# Multi-Bayesian Approach to Stochastic Feature Recognition in the Context of Road Crack Detection and Classification

John Josiah Steckenrider

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Master of Science
In
Mechanical Engineering

Tomonari Furukawa
A. Lynn Abbott
Robert G. Parker

12/4/2017
Blacksburg, Virginia

Keywords: Bayesian classification, Crack detection, Road condition monitoring, Recursive Bayesian estimation, Stochastic features, Machine learning, Computer vision

# Multi-Bayesian Approach to Stochastic Feature Recognition in the Context of Road Crack Detection and Classification

John Josiah Steckenrider

## ABSTRACT

This thesis introduces a multi-Bayesian framework for detection and classification of features in environments abundant with error-inducing noise. The approach takes advantage of Bayesian correction and classification in three distinct stages. The corrective scheme described here extracts useful but highly stochastic features from a data source, whether vision-based or otherwise, to aid in higher-level classification. Unlike many conventional methods, these features' uncertainties are characterized so that test data can be correctively cast into the feature space with probability distribution functions that can be integrated over class decision boundaries created by a quadratic Bayesian classifier. The proposed approach is specifically formulated for road crack detection and characterization, which is one of the potential applications. For test images assessed with this technique, ground truth was estimated accurately and consistently with effective Bayesian correction, showing a 33% improvement in recall rate over standard classification. Application to road cracks demonstrated successful detection and classification in a practical domain. The proposed approach is extremely effective in characterizing highly probabilistic features in noisy environments when several correlated observations are available either from multiple sensors or from data sequentially obtained by a single sensor.

# Multi-Bayesian Approach to Stochastic Feature Recognition in the Context of Road Crack Detection and Classification

John Josiah Steckenrider

## GENERAL AUDIENCE ABSTRACT

Humans have an outstanding ability to understand things about the world around them. We learn from our youngest years how to make sense of things and perceive our environment even when it is not easy. To do this, we inherently think in terms of probabilities, updating our belief as we gain new information. The methods introduced here allow an autonomous system to think similarly, by applying a fairly common probabilistic technique to the task of perception and classification. In particular, road cracks are observed and classified using these methods, in order to develop an autonomous road condition monitoring system. The results of this research are promising; cracks are identified and correctly categorized with 92% accuracy, and the additional "intelligence" of the system leads to a 33% improvement in road crack assessment. These methods could be applied in a variety of contexts as the leading edge of robotics research seeks to develop more robust and human-like ways of perceiving the world.

This work is dedicated to my first love, the One who means everything,
without whom this is all pointless.
When I meet you, will you tell me, “Well done, good and faithful servant”?

And to my beautiful, wonderful, supportive wife. Let’s never stop dreaming together ~~~

And to my father, who told me of a man who was before the world began…

“For am I now seeking the approval of man, or of God? Or am I trying to please man? If I were still trying to please man, I would not be a servant of Christ.”
Gal. 1:10

This work was generously supported by Murata Manufacturing Co., Ltd.
Special thanks to the Virginia Tech Department of Mechanical Engineering and the people of the Computational Multiphysics Systems Laboratory.

# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS*

PCI – Pavement Condition Index

LDR – Load-related Distress Rating

NDR – Non-load-related Distress Rating

CCI – Critical Condition Index

SDR – Slab Distress Rating

SFI – Slab Faulting Index

CPR – Concrete Punchout Rating

CDR – Concrete Distress Rating

PDF – Probability Distribution Function

QBC – Quadratic Bayesian Classifier

RBE – Recursive Bayesian Estimation

NC – No Crack

LT – Longitudinal/Transverse [Crack]

LA – Light Alligator [Crack]

MA – Moderate Alligator [Crack]

SA – Severe Alligator [Crack]

* Only specialized abbreviations are given. Abbreviations such as JPEG and CMOS are either assumed to be understood by the reader or are defined at first use in the text.

# CHAPTER 1. Introduction and Review of Literature

## 1.1 Introduction

The ability of humans to perceive, comprehend, and interact with their surroundings is remarkable and unequaled by any other form of life on earth. This exceptional skill has allowed us to dominate the animal kingdom, build incredibly complex civilizations, and even break the gravitational restrictions that hold us to our planet, sending us to the moon and, perhaps one day, even beyond. Human perception is powered by our capacity for learning and making the best use of all available information in decision making, particularly when such decisions are made difficult by the presence of high uncertainty.

In an increasingly technologically-oriented world, we seek to replace human workers with high-functioning robots or autonomous systems in tasks associated with danger, monotony, or difficulty. Robot functionality will never match that of human beings unless robots can be taught to “think” like humans. This turns out to be an extraordinary challenge on many levels. For one, most modern autonomous systems are not equipped, nor can they easily be equipped, with the full variety of sensors that humans possess. Not only do humans have the five commonly cited senses of sight, hearing, taste, smell, and touch, but we are also equipped with proprioception (the innate ability of a body to localize its own parts via subtle internal cues) [1], inner ear balance mechanisms (a form of inertial measurement) [2], and even possibly a magnetic sense [3], among others. We fuse this array of sensory input information, both consciously and subconsciously, in order to function exceptionally well in our unpredictable and often highly stochastic surroundings. As we interact with our environment, signals are received by various means and transformed into electrical impulses which are used by the brain to adapt actions and understandings in a constant feedback loop [4]. Not only are robots often ill-equipped with any more sensors than the minimum number deemed necessary for a given task, but they also need improvement in the area of sensor fusion for uncertainty handling if human duties are to be safely taken over by autonomous systems. Because the signals which provide information about an environment are so often corrupted with noise intrinsic to both the observation target and the sensors themselves, there is a need for robust estimation and correction with regards to perception [5].

Whereas much previous work has been done in handling sensor-intrinsic noise in dynamic systems, electrical systems, state estimation, etc. [6], [7], [8], [9], the work described here details a novel approach for addressing uncertainty with regards to the identity of an observation target (i.e. a stochastic feature). This falls under the domain of classification and machine learning. Many machine learning platforms, particularly those found in autonomous systems, rely on computer vision based interrogation of a set of objects. Oftentimes, if possible, great care is taken to ensure that the objects being characterized are easily distinguishable in order to achieve high accuracy in classification. However, in the natural world there is an abundance of uncertainty and noise, which even the well-tuned human eye sometimes cannot adequately, quickly, or consistently overcome. Road cracks are an example of this kind of feature. Not only do road type and condition greatly affect the detectability of cracks, but the highly organic and stochastic nature of cracking mechanisms makes it nearly impossible to predict the long-term growth of such a crack. In addition, the significant variability in the appearance of road cracks makes form prediction infeasible. Although this work can be readily extrapolated to any appropriate context, the methods described here are validated in the context of road cracks. The multi-Bayesian techniques developed in this research consist of detection, classification, and classification correction of road cracks, each stage of which is undergirded by Bayesian theory.

## 1.2 Importance of road condition monitoring

Road condition monitoring is a critical aspect of civil infrastructural maintenance. When the quality of local roads deteriorates, the effects can be widespread. Not only is it harmful for vehicles to traverse poor roads, but the cost of road repair is ultimately a financial burden to the taxpayer. In the most extreme cases, there is even a risk to human health and safety. Deterioration happens over periods of time of varying length, depending on the conditions to which a given road segment is subjected. As such, it is crucial to consistently

*Figure 1.1: Road in disrepair. Public Domain.*

monitor pavement health in order to know the optimal time for making repairs [10]. If defects go unnoticed, the results can be as severe as shown in Fig. 1.1 and worse. Current methods of local road assessment largely depend on manual methods in which a qualified technician investigates and categorizes pavement quality. This form of structural health monitoring is both lengthy and subjective. Although a trained assessor's judgement is generally trusted, there is inevitable variation across technicians that can result in a lack of standard quality control. These are the main motivators for an automated road condition monitoring system.

A standard of pavement condition assessment called the Pavement Condition Index (PCI) was established by the Army Corps of Engineers in the 1970s in an attempt to standardize road quality metrics [10]. This system ranks road condition according to six categories (see Table 1.1). Workers are trained to rate pavement quality as consistently as possible, with the goal of achieving a standard measure of surface severity. Although this sytem greatly aids scoring, human variability must be taken into account. Quantifying road distress is a job best executed with a consistent computational approach. This is a motivator of the work described here. Another factor to consider is not only observation, but also decision-making. At a critical value of 65, road repair becomes rapidly more expensive as time goes on and distress increases. As such, the most important time for accurate assessment is up until, and especially at, a PCI score of 65. This kind of decision making is a job well-suited for the estimation methods introduced here.

*Table 1.1: PCI rating, description, and cost. Adapted from [10].*

| Pavement score | Rating (PCI) | Description | Maint. Cost (per $yd^2$) |
|---|---|---|---|
| 80-100 | Excellent | New road, near perfect | ~ $0 |
| 70-79 | Good | Some distress | ~ $3 - $5 |
| 60-69 | Fair | Significant distress | ~ $7 - $9 |
| 50-59 | At risk | Deteriorated pavement | ~ $9 - $13 |
| 25-49 | Poor | Extensive distress | ~ $25 - 45 |
| 0-24 | Failing | Needs reconstruction | ~ $90 - $125 |

Other metrics have been formulated in an attempt to quantify road distress severity. This is largely of interest to local and state government agencies, as they are ultimately responsible for the care of highways and city streets. In the state of Virginia, asphalt pavement is assessed according to two different parameters: load-related and non-load-related distress ratings (LDR and NDR, respectively) [11]. Of these two criteria, the more severe rating is given as the critical condition index (CCI). Similarly to the PCI, the CCI has rating categories based on score ranges (Table 1.2).

*Table 1.2: CCI rating. Adapted from [11].*

| Pavement score | Rating (CCI) |
|---|---|
| 90-100 | Excellent |
| 70-89 | Good |
| 60-69 | Fair |
| 50-59 | Poor |
| 0-49 | Very Poor |

The Virginia Department of Transportation (VDOT) has further subdivided pavement assessment techniques by surface type. The LDR and NDR describe only asphalted roads, whereas sectioned concrete roads are rated by a Slab Distress Rating (SDR) and Slab Faulting Index (SFI), and continuous stretches of concrete are characterized by their Concrete Punchout Rating (CPR) and Concrete Distress Rating (CDR) [12]. Visual distress in jointed concrete is quantified by the SDR, but the SFI is used to characterize the underlying causes for future degradation in the form of elevation changes between concrete slabs. In continuous concrete, the CPR is a measure of impact-driven concrete disrepair, including such things as cluster cracks, punchouts (the concrete equivalent of potholes in asphalt), and patching. The CDR is intended to describe more natural, quasi-static damage including singular cracks (longitudinal or transverse) and spalling [12]. Examples of concrete punchout, cluster cracking, spalling, and longitudinal/transverse cracking are given in Fig. 1.2.

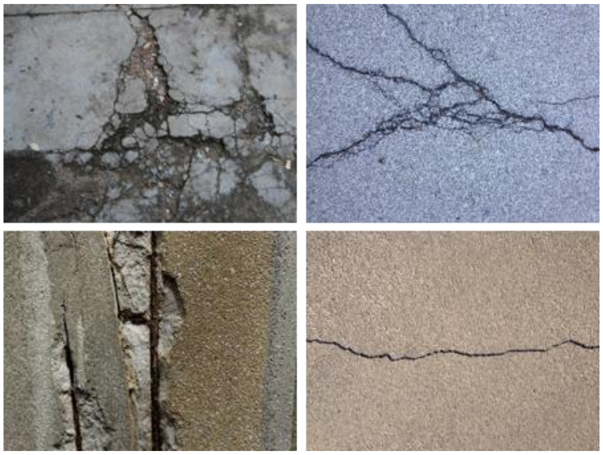


*Figure 1.2: Concrete distress. From top left to bottom right: punch-out, cluster cracks, spalling, and longitudinal/ transverse crack. Public Domain.*

It is important to mention that the aim of this research is to aid in the monitoring and maintenance stage in the life cycle of road pavement. This refers in particular to road segments which fall into the top two or three PCI categories (about 65 and up). This is the most important stage when considering cost effectiveness, and it also precludes extreme distress, for which discrete detection techniques are no longer as reliable. The poorest of roads are those in which pavement is so deteriorated that cracks and other descriptive features are not easily distinguishable.

## 1.3 Pavement cracking mechanics

Let it be stated early on that this work is not an effort to develop the field of materials science and structural mechanics. However, since the context of the multi-Bayesian theory developed here is pavement cracking, a basic understanding of fracture mechanics is critical to a rigorous physics-driven knowledge base for theoretical development. Formulation of fracture mechanics in solid homogeneous materials depends on two primary elements: crack tip stresses, and fatigue (failure due to cracking resulting from cyclical loading) [13]. These mechanisms can both be developed theoretically and derived empirically. It can be shown from energy methods that the stress $\sigma$ developed in a material as a uniform force is applied is given as

$$\sigma = \sigma_{max}^2 \frac{x}{\gamma} \tag{1.1}$$

where $x$ is the axial displacement of the material, $\sigma_{max}$ is the maximum theoretical strength of the material (the maximum stress that can be applied the instant before failure), and $\gamma$ is the specific surface energy of the material (Griffith defined $\gamma$ as the surface tension divided by Poisson's ratio for glass [14], though a more accurate derivation of the value requires a more extensive understanding of the material's properties for more complex materials). Furthermore, the atomic-level strain $\epsilon$ can be defined as

$$\epsilon = \frac{x}{a_0} \tag{1.2}$$

In Eq. 1.2, $a_0$ represents the equilibrium distance between atoms in the material. From the definition stress and strain in linear elastic material, Young's modulus $E$ is

$$E = \frac{d\sigma}{d\epsilon} \tag{1.3}$$

And by substitution of Eq. 1.2,

$$E = \frac{d\sigma}{dx} a_0 = \frac{d}{dx}\left(\sigma_{max}^2 a_0 \frac{x}{\gamma}\right) = \sigma_{max}^2 \frac{a_0}{\gamma} \tag{1.4}$$

Finally, the theoretical strength of the material is

$$\sigma_{max} = \sqrt{\frac{E\gamma}{a_0}} \tag{1.5}$$

This formulation requires a number of assumptions. First, it assumes that the material in study is homogeneous, linear, and elastic. This series of assumptions is generally valid for most materials studied in the context of fracture mechanics. Second, there is an implicit assumption that the material is composed of a perfect lattice structure without defect. This is where the theoretical model usually fails. Consequently, the true yield strength (stress at fracture) is often meaningfully lower than predicted.

By similar energy methods, Griffith showed [14] that

$$\sqrt{\frac{2E\gamma}{\pi}} = constant \tag{1.6}$$

once again, for a homogeneous linear elastic material. Additionally, he noted that this constant was also related to the stress at fracture $\sigma_f$ and crack length $a$ in the following way:

$$\sigma_f\sqrt{a} = constant \tag{1.7}$$

Upon substituting, the following expression for $\sigma_f$ is found:

$$\sigma_f = \sqrt{\frac{2E\gamma}{\pi a}} \tag{1.8}$$

When the critical stress is reached (i.e. $\sigma_f = \sigma_{max}$), critical crack length for the failure of a homogeneous elastic material is

$$a_{crit} = \frac{2E\gamma}{\pi\sigma_{max}^2} \tag{1.9}$$

More recent studies have observed a plasticization at the crack tip of metals, which along with material defects causes the actual stress at the crack tip to be lower than predicted by Griffith [15]. For such materials and circumstances, this error is corrected by the inclusion of an additional property-dependent term as shown below:

$$\sigma_f = \sqrt{\frac{E(2\gamma + G_p)}{\pi a}} \tag{1.8}$$

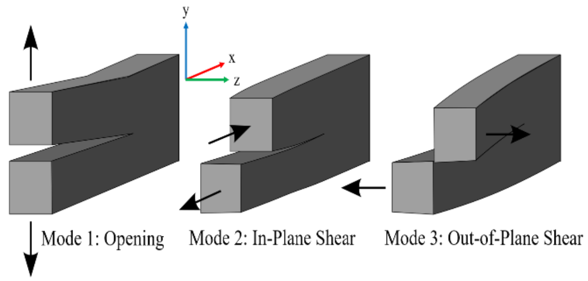


*Figure 1.3: Stress modes. Public Domain.*

In order to arrive at a more detailed description of the stress state near the tip of a fracture, cracking must be considered under different stress conditions. These stress modes are shown in Fig. 1.3. For each mode, the stress distribution at a crack tip depends on the stress intensity factor [13]. Eqs. 1.9-1.11 give the stress intensity factors for modes 1-3, respectively.

$$K_1 = \lim_{r\to 0} \sqrt{2\pi r}\sigma_{yy}(r,\theta)|_{\theta=0} \tag{1.9}$$

$$K_2 = \lim_{r\to 0} \sqrt{2\pi r}\sigma_{yx}(r,\theta)|_{\theta=0} \tag{1.10}$$

$$K_3 = \lim_{r\to 0} \sqrt{2\pi r}\sigma_{yz}(r,\theta)|_{\theta=0} \tag{1.11}$$

These values are used to compute the stress tensor $\sigma_{ij}(r,\theta)$ at radial distances $r$ and angles $\theta$ from the crack tip [16].

$$\sigma_{ij}(r,\theta) = \frac{K}{\sqrt{2\pi r}} f_{ij}(\theta) \tag{1.12}$$

where $f_{ij}(\theta)$ is often a polynomial expression in $\theta$ to fit experimental data. For the simplest mode 1 cracking configuration, shown in Fig. 1.4, $K_1 = \sigma\sqrt{\pi a}$. Of course, this makes several assumptions which will be addressed shortly.

The most important formulation in investigating road fracture mechanics is fatigue. Fatigue, the failure of a material in fracture due to cyclical loading, makes use of the stress intensity factor as well. Many laws have been postulated, and most adequately represent the observed phenomenon of crack growth [17]. Paris' Law is the most widely accepted governing differential equation for fatigue [18]. This relationship, as all others, was not derived from first principles, but was rather extracted from experimental data. It relates the rate of crack growth to the stress intensity factor and two empirically determined constants, $C$ and $m$, aptly named the Paris constants. These variables depend on loading and boundary configurations as well as material properties and other subtleties. Paris' Law is given as:

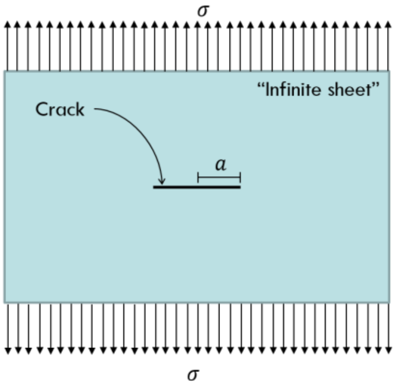


*Figure 1.4: Simple mode 1 cracking.*

$$\frac{da}{dN} = C\Delta K^m \quad (1.13)$$

If the length of a transverse road crack is small compared to the length of the road, which is most often a reasonable assumption, the following substitution can be made:

$$\frac{da}{dN} = C(\Delta\sigma\sqrt{\pi a})^m \quad (1.14)$$

By substituting the loading frequency in Hertz, the time-rate of crack growth is given as a function of the stress cycle, Paris constants, and crack length, making the equation a nonlinear differential equation. In an attempt to apply Paris' Law to road pavement cracking, several groups have carried out laboratory experiments in order to estimate the Paris constants for road materials [19], [20], [21]. In each case, the specimens under investigation were cut into slabs and loaded by a three-point press. Molenaar tested four different kinds of pavements at varying temperatures and frequencies [19]. Table 1.3 shows the results from this study.

*Table 1.3: Experimentally determined Paris constants [19].*

| **Mix** | **Temp. (°C)** | **Freq. (Hz)** | $\boldsymbol{C}$ | $\boldsymbol{m}$ |
|---|---|---|---|---|
| Gravel sand asphalt | 25 | 1 | 6.39E-5 | 2.89 |
| | 15 | 1 | 8.98E-8 | 4.03 |
| | 15 | 10 | 1.01E-9 | 4.37 |
| | 5 | 10 | 5.04E-9 | 3.09 |
| Open AC | 15 | 1 | 1.67E-6 | 3.79 |
| | 15 | 10 | 9.18E-7 | 2.88 |
| | 5 | 10 | 2.61E-16 | 8.70 |
| Dense AC | 15 | 1 | 3.06E-6 | 3.24 |
| | 15 | 10 | 1.40E-6 | 2.57 |
| Cold AC | 15 | 1 | 4.91E-3 | 2.99 |
| | 5 | 10 | 5.16E-4 | 1.26 |

The criteria of the above experiments are fairly restrictive, not only in the controlled application of force and lack of boundary conditions, but also in that the magnitudes of cracks investigated in such procedures are on the sub-millimeter scale. In order to extrapolate Paris' Law to the macroscopic problem of road cracking as it occurs in real situations, the following significant assumptions must be made:

1. The road is effectively an infinite plane of material compared to the crack length
2. Cyclical fatigue loading takes the form of regular traffic
3. Cars induce traction forces as they overcome friction to move
4. Car tire contact is evenly distributed
5. Traction forces produce normal stresses which create mode 1 cracking
6. Crack occurs only in mode 1
7. Crack length can be accurately estimated
8. Material properties can be accurately estimated and the Paris constants accurately represent the fracture mechanics of their target materials
9. Other environmental factors are negligible

Because all the above criteria must be met in order to begin to use Paris' Law to even predict simply crack growth, the strength of this formulation is greatly reduced in macroscopic real-world contexts. Using Molenaar's values to numerically propagate the differential equation for cracks of significant length yields results which are entirely unreflective of reality. This is because the simplified lab experiments hardly begin to approach the complex and multifaceted crack growth problem found *in natura*.

## 1.4 Crack Detection

Because many groups have recognized the importance of automatic road crack detection, several methods have been proposed over the last few decades. The problem has been considered not only in the research community, but also in the private sector with companies like Fugro (Leidschendam, Netherlands) commissioning high-tech proprietary road assessment vehicles worldwide [11]. These platforms have not been publicly validated, and the limits of their techniques have yet to be investigated. Since this work focuses on the development of a sound theoretical framework for crack detection, the research aspect of crack detection is emphasized here.

Some of the first efforts to automatically detect road cracks arose as early as the 1990s [22], [23], [24]. Due to the complex and stochastic nature of crack detection, rarely has the approach to this problem been consistent, nor certainly comprehensive. The review given here will focus on a few broad categories of approaches, some of the work done within each, and a basic presentation of their results.

The first general category of road crack detection makes use primarily of image processing and manipulation on the pixel level. Gavilán, et al. used a line-scan camera and laser illumination to acquire single-pixel-wide image vectors which were then used to reconstruct the 2D road surface in off-site data processing [25]. The system included an algorithm for pavement type assignment in order to better assign parameters critical to accurate crack detection. The first step in image processing was a pre-processing operation in which pixel windows of variable size and step are taken across an image, and a histogram is constructed for each window. A cumulative area threshold is determined beforehand, and the histogram bins are summed up to this threshold value. The theory behind this approach is that pixel windows with darker (crack) pixels will exceed this threshold before those without crack pixels. Fig. 1.5 illustrates this concept. A new image is given the value at which the histogram reaches this threshold. The resulting image is then thresholded and binarized.

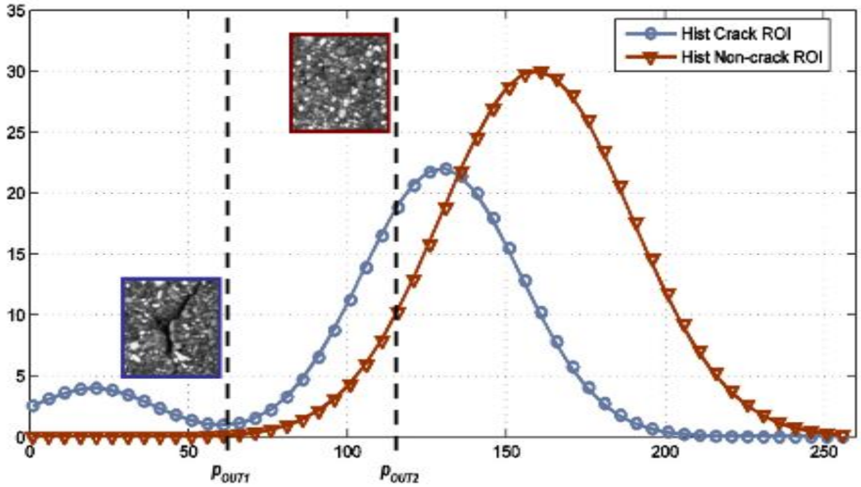


*Figure 1.5: Histograms for different kinds of windows. $P_{OUT1}$ and $P_{OUT2}$ reflect the values of pixels in new image corresponding to window in old image [25].*

This approach first seeks to mask out non-crack regions, or regions in which cracks are not desired for detection. Features meeting these criteria are already-sealed cracks, white road paint, and pavement joints.

In order to eliminate sealed cracks from consideration, a series of dilation and erosion operations are completed on the pre-processed image. A comparison is made between the pixels along eroded edges and those along dilated edges; if the difference is significant, the feature is most likely a sealed crack, as true cracks are not wide enough to have an inner edge after erosion. Paint patches are removed by assessing the rectangularity of bright objects and the straightness of their boundaries. Finally, expansion joints are detected for linearity by Hough transforming the pre-processed image. Line segments with an orientation of 0° ± 5° or 90° ± 5° are removed under the assumption that they have been intentionally incorporated into the pavement [25].

Pavement classification is accomplished by this group via a support vector machine (SVM). Features are extracted in both the spatial and frequency domain, and the first four statistical moments (mean, standard deviation, skewness, and kurtosis) are also extracted. The SVM algorithm categorizes pavements according to ten predefined classes, and pavement class drives parameter assignment in crack detection. Non-minimum suppression using four linear windows at interest points determines candidate crack pixels, and subsequent crack path growing is carried out to reconstruct pavement defects. This work proved successful in making use of non-crack pavement regions to make better-informed decisions about crack pixels on a road image. However, the data acquisition approach requires specialized equipment and results were not conclusively demonstrated for obscured or highly noisy cracks. Additionally, this work did not address crack classification to any extent.

Oliveira, et al. used a downward-facing camera to obtain grayscale images of road surfaces, with each pixel corresponding to approximately one square millimeter of road surface [26]. Pre-processing was accomplished by subdividing each raw image into a grid and computing statistical comparisons between image grids in order to determine regions which likely contain crack pixels. Each region was then cast as a point in a two-dimensional (mean-standard deviation) feature space and the feature space was normalized for offsets and skews. Normal pavement regions appear in the feature space as a cluster separate from crack region outliers (see Fig. 1.6). In order to determine an appropriate boundary for identifying crack kernels vs. normal kernels, a database of crack images manually tagged by an industry professional was analyzed in the same way. The decision boundary was established by a quadratic Gaussian classifier [26].

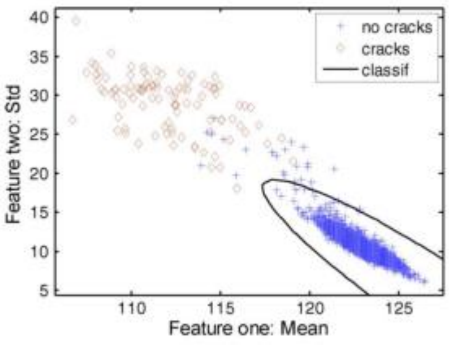


*Figure 1.6: Normal pavement regions appear in a cluster, whereas crack regions appear as outliers [26].*

After preliminary crack region identification, further steps were taken to determine finer crack characteristics. Cracks were determined to be either longitudinal, transverse, or miscellaneous based on average orientation. The single-pixel-wide skeletons of identified cracks were established to give approximate crack length in pixels. Finally, a thresholding and binarization operation was implemented to determine total crack area. The average width was then computed simply as the ratio of a crack's area to its length. This approach gave good results for detection and basic characterization of pavement cracks, but is conceptually incapable of detecting more complex cracking networks, for two reasons. First, if a significant portion of the surface area in an image contains crack pixels, there are not enough "normal pavement" regions to strongly establish a "non-crack" cluster in the feature space. Second, in the case of a multi-branched cracking network, the descriptions longitudinal and transverse have no real meaning, and the basic crack categories presented cannot accurately describe road condition [26].

Tanaka, et al. [23] describe a four-step approach involving multiple morphological and combinational techniques. First, to extract dark pixels, an image is thresholded and binarized such that a maximum of 30% of the image area is postulated as belonging to a crack. After this operation, saddle points are found via a "top-hat transformation": a normalization and thresholding of an image which is a closed version of the original. Closing is defined as a dilation followed by an erosion. The intersection of the saddle-point detection and dark pixel extraction is then used to find linear features based on directional path statistics. Finally, crack objects are connected via the combinations of further closing operations using different structuring elements [23]. This overall method is somewhat more abstracted from the pixel-wise procedures taken by others in its combinational and morphological operations, but still fully relies on the spatial domain. The restrictions of this method again include a likely inability to detect large cracking networks based on the 30% threshold, and a potential loss of some crack detail due to the binary intersection of multiple procedures.

A common problem in the noisy platform of road crack detection is fragmentation. When a crack is obscured by something in the raw image (often a protrusion in pavement aggregate), most pre-processing algorithms yield an image in which segments are falsely disconnected. Sun, et al. proposed a method by which such disjoint segments are

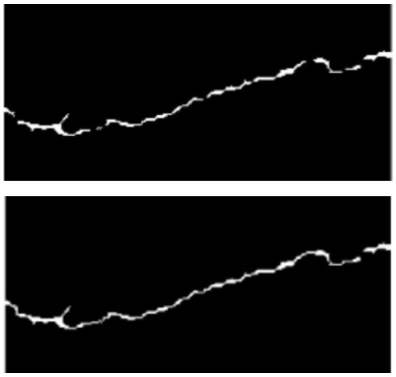


*Figure 1.7: Results of Sun, et al. crack connection algorithm [27].*

reconnected. After detecting cracks via a nonlinear thresholding and subsequent binarization, endpoints are detected as bright (crack) pixels for which other bright pixels can be found certain distances in the x- and y-directions [27]. These distances change for longitudinal and transverse cracks. When these endpoints are found, they are traced out until gaps are located and connected. Fig. 1.7 shows the results of this procedure. Although the pixel-wise method presented solved the crack gap problem for transverse and longitudinal cracks, it was not extrapolated to the case where connections must span diagonal gaps. Additionally, the scope of the advantages of this work was limited in description to only one aspect of enhancing already-detected cracks.

Other methods of crack detection have been approached from more mathematical and data-driven/heuristic perspectives. Tomikawa approached crack detection using a meta-genetic algorithm and template matching. With a rectangular pixel kernel template containing a vertical “U-shaped groove”, crack region candidates were assigned according to their fitness as determined by a genetic algorithm [24]. This method demonstrated a basic theoretical proof of concept for genetic algorithms in the context of crack detection, but the restrictiveness of its assumptions in terms of crack simplicity and orientation rule it out as a dominant player in the field today.

One of the most recent mathematically driven methods is that proposed by Amhaz, et al. [28]. This group took a minimal path selection approach which relied on the detection of crack endpoints to propose possible paths between endpoints as being cracks. The paths chosen correspond to a minimized energy (cost) function $E(C)$ which [28] defined in the most general case as

$$E(C) = \int_0^L [w_1\|C'(v)\|^2 + w_2\|C''(v)\|^2 + P\big(C(v)\big)]\, dv \qquad (1.15)$$

where $[0, L]$ is the domain over a crack with length $L$, $w_1\|C'(v)\|^2$ and $w_2\|C''(v)\|^2$ correspond to weighted contributions of the first and second derivatives at crack endpoints, and $P\big(C(v)\big)$ represents an external imposition on the function. This group chose to simplify the expression to a purely external restriction on the cost function, given as

$$c(p_{ij}) = \sum_{m=i}^{j} I(m) \tag{1.16}$$

where the cost function $c(p_{ij})$ of a path $p_{ij}$ is given as the sum over the intensity $I(m)$ of each pixel $m$ from endpoint $i$ to endpoint $j$ [28]. Endpoint candidates are chosen by dividing the image into a grid and choosing the lowest-valued pixel per grid, so long as the pixel is below a threshold. Once all path candidates have been established, their cost functions are normalized by dividing by path length and only paths with sufficiently low normalized cost functions are kept. In sum, crack detection is reduced to endpoint detection and optimal path estimation. This method proved highly capable of crack detection for a variety of crack images. However, due to the under-selective endpoint detection technique, many small artificial crack branchpoints were introduced, which require extensive subsequent cleaning. It becomes difficult, then, to decide which short branches are truly crack features, and which are false. Fig. 1.8 illustrates this issue. As the figure shows, spurious branchpoints occur at noisy crack boundaries. If image characteristics such as crack length or number of endpoints are to be extracted, these features will introduce significant uncertainty and may result in inaccuracy.

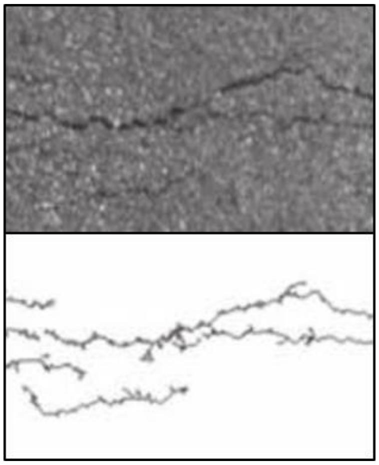


*Figure 1.8: Results of the minimal path selection approach [28].*

One of the more popular frequency-domain approaches to crack detection is that described by Subirats and Chambon, et al. [29], [30]. This method made use of a continuous wavelet transform (CWT) to isolate dark, linear (crack) features in a raw image. As defined by [30], the CWT $W_f(a, \vec{b}, \theta)$ of a signal $f(\vec{x})$ is given by

$$W_f(a, \vec{b}, \theta) = \frac{1}{a} \int \int f(\vec{x}) \psi \left( R^{-\theta} \left( \frac{\vec{x} - \vec{b}}{a} \right) \right) d\vec{x} \tag{1.17}$$

where $a$ and $\vec{b}$ represent scaling and translation, respectively, of a basis wavelet $\psi$ and $R^{-\theta}$ is a representation of the direction at which the transform is computed for a 2D function (in this case, the 2D image). The basis function chosen was a “Mexican hat”, illustrated in Fig. 1.9.

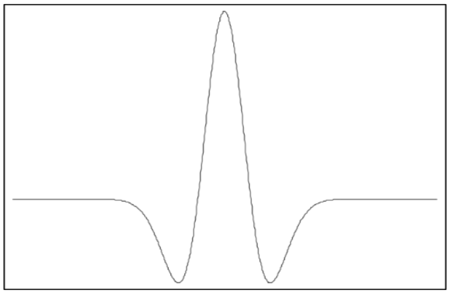


*Figure 1.9: Mexican hat wavelet chosen as continuous wavelet transform basis function by [30]. Public domain.*

The CWT was computed for angles of 0° and 90° in order to obtain a complex result, with the imaginary part corresponding to the 0° CWT and the real part corresponding to the 90° CWT. These operations were performed on the image through an image pyramid at various scales in order to attempt to capture thin cracks as well as wide ones. The phase and magnitude maps, $\varphi$ and $|Z|$, can be obtained using the real and imaginary components of the CWT in the following way:

$$\varphi = \tan^{-1}\left(\frac{im\left(W_f\left(a,\vec{b},\theta\right)\right)}{re\left(W_f\left(a,\vec{b},\theta\right)\right)}\right) \tag{1.18}$$

$$|Z| = \sqrt{im\left(W_f\left(a,\vec{b},\theta\right)\right)^2 + re\left(W_f\left(a,\vec{b},\theta\right)\right)^2} \tag{1.19}$$

For each scale, crack pixels are segmented as those whose phase is between -90° and -180° and whose magnitude is above a certain threshold. These heuristics are imposed by the form of the basis function used in the CWT [30]. Crack region candidates are then propagated through the scales to fuse crack hypotheses of increasing resolution. Subsequent skeletonization takes a morphologically erosive approach, and the resulting single-pixel-wide crack representation is used in an attempt to link segments together. Fig. 1.10 shows an example of the results given using this technique.

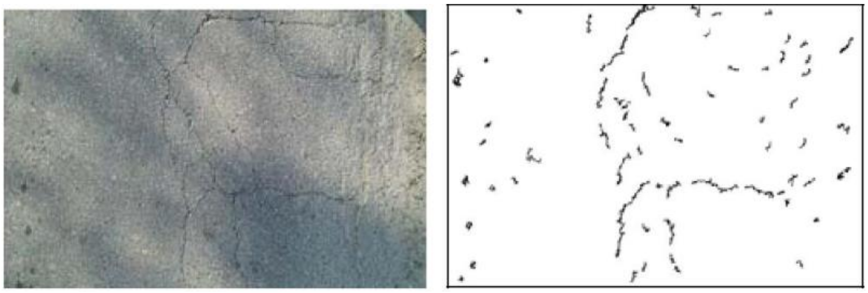


*Figure 1.10: Results obtained using the CWT approach [30].*

As the figure shows, one of the tremendous benefits of using a kind of frequency-transformed method is its resilience to low-frequency pixel intensity gradations such as the shadows seen above. However, the detection results show a significantly fragmented representation of crack regions. This loss of detail has adverse implications for accurate higher-level assessment of pavement distress which this work also does not address.

The most recent work in road crack detection uses the data-driven approach of convolutional neural networks (CNNs) to estimate crack pixels in an image. This work is described by Zhang, et al. in [31]. For a set of manually tagged training images, RGB pixel blocks of size 99x99 are extracted and labeled as either positive (crack is present in pixel block) or negative (crack is not present in pixel block). The center pixel in the corresponding block in the detected image is given a value of one if a crack is present for that block, and a value of zero otherwise. Positive blocks are chosen to have no more than 25% overlap and negative blocks are chosen to have no overlap, such that sufficient uniqueness is introduced in the training set. The pixel blocks are also randomly rotated at angles between 0° and 360° in order to remove orientation bias. A training set of 650,000 blocks is used to establish the four-layer-deep CNN, and the network is validated by 200,000 samples [31]. This technique boasts 92.5% recall rate and 87.0% precision, greatly improving upon other learning-based methods including SVM [32] and Boosting [33]. However, the scope of this method remains unclear, as results are only shown for crack images of relatively low severity and high contrast. Once again, despite using a technique generally associated with machine learning and classification, higher-level pavement distress analysis was not conducted.

Crack detection has seen applications in contexts apart from just road pavement. Sinha, et al. employed a morphological and combinatorial technique to detect cracks in underground concrete pipelines [34], [35]. This approach explored the statistical differences observed in different kinds of defects after application of an opening (erosion followed dilation) operation. Let $t\left(I_O S_c(r)\right)$ represent the number of remaining feature pixels in an opened image $I_O$ using a circular structuring element $S_c(r)$ of a certain radius $r$. The ratio of this value to the radius $r$ can be used to distinguish different classes of features (e.g. cracks, joints, holes, etc.). In addition to this information, two detection methods are used jointly to identify crack regions. The first kind of detector used, a ratio detector, compares the mean pixel intensity on either side of a crack candidate with the average center intensity; the ratio of these average intensities is used discriminatively to estimate crack regions. The second detector uses cross-correlation values across the adjacent and central pixel regions to determine crack likelihood. This approach was not well validated even for simple cracks, and the application to more textured surfaces than concrete would not see much success.

As this section shows, crack detection takes many forms and has been approached in a multitude of ways over the last three decades. To fully address each method would take an entire devoted book. For completeness, one final crack detection scheme will be discussed here, though like the framework of Sinha, et al., the theories developed by Cornelis, et al. [36] were not intended for road crack detection. Rather, the application for this work is in detection of cracks present on the surface of a painting, the *Ghent Altarpiece* (1432), for restoration purposes. Beyond the capabilities of any road condition monitoring system, this group had access to both infrared and x-ray imaging of the surface in question. In order to pre-process the images, the wood grain pattern behind the cracked paint needed removal, as it introduces background artifacts in the image. This was accomplished by removing dominant periodic elements associated with the regular gain pattern via a discrete cosine transform (DCT). Furthermore, in order to remove low-frequency intensity variation induced by wood support framework behind the painting, a blurred version of the image was subtracted from the original, in a manner similar to the Laplacian-of-Gaussian technique common in image processing [37]. Following this pre-processing step, a series of nine image filters are imposed upon the painting as imaged in the visible spectrum, infrared, and x-ray. For each pixel, the results of all filters composes a multi-dimensional feature vector. A Bayesian approach is then pursued in which a conditional probability tensor $P(Y|X)$, where $X$ is the set of feature vectors and $Y$ is the set of outcomes for each pixel (1 for crack, 0 for non-crack), is decomposed via higher-order singular value decomposition (HOSVD) [38]. This is to impose sparsity on the correlations between $X$ and $Y$. Other restrictions are also imposed to increase computational efficiency and eliminate redundancy. Using Bayes' Theorem, a feature vector $X$ maps to a class $y$ according to the following formula:

$$P\big(Y = y \big| X_1 = x_1, \dots, X_p = x_p\big) = \sum_{j_1=1}^{k_1} \cdots \sum_{j_p=1}^{k_p} \lambda_{j_1,\dots,j_p}(y) \prod_{m=1}^{p} \pi_{j_m}^{(m)}(x_m) \qquad (1.20)$$

where $p$ is the number of predictors (features), $j$ and $m$ are summation and product indices, respectively, and $\lambda$ and $\pi$ are chosen to be the Dirichlet priors [36]. The output of this detector is, rather than a binary map, a probability map where each pixel's value represents its probability of being a crack pixel. Fig. 1.11 shows the binarized results of this crack detection technique overlaid on the raw image in grayscale, next to a color version of the raw image.

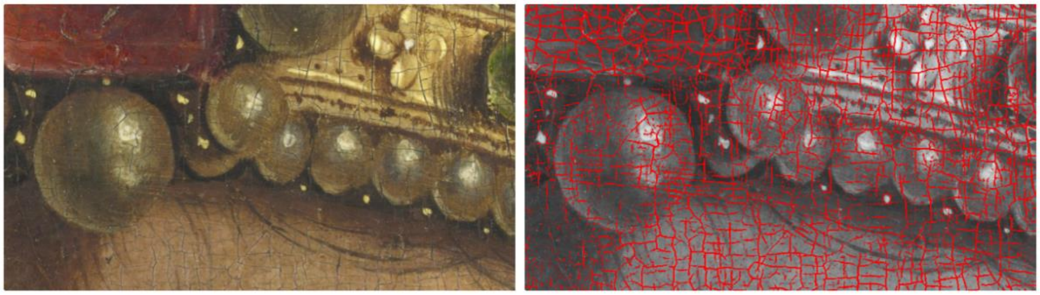


*Figure 1.11: Cornelis, et al. Bayesian predictive crack detection in paintings [36].*

As the figure shows, this probabilistic approach works remarkably well, especially considering that it is only partially supervised and runs at 10 seconds per 256x256 pixel patch, 1.8 times faster than the state-of-the-art in painting crack detection. However, some major caveats greatly restrict its application to road crack detection. Most importantly, x-ray imaging of road cracks is infeasible considering the expense of equipment and public irradiation danger. Secondly, although the method described above is fast compared to its competitor, processing an image covering a roadway with a resolution of 1 $mm^2$ per pixel would take over three minutes per image. The design of the above Bayesian detection technique caters to static detection, without time constraint.

## 1.5 Bayesian Classification

A common problem in machine learning, computer vision, and data mining is classification. Classification is a necessary part of recognition, perception, and understanding, especially with regards to robotics and computer science. In order to analyze sensory input at a high level of abstraction, a computer must be capable of acquiring data about a set of surroundings or an object of interest and be able to make decisions similarly to the way a human would. The aim of classification is to accomplish this objective accurately and efficiently. Classification methods have been used to approach crack detection, where detection is considered to be a matter of binary classification [39] (crack is present or not). Some examples of this, which have already been discussed, are seen in [26], [31], and [36]. These groups had the goal of classifying cracks into either non-crack or crack categories. However, little to no literature exists describing work in which cracks are classified on a higher level following detection. Nevertheless, methods do exist for solving such problems. This section will give an overview of one of the most prominent classification methods that have been well developed over the past few decades, Bayesian classification, and set the foundation for the novel approach described later on.

Classification relies on the extraction of useful descriptive features from a target object [40]. In vision applications, the feature vector can be composed of spatial domain information, frequency domain information, or more complex combinations of these descriptors. Oftentimes, geometrical properties, in both 2D and 3D, are useful in object classification. In auditory classification, spectral information most often contains the best kinds of quantized features to accurately describe a sound source [41]. Other times, data may not come from sensory input at all, but instead from large databases acquired for the purposes of learning and statistical analysis [42]. Examples include Facebook data mining, Google's targeted advertisements, medical databases, and so on. Descriptive features, or data, are used to formulate decisions about an object or observation either by creating decision boundaries in a hyperdimensional feature space or by training a network of hundreds to trillions of parameters to deterministically "trickle down" an intelligent outcome. Bayesian classification (among other methods) takes the first approach, whereas neural networks (which, composed of many variants, is often the method of choice in the state-of-the-art) take the second. This work currently does not incorporate neural networks, so detailed review of this approach is not developed here.

Bayesian classification has been investigated and improved upon since the mid-20th century [43]. The foundation of this technique, apparent in the very name, is Bayes' Theorem. In probability theory, Bayes' Theorem relates the conditional probability of one event (say, event $A$) given an event $B$ to the conditional probability of event $B$ given $A$ and the independent probabilities of the two events [44]. In mathematical form, this is expressed as

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{1.21}$$

where the notation $P(\cdot)$ refers to the probability of the enclosed event(s). This formula is inherently predictive in nature, as it can assign a probability to a conditional event, often an intuitively un-knowable event, given enough other information. This property is exploited in the use of Bayes' Theorem as a classifier. It allows the probability that a particular object $i$ with a set of attributes $\boldsymbol{z}$ belongs to a particular estimated class $\widetilde{w}_i$ to be derived from the probability that an object of a particular class has a certain set of attributes. Eq. 1.22 shows Bayes' Theorem with this notation.

$$P(\widetilde{w}_i|\boldsymbol{z}) = \frac{P(\boldsymbol{z}|\widetilde{w}_i)P(\widetilde{w}_i)}{P(\boldsymbol{z})} \tag{1.22}$$

In this form, $P(\widetilde{w}_i|\boldsymbol{z})$ is referred to as the posterior probability, $P(\widetilde{w_i})$ is the prior probability (representing the proportion of objects belonging to class $w_i$ in the population of interest), and $P(\boldsymbol{z})$ is a constant scaling factor to account for dataset bias. Functional variants of the Bayesian classifier arise from different estimations of the conditional likelihood $P(\boldsymbol{z}|\widetilde{w}_i)$, differently weighted cost functions of the posterior probability, and various assumptions or simplifications in the process.

In their explanation of Bayesian classifiers, van der Heijden, et al. begin with a conditional risk function defined in [45] as

$$\boldsymbol{R}(\widetilde{w}_i|\boldsymbol{z}) = \sum_{k=1}^{K} \boldsymbol{C}(\widetilde{w}_i|\, w_k)P(w_k|\boldsymbol{z}) \tag{1.23}$$

where $\boldsymbol{C}(\widetilde{w}_i|w_k)$ is cost the function associated with assigning a class $\widetilde{w}_i$ to an object with true class $w_k$ and $P(w_k|\boldsymbol{z})$ is the probability of correctly assigning the true class $w_k$ to an object, given a measurement (feature) vector $\boldsymbol{z}$. This is essentially the summation over all classes of the products of cost weightings with their associated class assignment probabilities. For example, if computer vision-based detection of a fruit yields a feature vector $\boldsymbol{z}$ that leads to 95% certainty that it is an apple and 5% certainty that it is an orange, risks can be given by the weighted sum of these probabilities according to a cost function such as the one given in Table 1.4.

*Table 1.4 Example cost function for fruit classification.*

| | Assigned class $\widetilde{w}$ | | |
|---|---|---|---|
| Actual class $w_k$ | | Apple | Orange |
| | Apple | 0 | 0.5 |
| | Orange | 0.75 | 0 |

Suppose an orchard sends its fruit to a distributor in a mixed vessel and the distributor wishes to automatically classify them as they pass along a conveyor belt. If oranges are more expensive than apples, the distributor loses more money by incorrectly including an orange in a batch of apples than by including an apple in a batch of oranges. Therefore, $\boldsymbol{C}(apple|orange)$ is given a value of 0.75 whereas $\boldsymbol{C}(orange|apple)$ is only 0.5. The diagonal terms of the cost table, $\boldsymbol{C}(apple|apple)$ and $\boldsymbol{C}(orange|orange)$, are given zero cost, as they correspond to correct classifications. In this example, the risk associated with assigning this particular fruit as an apple is given by

$$\boldsymbol{R}(apple|\boldsymbol{z}) = \boldsymbol{C}(apple|orange)P(orange|\boldsymbol{z}) + \boldsymbol{C}(apple|apple)P(apple|\boldsymbol{z})$$

$$= 0.75 * 0.05 + 0 * 0.95 = 0.0375$$

whereas the risk associated with assigning this fruit as an orange is given by

$$\boldsymbol{R}(orange|\boldsymbol{z}) = \boldsymbol{C}(orange|apple)P(apple|\boldsymbol{z}) + \boldsymbol{C}(orange|orange)P(orange|\boldsymbol{z})$$

$$= 0.5 * 0.95 + 0 * 0.05 = 0.475.$$

With a 95% probability that the fruit is an apple, the lowest risk is given by classifying the fruit as an apple. However, because of the unequal weighting of the cost function, if the vision-based detection and feature extraction of a fruit yields a confidence of 50% for both fruits, the fruit will be classified as an orange by default.

The objective of a Bayesian classifier is simply to establish decision boundaries in a multi-dimensional feature space in such a way as to minimize the risk of misclassifying an object. Stated mathematically, the estimated class to assign to a given object with a Bayesian classifier, $\widetilde{w}_{BC}$, is given by

$$\widetilde{w}_{BC} = \underset{\widetilde{w}_i}{\operatorname{argmin}}\, \boldsymbol{R}(\widetilde{w}_i|\boldsymbol{z}) \tag{1.24}$$

By substitution of 1.23, and assuming a uniform cost function (zeros on the diagonal and ones on the off-diagonal), this simplifies to

$$\widetilde{w}_{BC} = \underset{\widetilde{w}_i}{\operatorname{argmin}}[1 - P(\widetilde{w}_i|\boldsymbol{z})] \tag{1.25}$$

Now the problem becomes one of maximizing the conditional probability

$$\widetilde{w}_{BC} = \underset{\widetilde{w}_i}{\operatorname{argmax}}\, P(\widetilde{w}_i|\boldsymbol{z}) \tag{1.26}$$

By Bayes' Theorem, this can be rewritten as

$$\widetilde{w}_{BC} = \underset{\widetilde{w}_i}{\operatorname{argmax}}\, P(\boldsymbol{z}|\widetilde{w}_i)\, P(\widetilde{w}_i) \tag{1.27}$$

where the dataset bias $P(\boldsymbol{z})$ falls out since it does not affect the argmax due to its lack of dependency on $\widetilde{w}_i$. The conditional probability $P(\boldsymbol{z}|\widetilde{w}_i)$ and prior probability $P(\widetilde{w}_i)$ must be estimated. $P(\widetilde{w}_i)$ can be approximated fairly simply as the proportion of objects of class $w_i$ in the set of all possible objects. However, a variety of approaches exist for modeling $P(\boldsymbol{z}|\widetilde{w}_i)$. Oftentimes, a Gaussian distribution with mean vector and covariance matrix computed from the feature vectors is adequate and sufficient. However, other probability distribution functions (PDFs) have been proposed [46], [47], [48]. Among them are multinomial distributions, Bernoulli distributions, and Gaussian-mixture kernel density distributions. For conciseness of scope, only the basic Gaussian model will be discussed here. The formula for a Gaussian (normal) distribution in $M$ dimensions is

$$N(x;\mu,\mathbf{\Sigma}) = \frac{1}{\sqrt{(2\pi)^2|\boldsymbol{\Sigma}|}}e^{-\frac{1}{2}[x-\mu]^T\boldsymbol{\Sigma}^{-1}[x-\mu]} \tag{1.28}$$

where $\mu$ is the $M \times 1$ mean vector and $\boldsymbol{\Sigma}$ is the $M \times M$ covariance matrix. By substitution, it can be said that the correct class to assign to an object with feature vector $x$ is given by $\widetilde{w}_i$ where $i$ is found as

$$i = \underset{k\in K}{\operatorname{argmax}}\left\{\frac{1}{\sqrt{(2\pi)^2\left|\boldsymbol{\Sigma}_{w_k}\right|}}e^{-\frac{1}{2}\left[x-\mu_{w_k}\right]^T\boldsymbol{\Sigma}_{w_k}^{-1}\left[x-\mu_{w_k}\right]}P(w_k)\right\} \tag{1.29}$$

$K$ is the set of all possible classes, and the variable change to $k$ subscripts simply enables consistency with prior formulations. Because the natural logarithm is monotonic, taking the log of the argument of the argmax function does not change the function. Consequently, Eq. 1.29 can be rewritten as

$$i = \underset{k\in K}{\operatorname{argmax}}\left\{-\ln\left|\boldsymbol{\Sigma}_{w_k}\right| + 2\ln\bigl(P(w_k)\bigr) - \mu^T\mathbf{\Sigma}_{w_k}{}^{-1}\mu + 2x^T\mathbf{\Sigma}_{w_k}{}^{-1}\mu - x^T\mathbf{\Sigma}_{w_k}{}^{-1}x\right\} \tag{1.30}$$

This can be grouped into constant, linear, and quadratic terms in $x$ as follows:

$$i = \underset{k\in K}{\operatorname{argmax}}\{\alpha_k + x^T\boldsymbol{\alpha}_k + x^T\boldsymbol{A}_k x\} \tag{1.31}$$

where $\alpha_k = -\ln\left|\boldsymbol{\Sigma}_{w_k}\right| + 2\ln\bigl(P(w_k)\bigr) - \mu^T\mathbf{\Sigma}_{w_k}{}^{-1}\mu$, $\boldsymbol{\alpha}_k = 2\mathbf{\Sigma}_{w_k}{}^{-1}\mu$, and $\boldsymbol{A}_k = \mathbf{\Sigma}_{w_k}{}^{-1}$. This quadratic equation in $x$ shows that the decision boundaries in the feature space are constrained to being linear-type (lines, planes, and hyperplanes), quadratic-type (parabolae, hyperbolae, ellipsoids, and their hyperdimensional equivalents), or combinations of all or some of the above [45]. The form of the classifier described above is quadratic, but simplifications in the process (usually assuming a uniform diagonal covariance matrix) yields other forms, namely a classifier resulting in purely linear-type boundaries. Additionally, employing a Gaussian mixture model to estimate the conditional probability $P(\boldsymbol{z}|\widetilde{w}_i)$ (the kernel density distribution method), gives decision boundaries with finer resolution, though a closed-form solution cannot be derived as for the quadratic case.

Traditional applications of the Bayesian classifier require training data to establish decision boundaries (or more precisely, to compute the coefficients $\alpha_k$, $\boldsymbol{\alpha}_k$, and $\boldsymbol{A}_k$), and subsequent test data for actual classification. Oftentimes, test data are considered to belong to the class corresponding to the region in which the mean of the feature vector falls [45], though others have improved on this method by instead incorporating an approximate PDF for each datum in decision-making [49], [50]. It is important to fully incorporate the uncertainty of each observation in classification, if possible. This concept has gone largely unaddressed in prior work and literature thus far.

Upon thorough investigation of past work and the state of the art in the areas related to this research, it is clear that there is a need for a rigorous academic approach to the higher-level classification of road defects, namely pavement cracks. Such a method depends on accurate and human-like detection of crack features. Classification also ought to be modeled after the human thought process, making use of all available information to produce informed decisions about something as stochastic as road cracks. The necessity of a probabilistic approach is what drives the multi-Bayesian techniques implemented here. The most important outcome of this work, however, is a more generally applicable data-fusion-based method for detecting and classifying stochastic features of any kind, whether vision-based, audio-based, or data-based.

# Chapter 2. Approach

## 2.1 Crack Detection

### 2.1.a Data Acquisition and Algorithm Overview

Although it is not the focus of this thesis, the means by which road pavement data was collected is important to the overall framework of the research presented. The details are described more thoroughly by Hu, et al. in [51]. The methods described here use two downward-facing complementary metal oxide semiconductor (CMOS) cameras with adjustable shutter speed and frame rate controlled by a field programmable gate array (FPGA). The cameras are attached to the rear of a moving vehicle (for these experiments, a sports utility vehicle) in order to capture rapid-succession images of road surfaces for off-site processing. In order to capture images with sufficient resolution as well as field of view (FOV), the two cameras are positioned to cover approximately 2 meters of road width with about 20% overlapping FOV and a spatial resolution of ~1 mm$^2$ per pixel. See Fig. 2.1 for experimental setup. In addition, a global positioning system (GPS) and inertial measurement unit (IMU) are integrated into the vehicular road monitoring system. Off-site processing includes image stitching, pothole detection, crack detection, and map synchronization [51]. One unique advantage of the approach taken in this research is its high-speed data acquisition capabilities. The hardware listed above is able to capture road images at a frame rate of up to 150 fps. This ensures that each subsequent image shares some partially overlapping area with the image before. Scale Invariant Feature Transform (SIFT) [52] features from this overlapping area are used to perform image stitching for map integration. In addition, the CMOS cameras are capable of high shutter speeds,

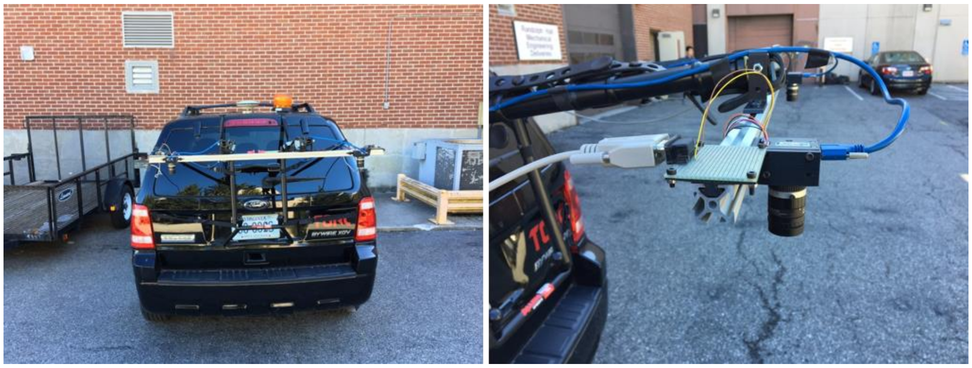


*Figure 2.1: Car camera setup [51].*

yielding exposure times as low as 0.006 ms. Consequently, motion blur is greatly mitigated, giving detailed images of road segments for analysis.

The sequence of image data is continuously saved in JPEG format to a hard drive for off-site processing. Although an uncompressed format would offer higher accuracy, it can cause problems with real-time acquisition. Additionally, the detail lost by compression is negligible at the scales considered here. In sum, the system is configured to accurately capture a wide area of road surface, with future plans to comprehensively map the entire width of road segments. The acquired road images are converted to 8-bit grayscale images for processing efficiency in crack detection; each grayscale image has a resolution of 1024x1280 pixels. Image processing is done primarily in LabVIEW 2017 (National Instruments, Austin, TX, USA). Corrective classification is done in Matlab 2017a (Mathworks, Natick, MA, USA).

A brief overview of the presented crack detection scheme is given in Fig. 2.2. The aim of crack detection is to reliably and accurately produce a feature vector for subsequent classification. This consists of several image processing steps. First, the raw images must be conditioned by pre-processing in order to obtain a cleaner representation of road information. The pre-processing method used here was specifically developed to selectively enhance dark and linear crack feature candidates. Object detection then simply aggregates possible crack regions for further analysis which includes noise removal and geometry filtration. In order to further simplify cracks for feature extraction, skeletonization is performed. Because this step often incurs loss of some important information, crack cleaning is a necessary correction for accurate assessment. This section focuses on detection only, whereas classification is addressed in the following section.

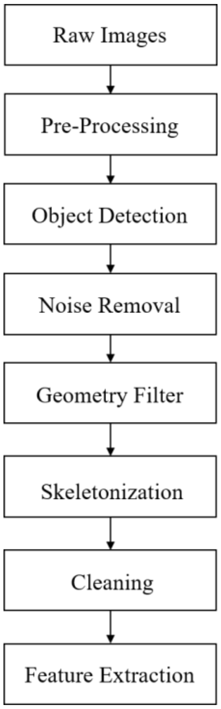


*Figure 2.2: Crack detection diagram.*

### 2.1.b Pre-Processing

The selective pre-processing technique implemented in this research makes use of raw image down-resolving before any other image processing takes place. A window of width $\tilde{x}$ pixels and height $\tilde{y}$ pixels samples the raw image (with width $\bar{x}$ and height $\bar{y}$) at row $i$, column $j$ (starting at the upper-left corner, $i = j = 0$). Let $\boldsymbol{p_{i,j}}$ represent the set of pixel values in this window; let $\boldsymbol{p}'_{i,j}$ represent the elements of $\boldsymbol{p}_{i,j}$ ordered from lowest (darkest) to highest (brightest). After the pixels have been ordered, the following operation is computed:

$$x_{k,l} = \left[\sum_{m=1}^{\lfloor N \rfloor} p'_m\right] + (N - \lfloor N \rfloor)p'_{\lfloor N \rfloor+1}\, , p'_m \in \boldsymbol{p}'_{i,j} \tag{2.1}$$

where $x_{k,l}$ is the value of a pixel in a new down-resolved image at row $k$, column $l$, $p'_m$ is the $m^{th}$ element of $\boldsymbol{p}'_{i,j}$, and $N$ is a decimal value chosen by the operator or optimized dependent on pavement type. In essence, the pixel of a new image corresponding to a block of pixels in the original image is given the value of the sum of the lowest $\lfloor N \rfloor$ integer pixel values plus the $(N - \lfloor N \rfloor)$ remainder percentage of the next-lowest pixel value. Since the image data is 8-bit and 0-indexed, a given pixel block's summation saturates at a value of 255 without wrapping.

The window that sweeps the raw image increments its indices according to a percent overlap specified by the user and can be tuned in order to optimize achievement or computation time. For example, if the window width is 4 pixels and the percent overlap in the x-direction is 25%, $i$ will increase in multiples of 3. However, indices $k, l$ are restricted to increment by steps of one in a nested loop structure to fill out the pixels of a new image. Let $x'$ and $y'$ be the width and height, respectively, of the new image and $1 - \delta_x$ and $1 - \delta_y$ be the percent overlap in the x- and y- directions. We can also define a down-resolution ratio $\alpha$ such that

$$\alpha = \frac{x'}{\bar{x}} = \frac{y'}{\bar{y}}. \tag{2.2}$$

This down-resolution ratio can be specified by the operator along with the percent overlaps. With this information and the given original image dimensions, the following equations determine the necessary window width and height:

$$\tilde{x} = \frac{\bar{x}}{\delta_x(x' - 1) + 1}, \qquad \tilde{y} = \frac{\bar{y}}{\delta_y(y' - 1) + 1}. \tag{2.3}$$

These adjustable parameters allow for varying precision and resolution in the final pre-processed image.

For images with uniform lighting and pavement conditions, it is sufficient to apply a single summation limit $N$ across the entire spatial domain. However, since the data acquisition mechanism does not incorporate controlled illumination but rather relies on ambient light (at this stage of work), the presence of uneven lighting conditions is highly likely. In particular, shadows of near-road objects including trees and telephone poles can create problems for this technique. Even transitioning pavement types could significantly affect the results. Consideration was given to employing a low-pass filter to remove crack features, where the resulting image was then subtracted from the raw image in order to isolate cracks on a uniform background. This method was deemed unsuitable for these purposes for two reasons: it added unnecessary computation time, and it was apt to wash out crack features in shadowed regions, causing a loss of important detail. Consequently, an improved method was investigated in which an image is segmented into a $P$x$P$ grid and the above procedures are executed on each image sub-domain. For dark image sub-domains, higher summation limits are needed to differentiate crack features from background. Conversely, lighter images require low limits. This inverse relationship allows for interpolative parameter assignment.

In order to assign the proper summation limit to each subs image sub-domain, the raw image is down-resolved by a factor of 16 and mean filtered. The down-resolving step is crucial to save time in the filtering step. The filtered image is then further down-resolved to the dimensions $P$x$P$. In this image, let $B$ be the value of the brightest (maximum) pixel and $S$ be the value of the darkest (minimum) pixel. The user specifies maximum and minimum summation values, $H$ and $L$, respectively. The value $L$ is assigned to the segment sub-domain whose average value is $B$, and likewise $H$ is used to pre-process the segment sub-domain corresponding to $S$. All intermediate segments' sub-domains' summation limits are determined by a linear interpolation according to the following formula:

$$y = \frac{(H - L)}{(S - B)}[x - S] + H \tag{2.4}$$

where $y$ is the limit assigned to a segment domain with average value $x$. In this way, the array of summation limits needs not be manually controlled; instead, only the maximum and minimum limits are given. Fig. 2.3 shows this process for $P = 8$, $H = 4.8$, and $L = 2.5$.

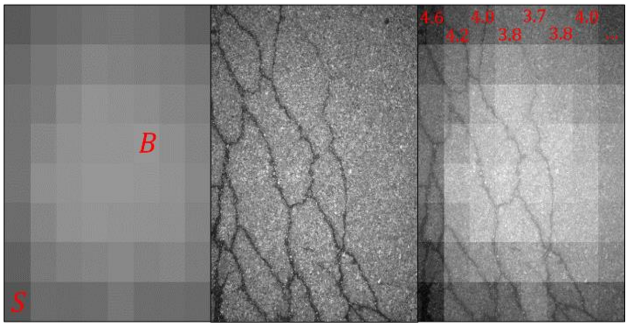


*Figure 2.3: Summation limits are determined from interpolation between darkest and brightest image sub-domains.*

Following the pre-processing procedures described above, each image undergoes an object detection operation to identify and characterize continuous regions of dark pixels (pixels that have not been saturated to a value of 255). Small spurious dark pixel groups not connected to crack features are eliminated by a simple object area threshold; objects with an area less than 100 pixels are removed. To further differentiate thin and lengthy features, both common characteristics of cracks, two additional geometry filters are implemented. First, each object's bounding rectangle's aspect ratio is computed; objects which are sufficiently thin and vertical or horizontal have an aspect ratio greater or less than one. Since orientation does not matter, values above one are inverted so that a single threshold can be used. Objects whose bounding rectangle's aspect ratio is greater than a specified threshold are eliminated; the threshold value that was experimentally optimized for this filter is 0.4. If an object meets this criterion, it is retained as a crack. However, due to the Cartesian bias of bounding rectangles, diagonal cracks or cracking networks cannot be identified by this method. Therefore, the second requirement (only assessed if requirement one has not been met) is that the pixel area of an object be sufficiently less than the area of its bounding rectangle. This eliminates thick and short objects, while retaining simple diagonal cracks and even more complex alligator cracking networks. Each object's area is divided by that of its bounding box, and the object is maintained if the ratio is 0.3 or less. This is illustrated in Fig. 2.4.

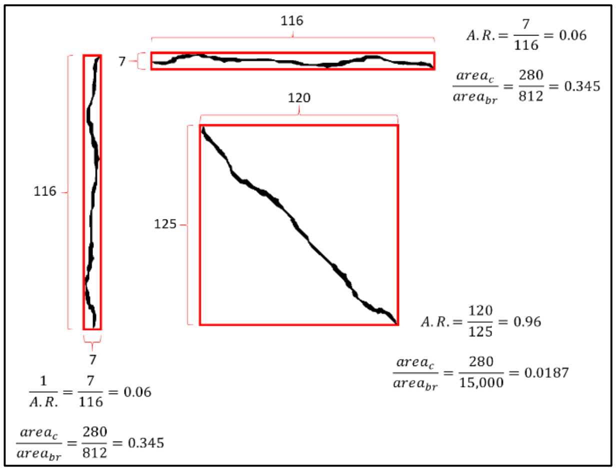

116
7
$A.R. = \frac{7}{116} = 0.06$
$\frac{area_c}{area_{br}} = \frac{280}{812} = 0.345$
120
116
125
$A.R. = \frac{120}{125} = 0.96$
7
$\frac{area_c}{area_{br}} = \frac{280}{15{,}000} = 0.0187$
$\frac{1}{A.R.} = \frac{7}{116} = 0.06$
$\frac{area_c}{area_{br}} = \frac{280}{812} = 0.345$

*Figure 2.4: Geometry filter criteria for isolating crack objects.*

The final result is a binary image with crack regions specified by pixel values of one and background regions designated with zeros. This can then be further reduced to obtain more basic information about a crack image, such as the number of endpoints, number of enclosed pavement objects, total crack length, average crack width, overall orientation, jaggedness, etc. (see future work). At this stage of work, an underlying assumption is clean road surfaces (i.e. no 3-dimensional debris is present on the road). The shadows of objects in the field of view may give inaccurate results, but that is also outside the scope of current research. In future work, false positives caused by leaves and twigs can be avoided by implementing a structure-from-motion rejection filter. Two-dimensional false positives can be identified by the jaggedness of their edges. Assuming diffraction has not diffused their borders, thin shadows occurring from roadside objects will not have the uniquely high-frequency jagged shape of true cracks. If such shadows are not well-defined, they do not pose a threat to this crack detection technique. Preliminary pre-processing results are shown in Fig. 2.5, though more thorough results will be given in the results section. These results were obtained with $\alpha = 0.247$, $\delta_x = \delta_y = 0.4$, and $N = 5.5$.

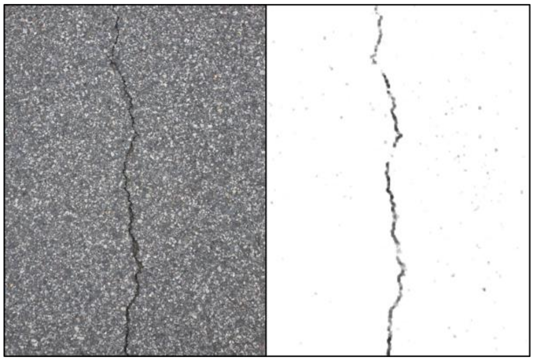


*Figure 2.5: Pre-processed crack example.*

In some cases, especially those in which pavement unevenness and low natural illumination angles cause small shadows to be cast across the entire image, unwanted dark artifacts may become present at the borders of true cracks. Although these effects can usually be mitigated by fine tuning of the summation limit parameter, such precision cannot always be reliably executed by an automated process. Therefore, in order to account for some variation from the ideal summation limit parameter, multiple summation limits in a small neighborhood are applied, and the resulting set of binary pre-processed images are used to correct one another in a form of sensor fusion. Fig. 2.6 gives an example of a pre-processed crack image for three different summation limits.

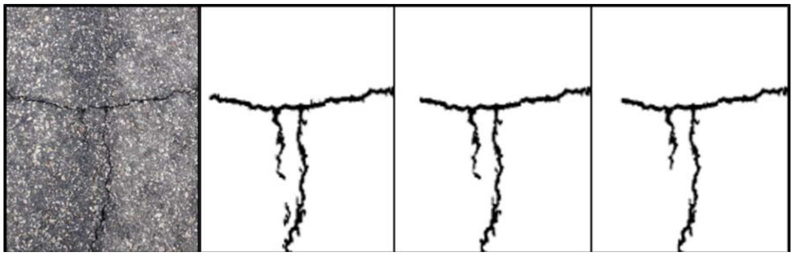


*Figure 2.6: Magnified view of pre-processed crack for three summation limits. From left to right: N = 7.75, N = 8, N = 8.25.*

Using multiple detections to establish crack truth has been explored by others in the area of crack detection. References [34] and [35]combine two different detection methods to increase certainty of crack features, and [23] considers a combinational logic approach to join results from four separate morphological operations. These techniques appear to work well, but the discrete nature of the detection fusion that they employ can sometimes eliminate important detail or allow false positives. As an alternative approach, Bayesian correction is considered. In order to implement Bayesian correction, the binary pre-processed images must be converted into a bitmap with the properties of a probability distribution function. To do this, the images are Gaussian blurred, an operation in which image pixels are convolved with a 2D Gaussian kernel of varying size (larger kernels correspond to more blur). The resulting image is one in which crack features' edges have been softened. This reflects the uncertainty associated with any given crack, in that falsely attributed pixels are most likely to occur at the edges of such features. Fig. 2.7 shows the magnified result of the Gaussian blur for two images in addition to the cross-section of the images at the red lines. Note that the images have been inverted to match the convention in which low pixel values are dark and high pixel values are light.

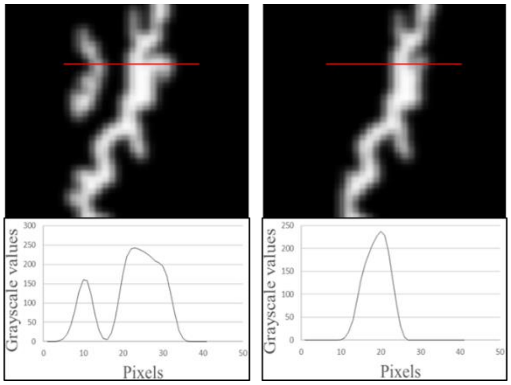


*Figure 2.7: Blurred crack images, cross-sections showing PDF nature of blur.*

In keeping with the definition of a PDF, the blurred image must be normalized (i.e. scaled such that its integration over the entire image space is one). Since only the functional form of the distribution yields the desired corrective behavior, normalization serves only to keep the crack image from saturating over several iterations. Correction now consists of the normalized multiplication of two or more pre-processed and Gaussian blurred images with slightly varying summation limits. Stated mathematically, let $p(\boldsymbol{x}^c|\tilde{\boldsymbol{z}}^c_{1:K})$ be the probability that a pixel array $\boldsymbol{x}$ corresponds to a crack $c$ given a sequence of $K$ particular measurements about that pixel array $\tilde{\boldsymbol{z}}^c_{1:K}$ (the sequence of binarized pre-processed images). The above reasoning leads to the following definition:

$$p(\boldsymbol{x}^c|\tilde{\boldsymbol{z}}^c_{1:K}) = \frac{\prod_{k=1}^{K} p(\boldsymbol{x}^c|\tilde{\boldsymbol{z}}^c_k)}{\int \prod_{k=1}^{K} p(\boldsymbol{x}^c|\tilde{\boldsymbol{z}}^c_k)\ d\boldsymbol{x}^c}. \tag{2.5}$$

After correction, the crack image is often scaled such that even the highest pixel values are on the order of $10^{-2}$ due to normalization. Consequently, the image is multiplied by a gain factor such that the brightest pixel is given a value of 255 to aid in 8-bit grayscale conversion. Fig. 2.8 illustrates the effect of correction on crack images. The two cross-sections from Fig. 2.7 are multiplied and normalized, and the result is scaled to a peak value of 255. Notice how the small spurious peak is greatly attenuated by the reinforced knowledge that a crack is not truly in that location.

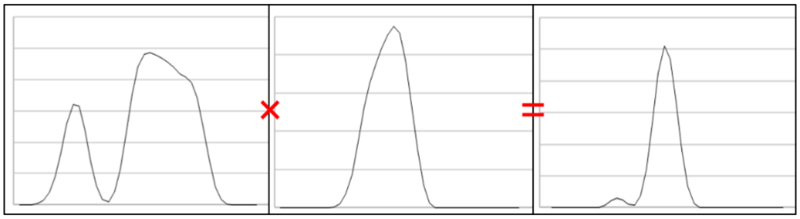


*Figure 2.8. Cross-sections of two crack PDFs fused via Bayesian correction to give more accurate PDF.*

A subsequent thresholding step allows the resulting corrected image to be binarized once more for further processing. The threshold level determines the width of the final pre-processed image. In order to preserve as many features as possible, the threshold is chosen to be as low as possible while still rejecting the darkest, but still non-zero (a consequence of the converging Gaussian), regions. After crack pixels have been established, the binary image is skeletonized and cleaned so as to obtain a single-pixel-wide representation of cracks for further characterization (see the following sections).

### 2.1.c Skeletonization

In order to easily extract useful information about the crack(s) contained in an image, the binary form of the crack(s) must be reduced to a single-pixel-wide representation; this is called skeletonization. This is a problem that has been approached many, many different ways over the past several decades [53], [54], [55], [56]. Some of the earliest methods accomplished skeletonization with a morphologically erosive thinning approach [53], [54], whereas other later techniques focus on binary object topography [55], [56]. For this work, both approaches were investigated and ultimately the best approach was found to be a combination of the two. The theory and application of these methods will be explained here in the context of crack feature reduction.

Morphological skeletonization essentially requires the erosion of binary crack edges until the only pixels left are those which fulfill certain requirements. The advantage of this method is its relative simplicity and computational speed. Also, it is always guaranteed to leave at least one pixel in a skeletonized object and will not break continuity of the object. However, the shape of most objects, especially thick ones, is easily distorted by the directional bias of this method. Jaggedness of shape borders often introduces artificial branches that are not present in an intuitive representation of the object. In addition, some valuable information about object size may be lost as the object is eroded. Nevertheless, this technique has some value for crack skeletonization which will be described below.

For each crack pixel in a binary image (cracks are given by “1”s and background is given by “0”s), the configuration of the surrounding 3x3 pixel block is analyzed. This is to reduce the possible combinations of pixel arrangements while still extracting the necessary information for deciding whether or not to eliminate (assign a logical 0 to) the center pixel. The consistent way to tell if a center pixel is essential to the crack skeleton is to check the pattern of the surrounding 8 pixels. By unwrapping these 8 pixels and checking the number of transitions that occur, it can be deduced whether or not the center pixel is a skeleton pixel. Non-skeleton pixels, those which occur at the edge of an object with width greater than one, will only ever have a maximum of two pixel value transitions. Skeleton pixels always have two or more transitions. When only edge pixels are iteratively removed from an image based on transition number discrimination, eventually the only pixels that remain will belong to an object’s skeleton.  Fig. 2.9 shows how this is done; center pixels whose adjacent pixels have 2 or fewer transitions are eliminated. Note that, although their values are one, crack pixels are represented as dark in keeping with reality.

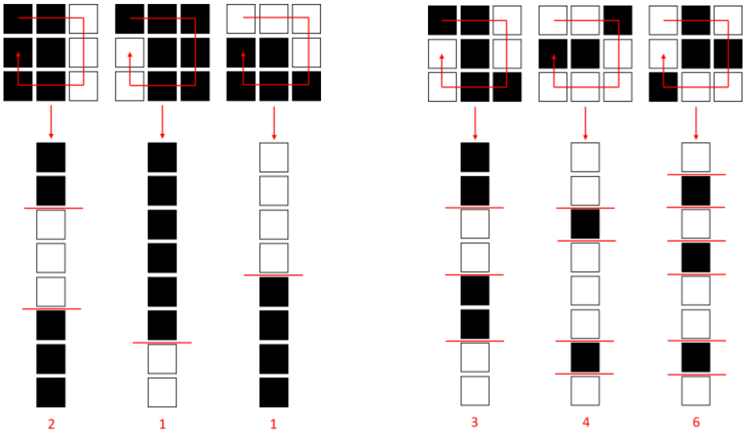
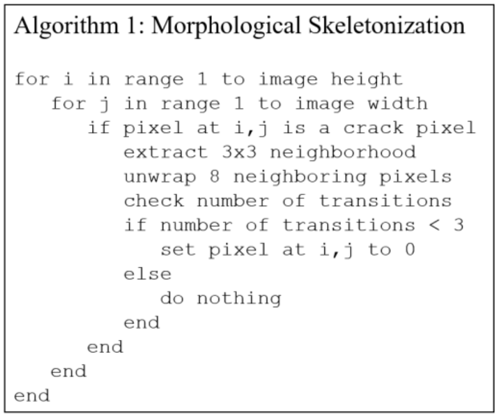


*Figure 2.9: Three examples of edge pixel kernels and three examples of skeleton pixel kernels.*

Pseudo-code of the morphological skeletonization method is provided in the box for Algorithm 1. Note the pixel scanning sequence, as it effects the viability of this technique.



Algorithm 1: Morphological Skeletonization

```
for i in range 1 to image height
   for j in range 1 to image width
      if pixel at i,j is a crack pixel
         extract 3x3 neighborhood
         unwrap 8 neighboring pixels
         check number of transitions
         if number of transitions < 3
            set pixel at i,j to 0
         else
            do nothing
         end
      end
   end
end
```

This technique does an excellent job of consistently reducing all linear features to their single-pixel-wide skeleton representations. However, because pixels are scanned in reading order (upper left to lower right), features get eroded preferentially on the left and top sides. In addition, some vertical features are reduced to skeletons which are unrealistically straight. Finally, and most critically, thick and especially jagged objects tend to introduce false branches, a common side-effect of this variety of skeletonization. Fig. 2.10 shows these consequences on the binarized version of the pre-processed image given in Fig. 2.5. As the figure shows, certain segments of the crack have been incorrectly straightened, and the thickest sections have 1-2 layers of induced false branches. Nevertheless, the algorithm has strictly imposed the rule that no multi-pixel wide segments are present anywhere in the image.

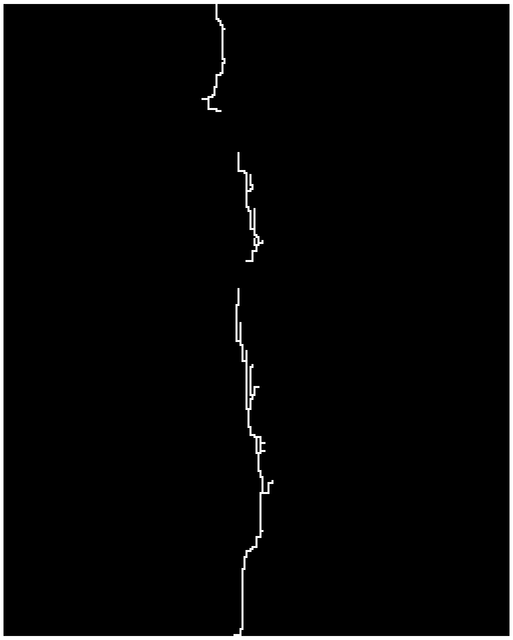


*Figure 2.10: Skeletonized crack using algorithm 1.*

As is fairly evident, there is significant uncertainty in the attributes of a crack skeletonized in this manner. Arbitrariness of crack features is introduced by the morphologically erosive approach. As it happens, these errors can be improved upon by down-resolving the binary image before skeletonization, as this reduces the sensitivity of the pixels to erosion by reducing the degrees of freedom in a sense. It becomes clear, then, that the true features by which a road crack can be described can only be approximated by skeletonization, with error introduced by the sensitivity of the algorithm. This uncertainty is of great interest in the formulation of corrective classification described in section 2.2. In keeping with the focus of this section, however, a subsequent skeletonization approach was considered which significantly improves on induced error.  This approach uses a topographical transformation to extract crack skeletons.

Topographical skeletonization is explored by Guo, et al. and Wan, et al. [55], [56], though the technique used and described here was uniquely developed to give accurate estimations of crack features specifically. The first step in this topographical skeletonization algorithm is to blur the binary image by convolution with a Gaussian kernel [57] such that crack cross-sections resemble Gaussian mixtures, but not so severely that edge detail is over-smoothed; this was described above. Fig. 2.11 shows the effect of blurring on the crack of Figs. 2.5 and 2.10. By blurring cracks in an image sufficiently, it can be known with near certainty that all local maxima in at least one direction belong to the crack skeleton. This works well for thin features like cracks, but it should be noted that objects with small edge length-to-surface area ratios may not be characterized as well using this method. The efficacy of the topographical approach is largely dependent on context and desired shape conservation.

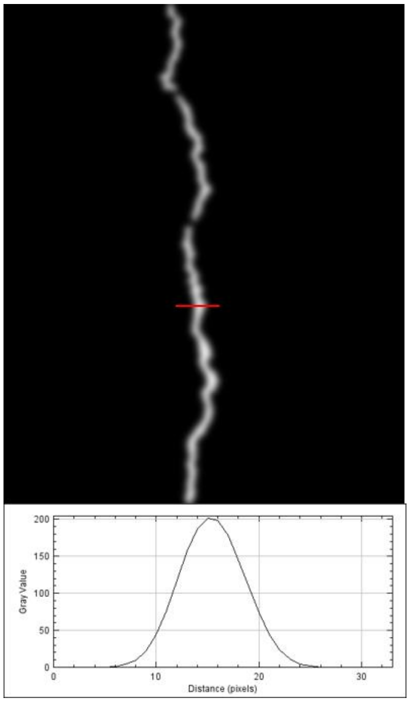
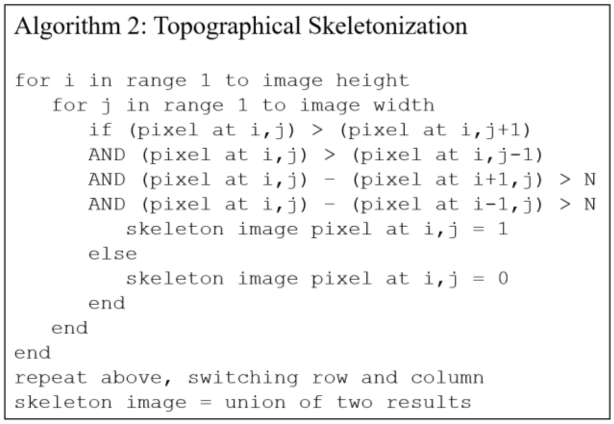


*Figure 2.11: Gaussian-blurred crack and profile.*

To extract skeleton pixels from the Gaussian-blurred image is in essence a form of non-maxima suppression, a common procedure in image processing [58]. A common method of non-maxima suppression extracts the eight surrounding pixels around a given pixel of interest and interpolates the local field. If the field gradient along any direction passing through the pixel of interest is zero, the pixel is recognized as a local maximum and incorporated into the crack skeleton [59]. This method works well for edge detection, but it requires magnitude and orientation information, usually obtained via the image gradients required in edge detection. Since only the Gaussian-blurred crack image is available, this non-max suppression method is overcomplicated. Instead, a very simple yet effective “homemade” non-max suppression algorithm was implemented. Because cracks cannot be assumed to be generally vertical or horizontal, both the rows and columns of the blurred image are scanned, and locations for which a given pixel is greater than both the pixels on either side are tagged as possible skeleton pixels. However, since this may also be true at a "micro" scale in the dark regions beyond the endpoints of a crack due to the fact that blurring "bleeds" non-zero values into background regions, an additional check is also needed. The pixel of interest must also be sufficiently greater than the pixels on either side orthogonal to the scanning direction. This procedure is executed across the rows and columns of the blurred image, and the skeletonized image is taken as the union of two results. See the pseudo-code provided in the box for Algorithm 2.



Algorithm 2: Topographical Skeletonization

```
for i in range 1 to image height
   for j in range 1 to image width
      if (pixel at i,j) > (pixel at i,j+1)
      AND (pixel at i,j) > (pixel at i,j-1)
      AND (pixel at i,j) - (pixel at i+1,j) > N
      AND (pixel at i,j) - (pixel at i-1,j) > N
         skeleton image pixel at i,j = 1
      else
         skeleton image pixel at i,j = 0
      end
   end
end
repeat above, switching row and column
skeleton image = union of two results
```

The major advantage of this technique is its preservation of crack shape and resilience to false branches. However, the direct results of the algorithm as presented are fairly fragmented and prone to noise. This is because blindly scanning the image in the x and y directions for maxima often results in the inadvertent slicing of a curved crack tangential to the path rather than normal to it. Because of the isotropic blurring of the crack image, such a slice would also yield a local maximum. This is illustrated in Fig. 2.12. Crack jaggedness will cause these locations to be flagged as belonging to the crack skeleton. In many cases, these points are isolated from the main crack body as spurious pixels which can be fairly easily removed without much harm to the true crack. Nevertheless, there are inevitably places where these points connect to the crack body, introducing false branches (albeit much less severe than those introduced by the morphological method). Fig. 2.13 presents the raw result of the topological approach. As the figure shows, there are a number of disjoint regions in the crack body. The figure also points out false branches and spurious pixels. Each of these three errors is dealt with in a different way.

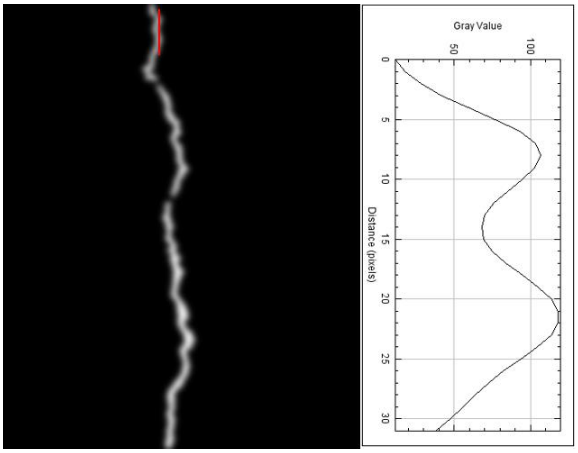


*Figure 2.12: Erroneous local maxima from jaggedness.*

Spurious pixels are removed with a simple noise removal filter. False branches will be addressed in the crack cleaning section. Since it directly pertains to establishing the crack skeleton, gap closing will be addressed here.

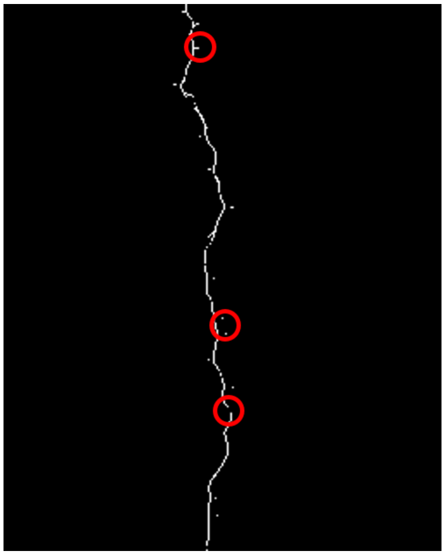


*Figure 2.13: Raw topographical skeletonization results.*

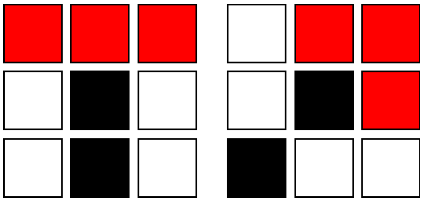


*Figure 2.14: Red pixels are new potential endpoints, and black pixels are current ultimate and penultimate pixels.*

The gap closing technique implemented at this stage was described by Fernandez [60]. First, endpoint pixels are detected as those crack pixels for which only one of the eight neighbor pixels is also a crack pixel. The direction of an endpoint is estimated by the relative orientation of the ultimate and penultimate pixels. Candidate pixels for end growing are determined as the three pixel neighbors in the current crack end direction. In other words, the pixel opposite the penultimate pixel in a 3x3 kernel and its two direct neighbors are considered (see Fig. 2.14). Of the three candidate pixels, the one with the maximum value in the blurred image is chosen as the next endpoint. This is intended to converge the crack end to the center line of the crack. In this way, end growing can be thought of as a trickle of water flowing down a gorge; the leading edge will always prefer a path which minimizes potential energy. If this operation was unbounded, cracks would either grow across the entire image without restraint or form a loop. Consequently, the operation is halted if endpoints reach the edge of the binary mask developed from pre-processing. In the case of loops, there is not much issue since closing a loop eliminates the endpoint and the crack is no longer grown from that end. Alternatively, if a gap such as the one in Fig. 2.13 is bridged, the endpoint is also eliminated and the goal of gap closing is accomplished. Algorithm 3 presents the general

Algorithm 3: End Growing

```
for i in range 1 to image height
   for j in range 1 to image width
      if pixel at i,j is an endpoint
      AND pixel at i,j is within binary mask
         extract 3x3 kernel
         identify penultimate pixel
         extract pixel values of three opposite
         pixels from blurred image
         pixel at max of three opposite pixels = 1
      end
   end
end
repeat sufficient number of times such that no
change occurs
```

procedure for gap closing via end growing. As Fig. 2.15 below shows, the algorithm is extremely effective in automatic crack skeleton repair without distorting the shape or size of the crack. Computational speed is 200 ms or less on a non-dedicated CPU with 12 GB RAM without optimization of any kind. There are two loops at one of the larger gaps, a sometimes-inevitable result of image variability. Two other loops result from paths taken by false endpoints that did not follow the original skeleton path. Artifacts like this are addressed in the following section.

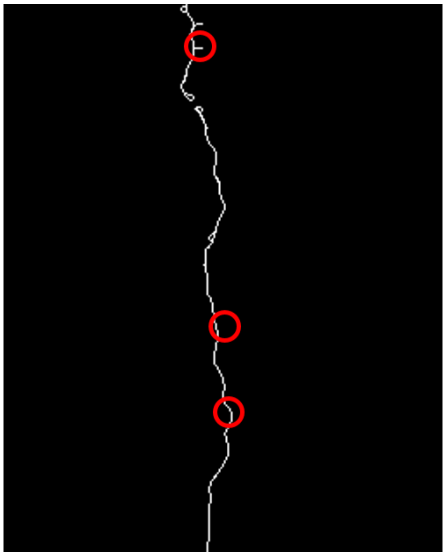
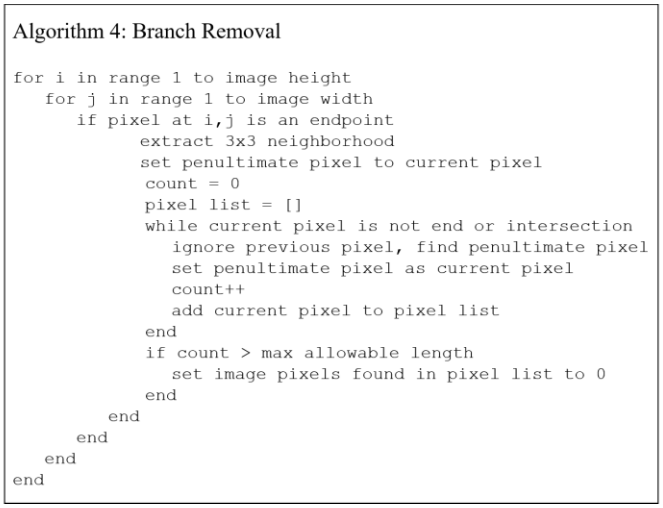


*Figure 2.15: Noise removed and gaps closed.*

### 2.1.d Cleaning

Loops and false branches are the remaining features requiring removal in order to clean crack skeletons. To do this, loops are first transformed into branches and all sufficiently small branches are eliminated. First, dark objects (pixels with values of zero surrounded by bright pixels) are detected and the corresponding pixels are reassigned to the logical high value. In essence, the loops are filled. The resulting features are multi-pixel-wide lobes that protrude from a crack skeleton. These lobes must be reduced to single-pixel-wide branches. The morphological skeletonization technique is employed in this operation because of its ability to handle features with relatively small width and area. For loops incorporated into a crack skeleton, no further processing is required. Loops at true crack endpoints are essentially reduced to the endpoints that existed before the end growing step. Loops which protrude from the sides of cracks are reduced to short false branches. Only the latter case is considered a defect and requires removal.



Algorithm 4: Branch Removal

```
for i in range 1 to image height
   for j in range 1 to image width
      if pixel at i,j is an endpoint
            extract 3x3 neighborhood
            set penultimate pixel to current pixel
            count = 0
            pixel list = []
            while current pixel is not end or intersection
               ignore previous pixel, find penultimate pixel
               set penultimate pixel as current pixel
               count++
               add current pixel to pixel list
            end
            if count > max allowable length
               set image pixels found in pixel list to 0
            end
         end
      end
   end
end
```

Algorithm box 4 provides pseudo-code for branch removal. This requires a pixel-wise tracing of each endpoint from tip to tip or tip to intersection, in order to count the pixel length of each branch. If the length of any given branch is too small according to a threshold adjusted by the user, that branch is removed by setting its pixels to zero in the skeletonized image. The resulting image presumably contains only true crack skeleton pixels.

In some cases, pavement texture or asphalt aggregate obscures a portion of the crack of interest. This causes a fracture in the crack object from the very first pre-processing stage, and since the results of this stage mask the bounds of end growth, any breaches in crack features from pre-processing onward cannot be fixed by the gap closing described in the previous section. However, it can be reasonably assumed that the two endpoints resulting from such a break have two distinguishing characteristics setting them apart from any other combination of endpoints. First, the distance between the two ends is likely relatively small; second, the ends will point towards each other. The first of these criteria can be relatively easily checked by simply computing the $L_2$ norm of the difference between endpoint coordinates. The second, however, requires more knowledge about relative endpoint orientation.

To consider whether or not two endpoints are compatible, the Euclidean distance is first computed. If the two points are too far from each other, they are rejected as a pair. Otherwise, each end is traced back three pixels and the approximate end slope is given as the slope between these two pixels. There are four possible end types, each characterized by the direction in which they point: type one ends points northeast, type two ends point southeast, type three ends point southwest, and type four ends point northwest. Orientation and relative location are used to determine what ends match. For example, if a type two end is found northwest of a type four end, the two are pointing towards each other and are likely compatible. However, any two ends of the same type are not compatible regardless of relative location. Table 2.1 lays out each case for which two endpoints would be considered compatible. The contents are read as “endpoint 2 must be _ of endpoint 1”.

*Table 2.1: Pairings for possible endpoints to be connected.*

|  |  | Endpoint 1 |  |  |  |
|---|---|---|---|---|---|
|  |  | Type 1 | Type 2 | Type 3 | Type 4 |
| Endpoint 2 | Type 1 | - | S | SW | W |
|  | Type 2 | N | - | W | NW |
|  | Type 3 | NE | E | - | N |
|  | Type 4 | E | SE | S | - |

If two endpoints are found to be compatible, they are connected by a straight line of positive pixels. Although it is almost certain that the true crack does not follow a straight line across that gap, since the distance is relatively small, a straight-line approximation works well. The pseudo code is presented in the box for Algorithm 5. Occasionally, for more complex cracking networks, this step introduces additional loops in the crack skeleton. As a result, the loop filling and re-skeletonization steps described earlier are repeated with minimal increase in computation time. Repeating these cleaning operations just a few times very quickly converges the crack skeleton to its final estimated form. Fig. 2.16 shows side-by-side the results of raw skeletonization, end growing, and crack cleaning. In comparison with the morphological skeletonization results from Fig. 2.10, this skeleton is far superior in both accurately estimating crack shape and avoiding false branches. The end result shares near perfect similarity to human ground truth. This is of great importance when it comes to classification, since the approach used in this research relies on “engineered” feature extraction for accurate crack severity assessment. It is important to note at this

Algorithm 5: Gap Closing

```
for i in list of endpoints
   for j in list of endpoints
      if i ≠ j
         compute distance between endpoints i and j
         if distance < threshold
            extract orientation of i and j
            extract relative position of i and j
            if compatibility is satisfied
               draw line between i and j
               remove i and j from list
            end
         end
      end
   end
end
```

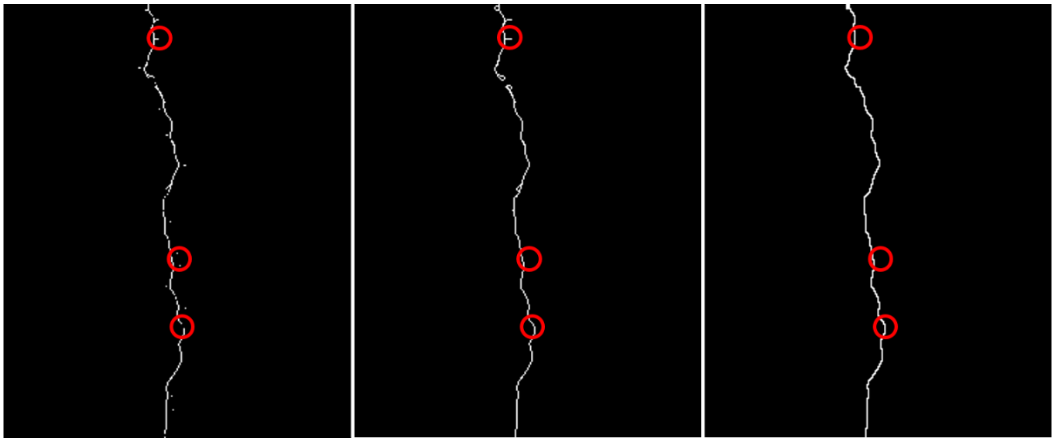


*Figure 2.16: Results of each stage of topographical skeletonization.*

juncture that the crack detection methods developed were explicitly designed to avoid any and all assumptions about crack form, thus working equally well for complex cracking networks as for simple longitudinal cracks like the one shown throughout this chapter.

## 2.2 Crack Classification

### 2.2.a Feature Extraction

Whereas the leading edge in deep and unsupervised learning techniques (e.g. deep neural networks, convolutional neural networks, fully convolutional networks, etc.) do not require "engineered" feature extraction, the classification techniques used for this research intentionally rely on manually targeted features for characterizing cracks. The reasoning behind this is twofold. First, there is explicit knowledge of the kinds of objects that are being detected: road cracks. The classification scheme needs not be fully equipped to recognize the full breadth of human perception. Second, because of the relatively narrow scope of interest, training can be made much more efficient and effective by extracting and using only the features that have a significant bearing on crack severity as determined by human visual inspection. For basic quadratic Bayesian classification (QBC; see section 1.5), a two-dimensional feature set was chosen. As the Bayesian classification theory previously developed stated, the feature space is not limited to two dimensions; however, it happens that if the features chosen are highly descriptive and independent, results are quite good. From a practical standpoint, a two-dimensional feature space is also much easier to visualize and present.

In order to most meaningfully utilize the information available in a single image, it is important that the features selected for classification be as mutually independent from one another as possible. For example, if someone was given the task of describing their friend with only three statements such that a stranger could recognize them, it would be much less efficient to use the words “male, dark eyebrows, and dark beard” because many males have dark hair and almost all males with dark eyebrows also have dark facial hair; the parameters are highly correlated, or dependent. The group of people fitting those descriptions is far larger and less unique than a group fitting the description, “male, 5 feet, bald”. With this idea of independence in mind, the two descriptive parameters selected for crack detection were the number of endpoints and the number of crack enclosures (i.e. the number of pavement “objects” completely disconnected from the continuous strip of road pavement). These two attributes were considered independent since a cracking network can have many endpoints with few objects, and the converse is also readily conceivable. In addition, a crack may have many or few of both features. Other features considered were crack length, crack jaggedness, object size and shape, average crack width, crack centroid, average crack orientation, etc. Incorporating these into a higher-dimensional QBC is quite possible and may be insightful with regards to decision-making. This will be explored in work in the near future.

Extracting the number of endpoints and objects is relatively simple. Endpoint extraction was described in section 2.1.c; the crack pixels in a skeletonized image for which only one neighbor is also a crack are endpoints. The number of objects is computed by continuity checks between dark pixels in the skeletonized image. The minimum number of objects is one, as even a crack-less road image has one continuous region of pavement. The minimum number of endpoints is zero (no crack present, or crack endpoints are out of view). Intersections between cracks and image edges are not counted as endpoints, as it is assumed that cracks do not end as they move out of view, but are only obscured. For the same reason, objects at image edges are counted. Objects that are sufficiently small (for instance, ten pixels or less) are not counted, as they are most likely introduced by noise and error rather than being true detectable pavement enclosures.

### 2.2.b Uncertainty Modeling

The heart of this work is probabilistically overcoming the uncertainty inherent in detection of noisy and stochastic features. In order to do this, the uncertainty in each detection event (road image) must be estimated by some means. This took two slightly different forms over the course of this work, both of which will be discussed here, but only the latter of which was accepted for the long term. The first uncertainty estimator made use of the arbitrariness with which the morphological skeletonization technique presented features for extraction. As was discussed before, the resolution of the binary image for skeletonization makes a significant difference in the accuracy of this method. Fig. 2.17 makes this quite clear.

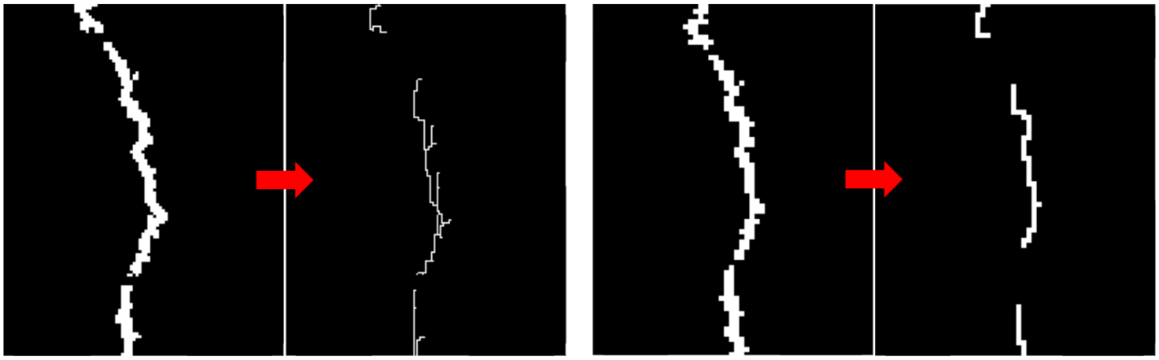


*Figure 2.17: The effect of resolution on morphological skeletonization accuracy.*

Higher resolution images have thicker features with more intricate edges, both of which cause problems with false branches and shape loss in morphological skeletonization. However, for low-resolution images, crack detail will be lost to such an extent that feature extraction is altogether unreliable. It is reasonable to expect similar measurement uncertainty in any stochastic feature classification context, which gives broader urgency and impact to the methods described here.

Assuming a particular crack image is assigned the ideal summation limit in pre-processing, the arbitrariness with which the skeletonization and cleaning operations are able to characterize the crack image can be approximated by extracting the two essential features (number of ends and number of objects) under various resolutions. A training set of 107 crack images of varying severity and pavement type was analyzed such that pre-processing and detection yielded binary crack images with a resolution of 253 x 316 pixels and skeletonization was performed for down-resolution ratios ranging from 0.05 to 1.00. The cracks were manually classified as being longitudinal/ transverse (LT), light alligator (LA), moderate alligator (MA), or severe alligator (SA), four broad categories loosely based on standard road crack descriptions given by industry experts. Fig. 2.18 gives

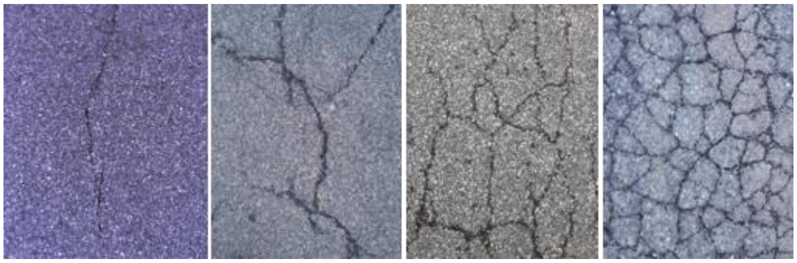


*Figure 2.18. From left to right: LT, LA, MA, and SA cracking networks.*

examples of each crack type. The classification features were detected and averaged for each of four classes across each down-resolution ratio.  Fig. 2.19 shows the number of objects and endpoints plotted as a function of down-resolution ratio. The standard deviation between the four classes per ratio is plotted with the red dashed line, and the product of the standard deviations for both features is given as the interclass variance.

This analysis shows that the ideal down-resolution ratio for class distinction is 0.95 (a resolution of 240 x 300 pixels), as interclass variance is maximized at this resolution. Additionally, since this can be performed on each image with reasonable computation speed, any test image can be represented not as a single point in the feature space, but rather as a Gaussian distribution reflecting the uncertainty of each measurement. The mean vectors and covariance matrices are obtained by the measurement data for the two features over down-resolution ratios from 0.75 to 1.15 (centered at the ideal ratio of 0.95). By modeling the uncertainty in sensor measurement as a Gaussian PDF, corrective classification is readily achievable, though this will not be discussed until sub-section 2.2.e.

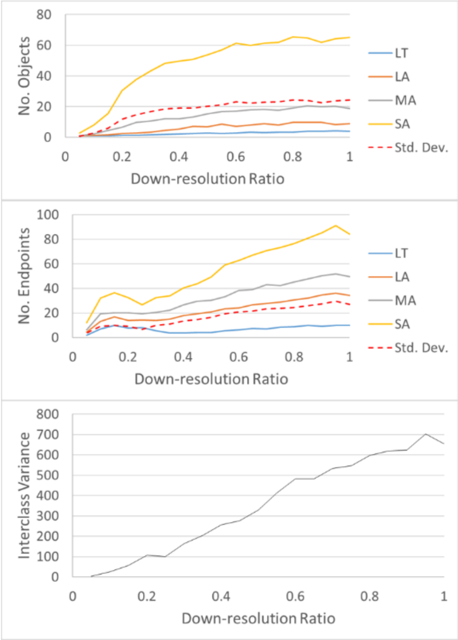


*Figure 2.19. Effect of resolution on uncertainty of crack features.*

Because it was ultimately clear that morphological skeletonization was not ideal for accurate feature extraction, the above uncertainty modeling was replaced by a model for topographical skeletonization. Topographical skeletonization is more or less invariant to image resolution, so a different parameter was needed to assess the arbitrariness of detection. Since a critical step in this technique is Gaussian blurring, and the blur radius can significantly affect skeleton quality by either over- or under-smoothing, features are instead extracted over a range of blur radii and averaged for each class similarly to the description above. Fig. 2.20 gives the inter-class variance results for this analysis. As the figure shows, blurring with a radius higher than 20 pixels decreases the ability to distinguish between classes. Although blur ratios below 20 are not shown in the plots, observation showed that significantly less blurring caused much more severe defects in skeletonization. As a result, the mean and covariance for modeling each crack image as a PDF in the feature space are given by the feature vectors over blurring radii from 15 to 25 (centered at 20, the ideal radius).

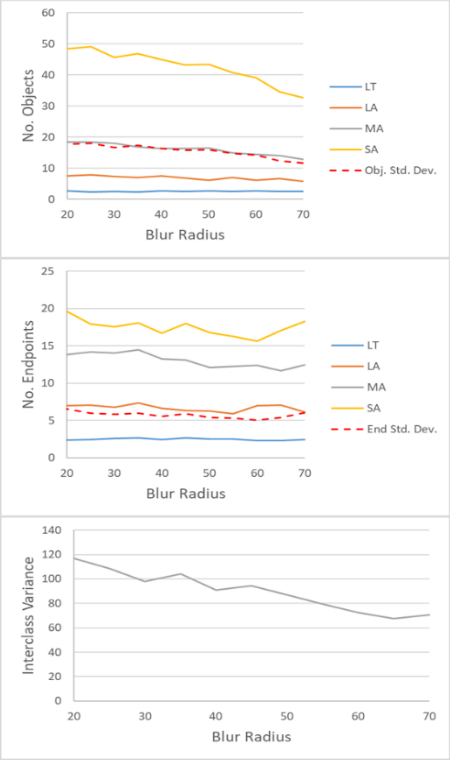


*Figure 2.20. Effect of blurring on uncertainty of crack features.*

2.2.c Quadratic Bayesian classifier training

The set of 107 crack images were scored according to their detection quality as manually determined by the operator. Those images which had the fewest false crack feature deletions or additions due to noise were weighted more heavily by means of duplication in the data set. After weighting, the training data set consisted of 211 points, of which 80 were LT (32 unique), 38 were LA (26 unique), 53 were MA (26 unique), and 40 were SA (23 unique). The mean vector and covariance matrix of each class were used to compute the QBC coefficients of Eq. 1.31, and the decision boundaries formulated according to the theory developed in chapter 1. The boundaries are seen as color changes between classes in Fig. 2.21. 92% of all training data used to train the QBC were correctly reclassified by the system. The misclassified 8% can be explained by the highly transitional nature of crack classification; since road cracking is highly stochastic, the boundaries between classes are largely arbitrary and dependent on the subjective human definition of a class. This “softens” the decision boundaries, adding leniency to the classification problem. The next section also addresses the principle that a crack often cannot be assigned a hard category and seeks to remedy this issue by implementing Bayesian correction.

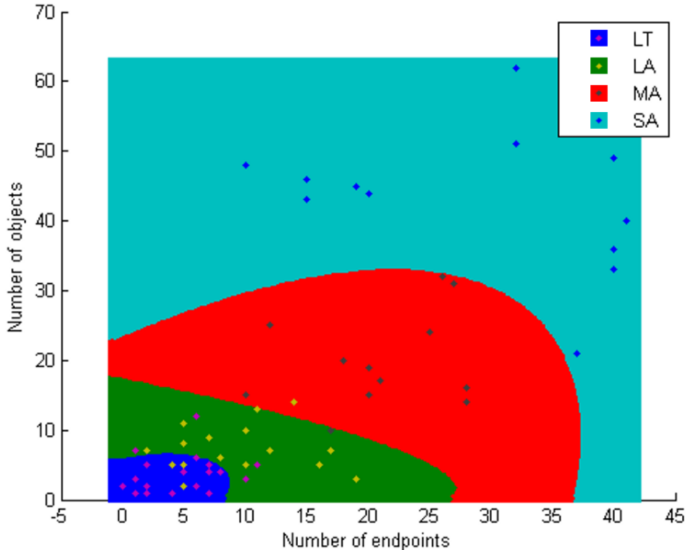


*Figure 2.21. 2D feature space with decision boundaries. Data points are training data.*

It is helpful to note at this point that the 8% error could be even further eliminated by increasing the size of the training set and by incorporating a Parzen (or kernel density estimation) classifier. This gives decision boundaries the flexibility to contour around irregular boundaries in the data which the quadratic classifier does not provide. To do this, the conditional PDF $P(\boldsymbol{z}|\tilde{w}_i)$ is given by a mixture of Gaussians rather than a single Gaussian. In some cases, the Gaussians are estimated by histogramming the training data and assigning a smoothing parameter which simply corresponds to the covariance of the contributing Gaussians [45]. In experimenting with this form of Bayesian classifier, each training datum was given a unique PDF with mean and covariance determined by the accuracy of detection. These PDFs are shown for each class in Fig. 2.22. Creating the decision boundaries can be thought of as finding the contours along which the values of the PDF "surface" for two or more neighboring classes are equal. The decision boundaries are illustrated in Fig. 2.23. The prior probability $P(\tilde{w}_i)$ given in Eqs. 1.27 and 1.29 can be thought of as simply a weighting factor for each of the class PDFs. If, for instance, the probability of randomly encountering an LT road crack is 7 times greater than all the other classes, $(P(\tilde{w}_{LT}) = 0.7, P(\tilde{w}_{LA}) = 0.1, P(\tilde{w}_{MA}) = 0.1, P(\tilde{w}_{SA}) = 0.1)$, the magnitude of the LT PDF would be scaled by a factor of seven, resulting in a growth of the LT region in the feature space.

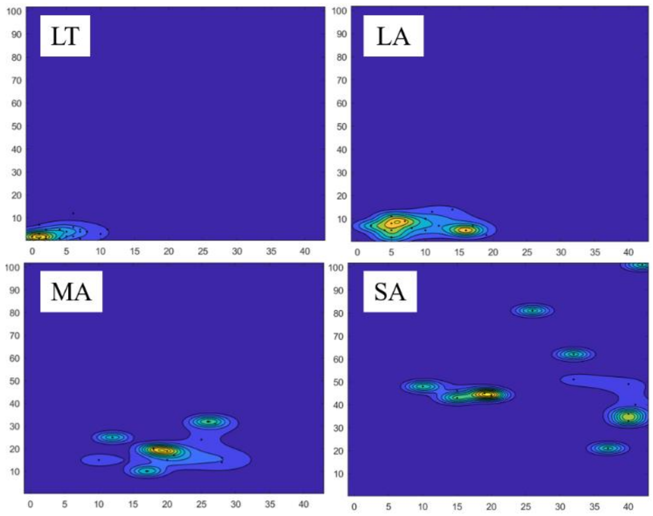


*Figure 2.22: Conditional PDFs for each class.*

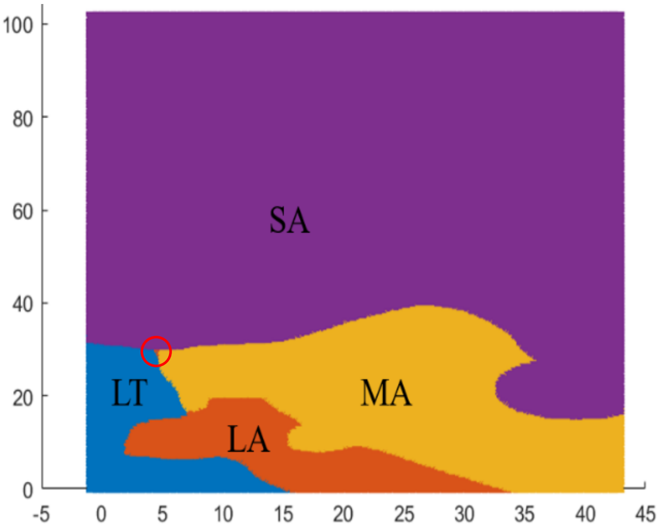


*Figure 2.23: Parzen class decision boundaries.*

The Parzen classifier is only really appropriate when the training data set fills out the feature space more completely. Otherwise, the borders become somewhat arbitrary and presumptuous. This is apparent in Fig. 2.23 with the presence of a small isolated LA region (circled in red). A crack landing in precisely that region should be assigned to LT, MA, or SA classes, but instead would be erroneously classified as LA. Another issue with an undertrained Parzen classifier is that there is a direct adjacency between the least severe class (LT) and the most severe class (SA). It makes no sense that a crack is more likely to be LT or SA than either LA or MA. These issues are avoided with the quadratic classifier at this stage.

### 2.2.d Bayesian correction

Bayesian correction has been used once before in the framework presented here, though that implementation was spatial in nature. Image pixels' crack likelihoods were corrected by pre-processing the binary crack image over a series of summation limits, Gaussian blurring, and computing the resulting normalized product (see section 2.1.b). Now, the problem becomes more theoretical, as correction takes place in the feature space. As was mentioned before, each test image can be represented in the feature space by a Gaussian PDF with mean and covariance given by measurement vectors taken over a series of blurring radii. On their own, these PDFs have two notable characteristics: they are widely spaced from each other, and they have wide distributions. This is because arbitrary noise in one measurement may cause it to differ substantially from its neighbors, and significant variation in the measurement vectors of an individual crack image causes uncertainty in the estimation of its true location in the feature space. The goal of corrective classification is to mitigate both these issues by enforcing coherence and enhancing certainty.

The data acquisition method employed in this research lends to a Bayesian correction approach for classifying test images. Road images are sequentially obtained by a moving vehicle with variable spacing and overlap and over such short exposure times that motion blur is negligible.  Since each crack image shares some overlapping features with the previous and future images for image stitching, and because a segment of road pavement will not naturally demonstrate an abrupt change in cracking structure, the probability distribution functions from prior and subsequent images can correct new observations for a current image. The traditional form of Bayesian correction comes from the recursive Bayesian estimation framework and is given as follows:

$$p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{1:k}^o) = \frac{l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{1:k-1}^o)}{\int l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{1:k-1}^o)\,d\boldsymbol{x}_k^o}. \qquad (2.6)$$

Here, $l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)$ is the likelihood of the position of an object in the feature space given only the most recent measurement $\tilde{\boldsymbol{z}}_k^o$, and $p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{1:k-1}^o)$ is the probability density of the position of the object given all measurements up to the most recent one. This is the general form of Eq. 2.5 which was presented for corrective detection. Eq. 2.6 was derived for continuous operation within the RBE framework; in this formulation, all previous observations have an indirect effect on the correction of a new observation, and future observations are unknown. In order to adapt this theory to the context of classification, (2.6) must be augmented as follows:

$$p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k}^o) = \frac{l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k-1}^o)}{\int l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k-1}^o)\,d\boldsymbol{x}_k^o}. \qquad (2.7)$$

In this modified form of the correction formula, only $M$ past measurements are used to update belief about a current measurement. In this sense, Eq. 2.7 describes a limited look-behind form of correction. $M$ is a variable parameter which reflects the amount of overlap or spacing between images. Only $M$ previous measurements are taken into account because pavement condition can change drastically over several meters of road. In addition, it would be unrealistic to restrict the location of a crack image in the feature space with information about road condition from an image taken a mile previous. However, if $M$ is too small, the effect of correction is minimal and the full extent of available information is not taken advantage of.

Corrective classification is guaranteed to increase the certainty of class estimation, but some caution is imperative. To a certain extent, the formulation in Eq. 2.7 is heuristic in that $M$ is tuned largely by intuition. It is possible to overestimate this parameter, in which case the certainty of the resulting PDFs would be much too liberal. When properly used, Bayesian correction in the feature space very nicely tunes the crack class estimates of a sequential data set. Additionally, because data is processed off-site and every image from an entire sequence is available, limited look-behind correction can be replaced with limited look-ahead correction or a combination of both. Although the results given in this work used limited look-ahead correction, a more accurate approach would be one in which an equal number of images before and after a given image are used in its correction. The only difference is a shift in image class assignment which, from a high level, is a negligible effect. At a future stage of work, a windowing approach will be taken, in which the contribution of each crack image PDF used to correct a center crack image PDF is weighted according to its distance from the center. The correction equation for this approach would take the following form:

$$p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k}^o) = \frac{C_k l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k-1}^o)}{\int l(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_k^o)p(\boldsymbol{x}_k^o|\tilde{\boldsymbol{z}}_{k-M:k-1}^o)\, d\boldsymbol{x}_k^o}. \tag{2.8}$$

where $C_k$ is given by

$$C_k = N(0,1,v)\delta(v-k). \tag{2.9}$$

This is a discretized Gaussian distribution sampled at the first, second, third, etc. standard deviations. In essence, the weight of the $k^{th}$ crack image is given the value of a standard normal distribution at the $k^{th}$ standard deviation. Although this is admittedly heuristic, it bears a strong resemblance to human reasoning.

# CHAPTER 3. Results

## 3.1 Pre-processing analysis

The pre-processing and filtering methods described above were implemented on a data set of 110 crack images of varying severities and pavement types. For all simple single-crack images assessed, the performance rate is nearly perfect (all cracks are detected, no false positives are detected). However, the scope of this work is much broader than simple isolated cracks. Because performance is difficult to objectively quantify for complex cracking patterns, some sample images will be provided and discussed. This sub-section presents the results of crack identification for the uniform and segmented summation limit techniques for three representative kinds of crack images.

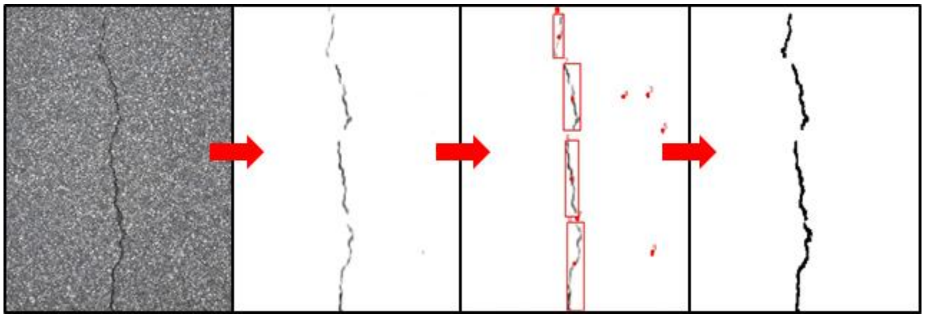


*Figure 3.1: Uniform pre-processing, linear crack. $\alpha$ = 0.247, $\delta_x$ = 0.5, $\delta_v$ = 0.2, N = 7.*

For a summation limit of 7 and vertically biased overlaps, the uniform pre-processing method clearly detects linear crack features. The few places in which the crack is broken in the final image are a result of pavement overlap. These gaps are corrected in future steps as described in section 2.1.d. The computation time for this operation is an average of 0.8 seconds.

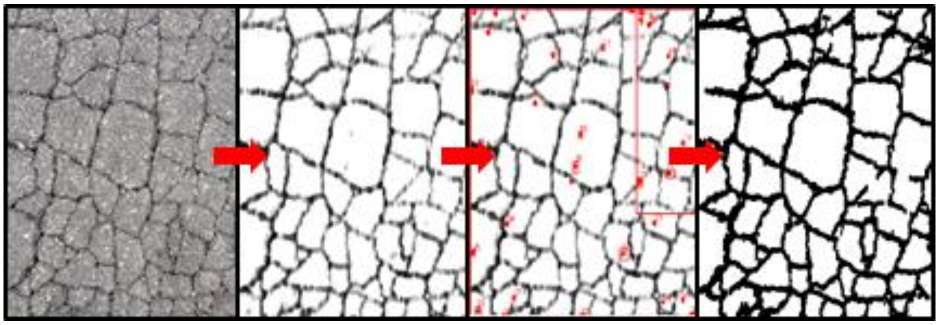


*Figure 3.2 Uniform pre-processing, alligator crack. $\alpha$ = 0.247, $\delta_x$ = 0.5, $\delta_y$ = 0.2, N = 2.6.*

For a summation limit of 2.6 and vertically biased overlaps, the uniform pre-processing method also works well for complex cracking patterns. The basic forms of the cracks are retained with relatively little deviation. The computation time for this operation is an average of 0.7 seconds, because $N$ is smaller than before.

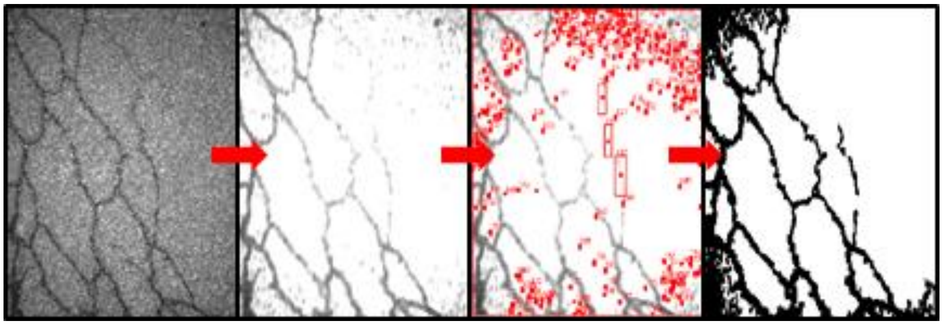


*Figure 3.3: Uniform pre-processing, uneven lighting.* $\alpha = 0.247$, $\delta_x = 0.5$, $\delta_y = 0.2$, $N = 3.5$.

For an unevenly illuminated road image such as that shown in Fig. 3.3, regions of darker pixels do not saturate with the same summation limit, resulting in significant issues at the corners. The summation limit could be adjusted to resolve crack features in the dark regions, but this would cause sub-domains in brighter areas to saturate and be lost. Solving this problem is the intent of segmented pre-processing. The computation time for this operation is an average of 0.71 seconds.

Below, the same three crack images are pre-processed using the segmented pre-processing technique. The aim of this modification is to avoid the problems associated with spatial parameter variation across the sub-domains of a single image. For review of the parameters involved in segmented pre-processing, see section 2.1.b.

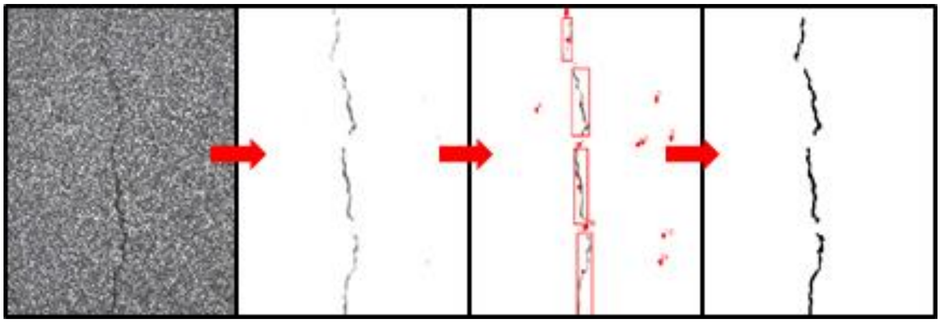


*Figure 3.4: Segmented pre-processing, linear crack.* $P = 18$, $\alpha = 0.247$, $\delta_x = 0.5$, $\delta_y = 0.2$, $L = 6.5$, $H = 7$.

As Fig. 3.4 shows, the segmented pre-processing approach yields results with very few differences from the uniform pre-processing approach for an image with constant background illumination. The computation time for this operation is an average of 0.8 seconds, negligibly different from the computation time with uniform pre-processing.

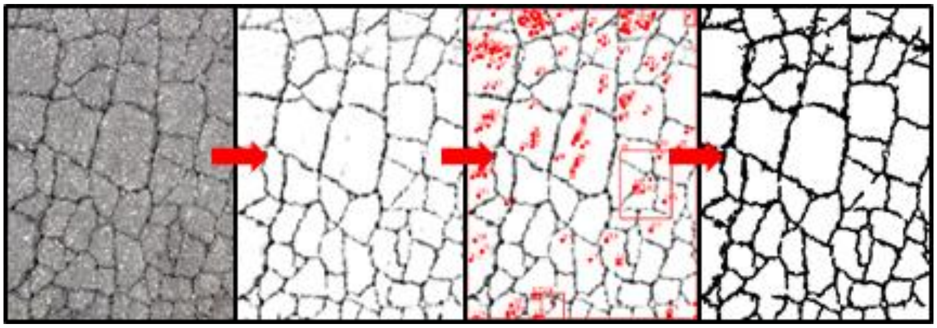


*Figure 3.5: Segmented pre-processing, alligator crack. P = 18, α = 0.247, $\delta_x$ = 0.4, $\delta_y$ = 0.4, L = 2, H = 3.5.*

Once again, these results match closely to those from the uniform pre-processing method. In this case, however, the percent overlaps were set equal at 60% in order to gain some vertical resolution. This also has the effect of reducing computation time: this operation took an average 0.65 seconds.

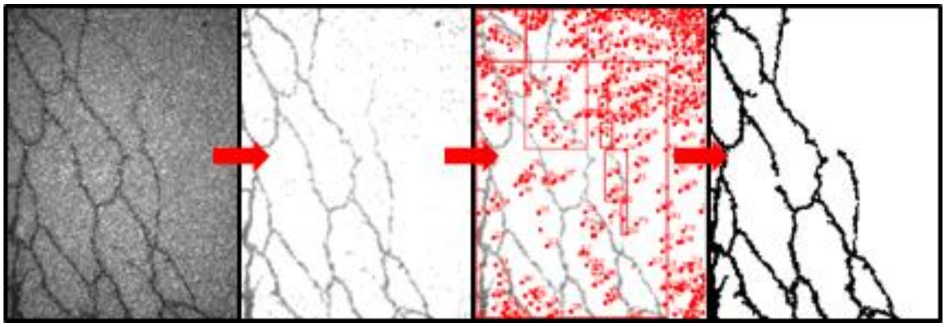


*Figure 3.6: Segmented pre-processing, uneven lighting. P = 18, α = 0.247, $\delta_x$ = 0.4, $\delta_y$ = 0.4, L = 2.5, H = 4.8.*

This crack image's non-uniform illumination caused problems when a single summation limit was applied across the whole image. As Fig. 3.6 clearly shows, the segmented pre-processing technique eliminates the noise at the corners of the processed image, yielding a much cleaner representation of the cracks. The degree of subdivision can be adjusted according to the coarseness of the shadow features, but in this example P = 18 resulted in sufficiently distinguishable crack features. The computation time for this operation is an average of 0.65 seconds, less than that for uniform pre-processing only because the percent overlaps were modified.

In order to show the reliability of this pre-processing approach for road crack images of varying average pixel intensity and standard deviation, a theoretical numerical evaluation was completed. For a downsampling ratio of 0.247 and percent overlaps of 60%, a pixel block (given by $\boldsymbol{p_{i,j}}$ previously) is filled with normally distributed random values with a mean of 128 and standard deviation of 80. These values are typical of the average road image, and it is assumed that the background noise in any given road image (i.e. uncracked pavement) is approximately Gaussian over a sufficiently large sampling. A single-pixel-wide "crack" is then inserted into this kernel of pixels to represent the smallest possible representation of a crack in a pixelated display. The crack's pixel values are also randomly distributed, with a mean given by $\mu_C = \beta_m \mu_B$ and standard deviation given by $\sigma_C = \beta_s \sigma_B$, where $\mu_C$ is the crack mean value, $\mu_B$ is the background mean value, $\beta_m$ is a mean ratio, $\sigma_C$ is the crack standard deviation, $\sigma_B$ is the background standard deviation, and $\beta_s$ is a standard deviation ratio. For combinations of $0.5 \leq \beta_m \leq 1$ and $0.05 \leq \beta_s \leq 1$, the pre-processing summation limit was varied from 1 to 10 in increments of 0.1 until a value was found that saturates the crack-less kernel to 255. If the kernel with the crack was also found to saturate at this value, the summation limit was said to fail and the values of $\beta_m$ and $\beta_s$ led to undistinguishable cracking for the worst-case scenario of a single-pixel-wide crack. Given the stochastic nature of this simulation and the similarly stochastic underlying nature of road pavement, failure may only occur for each $\beta_m$ and $\beta_s$ combination for a certain percentage of the trials. Consequently, a percent reliability (the conjugate of percent failure) matrix can be constructed for combinations of $\beta_m$ and $\beta_s$. Fig. 3.7 shows the interpolatively smoothed surface plot of these findings. Note the approximately linear drop-off boundary.

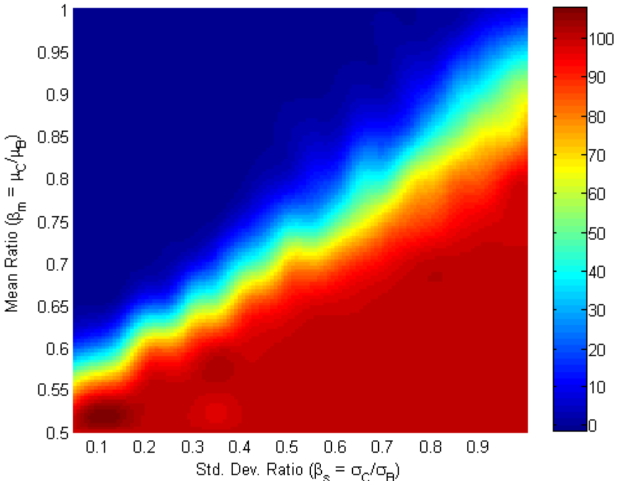


*Figure 3.7: Percent reliability as function of mean ratio and standard deviation ratio.*

Example kernels are shown in Fig. 3.8. The kernels are sampled from near each of the four corners of the above plot. The images shown in Fig. 3.8 are spatially organized according to their sampling region in Fig. 3.7. As both figures corroborate, crack features are most reliably detected when both the standard deviation and mean of crack pixels differ significantly from those of background pixels. Although this may be intuitively obvious, it is initially surprising that crack pixels are also essentially just as reliably detected when their means are lower than the background and their standard deviations are nearly the

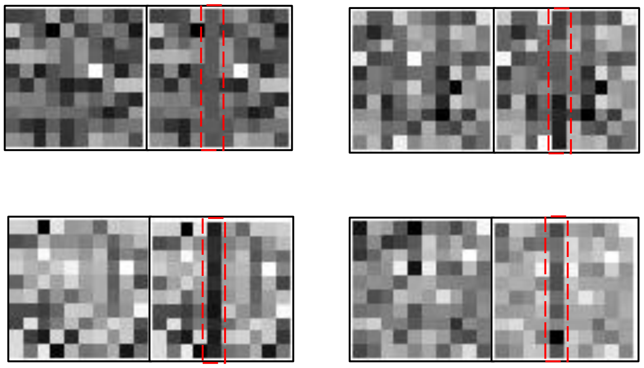


*Figure 3.8: Example kernels without and with crack pixels. Counter-clockwise from upper left: $\beta_m$ = 0.95, $\beta_s$ = 0.15; $\beta_m$ = 0.9, $\beta_s$ = 0.9; $\beta_m$ = 0.55, $\beta_s$ = 0.15; $\beta_m$ = 0.55, $\beta_s$ = 0.9.*

same as the background. This is simply because high variance in crack pixels results in a greater likelihood that darker pixels will be present. Since Eq. 2.1 preferentially weights the darkest $N$ pixels, accurate detection is more probable if even only a few exceptionally dark pixels are present in a given image.

Admittedly, there is a point at which increasing the mean ratio no longer realistically represents a crack. For instance, a human evaluator would not assign the streak in the center of the kernel in Fig. 3.8a as being a crack feature. Therefore, much of the upper region in Fig. 3.7 can be disregarded since it does not apply to ground truth. This beneficially impacts the conclusions of this numerical study, as it increases the region in the $\beta_m$-$\beta_s$ space for which crack images will likely give correct detection results. It is also worth noting, though it is perhaps obvious, that the region corresponding to $\beta_m < 0.5$ yields 100% reliability; this region was excluded from the figure for conciseness.

## 3.2 Detection Results

The results of correction in the pre-processing stage of road crack detection are best given by visual example. Fig. 3.9 shows the effects of correction on a crack image. The first three images display a crack pre-processed at summation limits of 7.75, 8, and 8.25, respectively. Correction was performed with these three images; note how the first image contains significant border noise and one non-crack region, but maintains crack length; conversely, the third image shows improvement in noise rejection but at the cost of some lost crack length. The final image is the corrected result which eliminates isolated non-crack features located in the minority of images and smooths crack edges to some extent.

This process increases certainty about crack locations (see Fig. 2.8c in comparison with Fig. 2.8a and 2.8b) and provides forgiveness for uncertainty in the ideal summation limit.

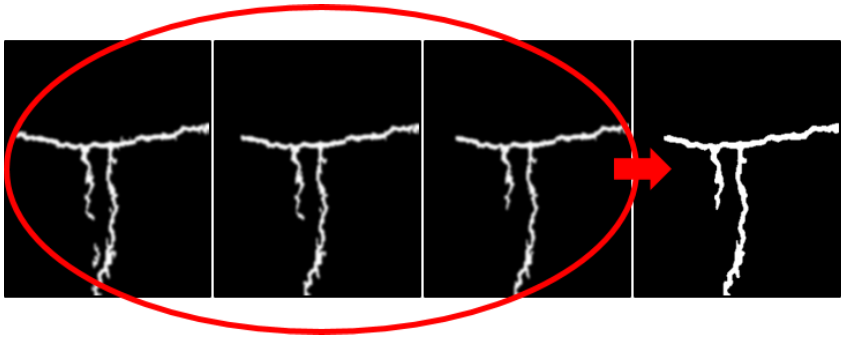


*Figure 3.9: Correction in pre-processing: three crack PDFs corresponding to three different summation limits followed by final corrected image.*

For best results, several closely spaced summation limits ought to be incorporated into the correction process. This would allow for greater preservation of true crack detail and rejection of false positives. Unfortunately, since image pre-processing is the most computationally costly operation in crack detection, using more than a few images for correction takes a disadvantageously long amount of time. Consequently, the error in an automatically determined summation limit can be best handled when uncertainty in this parameter is restricted to only ±0.25 or less. This places the burden of accuracy on automated parameter assignment, which is discussed in the conclusions section. In any case, the method of correction in crack detection described here successfully enhances crack belief according to its intended purpose.

For a representative testing data set consisting of 120 consecutive road images, detection accuracy is given in Table 3.1. This table shows that 92% of the images were correctly identified as either having a crack or not. The data used for this analysis came from a road segment containing some regions of unaffected pavement and some regions of longitudinal/transverse cracking. Most local roads assessed with this system fit into this category, and for binary detection assessment (crack is either present or not), it can be considered a worst-case scenario. For more complex cracking networks, it is highly unlikely that at least some crack regions would not be detected.

*Table 3.1: Truth table for binary detection in a representative road image set.*

| | | Detection | |
|---|---|---|---|
| | | Crack | No Crack |
| Ground Truth | Crack | 85.8% | 4.2% |
| | No Crack | 4.2% | 5.8% |

## 3.3 Training and Classification Results

The correction formula given in Eq. 2.7 was used to cast a sequence of twelve road crack images into the feature space. Although the notation in Eq. 2.7 implies a “look-behind” approach, the cracks of Fig. 3.10 were corrected using the limited look-ahead approach described in section 2.2.d. As the figure shows, cracking begins in the severe alligator category, with many pavement objects completely enclosed by cracks, a pattern reminiscent of the appearance of alligator scales. After the first seven images, road condition improves somewhat, transitioning to moderate alligator cracking. Finally, the last image shows light alligator cracking.

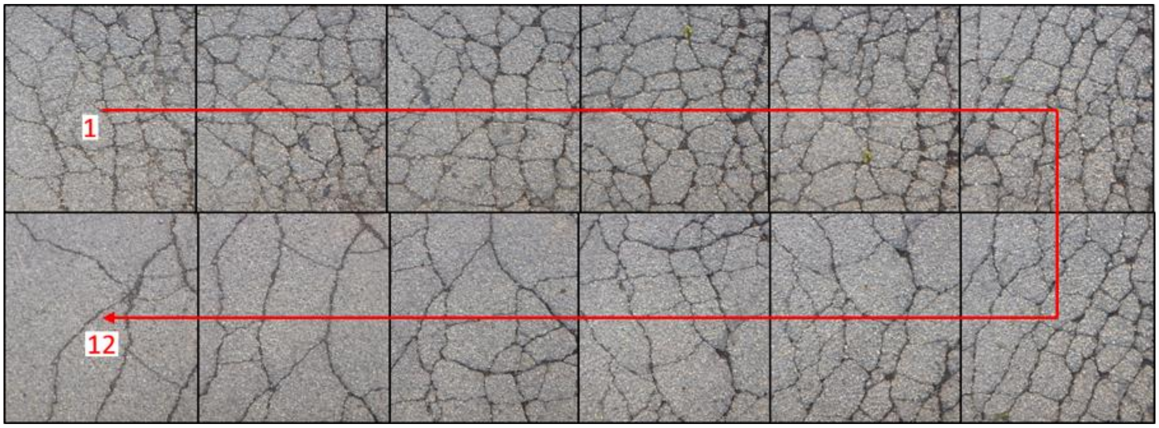


*Figure 3.10: Twelve sequential road crack images with approximately 33% overlap (top left wrapping to bottom left).*

Fig. 3.11 depicts these 12 images in the feature space, first without and then with limited look-ahead correction implemented. The look-ahead parameter $M$ was given a value of three for this correction sequence. As is apparent from the figure, correction has two effects: to adjust the position of a crack image in the feature space according to local crack type as determined by other neighboring images, and to increase the certainty of a crack image’s position in the feature space. Notice, however, that the final PDF remains unaffected, with relatively large uncertainty. This is because, as intuition would guide, the last crack image is not reinforced by subsequent observations, thus reducing its certainty in the feature space.

According to the relatively subjective human assessment of the road crack images given in Fig. 3.10, the results of corrective classification show that class assignment of each image has been successfully achieved. The recall rate (in this case, the ratio of correctly classified cracks to all cracks) for the uncorrected data sequence is 75%, whereas

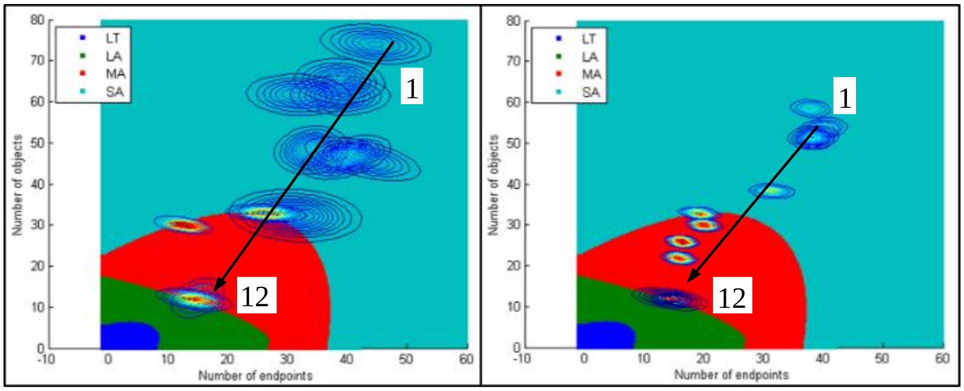


*Figure 3.11: Sequential crack images cast into the feature space a) without correction and b) with limited look-ahead correction.*

corrective classification yields a 100% recall rate for this twelve-image data set. The mean vector of every test datum falls within the class to which it most likely belongs, and the “softness” of decision boundaries in stochastic feature classification offers flexibility for crack images cast onto borders between classes. In order to better quantify belief for such cases, each Gaussian PDF can be integrated over each of the four class regions to develop a probability that the given crack image belongs to a particular class. These results are given in Table 3.2 below.

*Table 3.2: Probability of crack images for each class.*

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LT | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% |
| LA | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 59.6% | 59.8% |
| MA | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 0.0% | 64.8% | 100.0% | 100.0% | 40.4% | 40.2% |
| SA | 100.0% | 100.0% | 100.0% | 100.0% | 100.0% | 100.0% | 100.0% | 35.2% | 0.0% | 0.0% | 0.0% | 0.0% |

Table 3.2 gives the approximate probabilities of each class for every crack image in the test sequence. For the crack images whose position in the feature space was well within the decision boundaries, the probability of such cracks belonging to a different class are so small (often on the order of $10^{-150}$ or smaller) that they’ve been approximated as zero. For transitional crack images such as images 8, 11, and 12, although the class may be assigned as that with the greatest probability, the level of uncertainty reflected in the nontrivial likelihood that the crack image could belong to another class is useful in assessing road condition and making a decision about how a particular road segment ought to be repaired.

# Chapter 4. Conclusions and Future Work

This work has presented a number of techniques for rigorous detection and classification of stochastic features; namely, road pavement cracks. A new pre-processing technique for road crack detection by selective tunable down-resolving was presented, followed by crack feature isolation, correction, and cleaning. A quadratic Bayesian classifier was trained, validated, and used to classify real sequential road crack images with promising accuracy. The fundamental algorithm for pre-processing was applied for both constant and uneven illumination conditions. After applying the pre-processing approach to three different but common crack image types, the benefits of both methods were seen. Furthermore, corrective detection consistently and reliably extracted useful features to aid in high-level classification. These features' uncertainties were computed in order to correctively cast data into the feature space with PDFs that can be integrated over class decision boundaries created by the quadratic Bayesian classifier.

Furthermore, the sensitivity of the detection approach was numerically characterized. A single-pixel-wide crack can be reliably detected from background pavement with the standard 0.247 down-resolution ratio and 60% x- and y-overlaps so long as its mean pixel value is half that of the background pixel mean, or so long as its pixel values' standard deviation is relatively high. These characteristics can be used to probabilistically assign confidence ratings to pavement types or illumination conditions, both macroscopically (across an entire image) and microscopically (for regions within an image). This could be readily implemented into the already-established Bayesian classification framework, though the current state of work does not include this capability.

The results that have thus far been obtained in the course of this work have proven the effectiveness of this crack detection approach across a broad spectrum of crack and pavement types. All single-crack images are properly detected as such approximately 92% of the time, and no false positives occur under the conditions described previously. Additionally, these methods have proven ubiquitous enough to successfully detect most, if not all, segments of severe cracking even under spatially varying average background intensity. For several more examples of detected cracks, see the included Appendix.

The novel multi-Bayesian approach for stochastic feature detection and classification has been used to draw useful and meaningful conclusions about environments containing excessive sensor noise. Although the efficacy of these methods can be extrapolated to any number of conceivable contexts in which the reliable detection and classification of stochastic features is paramount, the proposed approach has been specifically formulated for road crack detection and characterization. Bayesian correction in detection improves certainty about crack pixels in a test image, and this method could be applicable in computer vision based detection of many useful environmental features, particularly robotics applications. Bayesian correction in classification makes use of a rich set of environmental observations to aid in high-level decision making. Corrective classification can be adapted to any application of classical categorization methods, whether in computer vision, robot audition, or even data mining with observational uncertainty.

As this paper has expressed, autonomy hinges on the critical need for an accurate determination of system parameters; in particular, an optimal pre-processing summation limit is most important for adequate detection results. The difficulty with this problem is that the best parameter to use in a given image is highly unpredictable (stochastic), and even a human cannot effectively choose a value without some trial and error. Currently, the system is semi-automated, in that a user manually determines summation limits for every N$^{th}$ image, and the frames in between are interpolated with a Gaussian windowing function. Given the stochastic nature of parameter assignment, a logical approach to take for summation limit determination in the future is to supervise this parameter setting for one or more crack images, establishing one-dimensional *a priori* PDFs, and correcting the summation limit according to recursive Bayesian estimation for subsequent images based on adjustment criteria specific to each image. The adjustment criteria would be developed from a study of the correlation between image statistics and appropriate summation limit designations. Under this RBE-based parameter estimation scheme, user-provided values would be considered the “observations”, and the adjustment criteria would take the form of “predictions”. Investigation of the correlation between image statistics and ideal summation limits has been initiated, but future work in this area remains.

One probable criticism of the proposed method is its lack of dimensional complexity; only two feature types are used for classification. This is true, though justified for three reasons: first, visualization is far easier in 2D space, and this thesis benefits from understandable graphics. Second, even just 2D classification has proven highly capable of the basic four-class categorization proposed here. Third, increasing the dimensionality of the feature space requires much more training data to avoid underdetermining large regions of the space. This can lead to bad classification of atypical test data. Nevertheless, the ultimate goal of this research is to incorporate more crack descriptors in classification for more thorough and accurate results. Some of these features may include crack length, centroid, orientation, shape, etc. However, a problem arises when many dimensions are used in corrective classification. Because the correction formula requires a spatial multiplication of PDFs, the N-dimensional coordinates of every discrete element in a feature space must be given a value corresponding to the PDF at that location. For the Gaussian case, this means each location is substituted into Eq. 1.28. The computation time associated with this operation increases exponentially as the number of dimensions increases. This means that, using an explicit discretized spatial representation of uncertainty, a large data set could take hours to be processed. An advantage of the Gaussian PDF assumption, however, is that only two parameters need to be "tracked" through Bayesian correction: the mean and standard deviation. In addition, it can be proven that the product of two Gaussians has a normal behavior. The mathematical consequence, then, is that a corrected PDF's mean vector and covariance matrix in N-dimensional space can simply be fully defined in terms of the constituent PDFs' means and covariances rather than by a spatial object fully populated in N dimensions by discrete PDF values. Additionally, the mean and covariance are able to precisely define the corrected PDF at all continuous test points. The formulation is given below.

Suppose $\boldsymbol{G_1} \sim N(\boldsymbol{\mu_1}, \boldsymbol{\Sigma_1})$ and $\boldsymbol{G_2} \sim N(\boldsymbol{\mu_2}, \boldsymbol{\Sigma_2})$. If $\boldsymbol{G_3} = \frac{\boldsymbol{G_1} .* \boldsymbol{G_2}}{\int \boldsymbol{G_1} .* \boldsymbol{G_2}}$, it is known that $\boldsymbol{G_3} \sim N(\boldsymbol{\mu_3}, \boldsymbol{\Sigma_3})$. The question, then, is what are $\boldsymbol{\mu_3}$ and $\boldsymbol{\Sigma_3}$? It can be shown from manipulation of the multiplied Gaussian functions (Eq. 1.28) that

$$\boldsymbol{\mu_3} = (\boldsymbol{\Sigma_1} + \boldsymbol{\Sigma_2})^{-1}(\boldsymbol{\Sigma_2}\boldsymbol{\mu_1} + \boldsymbol{\Sigma_1}\boldsymbol{\mu_2}) \tag{4.1}$$

and

$$\boldsymbol{\Sigma_3} = (\boldsymbol{\Sigma_1} + \boldsymbol{\Sigma_2})^{-1}(\boldsymbol{\Sigma_1}\boldsymbol{\Sigma_2}). \tag{4.2}$$

In this way, uncertainty and estimation are tracked efficiently, in a "Kalman-like" fashion. At this point, the computational challenge is to non-numerically integrate the corrected PDF over decision boundaries created by the QBC. This task has not yet been formulated.

Another limitation to current methods is the manner in which images are acquired. The camera setup can be described as minimalistic, in that no external illumination or structural constraints are enforced, with the exception of the requirement that the cameras be downward-facing. This makes for a more ubiquitous system, but there is an accuracy cost. The quality of data is highly dependent on cloud cover, sun angle, nearby shadow-casting objects, and weather. To be able to more reliably obtain and assess road pavement data, a future step would be to incorporate controlled illumination. When lighting is ideal, pavement texture will not have such an extreme effect on crack detection. An example of surface texture interference is shown in Fig. 4.1.

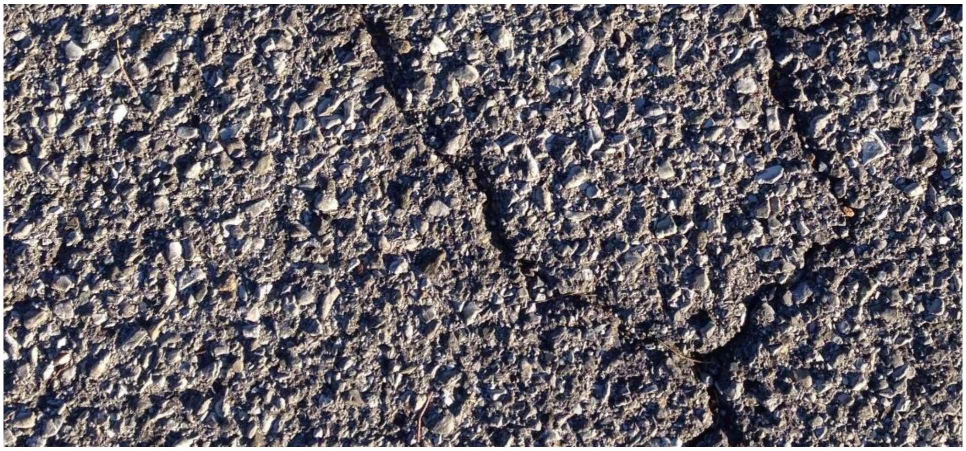


*Figure 4.1: Example of micro-shadows cast by low sun angle and significant surface texture caused by pavement erosion.*

As the figure shows, the dark regions corresponding to "micro-shadows" greatly hinder accurate crack detection. In some areas, the crack blends in with the pavement, while in other areas it stands out better. This lack of continuity and confusion with pavement texture results in poor detection and therefore poor feature extraction and classification.

As was mentioned at the very outset of this work, a common standard for pavement assessment is the pavement condition index (PCI) and its variants. It may seem perplexing that these standards have not resurfaced throughout the paper, but this is not without reason. At this stage of work, industry experts have not yet been brought on for professional assignment of ground truth with regards to classification and QBC training. In the near future, cooperation with field experts will be critical in casting this work into a more realistic and helpful light. Rather than using the four naïve crack classes described throughout, categories and sub-categories of cracks will be established by professionals and their discriminating features detected for more intricate classification. This opens the door for the possibility of branching classification, and adaptive feature extraction. For example, a slippage-type cracking network has a high density of pavement objects, but the shape and size of the objects is much different from ordinary alligator cracks. This information is crucial to understanding how the road is deteriorating, and cannot be ignored. However, including pavement object shape and size in the basis feature space does not help to distinguish between longitudinal cracks and more severe cracking networks; it would in fact waste entire regions of the feature space and compromise efficiency. A hierarchy of classification stages, each with its own multi-dimensional feature space, would yield much more useful results without much waste.

Finally, it is worth mentioning that a possible avenue for future work is implementing a variant of neural networks for crack classification. Neural networks have the benefit of widespread use and increasing impact in the field today. They are capable of accurately classifying objects without much mathematical formulation or restraints and assumptions. However, adapting the Bayesian methods presented here in a neural network framework would require a means of propagating uncertainty through a network. To the author's knowledge, this has not been investigated in much detail by other researchers in the field as of yet, so formulation of such a technique is necessary. Assuming this is achievable, the classes and uncertainties assigned to a series of crack images by a neural network would be used correctively to enhance the certainty with which a single particular crack image is assigned a class, in a manner similar to what was described in section 2.2.d. The minutia of this technique adaptation have not yet been firmly established, but it is in theory a promising prospect. SDG

# References

*Note: Some content of this thesis was copied and adapted from two previous publications by the author, listed immediately below.*

J. Josiah Steckenrider, Tomonari Furukawa, "Selective Pre-Processing Method for Road Crack Detection with High-Speed Data Acquisition," International Journal of Automotive Engineering, 2017.

J. Josiah Steckenrider, Tomonari Furukawa, "Detection and Classification of Stochastic Features using a Multi-Bayesian Approach," IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI), 2017.

[1] Mark R. Mine, Frederick P. Brooks, Carlo H. Sequin, "Moving Objects in Space: Exploiting Proprioception In Virtual-Environment Interaction," *Proceedings of the 24th annual conference on Computer graphics and interactive techniques,* pp. 19-26, 1997.

[2] Merle Lawrence, "Fluid Balance in the Inner Ear," *Annals of Otology, Rhinology & Laryngology*, vol. 74, pp. 486-499, 1965.

[3] S. Carrubba, C. Frilot, A. L. Chesson, A. A. Marino, "EVIDENCE OF A NONLINEAR HUMAN MAGNETIC SENSE," *Neuroscience*, vol. 144, no. 1, pp. 356-367, 2007.

[4] E. B. Goldstein, Sensation and Perception, Belmont, CA: Wadsworth, 2010.

[5] Michael J. McGrath, Cliodhna Ní Scanaill, Sensing and Sensor Fundamentals, New York City, New York: Apress, 2014.

[6] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering,* vol. 82, no. D, pp. 35-45, 1960.

[7] Eric A. Wan, Rudolph van der Merwe, "The Unscented Kalman Filter for Nonlinear Estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium,* Lake Louise, Alberta, Canada, 2000.

[8] Garry A. Einicke, Langford B. White, "Robust Extended Kalman Filtering," *IEEE TRANSACTIONS ON SIGNAL PROCESSING,* vol. 47, no. 9, pp. 2596-2599, 1999.

[9] M. Sanjeev Arulampalam, Simon Maskell, Neil Gordon, Tim Clapp, "A Tutorial on Particle Filters for Online Nonlinear/Non-Gaussian Bayesian Tracking," *IEEE TRANSACTIONS ON SIGNAL PROCESSING,* vol. 50, no. 2, pp. 174-188, 2002.

[10] S. W. Erickson, "Street Pavement Maintenance: Road Condition Is Deteriorating Due To Insufficient Funding," Office of the City Auditor, San Jose, CA, Feb. 2015.

[11] B. Vlacich, "State of the Pavement 2016," Virginia Department of Transportation, Richmond, VA, 2016.

[12] K. H. McGhee, Affan Habib, Tanveer Chowdhury, "DEVELOPMENT OF PAVEMENT CONDITION INDICES FOR THE VIRGINIA DEPARTMENT OF TRANSPORTATION," Virginia Department of Transportation Maintenance Division, Richmond, VA, 2002.

[13] P. Schreurs, *Fracture Mechanics,* Eindhoven, Netherlands: Eindhoven University of Technology, 2011.

[14] A. A. Griffith, "The Phenomena of Rupture and Flow in Solids," *Philosophical Transactions of the Royal Society of London,* vol. 221, no. A, pp. 163-198, 1921.

[15] F. Erdogan, "Fracture mechanics," *International Journal of Solids and Structures,* vol. 37, pp. 171-183, 2000.

[16] Mingchao Liu, Yixiang Gan, Dorian A.H. Hanaor, Bin Liu, Changqing Chen, "An improved semi-analytical solution for stress at round-tip notches," *Engineering Fracture Mechanics,* vol. 149, pp. 134-143, 2015.

[17] P. Paris, F. Erdogan, "A Critical Analysis of Crack Propagation Laws," *Transactions of the ASME,* pp. 528-533, 1963.

[18] N. Pugno, M. Ciavarella, P. Cornettia, A. Carpinteri, "A generalized Paris’ law for fatigue crack growth," *Journal of the Mechanics and Physics of Solids,* vol. 54, pp. 1333-1349, 2006.

[19] A. A. Molenaar, "Structural Performance and Design of Flexible Road Constructions and Asphalt Concrete Overlays," Delft University of Technology, Delft, Netherlands, 1983.

[20] S.M.J.G. Erkens, J. Moraal , "Cracking in asphalt concrete," *HERON,* vol. 41, no. 1, pp. 53-70, 1996.

[21] Y. Li, "Asphalt Pavement Fatigue Cracking Modeling," Louisiana State University, Baton Rouge, LA, 1999.

[22] Toshihiko Fukuhara, Keiji Terada, Makoto Nagao, Atsushi Kasahara, Shigeki Ichihashi, "AUTOMATIC PAVEMENT-DISTRESS-SURVEY SYSTEM," *Journal of Transportation Engineering,* vol. 116, no. 3, pp. 280-286, 1990.

[23] Naoki Tanaka, Kenji Uematsu, "A Crack Detection Method in Road Surface Images Using Morphology," *IAPR Workshop on Machine Vision Applications,* pp. 154-157, 1998.

[24] T. Tomikawa, "A Study of Road Crack Detection by the Meta-Genetic Algorithm," *Africon,* pp. 543-548, 1999.

[25] Miguel Gavilán, David Balcones, Oscar Marcos, David F. Llorca, Miguel A. Sotelo, Ignacio Parra, Manuel Ocaña, Pedro Aliseda, Pedro Yarza, Alejandro Amírola, "Adaptive Road Crack Detection System by Pavement Classification," *Sensors,* vol. 11, pp. 9628-9657, 2011.

[26] Henrique Oliveira, Paulo Lobato Correia, "Automatic Road Crack Detection and Characterization," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* vol. 14, no. 1, pp. 155-168, Mar. 2013.

[27] Y. Sun, E. Salari, E. Chou, "Automated Pavement Distress Detection Using Advanced Image Processing Techniques," in *IEEE International Conference on Electro/Information Technology,* Windsor, Canada, 2009.

[28] Rabih Amhaz, Sylvie Chambon, Jérôme Idier, Vincent Baltazart, "Automatic Crack Detection on Two-Dimensional Pavement Images: An Algorithm Based on Minimal Path Selection," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS,* vol. 17, no. 10, pp. 2718-2729, 2016.

[29] Sylvie Chambon, Peggy Subirats, Jean Dumoulin, "Introduction of a wavelet transform based on 2D matched filter in a Markov Random Field for fine structure extraction," *Application on road crack detection, SPIE-IS&T,* vol. 7251, 2009.

[30] Peggy Subirats, Jean Dumoulin, Vincent Legeay, Dominique Barba, "AUTOMATION OF PAVEMENT SURFACE CRACK DETECTION USING THE CONTINUOUS WAVELET TRANSFORM," in *International Conference on Image Processing,* Atlanta, GA, 2006.

[31] Lei Zhang, Fan Yang, Yimin Daniel Zhang, Ying Julie Zhu, "ROAD CRACK DETECTION USING DEEP CONVOLUTIONAL NEURAL NETWORK," in *International Conference on Image Processing*, Phoenix, AZ, 2016.

[32] Chih-Chung Chang, Chih-Jen Lin, "LIBSVM: A Library for Support Vector Machines," *ACM Transactions on Intelligent Systems and Technology,* vol. 2, no. 3, 2011.

[33] Yoav Freund, Robert E. Schapire, "A Short Introduction to Boosting," *Journal of Japanese Society for Artificial Intelligence,* vol. 14, no. 5, pp. 771-780, 1999.

[34] S. K. Sinha, "Automated Condition Assessment of Buried Pipeline Using Computer Vision Techniques," *E(I) Journal-CP,* vol. 85, pp. 38-43, 2004.

[35] Sunil K. Sinha, Paul W. Fieguth, "Automated detection of cracks in buried concrete pipe images," *Automation in Construction,* vol. 15, pp. 58-72, 2006.

[36] Bruno Cornelis, Yun Yangi, Joshua T. Vogelsteinf, Ann Dooms, Ingrid Daubechies, David Dunson, "Bayesian crack detection in ultra high resolution multimodal images of paintings," in *Digital Signal Processing*, Fira, Greece, 2013.

[37] Peter J. Burt, Edward H. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE TRANSACTIONS ON COMMUNICATION,* vol. 31, no. 4, pp. 532-540, 1983.

[38] "A multilinear singular value decomposition," *Journal on Matrix Analysis and Applications,* vol. 21, pp. 1253-1278, 2000.

[39] Christopher K.I. Williams, David Barber, "Bayesian Classification With Gaussian Processes," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE,* vol. 20, no. 12, pp. 1342-1351, 1998.

[40] H. Liu, H. Motoda, Feature Selection for Knowledge Discovery and Data Mining, Norwell, MA: Kluwer Academic Publishers, 1998.

[41] Shengkun Xie, Feng Jin, Sridhar Krishnan, Farook Sattar, "Signal feature extraction by multi-scale PCA and its application to respiratory sound classification," *Medical & Biological Engineering & Computing,* vol. 50, no. 7, p. 759–768, 2012.

[42] Chris Snijder, Uwe Matzat, Ulf-Dietrich Reips, "“Big Data”: Big Gaps of Knowledge in the Field of Internet Science," *International Journal of Internet Science,* vol. 7, no. 1, pp. 1-5, 2012.

[43] Stuart J. Russell, Peter Norvig, Artificial Intelligence: A Modern Approach, Upper Saddle River, NJ: Prentice Hall, 2009.

[44] Alan Stuart, Keith Ord, Kendall's Advanced Theory of Statistics, Volume 1, Distribution Theory, 6th Edition, Hoboken, NJ: Wiley, 1994.

[45] Ferdinand van der Heijden, Robert P. Duin, Dick de Ridder, David M. J. Tax, Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB, West Sussex, England: John Wiley & Sons, 2004.

[46] Andrew McCallum, Kamal Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," in *Association for the Advancement of Artificial Intelligence*, Madison, WI, 1998.

[47] Jason D. M. Rennie, Lawrence Shih, Jaime Teevan, David R. Karger, "Tackling the Poor Assumptions of Naive Bayes Text Classifiers," in *Twentieth International Conference on Machine Learning*, Washington, DC, 2003.

[48] George H. John, Pat Langley, "Estimating Continuous Distributions in Bayesian Classifiers," in *Conference for Uncertainty in Artificial Intelligence*, San Mateo, CA, 1995.

[49] Peng Sun, Zhengfeng Ming, Ziyan Wu, Haifeng Yang, Xin Guo, "Research Methods of Sensors Validation Based On Naive Bayesian Classifier," in

*International Conference on Computational Intelligence and Security*, Shenzhen, China, 2015.

[50] Dhiman Mondal, Aruna Chakraborty, Dipak Kumar Kole, D. Dutta Majumder, "Detection and Classification Technique of Yellow Vein Mosaic Virus Disease in Okra Leaf Images using Leaf Vein Extraction and Naive Bayesian Classifier," in *International Conference on Soft Computing Techniques and Implementations*, Faridabad, India, 2015.

[51] Yazhe Hu, Tomonari Furukawa, "A High-Resolution Surface Image Capture and Mapping System for Public Roads," *SAE Int. J. Passeng. Cars – Electron. Electr. Syst.*, vol. 10, no. 2, 2017.

[52] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision,* vol. 60, no. 2, p. 91–110, 2004.

[53] D. Rutovitz, "Pattern recognition," *Journal of Royal Statistical Society,* vol. 129, no. 4, pp. 504-530, 1966.

[54] A. Rosenfeld, "A characterization of parallel thinning algorithms," *Information and Control,* vol. 29, no. 3, pp. 286-291, 1975.

[55] Z. Guo, R.W. Hall, "Fast fully parallel thinning algorithms," *CVGIP: Image Understanding,* vol. 55, no. 3, p. 317–328, 1992.

[56] Y. Wan, L. Yao, B. Xu, P. Zeng, "A Distance Map Based Skeletonization Algorithm and its Application in Fiber Recognition," in *International Conference on Audio, Language and Image Processing*, Shanghai, China, 2008.

[57] L. G. Shapiro, G. C. Stockman, "Computer Vision", Upper Saddle River, NJ: Prentice Hall, 2001.

[58] J. Canny, "A computational approach to edge detection," *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE,* vol. 8, no. 6, pp. 679-698, 1986.

[59] D. Martin, C. Fowlkes, D. Tal, J. Malik, "A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring

Ecological Statistics," in *Proc. 8th Int'l Conf. Computer Vision*, Vancouver, BC, Canada, 2001.

[60] J. Fernandez, "Image Processing to Detect Worms," Uppsala Universitet, Uppsala, Sweden, Sep. 2010.

# Appendix

This appendix is provided as a resource to show some of the best and worst examples of crack detection. Little past crack detection research has provided such an exhaustive and brutally honest presentation of results, so this is intended to give a visual glossary of the proposed technique performing in a variety of situations (crack type, pavement type, illumination angle, etc.). Each image is given for the optimal summation limit as determined manually. The first six images are some of the best results for their class, and the last five are some of the worst for their class.