Automated Underground Pipe Inspection Using a Unified Image Processing and Artificial Intelligence Methodology

By

Sunil K. Sinha

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Doctor of Philosophy

in

Civil Engineering and Systems Design Engineering

Waterloo, Ontario, Canada, 2000

The University of Waterloo requires the signatures of all persons using or photocopying this thesis. Please sign below, and give address and date.

Abstract

The National Science Foundation (NSF) has estimated the total U.S. investment in civil infrastructure systems at US $ 20 *trillion*. The investments in underground infrastructure systems represent a major component of this overall investment. Recent studies have shown that the cost of replacing all water mains in the United States would run to US $ 348 *billion*. The estimated cost to upgrade the water transmission and distribution system is US $ 77 billion. In Canada, the estimated cost to bring the water mains and sewer pipelines to an acceptable level is CDN $ 11.5 *billion* and 47 *billion*, respectively. Many of these pipeline systems are eroding due to ageing, excessive demand, misuse, poor construction, mismanagement and neglect. Due to their lack of visibility, rehabilitation of underground pipeline is frequently neglected until a catastrophic failure occurs, resulting in difficult and costly rehabilitation.

The enormity of the problem of deteriorating municipal pipeline infrastructure is apparent. Since neglecting or rebuilding the pipeline system is not financially realistic, asset managers require the capacity to monitor the condition of underground pipes. Thus, reliable cost effective pipeline assessment methods are necessary so that pipeline managers can develop long-term cost effective maintenance and rehabilitation programs. These programs are necessary to ensure that critical pipeline sections are repaired or replaced before they fail.

Closed circuit television (CCTV) surveys are used widely in North America to assess the structural integrity of underground pipes. The video images are examined visually and classified into grades according to degrees of damage. The human eye is extremely effective at recognition and classification, but it is not suitable for assessing pipe defects in thousand of miles of pipeline images due to fatigue and cost. In addition, manual inspection for surface defects in the pipeline has a number of drawbacks, including subjectivity, varying standards, and inspection time. These concerns have motivated us to conduct a research for the development of an automated pipe inspection system, based on the scanned images of underground pipes.

This thesis presents a system for the application of computer vision techniques to the automatic assessment of the structural condition of underground pipes. Automatic recognition of various pipe defects is of considerable interest since it has the potential to solve problems of fatigue, subjectivity, and ambiguity, leading to economic benefits. The main efforts of the research are placed on investigating algorithms and techniques for image pre-processing, segmentation of pipe objects (i.e., cracks, holes, joints, laterals, and collapse surface), crack detection, feature extraction and classification of defects. In this study, an attempt has also been made to develop a framework for an integrated pipeline network management. The proposed integrated pipeline management system is necessary to help municipal managers to make consistent and cost-effective decisions related to the preservation of underground pipeline systems.

Acknowledgements

There are a great number of people who have contributed significantly to my Ph.D. years at the University of Waterloo.

Most significant, of course, is the help and encouragement which I received over the past few years from Paul Fieguth. I thank him for his enthusiasm and for his careful supervision of my research, from which I benefited tremendously over the course of my Ph.D., and also for his attitudes and insights regarding the supervisor-student mentoring process, which may very well be some of the most valuable lessons that I take with me from the University of Waterloo. I also appreciate the considerable assistance from Marianna Polak, particularly her valuable suggestions and discussions in the phases of developing an integrated underground pipeline management system.

The precious comments and suggestions of the committee comprised of Frank Saccomanno, Mahesh Pandey, Deborah Stacey, and Edward Vrscay (University of Waterloo), and Carl Haas (University of Texas, Austin) are highly appreciated.

I greatly appreciate the financial support provided by the Natural Science and Engineering Research Council of Canada, the Ontario Graduate Scholarship, the University of Waterloo Graduate Scholarship, and the Faculty of Engineering Scholarship during my three years of doctoral study at the University of Waterloo.

Thanks to my fellow students at the Visual Image Processing Group for their valuable friendship and cooperation in the study.

Finally my wife, Dr. Anju Sinha, and our son, Sushruth deserve thanks for their remarkable tolerance and sharing so many hours that were rightfully theirs. Without their encouragement and love this work could not have been completed.

Contents

List of Figures

List of Tables

Chapter 1

1. Introduction

Beneath North America's roads lie thousands of miles of pipe that bring purified water to homes and carry away wastewater (sewage and storm water). For the most parts, these systems have been functioning longer than their intended design life (i.e., fifty years for concrete pipe), with little or no repair. They are in a state of deterioration.

Maintenance and rehabilitation (M&R) of pipeline systems pose a major challenge for most municipalities in North America given their budgetary constraints, the demand on providing quality service and the need for preserving their pipeline infrastructure. Neglecting regular M&R of these underground pipelines adds to life-cycle costs and liabilities, and in extreme cases causes stoppage or reduction of vital services.

Accurate pipeline condition assessment is vital to developing a cost effective and efficient pipeline M&R program. At present, the assessed condition of underground pipes is based on the subjective visual inspection of closed circuit television (CCTV) surveys [84]. CCTV surveys are conducted using a remotely controlled vehicle carrying a television camera through an underground pipe. The data acquired from this process consist of videotape, photographs of specific defects, and a record produced by the technician. Typical scanned images of CCTV surveys and the data section of the standard coding form are shown in

Figures 1-1 and 1-2, respectively. Diagnosis of defects depends on the experience, capability, and concentration of the operator, making the detection of defect error prone. A large number of new technologies such as Pipe Scanner and Evaluation Technology (PSET) [83], laser-based systems [179], etc. have made it possible to obtain high quality images of pipes. PSET is an innovative technology for obtaining unfolded images of the interior of pipes [83]. This is accomplished by utilizing scanner and gyroscope technology. Typical scanned image of PSET surveys are shown in Figure 1-3. Although underground imaging technology has made substantial strides in recent years, the basic means of analysis are unchanged: a technician is required to identify defects on a television monitor. The research of this thesis seeks to address this latter limitation; thus, allowing the technician to do exactly what they have been trained to do, which is to insure that the inspection equipment is being operated properly.

This thesis addresses the development of an automated underground pipe inspection system. Main efforts have been placed on investigations of algorithms and techniques for image processing, feature extraction and pattern classification. In particular, this research has explored how various signals and image processing concepts, nonlinear filtering, feature extraction, pattern classification and artificial intelligence techniques can be judiciously synthesized for computationally efficient and robust identification of underground pipe defects (i.e., cracks, holes, collapse surface, and defective joints and laterals). The proposed automated system could overcome many of the limitations of the current manual inspection of pipes, and can provide a more accurate assessment of underground pipe conditions. An attempt has also been made to develop a framework for integrated underground pipeline management system. An integrated pipeline management system is necessary to help municipal engineers to make consistent and cost-effective decision related to the preservation of underground pipeline systems.

This introduction should serve only to introduce the notions of an automated pipe inspection system; a more thorough description of the computer vision techniques and related mathematical definitions, and a detailed explanation of the different approaches may be found in Chapter 2. The next section will outline the thesis research motivation, followed by the contribution of this thesis and a description of the thesis organization.

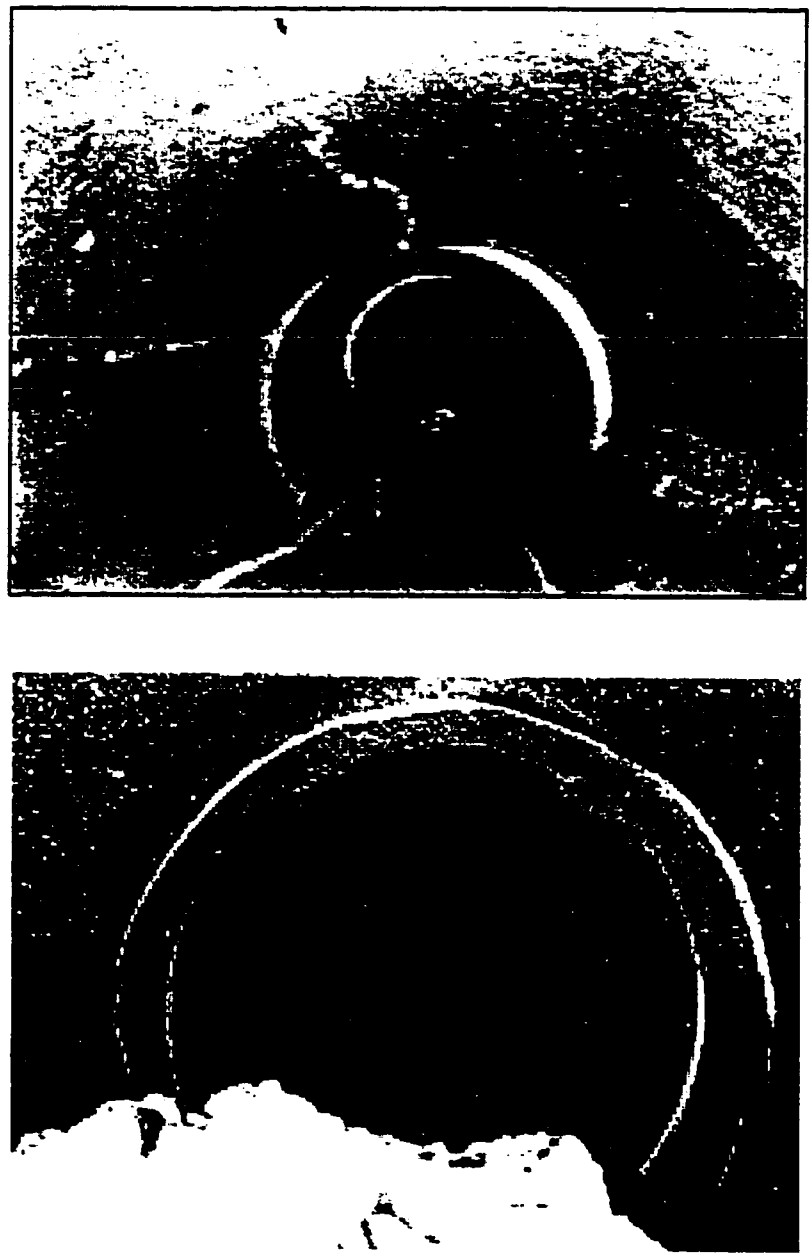


Figure 1-1. Typical images of underground pipe scanned by closed circuit television (CCTV) camera in the city of Toronto.

CONSTRUCTION MATERIALS						
STRUCTURE DATA	Concrete	Brick Lined	FRP (Resin)	PE	Lined	Other
Base Structure						
Other						

Install Date	Length	Width	Depth

CONNECTION DATA (Pipe or Trench)							CONSTRUCTION MATERIALS		
LOCATION Refer on Sketch		Elev **	Dia	VT	CI	PE	FRP? Insitu	Brick Trench	OTHER (See comments)
Out	6:00								
Out									
In									
In									
In									

PIPE SEGMENT LOG SHEET

Event Number:		Segment Number:		Date:	
Work Order #:		Time of Inspection:		Diameter:	
Direction of Camera:		☐ Upstream ☐ Downstream			
Inspectors :					

Clock Position	Defect	Severity		Clock Position
			NFFU	
Start Footage	Finish Footage	Comments		

Figure 1-2. Data section of the standard coding form of CCTV surveys used by the North American Association of Pipeline Inspectors (NAAPI).

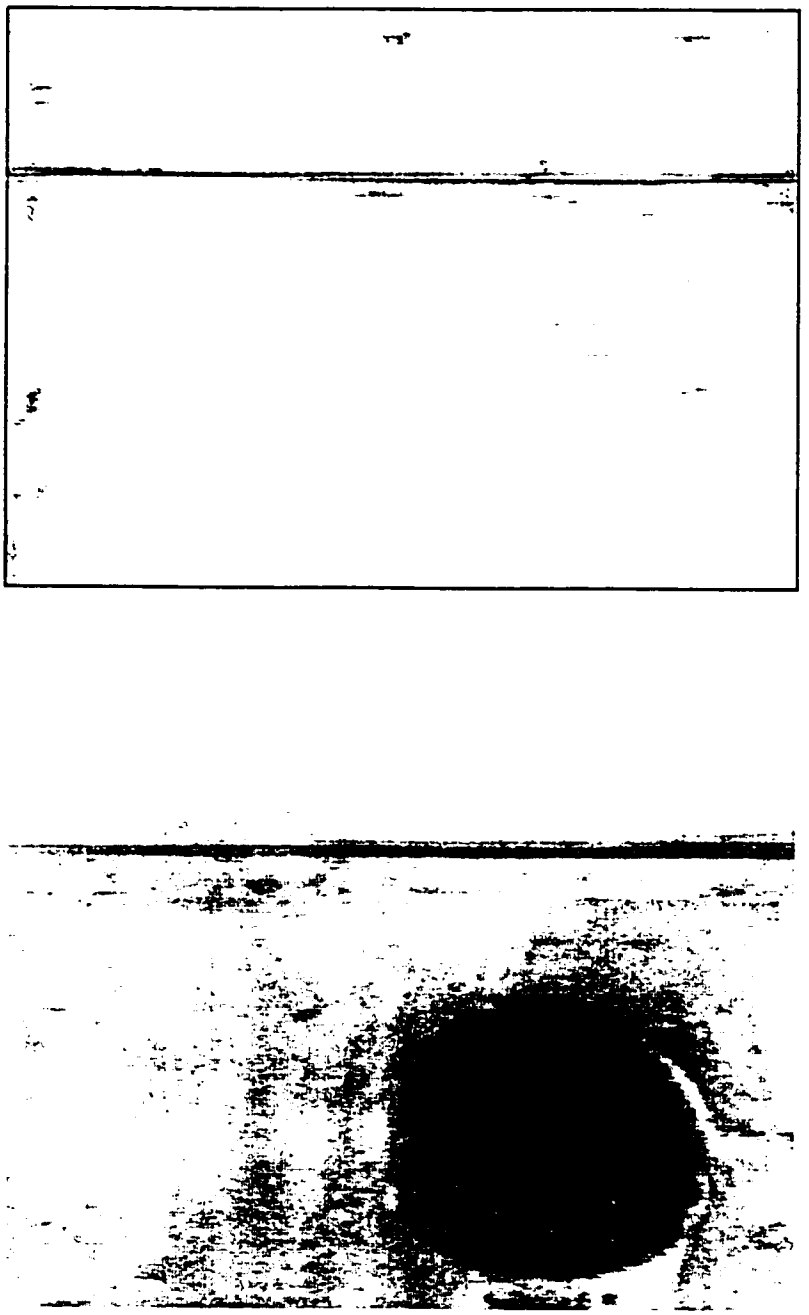


Figure 1-3. Typical images of underground pipe scanned by Pipe Scanner and Evaluation Technology (PSET) Camera in the city of Toronto.

1.1 Thesis Motivations

Visual inspection based on closed circuit television surveys is used widely in North America to assess the condition of underground pipes [193]. The human eye is extremely effective at recognition and classification, but it is not suitable for assessing pipe defects in thousand of miles of pipeline due to fatigue, subjectivity, and cost. This drawback of the present visual inspection of underground pipes has been one of the main motivations behind our interest in developing an automated pipe inspection system. Automatic recognition of various pipe defects is of considerable interest since it has the potential to solve problems of fatigue, subjectivity, and ambiguity, leading to economic benefits.

Most of the literature concerning the detection/classification of defects in civil structures deal with the analysis of pavement [32,33,66,183] and concrete/steel distresses [30,36], analyses which are not directly applicable to underground pipe inspection. In analyzing scanned underground pipe images, one needs to consider complications due to the inherent noise in the scanning process, irregularly shaped cracks, as well as the wide range of pipe background patterns. One of the major problems is detecting defects (especially cracks) that are camouflaged in the background of corroded areas, debris, patches of repair work, and areas of poorly illuminated conditions. The above aspects, also motivated the development of computationally efficient and robust algorithms for segmenting, detecting, and classifying various defects in underground pipe images.

1.2 Contributions

The primary objective of this research is to develop an automated underground pipe inspection system, based on the scanned images obtained from PSET survey for major cities in North America. An attempt has also been made to set out the framework for an integrated pipeline management system based on automated inspection. In an underground pipe inspection system, the main objective is to identify, extract and classify defects to facilitate preventive maintenance.

To put the work of this thesis into context, when this research began the following goals had already been accomplished:

- Automatic segmentation of underground pipe images
- Detection of cracks in segmented pipe images
- Classification of pipe defects based on feature extraction
- Development of a framework for integrated pipe management

The further contributions of this thesis, listed below, build upon these accomplishments.

- **Morphological Segmentation of Pipe Images**

Image segmentation is a crucial pre-processing step before object classification. A poor segmentation will greatly affect the shape of the object in the resultant binary image and hence the successful rate of subsequent image classification processes. Analyses of images have shown that there are two important characteristics that complicate the segmentation of underground pipe images: firstly, the presence of a complicated background pattern due to earlier runoff, patches of repair work, corroded areas, debris, non-uniformities in illumination, and flaws in the image acquisition process; secondly, because the three main objects of interest - cracks, joints, and laterals - are all dark features that cannot be distinguished by intensity criteria alone.

In this thesis, a simple, robust and efficient image segmentation algorithm for the automatic analysis of underground pipe images has been developed. The algorithm consists of a gray-scale conversion step followed by a sequence of morphological operations to accurately classify pipe cracks, holes, joints, laterals and collapse surface, a crucial step in the detection of surface cracks. The proposed approach can be completely automated and the experimental results demonstrate that the segmentation algorithm can precisely classify various objects in the underground pipe images.

- **Development of Crack Detection Filters**

Usual edge detectors and thresholding techniques based on the difference between pixel values are inefficient when applied to the underground pipe images. In analyzing underground pipe scanned image, one needs to consider complications due to the inherent noise in the scanning process, irregularly shaped cracks, as well as the wide range of pipe background patterns. One of the major problems is detecting defects (especially cracks) that are camouflaged in the background of corroded areas, debris, patches of repair work, and areas of poorly illuminated conditions.

A significant contribution of this thesis is the development of statistical filters for detection of cracks in the underground pipe images. Comparing the proposed crack detection filters and the conventional detection techniques, the improved experimental results have been achieved by the proposed statistical filters.

- **Development of Neuro-Fuzzy Classifier**

In underground pipe defect analysis, the main objective is to accurately classify cracks, holes, laterals, joints, and pipe collapse by type, severity, and extent of distress. This thesis makes a relevant contribution in developing a neuro-fuzzy classifier based on suitable features extracted from segmented underground pipe images. The proposed algorithm employs a fuzzy membership function and a projection neural network. The former absorbs variation of feature values and the latter shows good learning efficiency.

- **Development of Pipe Performance Prediction Model**

Most municipal pipeline repair and rehabilitation have been performed using 'management by crisis' techniques. Research in the pavement management area has shown that 'worst first' or 'crisis management' does not lead to the optimum use of resources. Municipal managers and engineers are realizing that improvement in public health and safety as well as cost benefits can be achieved with a proactive municipal pipeline management system.

A significant contribution of this thesis is the development of a framework for an integrated pipeline management. The proposed pipeline management system combines automated condition assessment, and predictions of future conditions and will help asset managers and engineers to make consistent and cost-effective decision related to the preservation of municipal pipeline systems.

1.3 Thesis Organization

Chapter 2 presents the background relevant for the development of an automated pipe inspection system. It begins with a broad overview of various pipeline assessment techniques. It then briefly introduces the methodology for automated image-based inspection. The next two sections discuss image segmentation and feature extraction methods. Next, we review the pattern recognition tasks followed by the performance model and optimization methods.

Chapter 3 presents the work on the pre-processing and segmenting of underground pipe images. The chapter begins by enhancing the contrast between pipe background and defects. Next, the chapter discusses a sequence of morphological operations to accurately segment pipe cracks, holes, joints, laterals, and collapse pipe, a crucial step in the classification of defects in underground pipe images.

Chapter 4 presents simple, robust, and efficient statistical filters for the detection of cracks in underground pipe images. The chapter begins by presenting the development of statistical filters, followed by the performance evaluation of the proposed crack detection filters with that of other conventional detection techniques.

Chapter 5 describes the feature extraction techniques and pattern recognition strategies for the classification of underground pipe defects. A formulation of a neuro-fuzzy system is presented in this chapter, together with the model input requirements and output format.

Chapter 6 provides a conceptual framework of the proposed integrated system for predicting pipe deterioration and multi-year priority programming of pipeline network maintenance. Three major components underlying the integrated system are described, including model development for pipe deterioration, standardized alternative pipe maintenance and rehabilitation treatment strategies, and priority programming on the basis of cost-effectiveness.

Chapter 7 summarizes the result of this thesis, presents the major contributions and details a number of avenues for further research.

Chapter 2

2. Background

The main objective of this chapter is to introduce and to motivate a foundation for development of an automated underground pipeline inspection system. Section 2.1 presents an overview of underground pipeline inspection techniques while Section 2.2 discusses the methodology for the development of an automatic image-based inspection system. Section 2.3 introduces the concept of image processing and segmentation. Section 2.4 briefly mentions the feature extraction and analysis techniques. Section 2.5 discusses the basic philosophy behind the development of pattern recognition models to classify various underground pipe defects. Finally Section 2.6 presents the performance models and optimization methods for the development of an integrated pipeline management system.

2.1 Overview of Pipeline Assessment Techniques

Pipelines are now an integral part of the world's economic structure and literally billions of dollars worth of products are now moved annually in pipelines [8]. Both economic and environmental factors are influential in pipeline operation, and therefore integrity monitoring is vitally important in the control and operation of complex system.

Pipeline defect detection systems range from simple, visual inspection to complex inspection systems. No one method is universally applicable and operating requirements dictate which method is the most cost effective. The aim of this section is to review the basic techniques of defect detection that are currently in use. The advantages and disadvantages of each method are discussed and some indications of applicability are outlined.

2.1.1 Destructive Testing Methods for Pipeline Defect Assessment

Destructive testing methods consist of testing a product beyond its design load until a brittle failure develops. Often these tests are specified to ensure that the pipe material meets or exceeds design specifications. Common destructive test methods are discussed in the following.

Tension. Tension testing [9] consists of placing coupons of pipe material in a testing apparatus and pulling the coupon apart until failure occurs. During a test, the coupon load and deflection are measured. This test ensures that the pipe material meets or exceeds tensile strength design requirements and is useful for determining the load bearing capabilities of the material and its ductility under load.

Hardness. Hardness is usually defined as resistance to penetration. Other tests are also used, such as amount of rebound of a weight and scratch tests. Hardness tests include Rockwell [42], Scleroscope [42], Brinell [42], and Vickers[42]. Good correlation exists between hardness and strength. Thus, the hardness test can give a quick estimate of the pipe strength and/or wear resistance.

Impact. The pendulum impact test, also known as the Charpy test [10], gives an indication of the amount of energy that can be absorbed by the material. The test procedure consists of cutting a notch in a sample then hitting the sample with a hammer. The difference in energy that the hammer has after striking the material is the material

impact strength.

Flexural. The three-edge-bearing flexural test is a severe pipe loading condition. This test allows for no lateral pipe support (unconfined) and applies forces that are point loads [42].

The Parallel Plate. The parallel plate test [42] is an accepted method for measuring pipe stiffness. This test consists of placing the pipe between two parallel plates located at the top and bottom of the pipe. The pipe stiffness is determined from the load required to displace the top plate a specified deflection.

2.1.2 Non-Destructive Testing Methods for Pipeline Defect Assessment

Non-destructive testing (NDT) [41] is the branch of engineering concerned with non-contact methods of detecting and evaluating defects in materials. Defects can affect the serviceability of the material or structure, so NDT is important in guaranteeing safe operation as well as in quality control and assessing pipe life. The defect may be cracks or inclusions in welds and castings, or variations in structural properties that can lead to loss of strength or failure in service. Non-destructive testing is used for in-service inspection and for condition monitoring [179]. It is also used for measurement of components and for the measurement of physical properties such as hardness and internal stress. The essential feature of NDT is that the test process itself produces no deleterious effects on the material or structure under test. The subject of NDT has no clearly defined boundaries: it ranges from simple techniques such as visual examination of surfaces, through the well-established methods of radiography, ultrasonic testing, magnetic particle crack detection, to new and very specialized methods. NDT methods can be adapted to automate production processes as well as to the inspection of localized problem areas. All NDT techniques have the ability to measure only specific types of defects, material properties, and/or material response. Therefore, the best choice of an NDT method in a specific pipeline application will depend on the pipeline physical properties and defects. Thus, before the selection of an appropriate NDT method, a through knowledge of each

NDT method application and its limitations is required along with a good understanding of the piping system.

Non-Visual NDT Methods

Ultrasonic Inspection (Sonar). Ultrasonic inspection [20] is performed using a beam of very high frequency coherent sound energy, with the frequency being many orders of magnitude higher than a human being can hear. The sound wave travels into the object being inspected and reflects whenever there is a change in the density of the material, with some of the energy in the wave returning to the surface and some passing on through the new material. The ultrasonic inspection process is shown in Figure 2-1. Ultrasonic beams can be used to image the human body, inspect aircraft, or examine oil pipelines. The technique is capable of detecting pits, voids, and cracks, although certain crack orientations are much more difficult to detect than others. The ultrasonic wave reflects most easily when it crosses an interface between two materials that are perpendicular to the wave. As an example, cracks that lie perpendicular to the wave are easily detected, but cracks that lie parallel to the beam are usually not identified by an ultrasonic examination. Evaluation is often difficult [20].

Eddy current testing. Eddy current testing [89] is an electromagnetic technique that can detect surface and subsurface discontinuities in tube walls up to about 3/8" (10mm) thick on conductive materials. Applications range from crack detection, to the rapid sorting of small components for defects, size variations, or material variation. When an energized coil is brought near to the surface of a metal component, eddy currents are induced into the specimen. These currents set up a magnetic field that tends to oppose the original magnetic field. The impedance of the coil in close proximity to the specimen is influenced by the presence of the induced eddy currents in the specimen. When the eddy currents in the specimen are distorted by the presence of a defect or a material variation, the impedance in the coil is altered. This change is measured and displayed in a manner that indicates the type of defect or material condition. This method is commonly performed on heat exchanger tubing by inserting a probe down the full length of each

tube to be inspected as shown in Figure 2-2. The probe contains a coil arrangement, energized by alternating currents operating at one or more frequencies. The electrical impedance of the test coil arrangement is modified by the proximity of the tube, tube dimensions, electrical conductivity, magnetic permeability of the tube material, and metallurgical and mechanical discontinuities. Wear on the tube surface under a support is also detectable. The electromagnetic response caused by passing these variables produces electrical signals, which are processed electronically to produce a visual response characteristic of the change encountered. Visual responses, often called "Signatures" are displayed on the test instrument monitor for evaluation by the field technician (Analyst).

Acoustic emission monitoring. This method involves listening to the sounds (which are usually inaudible to the human ear) made by a material, structure or machine in use or under load [16]. Conclusions are made about its "state of health" from what is heard, just like the Doctor who listens to your heart and lungs. The technique involves attaching one or more ultrasonic microphones to the object and analyzing the sounds using computer-based instruments. Acoustic emission inspection methodology is shown in Figure 2-3. The noises may arise from friction (including bearing wear), crack growth, turbulence (including leakage) and material changes such as corrosion. Applications include testing pipelines and storage tanks (above and below the ground), fiberglass structures, rotating machinery, weld monitoring, and biological and chemical changes.

Visual NDT Methods

Visual inspection is a NDT method used extensively to evaluate the condition or the quality of a component [100]. It is easy to perform, inexpensive and usually does not require special equipment. It is most effective for the inspection of welds where quick detection and the correction of defects or process-related problems can result in significant cost savings. It is the primary evaluation method of many quality control programs. The method requires good vision, good lighting and operator knowledge.

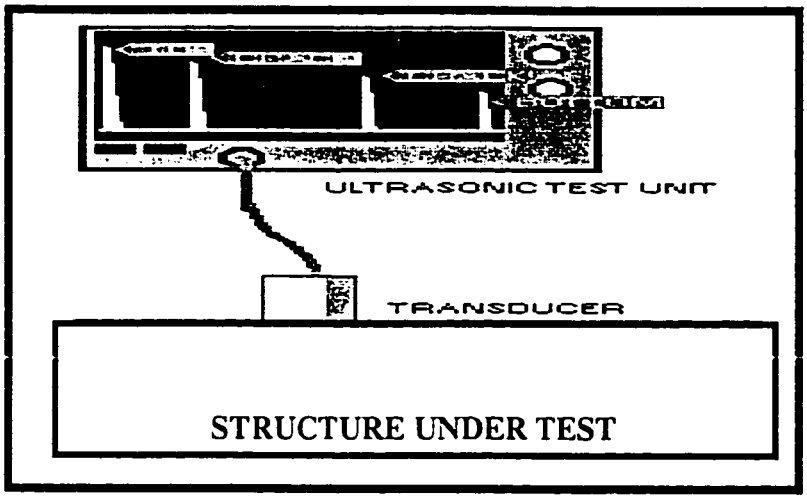


Figure 2-1. Ultrasonic inspection of pipes

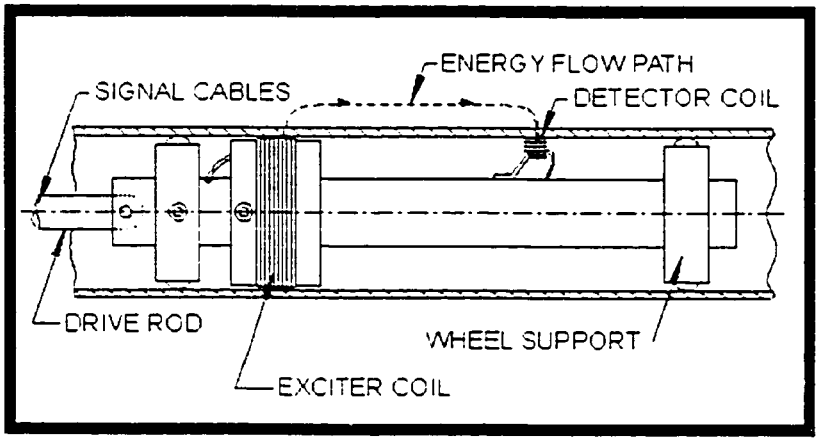


Figure 2-2. Eddy current inspection of pipes

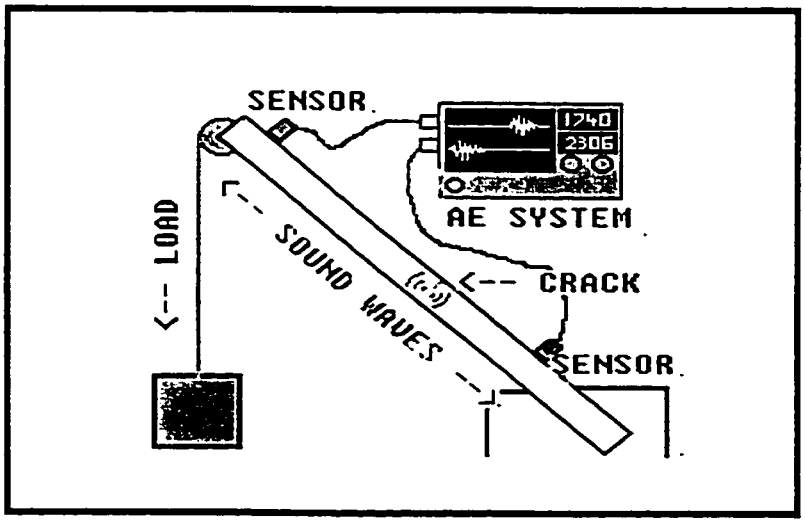


Figure 2-3. Acoustic emission for inspection of pipes

Most municipal pipeline systems are inspected visually by mobile Closed Circuit Television (CCTV) systems or human inspectors [84]. CCTV examination using a mobile camera system is the typical approach to this type of examination. However, there are several CCTV variants that may reduce the cost of the inspection or provide improved results. There are also alternative techniques that will work where CCTV will not or that will give direct measurements of pipe condition as opposed to the estimates produced by CCTV inspection. Each inspection technique is discussed separately below.

CCTV Inspection. There are two basic types of CCTV system [193]. Each uses a television camera in conjunction with a video monitor, videocassette recorders, and possibly other recording devices. In one case the inspection is performed using a stationary or zoom camera mounted at a manhole so that it looks into the pipe, whereas in the other a mobile, robotic system is placed within the pipe itself. The camera provides images to an operator who is trained to detect, classify and rate the severity of defects against documented criteria [83]. The typical CCTV camera and scanning process of underground pipes are shown in Figure 2-4. Either form of CCTV inspection may miss certain types of defects, especially those that are hidden from the camera by obstructions as it looks down the pipe. This method is also vulnerable to lapses in operator concentration, inexperience, and the inability of the image to reveal important defects [193]. Thus, the results are widely agreed to lack reliability and consistency and the reliability to track deterioration so that preventive maintenance can be undertaken with confidence. It does however; provide useful information on gross defects.

Pipe Scanner and Evaluation Technology. Pipe scanner and evaluation technology (PSET) [83] is an innovative technology for obtaining images of the interior of pipe. PSET was developed by TOA Grout, CORE Corp., California, and the Tokyo Metropolitan Government's Services (TGS) Company. PSET is a system that offers a new inspection method minimizing some of the shortcomings of the traditional inspection equipment that relies on a CCTV inspection. This is accomplished by utilizing scanning and gyroscopic technology. The mechanics of inspecting the pipes by PSET camera are

similar to the CCTV inspection. The PSET is designed to operate from a tractor platform to propel the tool through the pipe. Since the PSET utilizes state-of-the-art scanner technology, it can travel through the pipe at a uniform rate of speed. The PSET camera and digitized images of underground pipe with various defects are shown in Figure 2-5. The major benefit of the PSET system over the current CCTV technology is that the engineer will have higher quality image data to make critical rehabilitation decisions.

Laser-Based Scanning Systems. In addition to the simple light line system described above, lasers have been used in the past to evaluate both the shapes of pipelines and the types of defects they contain [77]. These systems are restricted to the part of the pipe above the waterline, but they can, in theory, make possible extremely accurate inspections of pipe condition. An additional advantage to this approach is that the information from the laser scans is readily recorded and analyzed by computer, substantially reducing operator errors. Although the initial equipment may be more expensive than a CCTV system, the reduced operator time necessary to use the technique may also mean that its operation will be more economical. The technology is still in the development stage.

2.2 Methodology for Automatic Image-Based Inspection

Industry is increasingly using machine vision systems to aid in the manufacturing and quality-control processes [129]. The goal of a machine vision [36] is to create a model of the real world from images. A machine vision system recovers useful information about a scene from its two-dimensional projections. Since images are two-dimensional projections of the three-dimensional world, the information is not directly available and must be recovered. To recover the information, knowledge about the objects in the scene is required. The emphasis in machine vision systems is on maximizing automatic operation at each stage, and these systems should use knowledge to accomplish this.

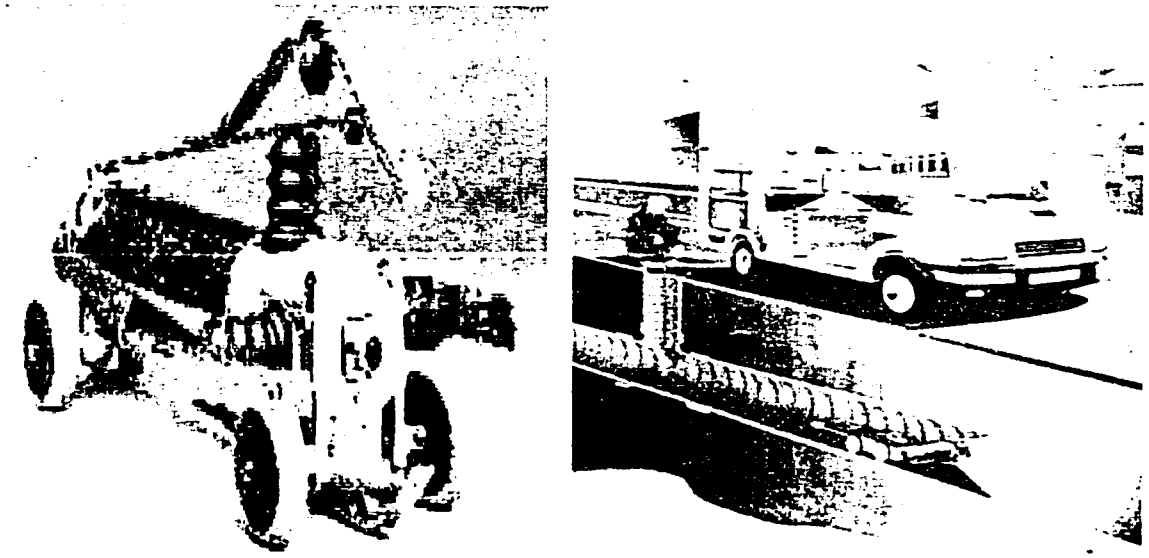


Figure 2-4. CCTV camera and underground inspection process (from trenchless technology magazine, April 1999) [145].

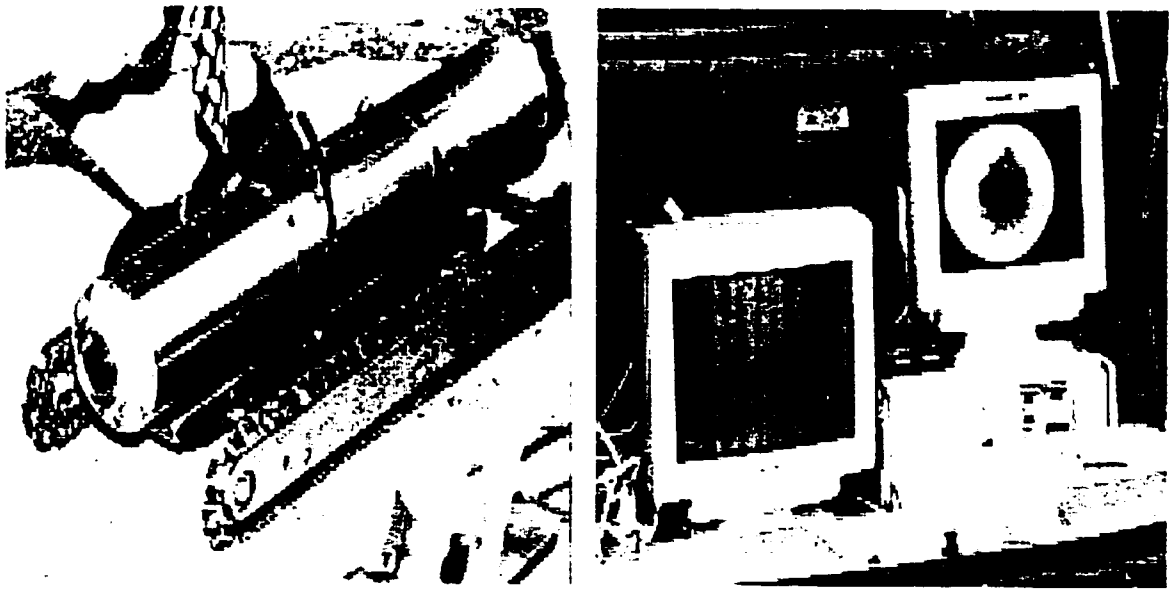


Figure 2-5. PSET inspection probe and digitized image of underground pipe (from trenchless technology magazine, April 1999) [145].

Machine vision emulates human vision in that it attempts to interpret images. Human vision deals with the global information available in a scene, resolves ambiguities due to perspective, lighting, and attribute, and can perform guidance through unfamiliar territories. In fact, human vision has been shown to be incapable of performing reliable inspection [6], since the human vision process is prone to subjective considerations, fatigue, and boredom, which interfere with consistent evaluations. Also, the human vision is limited to the visible spectrum while machine vision system can exploit a much larger range of the electromagnetic spectrum, including infrared radiation, X-rays, and ultrasounds, thereby making it suitable for a wide range of nondestructive testing and inspection process-related tasks [14].

An automated image-based inspection aims to extract information from an image on the conditions of objects represented in the image. Usually it is impossible to extract this information concerning the dimensions of objects or their defect properties directly from the image. Basically, all machine vision systems involve image acquisition, image pre-processing, segmentation, and extracting relevant features for classification of the type, severity, and extent of defects present in the image.

2.3 Image Processing and Segmentation

Vision allows humans to perceive and understand the world surrounding them. Computer vision aims to duplicate the effect of human vision by electronically perceiving and understanding an image. Giving computers the ability to see is not an easy task – we live in a three-dimensional (3D) world, and when computers try to analyze objects in 3D space, the visual sensors available (e.g., TV cameras) usually give two-dimensional (2D) images, and this projection to a lower number of dimensions incurs an enormous loss of information.

The term digital image processing generally refers to processing of a two-dimensional picture by a digital computer. In a broader context, it implies digital processing of any

two-dimensional data. The first step in the process is image acquisition – that is, to acquire a digital image. After a digital image has been obtained, the next step deals with processing that image. This sequence of operations-image capture, processing, region extraction, high-level identification, qualitative/quantitative conclusion-is characteristic of image understanding and computer vision problems. Many computer vision techniques use the results and methods of mathematics, pattern recognition, artificial intelligence, psychophysiology, computer science, electronics, and other scientific disciplines [18]. In order to simplify the task of computer vision understanding, two levels are usually distinguished: *low-level* image processing [68] and *high-level* image understanding [68]. Low-level methods usually use very little knowledge about the content of images. In the case of the computer-knowing image content, it is usually provided by high-level algorithms or directly by a human who knows the problem domain. Low-level methods often include image pre-processing methods for noise filtering, contrast enhancement, and edge extraction.

Image pre-processing and segmentation is the initial stage for any recognition process, whereby the acquired image is 'broken up' into meaningful regions or segments. The segmentation process is not primarily concerned with what the regions represent, but only with the process of partitioning the image. In the simplest case (binary images) there are only two regions: a foreground (object) region and a background region. In gray level images, there may be many types of region's or classes within the image; for example, in a natural scene to be segmented, there may be regions of sky, clouds, ground, building and trees. There are, broadly speaking, two approaches to image segmentation, namely thresholding [143] and region or edge-based [143] methods.

2.3.1 Image Pre-processing

The principal objective of image pre-processing is to process an image so that the result is more suitable than the original image for a specific application. The word 'specific' is important, because it establishes at the outset that the techniques discussed in this section

are very much problem-oriented. Thus, for example, a method that is quite useful for enhancing x-ray images may not necessarily be the best approach for enhancing images of underground pipes.

Gray Scale Transformation

Gray scale transformations [153] do not depend on the position of the pixel in the image. A transformation λ of the original brightness p from scale $[p_o, p_k]$ into brightness q from a new scale $[q_o, q_k]$ is given by

$$q = \lambda(p) \tag{2.1}$$

The most common gray scale transformations are shown in Figure 2-6; the straight line a denotes the negative transformation; the piecewise linear function b enhances the image contrast between brightness values p_1 and p_2. The function c is called brightness thresholding and results in a black-and-white image.

A gray scale transformation for contrast enhancement is usually found automatically using the histogram equalization technique [153]. The aim is to create an image with equally distributed brightness levels over the whole brightness scale (see Figure 2-7). Histogram equalization enhances contrast for brightness values close to histogram maxima, and decreases contrast near minima.

Image Smoothing

Image smoothing [28] is the set of local processing methods whose predominant use is the suppression of image noise. Calculation of the new value is based on the averaging of brightness values in some neighborhood. Smoothing poses the problem of blurring sharp edges in the image, and so we shall concentrate on smoothing methods which are edge preserving. They are based on the general idea that the average is computed only from those points in the neighborhood that have similar properties to the point being processed.

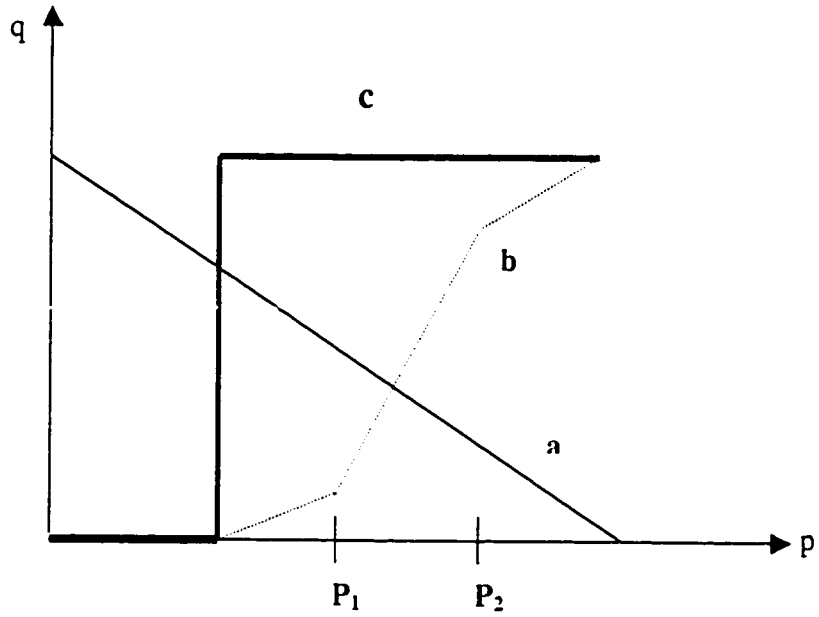


Figure 2-6. Some gray-scale transformations. The straight line *a* denotes the negative transformation; the piecewise linear function *b* enhances the image contrast between brightness values p_1 and p_2. The function *c* is called brightness thresholding and results in a black-and-white image.

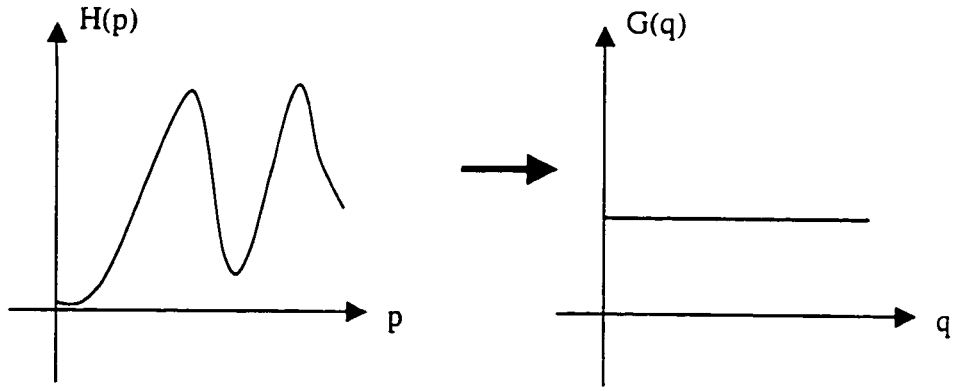


Figure 2-7. Histogram equalization. The aim is to create an image with equally distributed brightness levels over the whole brightness scale

Assume that the noise value v at each pixel is an independent random variable with zero mean and standard deviation σ. We can obtain such an image by capturing the same static scene several times. The results of smoothing is an average of the same n points in these images $g_1, \ldots, g_n$ with noise values $v_1, \ldots, v_n$:

$$\frac{g_1 + \ldots + g_n}{n} + \frac{v_1 + \ldots + v_n}{n} \tag{2.2}$$

Median filtering is a non-linear smoothing method that reduces the blurring of edges [119], in which the idea is to replace the current point in the image by the median of the brightness in its neighborhood. The median of the brightness in the neighborhood is not affected by individual noise spikes and so median smoothing eliminates impulse noise quite well. Further, as median filtering does not blur edges much, it can be applied iteratively.

Color Image Processing

The use of color in image processing [171] is motivated by two principal factors. First, in automated image analysis, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Second, in image analysis performed by human beings, the motivation for color is that the human eye can discern thousands of color shades and intensities, compared to about only two-dozen shades of gray [171]. Color is connected with the ability of objects to reflect electromagnetic waves of different wavelengths: the chromatic spectrum spans the electromagnetic spectrum from approximately 400nm to 700nm. Humans detect colors as combinations of the primary colors red, green, and blue, which for the purpose of standardization have been defined as 700nm, 546.1nm, and 435.8nm respectively [171], although this standardization does not imply that all colors can be synthesized as combinations of these three.

The purpose of a color model is to facilitate the specification of colors in some standard, generally accepted way. In essence, a color model is a specification of a 3-D coordinate system and a subspace within that system where each color is represented by a single point. The color models most often used are the RGB, the YIQ, and the HIS models.

RGB Color Model. The RGB (Red, Green, Blue) space is used most frequently in computer graphics and image processing applications [143,171]. A color in this space is represented by a triplet of values typically between 0 and 255. Each color can be broken down into its relative intensity in the three primaries corresponding to the spectral response of one of the three types of cones present in the human eye: red, green and blue. The space is easily represented as a three dimensional cube where each axis represents the strength of the color in one of the three primaries, as shown in Figure 2-8.

YIQ Color Model. The YIQ model (sometimes referred to as IYQ) [171] is useful in color TV broadcasting, and is a simple linear transform of an RGB representation:

$$\begin{pmatrix} Y \\ I \\ Q \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.275 & -0.321 \\ 0.212 & -0.523 & 0.311 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{2.3}$$

(with the inverse computed in the obvious manner). This model is useful since the Y component provides all that is necessary for a monochrome display; further, it exploits properties of the human visual system, in particular our sensitivity to luminance, the perceived energy of a light source. Details of this model and its use may be found in relevant texts [143,171].

HSI Color Model. The HSI (Hue, Saturation, and Intensity) [143] model can be said to follow more closely the human perception of color qualities. Hue (H) is the color as described by wavelength – for example, the distinction between red and blue. Hue represents the fundamental or dominant color. Saturation (S) represents the amount of a color present, where pastel shades (e.g. pink) have low saturation values while pure spectral colors (e.g. red) are completely saturated. The intensity (I) represents the overall brightness or the amount of light. It is independent of color and is a linear value. It is measured as an angle on a color circle with the three primary colors spaced 120° apart. The first two values specify the chromaticity of a color point. Figure 2-9 shows the relationship between the HSI and RGB spaces.

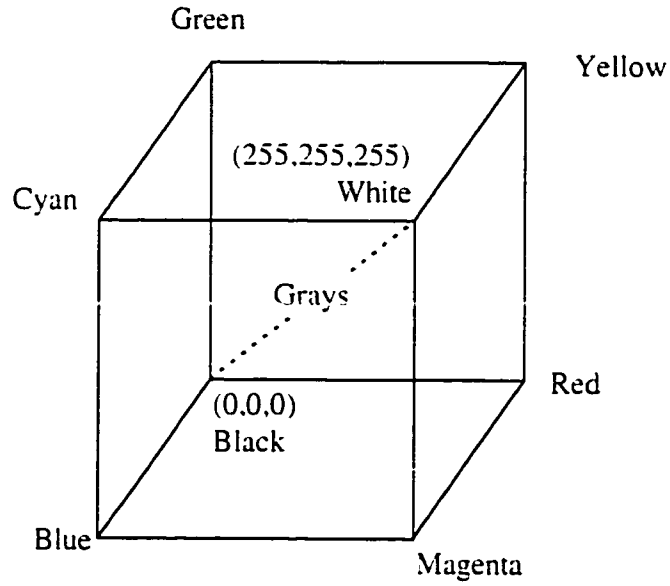


Figure 2-8. The RGB color space

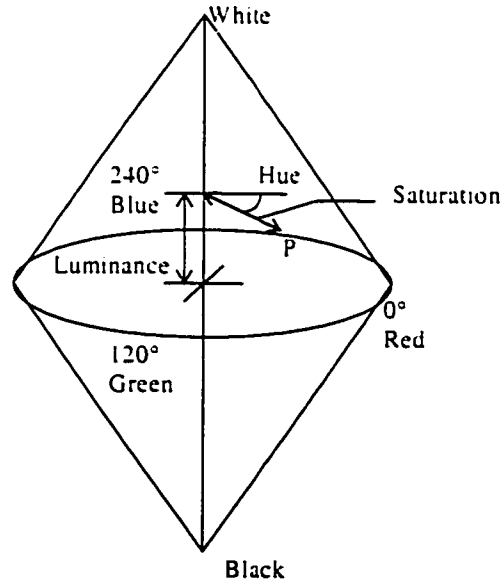


Figure 2-9. The HIS color space

It is noted that the HSI color space is one of several spaces that can be derived from the General Hue, Luminance and Saturation (GHLS) space [143]. There are other slightly different interpretations of hue, luminance and saturation. By setting certain parameters in this transformation, one can specify any of the transformations from the RGB space to an HSI, HLS or HSV (Hue, Saturation and Value) representation.

$$
\begin{aligned}
Cos\theta &= \frac{(R+G)+(R-B)}{2\cdot\left[(R-G)^2+(R-B)(G-B)\right]^{\frac{1}{2}}} \\
H &= \begin{Bmatrix} \theta & if\ B<G \\ 2\pi-\theta & otherwise \end{Bmatrix} \\
S &= 1-\frac{3\cdot\min(R,G,B)}{R+G+B} \\
I &= \frac{1}{3}(R+G+B)
\end{aligned}
\tag{2.4}
$$

2.3.2 Image Segmentation

Image segmentation [28,63,143] is one of the most important steps leading to the analysis of processed image data-its main goal is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image. Image data ambiguity is one of the main segmentation problems, often accompanied by information noise. Segmentation methods can be divided into three groups according to the dominant features they employ: First is global knowledge about an image or its part; the knowledge is usually represented by a histogram of image features [188]. Edge-based segmentation form the second group [102], and region-based segmentations the third-many different characteristics may be used in edge detection or region growing [29], for example, brightness, texture, velocity field, etc. The second and the third groups solve a dual problem. Each region can be represented by its closed boundary, and each closed boundary describes a region. Because of the different natures of the various edge- and region-based algorithms, they may be expected to give somewhat different results and consequently different information.

Threshold-based Segmentation

Gray level thresholding [63] is the simplest segmentation process. Many objects or image regions are characterized by constant reflectivity or light absorption of their surfaces; a brightness constant or threshold can be determined to segment objects and background. Thresholding is computationally inexpensive and fast-it is the oldest segmentation method and is still widely used in simple applications [63].

A complete segmentation of an image R is a finite set of regions $R_1,\ldots,R_S$,

$$R = \bigcup_{i=1}^{S} R_i \qquad R_i \cap R_j = \varnothing \qquad i \neq j \tag{2.5}$$

Segmentation can result from thresholding in simple scenes. Thresholding is the transformation of an input image f to an output (segmented) binary image g as follows:

$$g(i,j) = 1 \quad \text{if } f(i,j) \geq T \tag{2.6}$$

$$0 \quad \text{if } f(i,j) < T$$

where T is the threshold, $g(i,j)=1$ for image elements of objects, and $g(i,j) = 0$ for image elements of the background (or vice versa). If objects do not touch each other, and if their gray levels are clearly distinct from background gray levels, thresholding is a suitable segmentation method.

Correct threshold selection is crucial for successful threshold segmentation; this selection can be determined interactively or it can be the result of some threshold detection method. Only under very unusual circumstances can thresholding be successful using a single threshold for the whole image (global thresholding) since even in very simple images there are likely to be gray level variations in objects and background; this variation may be due to non-uniform lighting, non-uniform input device parameters or a number of factors. Segmentation using variable thresholds [188] (also called adaptive thresholding), in which the threshold value varies over the image as a function of local image characteristics, can produce the solution in these cases.

Methods based on approximation of the histogram of an image using a weighted sum of two or more probability densities with normal distribution represent a different approach called optimal thresholding [38]. The threshold is set as the closest gray level corresponding to the minimum probability between the maxima of two or more normal distributions, which results in minimum error segmentation [153]. The difficulty with these methods is in estimating normal distribution parameters together with the uncertainty that the distribution may be considered normal. These difficulties may be overcome if an optimal threshold is sought that maximizes gray level variance between objects and background [96].

Edge-based Segmentation

Edge-based segmentation represents a large group of methods [104,102,182,11] based on information about edges in the image; it is one of the easiest segmentation approaches and still remains very important. Edge-based segmentations rely on edges found in an image by edge detecting operators-these edges mark image locations of discontinuities in gray level, color, texture, etc. The image resulting from edge detection cannot be used as a segmentation result, supplementary processing steps must follow to combine edges into edge chains that correspond better with borders in the image. The most common problems of edge-based segmentation, caused by image noise or unsuitable information in an image, are an edge presence in locations where there is no border, and no edge presence where a real border exists. Clearly both these cases have a negative influence on segmentation results.

Edge detectors are a collection of very important local image segmentation methods used to locate changes in the intensity function; edges are pixels where this function (brightness) changes abruptly. Calculus describes changes of continuous functions using derivatives; an image function depends on two variables-co-ordinates in the image plane-and so operators describing edges are expressed using partial derivatives. A change of the image function can be described by a gradient that points in the direction of the largest growth of the image function. An edge is a property attached to an individual pixel and is

calculated from the image function behavior in a neighborhood of that pixel. It is a vector variable with two components, magnitude and direction. The edge magnitude is the magnitude of the gradient, and the edge direction ϕ is rotated with respect to the gradient direction ψ by -90°. The gradient direction gives the direction of maximum growth of the function, e.g., from black [$f(i,j) = 0$] to white [$f(i,j) = 255$)].

The gradient magnitude $|grad\ g(x,y)|$ and gradient direction ψ are continuous image functions calculated as

$$|grad\ g(x,y)| = \sqrt{\left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}$$
$$\psi = \arg\left(\frac{\partial g}{\partial x}, \frac{\partial g}{\partial y}\right) \qquad (2.7)$$

where $\arg(x,y)$ is the angle (in radians) from the x-axis to the point (x,y).

Edge detection represents an extremely important step facilitating higher-level image analysis and therefore remains an area of active research, with new approaches continually being developed. Recent examples include edge detectors using fuzzy logic [104], neural networks [182], or wavelets [11,38]. It may be difficult to select the most appropriate edge detection strategy; a comparison of edge detection approaches and an assessment of their performance may be found in [44,146].

Region-based Segmentation

The aim of the segmentation methods described in the previous section was to find borders between regions; the methods discussed in this section construct regions directly. It is easy to construct regions from their borders, and it is easy to detect borders of existing regions. However, segmentations resulting from edge-based methods and region-growing methods are not usually exactly the same, and a combination of results may often be a good idea. Region growing techniques are generally better in noisy images, where borders are extremely difficult to detect. Homogeneity is an important property of regions and is used as the main segmentation criterion in region growing, whose basic

idea is to divide an image into zones of maximum homogeneity. The criteria for homogeneity can be based on gray level, color, texture, shape, model (using semantic information), etc. [29,64,69,134]. Properties chosen to describe regions influence the form, complexity, and amount of prior information in the specific region-growing segmentation method. Methods that specifically address region-growing segmentation of color images are reported in [57,144,161].

2.3.3 Mathematical Morphology

A powerful set of binary image processing operations developed from a set-theoretical approach [118] comes under the heading of *mathematical morphology* [35,68,71,75,118,162,164]. Although the basic operations are simple, they and their variants can be concatenated to produce much more complex effects [165]. The basic morphological operations are erosion and dilation. While commonly used on binary images (Figure 2-10), this approach can be extended to gray-scale images as well (Figure 2-11) [60]. Mathematical morphology is very often used in applications where shapes of objects are the main consideration for example, analysis of microscopic images (in biology, material science, geology, and criminology), industrial inspection, optical character recognition, and document analysis [70,180,181].

In the general case, morphological image processing operates by passing a *structuring element* over an image in an activity similar to convolution [116]. Like the convolution kernel, the structuring element can be of any size, and it can contain any complement of 1's and 0's. At each pixel position, a specified logical operation is performed between the structuring element and the underlying binary image. The binary result of that logical operation is stored in the output image at that pixel position. The effect created depends upon the size and content of the structuring element and upon the nature of the logical operation.

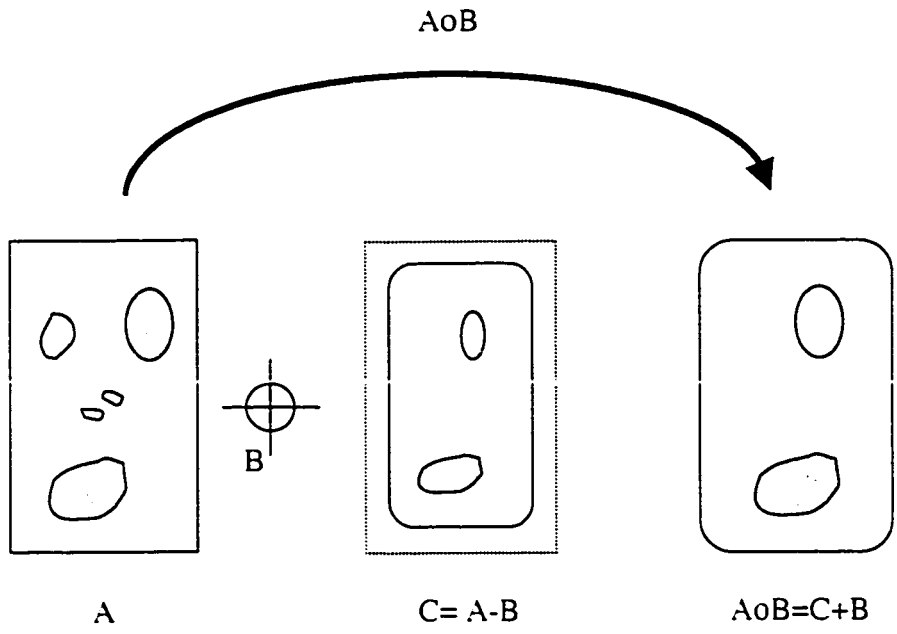


Figure 2-10 (a). Morphological opening operation: first erode the dark areas and then dilate.

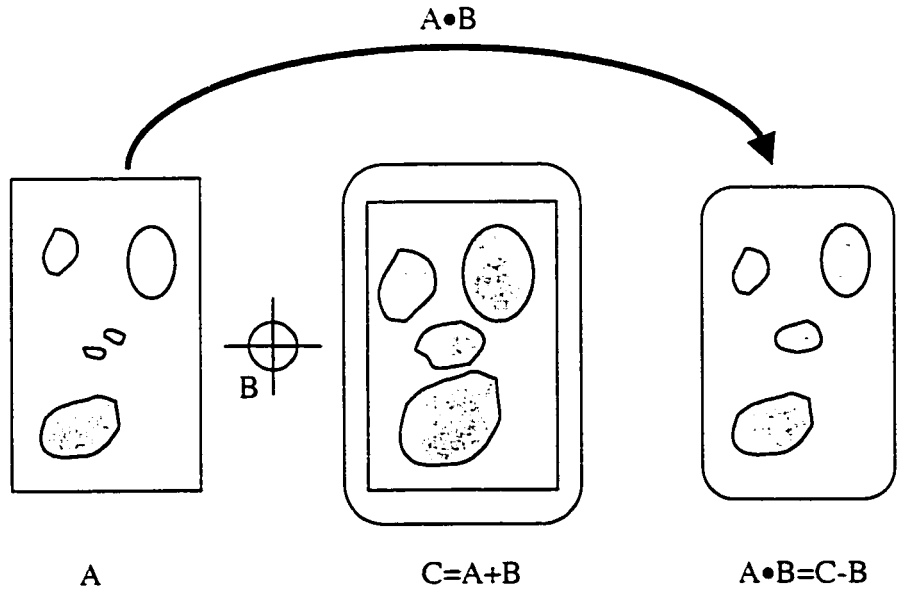


Figure 2-10 (b). Morphological closing operation: first dilate the dark area and then erode.

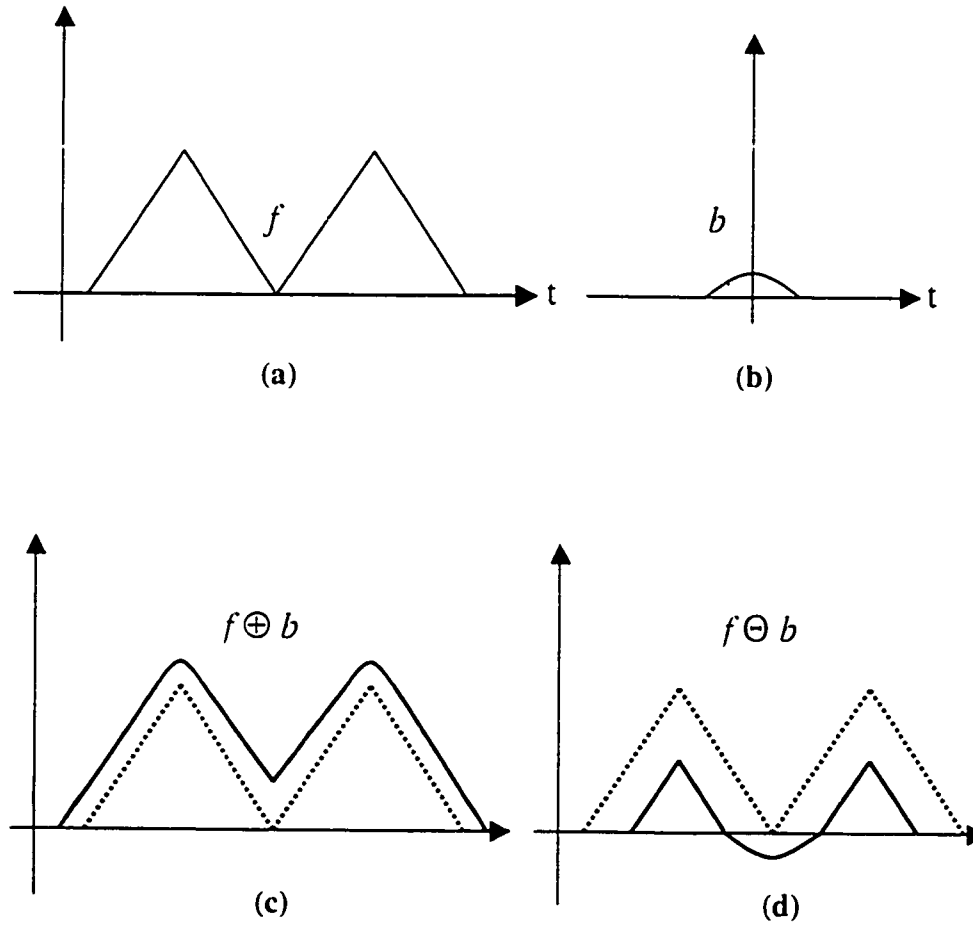


Figure 2-11. Illustration of gray-scale dilation and erosion by a semicircular structuring element. In Figure 2-11(c) the dilation is obtained by sliding structuring element *b* past function *f*. Figure 2-11(d) shows the erosion of the function in Figure 2-11(a) by the structuring element shown in Figure 2-11(b).

2.4 Feature Extraction

The previous section was devoted to image segmentation methods and showed how to construct homogenous regions of images and/or their boundaries. Recognition of image regions is an important step on the way to understanding image data, and requires an exact region description in a form suitable for a classifier. This description should generate a numeric feature vector, which characterizes properties (for example shape) of the region.

If a system to distinguish objects of different types is desired, then it is important to first decide which characteristics of the objects should be measured to produce descriptive parameters. The particular characteristics that are measured are called the features of the object, and the resulting parameter values comprise the feature vector for each object. Proper selection of the features is important, since only these will be used to identify the objects. There are few analytical means to guide the selection of features. Frequently, intuition guides the listing of potentially useful features. Feature-ordering techniques [28] compute the relative power of the various features. In practice, the feature selection process usually involves testing a set of intuitively reasonable features and reducing the set to an acceptable number of the best ones. Good features have four characteristics [28]:

- Discrimination: Features should take on significantly different values for objects belonging to different classes.
- Reliability: Features should take on similar values for all objects of the same class.
- Independence: The various features used should be uncorrelated with each other.
- Small Number: The complexity of a pattern recognition system increases rapidly with the dimensionality (number of features used) of the system. More importantly, the number of objects required to train the classifier and to measure its performance increases exponentially with the number of features [125]. In some cases, it may be impractical to acquire the amount of data required to train the classifier adequately. Finally, adding more features that are either noisy or highly correlated with existing features can actually degrade the performance of the classifier, particularly in view of the limited size of the training set [86,91].

In practice, the feature selection process usually involves testing a set of intuitively reasonable features and reducing the set to an acceptable number of the best ones. Many features can be used to describe an object. The most basic of all image features is the measure of image amplitude in terms of spectral value, luminance, tristimulus value, or other units [28]. Image transforms provide the frequency domain information in the data. Transform coefficient feature extraction has proved practical in several applications in which the transform domain features are used as inputs to a pattern recognition classification system [15,67,167]. The textural features of an object can often be used to discriminate between the surface finish of a smooth or coarsely textured object [67]. The gray level histogram of an image often contains sufficient information to allow analysis of the image content and, in particular, to discriminate between objects [67]. The most common object measurements made are those that describe shape.

2.4.1 Shape Representation and Description

Defining the shape of an object can prove to be very difficult. Shape is usually represented verbally or in figures, and people use terms such as elongation, rounded, with sharp edges, etc. The computer era has introduced the necessity to describe even very complicated shapes precisely, and while many practical shape description methods exist, there is no generally accepted methodology of shape description. Further, it is not known what makes a shape important. In general, shape descriptors are sorted according to whether they are based on object boundary information (contour-based, external description) or whether the information from object regions is used (region-based, internal description) [79]. This classification of shape description methods may be local or global and differ in sensitivity to translation, rotation, scaling, etc.

Contour-based shape representation and description

Chain codes are used to represent a boundary by a connected sequence of straight-line segments of specific length and direction [54,108,153]. Typically, this representation is based on the 4- or 8-connectivity of the segments. A chain code is very sensitive to noise,

and arbitrary changes in scale and rotation may cause problems if used for recognition. The smoothed version of the chain code (averaged directions along a specified path length) is less noise sensitive [108]. Boundary length is an elementary region property, which is simply derived from the chain code representation. Vertical and horizontal steps have unit length, and the length of diagonal steps in 8-connectivity is $\sqrt{2}$. It can be shown that the boundary is longer in 4-connectivity, where a diagonal step consists of two rectangular steps with a total length of 2. A closed boundary length (perimeter) can also be easily evaluated from run length [153] or quad tree representations [153].

Representation of curves using piecewise polynomial interpolation to obtain smooth curves is widely used in computer graphics. B-splines [14,153] are piecewise polynomial curves whose shape is closely related to their control polygon-a chain of vertices giving a polygonal representation of a curve. B-splines of the third order are most common because this is the lowest order that includes the change of curvature. Splines have very good representation properties and are easy to compute: they change their shape less then their control polygon, and they do not oscillate between sampling points as many other representations do. Many other methods and approaches can be used to describe two-dimensional curves and contours. The Hough transform has excellent shape representation abilities and is discussed in detail in [122]. Region-based shape description using statistical moments is covered in [137], where a technique of contour-based moments computation from region borders is also included. Further, it is necessary to mention the fractal approach to shape [115] that is gaining growing attention in image shape description [174]. Morphology [149,164] can also be used for shape description, typically in connection with region skeleton construction. A different approach is introduced in [112], where a geometrical correlation function represents two-dimensional continuous or discrete curves. This function is translation, rotation, and scale invariant and may be used to compute basic geometrical properties.

Region-based shape representation and description

A large group of shape description techniques [22,137] is represented by heuristic approaches that yield acceptable results in the description of object shapes. Region area, rectangularity, elongatedness, direction, compactness, etc., are examples of these methods. Region moment representations interpret a normalized gray scale image function as a probability density of a 2D random variable. Properties of this random variable can be described using statistical characteristics such as moments [137]. Other procedures based on region decomposition into smaller and simpler sub-regions must be applied to describe more complicated regions, then sub-regions can be described separately using heuristic approaches. Objects are represented by a planar graph with nodes representing sub-regions resulting from region decomposition, and region shape is then described by the graph properties [22].

2.4.2 Feature Analysis and Interpretation

In a pattern recognition problem, one is usually faced with the task of selecting which of the many available features should actually be measured and presented to the classifier. As mentioned before, a small set of reliable, independent, and discriminating features are required. In general, it is expected that the performance of the classifier degrades as features are eliminated, at least if they are useful features. In fact, eliminating noisy or highly correlated features can actually improve performance.

Feature selection is the process of reducing the number of features while maintaining or improving the classification accuracy. In the following discussion, the simple case of reducing a two-feature problem to a one-feature problem using a statistical approach is considered. In this approach three objective functions are used to measure the feature fitness. These functions are called intra-class variation (i.e. feature variance within the same class), inter-class variation (i.e. class seperation), and feature correlation. Suppose a training set is available that contains objects from M different classes. Let N_j be the

number of objects from class j. The two features obtained when the i^{th} object in class j is measured are x_{ij} *and* y_{ij}, computing the mean value of each feature for each class:

$$\hat{\mu}_{xj} = \frac{1}{N_j}\sum_{i=1}^{N_j} x_{ij} \quad \text{and} \tag{2.8}$$

$$\hat{\mu}_{yj} = \frac{1}{N_j}\sum_{i=1}^{N_j} y_{ij} \tag{2.9}$$

Ideally, the features should take on similar values for all objects within the same class. The estimated variance of the feature x within class j is

$$\hat{\sigma}^2_{xj} = \frac{1}{N_j}\sum_{i=1}^{N_j}(x_{ij} - \hat{\mu}_{xj})^2 \text{ and for feature } y \text{ it:} \tag{2.10}$$

$$\hat{\sigma}_{yj} = \frac{1}{N_j}\sum_{i=1}^{N_j}(y_{ij} - \hat{\mu}_{yj})^2 \tag{2.11}$$

A relevant measure of the ability of a feature to distinguish between two classes is the variance-normalized distance between class means. For feature x, this is given by:

$$\hat{D}_{xjk} = |\hat{\mu}_{xj} - \hat{\mu}_{xk}| \Big/ \sqrt{\hat{\sigma}_{xj} + \hat{\sigma}_{xk}} \tag{2.12}$$

where the two classes are j and k. Clearly, the superior feature is the one producing the widest class separation. The correlation of the feature x and y in class j can be estimated by:

$$\hat{\sigma}_{xyj} = \frac{\frac{1}{N_j}\sum_{i=1}^{N_j}\left(x_{ij} - \hat{\mu}_{xj}\right)\left(y_{ij} - \hat{\mu}_{yj}\right)}{\sigma_{xj}\sigma_{yj}} \tag{2.13}$$

This quality is bounded by −1 and +1. A value of zero indicates that the two features are uncorrelated, while a value near +1 implies a high degree of correlation. A value of −1 implies that each variable is proportional to the negative of the other. If the magnitude of the correlation is near 1, the two features might well be combined into one, or one of them might be discarded.

2.5 Pattern Recognition

Pattern recognition [47,55,132,160] is the scientific discipline whose goal is the classification of objects into a number of categories or classes. Depending on the application, these objects can be images, signal waveforms, or any type of measurements that need to be classified. Pattern recognition has a long history, but before the 1960s, it was mostly the output of theoretical research in the areas of statistics. As with everything else, the advent of computers increased the demand for practical applications of pattern recognition, which in turn set new demands for further theoretical developments. The theory of pattern recognition is thoroughly discussed in several references [47,55,132], and here only a brief introduction will be given.

Machine vision is an area in which pattern recognition is of importance. For example, in inspection, manufactured objects on a moving conveyor may pass the inspection station, where the camera stands, and it has to be ascertained whether there is a defect. Thus, images have to be analyzed, and a pattern recognition system has to classify the objects.

An object is a physical unit, usually represented in image analysis and computer vision by a region in a segmented image. The set of objects can be divided into disjoint subsets, which from the classification point of view, have some common features and are called classes. The definition of how the objects are divided into classes is ambiguous and depends on the classification goal. The classifier (similarly to a human) does not decide about the class from the object itself, rather, sensed object properties serve this purpose. For example, to distinguish steel from sandstone, their molecular structures need not have to be determined, although this would describe these materials well. Properties such as texture, specific weight, hardness, etc., are used instead. This sensed object is called the pattern, and the classifier does not actually recognize objects, but recognizes their patterns. The main pattern recognition steps are shown in Figure 2-12. As is apparent from the feedback arrows, these steps are not independent. On the contrary, they are interrelated and, depending on the results, one may go back to redesign earlier stages in order to improve the overall performance. Furthermore, there are some methods that

combine steps, that is, the feature selection and the classifier design stage, in a common optimization task. In the following sub-sections, we will discuss some of the methods commonly used for classification.

2.5.1 Statistical Classification

Statistical object description uses elementary descriptions called features, $x_1, x_2, \ldots, x_n$. In image analysis, the features result from object description as discussed earlier. The pattern (also referred to as pattern vector, or feature vector) $x=(x_1, x_2, \ldots, x_n)$ that describes an object is a vector of elementary descriptions, and the set of all possible patterns forms the pattern space X (also called feature space). If the elementary descriptions are appropriately chosen, similarity of objects in each class results in the proximity of their patterns in pattern space. The classes form clusters in the feature space, which can be separated by a discrimination curve (or hyper-surface in a multi-dimensional feature space) as shown in Figure 2-13.

If a discrimination hyper-surface exists which separates the feature space such that only objects from one class are in each separated region, the problem is called a recognition task with separable classes. If the discrimination hyper-surfaces are hyper-planes, it is called a linearly separable task. If the task has separable classes, each pattern will represent only objects from one class. The majority of recognition problems do not have separable classes, in which case the discrimination hyper-surfaces in the feature space can never separate the classes correctly and some objects will always be misclassified.

Classification Principles

A statistical classifier is a device with n inputs and 1 output. Each input is used to enter the information about one of n features $x_1, x_2, \ldots, x_n$ that are measured from an object to be classified. An R-class classifier will generate one of R symbols $w_1, w_2, \ldots, w_r$ as an output, and the user interprets this output as a decision about the class of the processed object. The generated symbols w_r are the class identifiers.

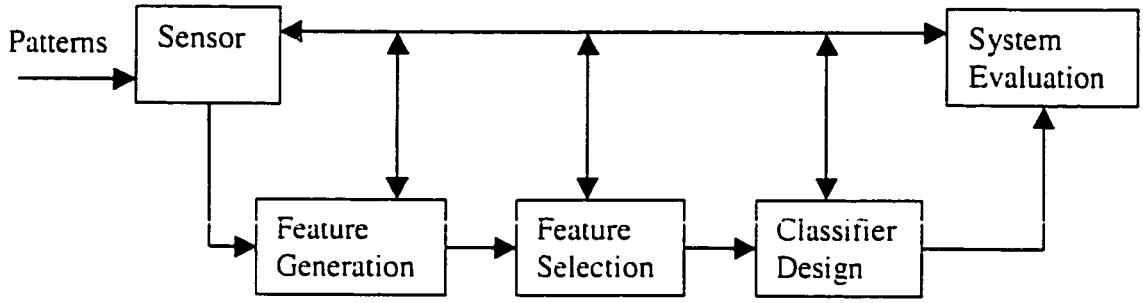


Figure 2-12. The basic stages involved in the design of a classification system

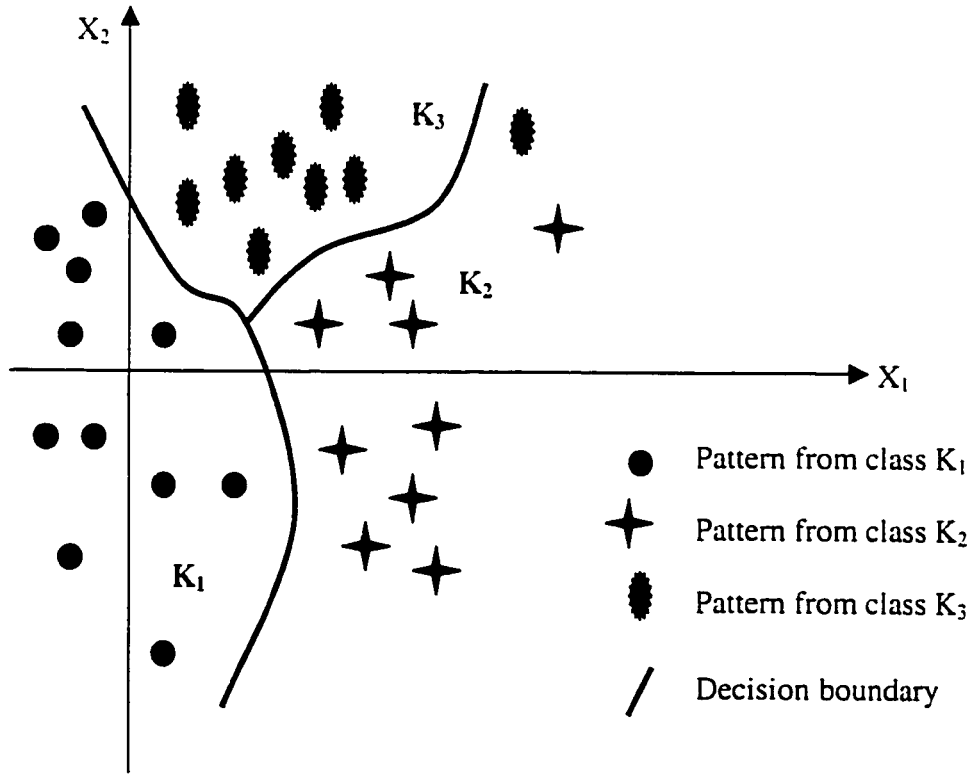


Figure 2-13. General discrimination functions

The function $d(x)=w_r$ describes relations between the classifier inputs and the output; the function d() is called the decision rule. The decision rule divides the feature space into R disjoint subsets K_r, $r=1,\ldots,R$, each of which includes all the feature representation vectors x of objects for which $d(x)=w_r$. The borders between subsets K_r, $r=1,\ldots,R$ form the discrimination hyper-surfaces mentioned earlier. The discrimination of discrimination hyper-surfaces (or definition of the decision rule) is the goal of classifier design. The descrimination hyper-surfaces can be defined by R scalar functions $g_1(x)$, $g_2(x),\ldots,$ $g_r(x)$ called discrimination functions. The design of discrimination functions must satisfy the following formula for all $x \in K_r$ and for all $s \in \{1,\ldots,R\}, s \neq r$:

$$g_r(x) \geq g_s(x) \tag{2.14}$$

Therefore, the discrimination hyper-surface between class regions K_r and K_s is defined by

$$g_r(x) - g_s(x) = 0 \tag{2.15}$$

The decision rule results from this definition. The object pattern x will be classified into the class whose discrimination function gives a maximum of all the discrimination functions:

$$d(x)= w_r \Leftrightarrow g_r(x)= \max_{s=1,\ldots,R} g_s(x) \tag{2.16}$$

Linear discrimination are the simplest and are widely used. Their general form is

$$g_r(x)= q_{r0} + q_{r1}x_1 + \ldots + q_{rn}x_n \tag{2.17}$$

for all $r=1,\ldots,R$. If all the discrimination functions of the classifier are linear, it is called a linear classifier.

Another possibility is to construct classifiers based on the minimum distance principle. The resulting classifier is just a special case of classifiers with discrimination functions, but they have computational advantages and may easily be implemented on digital computers. Assume that R points are defined in the feature space, $v_1, v_2, \ldots, v_R$ that represent exemplars (sample patterns) of classes $w_1, w_2, \ldots, w_R$. A minimum distance classifier classifies a pattern x into the class to whose exemplar it is closest.

$$d(x)= w_r \Leftrightarrow |v_r - x| = \min_{s=1,\ldots,R} |v_s - x| \tag{2.18}$$

Classifier Types

It is useful to distinguish among different types of classifiers based upon what is known about the underlying statistics and what must be estimated.

Parametric and Nonparametric Classifiers [47,160,176]. If the functional form of the conditional probability density functions (PDFs) is known, but some parameters of the density function (mean value, variances, etc.) are unknown, then the classifier is called parametric. Since the *a priori* probabilities are also parameters, they may be unknown. With parametric classifiers, the functional form of the conditional PDFs is assumed, on the basis of some fundamental knowledge about the objects themselves. Frequently, functional forms are assumed for mathematical expediency, as well as for more intrinsic reasons.

If the functional form of some or all of the conditional PDFs is unknown, the classifier is termed nonparametric. This means that all conditional PDFs must be estimated from training set data. To do so requires considerably more data than merely estimating a few parameters in a PDF of known functional form. Thus, nonparametric techniques are used when suitable parametric models are unavailable and large amounts of training data are within reach.

Supervised and Unsupervised Training [47,160,176]. The process of estimating the conditional PDFs or their parameters using object measurements is referred to as training the classifier. If the objects have been previously classified by some error-free process, the process is refereed to as supervised training. With unsupervised training, the conditional PDFs are estimated using samples whose class is unknown. The classes, and even the number thereof, must be determined by locating clusters of points in measurement space. This is called cluster analysis. Unsupervised training is normally used only when it is inconvenient or impossible to obtain a pre-classified training set or when the number and characteristics of the classes have not been otherwise determined.

2.5.2 Artificial Neural Networks

The immense capabilities of the human brain in processing information and making instantaneous decisions, even under very complex circumstances and under uncertain environments, have inspired researchers in studying and possibly mimicking the computational abilities of this wonder. What a human can achieve in very short time, for instance, in terms of pattern recognition and obstacle avoidance within an unknown environment, would have taken a computer very expensive resources (programmers, training experts, expensive hardware) and much longer time to get comparable results [13]. This is mainly due to the way humans process information. Researchers have shown indeed for many years, that brains make computations in a radically different manner as done by digital computers [7]. Unlike computers, which are programmed to solve problems using sequential algorithms, the brain makes use of a massive network of parallel and distributed computational elements called neurons [50]. The large number of connections linking these elements provides humans with the very powerful capability of learning. Motivated by this very efficient computational biological model, scientists have for the last few decades attempted to build computational systems, which can process information in a similar way as the brain does. Such systems are called artificial neural networks (ANNs) [72,73,130,197] or connectionist models. They are composed of a large number of highly interconnected processing elements analogous in functionality to biological neurons and are tied together with weighted connections corresponding to the brain synapses.

Learning and acquisition of knowledge

Learning is accomplished in general by developing algorithms that allow the system to learn for itself from a set of input/output training data. One major goal of learning algorithms is to combine the main features of a computing machine with those of human expertise to infer as many correct decisions as possible. An increasing number of systems are being designed today to have the very distinctive feature of learning. This is done by adjusting their parameters in response to unpredictable changes in their dynamics or their

operating environment without the need for an explicit knowledge of a system model or rules that guide its behavior. Once learning capability has been implemented within a given system, it continues acquiring knowledge and uses the stored information to draw right decisions for similar future situations. Two well-known approaches [168] for designing learning systems employ symbolic-learning and numerical-learning-based techniques. Although the focus here is on numerical learning, it is nevertheless interesting to preview as well the main features of symbolic learning.

Symbolic learning

Symbolic learning denotes the ability of a system to formulate and alter its knowledge base, composed mainly of facts and rules, from a set of well-structured feedback data related to the performance of the system. Systems with symbolic learning capabilities can rearrange their operating set of rules in response to changes in the system dynamics or the operating environment. This depends largely on the degree of supervision to which the system may be subjected. Several categories of symbolic learning have been proposed in the literature [5,151].

Numerical learning

In recent years, a large body of research on numerical learning has focused specifically on ANNs [130,168,197]. This is an area of research that has seen a renewed interest recently, after a marginally enthusiastic start in the 1960s and the early 1970s [155]. Numerical-learning-based algorithms represent a class of very useful learning tools used most often for the design of neural networks. They provide a given network with the capacity of adjusting its parameters in response to training signals. In some cases, learning is acquired by practice through training (supervised learning). In others the system is presented with a number of patterns and it is up the network, through well-defined guidelines, to group these patterns into categories (unsupervised learning). For instance, in the case of supervised learning, the system is exposed to a priori known set of input/output data. The learning mechanism is carried out through an optimization

process, during which, the system attempts to approximate the desired response vector with the output of the learning algorithm. This is done by solving an optimization problem involving m patterns of desired responses t^1, t^2, ..., t^m, and their corresponding inputs x^1, x^2, ..., x^m, and a set of *a priori* unknown optimization parameters called connection weights $W=[w_1, w_2, ..., w_p]$. The superscript m denotes the number of training patterns presented to the network and the subscript p represents the number of interconnection weights among all nodes of the network. Each element x of the training input vector is composed of l components, while each training output vector t is composed of q components. The optimization algorithm seeks the minimization of the sum of the error between the desired output vector t^k and the output vector o^k given by $h(w,x^k)$, corresponding to the actual output of the network in response to the excitation x^k. The mapping h represents here the input-output representation of the neural network. The optimization problem is now expressed as

$$\min_{w} \sum_{k=1}^{m} e(o^k, t^k)^2 = \min_{w} \sum_{k=1}^{m} e(h(w, x^k), t^k)^2 \tag{2.19}$$

The training algorithm keeps adjusting the weights according to well-defined learning rules. As more data become available, the learning system continues updating the weights to comply with the optimization criteria while adapting to new situations. Several techniques to solve this optimization problem have been proposed in the literature [90,93,103]. Some of the well-known approaches include genetic algorithms [62], annealing algorithms [12], and gradient-descent [17] based algorithms. The two major numerical learning algorithms most often used for the design of neural networks are the supervised and the unsupervised algorithms. Details on these algorithms and others along with an outline of their main features are provided in subsequent sections.

Features of artificial neural networks

As mentioned previously, an ANN is typically composed of a set of parallel and distributed processing units, called nodes or neurons. These are usually ordered into layers, appropriately interconnected by means of unidirectional (or bi-directional in some cases) weighted signal channels, called connections or synaptic weights. The internal

architecture of ANN provides powerful computational capabilities, allowing for the simultaneous exploration of different competing hypotheses. Massive parallelism and computationally intensive learning through examples in ANN make them suitable for application in nonlinear functional mapping [90], speech [130] and pattern recognition [130], categorization [13], data compression [7], and many other applications characterized by complex dynamics and possibly uncertain behavior [176]. Neural networks gather their knowledge through detection of patterns and relationships found in the data provided to them. Three important features generally characterize an artificial neural network: the network topology, the network transfer functions, and the network-learning algorithm.

Neural Network Topologies

The topology corresponds to the ordering and organizing of the nodes from the input layer to the output layer of the network. In fact, the way the nodes and the interconnections are arranged within the layers of a given ANN determines its topology. The choice for using a given topology is mainly dictated by the type of problem being considered. Some neural networks designers classify ANN according to how the nodes are organized and hence on how data is processed through the network. The two well-known ANN topologies are the feedforward [72] and the recurrent [72] architectures.

A network with feedforward architecture has its nodes hierarchically arranged in layers starting with the input layer and ending with the output layer. In between, a number of internal layers, also called hidden layers, provide most of the network computational power. The nodes in each layer are connected to the next layer through unidirectional paths starting from one layer (source) and ending at the subsequent layer (sink). This means that the outputs of a given layer feed the nodes of the following layer in a forward path as shown in Figure 2-14. Because of their structure, such networks are called feed-forward networks. They are also occasionally called open loop networks given the absence of feedback flow of information in their structure. The feedforward topology has been very popular due to its association with a quite powerful and relatively robust

learning algorithm called the backpropagation-learning algorithm [155]. The multilayer perceptron network and the radial basis function network [187] are among the well-known networks using the feedforward topology.

Unlike feedforward networks, recurrent networks allow for feedback connections among their nodes, as illustrated in Figure 2-15. They are structured in such a way as to permit storage of information in their output nodes through dynamic states, hence providing the network with some sort of 'memory'. While feedforward networks map input into output and are static in the sense that the output of a given pattern of inputs is independent of the previous state of the network, recurrent networks map states into states and as such are very useful for modeling and identification of dynamic systems. Several well known neural networks have been designed based on the recurrent topology, including the Kohonen network [98], the Hopfield network [80] and the adaptive resonance theory networks [27].

Neural network activation functions

The basic elements of the computational engine of a neural network are the neurons, which take the weighted sum of their inputs from other nodes and apply to them a mapping, called the activation function, before delivering the output to the next neuron(s). The output O_k of a typical neuron (k) having (l) inputs is given as:

$$O_k = f\left(\sum_l w_{lk} x_l + \theta_k\right) \tag{2.20}$$

where f is the node's activation function, $x_1, x_2, \ldots, x_l$ are the node's inputs, $w_{1k}, w_{2k}, \ldots, w_{lk}$ are the connections weights, and θ_k is the node's threshold. The processing activity within a given layer is done simultaneously hence providing the neural network with the powerful capability of parallel computing. The bias effect (threshold value) is intended to inhibit the activity of some nodes. As neural networks may vary in terms of their structure as described previously they may, as well, vary in terms of their activation function. Depending on the problem in hand and on the location of the node within a given layer, the activation functions can take different forms [168]: sigmoid mapping, signum function, or linear correspondence.

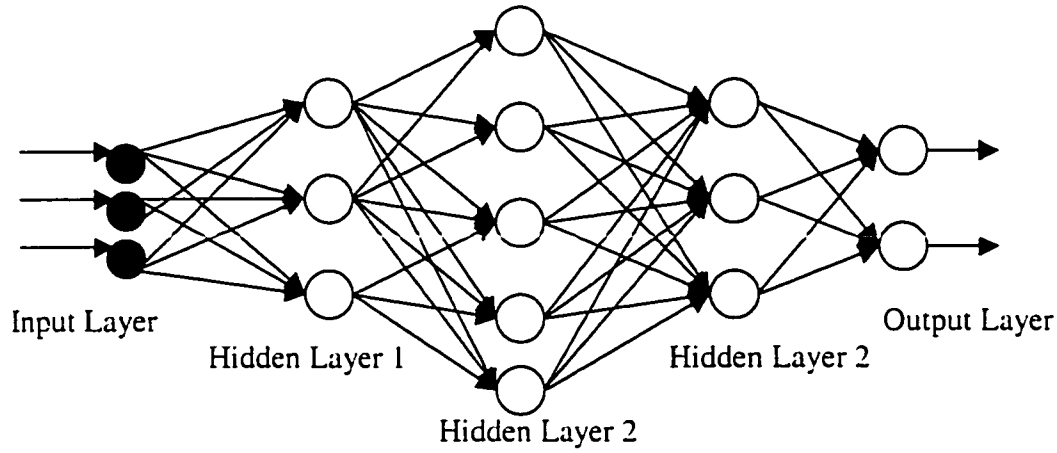


Figure 2-14. Typical structure of feed-forward neural network

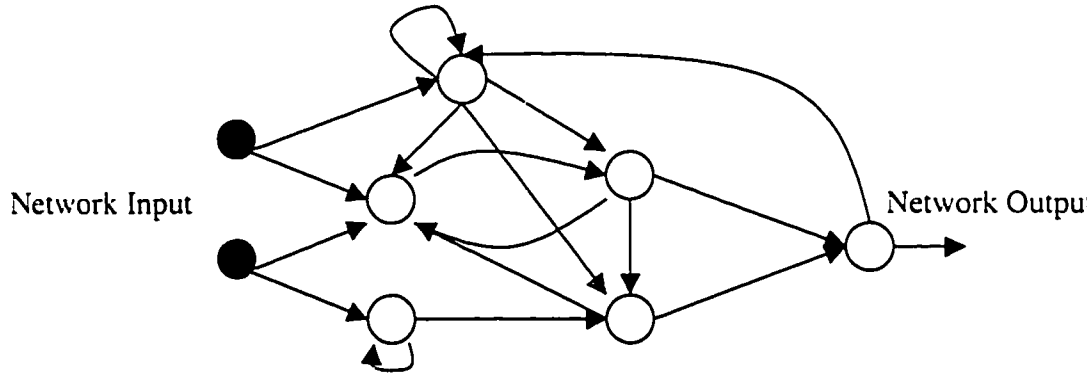


Figure 2-15. Typical structure of recurrent neural network

$$sigmoid(x) = \frac{1}{1+\exp(-x)}$$

$$signum(x) = \begin{bmatrix} 1 & if\ x>0 \\ 0 & if\ x=0 \\ -1 & if\ x<0 \end{bmatrix}$$

$$step(x) = \begin{bmatrix} 1 & if\ x>0 \\ 0 & otherwise \end{bmatrix} \qquad (2.21)$$

$$linear(x) = x$$

Network learning algorithms

Learning algorithms are used to update the weighting parameters at the interconnection level of the neurons during the training process of the network. While some designers have classified neural networks according to topologies or architectures, others have classified them according to the learning algorithm used by the network. The three most often used learning mechanisms [73] are the supervised [155], the unsupervised (or self organized) [98], and the reinforced [168]. A brief overview of each one of these weight-updating mechanisms is provided in the following sub-sections.

Supervised learning. The main feature of the supervised (or active) learning mechanism is the training by examples. This means that an external teacher provides the network with a set of input stimuli for which the output is known *a priori*. During the training process, the output results are continuously compared with the desired data. An appropriate learning rule (such as the gradient descent rule) uses the error between the actual output and the target data to adjust the connection weights so as to obtain after a number of iterations the closest match between the target output and the actual output. Supervised learning is particularly useful for feedforward networks. A number of supervised learning algorithms have been suggested in the literature [7,80,90,110]. The backpropagation algorithm first developed by Webos in 1974 [187], which is also based

on the gradient descent optimization technique and the least mean square algorithm [5] are among the most commonly used supervised learning rules.

Unsupervised learning. Unlike supervised learning, unsupervised or self-organized learning [124] doesn't involve an external teacher and relies instead upon local information and internal control. The training data and input patterns are presented to the system, and through predefined guidelines, the system discovers emergent collective properties and organizes the data into clusters or categories. Because of the way the network adjusts its connection weights in response to the presented input data, unsupervised learning systems have been known as open-loop adaptation learning schemes [98]. In a simplified manner, an unsupervised learning scheme operates as follows. A set of training data is presented to the system at the input layer level. The network connection weights are then adjusted through some sort of competition among the nodes of the output layer, in which the successful candidate will be the node with the highest value. In the process, the algorithm strengthens the connection between the incoming pattern at the input layer and the node output corresponding to the winning candidate. In addition to the strengthening of the connections between the input layer and the winning output node, the unsupervised learning scheme may be as well used for adjusting the weights of the connections leading to the neighboring nodes at the output layer. This is controlled by what is called the neighborliness parameter, and it has the major property of making groups of output nodes behaving as single entities with particular features.

Reinforcement learning. Reinforcement learning (RL) [124], also known as graded learning, has been receiving an increased interest given its many attractive learning features that, according to many, model in a fair way the adjusting behavior of humans when interacting with a given physical environment. This is another type of learning mechanism by means of which the network connections are modified according to feedback information provided to the network by its environment. This information simply instructs the system on whether or not a correct response has been obtained. In the case of a correct response, the corresponding connections leading to that output are

strengthened and weakened otherwise. This type of learning strategy based on the reward/penalty process has several similarities with the biological learning system. Unlike supervised learning, RL doesn't get information on what the output should be when the network is presented with a given input pattern. RL does also differ from unsupervised learning (UL) in that UL doesn't provide the network with information on whether the output is right or otherwise, but rather operates on the premises of finding pattern regularity among the exemplars presented to the system. Given the nature of this learning scheme, random search strategies have to be used to attain the correct output every time the system is presented with an excitation from its environment. This explorational aspect of learning ultimately leads to an improved capability of the system to deliver the expected output every time the system is presented with an input pattern. Among the strategies used to implement a reinforced learning algorithm are the reinforcement comparison [124], the adaptive heuristic critic [124], Q learning [124], and the policy only scheme [124].

Hybrid models. The term hybrid models have been used indiscriminately in the neural network literature [152]. While some authors use it to describe models in which both unsupervised and supervised neural network concepts are used in different parts of a connectionist model, others use it to describe models where neural networks are combined with rule-based, classic statistical, and other types of modeling approaches.

2.5.3 Neuro-fuzzy Systems

Fuzzy systems [196,135] and neural networks [72,124,194] have attracted the growing interest of researchers in various scientific and engineering areas. The number and variety of applications of fuzzy logic and neural networks have been increasing, ranging from consumer products and industrial process control to information systems, and decision analysis. Fuzzy logic is based on the way the brain deals with inexact information, while neural networks are modeled after the physical architecture of the brain. Although the fundamental inspirations for these two fields are quite different, there are a number of

parallels that point out their similarities. Fuzzy systems and neural networks are both numerical model-free estimators and dynamical systems [110]. They share the common ability to improve the intelligence of systems working in an uncertain, imprecise, and noisy environment. Both fuzzy systems and neural networks have been shown to have the capability of modeling complex nonlinear processes to arbitrary degrees of accuracy.

Although fuzzy systems and neural networks are formally similar, there are also significant differences between them. Fuzzy systems are structured numerical estimators. They start from highly formalized insights about the structure of categories found in the real world and then articulate fuzzy IF-THEN rules as a kind of expert knowledge. Fuzzy systems combine fuzzy sets with fuzzy rules to produce overall complex nonlinear behavior. Neural networks, on the other hand, are trainable dynamical systems whose learning, noise tolerance, and generalization abilities grow out of their connectionist structures, their dynamics, and their distributed data representation. Neural networks have a large number of highly interconnected processing elements (nodes) which demonstrate the ability to learn and generalize from training patterns or data; these simple processing elements also collectively produce complex nonlinear behavior.

In light of their similarities and differences, fuzzy systems and neural networks are suitable for solving many of the same problems and achieving some degree of machine intelligence. Their differences have prompted a recent surge of interest [194] in merging or combining them into a functional system to overcome their individual weaknesses. A neuro-fuzzy system is a combination of neural networks and fuzzy systems in such a way that neural networks, or neural network learning algorithms, are used to determine parameters of fuzzy systems [194]. This means that the main intension of a neuro-fuzzy approach is to create or improve a fuzzy system automatically by means of neural network methods. An even more important aspect is that the system should always be interpretable in terms of fuzzy if-then rules, because it is based on a fuzzy system reflecting vague knowledge. On the other hand, a fuzzy neural network is a neural network that uses fuzzy methods to learn faster or to perform better [110]. In this case, the improvement of a neural network is the main intention.

2.6 Performance Models and Optimization Methods

In recent years, the emphasis has shifted in the field of infrastructure engineering from the design and construction of new facilities to the maintenance and rehabilitation (M&R) of existing infrastructure facilities [105]. Information on current and future conditions is essential for M&R decision-making. Data on current conditions are obtained from facility inspection. These data are also used to develop facility deterioration models, which are used to predict future facility conditions. Both current condition and predicted future condition are used to select M&R activity. The success of an infrastructure management system is dependent largely, on its ability to predict future conditions accurately. The modeling of infrastructure deterioration and its relation to M&R treatment strategies have been investigated by many previous researchers [88,131,156]. Determination of the optimal investments in operating infrastructure network preservation programs is concerned with M&R priority programming prioritization, which may be carried out through infrastructure life-cycle analysis. There are many priority programming methods available in infrastructure management, ranging from simple subjective ranking to optimization [48,51,114].

In reality, it is difficult to obtain an optimization as many uncertain factors are involved in the mathematical programming model. Moreover, the optimization formula developed from one specific region is not likely to be fitted into another region. Consequently, a number of near optimization approaches, which are close to the true optimal and relatively easy to develop, have been widely utilized in infrastructure management system. Having a reliable infrastructure performance model is essential for asset managers to secure infrastructure life-cycle structural design, and to estimate the costs and benefits of the projects associated with M&R treatments. Infrastructure treatment actions should be planned based on current needs and projected future needs with considerations of budget constraints. An infrastructure performance model is a critical component of the entire management process and it is, therefore, the foundation of an integrated management system. In general, an infrastructure management system aims to provide the tools necessary to predict infrastructure future conditions so that the optimal

repair strategies and actions can be determined. Over the last 30 years, a considerable amount of effort has been made to the prediction of infrastructure deterioration in order to meet the requirements of each specific infrastructure management system. In this section the existing models for predicting infrastructure deterioration and optimization methods are briefly reviewed.

2.6.1 Infrastructure Performance Prediction Models

The typical infrastructure maintenance decision-making environment involves multiple objectives and uncertainty, and is dynamic. One of the most commonly used infrastructure models is a Markov decision process (MDP) [78]. MDP models have been applied to numerous sequential decision-making situation involving uncertainty and multiple objectives, including applications related to infrastructure problems [113]. Such probabilistic models also have the advantage of naturally lending themselves to computationally tractable discrete dynamic programming optimization [39]. The core of these probabilistic models consists of infrastructure transition probabilities. A review of the literature revealed one common method for the estimation of Markov transition probabilities using condition rating data of a set of facilities observed over time. The method consists of the following three steps.

In the first step, the facilities are classified into groups where each group consists of facilities having similar attributes. Each group of facilities generates a set of condition rating, Y, and age, t. The purpose of this grouping is to capture the fact that transition probabilities are a function of explanatory variables. The approach, therefore, is based on segmenting the population of facilities so that transition matrices are estimated for each group.

In the second step, for each group a deterioration model with the condition rating, Y, as the dependent variable and age, t, as the independent variable is estimated. The following is the mathematical expression.

$$Y_n = \beta_1 + \beta_2 t_n + \varepsilon_n \tag{2.22}$$

where n = index for facility; Y_n = condition rating of infrastructure facility n; β_1, β_2 = parameters to be estimated; t_n = age of facility n; and ε_n = random error term. The parameters β_1 and β_2 in model (2.22) are estimated by linear regression.

In the third step, a transition probabilities matrix is estimated for each group by minimizing a measure of distance between the expected value of the facility condition rating as predicted by model (2.22) and the theoretical expected value derived from the structure of the Markov chain. The theoretical expected value is a function of the transition probabilities to be estimated. The objective function used in practice is the sum of the absolute (or squared) differences between the two expected values [25]. The transition matrix, P, has the following general structure.

$$P = \begin{bmatrix} p_{kk} & p_{k(k-1)} & \bullet & \bullet & \Lambda & \bullet & p_{k1} \\ 0 & p_{(k-1)(k-1)} & p_{(k-1)(k-2)} & \bullet & \Lambda & \bullet & p_{(k-1)1} \\ \bullet & \bullet & \bullet & \bullet & \Lambda & \bullet & \bullet \\ \bullet & \bullet & \bullet & \bullet & \Lambda & \bullet & \bullet \\ 0 & 0 & 0 & 0 & \Lambda & p_{22} & p_{21} \\ 0 & 0 & 0 & 0 & \Lambda & 0 & p_{11} \end{bmatrix} \tag{2.23}$$

where k = highest condition state; and 1 = lowest condition state.

As is evident from the zero entries in the transition matrix, it is assumed that a facility can either stay in its current state or deteriorate to some lower state. This implies the absence of rehabilitation activities.

The mathematical representation of the minimization of the distance between the two expected values is given in the following.

$$\begin{aligned} &\min W = \sum_{t=\tau}^{\tau+\Delta T-1} \left| Y_t - E(t,P) \right| \\ &\textit{subject to } 0 \le p_{ij} \le 1;\ i,j = 1,2,\ldots,k \\ &\sum_{j=1}^{k} p_{ij} = 1;\ \ i = 1,2,\ldots,k \end{aligned} \tag{2.24}$$

where τ = earliest age observed in the group under consideration; ΔT = number of years in the group of facilities under consideration; $\hat{Y}_t$ = average condition rating for facilities in the group of age t, predicted by the deterioration model; k = number of states; P = transition probability matrix whose elements are to be estimated; and $E(t, P)$ = theoretical expected value of condition rating at age t as a function of the Markov transition probabilities. The first constraint ensures that the probabilities are bounded by 0 and 1, and the second constraint ensures that the elements of each row of the transition matrix sum to one.

2.6.2 Priority Programming and Optimization Methodologies

Priority programming [51] is a systematic process through which maintenance and rehabilitation actions needed by an infrastructure network are ranked in a certain order. It is based on certain criteria and priority judgments, such as available funds, degree of needs and urgency, costs and benefits. Therefore, a comprehensive analysis has to be conducted by considering all of the criteria and factors in the selection of the best alternative from a number of possible treatment strategies.

In recent years, a Markov process-based probabilistic model has been applied in many processes or subsystems of infrastructure management, such as dynamic programming of pavement maintenance incorporated with pavement deterioration modeling [107], pavement network budget planning [48], and cost-effectiveness analysis in financial planning of bridge management [166]. In the following sub-sections, several methods of priority programming for infrastructure management are briefly reviewed, including simple ranking, prioritization and optimization.

Subjective and Parameters Based Ranking Methods

Among many prioritization methods, the ranking method [81] is the simplest way of establishing priorities for infrastructure improvement needs. In this method, projects are ranked based on subjective judgment or criteria that are determined by the infrastructure agency's policy. The programming is based on either engineering judgment or on measured parameters, such as infrastructure condition, life-cycle cost, and benefit/cost ratio. The infrastructure is repaired or treated in rank order until the amount of money available for maintenance and rehabilitation is used up.

Priority Programming With Maximum Benefits or Minimum Costs

Costs and benefits are two major economic evaluation factors in network priority programming and project level design. There are generally three forms in priority programming [51]: benefit maximization, cost minimization, and benefits over costs ratio maximization. Benefit maximization or cost minimization with linear programming seems to have been the most popular approaches. With multi-year programming of projects, the effects of a large number of strategic treatment options to be applied in each year under the constraints of budgets and required performance standards are considered. Lytton [114] has investigated the multi-year prioritization analysis and mathematical optimization analyses such as dynamic programming. He concluded that the two approaches could achieve similar solutions. This is because the algorithms go through a similar sequence of operations to determine the projects that provide the greatest benefit for the same amount of money spent.

Prioritization techniques require a comprehensive performance model for the prediction of infrastructure deterioration and for the measurement of the benefit or effectiveness of alternative projects or treatments. The prioritization methodology is based on maximizing cost-effectiveness ratio from the selected M&R projects within a limited budget. The higher the weighted optimal benefit/cost ratio of the section is, the higher the priority of

that section will be for repair. A detailed description of this methodology is given by Feighan et. al. [51].

Optimization Methods

Optimization [45] is a branch of mathematics concerned with finding the most effective (optimum) solutions to complex problems in accordance with established objectives and constraints. The difference between optimization and the prioritization or ranking techniques described in the previous sections is that, in an optimization analysis, the functions of priority programming, program formulation and project scheduling are integrated into one operation which gives the optimum schedule of projects. Two important considerations that are not included in a prioritization analysis are a) the evaluation of inter-project trade-off in selecting strategies, and b) the selection of treatment strategies that strictly adheres to budget constraints. Consequently, mathematical optimization techniques have been recently employed by a number of highway agencies for programming investment priorities for pavement improvements. The use of mathematical optimization models is perhaps the most sophisticated for multi-year prioritization analysis, which include linear, non-linear, integer and dynamic programming methods [45,101]. It should be pointed that a reliable prioritization programming method is dependent, to a large degree, on the accuracy of the predicted infrastructure performance. The reason is that in the process of prioritization economic analysis of each treatment strategy is conducted based on the predicted future infrastructure deterioration versus time.

Chapter 3

3. Pipe Image Segmentation

In this chapter a simple, robust and efficient image segmentation algorithm for the automated analysis of scanned underground pipe images is presented. The algorithm consists of image pre-processing followed by a sequence of morphological operations to accurately segment pipe cracks, holes, joints, laterals, and collapsed surfaces, a crucial step in the classification of defects in underground pipes.

Section 3.1 presents an introduction and the necessary background of segmentation problem. Section 3.2 discusses the image pre-processing step, followed by morphological segmentation of underground pipe images in Section 3.3. Experimental results are discussed in Section 3.4.

3.1 Introduction

We have acquired a data set consisting of thousands of images of underground pipes from 15 major cities in North America. This data set has been used to explore basic characteristics of underground pipe images. Analyses of images have shown that there are two important characteristics that complicate the segmentation of pipe images: firstly

the presence of a complicated background pattern due to earlier runoff, patches of repair work, corroded areas, debris, non-uniformities in illumination, and flaws in the image acquisition process; secondly the three main objects of interest - cracks, joints, and laterals - are all dark features that cannot be distinguished by intensity criteria alone.

The goal of our research is to develop an automated method which, given a pipe image, classifies each pixel in the image into one of five classes: background, crack, hole, joint, and lateral. In principle, after the image has been segmented into its classes, each class could be separated further into extents of distress (e.g., minor crack, major crack, multiple crack, etc.). In general, this image segmentation problem is difficult to automate because the differences between classes such as joints and cracks, although obvious to a human, can be very difficult to encode mathematically at the pixel level.

A large number of segmentation algorithms have been proposed in the literature [26,31,63,158] and discussed in Chapter 2. However, the literature on segmentation of defects in concrete structures is very limited. Maser's algorithm [117] recommends a histogram thresholding approach, however it is not clear how the value of the threshold is originally determined. More recently, Chen et al. [30] applied a segmentation method, introduced by Kittler et al. [97], to pavement images, although the effectiveness of the method is unclear. An approach to the recognition of segmented pavement distress images is studied by Mohajeri and Manning [127], using directional filters to classify the objects. An entropy-based approach [3], which finds a bilevel threshold to maximize entropy criteria, did not improve pavement-surface images. The cluster classification process, which assigns a particular object to one of many groups by comparing typical features from each group, reported a significant amount of error [4].

In general, segmentation techniques take one of two possible approaches [74]: edge detection and thresholding. An edge is defined as the boundary between two regions with relatively distinct gray-scale characteristics, thus edge-detection techniques attempt to segment objects by outlining their boundaries. Thresholding, on the other hand, seeks to distinguish objects on the basis of their absolute intensities, for example separating a

darker object from a lighter background, in which case a good way of segmenting might be to determine a threshold T, such all pixels with an intensity above T are classified as being part of the background.

The literature on segmentation based on gray-level intensity is inapplicable in our context since cracks, holes, joints, and laterals all appear as comparably dark objects on a lighter background, as shown in Figure 3-1. Rather, it is the *geometry*, rather than the intensity, which distinguishes these objects. Mathematical morphology [35,71,75,118,162] provides an approach to the segmenting of digital images that is based on *shape*. Appropriately used, morphological operations tend to simplify images, preserving their essential shape characteristics and eliminating irrelevancies. Theoretical background of morphology may be found in Chapter 2.

In this chapter, we propose a simple, robust and efficient morphological segmentation algorithm to distinguish cracks, holes, collapse surface, pipe joints, and pipe laterals, based on geometric criteria. The algorithm consists of gray-scale conversion followed by a sequence of gray-scale morphological operations.

3.2 Image Pre-processing

There are several hindrances to identifying the features in underground pipe images. In the case of cracks, the principal difficulty is that cracks are set against a highly patterned background and thus discriminating edges are often not present. A different problem arises when attempting to identify the crack boundary where there is low contrast between the inside of the crack surface and the surrounding area. Pipe background surface is primarily determined by the color of the pipe material. As such, it is usually found in a restricted range of intensities. Typically the gray-scale histogram information is used to enhance the contrast between the object and the background, as discussed in the background chapter. Here it is shown that pattern classification techniques applied to color images can also be used to enhance the contrast between the background surface and the objects present in the pipe image.

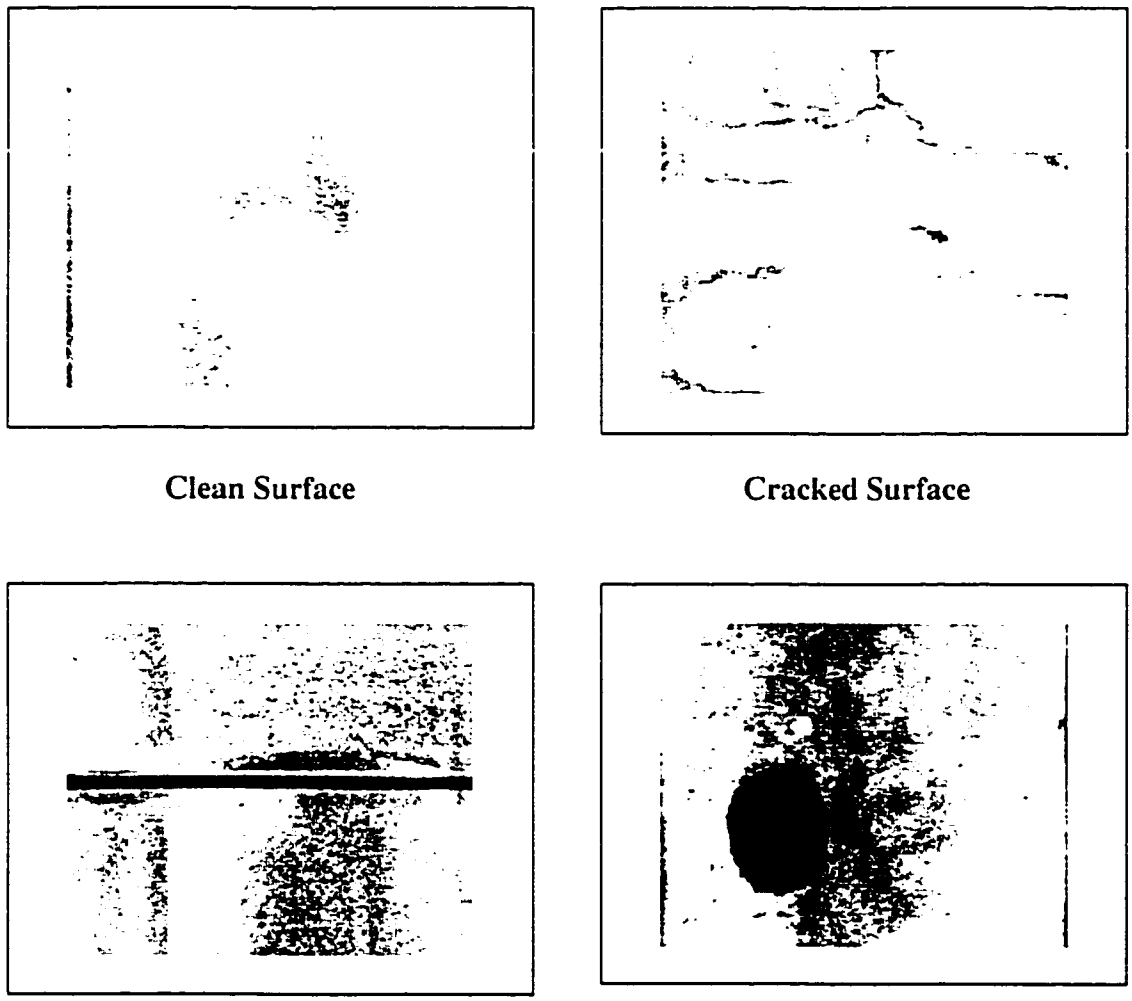


Figure 3-1. Typical images of underground concrete pipe showing different objects

3.2.1 Bayesian Classification

Identification of the boundary between the objects and their surround is formulated as a pattern classification problem. Specifically, it is desired to classify pixels as to whether it is more likely that they came from the objects or the neighboring background regions. In a Bayesian framework, a color pixel $x_c = (r,g,b)^T$, can be classified as a crack if its *a posteriori* probability $P(Crack|x)$ is greater than the corresponding *a posteriori* probability for the surrounding pipe background $P(Back|x)$. If the class-conditional probability densities $p(x|Crack)$ and $p(x|Back)$ are known, or can be learned from training images, then Bayes' Rule can be used to compute the corresponding *a posteriori* probabilities.

Standard parametric or non-parametric techniques can be used to learn the underlying class-conditional densities $p(x|Crack)$ and $p(x|Back)$. However, one must bear in mind that the *a posteriori* probabilities $P(Crack|x)$ and $P(Back|x)$ are evaluated for each pixel, along each search line, at each time step. Thus, in order for this approach to be usable in practical (real-time) systems, a premium is placed on the on-line processing time required to discriminate between the classes. Towards this end, Fisher's linear discriminant [47] is used to enhance the contrast between the objects and the pipe background.

3.2.2 Fisher's Linear Discriminant

Since we intend to apply morphological operators, we require a grey-scale image, in which each pixel x_g is a scalar $x_g = w^T x_c$, where w is some linear projection. In the case of a two class discrimination problem, such as distinguishing between cracks and pipe background, Fisher's linear discriminant [47] can be used to determine the axis, w, onto which vector color data can be projected which preserves as much of the discriminating capability of the color information as possible. The resulting 'Fisher linear discriminant' maximizes the separability of the two classes. Crack images representative of those likely

to be encountered during scanning of underground pipes can be used to learn the Fisher discriminant axis using the following algorithm.

1. Calculate mean color in class k=1,2

$$m_k = \frac{1}{n_k} \sum_{x \in \chi_k} x \tag{3.1}$$

2. Determine the within class scatter matrices, k=1,2

$$S_k = \sum_{x \in \chi_k} (x - m_k)(x - m_k)^T \tag{3.2}$$

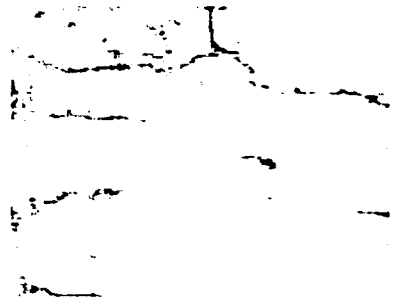
3. Find the Fisher discriminant vector

$$w = S_w^{-1}(m_1 - m_2) \qquad where\ S_w = S_1 + S_2 \tag{3.3}$$

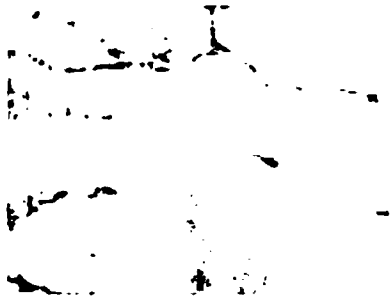
Pipe images representative of those likely to be encountered during object recognition and classification can be used to learn the Fisher discriminant axis. Figure 3-2 shows that Fisher's discriminant analysis can be used to enhance the contrast between the pipe background and cracks. In the gray-scale image (Figure 3-2(b)) there is little contrast between the background and cracks. The HIS and YIQ color models can also be used to provide additional contrast as shown in Figures 3-2(c) and 3-2(d), respectively; although the pipe image shows high contrast, the crack boundary is blurred and the image is noisy. The projection onto the Fisher axis (Figure 3-2(e)) enhances the contrast and enables better extraction of the crack features.

3.3 Morphological Segmentation

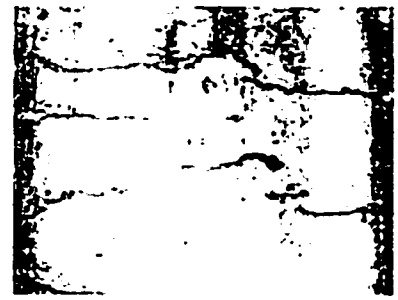
Mathematical morphology [35,68,71,75,118,162] is a widely used methodology for image analysis. Recently it has been used in image analysis as an approach for smoothing, image segmentation, edge detection, thinning, shape analysis and image coding. Based on a formal mathematical framework, mathematical morphology is a fast, robust method that analyzes the geometry of an image directly in the spatial domain. In this section, we present a morphological approach for segmenting underground pipe images, which includes the process of characterizing the object sizes in a pipe image, thresholding the image into a binary image, and finally classifying the segmented image.



(a) Original Color Image

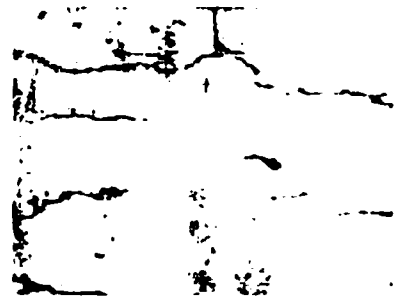


(b) Gray-scale Image



(c) HIS Color Model Image

(d) YIQ Color Model Image



(e) Fisher Discriminant Image

Figure 3-2. Fisher's discriminant analysis can be used to enhance the contrast between the pipe background and cracks. In gray-scale image (b) there is little contrast between the background and cracks. The HIS and YIQ color models can also be used to provide additional contrast (c) and (d), respectively; although the pipe image shows high contrast, the crack boundary is blurred and the image is noisy. Projection onto a Fisher axis (e) enhances the contrast and enables better extraction of the crack features.

3.3.1 Mathematical Morphology

Morphology operates on image regions (e.g., the light and dark portions of an image), where the regions can be reshaped (i.e., morphed) in various ways under the control of a structuring element. The structuring element can be thought of as a parameter to the morphological operation. The most fundamental operations are morphological *dilation* and *erosion*. Based on these, two compound operations, *opening* and *closing*, can be defined. We first define these in the context of binary images, then for the more complicated case of gray-scale images.

Consider a binary image I consisting of pixels $p(x,y)$:

$$I=\{p(x,y)\} \qquad p(x,y) \in \{0,1\} \tag{3.4}$$

We define sets A and B to represent the image and the selected structuring element, respectively:

$$A=\{(x,y) \mid p(x,y) = 1\} \tag{3.5}$$

$$B=\{(x,y) \mid (x,y) \textit{ in structuring element}\} \tag{3.6}$$

Then the *dilation* of $A \oplus B$ of A by structuring element B is defined as

$$A \oplus B = \{a + b \mid \textit{for all } a \in A \textit{ and } b \in B\} \tag{3.7}$$

That is, $A \oplus B$ is the union of all pixels in A surrounded by the shape of B. Similarly the *erosion* $A \ominus B$ is defined as

$$A \ominus B = \{p \mid b + p \in A \textit{ for every } b \in B\} \tag{3.8}$$

That is, all pixels within a "distance" B from the edge of A are removed.

Next, two compound operations can be defined, *opening* $A \circ B$ and *closing* $A \bullet B$:

$$A \circ B = (A \ominus B) \oplus B \tag{3.9}$$

$$A \bullet B = (A \oplus B) \ominus B \tag{3.10}$$

Intuitively, the opening operation, which we use in our classification, leaves unchanged all parts of A, except for those features and regions smaller than the structuring element, which are removed.

Any method which operates exclusively on binary images is somewhat limiting, since most images are color or gray-scale. A gray-scale image

$$I=\{p(x,y)\} \qquad p(x,y) \in \Re \tag{3.11}$$

can be interpreted as a three dimensional surface, with the pixel intensity or shade interpreted as a height. If we define

$$A=\{(x,y,z) \mid z \leq p(x,y)\} \tag{3.12}$$

and have a structuring volume

$$B=\{(x,y,z)\} \tag{3.13}$$

then we can interpret two-dimensional gray-scale morphological operations on image I as three-dimensional binary morphology on A, B.

Written directly in terms of a gray-scale image $f(x,y)$ and structuring function $b(x,y)$, grey-scale *erosion* and *dilation* can be defined as

$$(f \oplus b)(x,y) = \max_{(i,j)}\{f(x-i,y-j)+b(i,j)\} \tag{3.14}$$

$$(f \ominus b)(x,y) = \min_{(i,j)}\{f(x+i,y+j)-b(i,j)\} \tag{3.15}$$

Gray-scale opening and closing are defined as before, in equations (3.9) and (3.10).

3.3.2 Morphological Segmentation and Classification

In underground pipe image segmentation, the following classes are of general interest: the pipe joints (horizontal dark straight lines), pipe laterals (circular dark objects), surface cracks (irregularly shaped thin dark lines), and the pipe background (anywhere from a smooth to a highly patterned surface). The goal of our research is to segment pipe joints, laterals, and cracks based on the *geometric* differences between them, specifically based on morphological techniques.

In this section, we employ the morphological opening operation, using circular and horizontal structuring elements of various sizes, for the classification of objects in underground pipe images. A morphological opening with a circular structuring element (S_C) of radius r(mm), as shown in Figure 3-3, is equivalent to removing a strip of width r(mm) along the entire object boundary (erosion) and then, onto whatever portions remain of the object, adding a strip of width r(mm) back onto the object (dilation). A similar opening operation with a horizontal structuring element (S_H) of varying length l(mm) and fixed width w=3(mm), as shown in Figure 3-4, can be performed. In general, the effect of an opening is to remove small features, whereas large features shrink and then expand back to their original size. The choice of these two elements (circular and rectangular) is designed to approximately mimic the geometry of the laterals and joints to be extracted.

The key idea, then, is that we can isolate objects of a given 'size' by performing a series of opening operations, based on structuring elements of varying size. The 'size' of any object can then be defined mathematically as the largest structuring element (measured here in terms of radius r or length l) that can be inscribed in the object. Note that, aside from the general shape of the structuring element, we do not make any specific assumption regarding the shape of the object being measured, therefore this definition of size is quite general and will prove effective in measuring sizes of cracks, irregular laterals etc., which otherwise resist specific characterization. Our proposed segmentation algorithm consists of a sequence of processing steps, illustrated in Figure 3-5.

Morphological opening and thresholding

We performed a morphological opening operation on the underground pipe image with increasing sizes of the circular and horizontal structuring elements. Clearly as the size of the structuring element is increased features of increasing size are removed by the morphological opening. For example, a structuring element of intermediate size will preserve laterals and a collapsed pipe, but will remove cracks and small holes.

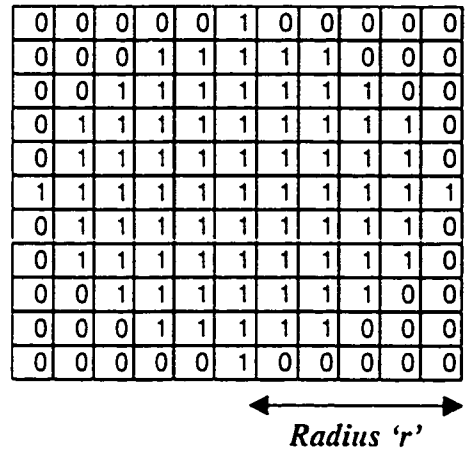


Figure 3-3. Circular structuring element, S_C

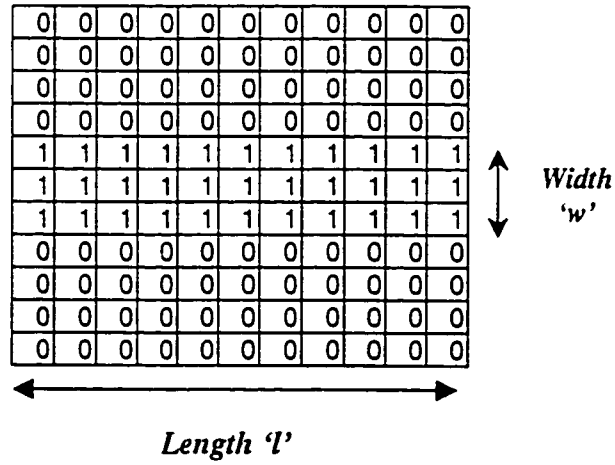


Figure 3-4. Horizontal structuring element, S_H

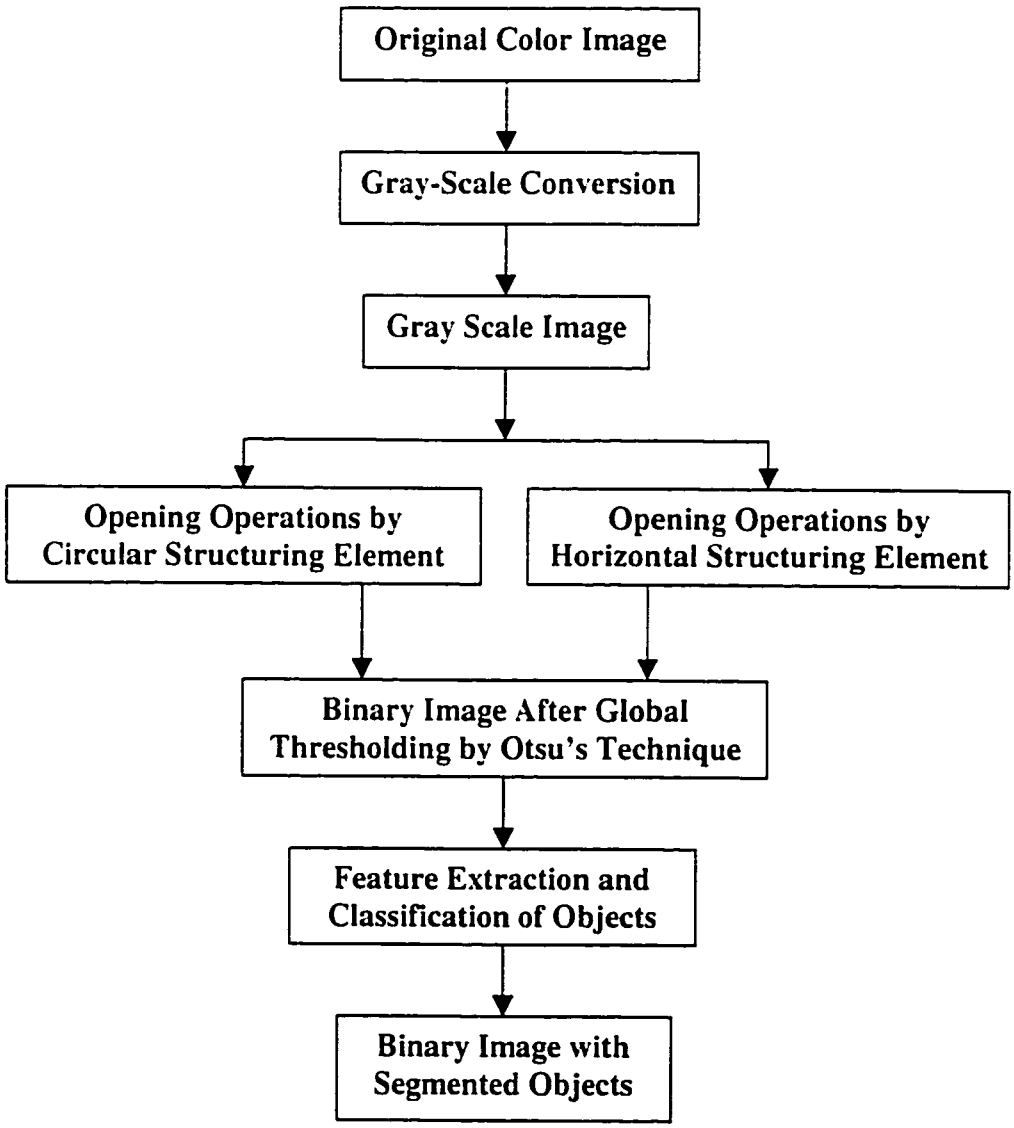


Figure 3-5. Overview of the proposed morphological segmentation approach

Figure 3-6 (b) shows two examples of the results of gray-scale opening. Although some of the original features in 3-6(a) are still clearly present, the output of the morphology is confusing and not easily interpreted. What we really want is an additional processing step, a thresholding function $t()$, classifying each pixel as:

$$t(p(x,y)) = \begin{cases} 0 & \textit{Pixel is geometrically consistent with structuring element} \\ 1 & \textit{Pixel is not consistent} \end{cases} \qquad (3.16)$$

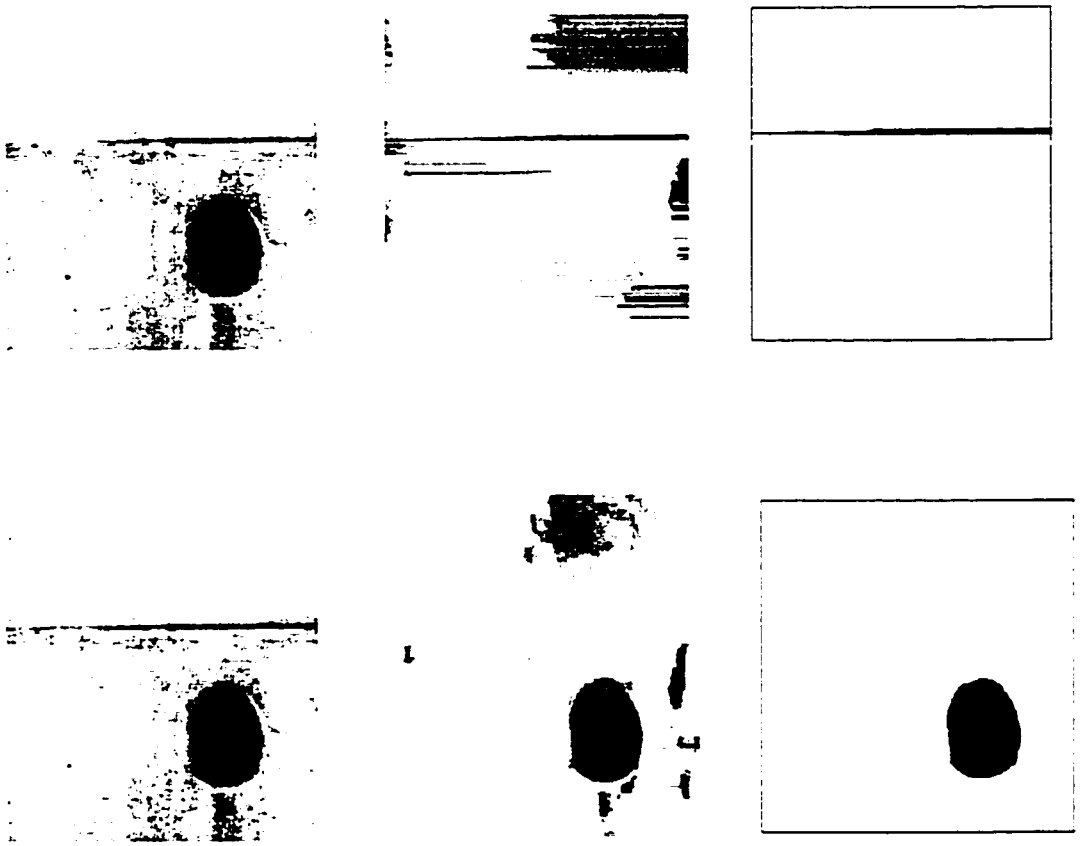
That is, the set of all 'dark' (zero valued) pixels will identify the object(s) in the image which are compatible (i.e., bigger than) the selected structuring element. Ideally, we would like a single global threshold T such that

$$t(p(x,y)) = \begin{cases} 0 & p(x,y) \le T \\ 1 & p(x,y) \ge T \end{cases} \qquad (3.17)$$

Unfortunately, it is difficult in general to find a single threshold that is best for an arbitrary gray-scale image. Many approaches have been proposed to find an optimal threshold level for certain image cases [23,109,133,138]. We propose to use Otsu's method [133] because it is non-parametric, unsupervised, and automatic. A discriminant criterion is computed for each possible threshold T; the optimal threshold is that gray-level where this measure is maximized. The results of Otsu's method are illustrated in Figure 3-6(c): the segmented joint and lateral stand out very clearly.

With a methodology in place for understanding the results of a given morphology, we can now study the choice of structuring elements that will be most effective in classifying each pipe object. Figures 3-7 and 3-8 plot the average area of objects in each class (crack, hole, joint, etc.) based on circular and horizontal structuring elements, respectively. That is, if we let $|t(I)|$ represent the number of dark pixels in I after binary thresholding, then Figures 3-7 and 3-8 actually plot the normalized areas

$$a_L(r) = \frac{|t(I \circ S_C(r))|}{|t(I_L)|} \qquad (3.18)$$



(a) Original Image (b) Gray-Scale Opening (c) Thresholded Image

Figure 3-6. This figure illustrates joint/lateral discrimination using different structuring element: a horizontal element (top) of length 285 mm, consistent with the geometry of a perfect joint, as opposed to a circular element (bottom) of radius 57 mm, tuned to the shape of a perfect lateral.

$$a_J(l) = \frac{|t(I \circ S_H(l))|}{|t(I_J)|} \tag{3.19}$$

where I_L, I_J are idealized, prototype images of the perfect lateral and joint. Note that all of the curves are monotonically decreasing,

$$|I \circ S_C(r_1)| \geq |I \circ S_C(r_2)| \quad for\, all\, r_1 \leq r_2 \tag{3.20}$$

since a larger structuring element cannot leave more pixels in place than a smaller element.

Although the plots in Figures 3-7 and 3-8 are interesting and intuitive in order to accurately isolate and classify different objects in an image we have to take into account the *variations* in the area of each class. That is, holes, laterals, etc. all come in a range of sizes, and this range *must* be taken into account in selecting the appropriate structuring element to serve as a classifier.

We can compute or assess the ability of any structuring element to discriminate between any two classes (e.g., crack and hole) by examining the degree to which the two classes are separated relative to their standard deviations:

$$D_{i,j}(r) = \frac{|\mu_i(r) - \mu_j(r)|^2}{\sigma_i^2(r) + \sigma_j^2(r)} \tag{3.21}$$

$$\mu_i(r) = \langle a_L(r) \rangle_i \tag{3.22}$$

$$\sigma_i^2(r) = \langle a_L(r)^2 \rangle_i - \langle a_L(r) \rangle_i^2 \tag{3.23}$$

where $\langle\ \rangle_i$ represents an average taken over images of class i. A parallel definition exists for discriminant $D_{i,j}(l)$ based on a horizontal structuring element. The value of r for which $D_{i,j}(r)$ is maximized represents the optimal feature by which to discriminate between classes i and j on the basis of the area (i.e. number of pixels) remaining after a morphological opening by element $S_C(r)$. By plotting $D_{i,j}(r)$ and $D_{i,j}(l)$ for different classes i, j we can deduce the set of features to be extracted for classification. Figures 3-9 and 3-10 plot $D_{L,i}(r)$ and $D_{J,i}(l)$ respectively, indicating peaks to identify these features.

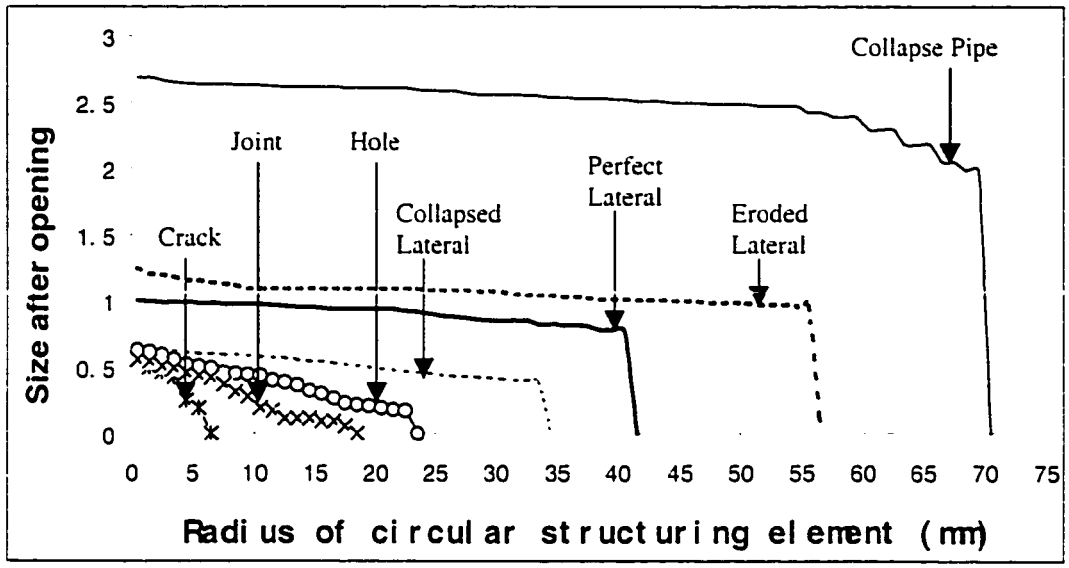


Figure 3-7. Morphological analysis based on a circular structuring element: the average area of each class is plotted as a function of the structuring element diameter; area is normalized to that of an ideal lateral. Clearly as the diameter is increased, classes with thin, elongated geometries (e.g. cracks, joints) are quickly eliminated.

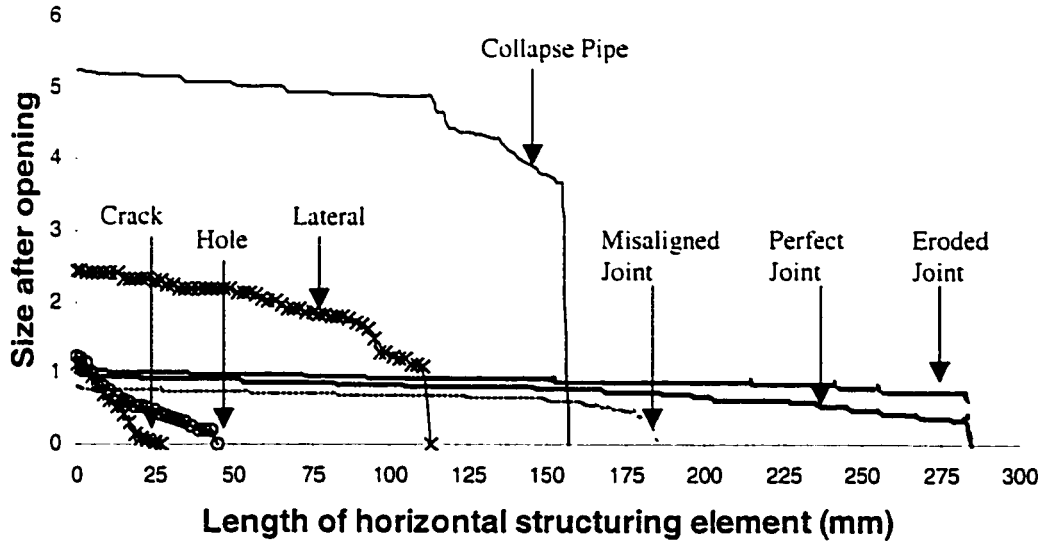


Figure 3-8. Morphological analysis based on a horizontal structuring element: the average area of each class is plotted as a function of the structuring element length; area is normalized to that of an ideal joint. Clearly as the length is increased, classes with small geometries (e.g., cracks, holes) are quickly eliminated.

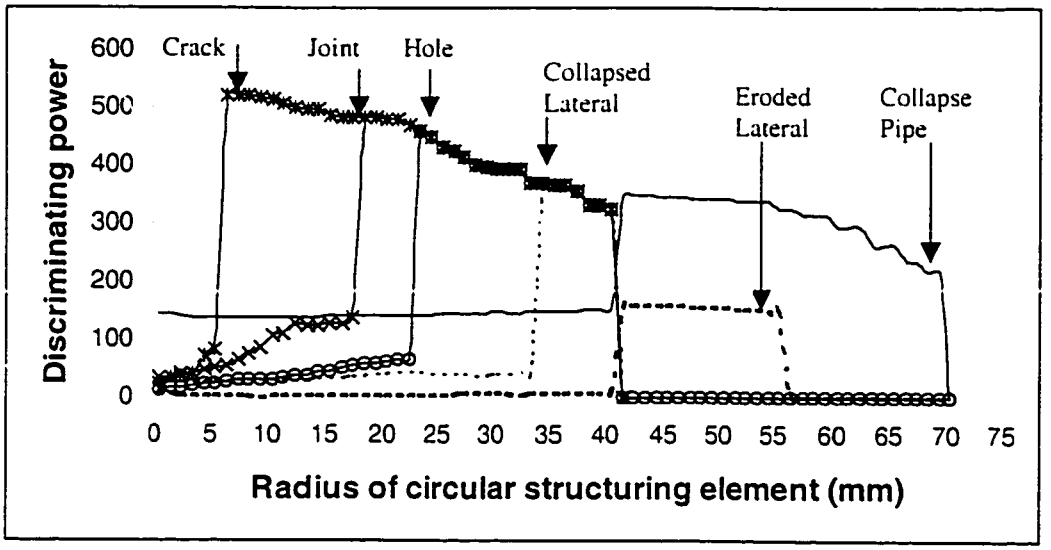


Figure 3-9. Lateral discrimination: we can distinguish between laterals and other classes by opening with a circular structuring element, as in Figure 7. We can plot the ability to discriminate '*D*' between an ideal lateral and any other class as the difference in response (normalized to standard error) to the structuring element.

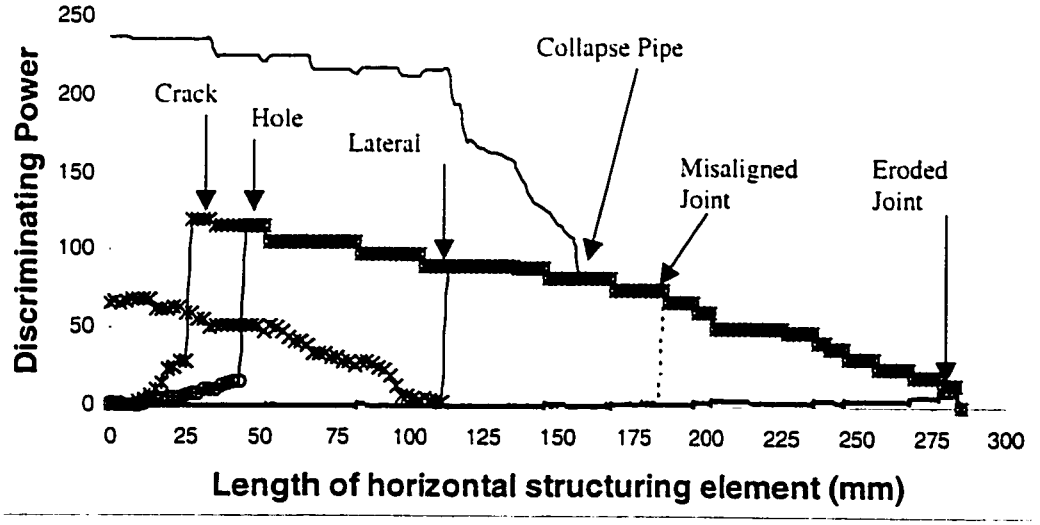


Figure 3-10. Joint discrimination: we can distinguish between joints and other classes by opening with a horizontal structuring element, as in Figure 8. We can plot the ability to discriminate '*D*' between an ideal joint and any other class as the difference in response (normalized to standard error) to the structuring element.

Feature extraction and classification

The sizes of structuring elements for the classification of objects in underground pipe images can be determined from the discriminant method described in the previous section. The optimal sizes of the structuring elements used for classification using circular and horizontal shaped elements are tabulated in Tables 3-1 and 3-2, respectively.

For example, if an image is opened with $S_C(2)$ – the circular structuring element of radius 2(mm) – then small objects (e.g., random background patterning) are removed. By repeating this process for different sizes of structuring elements $S_C(7)$, $S_C(23)$, $S_C(57)$, we can group objects by size, that is, into their respective classes.

Specifically, we propose to keep as our features

$$a_L(r) = \left| t\left(I \circ S_C(r)\right) \right| \qquad r \in \{2, 7, 23, 57\} \tag{3.24}$$

where the features are selected to discriminate between successive class pairs clean-pipe, cracks, holes-joints, laterals, and pipe-collapse. A further set of four features is chosen based on rectangular structuring elements:

$$a_J(l) = \left| t\left(I \circ S_H(l)\right) \right| \qquad l \in \{2, 47, 121, 155\} \tag{3.25}$$

selected to discriminate between successive pairs of clean-pipe, cracks-holes, laterals, joints, and pipe-collapse.

The classifier is then made up of pairwise discriminants, such as

$$a_L(2) = \begin{cases} \leq \tau_{0.1}(2) & Clean - pipe \\ < \tau_{0.1}(2) & Class\ of\ Crack\ or\ larger \end{cases} \tag{3.26}$$

Table 3-1. Optimal sizes of the circular structuring element selected for classification of various objects in the underground pipe images

Radius of Circular Structuring Element	Classified Class
2 (mm)	Clean-Pipe
7 (mm)	Crack
23 (mm)	Hole/Joint
57 (mm)	Laterals
71 (mm)	Collapse-Pipe

Table 3-2. Optimal sizes of the horizontal structuring element selected for classification of various objects in the underground pipe images

Length of Horizontal Structuring Element	Classified Class
2 (mm)	Clean
47 (mm)	Crack/Hole
121 (mm)	Laterals
155 (mm)	Pipe-Collapse
285 (mm)	Joints

Table 3-3. Optimal threshold selected for classification of various objects in the underground pipe images by circular structuring element

Threshold Values for Classification by Circular Structuring Element	Classifed Class
$A_L(2) \leq 150$	Clean-Pipe
$A_L(2) > 150$	Crack/Hole/Joint/Lateral/Collapse-Pipe
$A_L(2) \leq 1000$	Crack
$A_L(2) > 1000$	Hole/Joint/Lateral/Collapse-Pipe
$A_L(2) \leq 1700$	Hole/Joint
$A_L(2) > 1700$	Lateral/Collapse-Pipe
$A_L(2) \leq 3150$	Lateral
$A_L(2) > 3150$	Collapse-Pipe

Table 3-4. Optimal threshold selected for classification of various objects in the underground pipe images by horizontal structuring element

Threshold Values for Classification by Horizontal Structuring Element	Classifed Class
$A_L(2) \leq 150$	Clean-Pipe
$A_L(2) > 150$	Crack/Hole/Joint/Lateral/Collapse-Pipe
$A_L(2) \leq 1700$	Crack/Hole
$A_L(2) > 1700$	Joint/Lateral/Collapse-Pipe
$A_L(2) \leq 1250$	Lateral
$A_L(2) > 1250$	Joint/Collapse-Pipe
$A_L(2) \leq 1150$	Collapse-Pipe
$A_L(2) > 1150$	Joint

where the threshold $\tau_{0,1}(2)$ is based on our criterion in equation (3.21) at a radius of 2 (mm), discriminating between class 0 (Clean) and 1 (Crack). That is, given that we have chosen to maximize $D_{i,j}(r)$ – the separation of the class means normalized to the standard deviations – the optimum threshold in discriminating classes i and j is the weighted mean

$$\tau_{i,j}(r) = \frac{\sigma_i(r)\mu_i(r) + \sigma_j(r)\mu_i(r)}{\sigma_i(r) + \sigma_j(r)} \tag{3.27}$$

Thus

$$\tau_{0,1}(r) = \frac{\sigma_{crack}(r)\,\mu_{clean}(r) + \sigma_{clean}(r)\,\mu_{crack}(r)}{\sigma_{crack}(r) + \sigma_{clean}(r)} = 150 \tag{3.28}$$

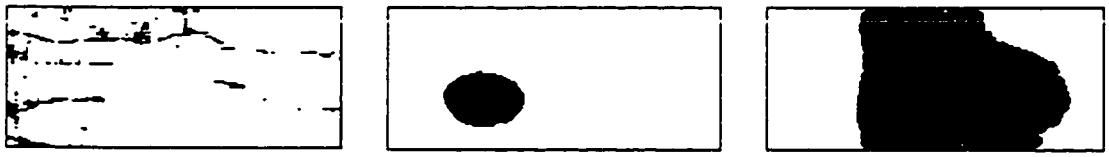
The remaining thresholds are listed as part of classifier summary in Tables 3-3 and 3-4.

4. Experimental Results

We have applied the proposed approach to more than 500 underground pipe images. Based on the experimental results, we conclude that the proposed method can segment and classify pipe images effectively and accurately. Figures 3-11 and 3-12 illustrate the proposed approach applied on sample of images by circular and horizontal structuring elements, receptively.

The proposed morphological segmentation and classification algorithm will work very well for underground pipe images containing one class of object only in a given frame. In the real world problem, underground pipe images may contain cluttered objects as shown in Figures 3-13(a) and 3-14(a). For segmentation and classification of such images, we suggest a slightly different approach. The new approach is based on taking the difference of image after each morphological opening and thresholding operations. Figures 3-13(c) and 3-14(c) illustrates the procedure of taking difference of image after successive opening operations to segment various objects present in the image. Once the objects are segmented, then feature extraction and classification can be done.

(a) Original Gray-scale images of underground pipe



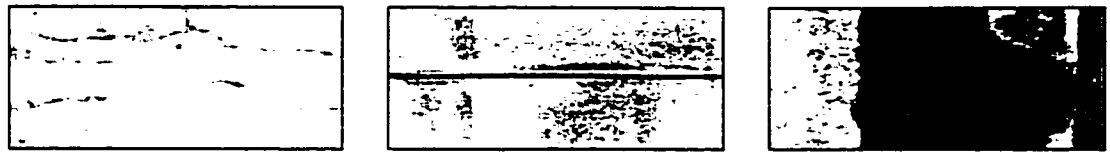
(b) After morphological opening by circular structuring element of radius 2(mm)

(c) After morphological opening by circular structuring element of radius 7(mm)

(d) After morphological opening by circular structuring element of radius 57(mm)

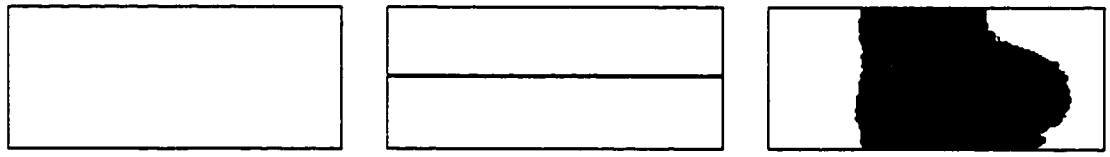
(e) After morphological opening by circular structuring element of radius 71(mm)

Figure 3-11. Classification results by using circular structuring element: the original images are opened by structuring element of different sizes as outlined in Table 1, and finally binary images are obtained by global thresholding technique.

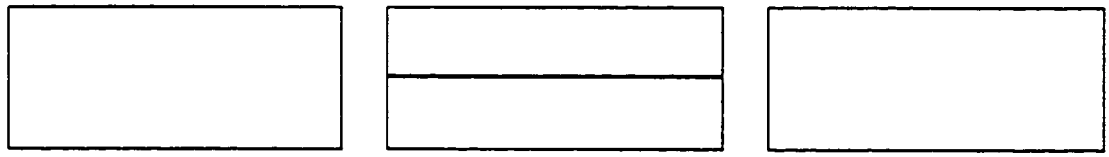


(a) Original Gray-scale images of underground pipe

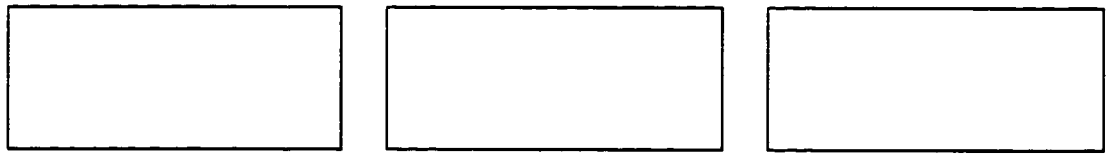
(b) After morphological opening by horizontal structuring element of length 2(mm)



(c) After morphological opening by horizontal structuring element of length 47(mm)

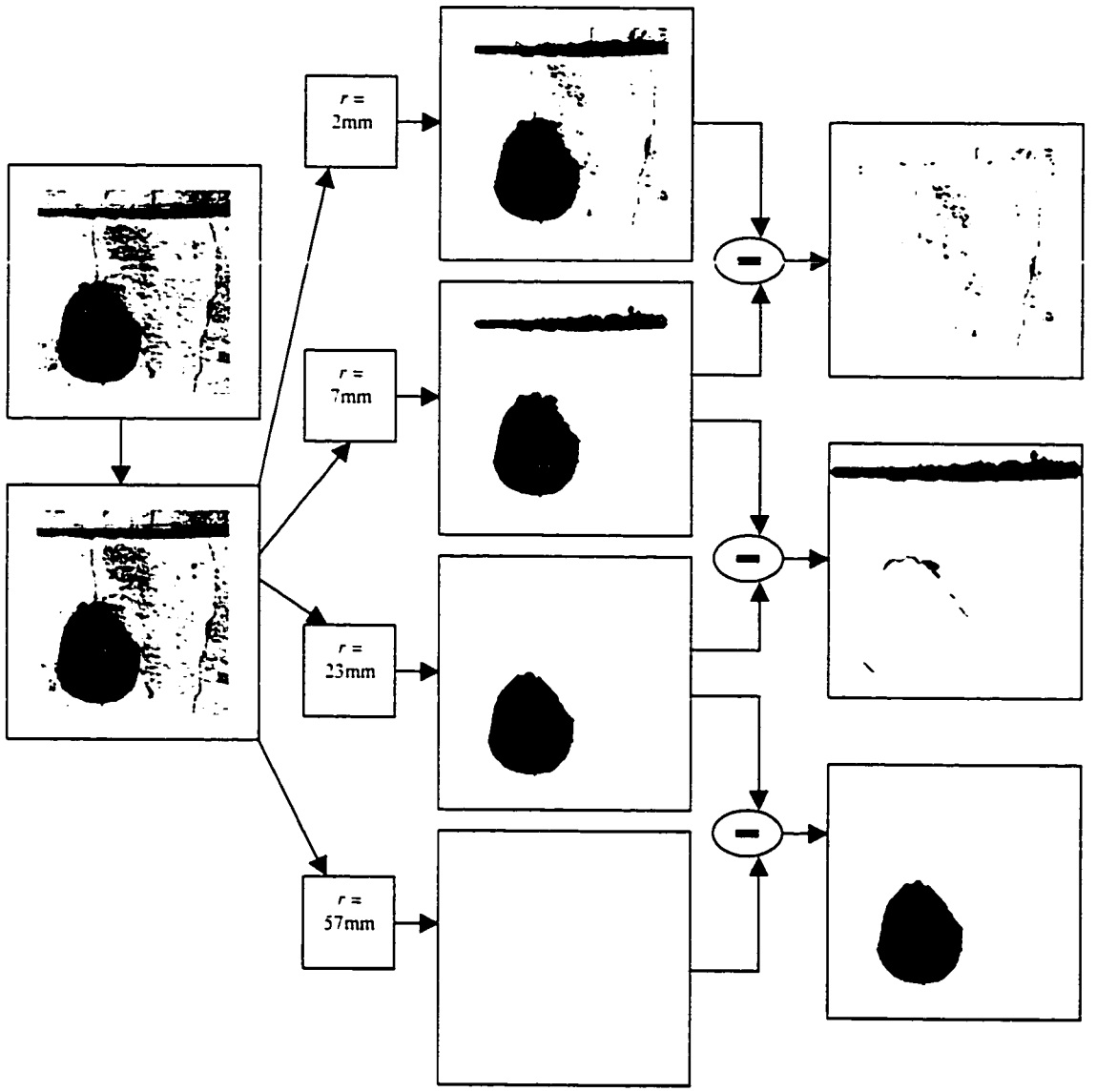


(d) After morphological opening by horizontal structuring element of length 155(mm)



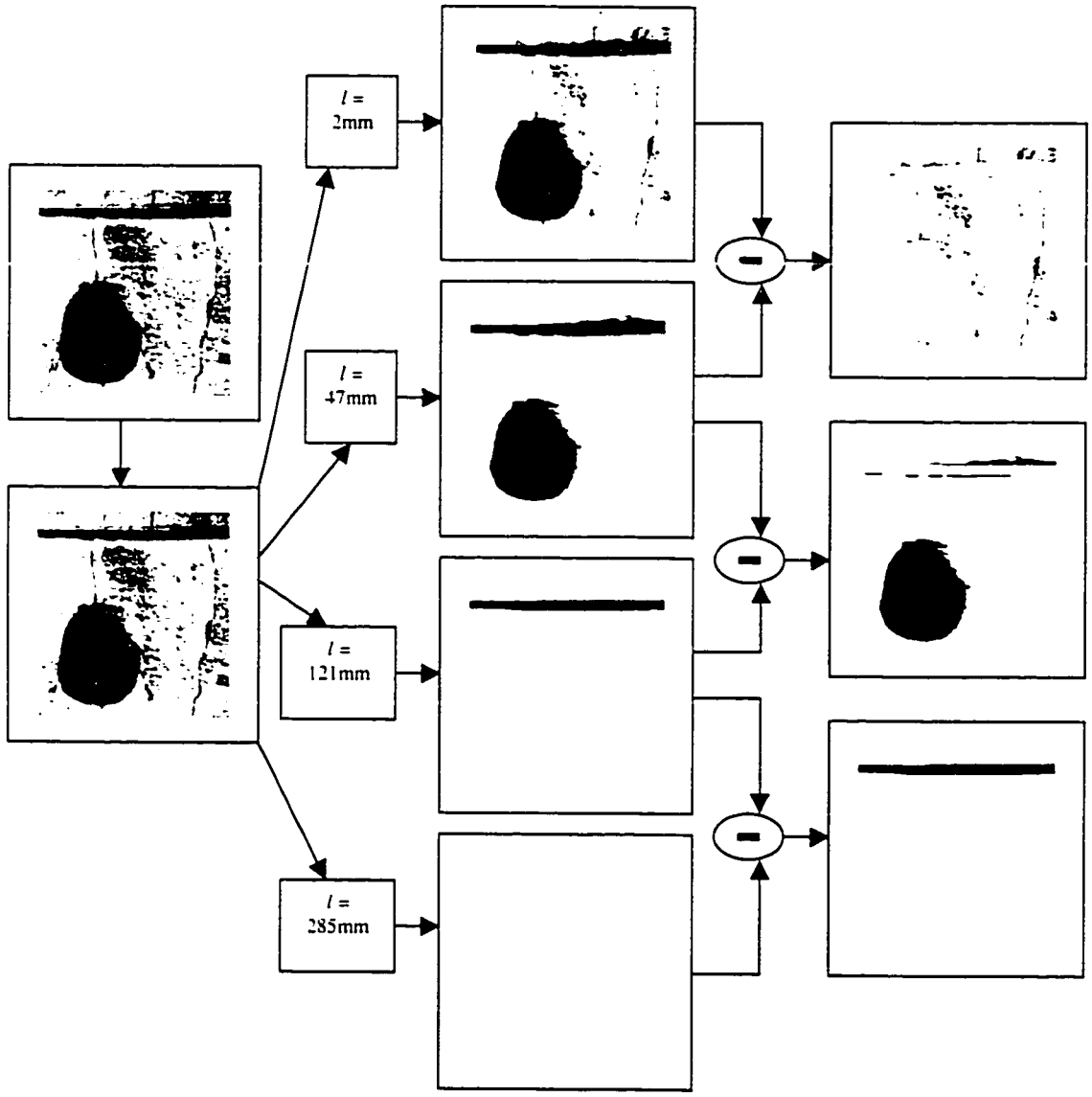
(e) After morphological opening by horizontal structuring element of radius 285(mm)

Figure 3-12. Classification results by using horizontal structuring element: the original images are opened by structuring element of different sizes as outlined in Table 2, and finally binary images are obtained by global thresholding technique.



14 (a) Gray-Scale Image 14(b) Thresholded Images After Opening Operations 14(c) Classified Images

Figure 3-13. Classification results by using circular structuring element: the original images are opened by structuring element of different sizes as outlined in Table 1, and finally binary images are obtained by global thresholding technique.



15(a) Gray-Scale Image 15(b) Thresholded Images After Opening Operations 15(c) Classified Images

Figure 3-14. Classification results by using horizontal structuring element: the original images are opened by structuring element of different sizes as outlined in Table 2, and finally binary images are obtained by global thresholding technique.

We may evaluate the performance of proposed segmentation and classification methods by a confusion matrix indicating whether the classification tendency is reasonable or not. On the confusion matrix, if we have a normal distributed matrix with few outliers, centered on the diagonal, the classification can be said to be reasonable. Tables 3-5 and 3-6 show the agreement and disagreement between the expert classification and the proposed classifier in terms of confusion matrix by using circular and horizontal structuring elements, respectively.

5. Conclusions

We have demonstrated a morphological approach to segment and classify images of underground concrete pipes. We assume that an image consists of five types of objects, namely the crack, hole, joints, laterals and surface collapse, and we estimate their sizes based on the concept of gray-scale morphological opening operations. With this size information, we provide a precise removal and classification of various objects present in an underground pipe image. Experimental results demonstrate that the proposed approach is effective for dealing with the underground pipe images with varying background pattern and non-uniform illumination.

Morphological segmentation and classification approach can be used to distinguish between laterals and joints, but it is difficult to classify defective joints and laterals from the perfect one. Therefore, the segmented joints and laterals have to be further processed to classified according to the severity of defects by using other shape or textural features, like roundness, compactness, etc. Again, the cracks in pipe images are extracted and classified well by the morphological segmentation approach, but the crack pixels are not detected precisely. Once the laterals, joints and holes are segmented and classified from the image then the crack detection filters, as described in [52], can be used for precise detection of crack features.

Table 3-5. Confusion matrix using the circular structuring element as classifier (Row-Morphological classifier results and Column-Expert labeling)

Class	Clean Pipe	Crack/Hole	Lateral	Pipe Collapse	Joints	Total	Percent Correct
	1	2	3	4	5		
1	**49**	0	0	0	1	50	98
2	0	**141**	0	0	9	150	94
3	0	0	**48**	0	2	50	96
4	0	0	0	**24**	1	25	96
5	0	0	0		**50**	50	100
Total	49	141	48	24	63	325	
	Overall Percentage Correct Classification -						96

Table 3-6. Confusion matrix using the horizontal structuring element as classifier (Row-Morphological classifier results and Column-Expert labeling)

Class	Clean Pipe	Crack/Hole	Lateral	Pipe Collapse	Joints	Total	Percent Correct
	1	2	3	4	5		
1	**49**	0	0	0	1	50	98
2	0	**141**	0	0	9	150	94
3	0	0	**48**	0	2	50	96
4	0	0	0	**24**	1	25	96
5	0	0	0		**50**	50	100
Total	49	141	48	24	63	325	
	Overall Percentage Correct Classification -						96

Chapter 4

4. Pipe Crack Detection

Segmentation aims at the separation of distresses (if any) from the image background. Pixels, as a result of the segmentation process, are classified into two categories: healthy pixels (background) and distress pixels (cracks). A binary image is subsequently obtained, where crack pixels have the value 0 and background pixels have the value 1. After segmentation, various operations can be performed on the binary image to determine the type and extent of distresses. The quality of the segmentation is thus very important: it may be difficult to classify the crack correctly in a poorly segmented image. Furthermore the distress – even if it has been correctly classified – may be measured with significant errors.

Given that we now have a segmented crack image (and have eliminated holes, joints, laterals, and pipe collapse) obtained by morphological approach as discussed in Chapter 3, in this Chapter we now present the development of a statistical filter for detection of cracks. We begin with an introduction in Section 4.1, followed by a development of the proposed crack detection filter in Section 4.2. In Section 4.3, we then present conventional techniques for detection of cracks. Experimental results and evaluation is given in Section 4.4.

4.1 Introduction

Many researchers have paid a great deal of attention to automated pavement cracking detection/classification. Based on the assumption that the histogram of the pavement containing cracking is bimodal (although it may not always be true), Li et al. [106] proposed an algorithm for pavement cracking detection. A standard model was proposed to represent pavement surface images toward a unified and automated acquisition of key characteristics for improving data quality [66]. However, this model did not discuss how to employ such a mode in crack detection/classification system. An approach to the recognition of segmented pavement distress images was studied in Mohajeri and Manning [127]. It uses directional filters to classify the cracks. The crack is longitudinal if there is a high concentration of object pixels in a narrow interval of x (transverse) coordinates, and it is transverse if there is a high count of object pixels in a narrow interval of y (longitudinal) coordinates. However, it is difficult to get a segmented crack image, and it is also not clear how to identify other crack types by analyzing these counts. Another statistical approach [99] recognized the imperfections of segmentation that cause difficulty in distinguishing pavement-cracking types, particularly between multiple and mushroom cracks. In this method, the original image is enhanced by subtracting an average of a few plain (nondistress) images from the same series to compensate for the lighting variations. A crack is detected by assigning one out of four values to each pixel, based on its probability of being an object pixel.

Usual edge detectors and thresholding techniques based on the difference between pixel values are inefficient when applied to underground pipe images. In analyzing an underground pipe scanned image, one needs to consider complications due to the inherent noise in the scanning process, irregularly shaped cracks, as well as the wide range of pipe background patterns. One of the major problems is detecting defects (especially cracks) that are camouflaged in the background of corroded areas, debris, patches of repair work, and areas of poorly illuminated conditions. The approach proposed in this thesis is based on the statistical properties of pixel values in the neighborhood. The decision threshold can be experimentally determined for a given probability of false alarm, as a function of

shape and size of the processing neighborhood. For a better and finer detection, the crack detection filter operates along the four usual directions over windows of increasing sizes. The experimental performance is in good agreement with the manual detection of cracks. The crack detection filter is finally compared with other conventional crack detectors.

4.2 Proposed Statistical Filters for Detection of Cracks

In the past 20 years, many approaches have been developed to deal with the detection of linear features on optic [58,126] or radar images [155,159]. Most of them combine two criteria: a local criterion evaluating the radiometry on some small neighborhood surrounding a target pixel to discriminate lines from background, and a global criterion introducing some large-scale knowledge about the structures to be detected.

Concerning the local criterion, most of the techniques used for pavement distress detection in scanned images are based either on conventional edge or line detectors [58,155]. These methods evaluate differences of averages, implying noisy results and variable false-alarm rates [127]. In addition, local criteria are in many cases insufficient for edge or line detection, and global constraints must be introduced [183]. For instance, dynamic programming is used to minimize some global cost functions, as in the original algorithm of Fishler [177] and its improvement [99]. Hough-transform-based approaches have also been tested for the detection of parametric curves, such as straight lines or circles [178]. Tracking methods are another possibility. They find the minimum cost path in a graph by using some heuristics, for instance, an entropy criterion [53]. Energy minimization curves, such as snakes, have been applied [126]. The Bayesian framework, which is well adapted for taking some contextual knowledge into account, has been widely used [170]. Regazzoni [147] defines a cooperative process between three levels of a Bayesian network, allowing the introduction of local contextual knowledge as well as more global information concerning straight line. Hellwich [76] uses *a priori* information concerning line continuity expressed as neighborhood relations between pixels.

We propose a two-step algorithm for detection of crack features in the segmented underground pipe images. The first step is local and is used to extract crack features from the segmented image, which are treated as crack segment candidates. A new local crack detection filter based on the statistical properties of pixel values is developed in this study. In the second global step, cleaning and linking operations obtain cracks.

4.2.1 Crack Detection Filters

The algorithm begins by performing a local detection of cracks. This is based on the fusion of the results from two crack detection filters, both taking the statistical properties of image into account. The first crack detector D1 is based on a ratio edge detector: An in-depth statistical study of its behavior is given in Lopes et al. [111]. The second crack detector D2, which has emerged from this research, uses the operators of Yakimovsky [195]. Responses from both the first and second detectors are merged to obtain a unique response as well as an associated direction in each pixel. The detection results are post-processed to provide candidate segments. Figure 4-1 shows the different steps of the proposed crack detection algorithms.

Crack Detector D1

The ratio crack detector is defined as the ratio of the average of pixel values of two nonoverlapping neighborhoods on opposite side of the points. Let index 1 denote the central region (cracks) and indices 2 and 3 both lateral regions (background), as shown in Figure 4-2. The amplitude of pixel x is noted A_x, so that the empirical mean μ_i of a given region i having n_i pixels is $\mu_i = \left(\frac{1}{n_i}\right)\sum_{x \in i} A_x$. The response of the edge detector between region i and j is defined as r_{ij}.

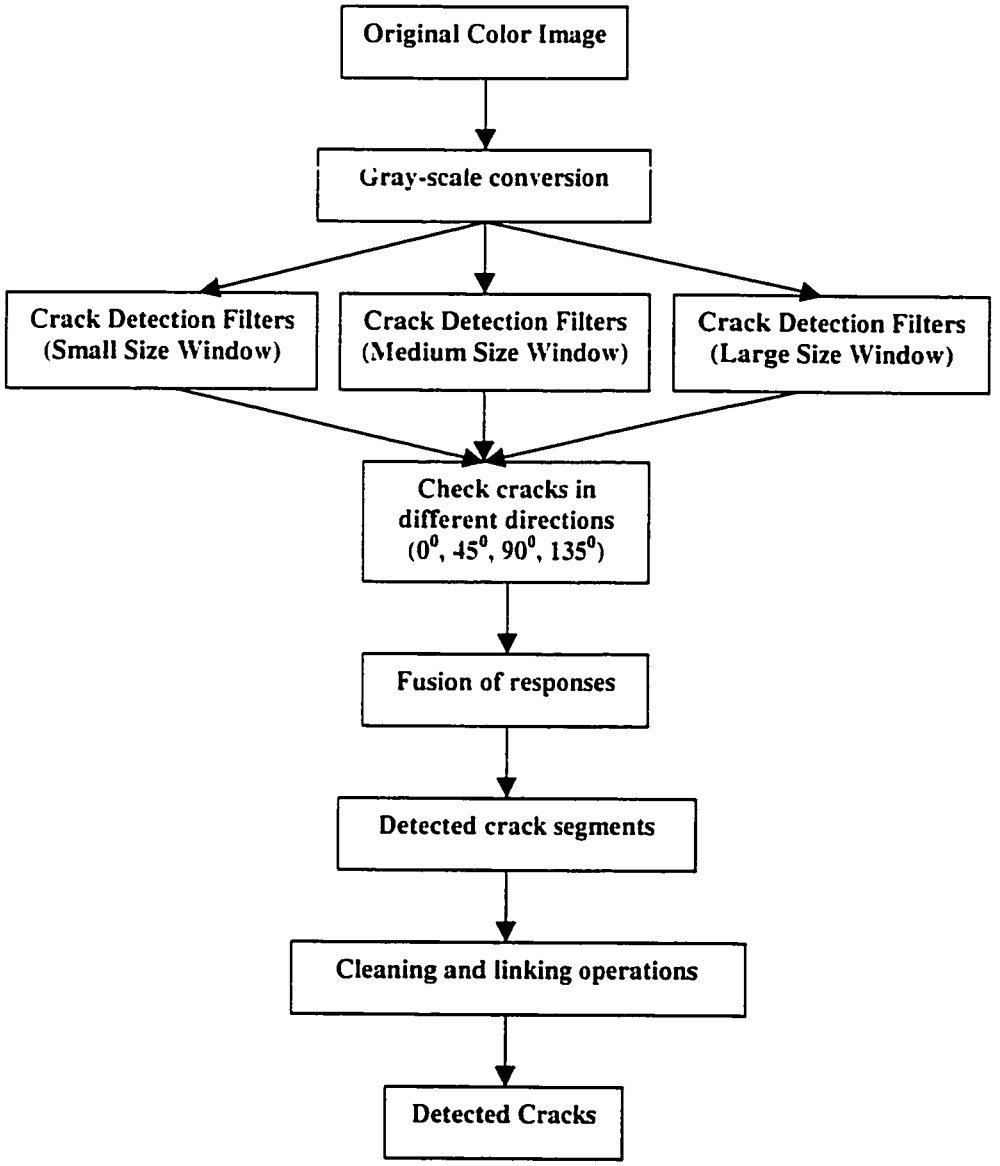


Figure 4-1. Diagram showing the different steps of the proposed method for crack detection in the underground pipe images.

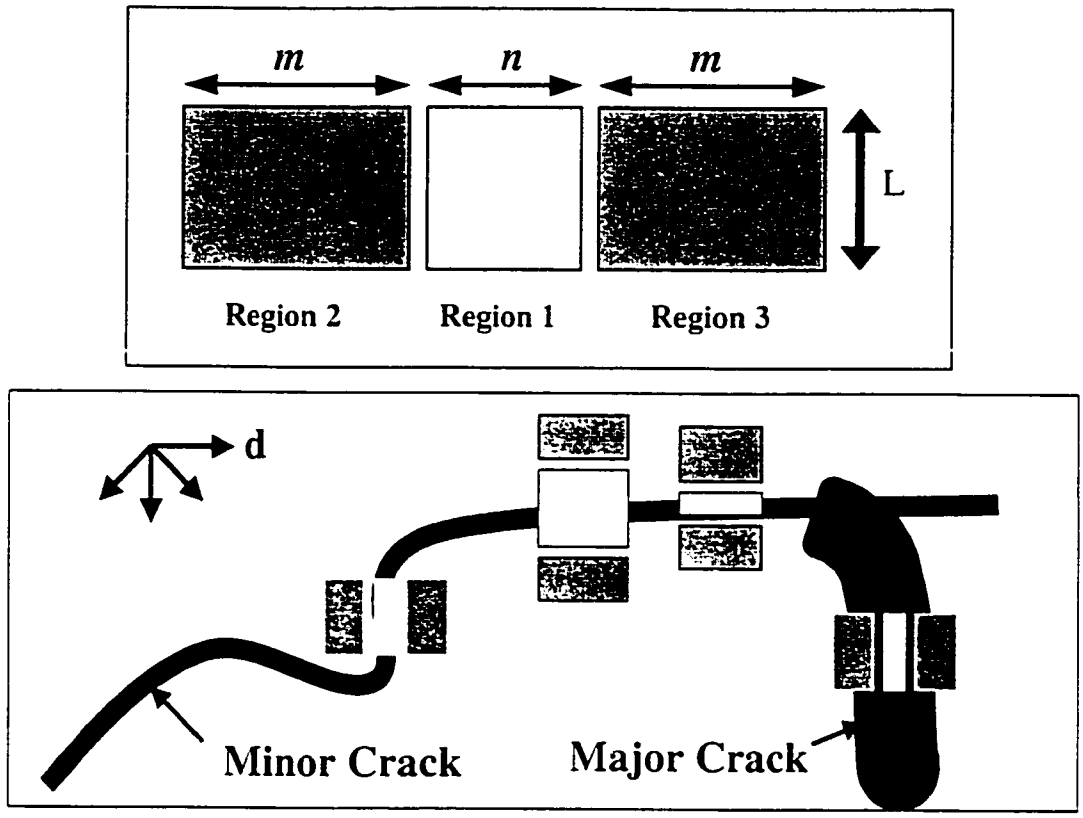


Figure 4-2. Crack model used by the two crack detectors D1 and D2.

$$r_{ij} = 1 - \min\left(\mu_i/\mu_j, \mu_j/\mu_i\right) \tag{4.1}$$

and the response to D1 as

$$r = \min(r_{12}, r_{13}), \tag{4.2}$$

the minimum response of a ratio crack detector on both sides of the crack structure. With detector D1, a pixel is considered as belonging to a crack when its response r is large enough, i.e., higher than some *a priori* chosen threshold r_{min}. The decision threshold r_{min} can be deduced from the statistical behavior of the detector. As usual, the detection probability (P_d) increases with a decreasing decision threshold (Figure 4-3(a)), but at the same time, the false-alarm (P_{fa}) rate increases (Figure 4-3(b)). Therefore, r_{min} may be deduced as a compromise between a P_{fa} and a minimum P_d rate.

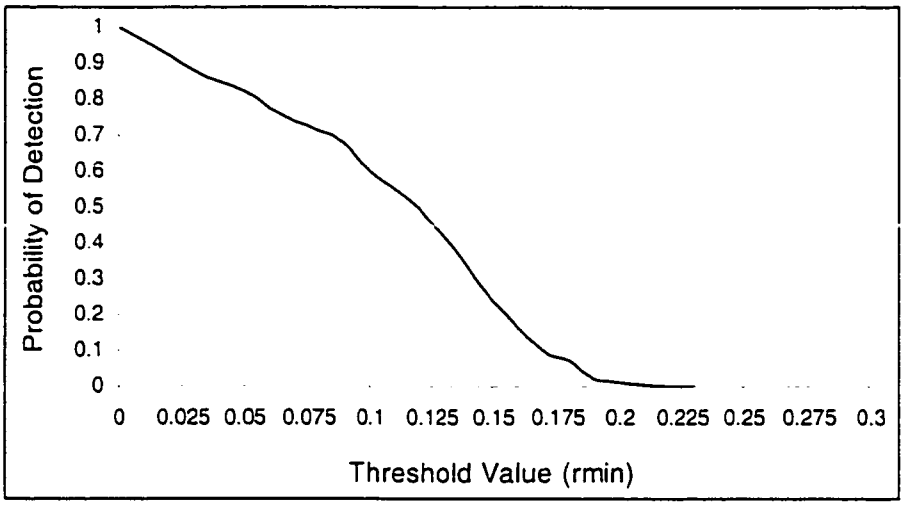


Figure 4-3(a). Probability of detection versus the minimum threshold value of crack detection filter

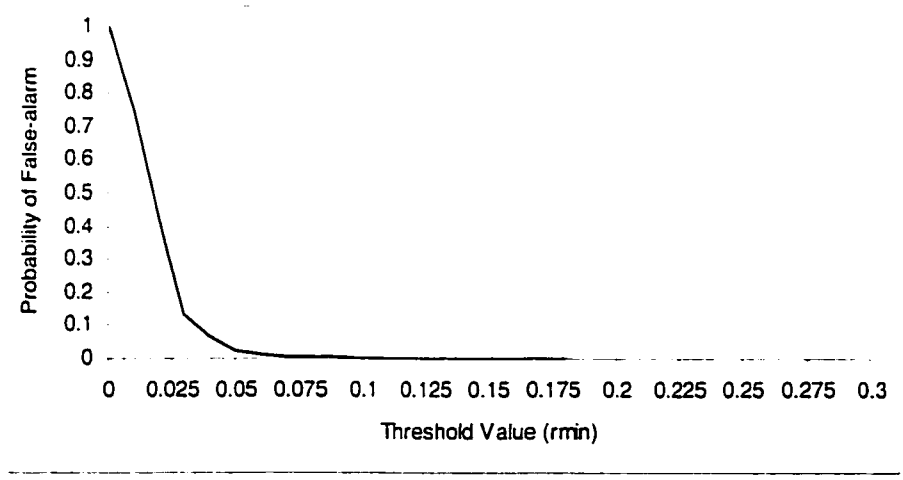


Figure 4-3(a). Probability of false-alarm versus the minimum threshold value of crack detection filter

Crack Detector D2

In practice, the ratio crack detector D1 is accurate, but only the mean comparison component behaves well when the operator is not centered on an edge interface. Therefore, we decided to use the variance comparison also as suggested by Yakimovsky [195]. The operators of Yakimovsky assume that cracks are interfaces between sets of points, each set being described by a normal distribution. The mathematics for distribution parameter comparison is used to form a function of crack strength in an area.

$$s = \left[\frac{(\sigma_0^2)^{m+n}}{(\sigma_1^2)^m (\sigma_2^2)^n} \right] \tag{4.3}$$

where,

σ_0^2 = Variance for region 1, 2, and 3 taken together.

$$= \frac{\left[m\sigma_1^2 + n\sigma_2^2 + m(\mu_0 - \mu_1)^2 + n(\mu_0 - \mu_1)^2 \right]}{(m+n)}$$

μ_0 = Mean for region 1, 2, and 3 taken together.

$$= (m\mu_1 + n\mu_2) / (m+n)$$

m, μ_1, σ_1^2 =Samples, mean, variance for region 1.

n, μ_2, σ_2^2 =Samples, mean, variance for region 2 and 3.

A pixel is considered as belonging to a crack when its response s is large enough, i.e., higher than some *a priori* chosen threshold s_{min}.

4.2.2 Fusion of Responses

In the previous sub-sections we have addressed the problem of detection of cracks in the segmented pipe images, using two detectors D1 and D2. Since there does not appear to exist a single detector suitable for reliable detection of cracks, we decided not to choose one of them, but to merge information from both D1 and D2 in each direction by using an associative symmetrical sum $\sigma(x, y)$, as defined in [21].

$$\sigma(x, y) = \frac{xy}{1 - x - y + 2xy}, \quad \text{with } x, y \in [0, 1] \tag{4.4}$$

This fusion operator has been chosen because of its indulgent disjunctive behavior for high values ($x > 0.5$, $y > 0.5$), its severe conjunctive behavior for small values ($x < 0.5$, $y < 0.5$), and its adaptive behavior, depending on x and y values in the other cases. Since the behavior of this operator depends on the position of the responses compared to the value 0.5, we first centered both D1 and D2 responses before applying the fusion, so that the decision threshold corresponds to 0.5. In order to do so, and constraining both r and s to lie in the interval [0, 1], we replace them by $\max[0, \min(1, x+0.5-x_{min})]$, where x equals r and s, respectively. As a result, the decision threshold applied on $\sigma(r,s)$ is automatically the central value 0.5 of interval [0,1].

To detect most of the cracks, the operators must be applied in all the possible directions. If the operator is applied separately for each direction, the same threshold must be used for all the considered directions. In this study, four directions (0^0, 45^0, 90^0, 135^0) are considered for the detection of cracks. The response image is thresholded with a threshold of 0.5, resulting in a binary image. The thresholded response from the crack detector is shown in Figure 4-4. As seen in Figure 4-4 the image contained many disjointed crack segments and noise. A necessary step for all detection methods using local detectors is the closing stage; starting from local information, a more global one must be deduced by a linking process.

4.2.3 Cleaning and Linking Operations

No detection methods are perfect, in the sense that it finds all the crack pixels and only crack pixels. Usually, unwanted noise is present in the form of short, erratic edges, and some crack pieces remain disconnected from other pieces by gaps. Both these problems are addressed in the cleaning and linking process.

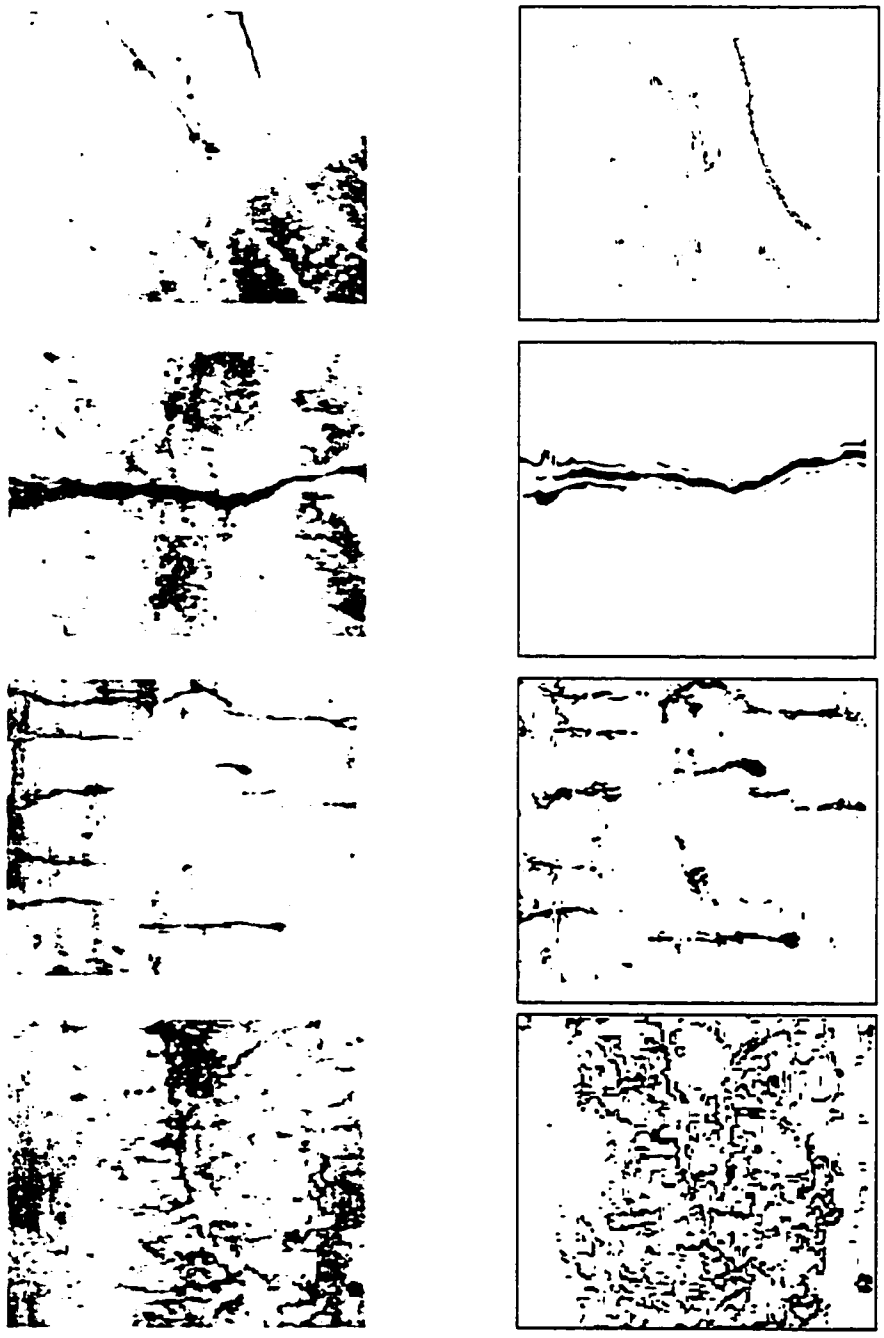


Figure 4-4. Crack detection filters response before cleaning and linking operations

As with the problem of crack detection, linking methods are either local or global. Local methods deal with a neighborhood around the pixel to be linked, analyzing the characteristics of these neighboring pixels to determine where the links should be. The simplest of local schemes focus on a small neighborhood and use information from earlier edge detection in order to find pixels with similar characteristics. As the size of the neighborhoods gets larger, searches are introduced based on a variety of path metrics [49]. Fuzzy reasoning has also been used to deduce which pixels in the search area should be linked together [104]. Global methods look at the overall pattern of cracks and try to describe the features using a few variables. Examples of global methods include modeling the crack image as a potential function [198] and using Hough transform [136], Markov random fields [59], and least-square-error curve fitting [141].

The linking procedure that follows in this study is a local method and specifically focuses on linking pixels that represent cracks. Cracks are usually long and relatively thin. These facts help determine which neighborhoods are searched for points to link together. The basic idea revolves around finding the direction in which the crack is headed and using this to outline a search area. A hierarchical clustering technique [47] is used for linking small gaps and removing unwanted noise in the form of short pieces. Clustering is a complex process that depends on several different kinds of information. Two important kinds are proximity and similarity [47]. A measure of similarity between clusters must be defined in order to have a specific algorithm. Nearest-neighbor [47] similarity measure is used because it has the characteristic of a chaining effect, which is well suited to extract string-like clusters. The first step in the linking process is to establish which crack points are end points and the direction in which the crack is heading. Once this is established, the crack clusters are linked if the distances between their end points are less than 15 pixels. Because many different cracks are often connected together, steps are taken to connect end points of each cluster only once. In this manner, links are made that fill small gaps in crack-segments. This also allows some of the noise to be eliminated by deleting isolated clusters with a total of 10 or fewer pixels. The filter response after cleaning and linking operations is shown in Figure 4-5.

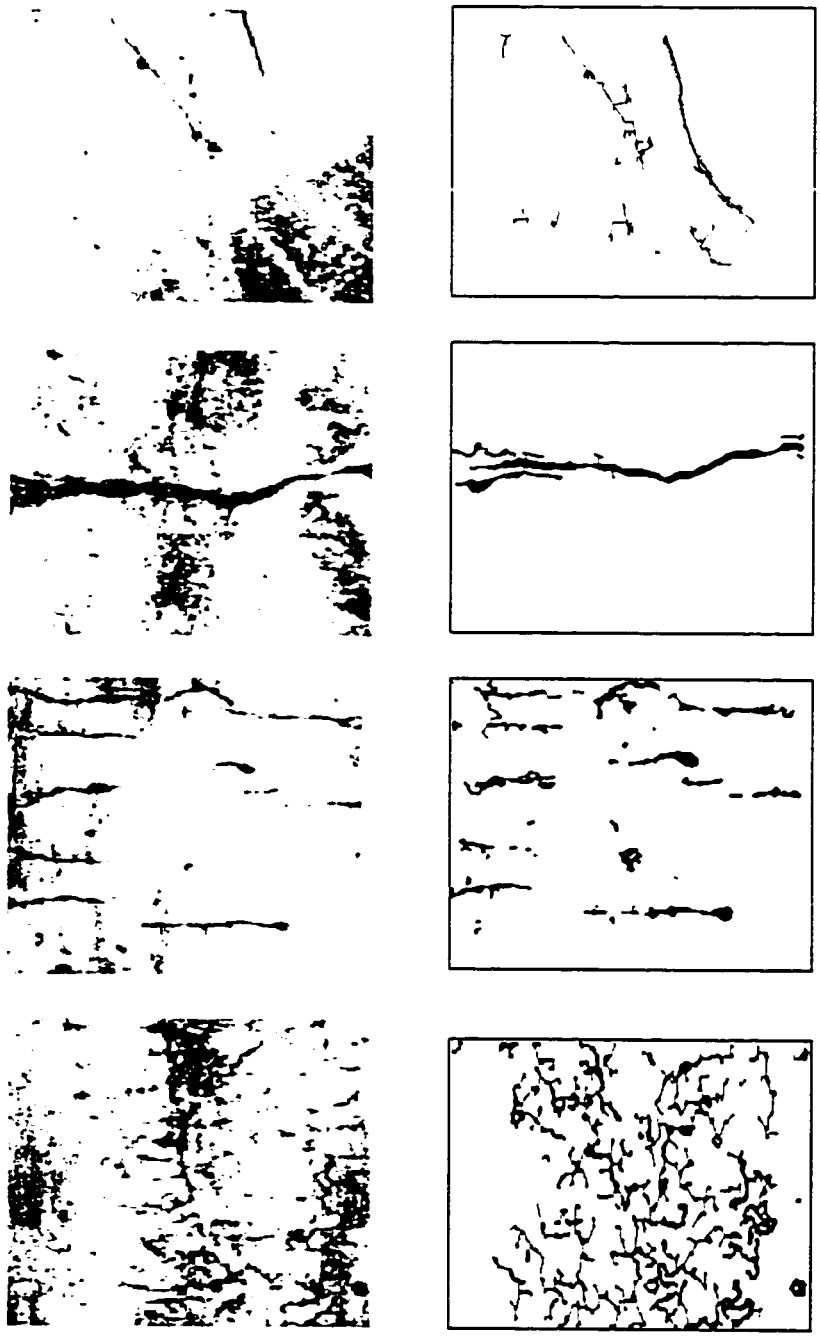


Figure 4-5. Filters response after cleaning and linking operations

The linking method can be further improved if special cases are accounted for. Linking only takes place between end points, ignoring the fact that other crack points may provide a better starting point for linking. The links are made with straight lines, neglecting the possibility that the crack may be in the process of curving. Overall, some crack gaps may be missed and some false links may be made. Nonetheless, provided that enough cracks are kept in the crack detection stage, this linking process will fill most of the gaps that should be filled and remove some of the unwanted noise.

4.2.4 Performance of the Crack Detectors

The evaluation of the crack detection filters is carried out by comparing the automatically detected cracks with manually plotted cracks. A set of connected pixels belonging to the cracks is manually extracted by developing software. The original crack image and the manually plotted reference map by the software are shown in Figure 4-6. The evaluation of proposed crack filters is processed by matching the extracted crack pixels to the reference map. There are various ways to perform matching of two images. The so-called 'buffer-method' [120], is a simple matching procedure in which a buffer of constant predefined width is constructed around the crack data in two steps. In the first step a buffer of constant width is constructed around the reference crack data by using morphological dilation operation by using structuring element of size 5x5, as shown in Figure 4-7(a). The parts of the extracted data within the buffer are considered as matched and is denoted as *true positive*. The unmatched extracted data is denoted as *false positive*. In the second step matching is performed the other way round. The buffer of same size is now constructed around the extracted crack data, as shown in Figure 4-7(b), and the part of the reference data lying in the buffer is considered as matched. The unmatched reference data is denoted as *false negative*. Figure 4-8 illustrates the matching procedure for obtaining true positive and false positive pixels. In order to quantitatively assess the filter effects, we use notation designed as follows.

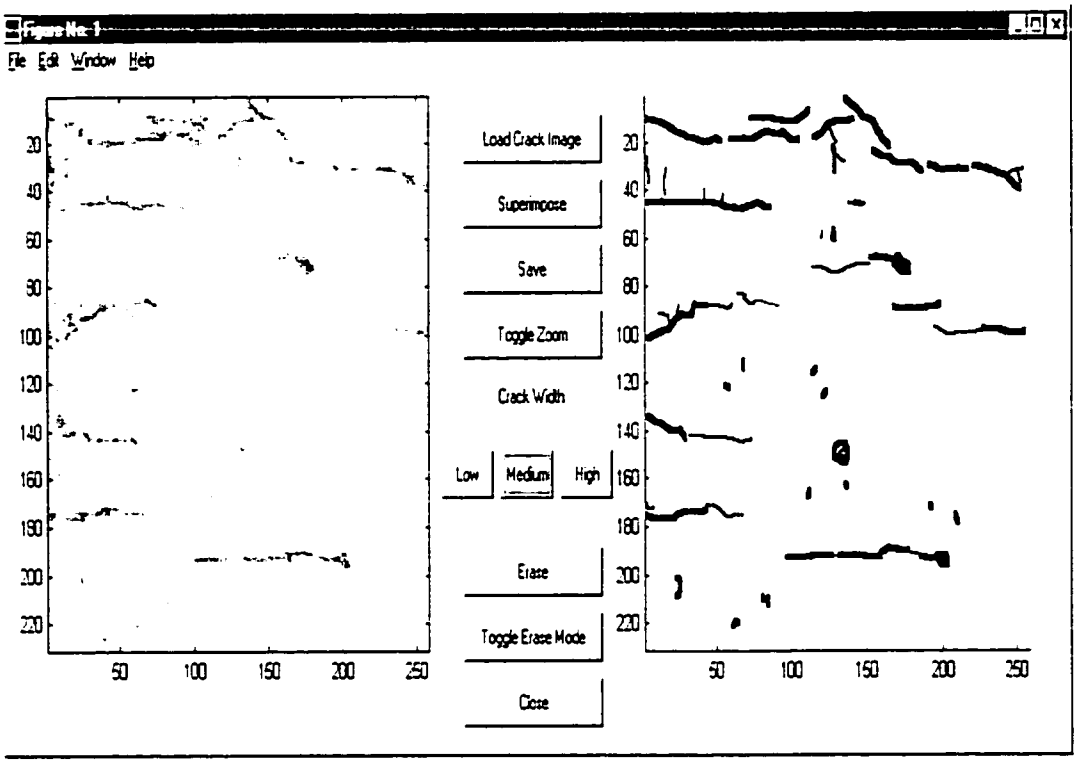


Figure 4-6. Software for extraction of crack reference map for evaluation of the proposed crack detection filters, original crack image is shown on the left side of the window and the manually plotted crack reference map on the right window.

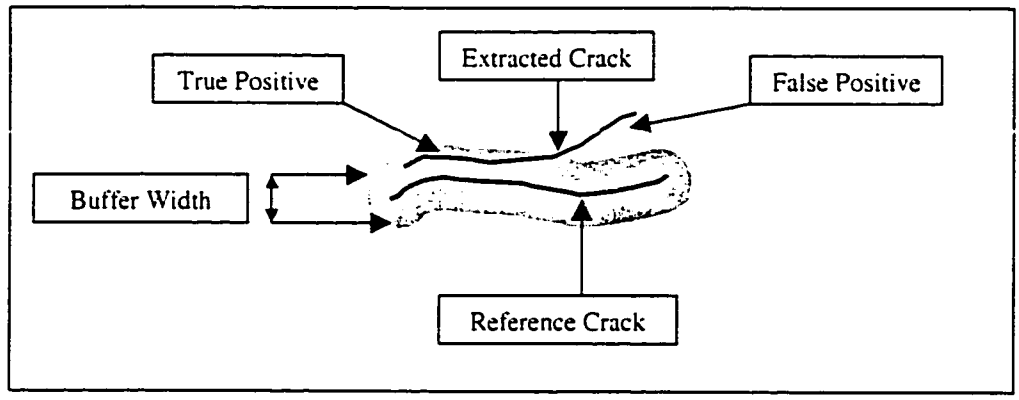


Figure 4-7(a). Matched Extracted

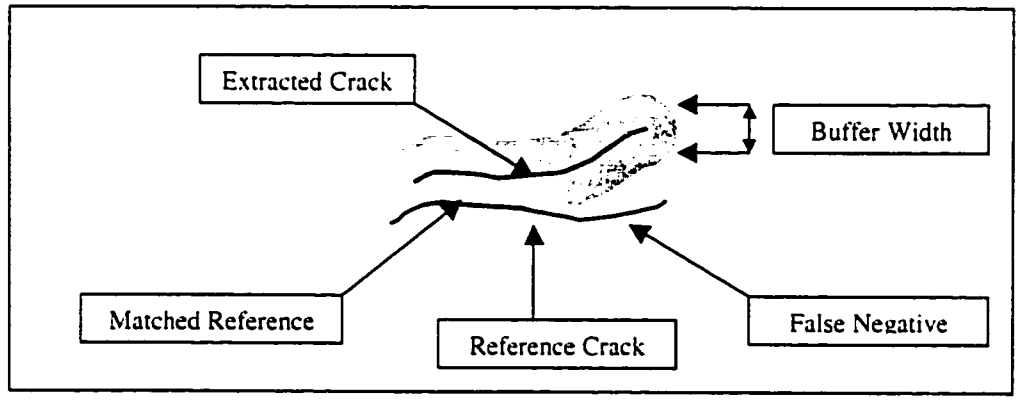


Figure 4-7(b). Matched Reference

Figure 4-7. Matching principle for the evaluation of crack detector responses.

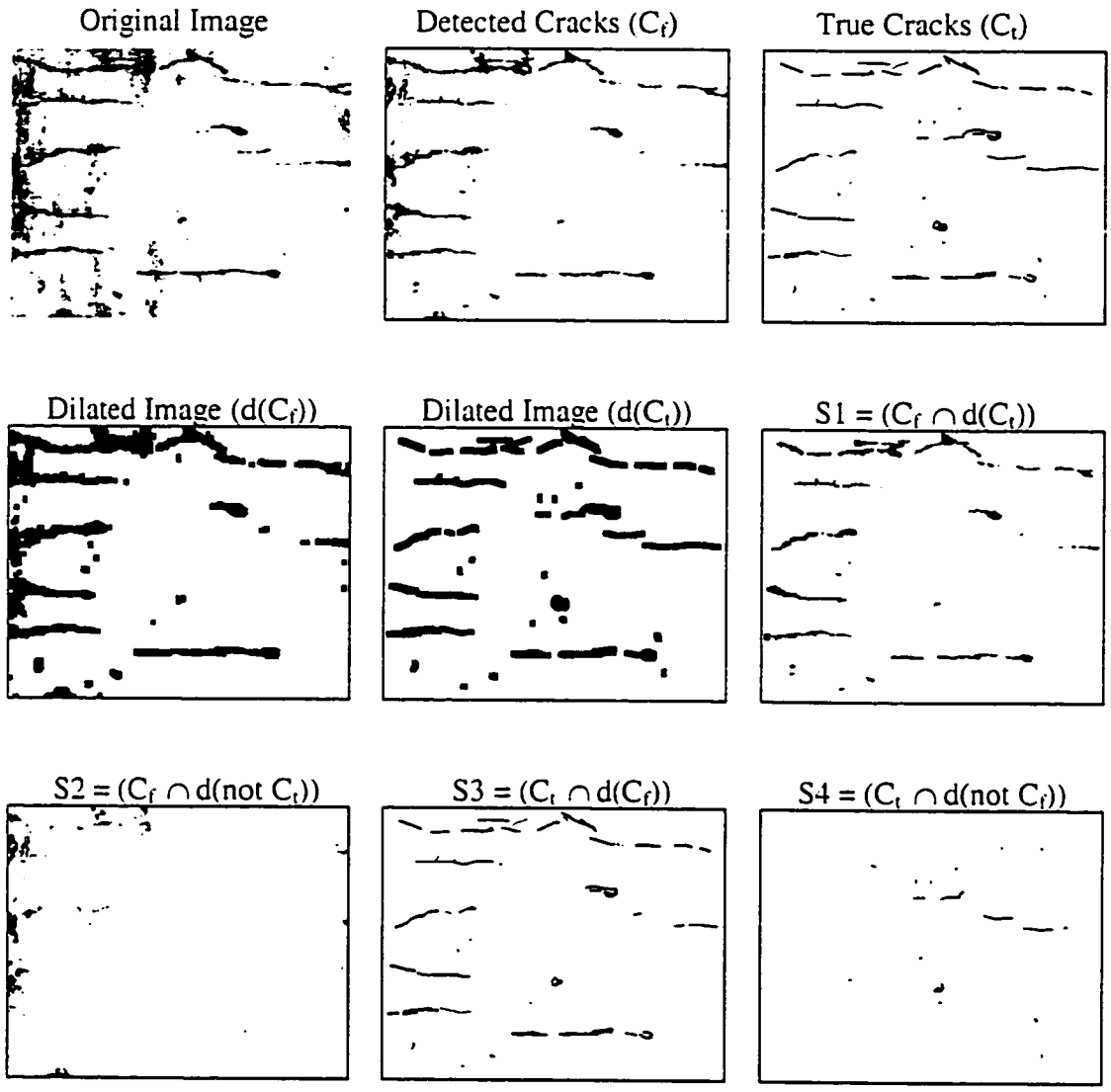


Figure 8. Illustrates the procedure of matching the images for detection of true and false pixels.

Let C_f be the set of pixels detected as crack pixels in the image filtered by filter f. A set of connected pixels belonging to the crack is extracted manually by the experts. Let C_t denote this set of true crack pixels. Let S_1 be the number of pixels belonging to $C_f \cap d(C_t)$. S_1 is the number of good crack pixels detected by the filter, and $d()$ denotes the morphological dilation by some structuring element. The matched extracted data is denoted as true positive, emphasizing the fact that the extraction algorithm has indeed found a set of cracks and is denoted as S_3 belonging to $C_t \cap d(C_f)$. Let S_2 be the bad points from the filter, which is called false positive and is defined as $C_f \cap d(C_t)$. Similarly a set of true points S_4 which were missed by the filter is defined as $C_t \cap d(C_f)$ and is called false negative. Now, the probability of detection (P_d) and probability of false-alarm (P_{fa}) can be defined as follows.

$$P_d = \frac{Number\ of\ \text{det}ected\ crack\ pixels}{Number\ of\ true\ crack\ pixels} \tag{4.5}$$

$$P_{fa} = \frac{Number\ of\ false-alarm\ pixels}{Number\ of\ non-crack\ pixels} \tag{4.6}$$

The performance of the crack detectors must be discussed as a function of the size of neighborhoods, and the decision thresholds. In order to quantitatively assess the different size and threshold of filter effects, we use operational curves designed as follows. To evaluate the performance of detector for different window sizes and threshold values, probability of detection (P_d) is plotted against probability of false-alarm (P_{fa}). To optimise the widow size (Figure 4-2), we performed experiments by varying the shape and size of the windows (L=3,5,7,9,11,13,15, m=3,5,7,9,11, and n=2,3,4,5,6,7). In each case the threshold values are varied from 0 to 0.3 in the increments of 0.025. In practice, it is found that with a large neighbourhood, the detection operator is less sensitive to the noise. Unfortunately, the small cracks, which may be detected by small neighbourhoods, may be missed. Therefore, in order to detect most cracks the detectors must operate over neighbourhoods of different sizes. Three sizes of crack detection filters are selected based

on empirical receiver operating characteristic (ROC) curves [104] for detecting minor, major and multiple cracks.

The ROC curve [104] is plotted as a function of P_d and P_{fa} (as shown in Figures 4-3(a) and 4-3(b) for different neighbourhood sizes. ROC curves from ten images in each class (i.e., minor, major, and multiple cracks) are aggregated into one by averaging the points from individual ROC curves at a set of points on the (percent of false-alarm) axis. The ROC curve for a given crack detector summarizes its range of tradeoffs between true positive and false positive crack pixels, as determined by comparing the detected crack pixels to the specified ground truth. Given a low P_{fa}, in this study P_{fa} is selected as 1% as suggested by the municipal engineers, the corresponding window size is selected which gives the maximum P_d value. The minimum false-alarm rate (0.01), corresponding to the most probable crack detection, is assigned to the considered threshold values r_{min} and s_{min}. This process is repeated for three different classes of cracks, i.e., which correspond to minor, major and multiple cracks. Therefore, three optimal sizes of windows and the corresponding threshold values r_{min} and s_{min}, as shown in Table 4-1, are used for detection of cracks in underground pipe images.

Table 4-1. Optimal window size for central and adjacent regions with threshold values for crack detection filters

Filter size	Window size for adjacent region [2]	Window size for central region [1]	Window size for adjacent region [3]	Threshold values for first crack detector D1	Threshold values for second crack detector D2
	1	2	3	4	5
Small	5x3	5x2	5x3	0.05	0.15
Medium	7x5	7x3	7x5	0.1	0.25
Large	11x7	11x5	11x7	0.175	0.35

4.3 Conventional Techniques for Detection of Cracks

To study the performance of proposed crack detection filters and to compare it with the performance of other conventional detection techniques, we have used Canny's edge detector [26] and Otsu's thresholding [133] technique to extract crack features. The Canny operator is used because it can perform very well when one is trying to detect edges due to intensity change. It is known for emphasizing weak edges and yet suppressing edge output due to noise. Otsu's thresholding method is selected for extraction of cracks because it is non-parametric, unsupervised, and automatic. The following sub-sections will briefly discuss the Canny's and Otsu's techniques.

4.3.1 Extraction of Cracks by Canny's Edge Detection

Edge-detection techniques segment objects by outlining their boundaries using information on gray-scale discontinuity. This step, however, seldom produces connected object edges due to noise and other factors. Thus, edge linking and other boundary detection methods usually follow to transform the set of edge pixels obtained into a meaningful set of object boundaries. The Canny edge detector [26] uses linear filtering with a Gaussian kernel to smooth the noise in the image. Next, the edge strength and direction are calculated for every pixel in the smoothed image. The Canny operator does this by differentiating the image in the horizontal and vertical directions, and then computes the gradient magnitude as the root sum of squares of the derivatives. The arctangent of the ratio of the derivatives is used to compute the gradient direction. The next step is called non-maximal suppression. In this process, the edge strength of each candidate edge pixel is set to zero if its edge strength is not larger than the edge strength of the two adjacent pixels in the gradient direction. The pixels that survive the non-maximal suppression thinning process are labeled as candidate edge pixels. An adaptive thresholding method is then applied on the thinned edge magnitude image to obtain the final edge map.

4.3.2 Extraction of Cracks by Otsu's Thresholding Technique

The Otsu method [133] is an automated and unsupervised method of thresholding using information on the gray-level histogram. A discriminant criterion between the two classes of pixels is computed for each possible threshold T; the optimal threshold is that gray-level where this measure is maximized. It has the advantages that it is simple and easy to implement and the threshold thus selected is not based on the differentiation (a local property) but rather on the integration (a global property) of the histogram. Therefore, as a result, the criterion measure is always unimodal and stable.

The brief summary of this threshold algorithm is given here. Let the pixels of the image be represented by V gray levels $\{0, 1, 2, \ldots, V\text{-}1\}$. The number of pixels in level v is denoted by n_v and thus the total number of pixels is $N = n_0+n_1+n_2+\ldots+n_{V-1}$. To simplify, the gray level histogram is normalized and regarded as a probability distribution function:

$$P_v = \frac{n_v}{N} \qquad P_v \geq 0 \qquad \sum_{v=0}^{V-1} P_v = 1 \tag{4.7}$$

Suppose we divide the pixels into two classes C_0 and C_1 (background and object) by a threshold value at k; C_0 denotes pixels with levels $[0, 1, \ldots, k]$ and C_1 denotes pixels with levels $[k+1, \ldots, V\text{-}1]$. The probabilities of class occurrences ω and class mean levels μ for both classes are given by:

$$\omega_0 = \sum_{v=0}^{k} P_v \qquad \textit{and} \qquad \mu_0 = \frac{\mu_k}{\omega_k} \tag{4.8}$$

$$\omega_1 = 1 - \omega_0 \qquad \textit{and} \qquad \mu_1 = \frac{\mu_T - \mu_k}{1 - \omega_k} \tag{4.9}$$

where

$$\mu_k = \sum_{v=0}^{k} vP_v; \quad \omega = \sum_{v=0}^{k} P_v; \quad \mu_T = \sum_{v=0}^{V-1} vP_v$$

To measure the "goodness" of the threshold, a criterion measure is introduced by Otsu:

$$\eta = \frac{\sigma_B^2}{\sigma_T^2} \tag{4.10}$$

where

$\sigma_B^2 = \omega_0(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2$ is the between – class variance

$\sigma_T^2 = \sum_{v=0}^{V-1}(v - \mu_T)^2 P_v$ is the total variance

We search for the optimal threshold T^*, which maximizes η, or equivalently maximizes σ_B^2, since σ_T^2 is independent of T. After finding T^*, we perform the global thresholding to obtain the final binary image.

4.4 Evaluations and Experimental Results

We illustrate the proposed method on real underground pipe images showing the potential of the proposed method and the difficulties remaining to be solved. External evaluation of the obtained results is of major importance for the relevance of any automatic system for practical applications. External evaluation needs some kind of reference data and compares them to the automatically obtained results. In this thesis we deal with the external evaluation of automatic crack extraction algorithms by comparison with manually plotted cracks used as reference data.

4.4.1 Quality Measures

The quality measures for crack extraction are intended to compare the results of different crack detection techniques, rather than to evaluate the extraction and the matching results in an absolute way as discussed in the sub-section 4.2.4. The definitions of the quality measures are presented in the following.

- **Completeness**

  Completeness = *length of matched reference / length of reference*

  $$\approx \frac{S_3}{S_3 + S_4}$$

Completeness $\in [0,1]$

The completeness is the percentage of the reference data that is explained by the extracted data, i.e., the percentage of the reference network that lies within the buffer around the extracted data. The optimum value for the completeness is 1.

- **Correctness**

Correctness = *length of matched extraction / length of extraction*

$$\approx \frac{S_2}{S_1 + S_2}$$

Correctness $\in [0,1]$

The correctness represents the percentage of correctly extracted crack data, i.e., the percentage of the extracted data that lies within the buffer around the reference network. The optimum value for he completeness is 1.

- **Quality**

Quality = *length of matched extraction / (length of extraction + length of unmatched extraction)*

$$\approx \frac{S_1}{S_1 + S_2 + S_4}$$

Quality $\in [0,1]$

The quality is a more general measure of the final result combining completeness and correctness into a single measure. The optimum value for quality is 1.

Quality = *completeness * correctness / {completeness – (completeness * correctness + correctness)}*

- **Redundancy**

Redundancy = *(length of matched extraction – length of matched reference) / length of matched extraction*

$$\approx \frac{S_1 - S_3}{S_1}$$

Redundancy $\in [-\infty, +\infty]$

The redundancy represents the percentage to which the correct (matched) extraction is redundant, i.e., it overlaps itself. The optimum value for the redundancy is 0.

4.4.2 Experimental Results

We have tested the proposed crack detection filter by applying it to a variety of segmented underground pipe crack images and compared the results with those obtained by using the Canny's edge detection and the Otsu's thresholding technique. Examples of the application of the proposed approach, Canny's edge detection method, and Otsu's thresholding technique are shown in Figure 4-9, 4-10, and 4-11, and the results of quality measures are summarized in Tables 4-2, 4-3, and 4-4.

It can be observed that the proposed filters perform better than Canny's method and Otsu's technique for detection of various kinds of cracks in underground pipe images. The second and third image in Figure 4-10(a) are part of Toronto sewer pipeline system, showing some minor cracks and major crack in the pipe surface, respectively. In this case, the crack detection step performs quite well [Figure 4-10(c)] detecting most of the minor and major crack structures in the images, while missing only the micro-cracks in the second image that may be there at the time of manufacturing the pipe, and experts in the pipe industry feel that this kind of micro-cracks do not cause any structural problem. The fourth scanned image [Figure 4-10(a)] is from the city of Boston. This image has dark background pipe surface with multiple cracks. In this case, the crack detection step performed well, but results are little noisy with few false alarms. The cleaning and linking operations shown to be a powerful method, which is able to fill gaps between the detected segments providing a map of the crack pipe surface, while suppressing most of the false-alarm detection [Figure 4-10(c)]. In fact, the results are close to those that could be obtained by a trained human operator. The last image in Figure 4-10(a) is from the city of Los Angeles. This image has a combination of many minor and major cracks (mushroom cracks) with patches in the background. In this case, the crack detection step did not perform very well, resulting in noise with many false alarm responses. The cleaning and linking operations are not very successful in this case because of the nature of the cracks. Performance evaluation of such kinds of cracks is not easy because it is difficult even for human operators to track down such cracks.

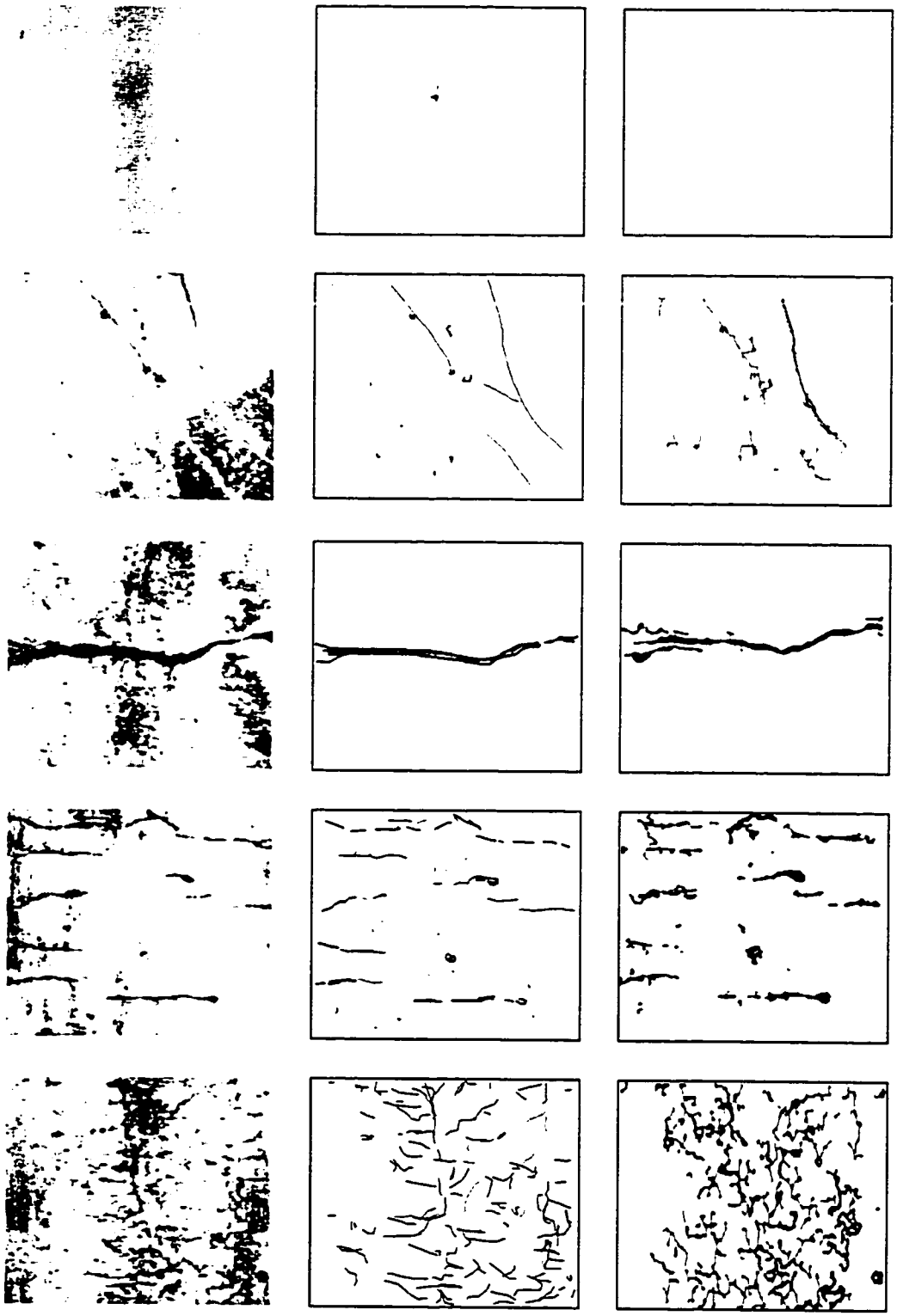


Figure 4-9. Crack detection filters response after cleaning and linking operations

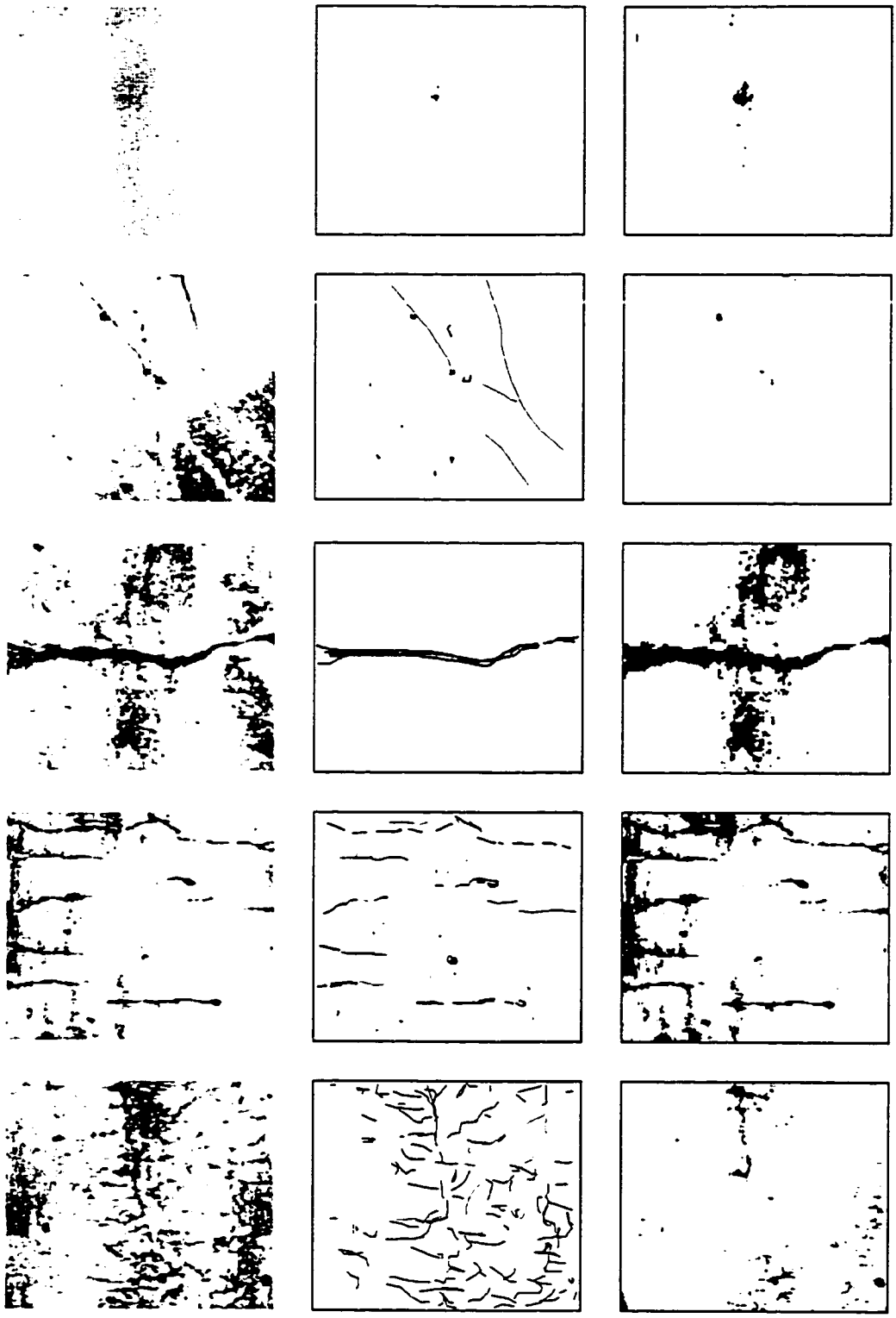


Figure 4-10. Otsu's method response after cleaning and linking operations

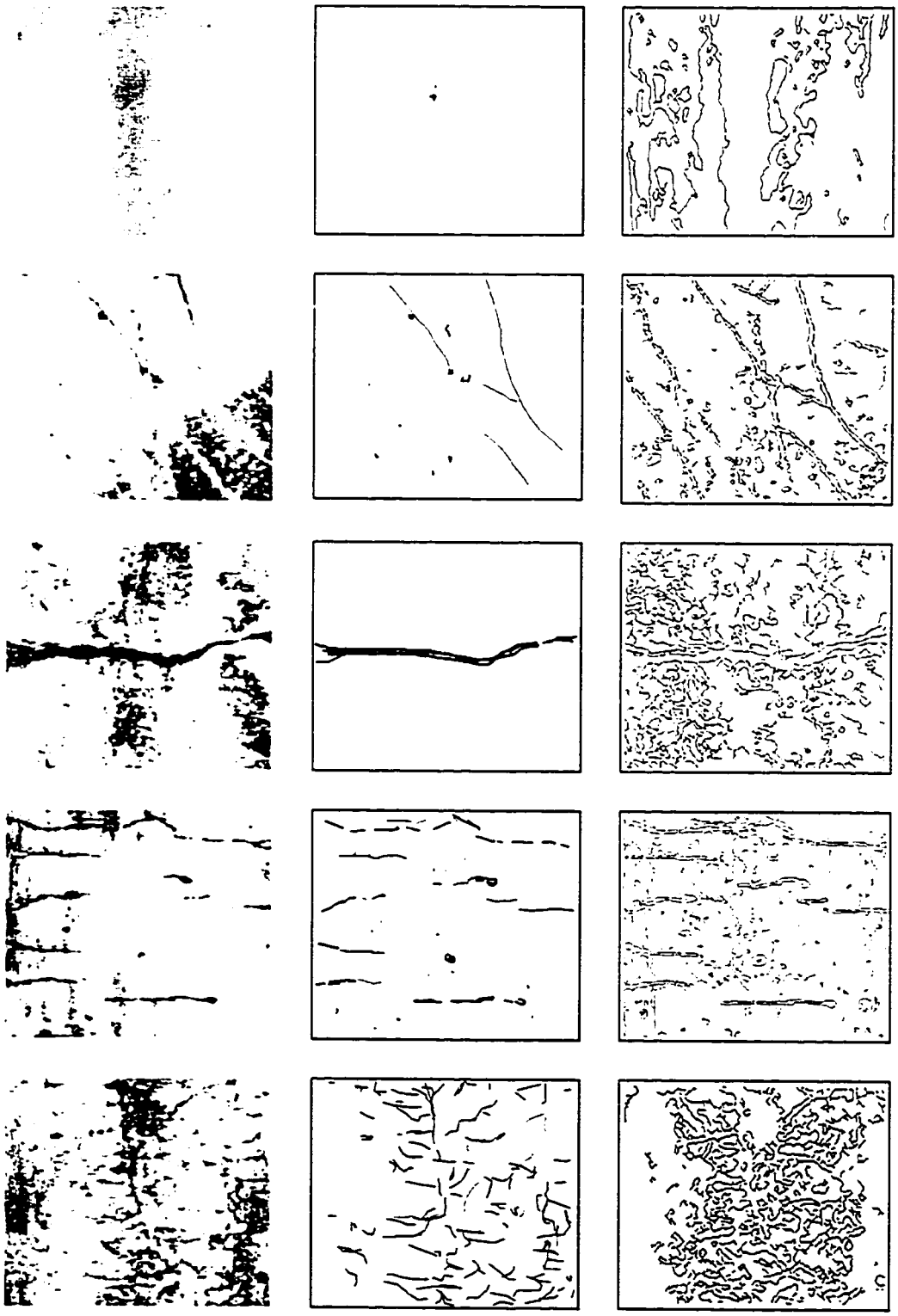


Figure 4-11. Canny's method response after cleaning and linking operations

Table 4-2. Quality measures for four different types of crack in the underground pipe images for the proposed crack detection filters.

Class	Minor Cracks	Major Cracks	Multiple Cracks	Mushroom Cracks
Completeness	0.9	0.85	0.8	0.77
Correctness	0.99	0.98	0.94	0.93
Quality	0.89	0.84	0.76	0.73
Redundancy	0	-0.01	0	0.02

Table 4-3. Quality measures for four different types of crack in the underground pipe images for Otsu's thresholding technique.

Class	Minor Cracks	Major Cracks	Multiple Cracks	Mushroom Cracks
Completeness	0.35	0.99	0.96	0.27
Correctness	0.52	0.29	0.41	0.65
Quality	0.37	0.29	0.4	0.25
Redundancy	0.24	-1.45	0.54	0.091

Table 4-4. Quality measures for four different types of crack in the underground pipe images for Canny's thresholding technique.

Class	Minor Cracks	Major Cracks	Multiple Cracks	Mushroom Cracks
Completeness	0.57	0.64	0.99	0.96
Correctness	0.15	0.06	0.37	0.35
Quality	0.14	0.06	0.38	0.36
Redundancy	-1.41	-1.72	-0.25	-0.31

4.5 Conclusions

The crack detection filters proposed in this Chapter can be simply divided into three steps. In the first step, the crack detection filters D1 and D2 are used to extract cracks by taking into account the statistical properties of pixels within a small neighborhood. In the second step, the responses from both the detectors are merged to obtain a unique response as well as an associated direction in each pixel. In the third step, the detection results are post-processed by cleaning and linking operations to provide crack segments.

In this chapter a methodology for the evaluation of automatic crack detection filters based on the comparison to manually plotted reference data is also presented. The proposed evaluation scheme adequately captures the characteristics of the individual detection results and can thus serve as a basis for their comparison. Depending on the application at hand, some of the quality measures such as completeness in an automated environment may be more relevant than others.

Comparing the proposed crack detection filters and the conventional detection techniques (i.e., Canny's and Otsu's), the improved experimental results have been achieved by the proposed statistical filters. The crack filters processing along four directions over windows of increasing size and followed by cleaning and linking operations, can detect minor cracks (with small windows) as well as major cracks (with larger windows). The overall performance of proposed crack detection filters is found to perform well for underground pipe images with minor, major, multiple, and mushroom cracks.

Chapter 5

5. Feature Selection and Classification

This chapter proposes a neuro-fuzzy classifier that combines neural networks and concepts of fuzzy logic for the classification of objects by extracting features in previously segmented underground pipe images. A comparative evaluation of the K-NN, fuzzy K-NN, conventional backpropagation network, and proposed neuro-fuzzy classifiers is carried out. The theoretical background of all four classifiers is presented and their relative advantages are discussed. All the classifiers are trained and tested by extracting suitable features from the segmented pipe images. After the introduction, Section 5.2 demonstrates the selection of various features for classification of pipe objects. Section 5.3 deals with the development of statistical and neural classifier models. Section 5.4 discusses the results obtained by various classifiers, and conclusions are presented in Section 5.5.

5.1 Introduction

Feature extraction and classification of objects is an important area of research and of practical applications in a variety of fields, including pattern recognition and artificial

intelligence, statistics, cognitive psychology, vision analysis, and medicine [46,47,61,82,150,172,186,192]. Over the last twenty-five years, extensive research has taken place in the development of efficient and reliable methods for the selection of features in the design of pattern classifiers, where the features constitute the inputs to the classifier. The quality of this design depends on the relevancy, discriminatory power and ease of computation of various features. Another important issue in object classification is the choice of an appropriate classifier. There are two types of classifiers: traditional classifiers (i.e. linear discriminant, maximum likelihood, k-nearest neighbour, etc.), and the neural network classifiers (i.e. backpropagation, self-organizing map, adaptive resonance theory, etc.).

Many researchers have paid a great deal of attention to automated pavement cracking classification [33,34]. Chou et al. [37] employed a fuzzy filtering image enhancement, fuzzy thresholding based on the maximum fuzzy entropies, and classification through a neural network trained with feature vectors as inputs. Automated real-time pavement distress detection using fuzzy logic and neural networks was studied [32] using fuzzy homogeneity for image enhancement and feature extraction. A methodology for automated pavement crack detection [94] demonstrated the potential of using neural network for classification and quantification of cracking on pavement, and it requires further improvement of the image segmentation process. However, in practice, most of the above approaches have been only partially successful and have been shown to produce high false-alarm rates, probably because of an inadequate segmentation algorithm.

In underground pipe defect analysis, the main objective is to accurately classify cracks, holes, laterals, joints, and pipe collapse by type, severity, and extent of distress. To achieve this objective, we propose a neuro-fuzzy classifier based on suitable features extracted from segmented underground pipe images. The proposed algorithm employs a fuzzy membership function and a projection neural network. The former absorbs variation of feature values and the latter shows good learning efficiency.

5.2 Feature Extraction

Feature extraction is an important stage for any pattern recognition task especially for pipe defect classification, since pipe defects are highly variable and it is difficult to find reliable and robust features. According to the study in [167], trained operators mainly rely on five criteria in visual interpretation of images. These are intensity, texture, size, shape, and organization. The intensity corresponds to the spectral features, which can generally be extracted easily. Textural features are those characteristics such as smoothness, fineness, and coarseness or certain pattern associated with an image [15]. They reflect the local spatial distribution property in a certain region. The spectral and textural features are most widely used in automatic object classification. Other features such as size, shape and organization information attribute to the large scale or global spatial distribution.

Generally, two broad categories of object features are most commonly used in the material/pavement classification field [94]: shape and textural features. The first class of features, which plays a more important role for object classification, extracts the information based on the geometric shape of the object. Some of the most commonly used methods in this category include area, length, roundness, etc. The second category, i.e., textural features, distinguish objects by using statistical measures based on gray-scale co-occurrence matrix [67] and its variant, such as gray-scale difference vector, moment invariants, and gray-scale difference matrix. The salient features of the data can also be extracted through a mapping, such as Fourier transform, discrete cosine transform, Karhunen-Loeve transform, or principal component method [85], from a higher dimensional input space to a lower dimensional representation space.

Depending on the analyzed parameters or features of each object, the most suitable set of features that represents the characteristics of each object in the underground pipe images is selected. We have used information based on the geometric shape and size of the objects present in the underground pipe images for feature extraction. The advantages of the proposed extraction of geometrical features from the image are its capability to

quantify distress features in terms of understanding parameters (area, lengths, roundness, etc.) and its ability to classify the segmented image based on such quantities. These features constitute the input parameters for the classifier, and are discussed briefly in the following sub-sections.

5.2.1 Selection of Crack and Hole Features

In the present case for classification of severity of cracks and holes, if the attributes are selected to be the major/minor axis length, and area, then the classifier can be trained for classifying different objects based on their geometry. For example, if an object has a width (minor axis length) of a few millimeters and its length (major axis length) is much greater than its width, then the object can be classified as a crack. On the other hand, if the ratio of major and minor is close to one and its minor axis length is a few centimeters, then the object can be classified as a hole rather than a crack. We can also use the information of four direction projections to classify the cracks. An image projection in a direction is done by adding the pixels values along four directions: horizontal (0^0), vertical (90^0), and two diagonals (45^0 and 135^0). The idea of using image projections to classify crack types is based on: (1) if it is a longitudinal crack, there is a peak in the vertical projection; (2) if it is a transverse crack, there should be a peak in the horizontal projection; (3) if the crack is diagonal, there is a peak in the diagonal direction; and (4) if it is a mushroom crack, there are peaks in all four direction projections. The five features selected for classification of the type of the crack and hole in the underground pipe image are:

1. Area
2. Number of Objects
3. Major Axis Length
4. Minor Axis Length
5. Projection of pixels and then taking mean and variance in each of the four projected directions (0^0, 45^0, 90^0, and 135^0)

Each segmented crack/hole image is to be classified into one of the following seven classes based on the extracted 12 feature vectors, which describes the existence and orientation of crack segments and severity of holes present in the image:

1. Transverse Crack
2. Longitudinal Crack
3. Diagonal Crack
4. Multiple Crack
5. Mushroom Crack
6. Minor Hole
7. Major Hole

5.2.2 Selection of Joint Features

To distinguish between different types of joints (i.e., perfect joint, eroded joint, and misaligned joint), we have selected five features based on the shape and size of the underground pipe joints. These features are:

1. Area
2. Number of objects
3. Elongation (ratio of major and minor axis length)
4. Extent (ratio of net area and bounding rectangle area)
5. Projection of pixels and then taking mean and variance in each of the horizontal and vertical projected directions (0^0, and 90^0)

An image of a segmented joint is to be classified into one of the following three classes based on the extracted eight feature vectors, which describes the severity of joint defects present in the image:

1. Perfect Joint
2. Eroded Joint
3. Misaligned Joint

5.2.3 Selection of Lateral Features

A major category of measurement parameters describes the shape of features. Since underground pipe laterals are more or less circular in shape, therefore shape features can be a key factor in being able to recognize or select different types of laterals (i.e., perfect lateral, eroded lateral, and collapsed lateral).

There are wide varieties of shape descriptors available. Probably the most widely used shape parameter is the 'form-factor' calculated as 4π Area / Perimeter^2. The form-factor is 1.0 for a perfect circle. Any other shape will have more perimeters for the same area, and the form-factor describes this increase. The other shape parameter is 'roundness' which looks very much like the form-factor at first glance. It is calculated as 4 * Area / π * Length^2, which again is just the formula for the area of a circle and gives a value of 1.0 for a perfect circle. Instead of the perimeter, roundness uses the length (longest chord) of the feature. This makes it more sensitive to how elongated the feature is, rather than how irregular its outline may be. Another set of shape parameters deal with how convex the feature is. For example, the 'aspect ratio' of an object, which we can calculate as the ratio of the measured maximum diameter to either the width or the minimum diameter. This is dimensionless, of course, and expresses a quality of feature shape that ignores local smoothness of surfaces, but provides a measure of how elongated the feature is. Based on the shape parameters, we have selected five feature based on the shape and size of the underground pipe joints. These features are:

1. Area
2. Number of Objects
3. Roundness
4. Form-factor
5. Aspect Ratio

An image with segmented lateral is to be classified into one of the following three classes based on the extracted five feature vectors, which describes the severity of lateral defects present in the image:

1. Perfect Lateral
2. Eroded Lateral
3. Collapsed Lateral

5.3 Classification of Objects

Pattern recognition can be generally defined as the allocation of objects to classes so that individual objects in one class are as similar as possible to each other and as different as possible from objects in other classes. Considered as a pattern recognition problem, there have been numerous techniques investigated for classification. Clearly, the more *a priori* information that is known about the problem domain, the more the classification algorithm can be made to reflect the actual situation. For example, if the *a priori* probabilities and the state conditional densities of all classes are known, then Bayes decision theory produces optimal results in the sense that it minimizes the expected misclassification rate [47]. However, in many pattern recognition problems, the classification of an input pattern is based on data where the respective sample sizes of each class are small and possibly not representative of the actual probability distributions, even if they are known. In these cases, many techniques rely on some notion of similarity or distance in feature space, for instance, clustering and discriminant analysis [40]. Under many circumstances, the K-nearest neighbour algorithm (K-NN) [40] is used to perform the classification. One of the problems encountered in using the K-NN classifier is that normally each of the sample vectors is considered equally important in the assignment of the class label to the input vector. This frequently causes difficulty in those places where the sample sets overlap.

Fuzzy sets were introduced by Zadeh in 1965 [196]. Since that time, researchers have found numerous ways to utilize this theory to generalize existing techniques and to develop new algorithms in pattern recognition and decision analysis [19,65,85,92]. In [196] Bezdek suggests that interesting and useful algorithms could result from the allocation of fuzzy class membership to the input vector, thus affording fuzzy decisions

based on fuzzy labels. This work is concerned with incorporating fuzzy set methods into the classical K-NN decision rule. In particular, a 'fuzzy K-NN' algorithm has been developed utilizing fuzzy class memberships of the sample sets and thus producing a fuzzy classification rule.

Recently, there has been a great resurgence of research in neural network classifiers [27,98,142,148,155]. New and improved neural network models have been proposed, models that can be successfully trained to classify complex data. Neural networks models have an advantage over the statistical methods in that they are distribution-free and no prior knowledge is needed about the statistical distributions of the classes in the data sources in order to apply these methods for classification. On the other hand, neural network models can be very complex computationally, need many training samples to be applied successfully, and their iterative training procedures usually are slow to converge. In addition, neural network models have more difficulty than model-based statistical methods in classifying patterns that are not identical to one or more of the training patterns. The performance of the neural network models in classification is therefore more dependent on having representative training samples, whereas the statistical approaches need to have an appropriate model of each class.

Given the potential advantages of neural networks over statistical methods for classification, the purpose of research in this chapter is to determine empirically how well these methods perform as classifiers for classification of underground pipe objects. The statistical and neural classifiers are evaluated by comparing their performance on the classification of extracted feature vectors by the severity of distress present in the pipe images. The data set used for the evaluation of the classifiers is generated from previously segmented underground pipe images. The actual classification of each image is determined by human visual observation (Ontario Pipeline Inspectors). Two data sets are generated; one is used as a training-data set, and the other is used as a test-data set. The training-data set is used to train each of the classifiers and the test-data set is used to evaluate the performance of each classifier in terms of how accurately it is to reproduce the actual image classifications in the test-data set.

5.3.1 Proposed Neuro-Fuzzy Classifier

The benefits of the neural network is the generalization ability about the untrained samples due to the massively parallel interconnections and the ease of implementation simply by training with samples for any complicated rule or mapping problem. The utility of fuzzy sets lies in their ability to model the uncertain or ambiguous data so often encountered in real life. Therefore, to enable a system to take care of real life situations in a manner more like humans, the concept of fuzzy sets has been incorporated into the neural network. It is also important for the neural networks to have fast learning ability in order to adapt to the new environment. Based on this consideration, a combination of two neural networks classifiers are examined in this thesis. The conventional backpropagation algorithm is improved by combining it with a restricted Coulomb energy network [148] to develop a new neural network model called the projection neural network.

The Projection Neural Network

The standard backpropagation training algorithm [155,187], while successful for problems of moderate size, suffers from slow training times, the potential to get stuck at local minima, and the need for a large number of hidden nodes when applied to complicated problems. However, in problems for which it does converge to a solution, it offers the advantage of ensuring error minimization. Therefore, when solving a classification problem, the network outputs will approach the Bayes conditional probabilities, given a statistically representative set of training data. On the other hand, there exist classification algorithms that train quickly but do not guarantee minimization of the classification error. Examples of these are the hypersphere classifiers, such as the restricted Coulomb energy network (RCE) [148], the models of adaptive resonance theory (ART) [27], and the Kohonen type networks [98]. In this study, we have used a projection network that combines the utility of both RCE [148] and backpropagation [187] approaches.

The classification algorithms that provide fast training do so by placing prototypes with closed decision boundaries around training data points and then adjusting their positions and/or sizes. As an example, a hypersphere classifier such as RCE places hyperspherical prototypes around training data points and adjusts their radii. Radial basis function networks can provide fast training as well as error minimization [128,140,142]. While several methods of determining the size, position and amplitude of the radial basis functions have been proposed they do not have the simplicity or computational efficiency of backpropagation training [139]. In contrast, the projection network [189] provides a means of implementing radial basis functions with a uniform approach to learning these parameters: backpropagation training of the weights and thresholds of a feedforward network [155]. This effectively leads to optimization of the prototypes' locations, size and amplitudes. Furthermore, both closed decision regions (hyperspheres or hyperellipses) and open ones (such as hyperplanes) are accommodated in the same network. Training of the network parameters may convert closed decisions regions to open ones and vice versa in the process of minimizing the error.

It is this ability to form closed prototypes with a single hidden node that allows the projection network to be initialized rapidly to a good starting point that is already close to a desirable error minimum. Any of a number of algorithms can be used for this initialization; Kohonen learning [98], RCE [148], and ART [27] are examples. Once the network has been initialized in this manner, modified backpropagation training is used to adjust the network weights and thresholds to ensure error minimization. Because the network begins near a good solution, one avoids the long training time which standard backpropagation would take to reach this point as well as the possibility of being stuck in local minima that might prevent one from reaching this point.

The projection network has the added attraction of modularity: One network can be trained to recognize inputs of a set of classes and another can be trained to recognize inputs of other classes or different members of the same classes. Then, the two networks can be combined into a single network. In general, some additional training after the combination will be required to optimize the network performance, but there is no need

to completely retrain the combined network, as would be generally required by conventional backpropagation network. This modularity is possible primarily because there is little or no interference between prototypes on the intermediate layer, particularly between prototypes of opposite classes, so that the addition of more prototypes does not necessarily destroy the signal to the output nodes.

The extension of a standard neural network to produce the projection network is a very simple one. The neural network inputs are projected onto a hypersphere in one higher dimension and the input and weight vectors are confined to lie on this hypersphere. A single hidden level node is now capable of forming either an open or a closed region in the original input space. This basic concept is not new. The need to normalize the input vector and the weight vector so that their dot product is a measure of their closeness has been recognized for a long time [98]. Telfer and Casasent [175] have used a projection onto a cylindrical hyperbola for initialization of a network with no hidden layers. Saffrey and Thornton [157] have applied stereographic projection to the Upstart algorithm. By projecting the input vector onto a hypersphere in one higher dimension, one can create prototype nodes with closed or open classification surfaces all within the framework of a backpropagation trained feedforward neural network. In this way, one achieves rapid prototype formation through initialization and subsequent optimization through backpropagation training. Figure 5-1 shows the typical structure of projection neural networks.

Classical models usually try to avoid vague, imprecise or uncertain information, because it s considered as having a negative influence in an inference process. Fuzzy systems [196] on the other hand deliberately make use of this kind of information. This usually leads to simpler, more suitable models, which are easier to handle and are more familiar to human thinking. A fuzzy neural network is a neural network that uses fuzzy methods to learn faster or to perform better. Neural networks have a learning capability and the fuzzy concepts can absorb variability in feature values. The fuzzy concept can be combined with neural networks in various ways.

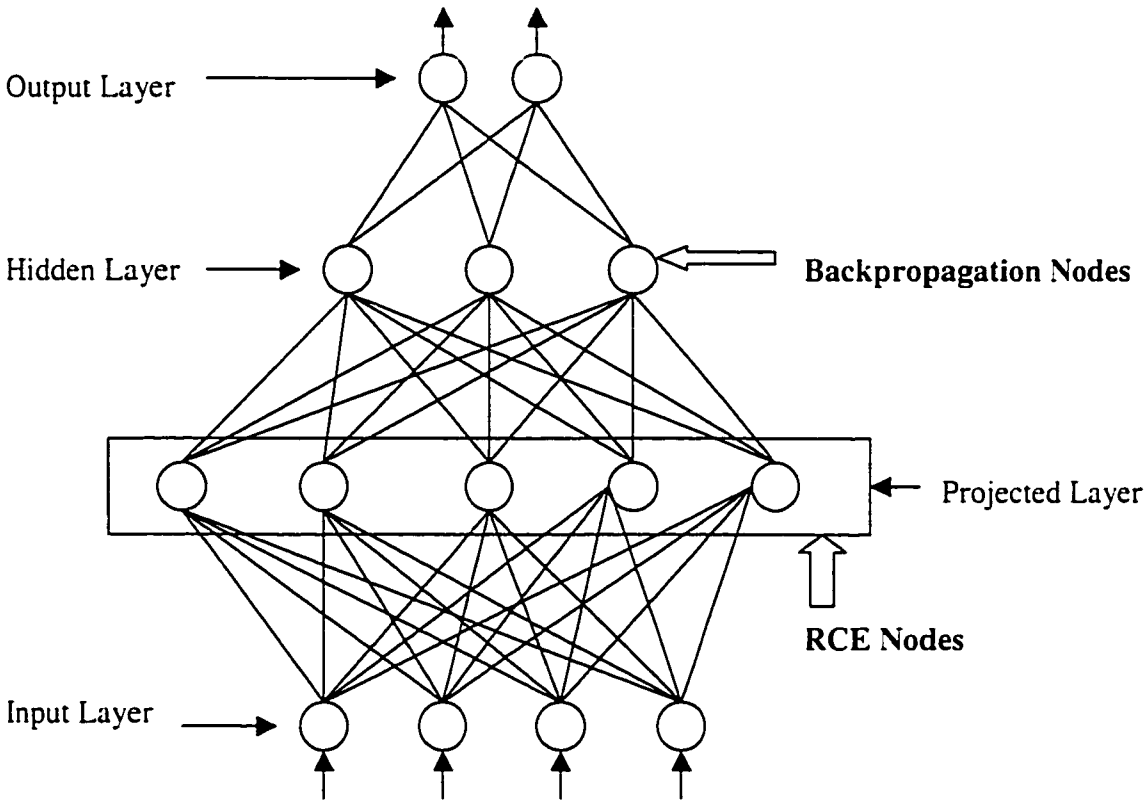


Figure 5-1. The projection neural network architecture showing combination of reduced Coulomb energy (RCE) network and backpropagation network.

The Neuro-Fuzzy Network

In general, there are two kinds of combinations between neural networks and fuzzy systems [110]. In the first approach neural network and fuzzy system work independently of each other. The combination lies in the determination of certain parameters of a fuzzy system by a neural network, or a neural network-learning algorithm. This can be done offline, or online during the use of the fuzzy system. The second kind of combination defines a homogenous architecture, usually similar to the structure of a neural network. This can be done by interpreting a fuzzy system as a special kind of neural network, or by implementing a fuzzy system using neural network. Besides these models, there are approaches in which a neural network is used as a pre-processor or as a post-processor to a fuzzy system. Such combinations do not optimize a fuzzy system, but only aim to improve the performance of the combined system. Learning takes place in the neural network only; the fuzzy remains unchanged [110]. In this case, the improvement of neural network learning is the main intention. In this study, we apply the fuzzy concept simply in converting feature values into fuzzified data, which are input and output to the projection neural network algorithm.

Input Pattern Representation in Linguistic Form

In the proposed neuro-fuzzy algorithm, we use the fuzzy data as input to neural networks. Sometimes, variation of feature values is large, and then it is difficult to classify objects correctly based on these feature values. To solve this problem, we first convert each object feature value into three fuzzy data [135], and then learning is performed with these fuzzy data using the projection network. Finally, we classify objects using the proposed neuro-fuzzy algorithm..

There are several types of membership functions in representing fuzzy phenomena [196]. The proposed object classification algorithms are simulated using triangular, trapezoidal, and Gaussian membership functions. To convert normalized features into fuzzy data, we determine the *MAX* and *MIN* values that are the maximum and minimum feature values

for entire dataset, respectively. As shown in Figure 5-2 we generate three membership functions denoted by '*S*' (small), '*M*' (medium), and '*L*' (large). Note that these membership functions are specified by *MIN* and *MAX*, as shown in Figure 5-2. Then we compute three fuzzy data for each feature value and use these data as the input data to neural networks. In Figure 5-2 $\mu_S(x_i)$, $\mu_M(x_i)$, and $\mu_L(x_i)$ are three fuzzy data of an input feature value (x_i), corresponding to linguistic variables of '*S*', '*M*', and '*L*', respectively. In the underground pipe object classification method using three membership functions (i.e., triangular, trapezoidal, and Gaussian), the extracted features are represented by means of linguistic variables specified by these membership functions.

The trapezoidal, triangular, and Gaussian membership function, as shown in Figures 5-2, 5-3, and 5-4, respectively, are located at the average value of features of the same image, and have a maximum value of one over the limited range that is specified by the standard deviation of the feature value. To generate a linguistic variable we first compute the average and standard deviation of the feature values of the image. Then we uniformly divide the interval between *MIN* and *MAX* into several subintervals, where *MIN* and *MAX* represents the minimum and maximum of average values of the specific feature, respectively. The membership function of each image is centered at the average value of the features of the image. Variation of feature values for the same image is allowed by employing the trapezoidal membership function, i.e. the width at the top of the trapezoidal and Gaussian membership function is set to σ_i, where σ_i denotes the standard deviation of the i^{th} feature value. Note that for input data greater (smaller) than *MAX* (*MIN*) we clip the membership value to 1 (0). These membership function values represent more than 500 images stored in a database for neural network training and testing. The training scheme for the fuzzy input neural network with trapezoidal fuzzy membership function is shown in Figure 5-5.

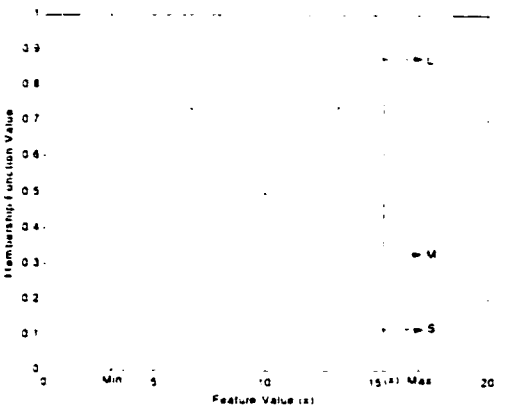


Figure 5-2. Linguistic representation of feature values by trapezoidal membership function

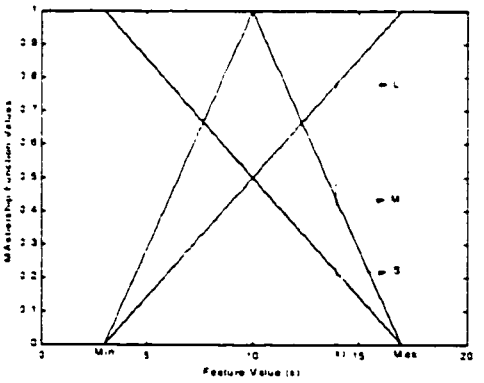


Figure 5-3. Linguistic representation of feature values by triangular membership function

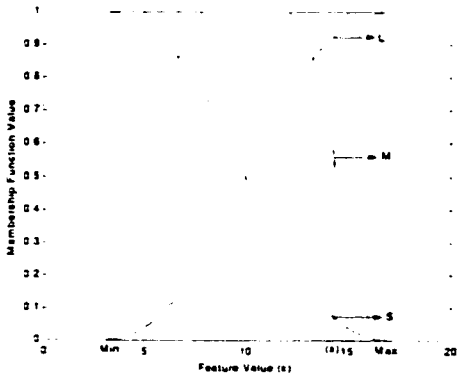


Figure 5-4. Linguistic representation of feature values by Gaussian membership function

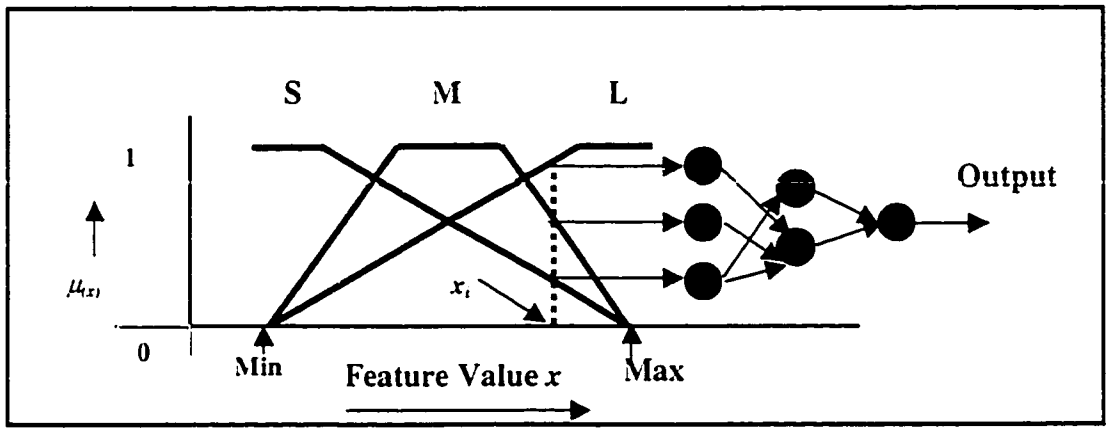


Figure 5-5. The neuro-fuzzy neural network architecture with fuzzy inputs

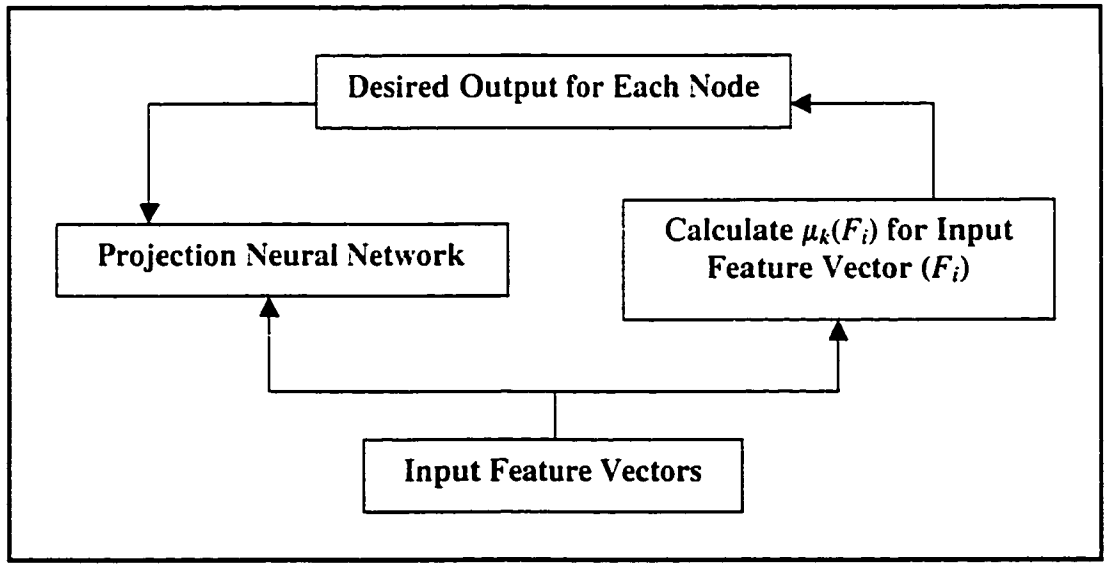


Figure 5-6. The training for the neuro-fuzzy neural network with fuzzy outputs

Output Class Representation in Linguistic Form

In this approach, the conventional neural network is manipulated only on the output layer with the fuzzy desired output. In general, the neural network passes through two phases, training and testing. During the training phase, supervised learning is used to assign the output membership values ranging in [0,1] to the training input vectors. Each output from the network may be assigned with a nonzero membership instead of choosing the single node with the highest activation. It allows modeling of fuzzy data when the feature space involves overlapping pattern classes, such that a pattern point may belong to more than one class with a nonzero membership. During training, each error in membership assignment is fed back and the connection weights of the network are appropriately updated. The backpropagated error is computed with respect to each desired output, which is a membership value denoting the degree of belongingness of the input vector to a certain class. The testing phase in a fuzzy network is equivalent to the conventional network.

In the case of an m-class problem with an n-dimensional feature space, let the n-dimensional vectors O_{kj} and V_{kj} denote the mean and the standard deviation for j^{th} input feature respectively of the numerical training data for the k^{th} class. The weighted distance, Z_{ik}, of the training pattern vector $\underset{F_i}{\rightharpoonup}$ from the k^{th} class is defined as

$$Z_{ik} = \sqrt{\sum_{j=1}^{n}\left[\frac{F_{ij} - O_{kj}}{V_{kj}}\right]^2} \quad \text{for } k = 1,\ldots,m \text{ and } j = 1,\ldots,n, \tag{5.1}$$

where F_{ij} is the value of the j^{th} input feature component of the i^{th} pattern point. The weight $1/V_{kj}$ is used to take care of the variance of the classes so that a feature with higher variance has less significance in characterizing a class. The membership of the i^{th} pattern to class C_k is defined as follows:

$$\mu_k(\underset{F_i}{\rightharpoonup}) = \left(\frac{Z_{ik} - \min\limits_{k}(Z_{ik})}{\max\limits_{k}(Z_{ik}) - \min\limits_{k}(Z_{ik})}\right) \quad \text{for } k = 1,\ldots,m. \tag{5.2}$$

Obviously $\mu_i(\underset{\sim}{F}_j)$ lies in the interval [0,1]. Except for the fuzzy membership desired values in the output layer, the training method and network structure is equivalent to the conventional neural network classifier. The training scheme for fuzzy output neural network is shown in Figure 5-6.

Fuzzy Input and Output Module and Neural Network Module

The concept of the proposed fuzzy input and output module and neural network module is illustrated in Figure 5-7. The fuzzy ANN model has three modules: the fuzzy input module, the neural network module, and the fuzzy output module. The neural network module is a conventional feed-forward artificial neural network. A simple projection network is used in this study. To increase the rate of convergence, a momentum term and a modified backpropagation training rule are used. The input layer of this network consists of $3I$ nodes (because of the use of fuzzy sets to screen the I input feature variables), and the output layer consists of C nodes (trained with fuzzy output C class values). As shown in Figure 5-7, the input layer of this fuzzy ANN model is actually an output of the input module. On the other hand, the output layer becomes an input to the output module. The input and output modules, for pre-processing and post-processing purposes, respectively, are designed to deal with the data of the ANN using fuzzy sets theory.

5.3.2 Performance Comparison with Other Classifiers

To study the performance of the proposed neuro-fuzzy classifier and to compare its performance with that of other statistical and neural classifiers, we have used K-NN [40], fuzzy K-NN [95], and conventional backpropagation network classifiers. The theoretical background of these classifiers is presented and their relative advantages are discussed.

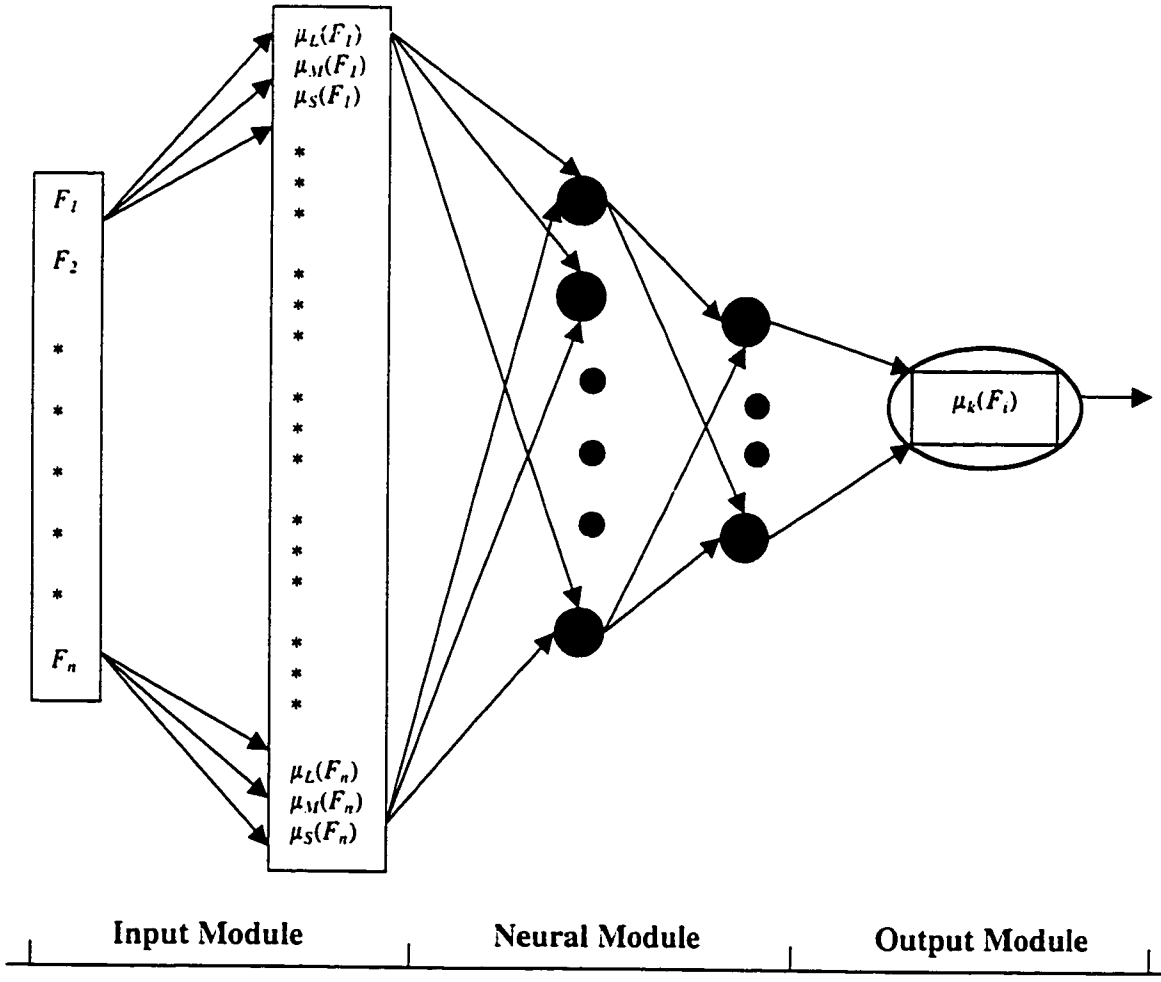


Figure 5-7. The proposed neuro-fuzzy neural network architecture with fuzzy inputs and outputs to projection neural network

The *K*-NN Algorithm

Many classification methods assume that the form of class-condition densities is known. The popular maximum likelihood estimation [47] approach assumes multivariate normality. The *K*-nearest neighbor (*K*-NN) [40] procedure is a nonparametric classification procedure. This rule classifies a new feature vector y by assigning it the label most frequently represented among the *K*-nearest of all training samples [40]. The decision is made by determining the majority class represented in the set of *K*-nearest neighbors of a pattern by examining the labels of each of the *K* neighbors. Randomization is used for breaking ties. In practice, one chooses $K=c\sqrt{n}$ where c is an appropriate constant and n is the size of the training set. In the present study, $c=1$ is used.

The Fuzzy *K*-NN Algorithm

The fuzzy *K*-NN algorithm is considered one of the most accurate algorithms in pattern recognition [95]. The classical (crisp) *K*-NN algorithm classification rule assigns an input sample vector y, which is of unknown classification, to the class that is represented by a majority amongst its *K*-nearest neighbors [47]. The *K*-nearest neighbors are chosen from a labeled data sample (data of known classification). The fuzzy *K*-NN algorithm assigns class membership to a sample observation based on the observation distance from its *K*-nearest neighbors and their memberships [95].

If $W=\{x_1, x_2, \ldots, x_n\}$ is the set of n labeled samples and u_{ij} is the membership of the j^{th} labeled data in the i^{th} class, then the fuzzy *K*-NN algorithm is simply described as follows [95].

Begin

Input y, of unknown classification.

Set K, $1 \leq K \leq n$

Initialize $i=1$

Do Until (*K*-nearest neighbours found)

Compute distance from y to x_i

If $(i \leq K)$ Then

Include x_i in the set of K-nearest neighbors

Else if (x_i is closer to y than any previous nearest neighbour) Then

Delete farthest in the set of K-nearest neighbours

Include x_i in the set of K-nearest neighbours

End If

Increment i

End Do Until

Initialize $i = 1$

Do Until (y assigned membership in all classes)

$$\text{Compute } u_i(y) = \frac{\sum_{j=1}^{K} u_{ij} \left(\frac{1}{\left\| y - x_j \right\|^{\frac{2}{(m-1)}}} \right)}{\sum_{j=1}^{K} \left(\frac{1}{\left\| y - x_j \right\|^{\frac{2}{(m-1)}}} \right)} \tag{5.3}$$

Increment i

End Do Until

End

As shown in (5.3), the assigned memberships of observations are influenced by the class memberships of the K-nearest neighbours. The memberships of the labeled sample can be assigned in several ways such as using fuzzy cluster analysis or based on expert opinions. The distance between observations can be represented by any distance measure such as the Euclidean distance, defined as [19]

$$d_{yx_i} = \sum_{v=1}^{p} (y_v - x_{iv})^2 = (y - x_i)'(y - x_i) \tag{5.4}$$

where p = number of variables for observation i. With this distance, the variables are given equal weights. The variable m in (5.3) defines how heavily the distance is weighted when calculating each neighbour's contribution to the membership value [19].

The Conventional Backpropagation Network

Several neural network models can be used in pattern classification (both supervised and unsupervised). For supervised pattern classification, the most commonly used ANN is the feed-forward network trained using the back-propagation algorithm [155], which is adopted in the present study. The backpropagation algorithm can be described in three equations. First, weight connections are changed in each learning step (k) with

$$\Delta w_{ij(k)}^{[s]} = \eta(t)\delta_{pj}^{[s]} x_j^{[s-1]} + m\Delta w_{ij(k-1)}^{[s]} \tag{5.5}$$

Second, for output nodes it holds that

$$\delta_{pj}^{[m]} = (d_j - o_j) f_j'(I_j^{[s]}) \tag{5.6}$$

and third, for the remaining nodes it holds that

$$\delta_{pj}^{[s]} = f_j'(I_j^{[s]}) \sum_k \delta_{pk}^{[s-1]} w_{jk}^{[s-1]} \tag{5.7}$$

where $x_j^{[s]}$ = actual output of node j in layer s; $w_{ij}^{[s]}$ = weight of the connection between node l at layer (s-l) and node j at layer (s); $\delta_{pj}^{[s]}$ =measure for the actual error of node j; $I_j^{[s]}$ = weighted sum of the inputs of node j in layer s; $\eta(t)$ = time-dependent learning rate; $f(\ )$ = transfer function; m = momentum factor (between 0 and 1); and d_j, o_j = desired and actual activity of node j (for output nodes only). The architecture of conventional backpropagation network is shown in Figure 5-8.

Parameter values (i.e. the learning rate $\eta(t)$, momentum factor m, and the number of hidden nodes h_j) are selected experimentally to be those that gave the best classification accuracy. The input and output nodes are selected according to the feature vectors and class of objects to be classified. Experimental results from all the above methods are presented in the following section.

5.4 Experimental Results and Evaluation

Five classifiers have been implemented to classify objects in the segmented underground pipe images based on the extracted feature vectors. To allow for comparison between the five classification methods, results are presented to show the difference in the magnitude of classification accuracy compared to expert classification. The overall classification accuracy percentage for each classifier is calculated by constructing a confusion matrix between expert's decision and classifier result. Table 5-1 shows the classification results in terms of the confusion matrix for the proposed neuro-fuzzy algorithm with Gaussian membership function and using crack/hole feature vectors.

In Table 5-2, we show the overall classification results of the fuzzy approach with different membership functions. The overall classification accuracy for each classifier and class is calculated in terms of the confusion matrix as discussed previously and shown in Table 5-1. We can observe that the fuzzy input and output network with Gaussian membership function has better classification rate. We have selected this network as best performing neuro-fuzzy network for comparison with other classifiers.

Table 5-3 shows the results of comparison for five classification methods. In general, both the ANN classifier and neuro-fuzzy classifier performed better and produced more consistent results than the *K*-NN and fuzzy *K*-NN classifier. As shown from the comparison of results, the overall performance of the proposed neuro-fuzzy model is better than that of other classifiers. Although there is a slight improvement in classification rate by using the projection network as compared to the conventional backpropagation network. However, the projection network learned much faster and required fewer nodes in the hidden layer. Figure 5-9 shows the convergence characteristics of the projection and backpropagation neural networks.

The results of both neural and neuro-fuzzy classifiers however are considered good. One of the most important attributes of both classifiers is their ability to spot patterns in data that classical pattern recognition systems may not be able to detect. Therefore,

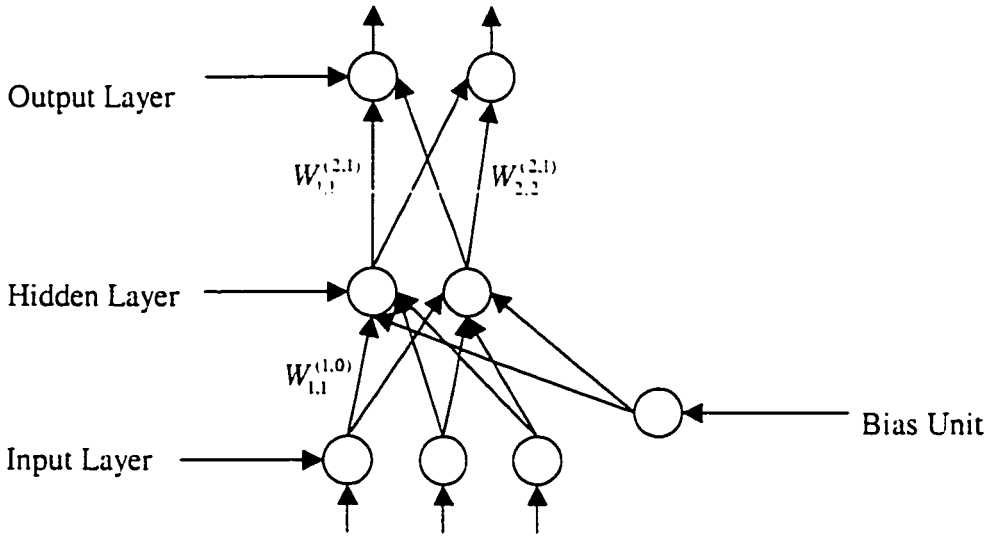


Figure 5-8. The typical structure of backpropagation neural networks

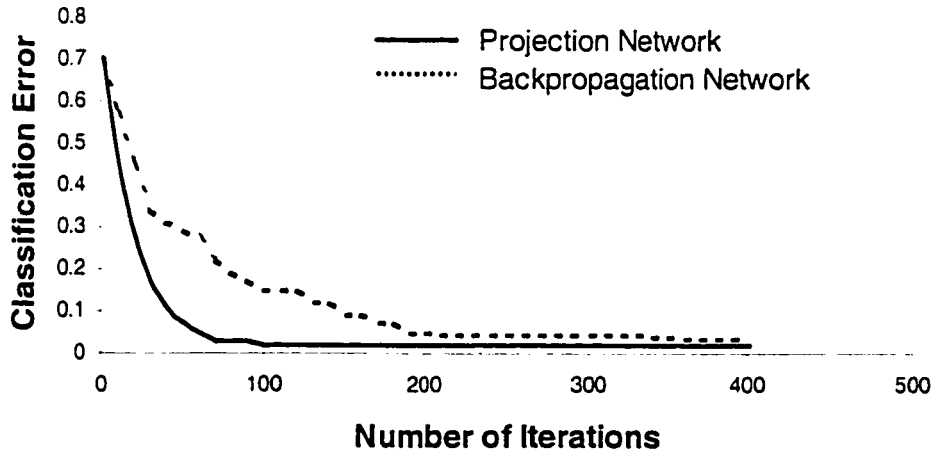


Figure 5-9. Comparison of convergence characteristics of the projection and backpropagation neural networks

Table 5-1. Confusion matrix for the proposed neuro-fuzzy classifier using a Gaussian membership function for classification of crack/hole defects.

Class		Trans. Crack	Long. Crack	Diag. Crack	Mult. Crack	Mush. Crack	Minor Holes	Major Holes	Total
		1	2	3	4	5	6	7	
Trans. Crack	1	**43**	0	2	0	0	0	0	45
Long. Crack	2	0	**33**	1	1	0	0	0	35
Diag. Crack	3	1	1	**18**	0	0	0	0	20
Mult. Crack	4	0	0	0	**58**	2	0	0	60
Mush. Crack	5	0	0	0	2	**13**	0	0	15
Minor Holes	6	0	0	0	0	0	**53**	2	55
Major Holes	7	0	0	0	0	0	3	**22**	25
Total									255

Table 5-2. Classification accuracy by fuzzy neural network with different membership functions.

Classifier	Fuzzy Input	Membership Function	Fuzzy Output	Classification Accuracy (%)		
				Crack/Hole	Joint	Lateral
1	Yes	Triangular	No	91	83.2	87.1
2			Yes	91.7	84.5	87.3
3		Trapezoidal	No	92.3	86.2	88.2
4			Yes	93.2	87.5	88.7
5		Gaussian	No	92.9	87.7	89.9
6			Yes	94.1	88.2	91.8
7	No	No	Yes	90.3	82.8	86.8

Table 5-3. Comparison of the performance evaluation of the proposed neuro-fuzzy classifier with that of four other classifiers

Classifier	Classification Methods	Classification Accuracy (%)		
		Crack/Hole	Joint	Lateral
1	K-NN	81	76.8	80.3
2	Fuzzy K-NN	84.6	79.3	82.9
3	Backpropagation Network	87.2	81.6	84.5
4	Projection Network	89.5	82.1	86
5	Neuro-Fuzzy Network	94.1	88.2	91.8

both classifiers are recognized as ideal tools for dealing with environments that are highly unstructured and that may involve incomplete or noisy data (such as underground pipe images). For a large design data set, the performance of K-NN and fuzzy K-NN should normally be good. However, there are two main drawbacks with these classifiers. It is necessary to store and keep all the training data in computer memory. In addition, during the classification process, each vector to be classified has to be compared to all the vectors in the training data set for classification. This results in a very computationally intensive procedure, which is a significant disadvantage for real-time application.

5.5 Conclusions

In this Chapter, we proposed a neuro-fuzzy classifier that combines neural networks and fuzzy concepts for the classification of objects in segmented underground pipe images. Fuzzy sets are used in the input module as well as in the output module to 'screen' data patterns before network training. With this technique, the proposed network can be trained with greater efficiency. In the feature extraction step, we extract different features of the object present in the segmented image based on the geometric shape and size. These features values are then fuzzified and applied to the neuro-fuzzy network in the classification step.

We show simulation results of the proposed neuro-fuzzy algorithm in comparison to the K-NN method, Fuzzy K-NN method, conventional backpropagation algorithm, and projection network. Simulation results show that the proposed neuro-fuzzy algorithm using a combination of Gaussian membership functions and projection neural networks gives better classification results than the other statistical and neural methods. The results show the promise of the proposed fuzzy-neural network as a tool for classifying objects in the segmented underground pipe images based on extracted feature vectors.

Chapter 6

6. Integrated Pipeline Management

In Chapters 3, 4, and 5 we discussed the development of an automated pipeline inspection system based on digital image processing and artificial intelligence methodologies. An automated condition assessment capability motivates the synthesis of an automated system with a robust predictive model for the management of underground pipelines. This chapter lays the groundwork for development of an integrated pipeline management system, and is organized as follows: Section 6.2 reviews pipeline network identification and classification; Section 6.3 presents a development of pipeline condition rating system; Section 6.4 details the construction of pipeline performance prediction and rehabilitation models. Conclusions of this study are presented in Section 6.5.

6.1 Introduction

A systematic approach for the determination of deterioration of pipeline systems and an integrated management system are necessary to fully understand the complete status of a municipal underground pipeline system. A well-defined integrated pipeline management system includes, but is not limited to, routine and systematic pipe structural and hydraulic

condition assessments, the establishment of a standard condition rating system, the development and updating of prediction models for pipeline performance, life-cycle cost analysis, and the development of prioritization schemes for the selection of rehabilitation options. An automated inspection system and a sound predictive model are key elements in the management of underground pipelines.

This chapter discusses the major aspects of integrated management for pipeline systems, namely, network identification, condition rating systems, condition prediction modeling, and the use of optimization techniques for maximizing benefit/cost ratio over a planning horizon. Figure 6-1 shows the major components of the proposed integrated pipeline management system.

The proposed integrated pipeline management system provides guidelines in the decision-making processes, because it uses optimization techniques to obtain minimum cost of maintenance/rehabilitation strategies over the life cycle of pipe systems. The current condition of pipes will be evaluated by the proposed automated system and the Markovian model in this life-cycle approach will predict future conditions of the pipe system. By taking into account future conditions as a consequence of present rehabilitation action, the best alternative, as well as the priority and schedule of rehabilitation, can be determined.

6.2 Pipeline Network Identification and Classification

The first step in setting up a pipeline management system is to 'divide' the pipeline system into separate smaller networks so that each network can be stored in a single database to provide efficient data entry and report generation. Some of the factors to consider when identifying different networks are use (sanitary, storm, combined, water, etc.), location/zone, and any other distinctive criteria (funding source, required condition, etc.). An example of a smaller network for separate inspection and analysis in the city of Waterloo's sewer system is combined sewers of 91 cm (36in.) diameter or smaller. The characteristics of this network have been recorded in a database including its condition

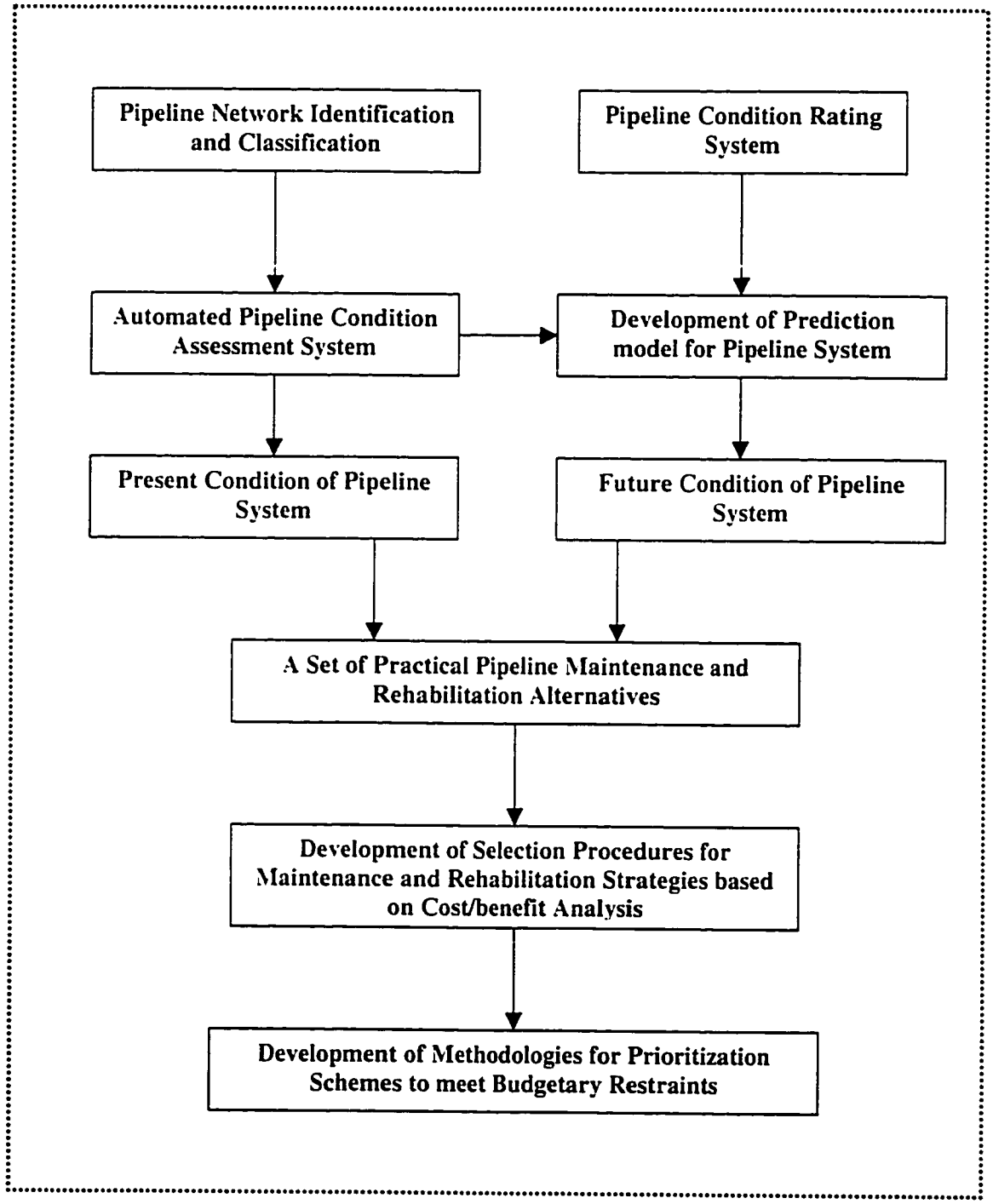


Figure 6-1. Framework for the Integrated Pipeline Management System

assessment, which was based on the detection and classification of defects by a proposed automated inspection system. Once the networks are identified, they are defined by 'pipe runs' and 'pipe segments.' The city of Waterloo refers to a 'sewer run' as a manageable group of sewer segments sorted from an upstream manhole to the furthest downstream manhole of the basin number. A 'sewer segment' is viewed as the smallest management unit when considering the application and selection of rehabilitation strategies. Pipelines are then classified into different categories to account for differences in material, diameter, etc., and to account for changes in condition characteristics and deterioration mechanisms.

6.3 Pipeline Condition Rating System

The goal of a condition rating system is to objectively rate, by means of a scoring system, the current condition of the pipeline. In the field of pavement management, the Pavement Condition Index (PCI) developed by the U. S. Army Corps of Engineers [121] has received wide acceptance and has been formally adopted as standard procedure by many highway agencies [88]. The PCI is a numerical index, ranging from 0 for a failed pavement to 100 for a pavement in excellent condition. The degree of pavement deterioration is a function of distress type, distress severity, and amount of density of distress. Assigning one index that considers the three factors is difficult, so 'deduct values' are introduced as a type of weighting factor to account for effects caused by each combination of distress type, severity level, and distress density. Based on in-depth knowledge of pavement behavior, input from many experienced pavement engineers, field testing and evaluation of the procedure, and accurate descriptions of distress type, severity levels and their corresponding deduct values were derived to develop a composite distress index, namely, the PCI [88].

No such standard procedure has been developed for municipal pipeline systems. A key reason could be a lack of linkages between the pipeline systems maintained and managed by the different municipalities, unlike the manner of the management of highways by the different states. While standard procedures for developing comprehensive municipal

pipeline condition ratings (similar to PCI) do not exist, some rating systems have been developed by certain municipalities in North America [2]. In the city of Toronto, the sewer condition is rated based on general defect criteria including crack patterns (hairline, transverse, longitudinal, or mushroom), joint conditions (eroded, or misaligned), service lateral conditions (eroded, or collapsed), and structural defects (sagging, collapsing, or crushed). For pipe surface defects the rating factors range from 0 to 9. Observance of a collapse pipe is rated 0, while intact pipe surface is rated 9, as shown in Table 6-1. Similarly, for joints and laterals the rating factors range from 0 to 4, as shown in Tables 6-2 and 6-3, respectively. No attempt has been made at this point to combine the information from pipe inspection with any applicable internal factors of the pipe (e.g., defective surface, joints, and lateral) and external factors (e.g., soil type, pipe material, and impact factor) to adjust the pipe condition rating.

6.4 Performance Prediction and Rehabilitation Model

The collapse and distress of a pipe is the result of complex interactions of various mechanisms that occur within and around a pipeline. Pipelines are prone to certain types of failures based on their type of material, physical design, age or functionality as well as its external and internal environment. The impact of the deterioration of the pipeline system depends upon its size, complexity, topography and service. A comprehensive study performed by the Water Research Center (WRC) [2] concluded that the concept of measuring the 'rate of deterioration' of municipal pipeline is unrealistic since deterioration is more influenced by random events in a municipal pipe life span and severe defects do not always lead immediately to collapse. However, while it may be impossible to predict when a municipal pipe will collapse, it is feasible to estimate whether a pipe has deteriorated sufficiently for collapse to be likely.

Assessment and deterioration models for pavements and bridges [1,25,87,191] have been the primary focus for research in infrastructure systems. Prediction models currently used for predicting the performance of infrastructure systems (in particular, pavement and bridge systems) include straight-line extrapolation, regression techniques, and the

Table 6-1. Pipe surface condition index and structural condition matrix for underground pipe surface deterioration

REINFORCED CONCRETE PIPE STRUCTURAL CONDITION MATRIX													
Pipe Surface Condition Index	Probability of pipe surface condition going from one state to another after five years of inspection period	Final Condition	Intact Pipe Surface	Hairline Crack	Minor Crack	Multiple Crack	Mushroom Crack	Minor Fracture	Major Fracture	Minor Hole	Major Hole	Pipe Collapse	SUM OF PROBABILITY
	Initial Condition		1	2	3	4	5	6	7	8	9	10	
9	Intact Pipe Surface	1											1
8	Hairline Crack	2											1
7	Minor Crack	3											1
6	Multiple Crack	4											1
5	Mushroom Crack	5											1
4	Minor Fracture	6											1
3	Major Fracture	7											1
2	Minor Hole	8											1
1	Major Hole	9											1
0	Pipe Collapse	10											1

Table 6-2. Pipe joint condition index and structural condition matrix for development of Markov prediction model for joint deterioration

REINFORCED CONCRETE PIPE STRUCTURAL CONDITION MATRIX								
Pipe Joint Condition Index	Probability of pipe joint condition going from one state to another after five years of inspection period	Final Condition	Intact Pipe Joint	Minor Joint Defect	Multiple Joint Defect	Major Joint Defect	Joint Collapse	SUM OF PROBABILITY
	Initial Condition		1	2	3	4	5	
4	Intact Pipe Joint	1						1
3	Minor Joint Defect	2						1
2	Multiple Joint Defect	3						1
1	Major Joint Defect	4						1
0	Joint Collapse	5						1

Table 6-3. Pipe lateral condition index and structural condition matrix for development of Markov prediction model for lateral deterioration

Pipe Lateral Condition Index	Probability of pipe joint condition going from one state to another after five years of inspection period	Final Condition	Intact Pipe Lateral	Minor Lateral Defect	Multiple Lateral Defect	Major Lateral Defect	Lateral Collapse	SUM OF PROBABILITY
	Initial Condition		1	2	3	4	5	
4	Intact Pipe Lateral	1						1
3	Minor Lateral Defect	2						1
2	Multiple Lateral Defect	3						1
1	Major Lateral Defect	4						1
0	Lateral Collapse	5						1

probability based Markov model. Straight-line extrapolation tends to produce unrealistic results, while regression techniques are valid only if the predictive variables can be found that are related to condition deterioration [1]. This thesis has followed the philosophy of previous researchers and discusses a management model similar to that used in the management of pavements. A strategic maintenance and rehabilitation plan or program that is based on projected need is required to ensure continuous and effective operation of the entire pipeline system at the lowest cost. Life cycle management of the pipeline network is necessary to obtain the maximum cost effective life span.

6.4.1 Probability based Markovian Prediction Model

Markovian models [154] provide a reliable mechanism for developing prediction models. Markov chains can be employed to model stochastic processes, which have the distinct property that probabilities involving how the process will evolve in the future depend only on the present state of the process and so are independent of events in the past. The model is restricted to Markov chains that have (1) a finite number of states; and (2) stationary transition probabilities (i.e., transition probabilities that do not change in time).

The Markov process imposes a rational structure on the deterioration model because it explains the rate of deterioration as uncertain, and it also ensures that the projections beyond the limits of data will continue to have a worsening condition pattern with time. This model has been successfully used in other types of infrastructure deterioration modeling. A Markov chain approach analogous to that which has been developed for pavement deterioration modeling [2] can be applied to pipeline systems. To model the manner in which a pipe deteriorates with time, it is necessary to establish a Markov probability transition matrix. A pipe is modeled to begin its life in near perfect condition, and to deteriorate as it is subjected to a sequence of duty cycles (time) t. A state vector indicates the probability of a pipe section being in each pf pipe condition states in any given year. Figure 6-2 shows the schematic of state, state vector, and duty cycle. The transition matrix P is a square matrix, m x m, where m is the number of possible states. Thus, if there are five categories in pipe conditions, then five possible states will be

	Intact Surface	9		1		Po (1)											Po(1) =
	Hairline Crack	8		2		Po (2)											Probability
	Minor Crack	7		3		Po (3)											of being in
Pipe	Multiple Crack	6		4		Po (4)											State 1 at
Condition	MushroomCrack	5	State	5	State	Po (5)											Duty
Index	Minor Fracture	4		6	Vector	Po (6)											Cycle = 0
	Major Fracture	3		7		Po (7)											
	Minor Hole	2		8		Po (8)											
	Major Hole	1		9		Po (9)											
	Pipe Collapse	0		10		Po (10)											
Duty Cycle (Age in Years)						0	1	2	3	4	5	6	7	8	9	10	

Figure 6-2. The schematic representation of state, state vector, and duty cycle

involved in the matrix of size 5x5. The components of P, namely p_{ij}, are the probabilities of being in state i at time 0 and transitioning to state j over a given period Δt. A time increment, Δt, of 5 years is suitable because municipal pipe inspections should generally be conducted every 5 years. If the assumption is accepted that the pipe condition will not drop by more than one state in any 5-year period, then the condition will either stay in its current state or move to the next lower state in 5-year period. Therefore, the one-step transition matrix can be represented as follows:

$$P = \begin{bmatrix} p_{11} & p_{12} & 0 & 0 & 0 \\ 0 & p_{22} & p_{23} & 0 & 0 \\ 0 & 0 & p_{33} & p_{34} & 0 \\ 0 & 0 & 0 & p_{44} & p_{45} \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{6.1}$$

For each row of the transition matrix, $\sum_j p_{ij} = 1$. The value of 1 in the last row indicates an 'absorbing' state corresponding to the fact that the pipe condition cannot move from this state (the worst possible state) unless rehabilitation is performed. In this particular transition matrix, the values of four unknown quantities (i.e., p_{11}, p_{22}, p_{33}, and p_{44}) have to be determined. The probability that the pipe is in state i at time t and will be in state j

after n periods is desired. The Chapman-Kolmogorov equations provide a method for computing the n-step transition probabilities, and the n-step transition probability matrix can be obtained by computing the n^{th} power of the one-step transition matrix [154]. Thus, if the one-step transition matrix P corresponds to a 5 years time period, then the two-step (10 year time period) transition matrix $P^{(2)}$ is represented by

$$P^{(2)} = P^2 = P \times P \tag{6.2}$$

Besides the transition matrix P, the state matrix X representing the probability distribution of being in m different states at time 0 (which is the fraction of pipe network currently in each of the m possible states) is also required. X is a single row matrix (or state vector) where $\sum X_i = 1$ for $i = 1, \ldots, m$. The state vector for any time cycle t is obtained by multiplying the initial state vector by the transition matrix P raised to the power t. Thus, the prediction of pipe condition 10 years from now is then represented by $X^{(2)}$.

$$[X^{(2)}] = [X] \times [P^{(2)}] \tag{6.3}$$

To estimate the transition matrix probabilities, for each age group the following nonlinear programming objective function was formulated:

$$\min \sum_{t=1}^{N} |S(t) - E(t, P)| \tag{6.4}$$

subject to $0 \le p(i) \le 1, \quad i = 1, 2, \ldots, 5$

where,

$N = 5$, the number of years in one age group;

$P = [p(1), p(2), \ldots, p(5)]$, a vector of length 5;

$S(t)$ = average of condition ratings at time t; and

$E(t, P)$ = estimated value of condition rating by Markov chain at time t.

The objective function was to minimize the absolute distance between the actual pipe condition rating at a certain age and the predicted pipe condition for the corresponding age generated by the Markov chain with the probabilities obtained by the nonlinear programming. The solution to this function was obtained by the gradient projection

method [87]. To find the trend of a performance curve, a polynomial regression procedure was performed first. The results of the regression were taken as the average condition ratings to solve the nonlinear programming.

A survey was developed, and municipal management experts were asked for their opinions on the degradation of sewer pipes. A detailed questionnaire was mailed to 65 municipalities in North America, and 40 responses were received. The condition of a pipe was described through words and pictures to these experts, and they were asked the probability that the condition of a sewer would change from the current (or initial) condition to a more severe condition in a period of 5 years. The collective experience of these participating experts was used to generate the structural condition matrix (SCM) and eventually to develop a prediction model for sewer deterioration. The SCM survey sheets for pipe surface, joint and lateral are shown in Tables 6-1, 6-2, and 6-3, respectively.

An example set of computations for underground concrete sewer pipe surface condition performance evaluation is given in the following. Pipe age was divided into groups, and within each age group the Markov chain was assumed to be homogeneous. Groups consisting of 5 years were used, and each group had its own transition matrix, which was different from those of the remaining groups. Using equation (6.1), the transition matrix for Group 1 was obtained:

$$P = \begin{vmatrix} 0.89 & 0.11 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.93 & 0.07 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.91 & 0.09 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.87 & 0.13 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.61 & 0.39 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.84 & 0.16 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.79 & 0.21 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.71 & 0.29 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.69 & 0.31 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{vmatrix} \quad (6.5)$$

The initial state vector of Group 1 was $p^{(0)}$ = (1,0,0,...,0). Therefore, the state vector of Group 1 for year t can be obtained by equation (6.3). For example, the state vectors for year 0 through 5 are given below:

$p^{(0)}$ = (1,0,0,0,0,0,0,0,0,0)
$p^{(1)} = p^{(0)} * P$ = (0.89,0.11,0,0,0,0,0,0,0,0)
$p^{(2)} = p^{(0)} * P^2$ = (0.7921,0.2002,0.0077,0,0,0,0,0,0,0)
$p^{(3)} = p^{(0)} * P^3$ = (0.7050,0.2733,0.0210,0.0007,0,0,0,0,0,0)
$p^{(4)} = p^{(0)} * P^4$ = (0.6274,0.3317,0.0383,0.0025,0.0001,0,0,0,0,0)
$p^{(5)} = p^{(0)} * P^5$ = (0.5584,0.3775, 0.0580,0.0056,0.0004,0,0,0,0,0)

Then $p^{(5)}$ obtained above for Group 1 was taken as the initial state vector of Group 2, and the corresponding transition matrix of Group 2 was used to continue the procedure. By this procedure, the pipe condition at any time t can be predicted in terms of initial state vector $p^{(0)}$ and transition matrix P. Figure 6-3 shows the pipe surface performance curve of concrete sewer pipe in city of Waterloo obtained by this method. Performance curves can be developed similarly for other pipe components (i.e., joints and laterals).

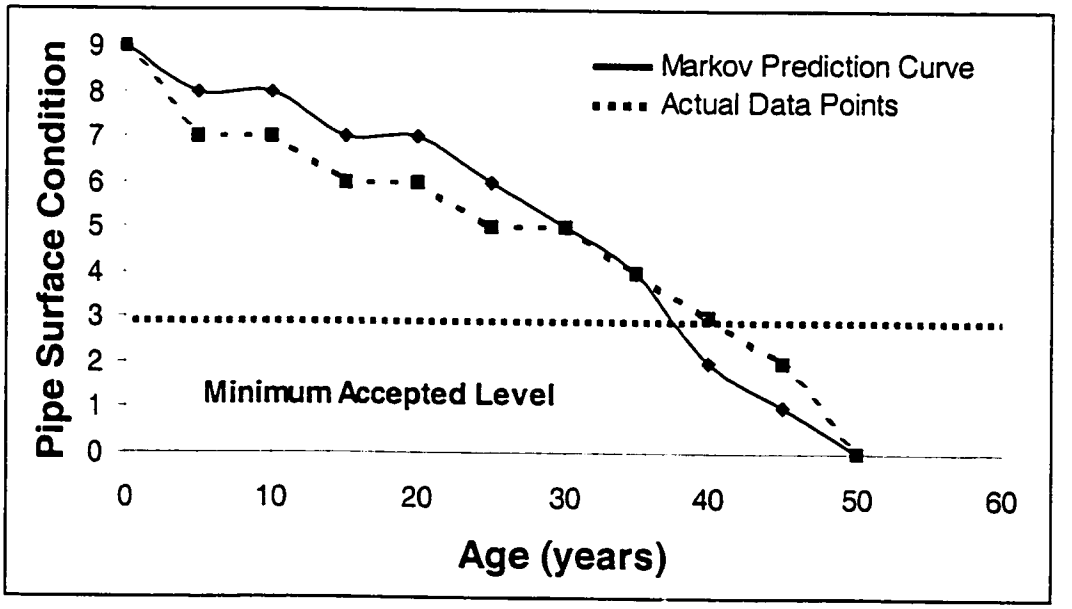


Figure 6-3. Markov prediction curve for concrete sewer pipe surface deterioration

6.4.2 Selection of Underground Pipeline Rehabilitation Options

The general decision process concerning the selection of the appropriate rehabilitation method for underground pipelines follows the logical sequence of assessment, decision, and execution. These three steps are essential if the appropriate method is to be selected (Figure 6-4). The choice of rehabilitation selection methods is related to the technical level of a municipality, available resources and environmental concerns.

In order to select appropriate treatments for pipeline long-term preservation, it is necessary to evaluate each treatment effects in terms of functional and structural improvement over the existing pipeline. It should be noted that a pipeline performance prediction model should be modified after each maintenance or rehabilitation treatment is applied to the pipeline except for the do-nothing treatment. Each rehabilitation treatment should be designed by considering the pipeline deterioration characteristics, treatment effects and the impacts of the treatment on the future deterioration or rehabilitation needs. Therefore, it is imperative to prepare a set of rehabilitation alternatives for the optimization model.

There is a need for a decision-making model that can help municipal engineers understand the essential aspects of the rehabilitation treatment and help identify the strategic elements of the performance of the treatment. The use of utility theory in evaluation models is well established [190]; however, such models require the establishment of the user's utility functions. Often, utility functions are hard to formulate and can change over time. The analytical hierarchy process [169] uses a three-level hierarchy-based model that reflects the goals and concerns of the decision maker and uses a series of criteria to evaluate rehabilitation alternatives. The comparison of the attributes of competing methods is based on determining the eigenvector for each matrix describing the attributes of each method. This method is excellent for the comparison of attributes between two methods but can become unwieldy when a large number of methods need to be evaluated, each with many specific attributes. A recent effort [8] uses a hierarchically based model that breaks the elements of the methods under consideration into their

essential physical components that describes their capabilities, and compares these capabilities to the requirements of the problem.

The rehabilitation method selection process as described in [123], is the logical matching of needs to method capabilities. This system breaks the needs and capabilities into a three-level hierarchy of attribute, function, and capacity, which uniquely identifies the characteristics of both the system needs and method capabilities (Figure 6-5). Using the characteristics of the system, it is possible to codify the system needs in a logical manner that will clearly identify the specific needs of that system. This process allows the needs of the system to be uniquely codified using the hierarchy of attribute/function/capacity. The codification of the system needs and method capabilities allows identification of potential candidate rehabilitation methods.

While installation issues, impact on hydraulics, and material and durability issues are key factors in the selection process, a hierarchy-based model does not explicitly include life cycle cost issues associated with each alternative. The process does not attempt to predict future pipe conditions, and accordingly life-cycle analysis is not performed. To select a final or most appropriate method, a more detailed examination of the problems and methods is required, including an economic analysis.

6.4.3 Optimizing Pipeline Rehabilitation Options

An approach that predicts the future condition of pipes is needed to perform life-cycle analysis for different pipe rehabilitation options. Probabilistic Markovian model similar to that shown in Figure 6-3 can be incorporated with dynamic programming for life cycle cost analysis of pipeline systems.

Dynamic programming can be employed in conjunction with a Markov probability based prediction model to obtain minimum cost maintenance and rehabilitation strategies over a given life cycle analysis period. Dynamic programming is an 'approach' to optimization, based on the principle of taking a single, complex problem and breaking it into a number

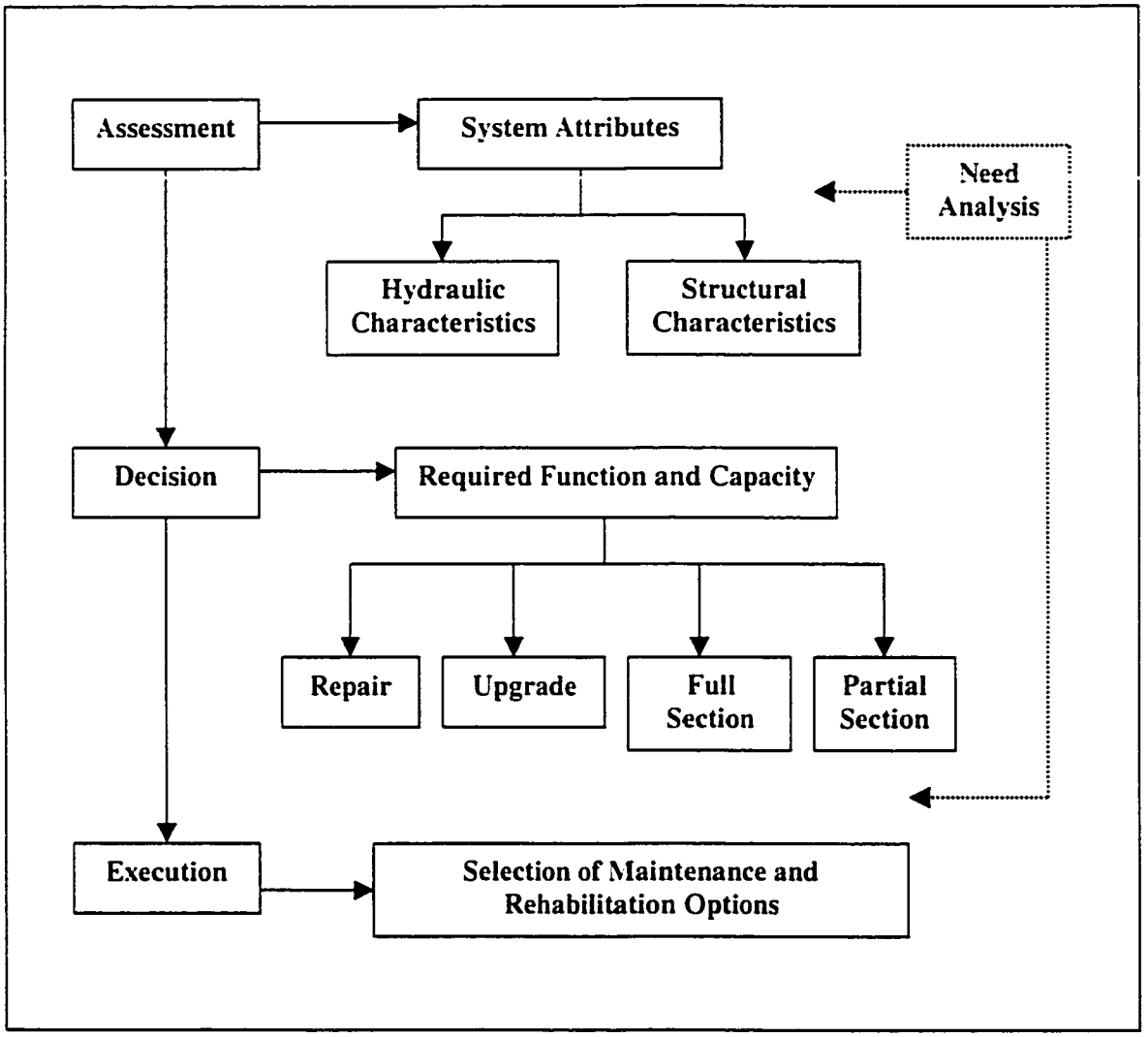


Figure 6-4. General decision process of selection of maintenance and rehabilitation methods for underground pipeline

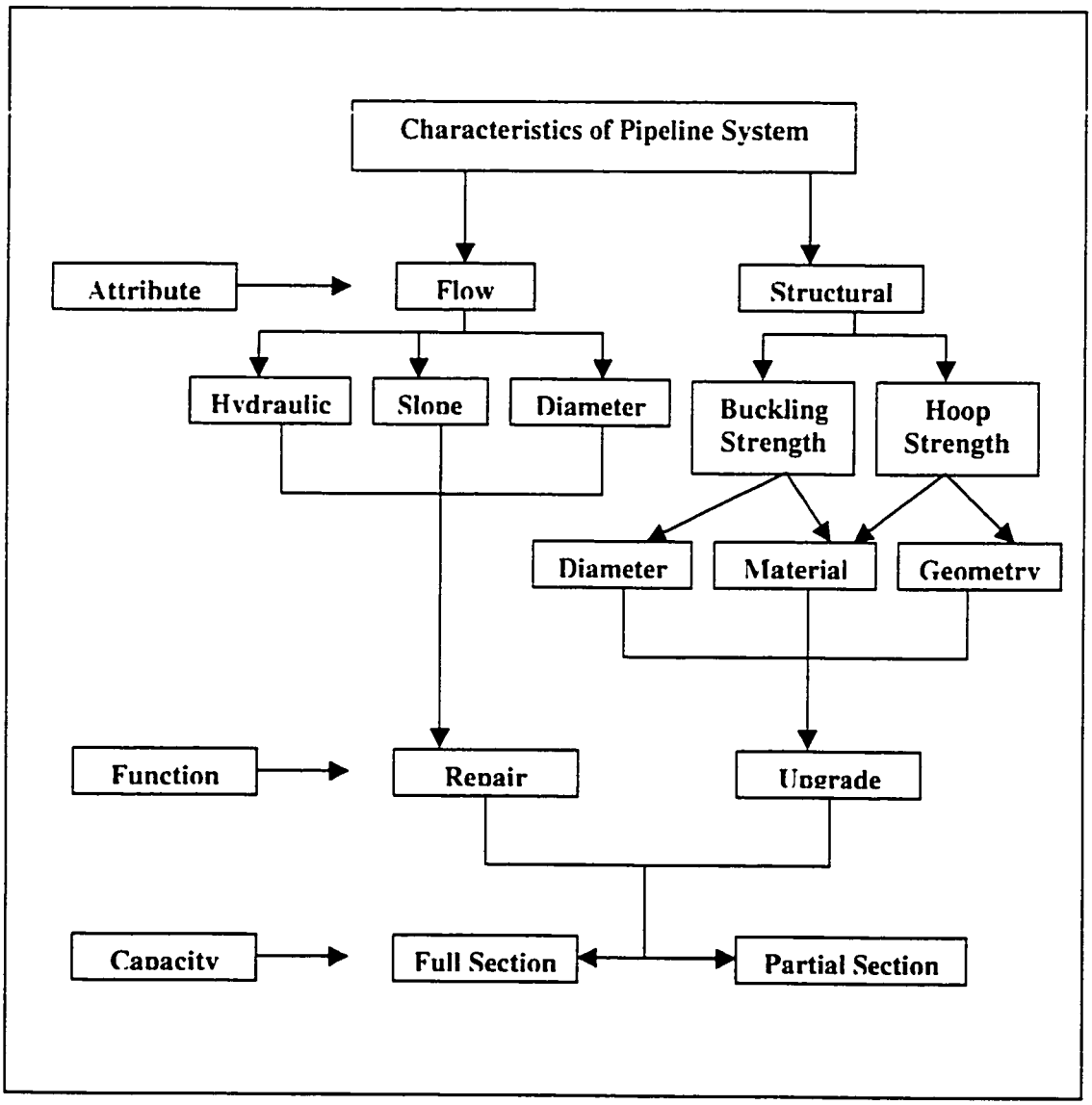


Figure 6-5. Characterization of system and rehabilitation methods for underground pipeline

of simpler and more easily solvable problems [51]. An extremely important advantage that dynamic programming has over almost all classical optimization techniques is that it will, if set up correctly, determine absolute optima rather than local optima [51].

For a given planning horizon, the proposed dynamic programming technique provides suggestions to decision makers by optimizing total cost (or maximizing benefit/cost ratio) of pipeline system. The prediction models are Markov processes, and the results from the Markov models are fitted into the multi-year dynamic priority programming model and the output from the priority programming is a list of optimal maintenance and rehabilitation recommendation during the analysis period for each pipeline section in the network. The prioritization uses cost-effectiveness based economic analysis to utilize the limited budget with a set of maintenance and rehabilitation strategies.

6.5 Conclusions

This chapter discusses a conceptual framework for the development of an integrated pipeline network management system. The probability-based model provides a reliable method to characterize the uncertainty inherent in underground pipeline deterioration. This model can be incorporated with dynamic programming as an optimization technique to maximize the benefit/cost ratio. If the aggregate cost over all pipeline segments within a network is higher than the available budget, then a prioritization scheme needs to be developed.

Numerous options are available for underground pipeline rehabilitation. Some techniques are more cost effective than others for specific applications. The number and method of rehabilitation options considered should be selected based on the pipeline deterioration characteristics, treatment effects and the impacts of the treatment on the future deterioration or rehabilitation needs.

The prioritization technique is useful for municipal managers when they are face with the need to optimize resources (in this case, limited budgets). However, the successful application of an integrated pipeline management system depends significantly on the predictive models. As both the computer vision technology and the predictive modeling technology advance it is likely that a synthesis of these two systems will lead to an integrated and automated pipeline management system whereby the input of present state conditions to the predictive model will be generated by using computer vision techniques.

Chapter 7

7. Contributions and Future Research

This final chapter reviews the contribution of this thesis and discusses possible avenues for continued research.

7.1 Thesis Contributions

The main objectives of this thesis were to develop an automated underground pipe inspection system and to establish the synthesis of an automated system with a predictive model for management of underground pipelines. The main concentration of this thesis has been on segmenting, detecting, and classifying different objects present in underground pipe images based on their class and severity of distress. The system is expected to overcome some of the limitations of the current manual inspection of underground pipes, and can provide a more accurate assessment of pipe conditions.

Chapter 3 presented a simple, robust, and efficient image segmentation algorithms for the automated analysis of scanned underground pipe images. The algorithm consists of a gray-scale conversion step followed by a sequence of morphological operations to

accurately segment pipe cracks, holes, joints, laterals, and collapse sections, a crucial step in the precise detection of surface cracks. The proposed morphological segmentation approach can be completely automated and has been tested on over five hundred scanned images from major cities in North America. Experimental results demonstrate that the proposed approach is effective for segmenting underground pipe images with varying background pattern, non-uniform illumination, and objects of different shape and size.

Chapter 4 proposed a development of crack detection filters, taking the statistical properties of underground pipe images. Conventional edge detectors and thresholding techniques are insufficient for detecting cracks when applied to underground pipe images, as demonstrated in this chapter. The approach proposed in this chapter falls within the scope of the Bayesian framework. Since our aim is to detect the cracks present in an image, contextual knowledge on the scale of pixels is insufficient and results in numerous, small, disconnected segments. However, on the scale of segments, *a priori* knowledge allows for the precise detection of cracks. Thus, detection of cracks is proposed in two steps. In the first step, crack segment candidates are detected taking the statistical properties of image into account. In the second, cracks are obtained by cleaning and linking operations.

In this chapter a methodology for the evaluation of the proposed crack detection filters based on the comparison to manually plotted reference data is also presented. In most of the cases the crack detection results are close to those that could be obtained by a trained human operator.

Chapter 5 contributed a development of neuro-fuzzy classifier for classification of objects in the segmented underground pipe images based on the extracted feature vectors. Fuzzy sets are used in the input module as well as in the output module to 'screen' data pattern before training the proposed network. With this technique, the proposed network always yielded higher performance compared to four other classifiers, namely, *K*-NN, fuzzy *K*-NN, conventional backpropagation, and projection network.

This chapter also discussed the extraction of suitable features for training the classifiers based on the geometric shape and size of the objects present in the segmented pipe images. An extensive comparative study of the proposed neuro-fuzzy network with other four classifiers, infer that neuro-fuzzy network can be an used as ideal tool for classifying segmented underground pipe images based on suitable features.

Chapter 6 proposed a framework for an integrated pipeline network management. An attempt has been made in this chapter to synthesize the automated inspection system, discussed in Chapters 3, 4, and 5, with a robust prediction model for management of underground pipelines. The prediction model developed in this chapter is based on the Markovian models, and is incorporated with dynamic programming optimization techniques. The proposed integrated management system will provide guidelines in the decision-making processes, because it uses optimization techniques to obtain the minimum cost of maintenance/rehabilitation strategies over the life cycle of pipelines.

7.2 Topics for Future Research

There are many interesting directions for continued research. A few of the more promising directions, organized by topics, are listed in the following subsections.

7.2.5 Color Image Segmentation

The use of color in image processing is motivated by two principal factors. First, in automated image analysis, color is a powerful descriptor that often simplifies object identification and extraction from a scene. Second, in image analysis performed by human beings, the motivation for color is that the human eye can discern thousands of color shades and intensities, compared to about only two-dozen shades of gray.

In this thesis, we have converted the colored pipe images into gray-scale before performing any analysis for computational simplicity. We believe that the color information in the image will probably help with classifying cracks, roots, water infiltration, holes, joints, and laterals.

7.2.2 Segmentation of Cluttered Images

The morphological segmentation approach proposed in Chapter 3 is mainly based on the images belonging to one class of object. In a real problem, images may not contain just one type of defect, but a clutter of defects. To segment such images we have proposed a new method based on gray-scale morphological operations and taking the difference of images. The use of this new method for segmenting cluttered pipe images is a promising approach that may need further practical validation.

7.2.3 Selection of Features for Classification

Over the last few years, extensive research has taken place on the development of efficient and reliable methods for the selection of features in the design of pattern classifiers, where the features constitute the inputs to the classifier. The quality of this design depends on the relevancy, discriminating power and ease of computation of various features. Selecting suitable features is an extremely difficult task, charged both with theoretical and computational problems.

In this thesis, we have selected features for classification based on the geometric shape and size of the objects present in the pipe images. One useful extension of our work would be to perform a thorough feature optimization to select an optimal set of features. This could significantly improve the classifier's performance.

7.2.4 Robust Dynamic Model for Prediction

The framework developed in this thesis for developing a prediction model and determining the optimal rehabilitation program for pipeline network appears to be satisfactory in its approach. However, there are still lots of problems to be solved for actual applications of these models as described in the following.

1. Although the probabilistic prediction model has conceptually been formulated for all types of underground pipelines, only concrete pipes have directly been considered in this study. Other types of pipes have not been taken into account in the research, due to time and resource limitations.

2. The relationship between observed pipe performance data and that predicted by the Markovian probabilistic prediction model should be examined for different site conditions for verification of the prediction accuracy.

3. Taking the expert's opinion generates the input of present state conditions to the probabilistic prediction model. It is recommended that the inputs should be generated by correlation of images obtained by inspecting pipe condition over the years.

4. In the present study, the effects of pipe age on pipe internal condition are emphasized. The effects of other external factors on pipe performance, such as soil types, pipe material, traffic load, hydraulic conditions, and climate are ignored. It is recommended that a dynamic model should consider both the internal and external factors effecting the pipe performance: the more factors considered by a model, the more realistic it can be.

5. The sensitivity analysis of the outputs of the dynamic programming models to changes in input parameters including discount rate, inflation rate, maintenance/rehabilitation costs, disruption costs, and expected service life is recommended so that asset managers can make decision with greater confidence.

7.2.5 Development of Software

It is fair to say that the algorithms used throughout the thesis have been written as a proof of concept, and therefore striving for efficient data structure was not the main focus. An immediate improvement would be to rewrite the code in a compiled language instead of the interpreted MATLAB language being used, and also to design new data structures to make it more efficient.

The development of a computer application software for this automated image analysis based, integrated pipeline management system is the final goal of this study. The engineering and model development for the software and its application has been considered at present. A Windows operating system based computer interface design for the application should be developed for the use in engineering departments of municipalities in North America.

Bibliography

[1] Abbas, A. B., Shahin, M. Y., Feighan, K. J., and Carpenter, S. H., 'Pavement Performance Prediction Model Using the Markov Process,' *Transp. Res. Record*, vol. 1123, Transportation Research Board, Washington, D.C., pp. 12-19, 1991.

[2] Abraham, D. M., Wirahadikusumah, R, Short, T. J., and Shahbahrami, S., 'Optimization Modeling for Sewer Network Management,' *J. of Constr. Engrg. and Mgmt.*, vol. 124(5), pp. 402-410, 1998.

[3] Abutaleb, A. S., 'Automatic Thresholding of Gray-level Pictures Using Two-dimensional Entropy,' *Computer Vision, Graphics, and Image Processing*, vol. 47, pp. 22-32, 1989.

[4] Acosta, J. A., Figueroa, J. L., and Mullen, R. L., 'Feasibility Study to Implement the Video Image Processing Technique for Evaluating Pavement Surface Distress in the State of Ohio,' *Proc. Automated Pavement Distress Seminar*, Federal Highway Administration, Ames, Iowa, 1992.

[5] Adeli, H., and Hung, S. L., 'Adaptive Conjugate Gradient Learning Algorithm for Efficient Training of Neural Networks,' *Appl. Math.*, vol. 62(1), pp. 81-102, 1994.

[6] Agin, G. J., 'Computer Vision Systems for Industrial Inspection and Assembly,' *J. of Computer*, pp. 11-20, 1980.

[7] Aleksander, I., '*Neural Computing Architectures: The Design of Brain-like Machines*,' MIT Press, Cambridge, MA, 1989.

[8] Alldritt, M., Russell, A., and El-Guindy, 'Developing a Knowledge Based System for the Selection of Trenchless Technology Methods,' *Proc. of INFRA Intl. Conf.*, CERIU, Montreal, Canada, pp. 451-464, 1996.

[9] American Society for Testing and Materials, '*ASTM C 597-83 – Concrete and Aggregates*,' American National Standards Institute, New York, 1995

[10] American Society for Testing and Materials, '*ASTM D 638-95 –Tensile Properties of Plastics*,' American National Standards Institute, New York, 1995

[11] Aydin, T., Yemez, Y., Anarim, E., and Sankur, B., 'Multidirectional and Multiscale Edge detection via M-band wavelet Transform,' *IEEE Trans. on Image Processing*, vol. 5, pp. 1370-1377, 1996.

[12] Azencott, R., '*Simulated Annealing: Parallelization Techniques*,' Wiley, New York, 1992.

[13] Baldi, P., and Chauvin, Y., 'Neural Networks for Fingerprint Recognition,' *Neural Computation*, vol. 5, pp. 402-418, 1993.

[14] Ballard, D. H., and Brown, C. M., '*Computer Vision*,' Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[15] Bankert, R. L., 'Cloud Classification of AVHRR Imagery in Maritime Using a Probabilistic Neural Network,' *J. Appl. Meterol.*, vol. 33, pp. 909-918, 1994.

[16] Bassim, M. N., and Houssny-Emam, M., '*Time and Frequency Analysis of Acoustic Emission Signals*,' Acoustic Emission, Gordon and Breach Scientific Publishers, pp. 139-163, 1983.

[17] Battiti, R., 'First and Second Order Methods for Learning: Between Steepest Descent and Newton's Method,' *Neural Computation*, vol. 4, pp. 141-166, 1992.

[18] Besl, P. J., Delp, E. J., and Jain, R., 'Automatic Visual Solder Joint Inspection,' *IEEE J. Robot. Automation*, vol. 1, pp. 42-56, 1985.

[19] Bezdek, J. C., '*Pattern Recognition with Fuzzy Objective Function Algorithms*, Plenum Press, New York, 1981.

[20] Birks, A. and Green, R., '*Nondestructive Testing Handbook*,' 2nd ed., vol. 7, American Society for Nondestructive Testing, Columbus, Ohio, USA, 1991.

[21] Bloch, I., 'Information Combination Operators for Data Fusion: A Comparitive Review with Classification,' *IEEE Trans. Syst. Man Cybern.*, vol. 26, pp. 52-67, 1996.

[22] Bribiesca, E., and Guzman, A., 'How to Describe Pure Form and How to Measure Differences in Shapes Using Shape Numbers,' *Pattern Recognition*, vol. 12(2), pp. 101-112, 1980.

[23] Brink, A. D., 'Gray-Level Thresholding of Images Using a Correlation Criterion,' *IEEE Pattern Recognition Letters*, vol. 9(5), pp. 335-341, 1989.

[24] Butt, A. A., Shahin, M.Y., Carpenter, S.H., and Carnahan, J.V., 'Application of Markov Process to Pavement Management Systems at Network Level, *Proc. 3rd Intl. Conf. on Managing Pavements*,' Transp. Res. Rec., National Research Council, Washington, D.C., pp. 85-96, 1994

[25] Butt, A.A. Shahin, M.Y. Feighan, K.J. and Carpenter, S.H. 'Pavement Performance Prediction Model Using the Markov Process,' *Transp. Res. Rec.*, 1123, National Research Council, Washington, D.C., 12-19, 1987.

[26] Canny, J. F., 'A Computational Approach to Edge Detection,' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8(6), pp. 679-698, 1986.

[27] Carpenter, G. A., and Grossberg, S., 'A Massively Parellel Architecture for a Self Organizing Neural Pattern Recognition Machine,' *Computer Vision, Graphics, and Image Processing*, vol. 37, pp. 54-115, 1987.

[28] Castleman, K. R., '*Digital Image Processing*,' Prentice-Hall, Inc., New Jersey, N.J., 1996.

[29] Chang, Y. L., and Li, X., 'Fast Image Region Growing,' *Image and Vision Computing*, vol. 13, pp. 559-571, 1995.

[30] Chen, K. B., Soetandio, S., and Lytton, R. L., 'Distress Identification by an Automatic Thresholding Technique,' *Proc. of Intl. Conf. On Application of Advanced Technologies in Transportation Engineering*, San Diego, 1989.

[31] Chen, S. Y., Lin, W. C., Chen, C. T., 'Split and Merge Image Segmentation based on Localized Feature Analysis and Statistical Tests,' *CVGIP-Graphics Models and Image Processing*, vol. 53(5), pp. 457-475, 1991.

[32] Cheng, H. D., 'Automated Real-Time Pavement Distress Detection Using Fuzzy Logic and Neural Networks,' *SPIE Proc. on Nondestructive Evaluation of Bridges and Highways*, pp. 140-151, 1996.

[33] Cheng, H. D., and Miyojim, M., 'Novel System for Automatic Pavement Distress Detection,' *J. of Computing in Civil Engrg.*, vol. 12(3), pp.145-152, 1998.

[34] Cheng, H. D., Chen, J. R., Glazier, C., and Hu, Y. G., 'Novel Approach to Pavement Cracking Detection Based on Fuzzy Set Theory,' *J. of Computing in Civil Engrg.*, vol. 13(4), pp.270-280, 1999.

[35] Cheng, J. C., and Don, H. S., '*Segmentation of Bilevel Images: A Morphological Approach*,' *Pattern Recognition: Architecture, Algorithm and Applications*, World Scientific Publishing Co., Singapore, 1991.

[36] Chin, R. T., and Harlow, C. A., 'Automated Visual Inspection: A Survey,' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 4, pp. 557-573, 1982.

[37] Chou, J. C., O'Neil, W. A., and Cheng, H. D., 'Pavement Distress Evaluation Using Fuzzy Logic and Moment Invariants,' *Transp. Res. Record*, vol. 1505, Transportation Research Board, Washington, D.C., pp. 39-46, 1995.

[38] Chow, C. K., and Kaneko, T., 'Automatic Boundary Detection of the Left Ventricle from Cineangiograms,' *Computers in Biomedical Research*, vol. 5, pp. 388-410, 1972.

[39] Cooper, L., and Cooper, M. W., '*Introduction to Dynamic Programming*,' Pergamon Press, Oxford, U.K., 1981.

[40] Cover, T. M., and Hart, P. E., 'Nearest Neighbor Pattern Classification,' *IEEE Trans. Inform. Theory*, vol. 13, pp. 21-27, 1967.

[41] Davies, T. G., and Mamlouk, M. S., 'Theoretical Response of Multilayer Pavement System to Dynamic Non-Destructive Testing,' *Transp. Res. Rec.*, vol. 1022, Transp. Res. Board, pp. 1-7, 1985.

[42] Davis, H. E., Troxell, G. E., and Hauck, G. F. W., '*The Testing of Engineering Materials*,' McGraw-Hill, Inc., Toronto, 1982.

[43] DeBoor, C. A., '*A Practical Guide to Splines*,' Springer Verlag, New York, 1978.

[44] Demigny, D., Lorca, F. G., and Kessal, L., 'Evaluation of Edge Detectors Performances with a Discrete Expression of Canny's Criteria,' *Proc. of Intl. Conf. on Image Processing*, IEEE, Los Alamitos, CA, pp. 169-172, 1995.

[45] Denardo, E. V., '*Dynamic Programming Models and Applications*,' Prentice-Hall, Englewood Cliffs, NJ, 1982.

[46] Dodwell, P., '*Visual Pattern Recognition*,' Holt, Rinehart, and Winston, New York, 1970.

[47] Duda, R. O., and Hart, P. E., '*Pattern Classification and Scene Analysis*,' Wiley Publication, New York, pp. 271-272, 1970.

[48] Edens, H. J., 'Budget Optimization for Urban Pavement Conference,' *Proc. of the First North American Conference on Managing Pavements*, Toronto, Canada, pp. 111-121, 1985.

[49] Farag, A., and Delp, E., 'Edge Linking by Sequential Search,' *Pattern Recognition*, vol. 28(5), pp. 611-633, 1995.

[50] Fausett, L., '*Fundamentals of Neural Networks*,' Prentice-Hall, Englewood Cliffs, NJ, 1994.

[51] Feighan, K. J., Shahin, M. Y., Sinha, K. C., and White, T. D., 'A Prioritization Scheme for the Micro PAVER Pavement Management System,' *Transp. Res. Rec.*, 1215, Transp. Res. Board, Washington, D.C., pp. 196-206, 1989.

[52] Fieguth, P. W., and Sinha, S. K., "Automated Analysis and Detection of Cracks in Underground Scanned Pipes," *Proc. '99 IEEE Image Processing Conf.*, Kobe, Japan, CD-ROM 28AP5A.2, 1999.

[53] Fischler, M. A., Tevenbaum, J. M., and Wolf, H. C., 'Detection of Roads and Linear Structures in Low Resolution Aerial Imagery Using Multisource Knowledge Integration Technique,' *Comput. Graph. Image Processing*, vol. 15, no. 3, pp. 201-223, 1981.

[54] Freeman, H., 'On the Encoding of Arbitrary Geometric Configuration,' *IRE Trans. on Electronic Computers*, vol. 10(2), pp. 260-268, 1961.

[55] Fu, K. S., '*Syntactic Pattern Recognition and Applications*,' Prentice-Hall, Englewood Cliffs, NJ, 1982.

[56] Fua, P., and Leclere, Y. G., 'Model Driven Edge Detection,' *Machine Vision Application*, vol. 3, pp. 45-56, 1990.

[57] Gauch, J., and Hsia, 'A Comparison of Three Color Image Segmentation Algorithms in Four Color Spaces,' *Proc. of the SPIE*, Bellingham, WA, vol. 1818, pp. 1168-1181, 1992.

[58] Geman, D., and Jedynak, B., 'An Active Testing Model for Tracking Roads in Satellite Images,' *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 1-14, 1996.

[59] Geman, D., Geman, S., Graffigne, C., and Dong, P., 'Boundary Detection by Constrained Optimization,' *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 609-628, 1990.

[60] Giardina, C. R., and Dougherty, E. R., '*Morphological Methods in Image and Signal Processing*,' Prentice-Hall, Englewood Cliffs, NJ, 1988.

[61] Gnanadesikan, R., '*Methods for Statistical Data Analysis of Multivariate Observations*,' Wiley, New York, 1977.

[62] Goldberg, D. E., '*Genetic Algorithms in search, Optimization, and Machine Learning*,' Addison-Wesley, Reading, MA, 1989.

[63] Gonzalez, R. C., and Wintz, P., '*Digital Image Processing*,' Addison-Wesley Publishing Co., Reading, Massachusetts, 1987.

[64] Grimson, W. E. L., and Lozano-Perez, T., 'Localizing Overlapping Parts by Searching the Interpretation Tree,' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9(4), pp. 469-482, 1987.

[65] Gupta, M., Ragade, R., and Yager, P., '*Advances in Fuzzy Set Theory and Applications*,' North Holland, Amsterdam, 1979.

[66] Haas, C., and Hendrickson, C., 'Computer-Based Model of Pavement Surfaces,' *Transp. Res. Rec.*, 1260, Transp. Res. Board, Washington, D.C., pp. 149-157, 1990.

[67] Haralick, R. M., 'Textural Features for Image Classification.' *IEEE Trans. Syst. Man Cybern.*, vol. 3, pp. 610-621, 1973.

[68] Haralick, R. M., and Shapiro, L. G., '*Computer and Robot Vision*,' vol. 1, Addison-Wesley, Reading, MA, 1992.

[69] Haralick, R. M., and Shapiro, L. G., 'Image Segmentation Techniques,' *Computer Vision, Graphics, and Image Processing*, vol. 29, pp. 100-132, 1985.

[70] Haralick, R. M., Katz, P. L., and Dougherty, E. R., 'Model-based Morphology: The Opening Spectrum,' *Graphical Models and Image Processing*, vol. 57(1), pp. 1-12, 1995.

[71] Haralick, R. M., Stenberg, S. R., and Zhuany, X., 'Image Analysis Using Mathematical Morphology,' *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9(4), pp. 532-550, 1987.

[72] Haykin, S., '*Neural Networks*,' Macmillan, New York, 1994.

[73] Hecht-Nielsen, R., '*Neurocomputing*,' Addison-Wesley, Reading, MA, 1990.

[74] Heijden, F. V. D., '*Image Based Measurement Systems*,' John Wiley and Sons, Inc., New York, 1994.

[75] Heijmans, H. T. A. M., '*Morphological Image Operators*,' Academic Press, Boston, M. A., 1994.

[76] Hellwich, O., Mayer, H., and Winkler, G., 'Detection of Lines in Synthetic Aperature Radar (SAR) Scenes,' *Proc. of Int. Archives Photogrammetry Remote Sensing (ISPRS)*, vol. 31, Vienna, pp. 312-320, 1996.

[77] Hibino, Y., Nomura, T., Ohta, S., and Yoshida, N., 'Laser Scanner for Tunnel Inspections,' *J. of Intl. Water Power and Dam Constr.*, vol. 6, pp. 17-28, 1994.

[78] Hillier, S.F. and Lieberman, G.J. '*Introduction to Operation Research*,' McGraw Hill Book Co. Inc., 2nd ed., New York, 1995.

[79] Hogg, D. C., 'Shape in Machine Vision,' *Image and Vision Computing*, vol. 11, pp. 309-316, 1993.

[80] Hopfield, J. J., and Tank, D. W., 'Neural Computation of Decisions in Optimization Problems,' *Biological Cybernetics*, vol. 52, pp. 141-152, 1985.

[81] Hott, F. B., Zaniewski, J., and Richards, M., 'Arizona Airport Pavement Management System,' *Proc. of 3rd Intl. Conf. on Managing Pavements*, San Antonio, vol. 2, pp. 207-216, 1994.

[82] Hunt, E., '*Artificial Intelligence*,' Academic Press, New York, 1975.

[83] Iseley, T., 'Pipeline Condition Assessment: Achieving Uniform Defect Ratings,' Proc. of Infra 99 Intl. Conf., vol. 1, pp. 121-132, 1999.

[84] Iseley, T., Abraham, D., M., and Gokhale, S., 'Intelligent Sewer Condition Evaluation Technologies,' *Proc. of Intl. NO-DIG Conf.*, pp. 254-265, 1997.

[85] Jain, A. K., '*Fundamentals of Digital Image Processing*,' Englewood Cliffs, NJ, 1989.

[86] Jain, A. K., and Chandrasekaran, B., '*Dimensionality and Sample Size Consideration in Pattern Recognition Practice*,' Handbook of Statistics, North Holland Publishing Co., 1982.

[87] Jiang, Y., Saito, M., and Sinha, K. C., 'Bridge Performance Prediction Model Using the Markov Chain,' *Transp. Res. Record*, vol. 1180, Transportation Research Board, Washington, D.C., pp. 39-46, 1992.

[88] Johnson, K. D., Cation, K. A., 'Performance Prediction Development Using Three Indexes for North Dakota Pavement Management System,' *Transp. Res. Rec.*, 1344, Transp. Res. Board, Washington, D.C., pp. 22-30, 1992.

[89] Joynson, R. E., McCary, R. O., Oliver, D. W., Silverstein-Hedengren, K. H., and Thumhart, L. L., 'Eddy Current Imaging of Surface Breaking Structures,' *IEEE Transactions on Magnetics*, vol. 22, pp. 11-24, 1986.

[90] Judd, J. S., '*Neural Network Design and the Complexity of Learning*,' MIT Press, Cambridge, MA, 1990.

[91] Kanal, L., and Chandrasekaran, B., 'On Dimensionality and Sample Size in Statistical Pattern Recognition,' *Pattern Recognition*, vol. 3, pp. 225-234, 1971.

[92] Kandel, A., '*Fuzzy Techniques in Pattern Recognition*,' John Wiley, New York, 1982.

[93] Karhunen, J., and Joutsensalo, J., 'Generalization of Principal Component Analysis, Optimization Problems and Neural Networks,' *Neural Networks*, vol. 8, pp. 549-562, 1995.

[94] Kaseko, M. S., and Ritchie, S. G., 'A Neural Network Based Methodology for Pavement Crack Detection and Classification,' *Transp. Research*, vol. 1(4), Transportation Research Board, Washington, D.C., pp. 275-291, 1993.

[95] Keller, J. M., Gray, M. R., and Givens, J. A., 'A Fuzzy K-nearest Neighbor Algorithm,' *IEEE Trans. Syst. Man Cybern.*, vol. 15, pp. 580-585, 1985.

[96] Kittler, J., and Illingworth, J., 'On Threshold Selection Using Clustering Criteria,' *IEEE Trans. on Systems, Man and Cybern.*, vol. 15(5), pp. 652-655, 1985.

[97] Kittler, J., Illingworth, J., Foglein, J., and Parker, K., 'An Automated Thresholding Algorithm and its Performance,' *Proc. 7th IEEE Intl. Conf. on Pattern Recognition*, pp. 287-289, 1984.

[98] Kohonen, T., '*Self Organizing Maps*,' Springer Verlag, New York, 1995.

[99] Koutsopoulos, H. N., and Downey, A. B., 'Primitive Based Classification of Pavement Cracking Images,' *J. Transp. Engrg.*, ASCE, vol. 119(3), pp. 402-418, 1993.

[100] Krstulovic, O. N., Woods, R. D., Al Shayea, N., 'Nondestructive Testing of Concrete Structures Using the Rayleigh Wave Dispersion Method,' *ACI Mat. J.*, vol. 93, pp. 75-86, 1996.

[101] Kulkarni, R. B., 'Dynamic decision Model for a Pavement Management System,' *Transp. Res. Rec.*, 997, National Research Council, Washington, D.C., pp.11-18, 1984.

[102] Kundu, A., and Mitra, S. K., 'A new Algorithm for Image Edge Extraction Using a Statistical Classifier Approach,' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 9(4), pp. 569-577, 1987.

[103] Kung, S. Y., and Diamantaras, K. I., 'A Neural Network Learning Algorithm for Adaptive Principal Component Extraction,' *IEEE Proc. of Intl. Conf. Acoustics, Speech, and Signal Processing*, Albuquerque, pp. 861-864, 1990.

[104] Law, T., Itoh, H., and Seki, H., 'Image Filtering, Edge Detection, and Edge Tracing Using Fuzzy Reasoning,' *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 481-491, 1996.

[105] Lee, H., and Deighton, R., 'Developing Infrastructure Management Systems for Small Public Agency,' *J. of Infrastructure Systems*, vol. 1(4), pp. 230-243, 1995.

[106] Li, L., Chan, P., Rao, A., and Lytton, R. L., 'Flexible Pavement Distress Evaluation Using Image Analysis,' *Proc. 2nd Intl. Conf on Applications of Advanced Technologies in Transp. Engrg.*, pp. 473-477, 1991.

[107] Li, N., Haas, R., and Xie, W., 'Multi-stage Dynamic Modeling Pavement Performance and Network Optimization of Highway Maintenance,' *Proc. of Intl. Road Fed. Asia-Pacific Regional Meeting*, Taipei, pp. 31-39, 1996.

[108] Li, X, and Zhiying, Z., 'Group Direction Difference Chain Codes for the Representation of the Border,' *Proc. of Digital and Optical Shape Representation and Pattern Recognition*, Orlando, FL, pp. 372-376, 1988.

[109] Lie, W. N., 'Automatic Target Segmentation by Locally Adaptive Image Thresholding,' *IEEE Trans. on Image Processing*, vol. 4, pp. 1036-1041, 1995.

[110] Lin, C. T., and Lee, C. S. G., '*Neuro-Fuzzy Systems*,' Prentice-Hall, Englewood Cliffs, NJ, 1996.

[111] Lopes, E. Nezry, Touz, R., and Laur, H., 'Structure Detection, and Statistical Adaptive Filtering in SAR Images,' *Intl. J. Remote Sensing*, vol. 14, no. 9, pp. 1735-1758, 1993.

[112] Loui, A. C. P., Venetsanopoulos, A. N., and Smith, K. C., 'Two-Dimensional Shape Representation Using Morphological Correlation Functions,' *IEEE Proc. of Intl. Conf. on Acoustics, Speech, and Signal Processing*, pp. 2165-2168, 1990.

[113] Lu, Y., and Madanat, S., 'Bayesian Updating of Infrastructure Deterioration Models,' *Transp. Res. Rec.*, 1442, Transp. Res. Board, Washington, D.C., pp. 110-114, 1994.

[114] Lytton, R. L., 'Moderator Report from Ranking to true Optimization,' *Proc. of the First North American Conference on Managing Pavements*, Toronto, Canada, pp. 3-17, 1985.

[115] Mandelbrot, B. B., '*The Fractal Geometry of Nature*,' Freeman, New York, 1982.

[116] Maragos, P., and Schafer, R. W., 'Morphological Filters – Part I: Their Set-theoretic Analysis and Relations to Linear Shift-invariant Filters,' *IEEE Trans on Acoustics, Speech and Signal Processing*, vol. 35(8), pp. 1153-1169, 1987.

[117] Maser, K. R., '*Computational Techniques for Automating Visual Inspection*,' Massachusetts Institute of Technology, Report, Cambridge, M. A., 1987.

[118] Matheron, G., '*Random Sets and Integral Geometry*,' Wiley, New York, 1975.

[119] McDonnell, M. J., 'Box Filtering Techniques,' *Computer Graphics and Image Processing*, vol. 17(3), pp. 65-70, 1981.

[120] McGlone, C., and Shufelt, J., 'Projective and Object Space Geometry for Monocular Building Extraction,' *Computer Vision and Pattern Recognition*, pp. 54-61, 1994.

[121] Mckay, D. T., Kevin, L. R., Greimann, L. F., and Stecker, J. H., 'Condition Index Assessment for U. S. Army Corps of Engineers Civil Works,' *J. of Infrastructure Systems*, vol. 5(2), pp. 52-60, 1999.

[122] McKenzie, D. S., and Protheroe, S. R., 'Curve Description Using the Inverse Hough Transform,' *Pattern Recognition*, vol. 23(3), pp. 283-290, 1990.

[123] McKim, R. A., 'Selection Method for Trenchless Technologies,' *J. of Infrastructure Systems*, vol. 3(3), pp. 119-125, 1997.

[124] Mehrotra, K., Mohan, C. K., and Ranka, S., '*Elements of Artificial Neural Networks*,' MIT Press, Cambridge, MA, 1997.

[125] Meisel, W., '*Computer Oriented Approaches to Pattern Recognition*,' Academic Press, New York, 1972.

[126] Merlet, N., and Zerubia, J., 'New Prospects in Line Detection by Dynamic Programming,' *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 8, pp. 426-431, 1996.

[127] Mohajeri, M. H., and Manning, P. J., 'ARIA: An Operating System of Pavement Distress Diagnosis by Image Processing,' *Transp. Res. Record*, vol. 1311, Transportation Research Board, Washington, D.C., pp. 120-130, 1991.

[128] Moody, J., and Darken, C., 'Fast Learning in Networks of Locally Tuned Processing Units,' *Neural Computation*, vol. 1(2), pp. 281-294, 1989.

[129] Newman, T. S., and Jain, A. K., 'A Survey of Automated Visual Inspection,' *Comput. Vision Image Understanding*, vol. 61, no. 2, pp. 231-262, 1995.

[130] Nigtin, A., '*Neural Networks for Pattern Recognition*,' MIT Press, Cambridge, MA, 1993.

[131] Novak, C., and Wen, H. K., 'Role of Pavement Management System Analysis in Preservation Program Development,' *Transp. Res. Rec.*, 1344, Transp. Res. Board, Washington, D.C., pp. 1-8, 1992.

[132] Oja, E., '*Subspace Methods of Pattern Recognition*,' Research Studies Press, Letchworth, England, 1983.

[133] Otsu, N., 'A Threshold Selection Method from Gray-Scale Histogram,' *IEEE Trans. Syst., Man Cybernet*, vol. 9(1), pp. 62-66, 1979.

[134] Pal, N. R., and Pal, S. K., 'Segmentation Based on Contrast Homogeneity Measure and Region Size,' *IEEE Trans. on Systems, Man and Cybern.*, vol. 17(5), pp. 857-868, 1987.

[135] Pal, S. K., and Mitra, S., 'Multilayer Perceptron, Fuzzy Sets, and Classification,' *IEEE Trans. on Neural Networks*, vol. 3(5), pp. 683-696, 1992.

[136] Palmer, P. C., Kittler, K., and Petrou, M., 'Using of Attention with the Hough Transform for Accurate Line Parameter Estimation,' *Pattern Recognition*, vol. 27(9), pp. 1127-1134, 1994.

[137] Papoulis, A., '*Probability, Random Variables, and Stochastic Processes*,' McGraw-Hill, New York, 3rd edition, 1991.

[138] Parker, J. P., 'Gray-Level Thresholding in Badly Illuminated Images,' *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13(1), pp. 813-819, 1991.

[139] Pineda, F. J., 'Recurrent Backpropagation and the Dynamical Approach to Adaptive Neural Computation,' *Neural Computation*, vol. 1(1), pp. 161-172, 1989.

[140] Poggio, T., and Girosi, F., 'Networks for Approximation and Learning,' *Proc. of the IEEE Neural Networks Conf.*, vol. 78(9), pp. 1481-1497, 1990.

[141] Porrill, J., 'Fitting Ellipse and Predicting Confidence Envelopes Using a Bias Correction Kalman Filter,' *Image and Vision Computing*, vol. 8(1), pp. 37-41, 1990.

[142] Powell, M. J. D., '*Radial Basis Functions for Multivariable Interpolation: A Review*,' Clarendon Press, Oxford, 1987.

[143] Pratt, W. K., '*Digital Image Processing*,' Wiley, New York, 1978.

[144] Priese, L., and Rehrmann, V., 'On HierarchicalColor Segmentation and Application,' *Proc. in Computer Vision and Pattern Recognition*, Los Alamitos, CA, pp. 633-634, 1993.

[145] Rollins, K., 'Inspection of Underground Sewer Pipes,' *Trenchless technology Journal*, vol.14(4), pp. 25-28, 1999.

[146] Ramesh, V., and Haralick, R. M., 'An Integrated Gradient Edge Detector. Theory and Performance valuation,' *Proc. in ARPA Image Understanding Workshop*, Monterey, CA, ARPA, Los Altos, CA, pp. 689-702, 1994.

[147] Regazzoni, C. S., Foresti, G. L., and Serpico, S. B., 'An Adaptive Probabilistic Model for Straight Edge extraction within a Multilevel MRF framework,' *Proc. of IGARSS*, Italy, pp. 458-460, 1995.

[148] Reilly, D. L., Cooper, L. N., and Elbaum, C., 'A Neural Model for Category Learning,' *Biological Cybernetics*, vol. 45, pp. 35-41, 1982.

[149] Reinhardt, J. M., and Higgins, W. E., 'Efficient Morphology Shape Representation,' *IEEE Trans. on Image Processing*, vol. (5), pp. 89-101, 1996.

[150] Reynolds, A., and Flagg, P., '*Cognitive Pyschology*,' Winthrop, MA, 1977

[151] Robel, A., 'Dynamic Pattern Selection: Effectively Training Backpropagation Neural Networks,' *Proc. of International Conference on Artificial Neural Networks*, Springer Verlag, vol. 1, pp. 643-646, 1994

[152] Rogers, S. K., and Kabrisky, M., '*An Introduction to Biological and Artificial Neural Networks for Pattern Recognition*,' SPIE Optical Engineering Press, Washington, 1991.

[153] Rosenfeld, A., and Kak, A. C., '*Digital Picture Processing*,' Academic Press, New York, 2nd edition, 1982.

[154] Ross, S. M., '*Introduction to Probability Models*,' Academic Press, N. Y., 1972.

[155] Rumelhart, D. E., and McClelland, J., '*Parallel Distributed Processing*,' vol. 1, MIT Press, Cambridge, Massachusetts, 1986

[156] Sadek, A., Freeman T., and Demetsky, M., 'Deterioration Prediction Modeling of Virginia's Interstate Highway System,' *Transp. Res. Rec.*, 1524, Transp. Res. Board, Washington, D.C., pp. 118-129, 1996.

[157] Saffery, J., and Thornton, C., 'Using Stereographic Projection as a Prepossessing Technique for Upstart,' *Proc. of the Intl. Joint Conf. on Neural Networks*, vol. 2, pp. 441-446, 1991.

[158] Sahoo, P. K., Soltani, S., and Wong, A. K. C., 'A Survey of Thresholding Techniques,' *Comp. Vision, Graphics, and Image Process.*, vol. 41(2), pp. 233-260, 1988.

[159] Samadani, R., and Vesechy, J. F., 'Finding Curvilinear Features in Speckled Images,' *IEEE Trans. Geosci. Remote Sensing*, vol. 28, pp. 669-673, 1990.

[160] Schalkoff, R. J., *'Pattern Recognition: Statistical, Structural, and Neural Approaches,'* John Wiley and Sons, Inc., New York, N. Y., 1992

[161] Schettini, R., 'A Segmentation Algorithm for Color Images,' *Pattern Recognition Letters*, vol. 14, pp. 499-506, 1993.

[162] Sera J., 'Introduction to Mathematical Morphology,' *Comput. Vision Graphics and Image Processing*, vol. 35, pp. 283-305, 1986.

[163] Serpente, R. F., 'Understanding the Models of Failure for sewers,' *Proc. of Intl. Conf. on Pipeline Div.*, ASCE, pp. 86-100, 1993.

[164] Serra, J., *'Image Analysis and Mathematical Morphology,'* Academic Press, London, 1982.

[165] Serra, J., 'Morphological Optics,' *J. of Microscopy*, vol. 145(1), pp. 1-22, 1987.

[166] Shahin, M. Y., Kohn, S. D., Lytton, R. L., and McFarland, W. F., 'Pavement M&R Budget Optimization Using the Incremental Benefit-Cost Technique,' *Proc. of the First N. America Conf. on Managing Pavement*, Toronto, pp. 96-107, 1985.

[167] Shaikh, M. A., and Tian, B., 'Neural Network Based Cloud Detection/Classification Using Textural and Spectral Features,' *Proc. of IGRASS*, Lincoln, NB, pp. 1105-1107, 1996.

[168] Skapura, D. M., *'Building Neural Networks,'* Addison-Wesley Publishing Company, New York, N. Y., 1996.

[169] Skibniewski, M. J., and Chao, L. C., 'Evaluation of Advanced Construction Technology with AHP Method,' *J. of Constr. Engrg. and Mgmt.*, vol. 118(3), pp. 88-101, 1984.

[170] Skingley, J., and Rye, A. J., 'The Hough Transform Applied to Images for Thin Line Detection,' *Pattern Recognit. Lett.*, vol. 6, pp. 61-67, 1987.

[171] Smith, A. R., 'Color Gamut Transform Pairs,' *Computer Graphics*, vol. 12(3), pp. 12-19, 1978.

[172] Spoehr, K. T., and Lehmkuhle, S. W., *'Visual Information Processing,'* Freeman, San Francisco, CA, 1982.

[173] Stewart, T. L., 'Operating Experience Using a Computer Model for Pipeline Leak Detection,' *Journal of Pipelines*, vol. 3, pp. 233-237, 1993.

[174] Taylor, R. I., and Lewis, P. H., '2D Shape Signature Based on Fractal Measurements,' *IEE Proc. of Vision, Image and Signal Processing*, vol. 141, pp. 422-430, 1994.

[175] Telfer, B., and Casasent, D., 'Minimum Cost Ho-Kashyap Associative Processor for Piecewise Hyperspherical Classification,' *Proc. of the Intl. Joint Conf. on Neural Networks*, vol. 2, pp. 89-94, 1991.

[176] Theodoridis, S., and Koutroumbas, K., '*Pattern Recognition*,' Academic Press, New York, 1999.

[177] Touzi, R., Lopes, A., and Bousquet, P., 'A Statistical and Geometrical Edge Detector for SAR Image,' *IEEE Trans. Geosci. Remote Sensing*, vol. 26, pp. 764-773, 1988.

[178] Tupin, F., Maitre, M., Mangin, J., Nicolas, J., and Pechersky, E., 'Detection of Linear Features in SAR Images: Application to Road Network Extraction,' *IEEE Trans. Geosci. Remote Sensing*, vol. 36(2), pp. 434-452, 1998.

[179] Van Cauwelaert, F. J., Alexander, D. R., White, T. D., and Barker, W. R., 'Multilayer Elastic Program for Backcalculating Layer Moduli in Pavement Evaluation,' *Nondestructive Testing of Pavements and Backcalculation of Moduli*, ASTM STP 1026, ASTM, pp. 171-188, 1989.

[180] Vincent, L., 'Efficient Computation of Various Types of Skeletons,' *Proc. of the SPIE Symposium Medical Imaging*, San Jose, CA, vol. 1445, pp. 253-267, 1991.

[181] Vincent, L., 'Morphological Grayscale Reconstruction in Image Analysis: Applications and Efficient Algorithms,' *IEEE Trans. on Image Processing*, vol. 2(2), pp. 176-201, 1994.

[182] Vrabel, M. J., 'Edge Detection with a Recurrent Neural Network,' *Proc. of Application and Science of Artificial Neural Networks II*, vol. 2760, SPIE, Bellingham, WA, pp. 365-371, 1996.

[183] Walker, R. S., and Harris, R. L., 'Noncontact Pavement Crack Detection Systems', *Transp. Res. Record*, vol. 1311, Transportation Research Board, pp. 149-157, 1991.

[184] Wang, D., Haese-Coat, V., and Rosin, J., 'Shape Decomposition and Representation Using a Recursive Morphological Operation,' *Pattern Recognition*, vol. 28, pp. 1783-1792, 1995.

[185] Wang, P. P., and Chang, S. K., '*Theory and Applications to Policy Analysis and Information Systems*,' Plenum, New York, 1980.

[186] Wardle, A., and Wardle, L., 'Computer Aided Diagnosis,' *Meth. Inform. Med.*, vol. 17(1), pp. 15-28, 1978.

[187] Werbos, P., '*Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences*,' Ph.D. Dissertation, Harvard, 1974.

[188] Weszka, J. S., and Rosenfeld, A., 'Histogram Modification for Threshold Selection,' *IEEE Trans. on Systems, Man and Cybern.*, vol. 9(1), pp.38-52, 1979.

[189] Wilensky, G., and ManuKian, N., 'The Projection Neural Network,' *Proc. of the Intl. Joint Conf. on Neural Networks*, vol. 2, pp. 358-367, 1992.

[190] Willenbrock, J. H., 'Utility Function Determination for Bidding Models,' *J. Constr. Div.*, ASCE, vol. 99(1), pp. 11-17, 1973.

[191] Williams, T. S., and Glagola, D. M., 'Markovian Models for Bridge Maintenance Management,' *J. of trans. Engrg.*, vol. 120(1), pp. 37-51, 1994.

[192] Winston, P., '*Artificial Intelligence*,' Addison-Wesley, Reading, MA, 1977.

[193] Wirahadikusumah, R., Abraham, D. M., Iseley, T., and Prasanth, R. K., 'Assessment Technologies for Sewer System Rehabilitation,' *J. of Automation in Const.*, vol. 7(4), pp. 259-270, 1998.

[194] Yager, R. R., and Zadeh, R. R., '*Fuzzy Sets, Neural Networks, and Soft Computing*,' Van Nostrand, Reinhold, New York, 1994.

[195] Yakimovsky, Y., 'Boundary and Object Detection in Real World Images,' *J. Assoc. Comput. Mach.*, vol. 23, pp. 599-618, 1976.

[196] Zadeh, L. A., 'Fuzzy Sets,' *Information Control*, vol. 8, pp. 338-353, 1965.

[197] Zhou, Y. T., '*Artificial Neural Networks for Computer Vision*,' Springer Verlag, New York, 1992.

[198] Zhu, Q., Payne, M., and Riordan, V., 'Edge Linking by a Directional Potential Function,' *Image and Vision Computing*, vol. 14(1), pp. 59-70, 1996.