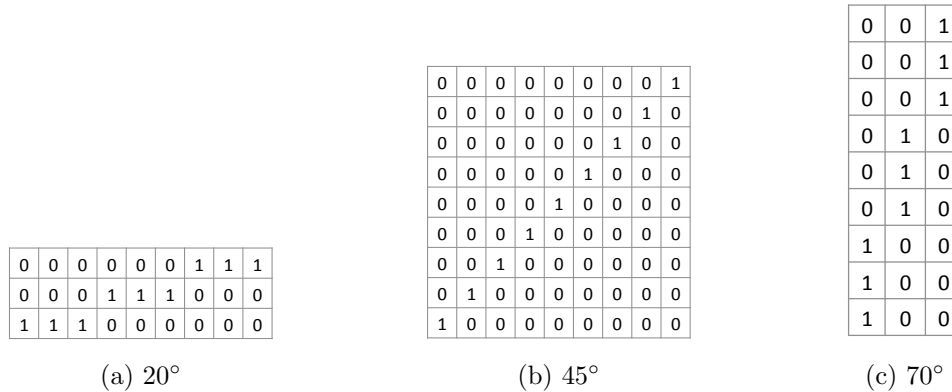


Figure 5.1: Structure Elements

Since “D”-cracking tends to appear on the diagonal around the intersection of the transverse and longitudinal joints, line structure elements at various angles were used. We determined the appropriate angles for this application were 20, 45, and 70 degrees (see figure 5.1), as well as their negatives, and a length of 9 pixels.

The resulting image from each structure element is combined with the other resultant images by a logical “or” operation. Therefore any cracked pixel from any of the structure elements will be considered cracked in the final crack map. Figure 5.2 is the resulting crack map.

5.2 Classification

In general, classifiers are used to make distinctions between one or more categories based on statistics gathered from the data. After a region is processed, the crack map is used to classify it as cracked or noncracked. Since “D”-cracking is primarily at the intersection of the joints, we analyzed the inner corner sections to calculate our metrics as shown in red in

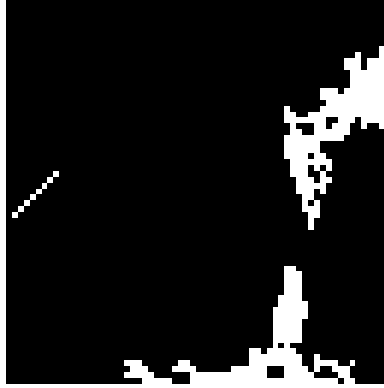


Figure 5.2: Crack Map After Postprocessing

figure 5.3. The metrics for classification are calculated from these quadrants.

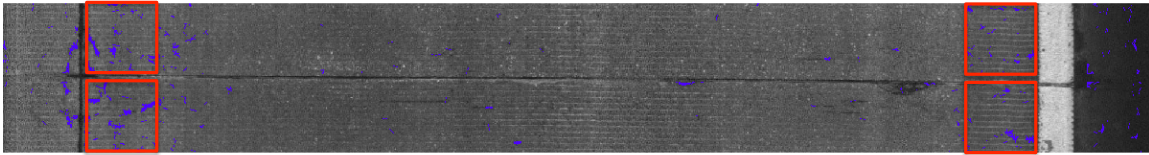


Figure 5.3: Quadrants of region to be analyzed

Multiple criteria were devised in order to measure the type and severity of cracking for a given image: number of cracks along a 45° line, percentage of quadrant cracked, angles of the crack segments, and average size of a crack in pixels.

The number of cracks along a 45° line from the joint intersection is calculated by determine the sets of cracked pixels using 8-connectivity. Then the number of unique cracks intersecting with the line is determined. For example, in figure 5.4 the line shown would intersect 2 cracks. This idea comes from the fact that more severe “D”-cracking exhibits multiple closely spaced cracks in this area. Next, percentage of region cracked is calculated simply as the sum of the cracked pixels divided by the total number of pixels in the region. The angles of the crack segments are analyzed by determining the angle of any crack larger than 40 pixels as well as the angle between the center of the crack and the joint intersection. The difference between these angles for each crack is found and differences less than 15 degrees are counted. This is due to the fact that the angles should have similar values, as

shown in figure 5.5, due to the shape of “D”-cracks. Finally, average size of a crack in pixels was calculated from the 8-connected sets by finding the mean of the size of each set. The histograms for cracked and noncracked images for each metric is in figure 5.6. As can be seen in figures 5.6a and 5.6c, there is separation between the means but the variances are such that there is too much overlap to build an effective classifier. The metrics of percent cracked and average crack size, however, have much more separation between the means.

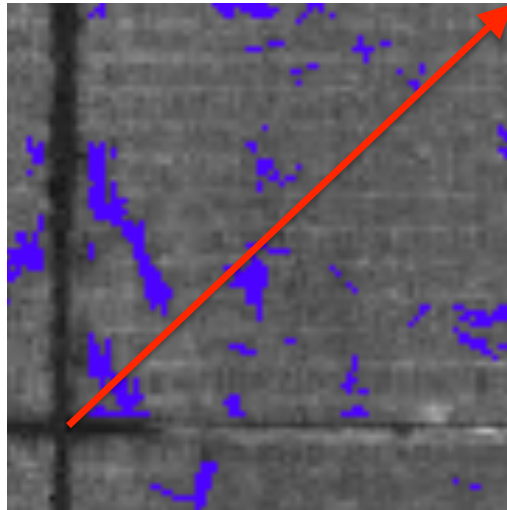


Figure 5.4: Cracks along 45° line

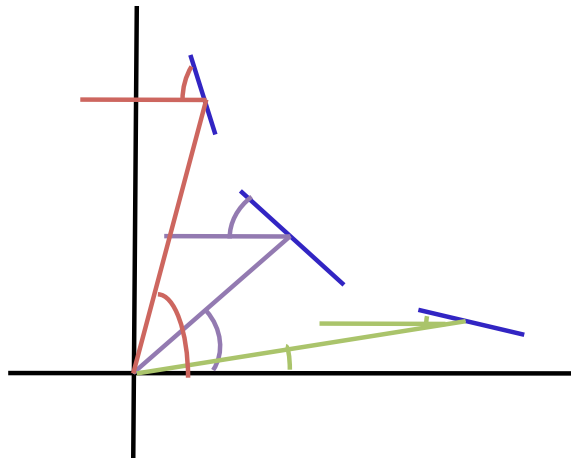


Figure 5.5: Angles of crack segments

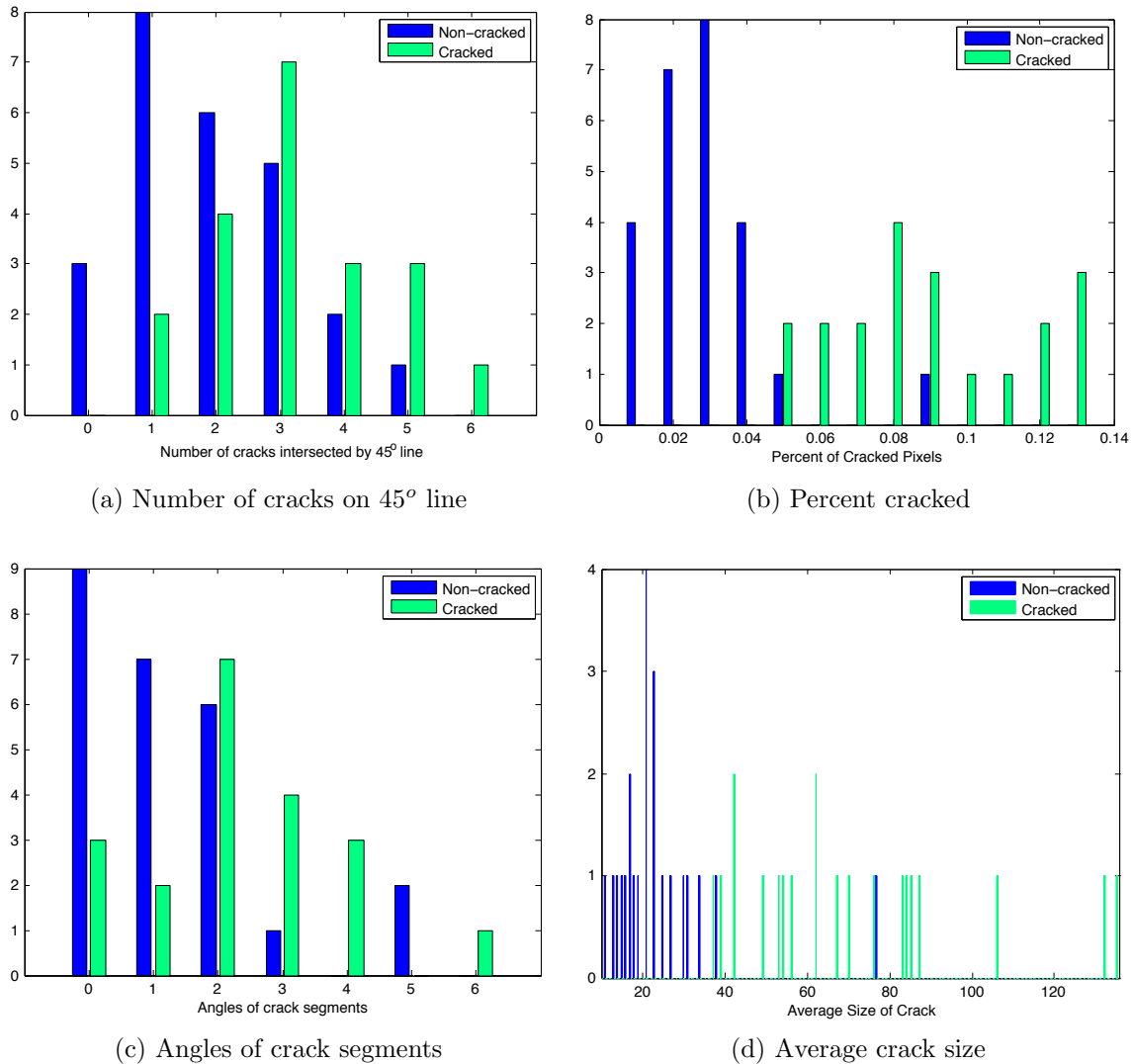


Figure 5.6: Metric histograms

To design the classifier, a set of 10 images were visually analyzed and the four quadrants of each region were determined to be cracked or noncracked. This resulted in 48 data points which comprised the training set. After calculating the four metrics for this set, the distribution for each metric was found to see if there was a correlation between the metric and the cracked/noncracked classification. The percentage of cracked pixels and the average crack set size had the best correlation. Figure 5.7 shows a scatter plot of these metrics with cracked points in red and noncracked in green.

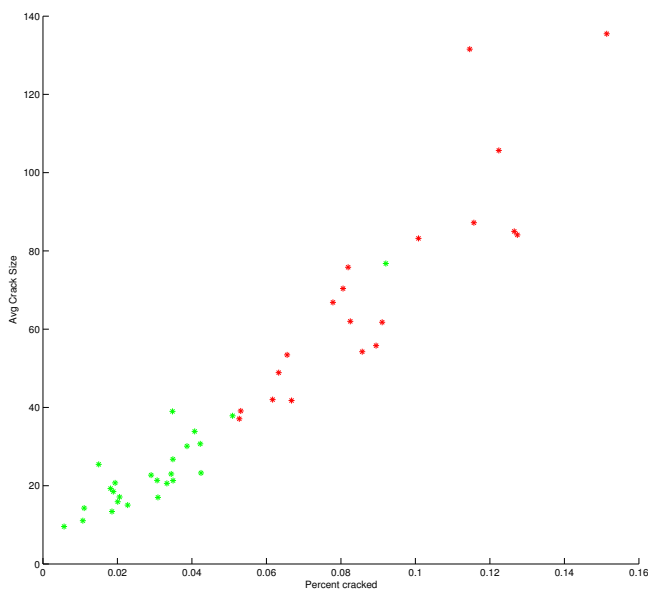


Figure 5.7: Percent cracked vs. average set size

The relationship between the 2 metrics can easily be seen in figure 5.7. The noncracked quadrants clustered around the lower left, but the cracked quadrants had a wider distribution as both metrics increase. To classify, the squared Mahalanobis distance in equation 5.1 was used, which estimates the distance a point is from the mean based on its distribution [3]. This distance was calculated from the noncracked distribution, then compared to a threshold. If the distance was greater than the threshold, which was empirically determined to be 4, the quadrant was classified as cracked.

$$D^2 = (\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu}) \quad (5.1)$$

If any quadrant was determined to be cracked, the entire region was classified as cracked. Overall a conservative classifier was desired so that any possible cracks could be reviewed visually by the user. Also, this approach could easily be extended to detect the severity of cracking. Figure 5.8 demonstrates the thresholds of 4 for mild cracking (green), 10 for moderate cracking (yellow), and 30 for severe cracking (red).

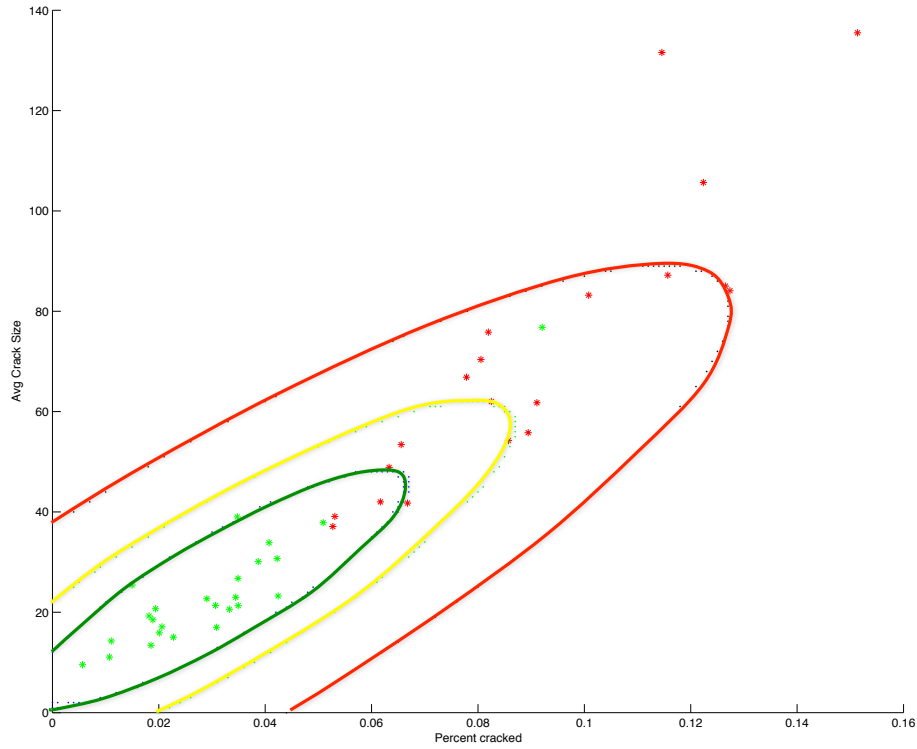


Figure 5.8: Thresholds for classification

5.3 Summary

This chapter described the postprocessing techniques used in order to reduce noise, as well as the design of the classifier. By combining a simple row/column sum technique, the error from any portions of joints, tining, or other artifacts were reduced. Morphological operations also aid in reducing noise from artifacts, especially isolated pixels. By analyzing a set of images with both cracked and noncracked quadrants, we were able to design a classifier using the Mahalanobis distance from the noncracked set. In the next chapter, results from large scale testing are discussed.

Chapter 6

Visualization and Results

After performing postprocessing, the final crack map needs to be displayed to the user in a way that will convey the information about the detected cracks. One of the goals of this thesis was to provide the user with a map detailing any possible cracks so that it could be reviewed to improve the classification of “D”-cracking. The visualization section describes the method of displaying the crack information to the user. The final section discusses the large-scale testing and the statistics about the performance of the classifier.

6.1 Visualization

Each image contains up to three joints, which are displayed independently. The light corrected image is displayed to provide the user with a frame of reference. Then the cracked pixels are overlaid on the image in blue, which was best for highlighting the cracks without distracting the user from the image.

An example of a final crack map displayed with the image after lighting correction is shown below in figure 6.1. These maps assist the user in quickly identifying areas of possible cracking. If the region is determined to be cracked by the classifier described in section 5.2, the region image is displayed on the screen. Also, the crack maps can be saved to an output file for future reference.

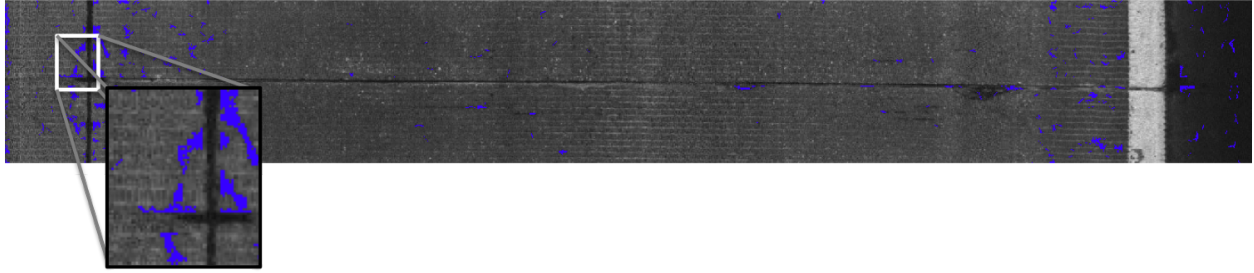


Figure 6.1: Final Crack Map

6.2 Large-scale Testing

The algorithm was implemented using MATLAB 2011a. In order to determine the performance, a set of 155 images were analyzed. Images were split into regions surrounding the transverse joint as described in section 4.1, resulting in 230 regions to be classified. These regions were visually determined to be either cracked or noncracked, then compared with the output of the algorithm classifier. Table 6.1 below shows the number of regions that were classified visually and by the algorithm. The performance is summarized by the statistics listed in table 6.2.

Table 6.1: Classification of test set

| | | Visual | |
|-----------|------------|---------|------------|
| | | Cracked | Noncracked |
| Algorithm | Cracked | 73 | 59 |
| | Noncracked | 25 | 73 |

To see how the threshold affected classifier performance, results were compiled using

Table 6.2: Performance Statistics

| Category | Percent |
|----------------------|---------|
| Correctly classified | 63.5% |
| False alarm | 25.7% |
| Missed cracks | 10.9% |

thresholds between one and 40. Figure 6.2 shows the resulting curves for the probability of false alarm and missed cracks. Depending on the desired performance of the system, the threshold can be adjusted to give the appropriate results. For example, if a low probability of missed cracks is desired, a low threshold should be used.

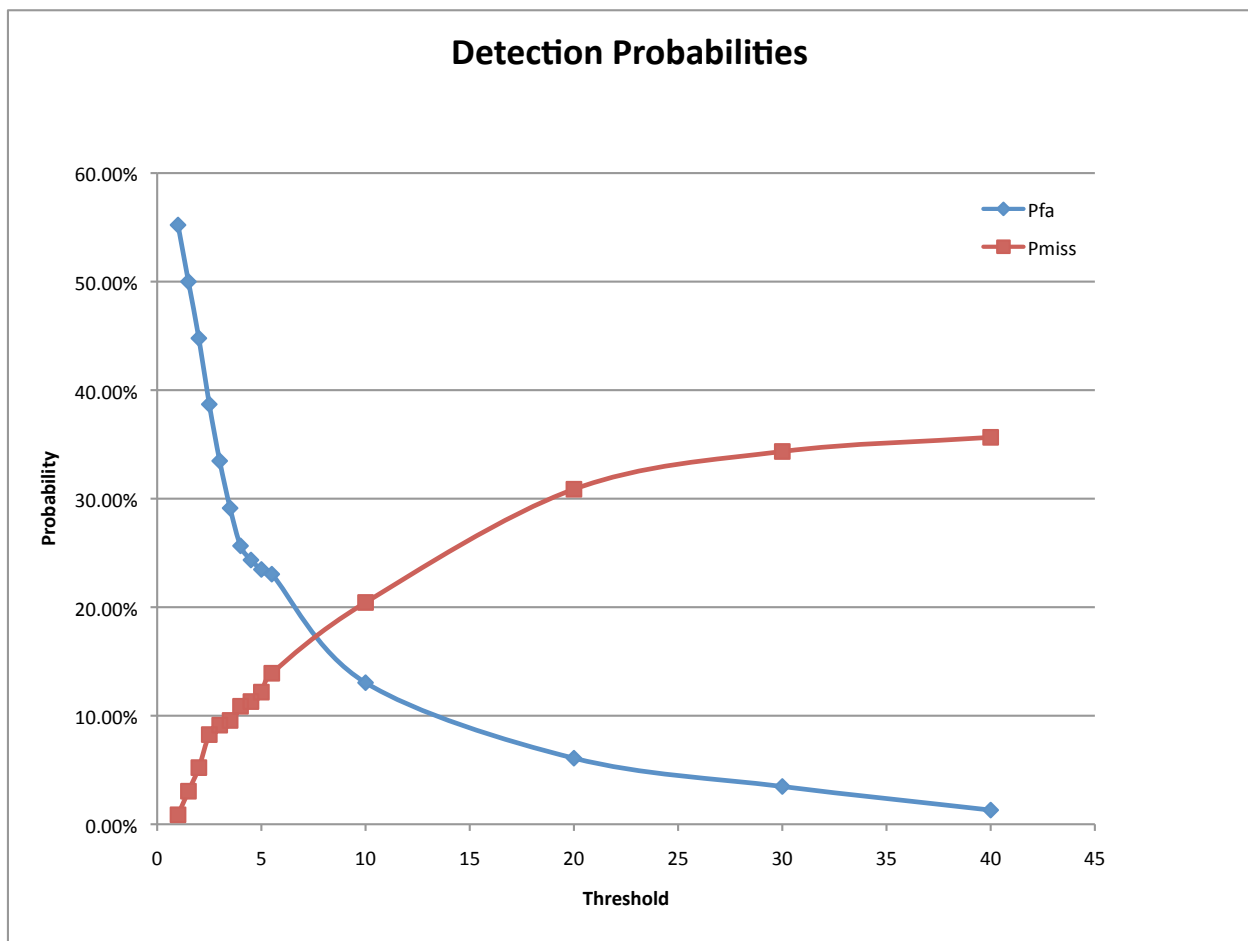


Figure 6.2: Probability of missed crack and false alarm for varying thresholds

Overall the performance is adequate for the current system. The percentage of cracks that are missed is relatively low, and a majority of images are correctly classified. The false alarm percentage is fairly high, but this is desirable since it is preferable for the user to review an image rather than miss possible cracks.

6.3 Summary

In this chapter, we described the method of visualization of the cracked images so that the information could be quickly gleaned from the image without distracting from the original image. Finally, large-scale testing of the classifier showed satisfactory performance of our classification algorithm. Next, the conclusion of this thesis summarizes the key contributions and possibilities for future work.

Chapter 7

Conclusion

Previously, we discussed the visualization of the algorithm output as well as the system performance. In this chapter, the key contributions of this thesis are summarized and possibilities for future work are identified.

7.1 Summary of Key Contributions

This thesis described an automated process capable of detecting and classifying “D”-cracking in pavement images, which requires minimal human interaction. Lighting correction enables improved detection of the fine cracks that we are interested in, as well as assist the user when looking at the final crack map. When detecting the cracks, the residual error worked well to threshold the subimages, and connectivity provided a way to detect continuous areas of cracking. In postprocessing, the row/column filter and morphological operation eliminated noise from image artifacts and isolated dark pixels.

Images are then classified based on the percentage of cracked pixels and the average size of a crack. Classification is based on the Mahalanobis distance from the mean of the non-cracked images from the training set. The cracked images are then displayed to the user for visual inspection. The results from the large-scale testing are encouraging, in that there is a low probability that a cracked image will be missed.

7.2 Future Work

The following topics are possible areas that could be investigated or expanded upon in future research.

- Preprocessing image. A filter could be devised to reduce the effects of some pavement artifacts such as tining. Also some effects from JPEG image compression could be reduced by using a filter. These filters could be applied before the crack detection, and would help reduce noise from those effects.
- Refine classification metrics. The metrics presented in this thesis provided an adequate means of classification, but other metrics could be found to better define the types of cracks. These could focus on the shape and orientation of the cracks.
- Conversion to C/C++ platform. In comparison to MATLAB, C/C++ could provide reductions in computational time. It would also enable greater portability across different platforms.

Bibliography

- [1] JaChing Chou, W.A. O'Neill, and H.D. Cheng. Pavement distress classification using neural networks. In *Systems, Man, and Cybernetics, 1994. 'Humans, Information and Technology', 1994 IEEE International Conference on*, volume 1, pages 397–401 vol.1, oct 1994.
- [2] Concrete Construction. Troubleshooting: D-cracking. *Concrete Construction*, February 2000.
- [3] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification, Second Edition*, pages 35–36. Wiley- Interscience, 2001.
- [4] Lihao Hong, E. Salari, and E. Chou. Pavement information system: Detection, classification and evaluation. In *Electro/Information Technology (EIT), 2010 IEEE International Conference on*, pages 1–5, may 2010.
- [5] Prithvi S Kandhal; Rajib B Mallick. Pavement recycling guidelines for state and local governments. Technical report, Federal Highway Administration, 1997.
- [6] John S Miller. *Distress identification manual for the Long-Term Pavement Performance Program*. U.S. Dept. of Transportation Federal Highway Administration, 4th rev. ed. edition, 2003.
- [7] Florida Department of Transportation. Existing pavement condition before rubblization. *Pavement Management*, August 2008.
- [8] P. Papachristou, M. Petrou, and J. Kittler. Edge postprocessing using probabilistic relaxation. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 30(3):383–402, jun 2000.

- [9] Krithika Rajan, Dwight D. Day, and Balasubramaniam Natarajan. Analysis of pavement condition data employing principal component analysis and sensor fusion techniques. In *Proceedings of 2008 International Conference on Image Processing, Computer Vision and Pattern Recognition*, volume 2, pages 422 – 428, Las Vegas, NV, United states, 2008. IPCV.
- [10] T. Saar and O. Talvik. Automatic asphalt pavement crack detection and classification using neural networks. In *Electronics Conference (BEC), 2010 12th Biennial Baltic*, pages 345 –348, oct. 2010.
- [11] P. Subirats, J. Dumoulin, V. Legeay, and D. Barba. Automation of pavement surface crack detection using the continuous wavelet transform. In *Image Processing, 2006 IEEE International Conference on*, pages 3037 –3040, oct. 2006.
- [12] N.T. Sy, M. Avila, S. Begot, and J.C. Bardet. Detection of defects in road surface by a vision system. In *Electrotechnical Conference, 2008. MELECON 2008. The 14th IEEE Mediterranean*, pages 847–851, 2008.