Joint Transportation Research Program

JTRP

OPTICAL IMAGING METHOD FOR BRIDGE PAINTING MAINTENANCE AND INSPECTION

Luh-Maan Chang
Yassir Abdelrazig
Po-Han Chen

April 2000

Indiana
Department
of Transportation

Purdue
University

Final Report

Optical Imaging Method for Bridge Painting Maintenance and Inspection

By
Luh-Maan Chang
Principal Investigator

Yassir Abdelrazig
and
Po-Han Chen
Research Assistant

School of Civil Engineering

1. Report No. FHWA/IN/JTRP-99/11	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Optical Imaging Method For Bridge Painting Maintenance And Inspection		5. Report Date April 2000
		6. Performing Organization Code
7. Author(s) Luh-Maan Chang, Yassir Abdelrazig, and Po-Han Chen		8. Performing Organization Report No. FHWA/IN/JTRP-99/11
9. Performing Organization Name and Address Joint Transportation Research Program 1284 Civil Engineering Building Purdue University West Lafayette, Indiana 47907-1284		10. Work Unit No.
		11. Contract or Grant No. SPR-2197
12. Sponsoring Agency Name and Address Indiana Department of Transportation State Office Building 100 North Senate Avenue Indianapolis, IN 46204		13. Type of Report and Period Covered Final Report
		14. Sponsoring Agency Code

15. Supplementary Notes

Prepared in cooperation with the Indiana Department of Transportation and Federal Highway Administration.

The term "quality" is defined as the conformance to predetermined requirements or specifications. These requirements may be set in terms of the end result required or as a detailed description of how work should be executed. Recently, there has been increasing interest in quality assurance in the construction industry. Quality assurance includes design and planning, sampling, inspection, testing, and assessment to ensure that end products perform according to specifications. This research proposes a new quality assessment model for highway steel bridges and more specifically for coating rust assessment. The research proposes a hybrid model using image processing and neural networks for defect recognition and measurement. The basic concept of the model is to acquire digital images of the areas to be assessed and analyze those images to recognize and measure defect patterns. Neural networks are incorporated into the model to learn from example and simulate human expertise to automate the process for future use. The model is supplemented with a statistical quality assessment plan to use the model efficiently and obtain consistent and reliable results. The statistical plan will determine the number and locations of assessment images to be taken. Moreover, the plan will address the risks associated with the estimated assessment. Finally, the plan will assist making the final acceptance/rejection decision based on the predefined criteria for acceptance and rejection.

17. Key Words Back-Propagation Algorithm, gray level, hybrid model, inspection, neural learning, painting, thresholding	18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 190	22. Price

Form DOT F 1700.7 (8-69)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1 INTRODUCTION

1.1 Background

The construction industry is considered one of the largest industries in the U.S. economy. Through the first quarter of 1998, total new construction adjusted annual rate was running at $630.1 billion (ENR 1998). Moreover, new federal transportation funding is expected to spark a surge of highway construction throughout the U.S. The new bill, the Transportation Equity Act for the 21st Century (TEA-21), will unleash an average of $26.2 billion a year in transportation funds over the next five years (ENR 1998).

Recently, there has been more interest among federal agencies and departments of transportation (DOT's) in adopting alternative contracting methods to speed up the construction process and improve the quality of constructed facilities. Quality Assurance (QA) specifications have been adopted by many DOT's emphasizing the end product and establishing criteria for accepting contractors' work (Brumel 1996; Chang and Hsie 1995). Moreover, many DOT's are adopting warranty clauses in their contracts and they require contractors to redo work if it's below a certain quality level (ENR 1998). Therefore stems the need for methods and techniques to measure quality.

Quality in the construction industry was defined by the American Society of Civil Engineers (ASCE 1988) as:

Totality of features, attributes, and characteristics of a facility, product,

process, component, service, or workmanship that bear on its ability to satisfy

a given need: fitness for purpose.

Quality is usually benchmarked and measured by the degree of conformance to predefined standards. In other words, quality is meeting the requirements or specifications (ASCE 1988; Brumel 1996; Calder 1997). Quality of a constructed project applies to all project parties; the owner, the design professionals, the contractor, and all regulating agencies. Each party views quality from a different angle and has different requirements to be satisfied. Quality requirements for each party is described by ASCE (1988) as:

1. Owner requirements include functional adequacy, completion within time and budget, and minimizing operation and maintenance cost.

2. Design professional requirements include provision of defined scope of work, determined project budget, contract for time and cost, and to furnish an adequately functional constructed facility within time and budget.

3. Contractor requirements include provision of design plans and specifications, contract for cost and time, and constructing the project within time and budget while making reasonable profit.

4. Regulating agencies require conformance with laws and regulations, environmental considerations, safety and health considerations, and protection of public interest.

Naturally, cost and quality expectations may appear to conflict. Time and cost pressures often reduce the time available for design and construction resulting in a conflict between quality and completion within time and budget. There is also the perception that quality increase cost. On the contrary, maintaining good quality during design, construction, and operation proves to decrease total cost over the life cycle of a constructed facility (Calder 1997). Maintenance and operation cost constitutes a major component in the total life cycle cost of a constructed facility (Brumel 1996). Hence the performance quality of the facility plays a significant role in a life cycle cost of any constructed facility.

1.2 Problem Statement

Recently there has been national cognizance to the current problems associated with civil and infrastructure systems in the United States (Abraham et al. 1997; Hunt et al. 1997; Shubinsky 1994; Wang et al. 1997). The Federal Highway Administration (FHWA) estimates that 45% of the nation's bridges are somewhat deficient, which led to $90 billion of needed maintenance (Hunt et al. 1997). This clearly demonstrates the need for better maintenance for civil and infrastructure systems.

Most prevailing practices in construction and infrastructure assessment and quality inspection rely merely on subjective criteria. Such subjective assessment techniques have been identified as a critical impediment to effective infrastructure management (Hunt et al. 1997). A more accurate and reliable assessment systems have to be introduced to improve the conditions of the infrastructure or generally to improve the quality of constructed facilities (Abraham et al. 1997; Shubinsky 1994).

The problem of inefficient quality assessment techniques is more acute and has more impact in civil infrastructure. These deficiencies are more conspicuous in the condition of highway bridges throughout the U.S. (Chang and Hsie 1995). The main reason for the deficiencies in highway steel bridges is the deterioration of anti-corrosion coatings, which comprises one of the largest components of most states' departments of transportation maintenance budget (Shubinsky 1994).

In spite of the fact that the quality of coating is directly related to bridge condition, there have been no much efforts to improve the process of coating quality assessment. The currently used assessment techniques, which depend mostly on visual inspection, are subjective, inconsistent, and time consuming (Shubinsky 1994). Moreover, these techniques have not taken advantage of the rapid advancements in technology, especially computer technology.

Usually, the condition of steel structures is described in terms of the percentage of the surface showing some sort of failure such as rust, blistering, or delamination. Alternatively, a rating system of 1 to 10 or excellent, good, fair, and poor may be used (Steel Structures Painting Manual 1989). For example, for bridges having less than 1-2% rust, the customary choice is to spot clean and spot prime, followed by one or two full topcoats. For structures having greater than 20-25% surface defect, it is usually more cost-effective to clean the entire structure and repaint it. In the intermediate range, the decision is more difficult and usually depends on local preference. (Steel Structures Painting Manual 1989).

The American Society for Testing and Materials (ASTM) has adopted a corrosion performance rating system for coating corrosion assessment. The system assigns a rating

from 0 to 10 for the structure based on the percentage of corroded areas (Steel Structures Painting Manual 1989). Moreover, the system determines recommended percentages of areas to be repainted according to that area. Table 1.1 illustrates the rating system (Steel Structures Painting Manual 1989).

Table 1.1 ASTM Corrosion Performance Rating

Corrosion Rating	Assessment Description	Areas to be Repainted (%)
10	No rust or less than 0.01% rust	0
9	Less than 0.03% rust	0
8	Few isolated spots, less than 0.1% rust	0
7	Less than 0.3% rust	0
6	Extensive rust spots, less than 1% rust	8
5	Less than 3% rust	18
4	Less than 10% rust	40
3	Approximately 1/6 of surface rusted	60
2	Approximately 1/3 of surface rusted	100
1	Approximately 1/2 of surface rusted	100
0	Approximately 100% of surface rusted	100

The problem with such a system is the subjectivity in determining the percentage of the defect area. It is extremely difficult for the human expert to detect such small differences in percentages. Moreover, the consistency problem arises because different inspectors may have different rating for the same area. Most departments of transportation in the U.S. adopt similar systems and hence have resembling problems.

This research proposes a hybrid model using image processing and neural networks for rust recognition and measurement. The basic concept of the model is to acquire digital images of the steel bridge coating areas to be assessed and analyze those images to recognize and measure rust patterns. Neural networks are inaugurated into the model to learn from example and simulate human expertise for future use in order to automate the recognition and measurement of defects.

1.3 Research Objectives

This research purpose is to develop a hybrid model for nondestructive quality assessment of highway steel bridge coating. The model will provide objective and quantitative analysis to help improve the decision process after assessment. The research utilizes the vast capabilities of computers to analyze data and to learn from examples to formulate the model.

The hybrid system utilizes Artificial Intelligence (AI) techniques such as neural networks as tools for visual data analysis, recognition, and classification (Fu and Xu 1998; Garris et al. 1998; Lang et al. 1998). AI techniques utilize computer efficiency and speed and they can also simulate human expertise objectively and consistently. The main objectives of the research are:

1. To develop a hybrid model that provides an objective, reliable, consistent, and quantitative procedure for nondestructive quality assessment of highway steel bridge coating by using image processing for analysis.

2. To provide means to help INDOT to take informed decisions and hence better decisions, which would reduce maintenance and repair costs of highway steel bridges.

3. To construct a statistical quality assessment plan and integrate it with the hybrid model. The plan will determine the quality assessment sampling procedure and acceptable error ranges for the model.

4. To supplement the implementation of INDOT warranty clauses in painting projects.

5. To establish a framework of images database for INDOT steel bridges painting inspection and maintenance.

1.4 Research Methodology

1. Background Review

Review existing painting systems used by INDOT and get familiar with the current painting inspection systems. This phase includes reviewing current practices of bridges condition assessment and pinpointing the deficiencies in these practices.

2. Literature Review

An extensive investigation on the topics of steel bridge painting and computer visual image processing used in other states, industries, and research institutes. Moreover, another step is to review artificial intelligence concepts and applications in construction and to review different intelligent decision support systems in order to identify procedures to develop an intelligent interactive decision support model.

3. Equipment Purchasing

Identify and purchase needed equipment including a digital camera and computer hardware and software.

4. System Testing

Experimental testing of the purchased equipment experiments is performed to set up the system and to ensure methodology and equipment implementation on site.

5. Data Acquisition

Some INDOT demonstration painting projects are selected for data acquisition. Digital images are acquired from these bridges for assessment.

6. Analysis

A thorough system analysis is made for data collected to achieve reliable results for step-by-step implementation. The main part of this stage is to obtain sample images for the system initial development and training using neural networks. The objective is to identify defects in coating and to provide quantitative measures of these defects. The objective of this phase is to formulate the basic structure of the hybrid system.

7. System Implementation

A generic framework for a statewide system implementation by INDOT will be developed. A database for bridge images will be formulated.

8. Personnel Training

Training sessions will be held to familiarize INDOT key painting personnel with the system.

9. Final Report

A technical report will be prepared and submitted to INDOT for final approval. The report would include; methodology used, references cited, technical specifications, experiments done and their results, and the implementation plan.

CHAPTER 2 PRIOR RESEARCH EFFORTS

There hasn't been much research work in the area of computerized construction quality assessment. Most previous research focus was on infrastructure assessment including bridge coating assessment, pavement analysis, and underground utility assessment. Furthermore, there has been some work done in the area of construction materials quality assessment especially concrete assessment (Abraham et al. 1997; Cox 1996; Hunt et al. 1997; Liu 1997; Shubinsky 1994).

There have been some integrated research efforts performed at the Infrastructure Technology Institute (ITI) at Northwestern University. The main thrust of that research was the non-destructive evaluation of bridge coatings using several techniques namely; Color Visible Imaging (CVI), Electrochemical Impedance Spectroscopy (EIS), and Infrared Imaging (Shubinsky 1994).

According to interviews conducted at ITI at Northwestern University, the result of that research was a generic framework and was not developed to the implementation stage. In that research, experiments using visual imaging technique were done on steel bridges to obtain the amount of damaged areas due to rusting. Additional tests were performed on

coated test panels inside the lab. Results from these experiments showed the damage area percentage due to rust.

One of the research recommendations was to develop an implementable system, which should be portable and user-friendly, so that inspection personnel can handle the system. Moreover, a recommendation was to incorporate some form of artificial intelligence to automate the assessment process (Shubinsky 1994).

There has also been some research in the area of pavement surface distress analysis. An example of that is using a high-performance digital line scan camera mounted in a moving vehicle. The system uses image processing for automated pavement distress analysis (Wang et al. 1997). The system uses high-performance camera for real-time analysis.

Abraham et al. (1997) proposed an integrated system for underground utility assessment. The system uses fuzzy Kohonen self-organizing maps that read data from sensors to create membership functions that indicate the utility condition. Data is collected from many sensors and fed into the neural network for mapping. A fuzzy estimator will gather mapped output to estimate utility condition.

Some research work focused on the quality assessment for construction materials. Examples for this may include research done on quality prediction for concrete manufacturing. Neural networks were used to acquire qualitative knowledge during the production of concrete. The system predicts concrete quality under different mixing formulations (Liu 1997).

Another example is the crack and damage assessment in concrete and polymer matrices. This system utilizes optic sensors to remotely measure crack occurrence in real

time and determine the location and volume of crack damage (Dry and McMillan 1996). Another example is the use of prefabricated sensor panels for large civil structure instrumentation to monitor performance of the structure (Fuhr et al. 1993).

2.1 Visual Quality Assessment

The construction process can be considered as an industrial application because many industrial concepts may apply to construction. Therefore, industrial assessment techniques can be applied to construction quality assessment. The most widely used inspection systems in industrial applications are based on visual data assessment. Assessment procedures make decisions about the condition of the inspected areas based on features extracted from the assessment data. (Croall and Mason 1992).

Features selection and features data acquisition for recognition and classification is an important step for assessment (Lang et al. 1998). Features can be extracted from different data types in three major levels of visual data processing. In the lowest level of processing, digital image processing techniques are used to handle image data in a numerical array format. Images can be enhanced and basic features such as edges and regions are extracted through a number of algorithms such as edge detection, line finding, and region segmentation algorithms (Navabi and Agarwal 1998).

At the intermediate level of processing, quantitative features such as geometric coordinates, area, ratio of maximum and minimum intensity, major axes, and many other features are extracted from the image. Finally, in the higher level, pattern recognition

techniques are applied to classify features into either defects or background based on the feature vectors generated (De La Blanca 1992).

Recently, computer technological advances have rendered image processing a practical and affordable solution for many applications that were extremely difficult prior to the availability of this technology. Furthermore, the increased computer processor speed, the utilization of signal and image processing hardware devices, and the developments of color image processing techniques have made complex image analysis possible at fast speeds (Fu and Xu 1998; Garris et al. 1998; Lang et al. 1998).

There are a wide variety of possible assessment tasks where image processing would greatly benefit the final assessment process, in terms of quality and consistency. For example, imaging systems are able to distinguish defects from the rest of the surface, and determine the size, shape, location and even the smallest attributes of the defect (Shubinsky 1994).

Many assessment tasks entail comparing the characteristics of the object in question with previous assessment results and verifying that there are no significant changes due to defects. Imaging is an ideal tool for such applications. An object image can be easily compared with a reference image based on its global features even if the images are not identical. Quantitative differences can be immediately identified and graphically displayed.

Implementation of visual imaging allows establishing of a database with detailed assessment results, including size and location of each defect found. Since images and assessment data can be stored in digital format, a history of assessment results can be appraised and compared in order to recognize changes that occur with time.

2.2 Computerized Visual Imaging for Bridge Coating Assessment

There was some research effort at the Infrastructure Technology Institute (ITI) at Northwestern University focusing on computerized steel bridge coating assessment. The research objective was to develop a computerized system for coating assessment using computerized visual imaging. There were also experiments using other techniques such as infrared imaging and electrochemical impedance spectroscopy (Shubinsky 1994).

Computerized Visual Imaging (CVI) is a technique that resembles regular optical techniques currently used for bridge inspection, such as cameras and camcorders. Nonetheless, CVI is functionally different than those techniques because it includes computerized image processing for assessment. Basically, CVI has three main components to provide quantitative and reliable assessment results on a real-time basis: A digital camera, processing computer, and image analysis hardware and software (Shubinsky 1994; Wang et al. 1997).

Additional features of the CVI method include image database management and image modification techniques such as labeling and annotation. Moreover, quantitative image analysis can be performed to identify and measure defects accurately. CVI can also be used to determine the type of failure and the percentage of defect areas by real-time image processing.

Most of the common types of coating failures, such as chalking, blistering, flaking, and rusting can be identified by CVI. Furthermore, assessment of surface preparation quality and coating application uniformity for a newly applied coating can also be performed. The only type of assessment not applicable to the CVI method is depth analysis such as film

thickness, which requires a contact-type of assessment technique such as an acoustic or magnetic method (Shubinsky 1994).

An intelligent machine vision system based on pattern recognition and image processing can simulate human visual expertise for coating assessment. Furthermore, the system can be designed for an objective, real-time recognition of the general types of coating failures. Current computer hardware and software technologies also allow implementation of a user-friendly interface.

CHAPTER 3 IMAGE PROCESSING

3.1 Introduction

Images significance in the history of human civilization is conspicuous. In ancient times, images, represented as drawings, have played a momentous role in all aspects of life. Recently, imaging technologies are developing rapidly and it is predicted that the importance of images will increase continuously in the future (Russ 1995).

When analyzing images, the absolute area of interest is always an object or a group of objects existing in the real world. For example, in photography, the object would be a human face, a building, landscape, etc. In remote sensing, it could be terrain, urban areas, etc. In medical diagnosis, objects may include organs and tissues. In quality control, it could be products and defects. Together with their surroundings, these objects constitute an image.

Generally, an image is formed by means of visible light. The light emanates from a light source and interacts with the surfaces of objects. After that, part of the light is captured by an image formation system, like a camera, which produces a two-dimensional distribution of the light defined on the image plane. It is this distribution of light that we call an image (Looney 1997).

The recognition of an object in an image is a complex process that involves a wide range of elaborate techniques. The first step in a typical image processing application is

image acquisition, which is acquiring an image in digital format. The term "image" refers to a two-dimensional light intensity function $f(x,y)$ where x and y are geometrical coordinates and the value of f at any point (x,y) is proportional to the brightness or gray level of the image at that point. A digital image can be considered as a matrix in which each row and column represents a point in the image and the corresponding matrix element value represents the gray level at that point. The elements of such an array are called picture elements or pixels (Russ 1995; Weeks 1996).

After a digital image has been acquired, the next step is preprocessing, which corresponds to low-level processing. A number of prepossessing techniques is available in practice. These may include gray scale manipulation, noise filtering, isolation of regions, geometric correction, restoration, and reconstruction.

The next stage is the recognition stage, which corresponds to high-level processing and it deals with classification. It assigns a classification to an object based on the parameters provided by its descriptors. Conventional classification techniques are grouped into two categories: supervised and unsupervised techniques. In a supervised mode, classifiers learn with the help of training sets. In the case of unsupervised mode, classifiers learn without training sets. Neural networks represent a powerful and reasonable alternative to conventional classifiers (Garris et al. 1998; Lang et al. 1998; Navabi and Agarwal 1998)

Artificial neural network models with learning algorithms such as back-propagation are being used as supervised classifiers, and self-organizing neural networks with learning algorithms such as competitive learning are being used as unsupervised classifiers (Looney 1997; Haykin 1999).

3.2 Image Segmentation

A critical domain of image processing is the segmentation of an image into different regions to separate the objects from the background. Image segmentation can be divided into three main categories (Weeks 1996). The first technique is image thresholding, which employs a predefined gray level as decision criteria to separate an image into different regions based upon the gray levels of the pixels.

The second method is edge detection, which uses the discontinuities between gray level regions to detect edges of objects within an image. Edges play a very important role in the extraction of features for object recognition and identification. The third method is region-based separation, which is to separate an image into several different regions based upon desired criteria (Weeks 1996).

Subsequent to segmenting an image into different objects, it is often recommended to describe these objects using a small set of descriptors, thus reducing the complexity of the image recognition process. Since edges play an important role in the recognition of objects within an image, contour description methods have been developed that completely describe an object based upon its contour. An object within an image can also be described using several region descriptors such as its area, height, and width (Bhanu and Lee 1994).

3.3 Image Thresholding

Thresholding is the operation of separating the image into different regions based upon its gray level distribution. Consider an image containing an object and background with mean gray levels of 220 and 80, respectively. Separation of the object pixel from the

background pixels is accomplished by selecting a gray level value T such that all pixels within the image with $f(x, y) > T$ will be classified as pixels belonging to the object (Russ 1995; Weeks 1996).

For the example image shown in Figure 3.1, any threshold value T between 80 and 220 will separate the object pixels from the background pixels. The goal of thresholding is to select a threshold value that separates an image into two distinct gray level groups:

$$h(x, y) = \begin{cases} GL_a \rightarrow f(x, y) \leq T \\ GL_b \rightarrow f(x, y) > T \end{cases} \tag{3.1}$$

Where GL_a and GL_b are the desired two gray levels in the thresholded image and $f(x,y)$ is the image function where x and y are spatial coordinates and the value of f at any point (x,y) is proportional to the brightness or gray level of the image at that point and $h(x,y)$ is the modified image function after applying the threshold operation. The process of thresholding, as described in Equation 3.1, reduces an image with multiple gray levels to a two gray levels image containing gray levels of GL_a and GL_b. Typically, this two gray level image is referred to as a binary or binarized image (Weeks 1996).
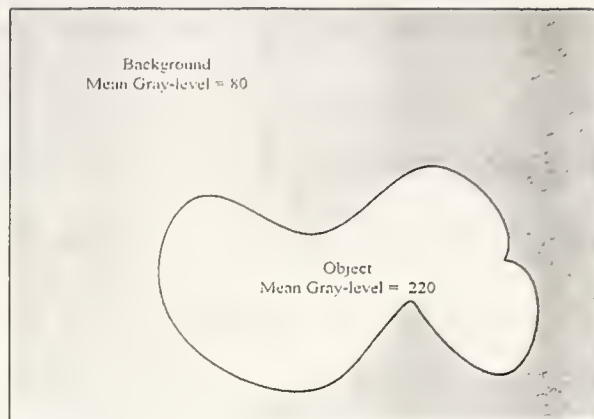
Figure 3.1 Example of Object and Background in an Image

Critical to the selection of a threshold value is an image's histogram, which is the count for the occurrence of each pixel value in the image to define the gray level distribution of its pixels (Russ 1995). Examining an image's histogram can instantaneously indicate the general location of best threshold value.

Figure 3.2 illustrates the image, shown in Figure 3.1, corresponding histogram showing the distribution of gray levels for this image. The bimodal character of this histogram is typical of images containing two prevailing regions of two distinctly different gray levels. The first mode of this histogram gives the frequency of pixels associated with the background, while the second mode gives the frequency of pixels associated with the object.

Examination of this histogram shows that the gray level distribution of pixels for both the background region and the object region appear to have bimodal histogram. The histogram have two peaks located at gray levels of 80 and 220, with the height of these peaks equal to the frequencies of background and object pixels present within this image.
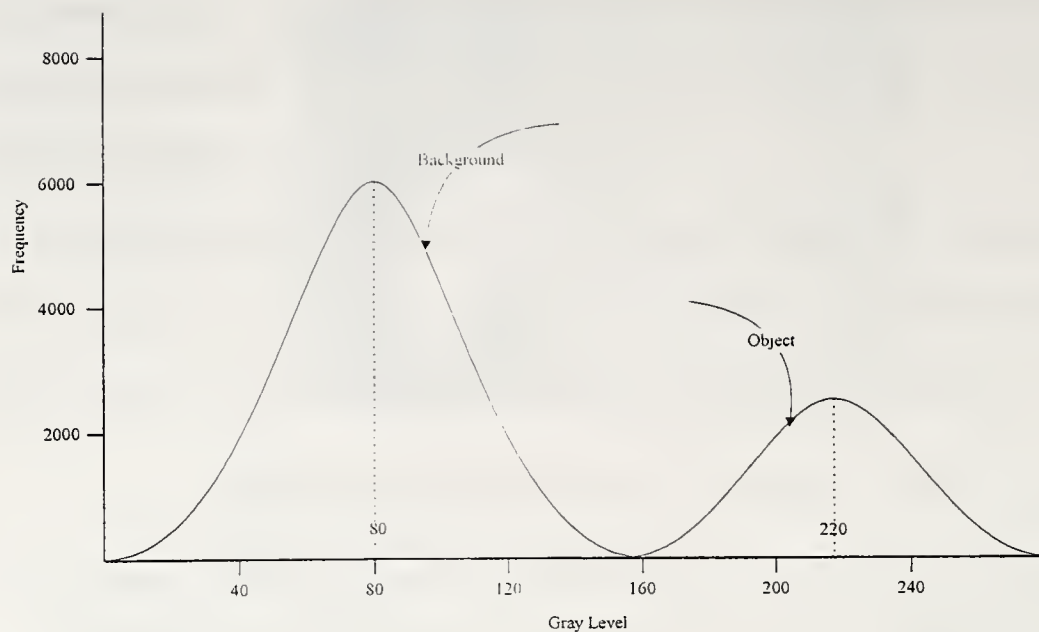
Figure 3.2 Threshold Example: Image Histogram.

Thresholding Figure 3.1 using Equation 3.1 with $GL_a = 0$, $GL_b = 255$, and $T = 128$ yields the binarized image shown in Figure 3.3.
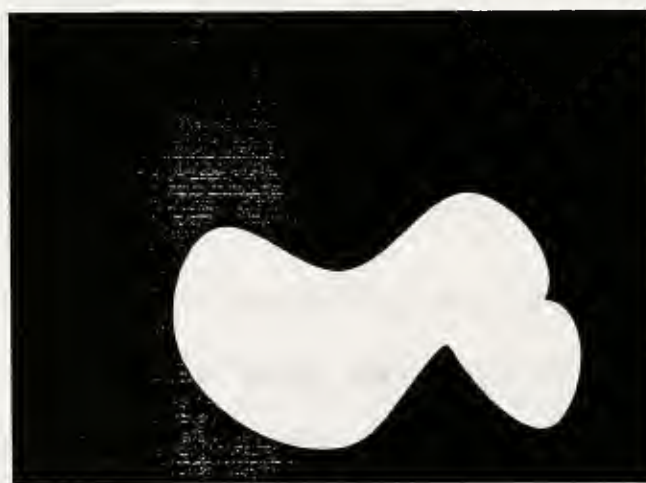
Figure 3.3 Threshold Example: Binarized Image.

The thresholding of the example image given in Figure 3.1 was effortlessly accomplished, since the gray levels of pixels associated with the object were distinctly separated from those associated with the background. When there are more than two distinct regions, a better method would be to use multi-level thresholding. The approach taken by multilevel thresholding is to expand Equation 3.1 to include more than one threshold value as follows:

$$h(x,y) = \begin{cases} GL_a \rightarrow 0 \leq f(x,y) < T_1 \\ GL_b \rightarrow T_1 \leq f(x,y) < T_2 \\ GL_c \rightarrow T_2 \leq f(x,y) < GL_{max} \end{cases} \tag{3.2}$$

Where GL_{max} is the maximum gray level of the image $f(x,y)$. Equation 3.2 segments the image into three gray level regions, GL_a, GL_b, and GL_c depending on two threshold values, T_1 and T_2. The thresholding of a grayscale image is usually easy if the gray levels of the pixels defining an object are clearly separated from the gray levels defining the background, as in the image example shown in Figure 3.1.

Under that condition, the histogram of the image will be bimodal, and the best threshold value is selected in the area between the two peaks. In a complex image, the gray levels of the objects may not be well separated, making the selection of the threshold value more complex (Russ 1995).

Typically, a complex image's histogram is anything but bimodal, and the selection of the best threshold is a compromise between the segmentation of the different features that are

present within the image. Figure 3.4 is an example of a complex image and Figure 3.5 represents its corresponding histogram. Note the lack of a clear separation between the objects in the image and the background.

Figure 3.4 Thresholding of a Complex Gray Level Image: Original Image
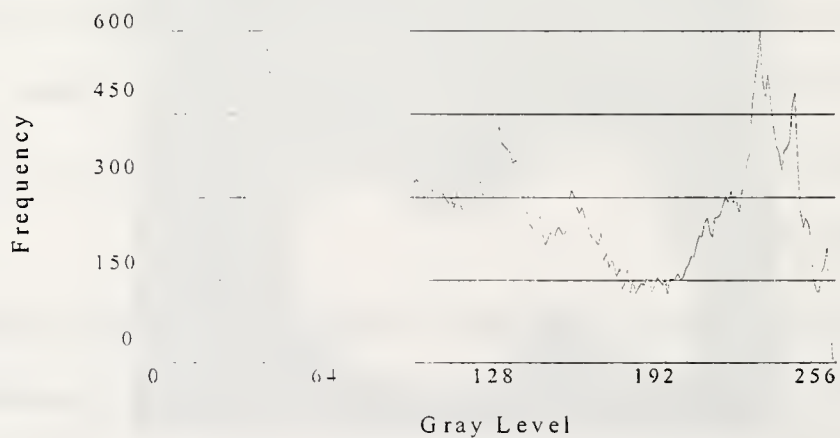
Figure 3.5 Thresholding of a Complex Gray Level Image: Corresponding Histogram

Figures 3.6 and 3.7 represent the thresholded images of the example image shown in Figure 3.4 with threshold values of $T = 60$ and $T = 180$, respectively. Depending on the threshold value selected certain features are highlighted or eliminated. In thresholding this example image, there is no one best threshold value. The value chosen depends on the desired features to be segmented from the other features present in the image. Usually a trial and error method is used to select the threshold value that involves a human interpretation of the thresholded image to determine if the threshold value selected has extracted the desired object features (Russ 1995).

Figure 3.6 Thresholding of a Complex Gray Level Image: Threshold $T = 60$
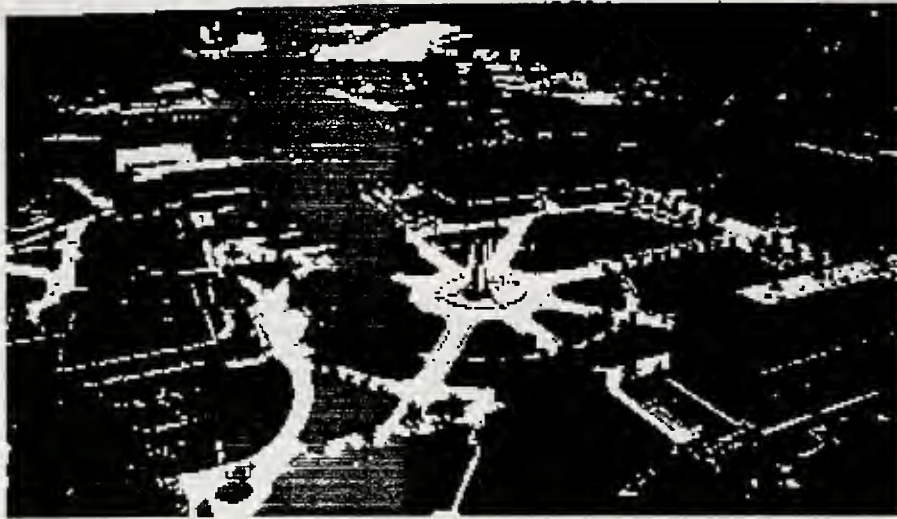
Figure 3.7 Thresholding of a Complex Gray Level Image: Threshold $T = 180$

3.3.1 Threshold Selection

There are many adaptive thresholding methods that aspire to select a threshold value that either maximizes the information present in the thresholded image or attempts to minimize the error associated with the selection of a threshold value. One of the most popular adaptive thresholding algorithms is optimum thresholding (Weeks 1996; Russ 1995).

Optimum thresholding algorithm is based upon a two-object model in which an image is separated into a set of object pixels and a set of background pixels (Bhanu and Lee 1994). Essentially, optimum thresholding assumes an image histogram that is bimodal, with a valley separating the two modal peaks.

Optimum thresholding is based upon the Minimum Probability of Error (MPE) classifier model for signal processing used in the detection of a binary signal in the presence

of a noise (Weeks 1996). Noise in an image can be defined as any external random factor, which affects the quality of an image such as camera motion, reflected light, etc. The MPE classifier model selects a threshold value that minimizes the error of detecting a 0 or 1 from a binary signal (Weeks 1996; Russ 1995).

Let P_o be the probability of an object in an image and P_b be the probability of the background; then

$$P_o = \frac{n_o}{n_t}$$

(3.3)

and

$$P_b = \frac{n_b}{n_t}$$

(3.4)

Where n_o is the number of object pixels, n_b is the number of background pixels, and n_t is the total number of pixels in the image. From Equations 3.3 and 3.4

$$P_o + P_b = 1$$

(3.5)

The sum of the two probabilities must add up to one since the assumption is that there only object and background pixels present within the image. In the derivation of optimum thresholding it is not required that the frequency of gray levels for the background be equal to the distribution of gray levels for the object.

Let $O(GL_i)$ be the histogram for the object pixels and $B(GL_i)$ be the histogram for the background pixels. Figures 3.8 and 3.9 illustrate the two histograms of the distribution of gray levels for the background and the object. Thresholding these images with a value of T

produces two errors. The first error is the misclassification of object pixels with gray levels below T as background pixels. This is shown in Figure 3.8 as the shaded area given by $E_1(T)$. The second error is the misclassification of background pixels with gray levels greater than T as object pixels. This error corresponds to the second shaded area $E_2(T)$ as shown in Figure 3.9.
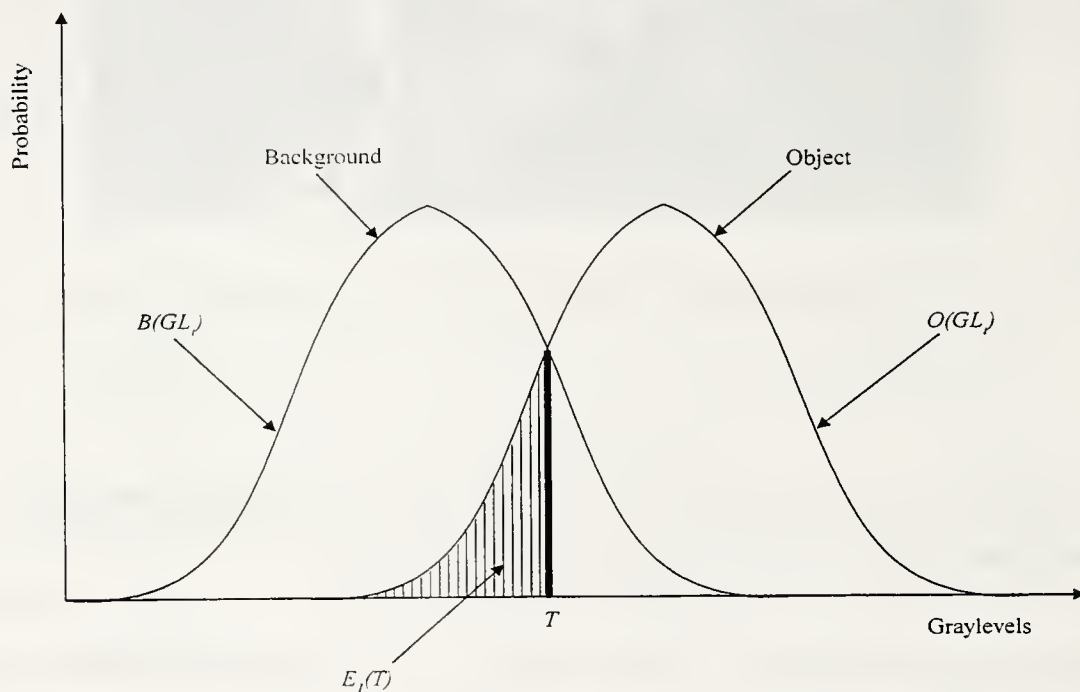
Figure 3.8 Error in Object and Background Histogram (Object Error)
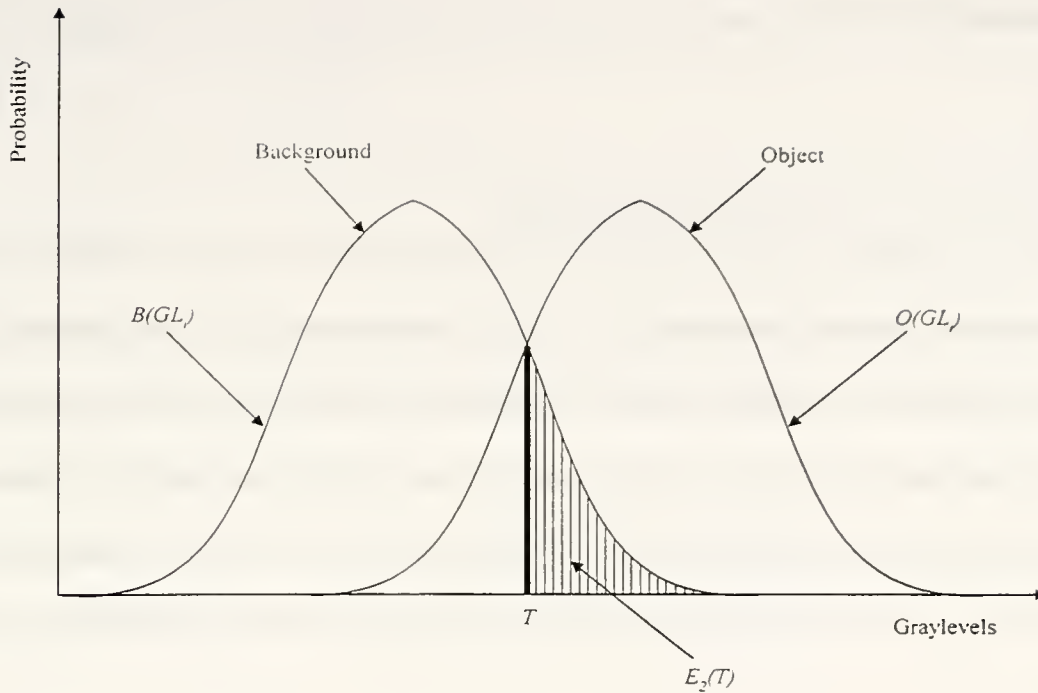
Figure 3.9 Error in Object and Background Histogram (Background Error)

The error $E_1(T)$ of assigning object pixels as background pixels is defined as the sum of the histogram $O(GL_i)$ of the object from gray level 0 to the gray level $(T-1)$:

$$E_1(T) = P_o \cdot \sum_{i=0}^{T-1} O(GL_i).$$

(3.6)

Likewise, the error $E_2(T)$ associated with selecting background pixels as object pixels is equal to the sum of the histogram $B(GL_i)$ of the background pixels from gray level T to gray level GL_{max}:

$$E_2(T) = P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i).$$

(3.7)

Where GL_{max} is defined as the maximum gray level contained within the image. The total error of thresholding this bimodal image is then equal to the sum of the two errors:

$$E_T(T) = E_1(T) + E_2(T),$$

(3.8)

or

$$E_T(T) = P_o \cdot \sum_{i=0}^{T-1} O(GL_i) + P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i).$$

(3.9)

The goal of optimum thresholding is to select a threshold value T that minimizes the total error $E_T(T)$. As the threshold value is decreased, the error $E_1(T)$ associated with incorrectly recognizing object pixel decreases but the error $E_2(T)$ of incorrectly recognizing background pixels as object pixels increases. The opposite occurs when increasing the threshold value T. Nevertheless, there is an optimum threshold value that minimizes these two errors.

Taking the derivative of Equation 3.9 and setting it to zero yields the minimum error as shown in Equation 3.10.

$$E_T(T) = P_o \cdot \sum_{i=0}^{T-1} O(GL_i) + P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i).$$

$$\frac{d}{dT}\left[P_o \cdot \sum_{i=0}^{T-1} O(GL_i) + P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i) \right] = 0.$$

$Minimum \Rightarrow E_T(T)$

$$P_o . O(T) = P_b . B(T).$$

(3.10)

Equation 3.10 states that the minimum classification error occurs when the product of the probability of occurrence of the object and its histogram evaluated at gray level T is equal to the product of the probability of occurrence of background and its histogram evaluated at

gray level T. To solve that equation, the histograms of the background and object pixels have to be known.

In many cases, if an image has been affected by a known type of noise, the object and the background histograms can be estimated directly from the noise histogram (Weeks 1996). The probabilities P_o and P_b are found from the percentage of area that the object and background occupy within the image. Consider the object pixels have a mean gray level of GL_o, and the background pixels have a mean gray level of GL_b, and the noise process can be approximated as Gaussian Distribution with the same variance σ^2 (Weeks 1996). The histograms for the object and background become:

$$O(GL_i) = \frac{e^{-(GL_i - GL_o)^2/\sigma^2}}{\sqrt[\sigma]{2\pi}}$$

(3.11)

and

$$B(GL_i) = \frac{e^{-(GL_i - GL_b)^2/\sigma^2}}{\sqrt[\sigma]{2\pi}}$$

(3.12)

Substituting Equations 3.11 and 3.12 into Equation 3.9 and solving for the optimum threshold value shown in Equation 3.10 yields:

Optimum threshold value is at minimum total error.

$$Minimum \Rightarrow E_T(T)$$
$$P_o.O(T) = P_b.B(T).$$

$$E_T(T) = P_o.\sum_{i=0}^{T-1} O(GL_i) + P_b.\sum_{i=T}^{GL_{max}} B(GL_i),$$

The values of the object and background histograms are:

$$O(GL_i) = \frac{e^{-(GL_i - GL_o)^2 / \sigma^2}}{\sqrt[\sigma]{2\pi}}$$

$$B(GL_i) = \frac{e^{-(GL_i - GL_b)^2 / \sigma^2}}{\sqrt[\sigma]{2\pi}}$$

Substituting the values of object and background histograms and solving for minimum error yields:

$$T = \frac{GL_o + GL_b}{2} + \frac{\sigma^2}{GL_o - GL_b} \ln\left(\frac{P_b}{P_o}\right) \tag{3.13}$$

Equation 3.13 represents the solution of the two-class discrimination problem under the presence of noise, which is known as the MPE classifier algorithm. For an equal probability of occurrence for the object and the background, the optimum threshold value is equal to the average of the gray level of the object and the background. From Equation 3.13, if the probability of the object exceeds that of the background, the optimum threshold value is reduced. The inverse is true when the probability of the background is greater than the probability of the object.

3.3.2 Numerical Example

The following example illustrates how a threshold can be selected for a simple (5 pixel x 5 pixel) image. Consider the image shown in Figure 3.10 as an image of a small portion of a steel beam in a highway bridge with coating on it. In this image the lighter areas are classified to represent defects (rust) and the dark areas to represent the background

(coating) based on randomly selected threshold value of 120. Figure 3.10 illustrates the image and the corresponding gray level values for the image pixels.

5	232	34	32	32	121
4	232	35	34	32	35
3	121	34	35	122	45
2	115	34	32	32	32
1	20	34	115	34	32
	1	2	3	4	5
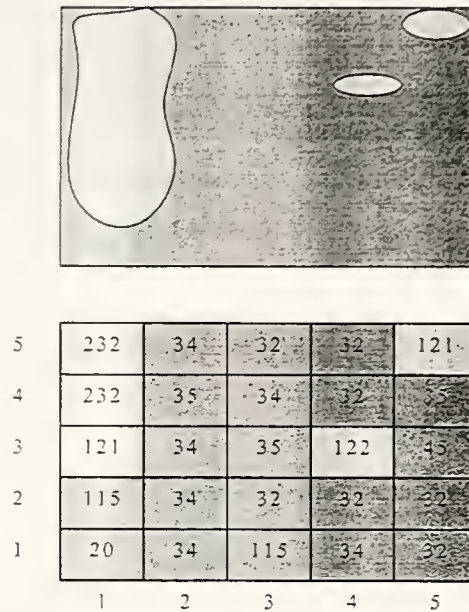
Figure 3.10 Example Image and Corresponding Gray Levels

The algorithm described in Section 3.3.1 is applied to obtain the optimum threshold value. The initial threshold T is randomly selected to be 120 and pixel values with greater values were considered objects and pixels with lesser values were considered background. The algorithm objective is to calculate the errors associated with the randomly selected threshold value and to calculate the optimum threshold value.

The probability of object pixels

$$P_o = \frac{n_0}{n_t}$$

$$= 5/25 = 0.2$$

The probability of background pixels

$$P_b = \frac{n_b}{n_t}$$

=20/25 =0.8

Figure 3.11 illustrates the histogram for the image using a threshold value of $T = 120$.
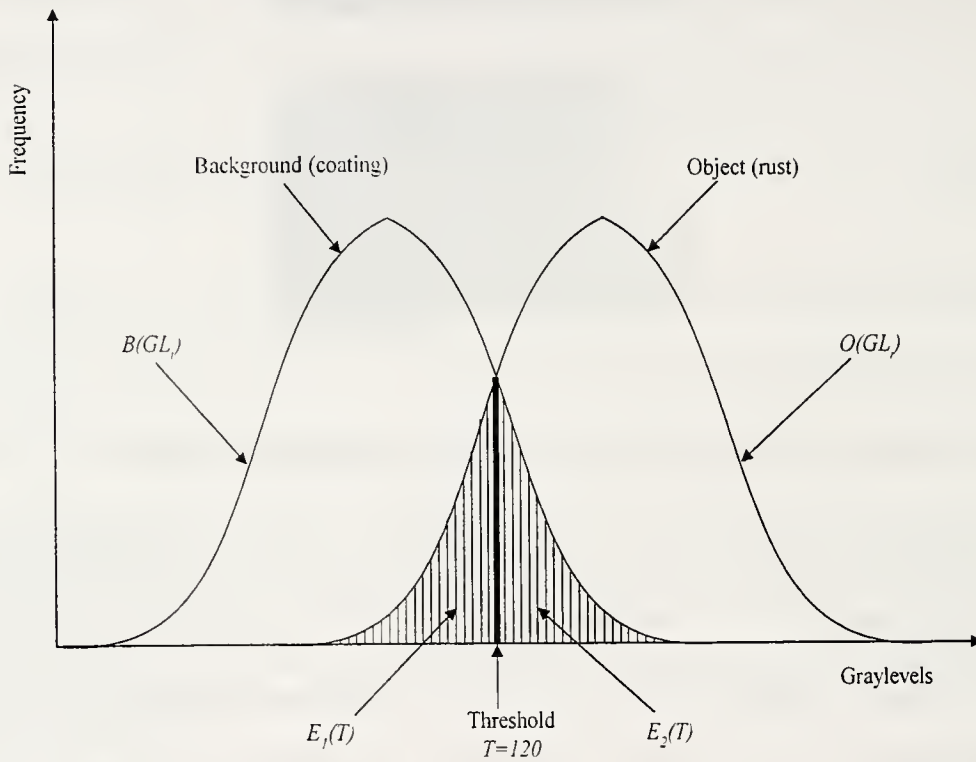
Figure 3.11 The Image Object and Background Histograms

The mean gray level for object pixels

$GL_o = (232+232+121+121+122)/5 = 165.6$

The mean gray level for background pixels

$GL_b=$ $(115+20+34+35+34+34+34+32+34+35+32+115+32+32+32+34+35+45+32$

$+32)/20$

$=41.4$

Optimum threshold value

$$T = \frac{GL_o + GL_b}{2} + \frac{\sigma^2}{GL_o - GL_b} \ln\left(\frac{P_b}{P_o}\right)$$

Assume that the variance of the noise distribution equals 1.

$$T = \frac{165.6 + 41.4}{2} + \frac{1^2}{165.6 - 41.4} \ln\left(\frac{0.8}{0.2}\right)$$

$= 103.51$ (which should be rounded to the nearest integer value).

$= 104$

That means the minimum error obtained is when selecting a threshold value of 104. Calculating the error associated with the threshold value of 104 yields 0 error, which is due to the simplicity of the image and the relatively clear distinction between the object and the background. Figure 3.12 shows the image after applying a threshold value of 104 with the lighter areas representing the objects (defect) and the darker areas representing the background (non-defect).

5	232	34	32	32	121
4	232	35	34	32	35
3	121	34	35	122	45
2	115	34	32	32	32
1	20	34	115	34	32
	1	2	3	4	5

Figure 3.12 Classified Image After Applying Threshold $T=104$

The following illustrates the calculation of the error associated with the initially selected threshold value of 120 given that the image is classified using an optimal threshold value of 104. The error $E_1(T)$ of assigning object pixels as background pixels is defined as the sum of the histogram $O(GL_i)$ of the object from 0 to the gray level $(T-1)$.

$$E_1(T) = P_o \cdot \sum_{i=0}^{T-1} O(GL_i).$$

$$E_1(T) = 0.2 \sum_{i=0}^{120-1} O(GL_i).$$

The value of the summation equals the count of all object pixels of gray level value from 0 to 119. In this example, there are two object pixels, out of 5 pixels, of gray level value 115.

$$=0.2[2/5_{(115)}]$$

$$=0.08$$

Likewise, the error $E_2(T)$ associated with selecting background pixels as object pixels is equal to the sum of the histogram $B(GL_i)$ of the background pixels from gray level T to

gray level GL_{max}. Where GL_{max} is defined as the maximum gray level contained within the image.

$$E_2(T) = P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i).$$

$$E_2(T) = 0.8 \sum_{120}^{232} B(GL_i).$$

The value of the summation equals the count of all background pixels of gray level value from 120 to 232. In this example, there are no background pixels in that range.

$$= 0$$

The total error of thresholding this bimodal image is then equal to the sum of the two errors.

$$E_T(T) = E_1(T) + E_2(T),$$

$$E_T(T) = 0.08 + 0.0 = 0.08$$

Note that clearer distinction between objects and background yields smaller errors. Moreover, the value of the variance was assumed to be 1. Assuming different values for the noise distribution variance doesn't affect the optimal threshold value substantially. Table 3.1 shows resulting threshold values T corresponding to different variance values. It can be safely concluded that assuming up 3 or 4 standard deviations, resulting in 9 or 16 variance will give the same optimal threshold value.

Table 3.1 Threshold Values for Different Variance Values

σ^2	1	4	9	16	25	100
T	103.5	103.54	103.6	103.63	103.78	104.6

3.4 Pattern Recognition for Thresholding

Some algorithms adopt a method that treats the segmentation of a grayscale image into a binary image as a classification problem, in which two classes are generated from the set of pixels within the grayscale image (Weeks 1996). The first class defines the set of pixels within the image with gray levels below or equal to the threshold value T, while the second class defines the set of pixels with gray levels above the threshold value T.

The concept of grouping data in classes based upon the similarity of the data is one of the fundamental concepts of pattern recognition. The concept is illustrated in Figure 3.13 which shows a plot of scattered data described by their x and y coordinates.
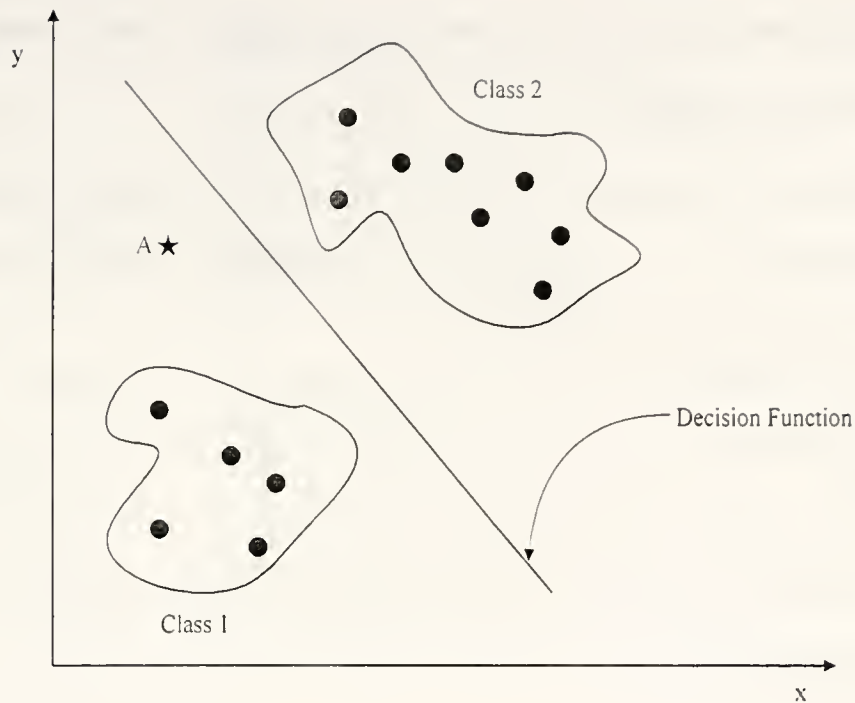
Figure 3.13 Concept of Classes

Figure 3.13 shows that these scattered data have been clustered into two different regions. A decision function can be drawn between the two clusters to separate the two regions into separate classes. All the scattered data below that decision function are grouped to form Class 1, while the data above the function will form Class 2. For example, an unknown point A will be classified as belonging to class 1.

The goal of the pattern recognition thresholding algorithm is to select a threshold value that minimizes the areas of the two classes while maximizing the distance (variance) between these two classes. For example, the unknown point A in Figure 3.13 is selected as belonging to the class in which its area increases the least to include that point or simply the

closer class. In that example, point A will be classified as belonging to Class 1. The algorithm application procedure is as follows:

The first step in the algorithm is to compute the gray level histogram $h(GL_i)$ of the image to be thresholded. The average gray level for the first class c_1 with set of pixels with gray levels below or equal to the threshold value T, is

$$c_1(T) = \sum_{i=0}^{T} i.h(GL_i).$$

(3.14)

The average gray level for the second class c_2 with set of pixels with gray levels above the threshold value T, is

$$c_2(T) = \sum_{i=T+1}^{GL_{max}} i.h(GL_i).$$

(3.15)

The average gray level for the entire image c_T is

$$c_T = \sum_{i=0}^{GL_{max}} i.h(GL_i).$$

(3.16)

The probability of Class 1 occurring within the image is

$$P_1(T) = \sum_{i=0}^{T} h(GL_i).$$

(3.17)

While the probability of Class 2 occurring is

$$P_2(T) = \sum_{i=T+1}^{GL_{max}} h(GL_i).$$

(3.18)

The total of these probabilities must equal to one

$$P_1(T) + P_2(T) = 1.$$

(3.19)

The distance between the two classes is given by the variance:

$$\sigma^2 = \frac{\left[c_1(T).P_2(T) - c_2(T).P_1(T)\right]^2}{P_1(T).P_2(T)}$$

(3.20)

The maximization of Equation 3.20 with respect to threshold value T gives the maximum separation between the two classes and therefore yields the optimum threshold value. The algorithm is an iterative algorithm and it starts with a threshold value of 1 and the variance is calculated. The process is repeated for all gray level values between 1 and $(GL_{max} - 1)$. The threshold value that yields the maximum variance is selected as the optimum threshold value (Weeks 1996).

3.4.1 Numerical Example

The same previous numerical example will be applied to select the threshold using the pattern recognition algorithm. The algorithm will classify the image's pixels into two classes: object pixels (defect) and background pixels (non-defect). Consider the same image of a small portion of a steel beam of a highway bridge with coating on it. The light areas represent defects (rust) and the darker areas represent the background (coating). Figure 3.14 illustrates the coating image at the top and the corresponding gray level values for the image pixels at the bottom.
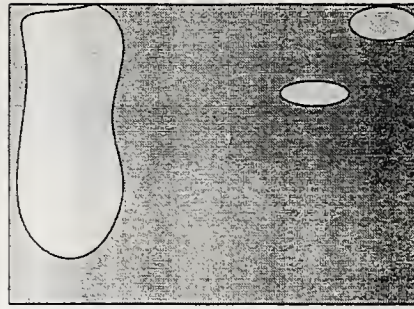
5	232	34	32	32	121
4	232	35	34	32	35
3	121	34	35	122	45
2	115	34	32	32	32
1	20	34	115	34	32
	1	2	3	4	5

Figure 3.14 The Example Image and Gray Levels

The first step in the algorithm is to compute the gray level histogram $h(GL_i)$ of the image to be thresholded. The algorithm is an iterative algorithm and the variance is calculated for corresponding threshold value T in the whole range of gray levels.

$$0 \leq T \leq 256$$

To illustrate the algorithm, assume a threshold value $T = 120$. The average gray level for the first class with set of pixels with gray levels below or equal to the threshold value T, is

$$c_1(T) = \sum_{i=0}^{T} i.h(GL_i).$$

$$c_1(120) = \sum_{i=0}^{120} i.h(GL_i).$$

The value $h(GL_i)$ is the histogram or frequency of occurrence of pixel gray level values in the image. For example, There are 7 pixels of gray level value 32 and the total number of background pixels is 20.

$$= [20x1+32x7+34x6+35x3+45x1+115x2]/20$$

$$=41.4$$

The average gray level for the second class with set of pixels with gray levels above the threshold value T, is

$$c_2(T) = \sum_{i=T+1}^{GL_{max}} i.h(GL_i).$$

$$=[121x2+122x1+232x2]/5$$

$$=165.6$$

The average gray level for the entire image is

$$c_T = \sum_{i=0}^{GL_{max}} i.h(GL_i).$$

$$=[20x1/25+32x7/25+34x6/25+35x3/25+45x1/25+115x2/25+121x2/25+$$

$$122x1/25+232x2/25]$$

$$=66.24$$

The probability of Class 1 occurring within the image is

$$P_1(T) = \sum_{i=0}^{T} h(GL_i).$$

=[20/25]

=0.8

While the probability of Class 2 occurring is

$$P_2(T) = \sum_{i=T+1}^{GL_{max}} h(GL_i).$$

=[5/25]

=0.2

The distance between the two classes is given by the variance:

$$\sigma^2 = \frac{\left[c_1(T).P_2(T) - c_2(T).P_1(T)\right]^2}{P_1(T).P_2(T)}$$

$$\sigma^2 = \frac{\left[(41.4)(0.2) - (165.6)(0.8)\right]^2}{(0.8)(0.2)}$$

=96410

The process is repeated for all values

$$0 \leq k \leq 256$$

The optimum threshold value k is the one that yields the maximum variance value σ^2 because the greatest distinction between classes is at the maximum variance value (Weeks 1996). Table 3.2 shows different threshold values and the resulting variance. For algorithm application all threshold values between 0 and 256 are applied and the one that yields the maximum variance is selected.

Table 3.2 Corresponding Variance for Different threshold Values

T	0	40	120	200	256
σ^2	0	31927.8	96410.3	58489.6	0

3.4.2 Algorithms Comparison

The pattern recognition for thresholding algorithm is more accurate in selecting the best threshold value because it examines all possible threshold values in the whole range of gray levels from 0 to 256. However, the optimum threshold algorithm applying the minimum probability of error classifier model is faster to execute. There are many adaptive threshold selection algorithms. The selection of the appropriate algorithm depends on many parameters such as the specific image processing application, image size, image quality, desired accuracy, hardware and software available, etc. (Weeks 1996; Russ 1995).

CHAPTER 4 NEURAL LEARNING

4.1 Introduction

The term "artificial intelligence" encompasses a wide range of technologies, which may include expert systems, neural networks, fuzzy logic systems, genetic algorithms, cellular automata, chaotic systems, and anticipatory systems (Tsoukalas and Uhrig 1997). Neural network technology is an accepted widely used field of artificial intelligence. Neural networks have biological basis as their design and functionality emulate the human brain.

Tsoukalas and Uhrig (1997) defined the artificial neural network as:

A data processing system consisting of a large number of simple, highly interconnected processing elements (artificial neurons) in an architecture inspired by the structure of the cerebral cortex of the brain.

Generally, neural networks are trained so that a particular input leads to a specific target output. Figure 4.1 illustrates the general functionality architecture for training. The network is adjusted, based on a comparison of the output and the target, until the output matches the target (ASCE 1997; Haykin 1999; kosko 1992; Tsoukalas and Uhrig 1997).
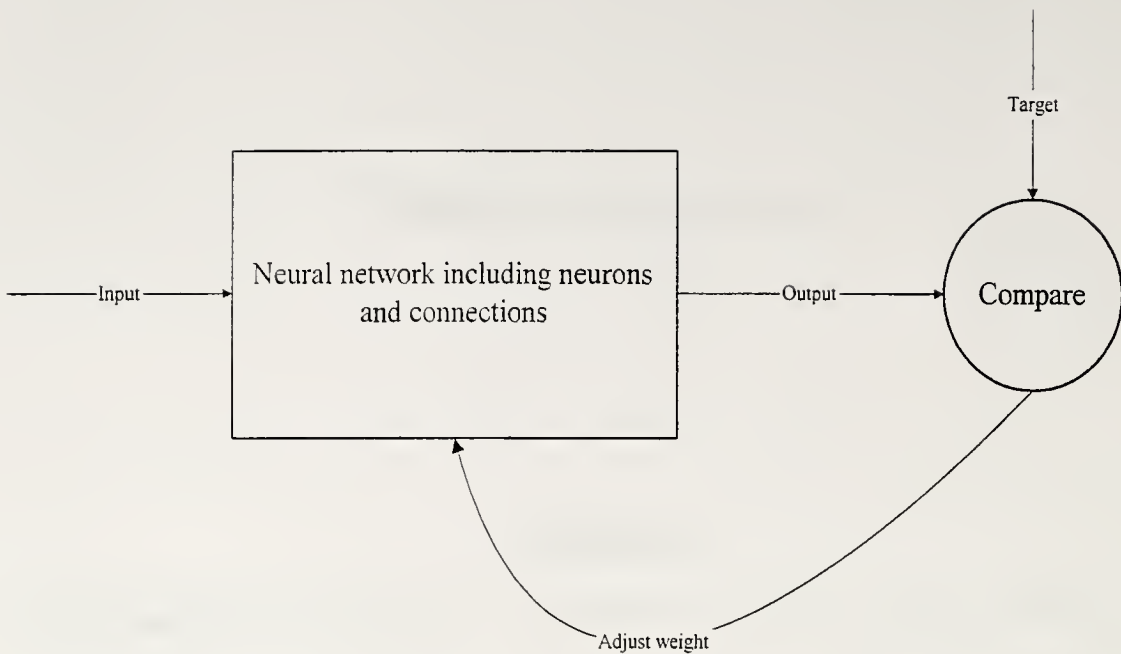
Figure 4.1 Neural Network Functionality

The neural network processing elements (neurons) are arranged in a sequence of layers with connections (weights) connecting the layers. Generally, there are three types of layers: input layer, hidden layer, and output layer. Typically, neural networks are fully connected, which means that all neurons in each layer are connected to each neuron in the next layer. Figure 4.2 shows the general architecture of a neural network (Haykin 1999).
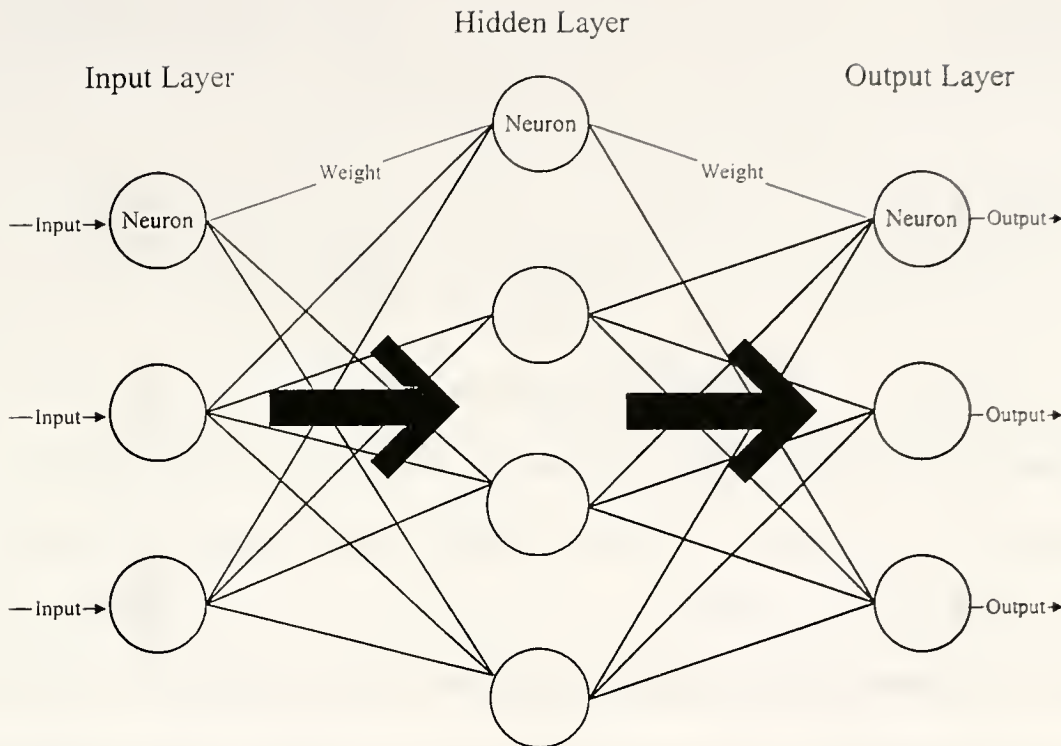
Figure 4.2 General Neural Network Architecture

The processing elements (neurons) process the data before transferring it to the next layer and all the neurons function simultaneously in a parallel mode. The neurons receive input signals from the previous layer after being modified by weights. The neurons have two parts. The first part sums the weighted input and the second part applies an activation function on the input. Figure 4.3 shows the architecture of a typical neuron where x_i is the neuron input, w_{ij} is the weight from i^{th} layer to the jth layer, and y_j is the neuron output (Tsoukalas and Uhrig 1997).
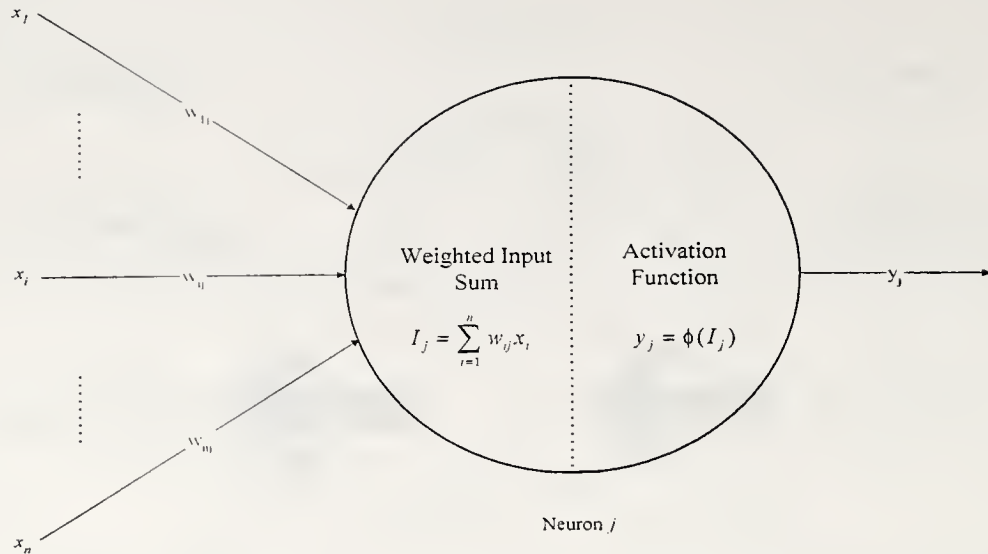
Figure 4.3 Neuron Architecture (Tsoukalas and Uhrig 1997)

The activation function is the function that each node applies to the input to compute its output. One of the most widely used activation functions is the Sigmoid Activation function. Figure 4.4 illustrates the graph of the Sigmoid Activation function.

$$f_1(x) = \frac{1}{1 + \exp(-x)}$$

Binary Sigmoid, range (0,1)
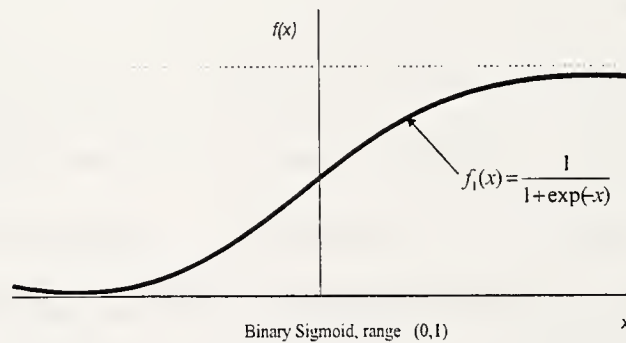
Figure 4.4 Binary Sigmoid Activation Function

Where:

$$f_1(x) = \frac{1}{1 + \exp(-x)}$$

(4.1)

with

(4.2)

$$f_1'(x) = f_1(x)[1 - f_1(x)]$$

4.2 Back-Propagation Algorithm

A popular artificial neural network model is the back propagation network algorithm. The basic back propagation model is a three-layered forward architecture. The first layer is the input layer, the second layer the hidden layer, and the third layer the output layer. Each layer contains a group of nodes that are linked together with nodes from other layers by connections among the nodes. Layers are connected only to the adjacent layers. The network is a feed-forward network, which means a neuron's output can only originate from a lower level, and a neuron's output can only be passed to a higher level. Figure 4.5 illustrates the basic structure of a feed forward neural network (Haykin 1999).
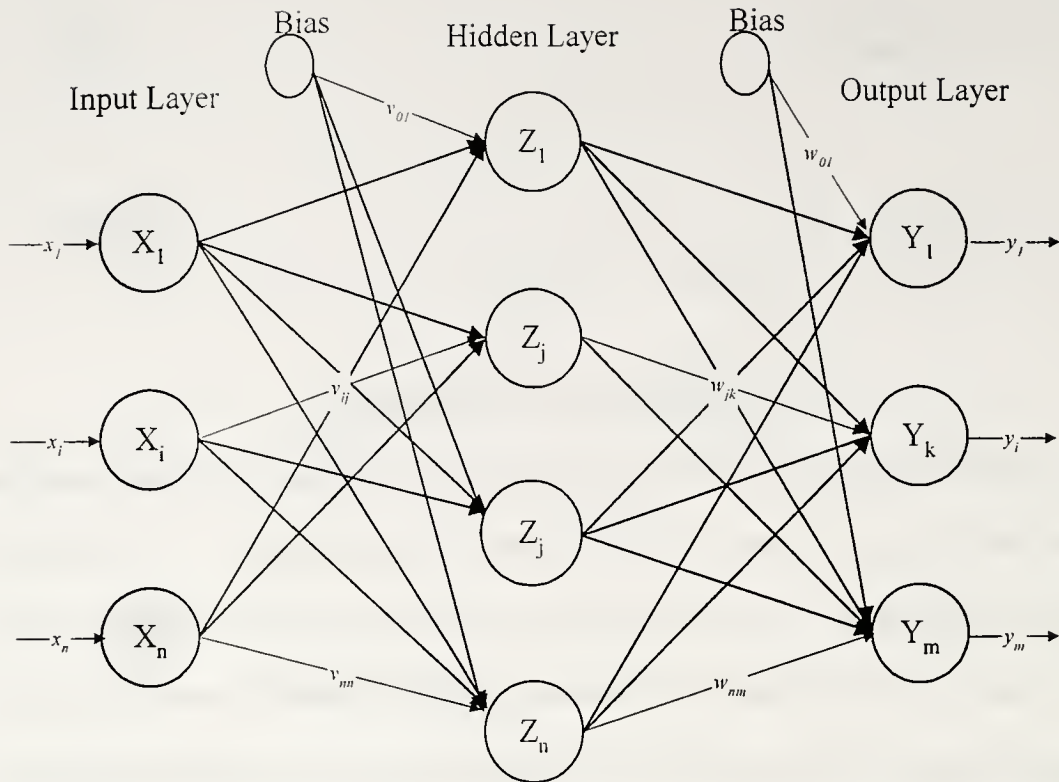
Figure 4.5 Neural Network for the Back-Propagation Algorithm

The input layer receives the features of the data that are entered into the neural network. If n feature values are to be entered into the input layer, then there must be n nodes, where n is the number of features supplied to the net. A single feature value is inputted into a single input node. The values are passed to the hidden layer through connections from the input layer. The nodes in the first layer distribute the individual inputs to all of the nodes in the hidden layer.

The hidden layer acts as the connection between the input layer and the output layer. The main function of the hidden layer is to process the input data during the network training

to connect to the output, i.e. to do the regression process in which the neural network tries to find correlation in the input data to predict the output (Haykin 1999).

The input layer does not operate on the feature data, but merely passes it to the hidden layer. Each node sums the n weighted inputs to the hidden layer. The value of the summation can be different for each node, due to different weight connections between the first and second layers. The function of the weights is to store knowledge during training by repeatedly adjusting themselves to reach minimum error desired. Additional weights are sometimes added and called biases.

The values that are summed in the hidden nodes are then passed to the nodes in the output layer via another set of weight connections. Each hidden node is connected with each output node. Although the hidden layer may actually consist of several hidden layers, most models' hidden layer consists of only one layer.

The output layer generates the output of the neural network. In the same way the hidden layer functions, the output layer's input values are summed, and the summation becomes the output value. Each output node corresponds to a desired output class. Each node sums the n weighted inputs in the hidden layer and the values from each output node are compared to each other (Haykin 1999).

These weighted inputs are summed and used as the node output value. Several output layers can exist for a back propagation network. Because the back-propagation algorithm is an iterative algorithm, it can be trained by adjusting the connection weights among the nodes.

The weights are recalculated after every complete cycle until the weights converge and the mean square error falls within the specified acceptable range.

4.2.1 Back-Propagation Training Algorithm

This section describes the mechanism of the back-propagation algorithm including the training algorithm and the algorithm application procedure after training. Figure 4.5 shows the basic architecture for a back-propagation network.

4.2.1.1 Notation

X Input training vector

$X = (x_1, x_2, \ldots, x_i, \ldots, x_n)$.

Y Output training vector

$Y = (y_1, y_2, \ldots, y_i, \ldots, y_n)$.

T Output target vector

$T = (t_1, t_2, \ldots, t_i, \ldots, t_n)$.

δ_k Portion of error correction weight adjustment for w_{jk} that is due to an error at output neuron Y_k; also, the information about the error at neuron Y_k that is propagated back to the hidden neurons that feed into neuron Y_k.

α Learning rate.

X_i Input neuron i: For an input neuron, the input signals and the output signals are the same, namely, x_i.

v_{0j} Bias on hidden neuron j.

w_{0j} Bias on output neuron j.

v_{ij} Weight from neuron i to neuron j.

w_{jk} Weight from neuron j to neuron k.

Z_j Hidden neuron j:

     The net input to Z_j is denoted z_in_j:

$$z\_in_j = v_{0j} - \sum x_i v_{ij} \tag{4.3}$$

     The output signal (activation) of Z_j is denoted z_j:

$$z_j = f(z\_in_j)$$

w_{0k} Bias on output neuron k.

Y_k Output neuron k:

     The net input to Y_k is denoted y_in_k:

$$y\_in_k = w_{0k} + \sum z_j w_{jk} \tag{4.4}$$

     The output signal (activation) of Y_k is denoted y_k:

$$y_k = f(y\_in_k)$$

4.2.1.2 Training Algorithm

Step 0.   Initialize weights. (Set to small random values).

Step 1.   While stopping condition is false, do steps 2-9.

Step 2.   For each training pair, do steps 3-8.

Feed-forward:

Step 3.   Each input neuron (X_i, $i = 1$, ..., n) receives input signal x_i and transfers

this signal to all neurons in the next layer (hidden layer).

Step 4.   Each hidden neuron (Z_j, $j = 1$, ..., n) sums its weighted input signals,

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \tag{4.5}$$

applies its activation function to compute its output signal,

$$z_j = f(z\_in_j) \tag{4.6}$$

and sends this signal to all neurons in the next layer (output layer).

Step 5.   Each output neuron (Y_k, $k = 1$, ..., m) sums its weighted input signals,

$$y\_in_k = w_{0k} + \sum_{j=1}^{p} z_j w_{jk} \tag{4.7}$$

and applies its activation function to compute its output signal,

$$y_k = f(y\_in_k) \tag{4.8}$$

Back-propagation of error:

Step 6.   Each output neuron (Y_k, $k = 1$, ..., m) receives a target pattern corresponding to the input training pattern, computes its error information term,

$$\delta_k = (t_k - y_k) f'(y\_in_k) \qquad (4.9)$$

calculates its weight correction term (used to update w_{jk} later),

$$\Delta w_{jk} = \alpha \delta_k z_j \qquad (4.10)$$

calculates its bias correction term (used to update w_{0k} later),

$$\Delta w_{0k} = \alpha \delta_k \qquad (4.11)$$

and sends δ_k to the neurons in the preceding layer.

Step 7.   Each hidden neuron (Z_j, $j = 1$, ..., p) sums its delta inputs (from neurons in the next layer),

$$\delta\_in_j = \sum_{k=1}^{m} \delta_k w_{jk} \qquad (4.12)$$

multiplies by the derivative of its activation function to calculate its error information term,

$$\delta_j = \delta\_in_j f'(z\_in_j) \qquad (4.13)$$

calculates its weight correction term (used to update v_{ij} later),

$$\Delta v_{ij} = \alpha \delta_j x_i \qquad (4.14)$$

and calculates its bias correction term (used to update v_{0j} later).

$$\Delta v_{0j} = \alpha \delta_j \qquad (4.15)$$

Update weights and biases:

Step 8: Each output neuron (Y_k, $k = 1, ..., m$) updates its bias and weights ($j = 0, ..., p$):

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk} \qquad (4.16)$$

Each hidden neuron (Z_j, $j = 1, ..., p$) updates its bias and weights ($i = 0, ..., n$):

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij} \qquad (4.17)$$

Step 9. Test stopping condition.

Algorithm Application Procedure

Step 0. Initialize weights (Set to small random values).

Step 1. For each input vector, do steps 2-4.

Step 2. For $i = 1, ..., n$: set activation of input neuron x_i,

Step 3. For $j = 1, ..., p$:

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij} \qquad (4.18)$$

$$z_j = f(z\_in_j) \tag{4.19}$$

Step 4. For $k = 1, ..., m$:

$$z\_in_k = w_{0k} + \sum_{j=1}^{p} z_j w_{jk} \tag{4.20}$$

$$y_k = f(y\_in_k) \tag{4.21}$$

4.2.2 Example Using Training Algorithm

Consider the network shown in Figure 4.6. It is a three-layer network with input, hidden, and output layers. The initial input, output, weights and bias are given as shown in Table 4.1 and Figure 4.6 and the learning rate is 1.

Table 4.1 Training data for Example Neural Network

#	Input	Output
1	-3	0.4
2	2	0.8
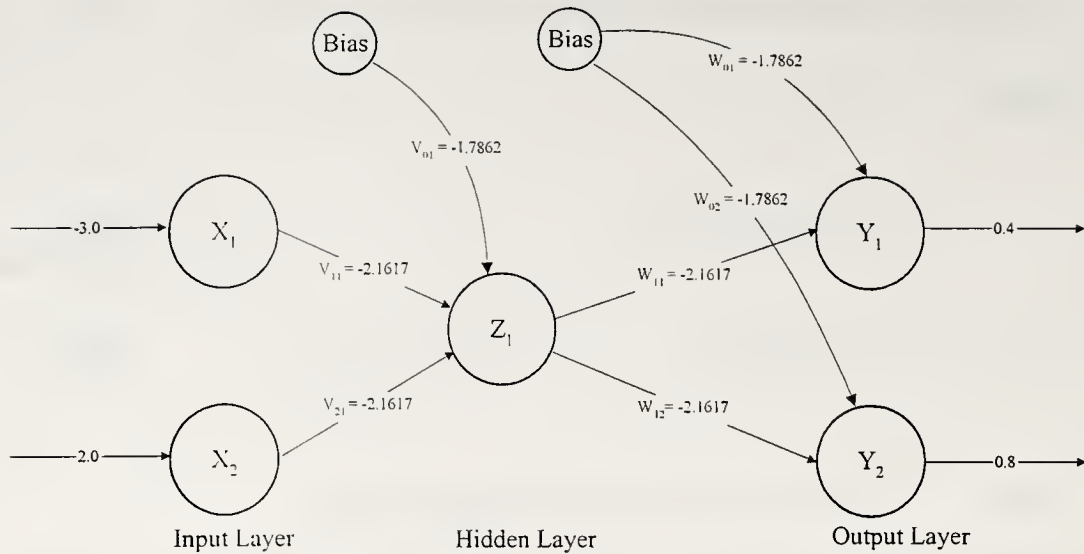
Figure 4.6 Example Neural Network

4.2.2.1 Training Algorithm

Step 0.   Initialize weights and biases (random numbers).

$v_{ij} = -2.1617$

$v_{0j} = -1.7862$

$w_{ik} = -2.1617$

$w_{0k} = -1.7862$

Step 1.   While stopping condition is false, do steps 2-9.

Step 2.   For each training pair, do steps 3-8.

Feed-forward:

Step 3.   Each input neuron (X_i, $i = 1, ..., n$) receives input signal x_i and transfers

this signal to all neurons in the next layer (hidden layer).

Step 4.   Each hidden neuron (Z_j, $j = 1, ..., n$) sums its weighted input signals,

$$z\_in_j = v_{0j} + \sum_{i=1}^{n} x_i v_{ij}$$

$z\_in_j = -1.7862 + (-3)(-2.1617) + (2)(-2.1617) = 0.3755$

applies its activation function to compute its output signal,

$$z_j = f(z\_in_j)$$

The activation function is:

$$f(z\_in_j) = \frac{1}{1 + \exp(-(z\_in_j))}$$

$$z_1 = \frac{1}{1 + \exp(-(0.3755))} = 0.5928$$

and sends this signal to all neurons in the next layer (output layer).

Step 5.   Each output neuron (Y_k, $k = 1, ..., m$) sums its weighted input signals,

$$y\_in_k = w_{0k} + \sum_{j=1}^{p} z_j w_{jk}$$

$y\_in_1 = -1.7862 + (0.5928)(-2.1617) = -3.0677$

$$y\_in_2 = -1.7862 + (0.5928)(-2.1617) = -3.0677$$

and applies its activation function to compute its output signal,

$$y_k = f(y\_in_k)$$

$$f(y\_in_k) = \frac{1}{1 + \exp(-(y\_in_k))}$$

$$y_1 = \frac{1}{1 + \exp(-(-3.0677))} = 0.5$$

$$y_2 = \frac{1}{1 + \exp(-(-3.0677))} = 0.5$$

Back-propagation of error:

Step 6.   Each output neuron $(Y_k, \ k = 1, \ ..., \ m)$ receives a target pattern corresponding to the input training pattern, computes its error information term,

$$\delta_k = (t_k - y_k)f'(y\_in_k)$$

$$\delta_1 = (0.4 - 0.5)\big[f(y\_in_1)(1 - f(y\_in_1))\big]$$
$$\delta_1 = (-0.1)(-3.6077)(1 - (-3.6077)) = 1.4675$$
$$\delta_2 = (0.8 - 0.5)\big[f(y\_in_2)(1 - f(y\_in_2))\big]$$
$$\delta_2 = (0.3)(-3.6077)(1 - (-3.6077)) = -4.987$$

calculates its weight correction term (used to update w_{jk} later),

$$\Delta w_{jk} = \alpha \delta_k z_j$$

$$\Delta w_{11} = (1)(1.4675)(0.5928) = 0.8699$$
$$\Delta w_{12} = (1)(-4.987)(0.5928) = -2.9563$$

calculates its bias correction term (used to update w_{0k} later),

$$\Delta w_{0k} = \alpha \delta_k$$

$$\Delta w_{01} = (1)(1.4675) = 1.4675$$
$$\Delta w_{02} = (1)(-4.987) = -4.987$$

and sends δ_k to the neurons in the preceding layer.

Step 7.　Each hidden neuron (Z_j, $j = 1, ..., p$) sums its delta inputs (from neurons in the next layer),

$$\delta\_in_j = \sum_{k=1}^{m} \delta_k w_{jk}$$

$$\delta\_in_1 = (1.4675)(-2.1617) + (-4.987)(-2.1617) = 7.6081$$

multiplies by the derivative of its activation function to calculate its error information term,

$$\delta_j = \delta\_in_j f'(z\_in_j)$$

$$\delta_1 = (7.6081)\left[f(z\_in_1)(1 - f(z\_in_1)) \right]$$
$$\delta_1 = (7.6081)(0.3755)(1 - 0.3755) = 1.7841$$

calculates its weight correction term (used to update v_{ij} later),

$$\Delta v_{ij} = \alpha \delta_j x_i$$

$$\Delta v_{11} = (1)(1.7841)(-3.0) = -5.3523$$
$$\Delta v_{21} = (1)(1.7841)(2.0) = 3.5682$$

and calculates its bias correction term (used to update v_{0j} later).

$$\Delta v_{0j} = \alpha \delta_j$$

$$\Delta_{01} = (1)(1.7841) = 1.7841$$

Update weights and biases:

Step 8:   Each output neuron (Y_k, $k = 1, ..., m$) updates its bias and weights ($j = 0, ..., p$):

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk}$$

$$w_{01}(new) = -1.7862 + 1.4675 = -0.3187$$
$$w_{02}(new) = -1.7862 + (-4.987) = -6.7732$$
$$w_{11}(new) = -2.1617 + (0.8699) = -1.2918$$
$$w_{12}(new) = -2.1617 + (-2.9563) = -5.118$$

Each hidden neuron (Z_j, $j = 1, ..., p$) updates its bias and weights ($i = 0, ..., n$):

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$

$$v_{01}(new) = -1.7862 + 1.7841 = -0.0021$$
$$v_{11}(new) = -2.1617 + (-5.3523) = -7.514$$
$$v_{12}(new) = -2.1617 + 3.5682 = 1.4065$$

Step 9.   Test stopping condition.

Calculate sum of the squared error (*SSE*):

$$(error_1)^2 = (t_1 - y_1)^2 = (0.4 - 0.5)^2 = 0.01$$
$$(error_2)^2 = (t_2 - y_2)^2 = (0.8 - 0.5)^2 = 0.09$$
$$SSE = 0.01 + 0.09 = 0.1$$

The algorithm is repeated until reaching desired error. Figure 4.7 shows the network after 1 epoch (cycle).
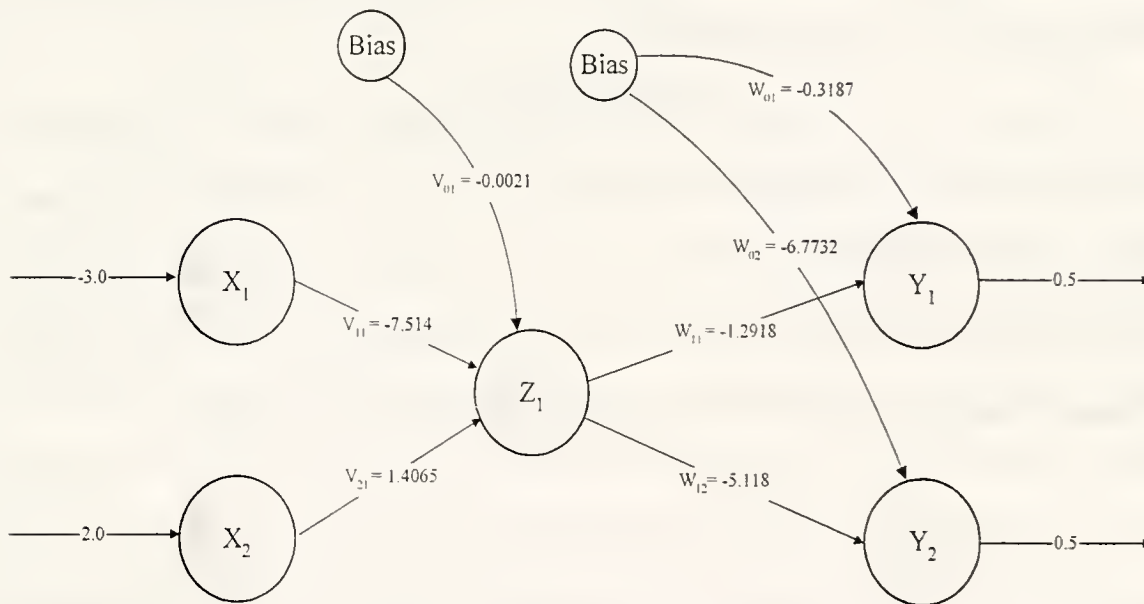
Figure 4.7 Example Network After 1 Training Cycle

4.3 Summary

The strength of neural networks stems in the fact that they learn from example by processing data, including input and output, and adjusting the weights accordingly. The neural network store knowledge throughout the weights to form a distributed associative memory structure. The term associative memory means the network memory can make up

for missing or partial inputs. This very fact makes neural network fault-tolerant and capable of processing noisy and incomplete data.

Neural networks have been applied successfully for pattern recognition and image processing applications (Bunke 1992; Croall and Mason 1992; De La Blanka 1992; Fu and Xu 1998; Garris et. al 1998). In this research neural networks are used for recognition and classification of defects in images for the purpose of construction quality assessment and specifically steel bridge coatings assessment.

The neural network is trained to identify defect patterns and classify the image into two classes: defects and non-defects. The purpose of the neural network is to produce consistent results even when having noisy or lower quality images. The learning method used is supervised learning in order to train the network from example to try to simulate human experience in classifying images with different parameters and different scenarios.

CHAPTER 5 THE HYBRID MODEL

5.1 The Hybrid Model

The basic concept of the model is to acquire digital images of objects to be assessed to identify and measure defects using image processing techniques. Moreover, neural networks are used to train the system to automate the process and simulate experts' knowledge in identifying the defects and using this knowledge to later assess other coatings. The model was trained based on 100 rust images through neural networks. The trained model can get the exact rust percentages of the 100 rust image samples and can further assess the rust percentage of new images. Neural networks can also make the classification model more fault-tolerant because of their capability of accommodating for noisy or low quality images. In this research, the model was trained to recognize rust through the use of neural networks. Effects from other parameters, such as cracking, blistering, scratching, contamination, etc., will be minimized when conducting image processing.

Figure 5.1 illustrates the general architecture of the system. The first phase is the data acquisition, which is obtained from various sources: the design specifications, the objects' digital images, and the experts' knowledge. Data is either directly passed for analysis or

stored in a database for later retrieval. The pre-processing and analysis parts, which are the main parts, utilize the computer for data processing and analysis. Pattern recognition and computer learning abilities are used in the analysis phase. The statistical plan is integrated with data acquisition and analysis. Assessment results and recommendations are integrated with both the analysis phase and the knowledge database.
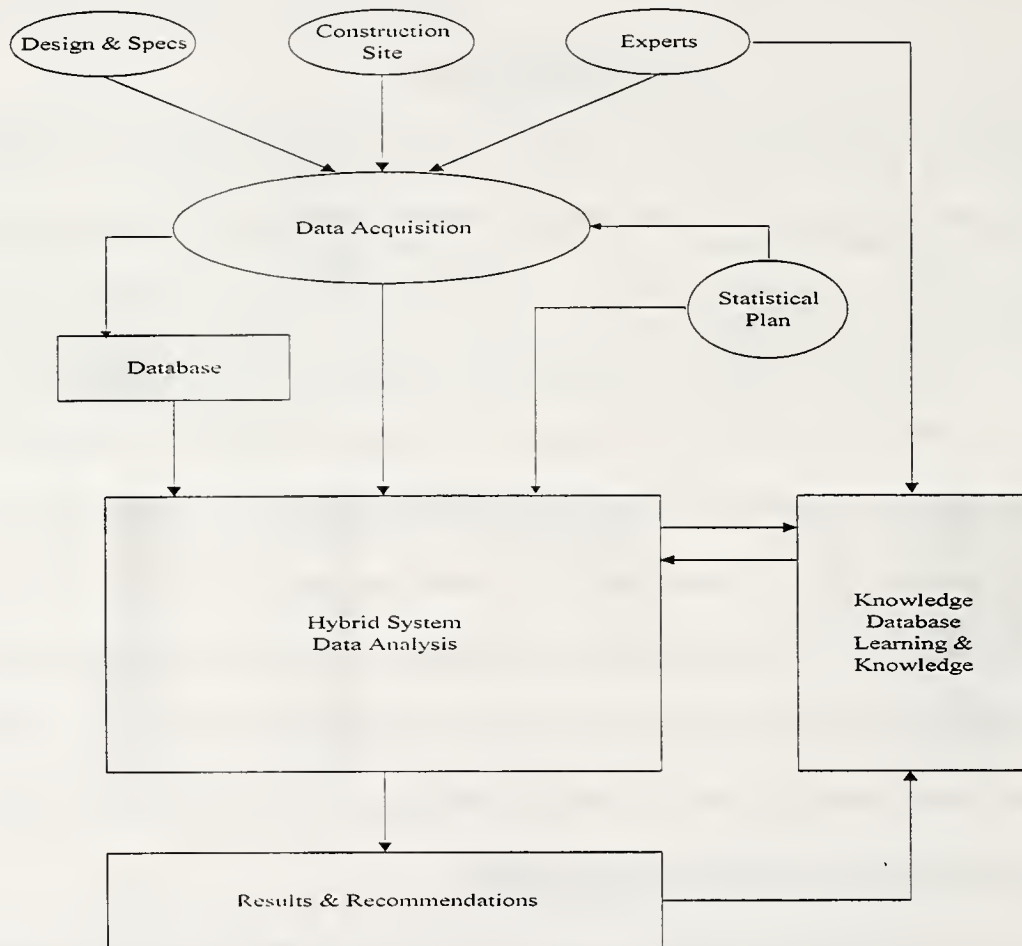
Figure 5.1 Hybrid System Architecture

Figure 5.2 illustrates the working system model. It consists of four main modules.
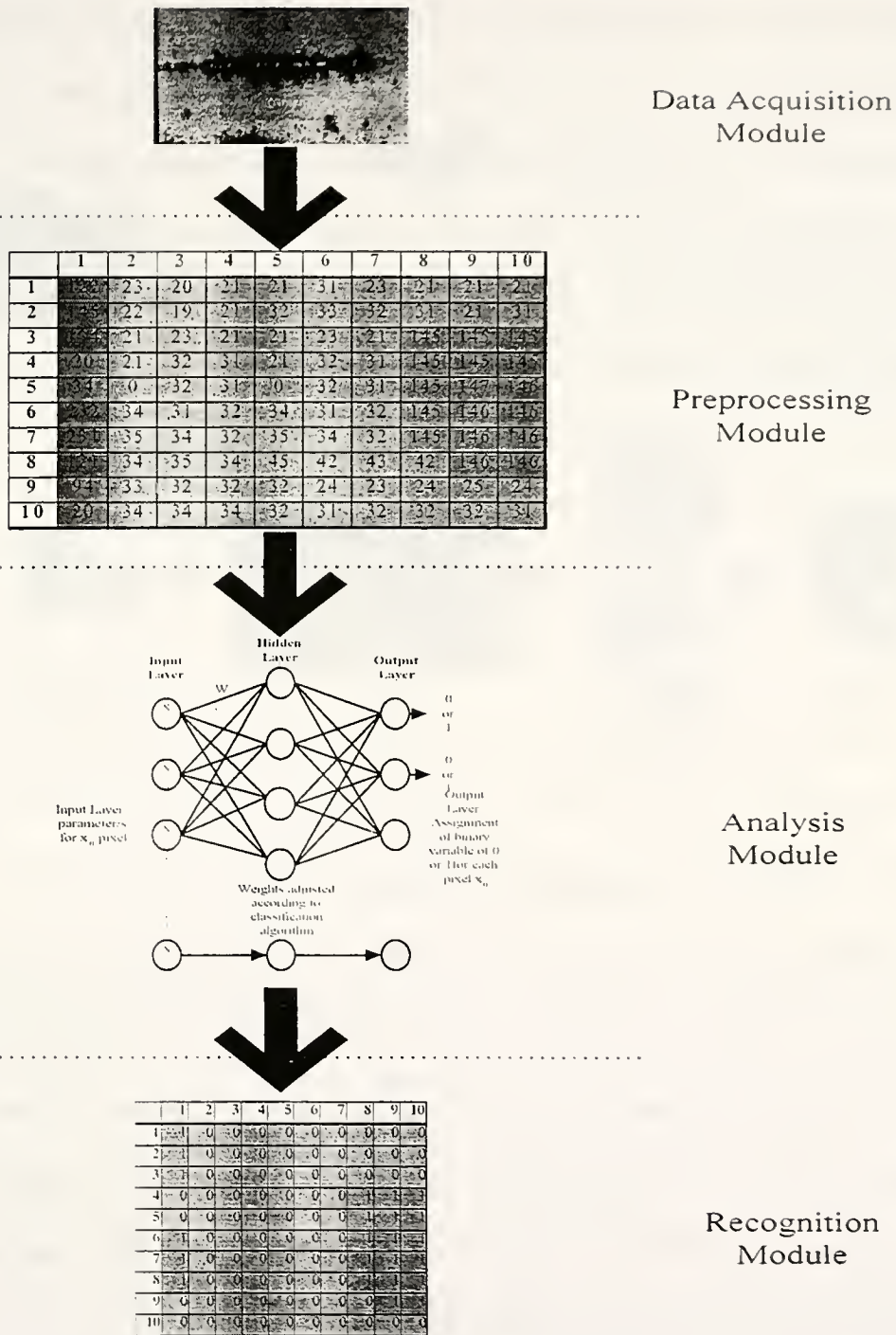
Figure 5.2 The Hybrid System Working Model

The first module is data acquisition. In this module, digital images of the constructed facility (objects) are obtained using a digital camera. A statistical sampling plan will be used for image acquisition selection and quantity. The plan will determine the number of sample images to be taken and the location of the sample images. After acquisition the images are transferred to the computer on site or in a remote office via any communication protocol. This module is illustrated in Figure 5.3.

Digital Camera

Constructed facility

Acquire images

Transfer images to computer
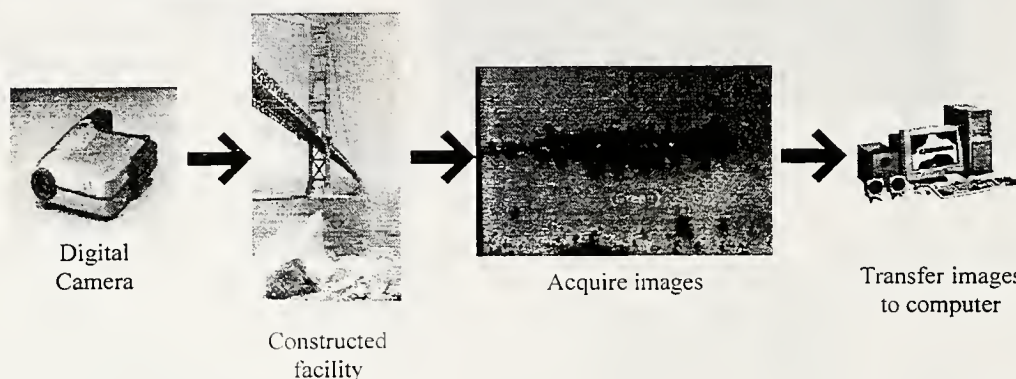
Figure 5.3 Data Acquisition Module

The next stage is the preprocessing module, where image analysis techniques may be used to analyze and enhance the image by applying algorithms such as filtering and noise reduction. Moreover, image pre-processing is used to obtain the parameters of the image such as the gray level in a numerical format. At this stage a statistical pattern recognition and image segmentation algorithms are utilized to identify defects according to the numerical representation of images. Figure 5.4 illustrates the pre-processing module.

Digital Image

Pre-processing Operations
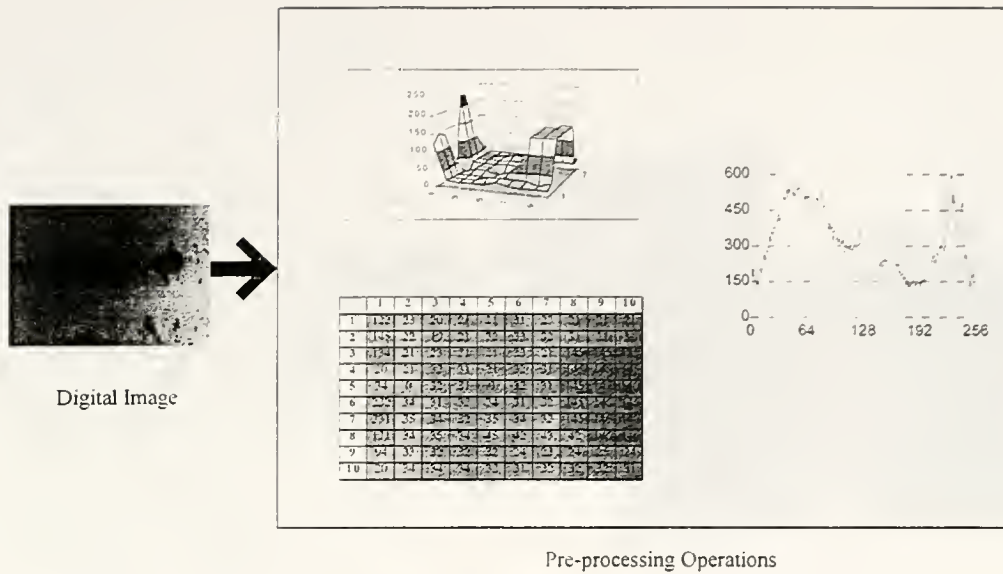
Figure 5.4 Pre-processing Module

The next stage is the processing stage. After being trained, neural networks are used to identify defects in the images by assigning binary variable of 0 or 1 for each pixel in the image. During the network training, the neural network is trained with different images and their parameters such as the pixels' gray levels. The network is also supplied mapped values of 0 or 1 for each pixel value depending on whether it is defect or not. The network will learn to assign the binary variable 0 or 1 for different scenarios according to image parameters. The network will assign a value of 1 for the objects (defects) and a value of 0 for the background (non-defect). Figure 5.5 illustrates the analysis module.

The statistical plan is integrated with the analysis module. The errors of assigning objects (defects) as background (non-defect) and vise-versa can be calculated. Hence, the model reliability can be estimated. Furthermore, the neural network acceptable error goal can be determined during training.

Pre-processed Image

Recognition Neural Network
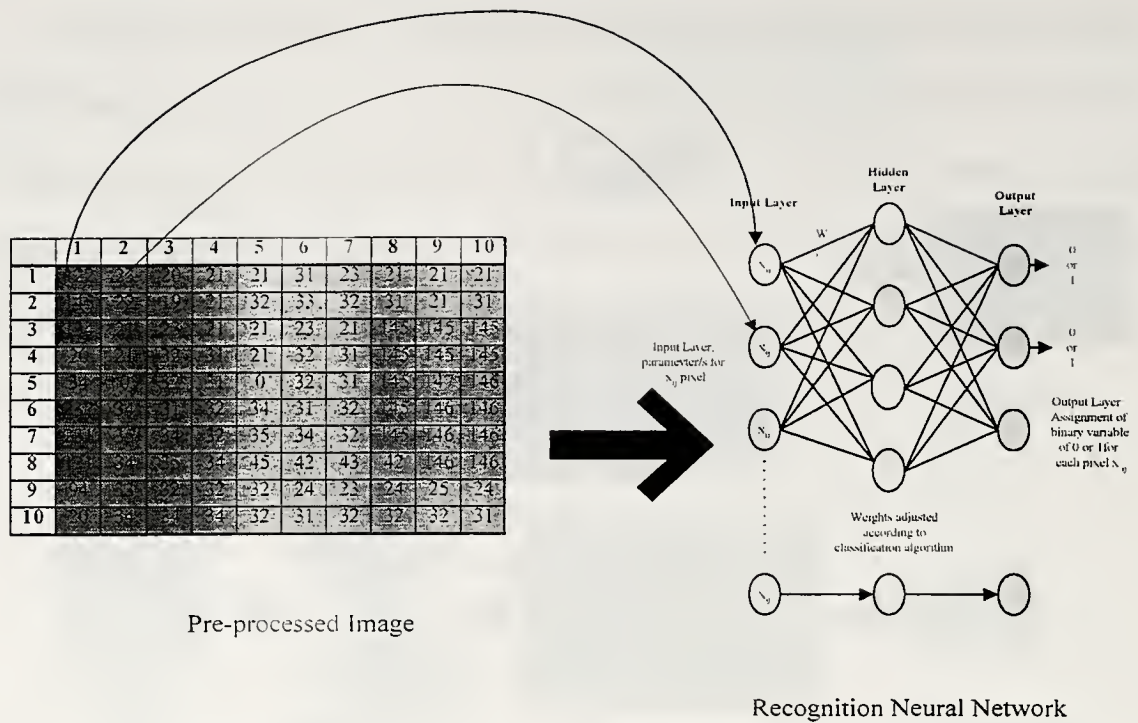
Figure 5.5 Analysis Module

The final stage is the recognition module, where quantitative measures of defects are obtained from the output of the previous stage. From mapped output of the neural network, the whole image is represented as 0's or 1's. The 1 values represent the defect; hence, defects can be identified and quantitatively measured as a percentage of the whole area. The system can be trained to identify different types of defects according to the specific application. The recognition module is shown in Figure 5.6.
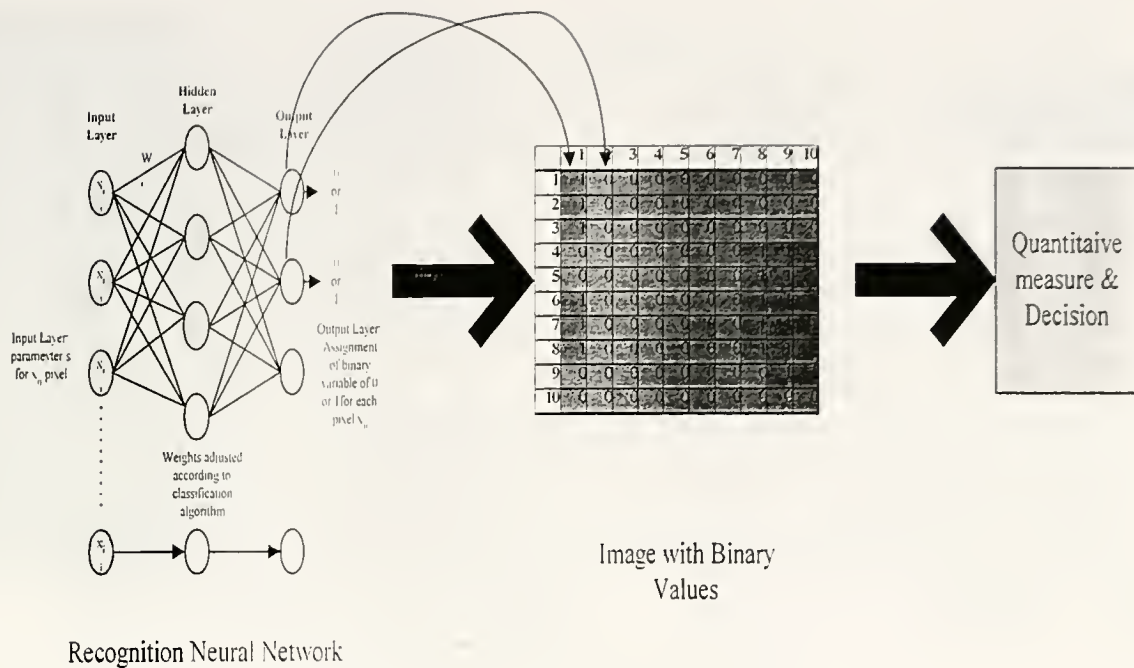
Figure 5.6 Recognition Module

5.2 Images Classification and Measurement

A good example of recognition for image analysis is a method used for visual character recognition. The same concept can be applied for pattern recognition in any image. The computer processor can distinguish 256 different shades of gray from 0 to 255 for each pixel of the image. An example of this is the recognition of numerical digits in a rectangular box. The rectangle contains 16 pixel rows by 13 pixel columns, for a total of 208 pixels, where each is 1 or 0 is for black or white, respectively. Figure 5.7 represents the digit "7" and illustrates the concept of pattern recognition in images.

1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	0	1	1	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0
0	0	0	0	0	1	1	0	0	0	0	0	0
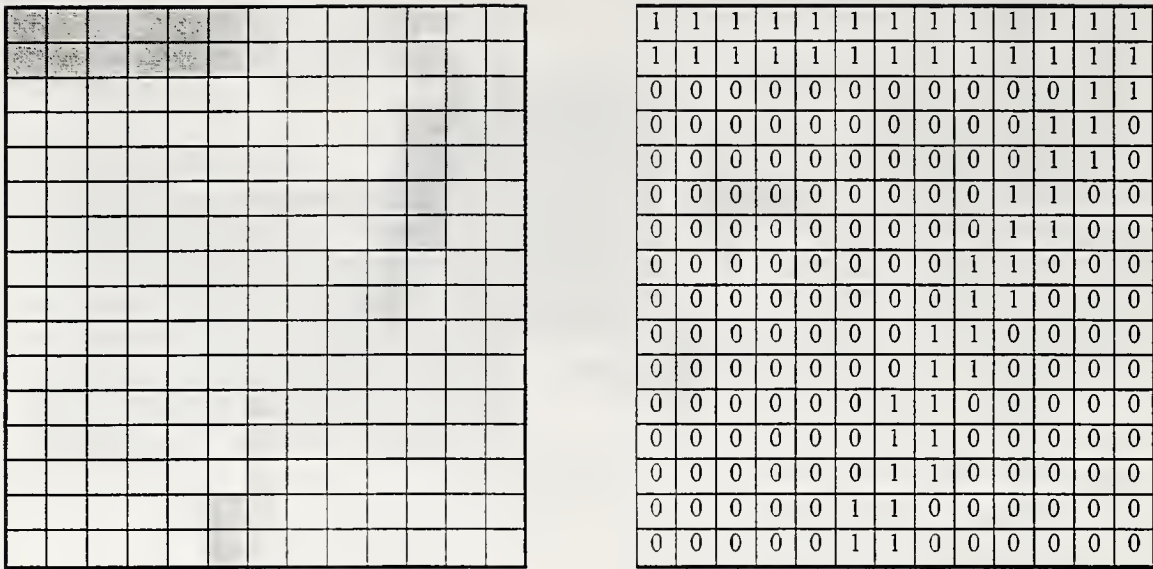
Figure 5.7 Visual Recognition of Digits

The rectangle on the left represents the pixels and the rectangle on the right represents the corresponding pixel values. This application is useful for recognizing patterns with different gray level. For example in coating assessment, rusted areas versus good areas can be presented in terms of 0's or 1's. This can be achieved by assigning a gray level benchmark (threshold) for rust and hence any level above that threshold would be assigned 1, (rust) and any level below that level would be assigned 0, (non-defect). All gray levels below the threshold value, say 120, are converted into 0 (background) and everything from 120 to 255 into 1 (object), so that a single bit (0 or 1) represents a pixel. Figure 5.8 illustrates how this concept is applied.
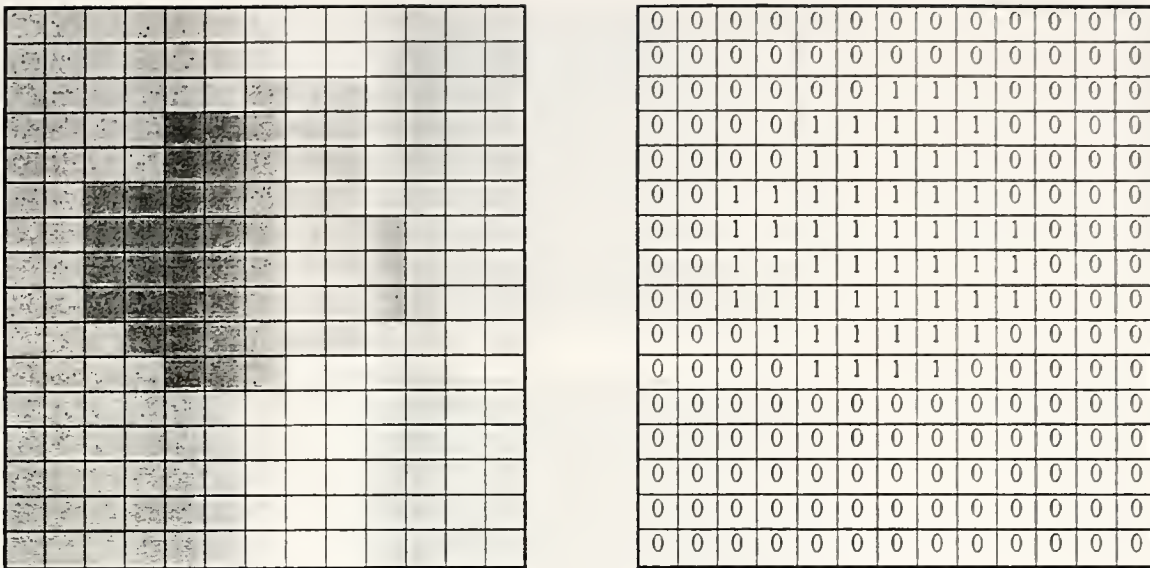
Figure 5.8 Visual Recognition of Rust

In Figure 5.8 there are 208 pixels (13 columns and 16 rows) and there are 54 pixels with a value of 1. Hence the rust area percentage is 26%. The same technique will be applied to real images with much higher number of pixels but the concept is the same. The intelligent system will assign each pixel a binary variable of 0 or 1 and calculate the percentage of defect accordingly.

5.3 Model Application Example

The following example illustrates how a simple (5 pixel x 5 pixel) image could be used to illustrate the methodology of the hybrid system. Consider the image in Figure 5.9 to represent a part of a highway steel beam and the beam coating is to be assessed. The image numerical pixel values are also illustrated in Figure 5.9.

5	232	34	31	32	110
4	231	35	34	32	35
3	135	34	35	118	45
2	94	33	32	32	32
1	20	34	115	34	32
	1	2	3	4	5
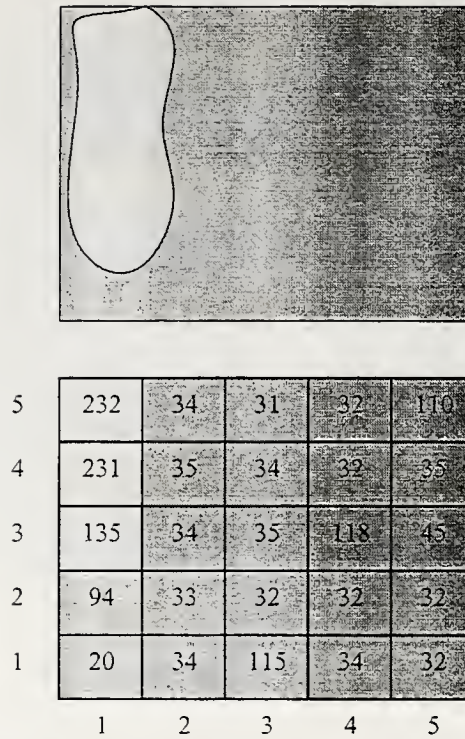
Figure 5.9 Example Image with Pixel Values

Step 1 Data Acquisition

A digital image of a constructed facility object is acquired. In this example a digital image of a steel bridge coating area is acquired and transferred to the computer.

Step 2 Preprocessing

Image analysis software is used to enhance the image and to obtain image representation in numerical format. Figure 5.9 shows the sample image with the gray level shown for each pixel. Optimum threshold algorithm is applied on the resulting classified image to determine the threshold value for analysis. The image shown in Figure 5.9 is used for algorithm application. The algorithm is described in detail in section 3.3.1.

The probability of object pixels

$$P_o = \frac{n_o}{n_t}$$

$= 3/25 = 0.12$

The probability of background pixels

$$P_b = \frac{n_b}{n_t}$$

$= 22/25 = 0.88$

Consider the object pixels have a mean gray level of GL_o, and the background pixels have a mean gray level of GL_b.

The mean gray level for object pixels

$GL_o = (232+231+135)/3 = 199.3$

The mean gray level for background pixels

$GL_b =$

$(94+20+34+35+34+33+34+31+34+35+32+115+32+32+118+32+34+110+35+45+32$

$+32)/22$

$= 46.8$

Optimum threshold value

$$T = \frac{GL_o + GL_b}{2} + \frac{\sigma^2}{GL_o - GL_b} \ln\left(\frac{P_b}{P_o}\right)$$

Assume that the variance of the distribution equals 1 as described in section 3.3.2. However, the value of the variance doesn't affect the threshold vale by a large amount.

$$T = \frac{199.3 + 46.8}{2} + \frac{1^2}{199.3 - 46.8} \ln\left(\frac{0.88}{0.12}\right)$$

$$= 123.55$$

A threshold value of 124 will be used to classify the image as shown in Figure 5.10.

	1	2	3	4	5
5	1	0	0	0	0
4	1	0	0	0	0
3	1	0	0	0	0
2	0	0	0	0	0
1	0	0	0	0	0

Figure 5.10 Example Image After Final Classification

Note that the resulting classified image is the same as the initial classified image. This is due to the image simplicity but it more likely that real images with more complexity will have changes after applying thresholding algorithm. The resulting classified images will be used for the neural network training in the analysis stage.

Step 3 Analysis

Data from the image shown in Figure 5.10 will be used to train the neural network. The purpose of using the neural network is to generate consistent results when having noisy or lower quality images. The learning method used is supervised learning in order to train the

network from example to try to automate the process and simulate human experience in classifying images with different parameters and different scenarios.

During training, some human interference might be required, specifically in threshold selection for defect identification because of different parameters such as image quality, external factors in images such as dirt, etc. Adjustments are made using experts' knowledge and then the neural network is trained in order to automate the process in the future.

The complete neural network will consist of three layers: the input layer, the hidden layer, and the output layer. The input layer consists of the image data and has four elements: The x-axis location of the pixel, then the y-axis location of the pixel, the value of the gray level of the pixel, and finally the difference between the gray level of the pixel and the predetermined threshold value. In this example, the predetermined threshold is 124 calculated in step 2. Table 5.1 shows the data entry for the neural network.

Table 5.1 Neural Network Data Processing

Input #	Location		Parameter 1 (Grey Level)	Parameter 2 (Difference from Image Threshold: 124)	Output (Binary Variable)
	(x)	(y)			
1	1	1	20	104	0
2	1	2	94	30	0
3	1	3	135	-11	1
4	1	4	231	-107	1
5	1	5	232	-108	1
6	2	1	34	90	0
7	2	2	33	91	0
8	2	3	34	90	0
9	2	4	35	89	0
10	2	5	34	90	0
11	3	1	115	9	0
12	3	2	32	92	0
13	3	3	35	89	0
14	3	4	34	90	0
15	3	5	31	93	0
16	4	1	34	90	0
17	4	2	32	92	0
18	4	3	118	6	0
19	4	4	32	92	0
20	4	5	32	92	0
21	5	1	32	92	0
22	5	2	32	92	0
23	5	3	45	79	0
24	5	4	31	93	0
25	5	5	110	14	0

Weights and biases are used by the network to store the knowledge acquired during training by repeatedly adjusting themselves to reach the minimum desired error. The initial values for weights and biases are chosen randomly but there are some heuristic rules guidelines for that choice. Generally, it is important to avoid choices of initial weights and biases that would make it likely that either activation or derivative of activation is zero. Otherwise, the network will not converge or reach the maximum training desired.

Step 4. Recognition

After training, the neural network will generate the output to classify the image's pixels into defects and non-defects. The same example was used to train another network with an additional set of inputs. The additional set of inputs is a membership function, in which each pixel is evaluated on how close its value is to the gray level threshold. For the output assignment of the binary variable, any membership above 95% was considered defect and assigned 1.

The percentage to be assigned can be flexible and depends on how strict the assessment process is. Data entry for the neural network is the same as in the previous example with the addition of the membership function element. Data entry is shown in table 5.2. Note that values above the threshold value of 124 were considered, logically, to have a membership function of 1 because they are classified as objects beforehand.

Table 5.2 Neural Network Data Input (With Memberships)

Input #	Location		Parameter 1 (Grey Level)	Parameter 2 (Difference from Image Threshold: 124)	Membership Function (124 – Parameter 1)/ 124	Output (Binary Variable)
	(x)	(y)				
1	1	1	20	104	0.838709677	0
2	1	2	94	30	0.241935484	0
3	1	3	135	-11	1	1
4	1	4	231	-107	1	1
5	1	5	232	-108	1	1
6	2	1	34	90	0.725806452	0
7	2	2	33	91	0.733870968	0
8	2	3	34	90	0.725806452	0
9	2	4	35	89	0.717741935	0
10	2	5	34	90	0.725806452	0
11	3	1	115	9	0.072580645	0
12	3	2	32	92	0.741935484	0
13	3	3	35	89	0.717741935	0
14	3	4	34	90	0.725806452	0
15	3	5	31	93	0.75	0
16	4	1	34	90	0.725806452	0
17	4	2	32	92	0.741935484	0
18	4	3	118	6	0.048387097	0
19	4	4	32	92	0.741935484	0
20	4	5	32	92	0.741935484	0
21	5	1	32	92	0.741935484	0
22	5	2	32	92	0.741935484	0
23	5	3	45	79	0.637096774	0
24	5	4	31	93	0.75	0
25	5	5	110	14	0.112903226	0

5.4 Model Reliability

The model reliability can be measured by calculating the error it produces in recognizing defects. There are two probable types of errors. First, the error in classifying defective parts as non-defective parts, which is called owner's risk, or β. The other type is classifying non-defective parts as defects, which is called producer's risk, or α. Both types of

errors can be calculated and the sum of both errors would constitute the total error in the model.

Consider the same simple image that was used as an example to illustrate the model. The image represents a part of a steel beam and the beam coating is to be assessed. The image numerical pixel values are also illustrated in Figure 5.11.

5	232	34	31	32	110
4	231	35	34	32	35
3	135	34	35	118	45
2	94	33	32	32	32
1	20	34	115	34	32
	1	2	3	4	5
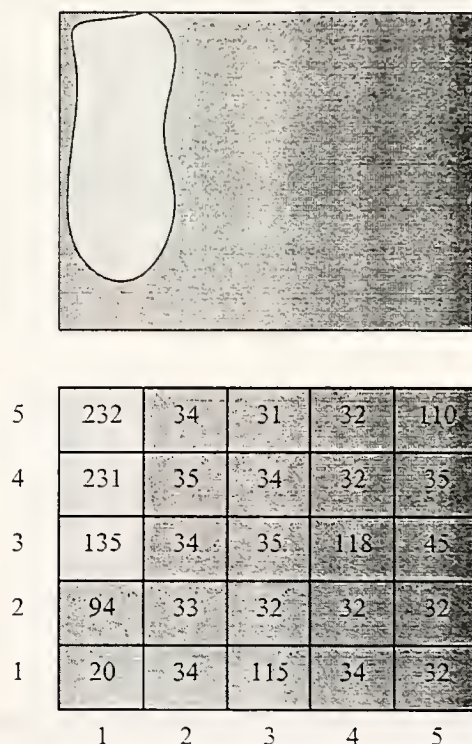
Figure 5.11 Example Image with Pixel Values

The image histogram is shown in Figure 5.12. The histogram represents the frequency of occurrence for each of the gray levels of the image's pixels. The histogram shows that the image gray level's frequencies have two peaks and two valleys. The peak frequencies are at

gray level values of "32" and "34" and the valley frequencies at gray level values of "33" and "45-232."
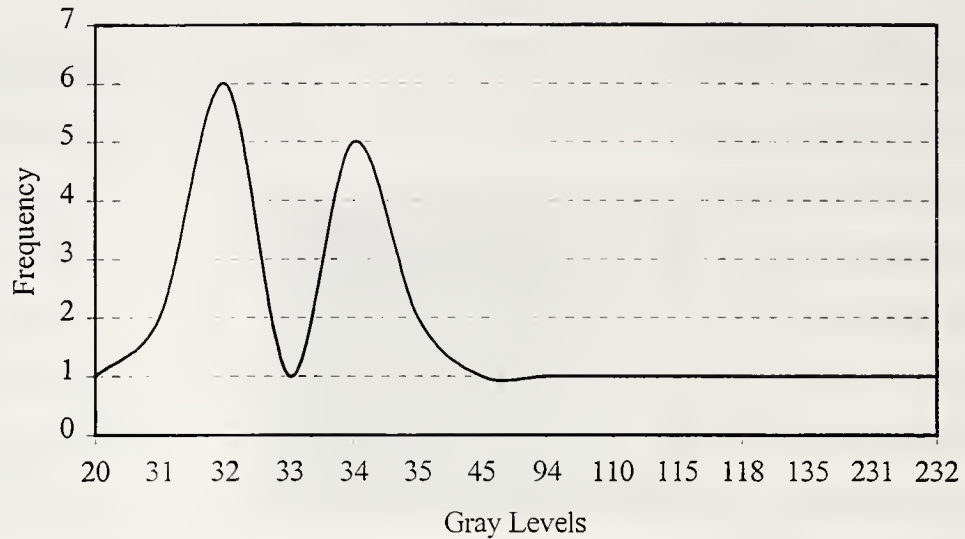
Figure 5.12 Coating Image Histogram

5.4.1 Error calculation

From the model procedure as described in section 5.3, the image was classified as shown in Figure 5.13. The image pixels were mapped to either 0 or 1 based on the classification whether the pixel represents defect or non-defect areas.

5	1	0	0	0	0
4	1	0	0	0	0
3	1	0	0	0	0
2	0	0	0	0	0
1	0	0	0	0	0
	1	2	3	4	5

Figure 5.13 Classified Coating Image

The probability of object pixels

$$P_o = \frac{n_o}{n_t}$$

= 3/25 = 0.12

The probability of background pixels

$$P_b = \frac{n_b}{n_t}$$

= 22/25 = 0.88

Consider the object pixels have a mean gray level of G_o, and the background pixels have a mean gray level of G_b.

The mean gray level for object pixels

$G_o = (232+231-135)/3 = 199.3$

The mean gray level for background pixels

$G_b=$

(94+20+34+35+34+33+34+31+34+35+32+115+32+32+118+32+34+110+35+45+32

+32)/22

=46.8

Optimum threshold value

$$T = \frac{GL_o + GL_b}{2} + \frac{\sigma^2}{GL_o - GL_b} \ln\left(\frac{P_b}{P_o}\right)$$

Assume that the variance of the distribution equals 1 as described in section 3.3.1.

$$T = \frac{199.3 + 46.8}{2} + \frac{1^2}{199.3 - 46.8} \ln\left(\frac{0.88}{0.12}\right)$$

$= 123.55 = 124$

The error $E_1(T)$ of assigning object pixels as background pixels is defined as the sum of the histogram $O(GL_i)$ of the object from 0 to the gray level $(T-1)$. If a value, other than 124, is selected initially, then, theoretically, there would be some error. For example, assume a threshold of 100.

$$E_1(T) = P_o \cdot \sum_{i=0}^{T-1} O(GL_i).$$

$$E_1(T) = 0.12 \sum_{i=0}^{124-1} O(GL_i).$$

The value of the summation equals the count of all object pixels of gray level value from 0 to 124. In this example, there are no object pixels, out of 3 total object pixels, in that range.

$$=0.12[0] = 0.0$$

Likewise, the error $E_2(T)$ associated with selecting background pixels as object pixels is equal to the sum of the histogram $B(GL_i)$ of the background pixels from gray level T to GL_{max}. Where GL_{max} is defined as the maximum gray level contained within the image.

$$E_2(T) = P_b \cdot \sum_{i=T}^{GL_{max}} B(GL_i).$$

$$E_2(T) = 0.88 \sum_{124}^{232} B(GL_i).$$

The value of the summation equals the count of all background pixels of gray level value from 124 to 232. In this example, there are no background pixels in that range.

$$= 0.88[0] = 0.0$$

The total error of thresholding this bimodal image is then equal to the sum of the two errors.

$$E_T(T) = E_1(T) + E_2(T),$$

$$E_T(T) = 0.0 + 0.0 = 0.0$$

Note that the error is 0%, which is due to the simplicity of the image and the clear distinction between object and background pixels. In real world situations a larger error should be expected. However, the total error can be always calculated to estimate the

confidence level of the model in recognizing defects. The proper selection of the threshold value can greatly influence the accuracy of the classification and hence error produced. To illustrate that assume a threshold value of 100 was selected for classification. The calculation of produced error will be as follows:

$$E_1(T) = P_o. \sum_{i=0}^{T-1} O(GL_i).$$

$$E_1(T) = 0.12 \sum_{i=0}^{100-1} O(GL_i).$$

The value of the summation equals the count of all object pixels of gray level value from 0 to 100. In this example, there are no object pixels, out of 3 total object pixels, in that range.

=0.12[0]

=0.0

Likewise, the error $E_2(T)$ associated with selecting background pixels as object pixels is equal to the sum of the histogram $B(GL_i)$ of the background pixels from gray level T to GL_{max}. Where GL_{max} is defined as the maximum gray level contained within the image.

$$E_2(T) = P_b. \sum_{i=T}^{GL_{max}} B(GL_i).$$

$$E_2(T) = 0.88 \sum_{100}^{232} B(GL_i).$$

The value of the summation equals the count of all background pixels of gray level value from 100 to 232. In this example, there are three background pixels, out of 21 total background pixels, in that range.

$$= 0.88[1/22_{(115)} + 1/22_{(118)} + 1/21_{(110)}] = 0.88(3/22)$$

$$= 0.14$$

The total error of thresholding this bimodal image is then equal to the sum of the two errors.

$$E_T(T) = E_1(T) + E_2(T),$$

$$E_T(T) = 0.0 + 0.14 = 0.14$$

The error $E_1(T)$ associated with selecting object pixels as background pixels means classifying defective parts as non-defective parts and is called owner's risk. While, the error $E_2(T)$ associated with selecting background pixels as object pixels means classifying non-defective parts as defective parts and is called producer's risk. In this model both types of error, produce's risk and owner's risk, can be calculated to measure the accuracy and reliability of the system.

CHAPTER 6 MODEL APPLICATION FOR COATING ASSESSMENT

6.1 Steel Bridge Coating

Highway steel bridges are painted for long-term conservation of the structure and enhancement of its appearance at the minimum cost. Paint should act as a barrier to hinder the corrosive attack of the steel by moisture, air, and oxidizing chemicals (Chang and Hsie 1995; Steel Structures Painting Manual 1989).

Typically, in highway steel bridges, the vertical supports including the abutments and the intermediate piers are concrete. The main beams and diaphragms are steel structures, which are usually painted. Figure 6.1 shows an image of a typical steel bridge. Figures 6.2 and 6.3 are a plan view and a profile of a typical steel bridge showing the main components. The number of beams or intermediate supports may vary but the principles are the same.
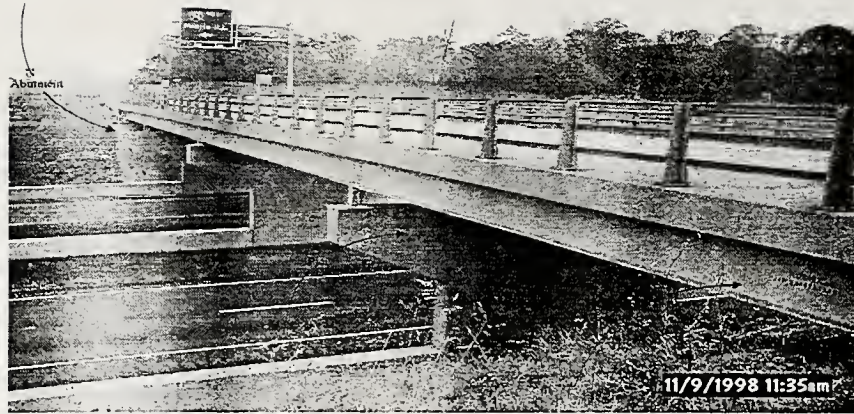
Figure 6.1 Image of A Typical Highway Steel Bridge
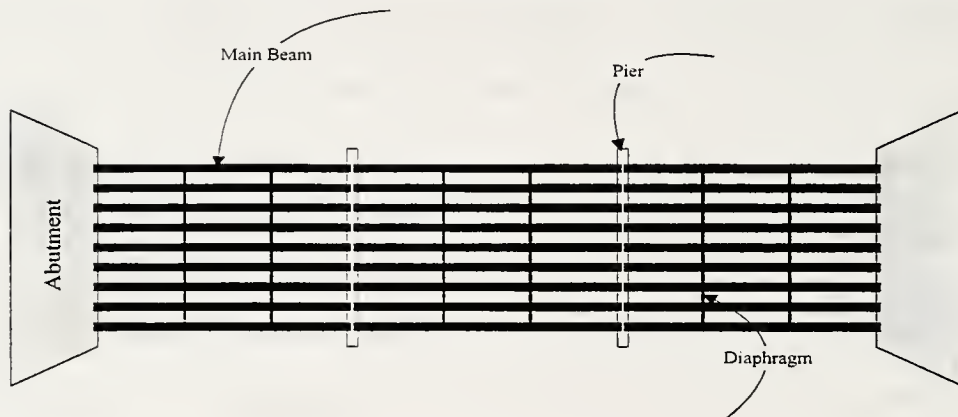
Figure 6.2 Plan of A Typical Highway Steel Bridge
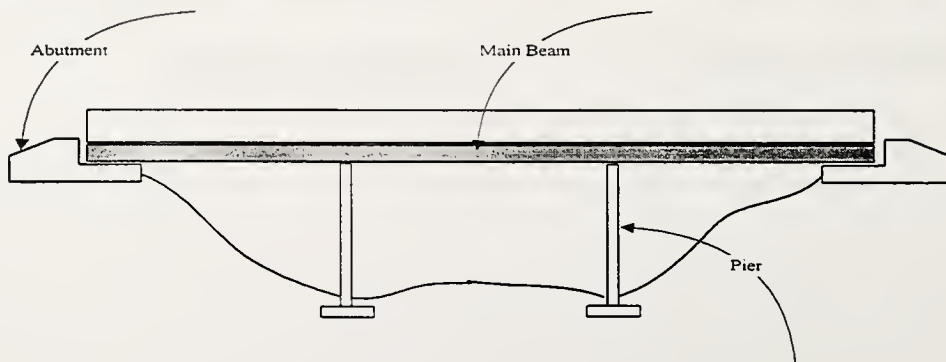
Figure 6.3 Profile of A Typical Highway Steel Bridge

Primer coats for steel bridges commonly embody rust inhibitive pigments or pigments such as zinc dust that reduces corrosion through cathodic protection. On the other hand, The topcoats provide barrier protection through the use of polymers and pigments that protect against moisture permeation and the constant attacks of weather. Thus the surface of a bridge consists of a collection of mini-surfaces with different conditions.

For each steel bridge, highway administrators have a variety of options to consider; ranging from leaving the surface completely untouched to completely blast cleaning and repainting the surface. In between those extremes, are the options of spot cleaning, touchup, and overcoating. The action taken and the type of coating system selected depend on the extent of the occurrence of the surface condition failure and on local preference (Steel Structures Painting Manual 1989).

6.1.1 Coatings Failure

Coatings are the prominent materials of corrosion protection for all steel structures. Coating failure can occur for a number of reasons, which are summarized as follows: (Steel Structures Painting Manual 1989).

1. Failure can occur from the protected surface, the substrate. The coating may be incompatible with the surface or there may be a chemical reaction between the surface and the coating.

2. The coating may fail because of poor or inadequate surface preparation or application. In this case, rapid failure can be caused by surface contamination, poor surface profile, inadequate thickness control, and improper drying.

3. The coating may fail from causes inherent to that specific type of coating. Such failure could be chalking, or checking.

4. Adhesion related failures are abundant and can be catastrophic. Such failures are blistering, flaking and peeling.

5. The coating may fail from exterior factors such as chemical exposure, abrasion or severe weathering.

6. Inappropriate selection of the coating for the specific application. The type of coating should be selected according to the specific use and location factors such as weather conditions.

7. The design of the structure itself is often a cause for severe failure. Sharp edges, skip welds, back-to-back angles are all focal points for failure.

6.1.2 Coating Assessment

Most Departments of Transportation coating inspection procedures are based on the corrosion performance rating system from the American Society for Testing and Materials (ASTM) D610 and D3359 Standards. Table 6.1 shows a numerical corrosion rating according to rust areas percentage and the actions required for all ratings.

Table 6.1 ASTM Corrosion Performance Rating

Corrosion Rating	Assessment Description	Areas to be Repainted (%)
10	No rust or less than 0.01% rust	0
9	Less than 0.03% rust	0
8	Few isolated spots, less than 0.1% rust	0
7	Less than 0.3% rust	0
6	Extensive rust spots, less than 1% rust	8
5	Less than 3% rust	18
4	Less than 10% rust	40
3	Approximately 1/6 of surface rusted	60
2	Approximately 1/3 of surface rusted	100
1	Approximately 1/2 of surface rusted	100
0	Approximately 100% of surface rusted	100

The condition of the bridge coating is frequently described in terms of the percentage of the surface showing some failure such as rust, blistering, or delamination. Alternatively, a rating system of 0 to 10 or excellent, good, fair, or poor may be used (Steel Structures Painting Manual 1989). The rating systems are rather subjective and their accuracy depends greatly on the experience of the inspector.

The general guidelines for ASTM standards specify that for bridges having less than 1 or 2% rust, the common choice is to spot clean and spot prime, followed by one or two full topcoats. For structures showing greater than 20-25% surface failure, it is usually more cost-effective to clean the entire structure and repaint it. In the intermediate range between those two extremes, the decision is more difficult and usually depends on budget, location, local preference, etc. (Steel Structures Painting Manual 1989).

6.1.3 Coating Maintenance

Maintenance of coatings is significant for the visual and physical sustentation of steel bridges. Often, recuperation of these structures is more difficult than coating a new structure. The problem is primarily caused by the surface preparation of the existing coating. For example, in spot painting, only isolated rust spots have to be removed. The contractor must locate these areas and remove the defects without damaging the adjacent coating that is still in good condition.

Furthermore, there should be a transition zone between the existing coating and the edges of the blast-cleaned areas to increase the mechanical adhesion between the old and new coatings. This zone, which receives a lower degree of blast cleaning, will improve the appearance of the rehabilitated areas. Choosing the proper recuperation method at the most economical cost is critical in coating maintenance. Overcoating a bridge before there is significant rusting on the surface can be both uneconomical and detrimental to the existing coating. The thick and heavy buildup of the coating will eventually lead to delamination and cracking.

6.1.4 Corrosion

The corrosion of steel is the deterioration and eventual destruction of the metal because of its reaction with the environmental factors. Steel is manufactured from iron, which is reduced from ore under very high temperatures. In the process, the iron has absorbed energy. This is typical in all metals and this absorbed energy is what makes metals chemically unstable. They continually try to revert back to their original form when combined with moisture and oxygen.

This "oxidation degeneration" of metals is the corrosion process (Steel Structures Painting Manual 1989).

Corrosion is an electrochemical reaction. An anode and cathode are joined by a conductor, and then immersed in an electrolyte. The anode has a tendency to lose ions and the cathode has a tendency to gain ions. As electrons leave the anode, positively charged ions are released into the electrolyte. These ions react with other materials to form corrosion products (rust). In general, the faster the flow of electrons the faster the rate of corrosion. The current flow is predicted using Ohm's law.

$$I = \frac{E}{R}$$
(6.1)

Where I = current flow, E = potential difference, and R = overall electric resistance of cell. The rate of corrosion is affected by several factors. Environmental factors include temperature, humidity and type of exposure. High temperature and humidity increase the rate of corrosion. Areas of structures exposed to wind or sun can dry quickly and are less prone to corrosion. Salt makes water a more efficient electrolyte so structures in a coastal environment or exposed to deicing salts will have increased rates of corrosion. Similarly, structures in industrial areas can have higher corrosion rates due to atmospheric pollutants. Even bird droppings can have a significant effect, leaving moisture and deposits that can form corrosion cells or chemically attack the metal.

Type of steel can also have an effect on the rate of corrosion. Different grades of steel used different alloying agents to provide various steel characteristics. ASTM-A36 is the most common grade of steel used for bridges in the U.S. built since the 1930's. Its main alloying

agents are carbon and manganese and require coating for corrosion protection. Some alloys that increase the corrosion resistance of steel include copper, nickel, silicon and chromium. ASTM-A588 "weathering" steel contains these alloys and has about 4 times the resistance of carbon steel.

The type of structure also has a major impact on the rate of corrosion. Locations where water is likely to collect will have a higher rate of corrosion than the rest of the structure. In the case of bridges, prone areas are inclined members and joints. Due to these variables, the rate of corrosion is one of the most difficult characteristics to predict.

6.2 Statistical Assessment Plan

A statistical assessment plan is adopted based on previously developed sampling plans to supplement the hybrid model (Chang and Hsie 1994). The statistical plan determines the sampling method, number and locations of images, and the risks associated with acceptance and rejection of sample images.

6.2.1 Sampling Acceptance Plans

An acceptance sampling plan applies statistics for quality inspection. The primary purpose of an acceptance sampling plan is to support the decision to accept or reject inspected products. There are several ways to categorize acceptance sampling plans, and there are several types of them. The most commonly used sampling plans are the single sampling plan and the double sampling plan (Montgomery 1991).

In a single sampling plan, samples are taken only once, and the decision to accept or reject is based on this one sampling. On the other hand, a double sampling plan uses a smaller sample size, and if the information from the first sampling clearly denotes acceptance or rejection, the decision will be based on one sampling. Otherwise, if the information gathered from the first sampling falls between acceptance and rejection, a second sampling will be performed.

6.2.2 Acceptance/Rejection Risk

Contractor's work that is accepted by highway agencies is called Acceptable Quality Level (AQL). Nevertheless, not all acceptable samples from that work are 100% guaranteed acceptance, because of errors and sampling variation (Hsie 1994). Furthermore, work is rejected by highway agencies is called Rejectable Quality Level (RQL). Likewise, not all samples from rejectable work are 100% guaranteed rejection.

The probability of rejecting AQL product is called producer's risk (α) because the producer has the risk of being rejected even though work quality is within AQL. On the other hand, the probability of accepting RQL product is called owner's risk (β). Figure 6.4 shows a typical operating characteristic curve with the associated two types of risks.
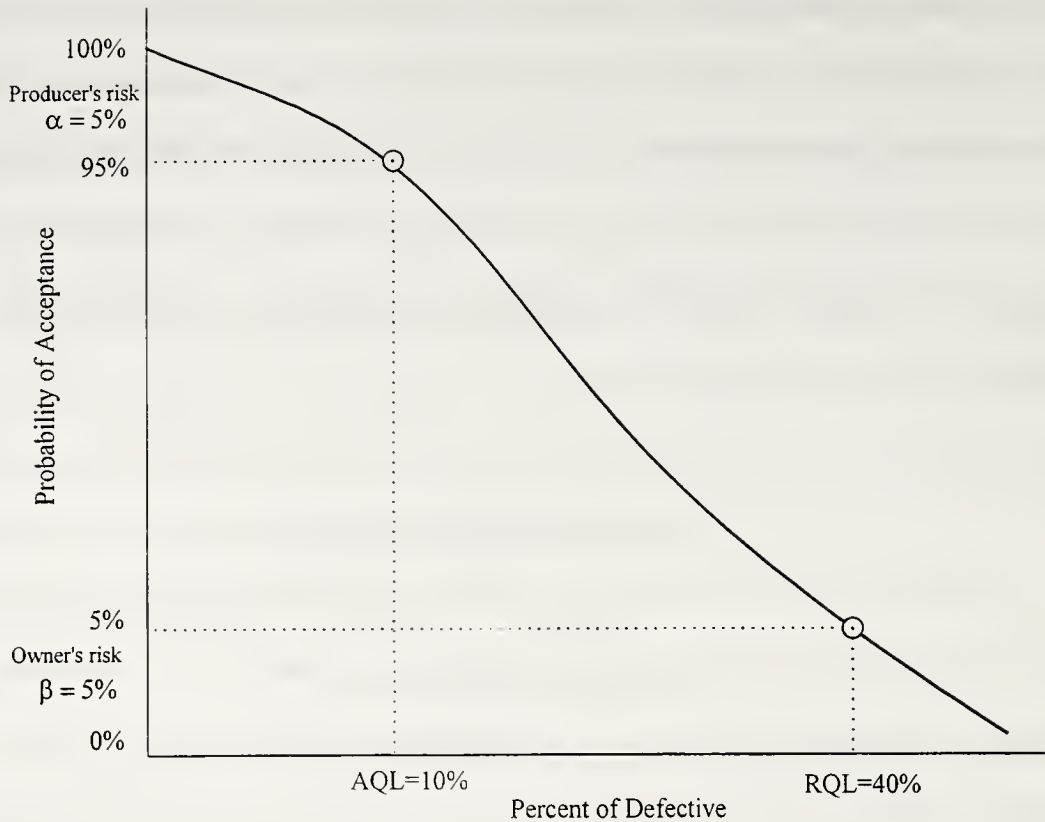
Figure 6.4 Typical Operating Characteristic curve

6.2.3 Hybrid Model Acceptance Sampling Plan

Choosing the best acceptance sampling plan depends on the nature of the assessment process (Hsie 1994; Webster 1992). There are many parameters that affect that choice including; the required accuracy, the inspection cost, logistics, time constraints, and so forth. A double sampling plan was chosen to supplement the hybrid model.

One reason for selecting the double sampling plan is its simplicity. It does not need executing complex statistical calculations and thus it enables instant decision making on site. Another reason is its flexibility because it allows for including multi-parameter assessment.

Many criteria can be applied to each sample and it's accepted only if it satisfies all criteria. Another advantage is the smaller sample size required for double sampling compared to other sampling plans (Chang and Hsie 1995; Hsie 1994).

The double sampling plan uses a small sample size, and if the information from the first sampling clearly denotes acceptance or rejection the decision will be based on one sampling. Else, if the information gathered from the first sampling falls between acceptance and rejection, a second sampling will be performed. The following notations and parameters are used for the double sampling plan.

N: Lot size

n_1:   Sample size on the first sample

n_2:   Sample size on the second sample

c_1:   Acceptance number on the first sample

c_2:   Acceptance number on the second sample

d_1:   Number of defective units in the first sample

d_2:   Number of defective units in the second sample

p: Fraction of defective items in the lot

r_1:   Rejection number on the first sample

For implementing the double sampling plan, first a sample of size n_1 is taken and if the number of defective units d_1 is equal to or less than the acceptance number c_1, the lot is

accepted. If the number of defective units is equal to or greater than the rejection number r_1, the lot is rejected.

If the number of defective units falls in between c_1 and r_1, a second sample of size n_2 is taken. The number of defective units in the second sample is d_2. If the total number of defective units in both samples (d_1+d_2) is equal to or less than the acceptance number on the second sample c_2, the lot is accepted; otherwise, the lot is rejected (Montgomery 1991; Hsie 1994). Figure 6.5 illustrates the adopted attribute double sampling plan (Chang and Hsie 1995; Hsie 1994).

The process of selecting all the parameters for the sampling plan is rather complex because of the many variables. The selection process is usually done by trial and error and depends on the specific application and experience. For the application of the hybrid system the producer and owner's risks were set at 5% levels. Hsie (1994) developed a method to estimate the sampling plan parameters using neural networks learning. Two example sampling plan parameters are adopted for this system. Table 6.2 shows the plan parameters.

Table 6.2 Acceptance Plan Parameters (Hsie 1994)

Plan	AQL	RQL	n_1	n_2	c_1	c_2	r_1
1	8.12%	42.87%	10	10	1	4	3
2	8.37%	42.87%	8	8	0	3	3
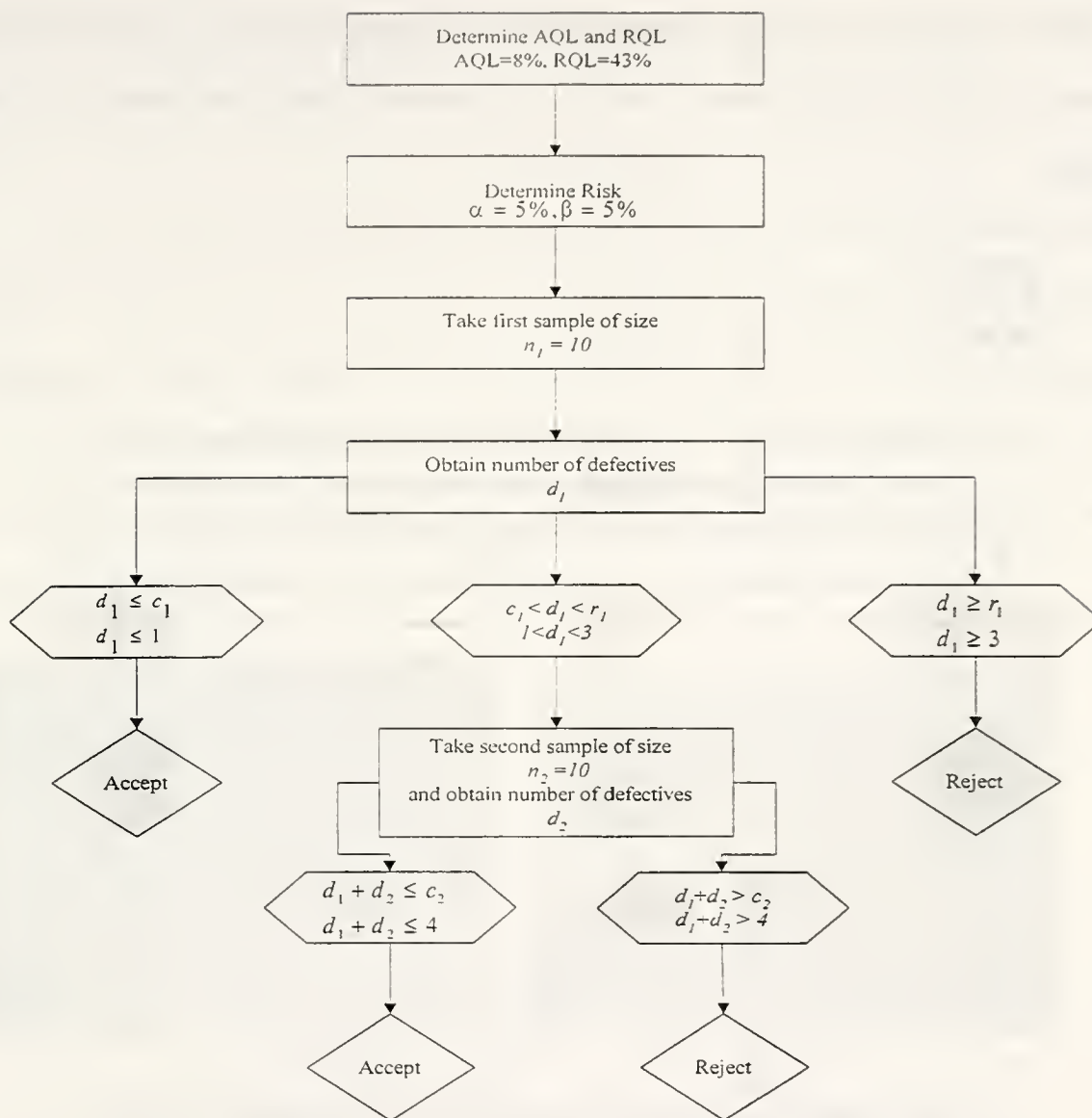
Figure 6.5 The Double Sampling Plan

A standard coding system for the steel bridge coating assessment is developed for efficient sampling and record keeping. Steel beams are numbered according to their geographical direction starting at the top left corner. For example, Figures 6.6 and 6.7 illustrate typical steel bridges with the coding system and the north point direction shown for two possible orientations. The coding starts at the top left corner and that beam is coded (NS1)

because it's running North-South starting from left and it's beam number 1. The second beam is coded (NS2) and so forth. Beams running in the opposite direction are coded in the same manner.
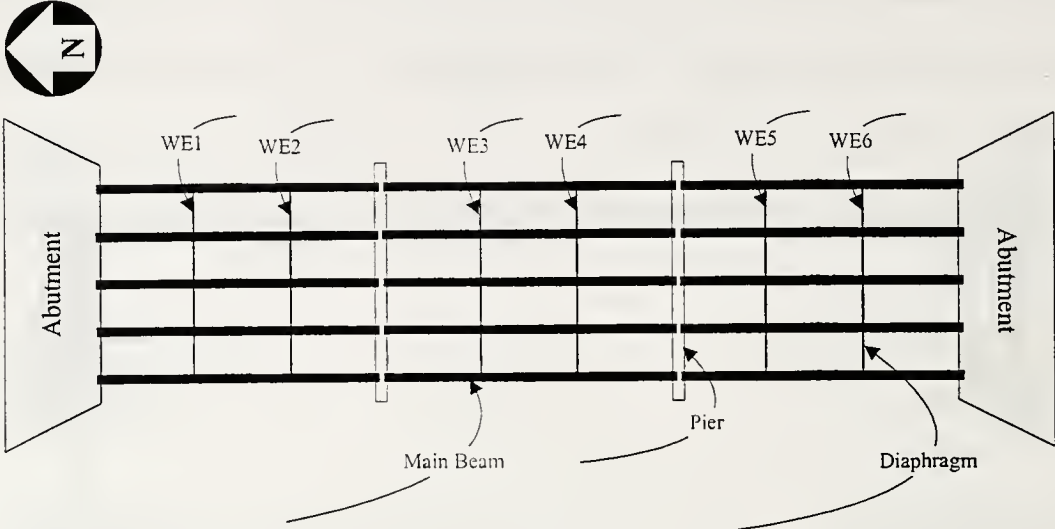
Figure 6.6 Statistical Plan Bridge Labeling Sequence (Bridge Plan View I)
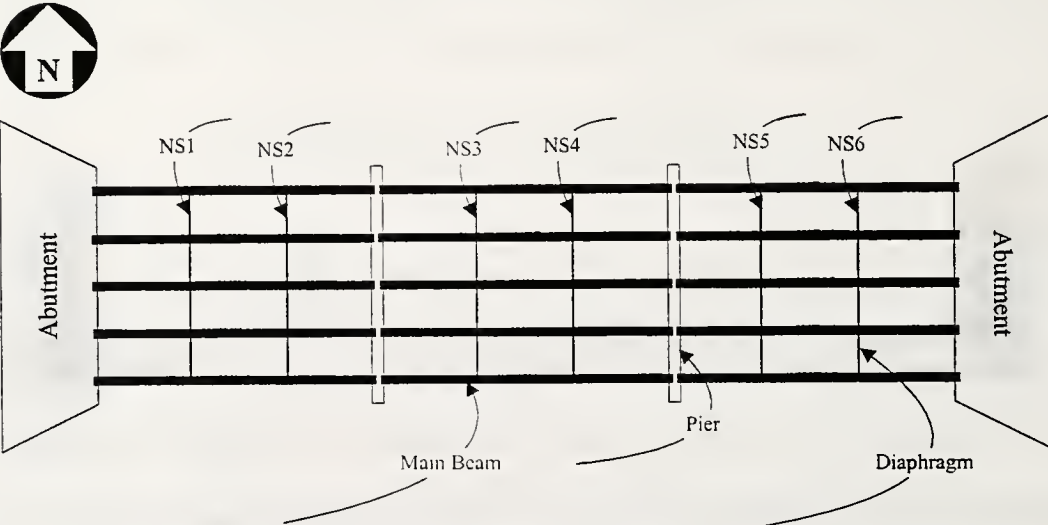
Figure 6.7 Statistical Plan Bridge Labeling Sequence (Bridge Plan View II)

6.3 Model Implementation

6.3.1 Data Acquisition

Over 150 digital images of steel bridges coating were acquired using a Kodak DC-260 digital camera using two types of resolutions; 1536 x 1024 pixels and 1152 x 718 pixels. Many highway steel bridge coating images were taken at different locations in the state of Indiana. The model testing images were taken from the two steel bridges on Highway US-41 at the junction of Interstate I-74. The images were transferred to the computer using a Universal Serial Bus (USB) port and cable. Figure 6.8 shows sample images from the two bridges.
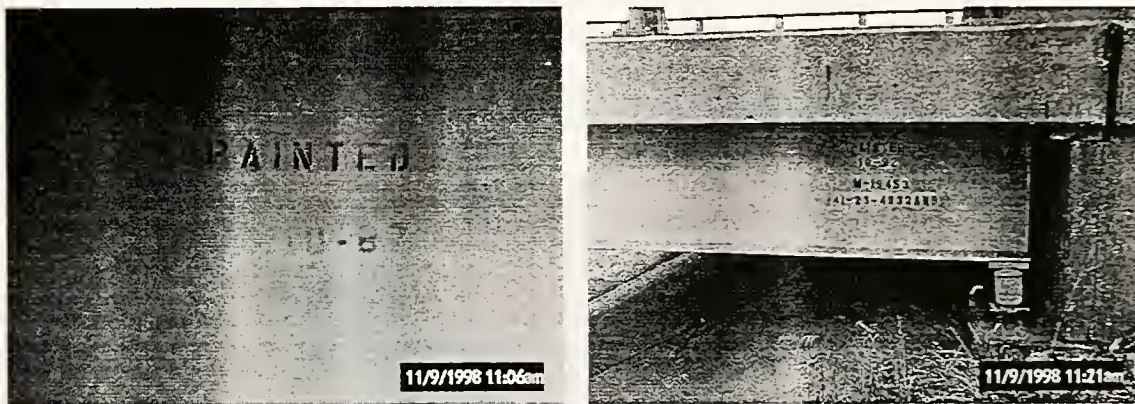
Figure 6.8 Sample Coating Images

The number and location of images taken follows the statistical sampling plan described in section 6.2. In each beam the location of the image is randomly selected for the overall statistical sampling plan. Furthermore, additional images were taken from areas with visible defect patterns for spot assessment.

6.3.2 Pre-Processing

The images' quality is enhanced using different image processing techniques and filters such as noise reduction, contrast and brightness adjustments. Moreover, images were converted to gray-scale from color for better threshold execution. Images' are converted to numerical matrices for neural training and analysis. Several experiments with different software packages were made for pre-processing. Packages such as Matlab® Image Processing Toolbox and Matrox® Inspector were used for preprocessing operations.

Prior to the neural network training, image processing algorithms are applied for images' classification. The algorithms use pattern recognition for image classification into two classes or thresholding. The procedure is described in section 5.3. Figure 6.9 shows an original coating image and Figure 6.10 shows the image after conversion to a graylevel image.

11/9/1998 11:22am

Figure 6.9 Original Example Coating Image
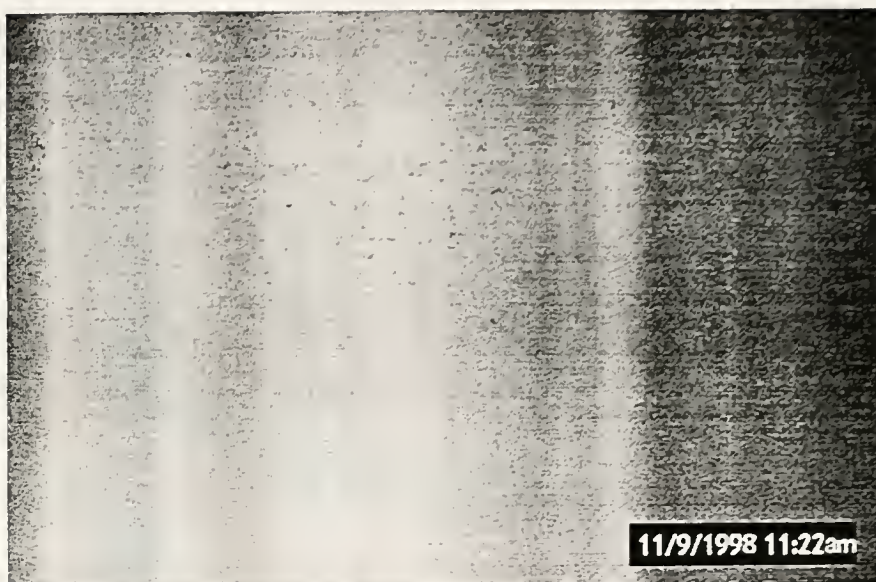
Figure 6.10 Graylevel Example Coating Image

Figure 6.11 shows the original image's histogram and Figure 6.12 shows the histogram for the image after it is converted to a gray scale image.
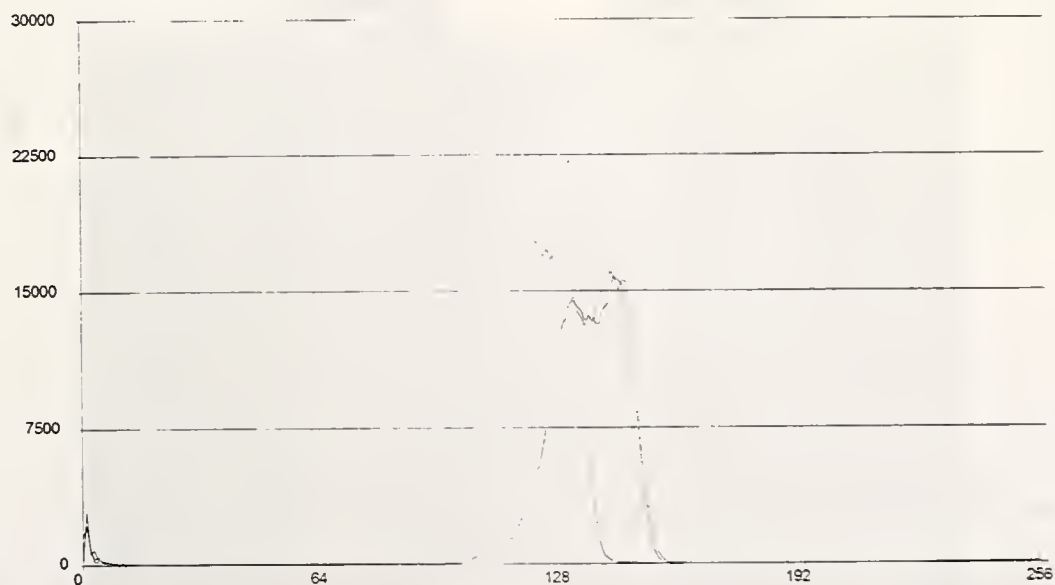
Figure 6.11 Original Image's Histogram
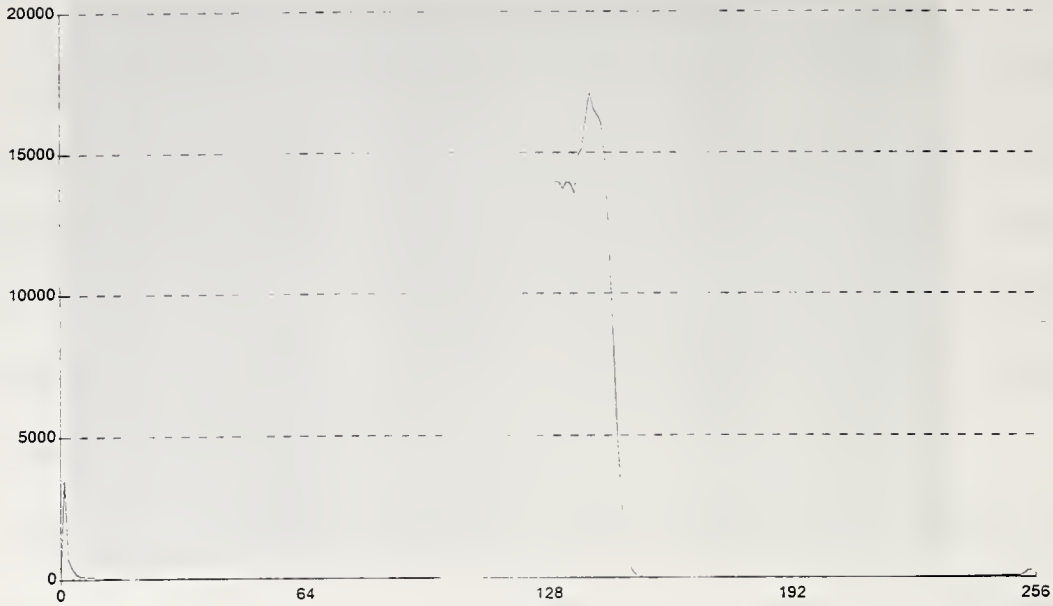
Figure 6.12 Gray Scale Image's Histogram

Figure 6.13 illustrates a thresholded image with object (rust) pixels recognized and

labeled.

Figure 6.13 Thresholded Image

6.3.3 Analysis

In the next stage, images are passed to the neural network for training. A three- layer neural network with Back-propagation training algorithm is used as described in sections 4.2.3 and 5.3. The input layer consists of the gray levels of the coating image and the output layer consists of a mapped gray level value of 0 or 1. The total number of images used for training was 50. Figure 6.14 shows the neural network architecture.
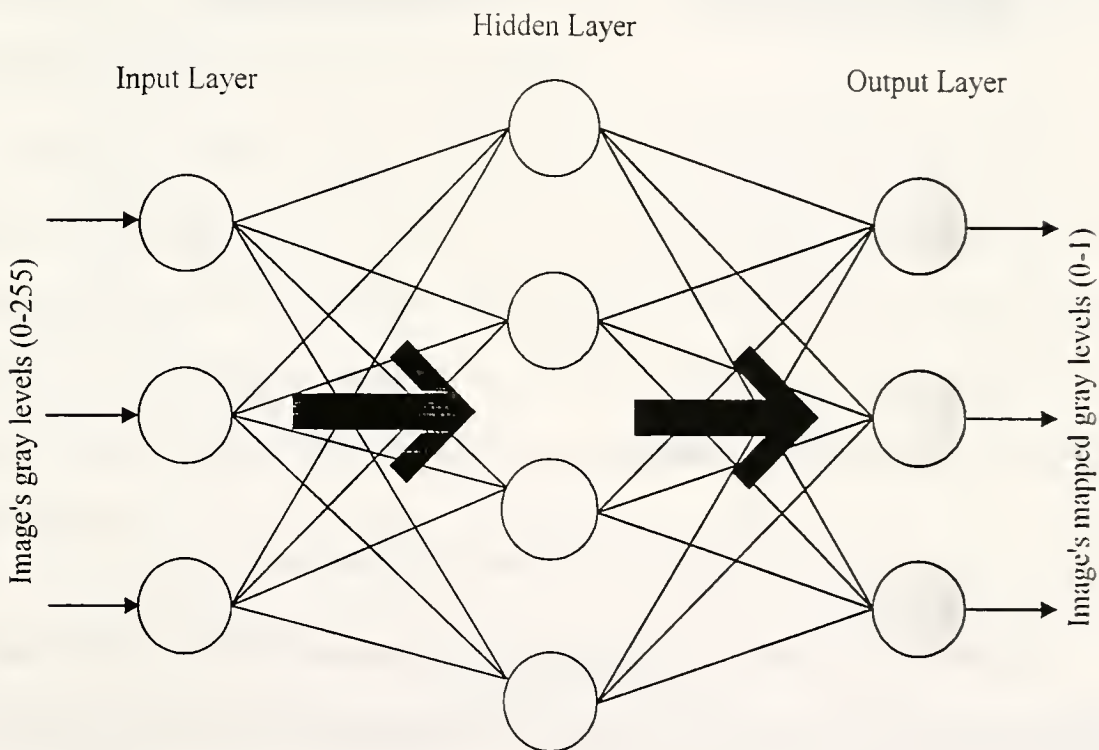
Figure 6.14 Neural Network Architecture

A data set of 50 images was used to train the neural network. Each image pixel values matrix and mapped pixel values matrix was considered a training pair for the network. Figure 6.15 illustrates the training set for the neural network.
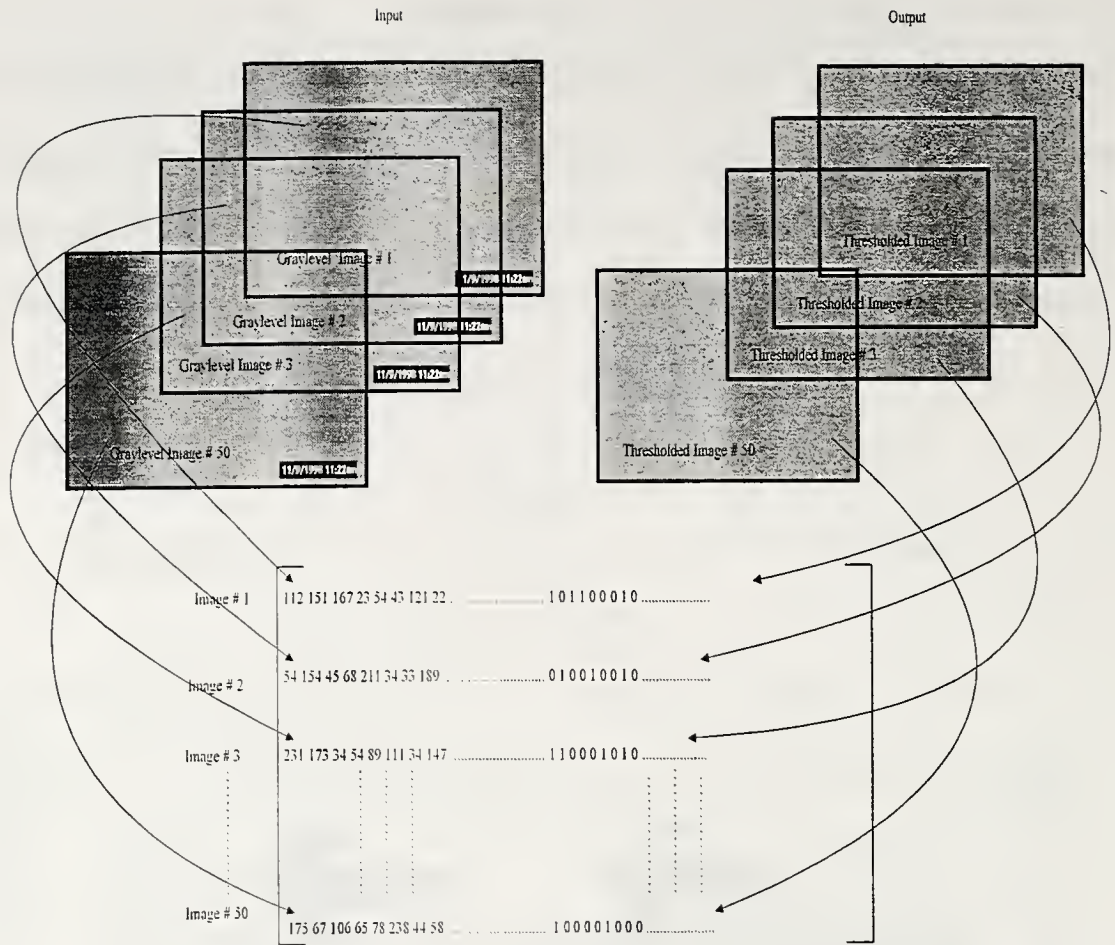
Figure 6.15 Neural Network Training Set

The aspiration of using the neural network is to generate reliable consistent assessment results even for noisy or lower quality images. The learning mode used is supervised learning so as to train the network from example images to simulate human experience in recognizing and classifying images with different parameters and different scenarios.

Prior to training, some human interference might be required, specifically in threshold selection for defect identification because of different parameters such as image quality,

external factors in images such as dirt, etc. Adjustments are made using experts' knowledge and then the neural network is trained in order to automate the process in the future.

The neural network is relatively fault tolerant because it makes up for distorted or missing data provided that good training sets are provided. The reason behind that is the ability of neural networks to store knowledge in a distributed memory fashion among the network weights. Weights adjust repeatedly during training to reach minimum desired error. Because of the vast number of weights and the distribution of knowledge among them, incomplete or noisy data should not affect the accuracy of the network.

Neural network limitation stems in the fact that good sets of data should be provided for training. Training the network with distorted or misrepresenting data will generate inaccurate results and hence weaken the network's performance. Another limitation is the large amount of computing required to train images with high resolution. There is a trade-off between image resolution and computation efficiency. Moreover, The training set should include a full range of possible scenarios.

6.3.4 Recognition

Subsequent to the neural network training, more images were processed to obtain the pixels' mapping to 0 or 1 values and hence identification and measurement of defects (rust) as described in the previous sections. Figure 6.16 illustrates the recognition procedure.
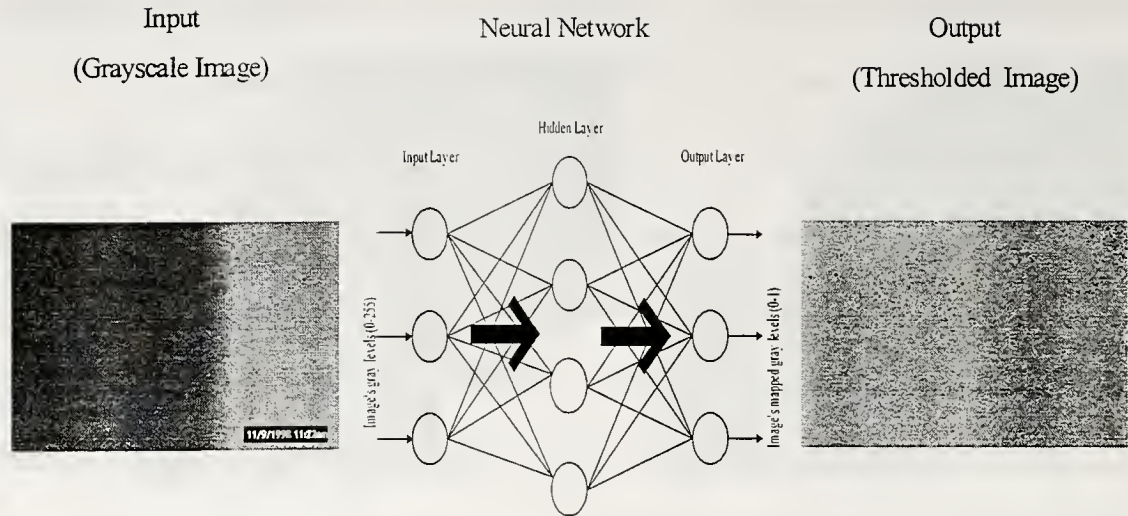
Figure 6.16 Recognition Process

Table 6.3 illustrates parts of the mapped output in the form of binary variable 0 or 1. The table shows all the gray levels range from 0 to 255 together with the frequency (count) of occurrence for each gray level. Moreover, the table shows the mapped gray levels to 0 or 1 together with counts for each. The rust percentage area is calculated by counting the total number of rust pixels with value 1, from the neural network output, and dividing by the total image's number of pixels.

Table 6.3 Mapped Model Output

Pixel gray level	Count	Mapped graylevel	Count
0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
...			
...			
...			
64	1013	0	1013
65	786	0	786
66	0	0	0
67	0	0	0
68	0	0	0
69	352	0	352
70	702	0	702
71	0	0	0
72	256	0	256
73	0	0	0
74	1546	0	1546
75	0	0	0
76	0	0	0
77	1305	0	1305
...			
...			
250	0	1	0
251	0	1	0
252	0	1	0
253	0	1	0
254	0	1	0
255	0	1	0

Finally, a highway steel bridge on US-41 near I-74 in Indiana was used as an example for the statistical assessment plan. The described procedure was done from many parts (beams) of the example steel bridge coating. The defect percentage was calculated for each image. The results from all images of a bridge are compiled together for the assessment decision. Figure 6.17 shows the plan for the bridge with the associated beam labels.
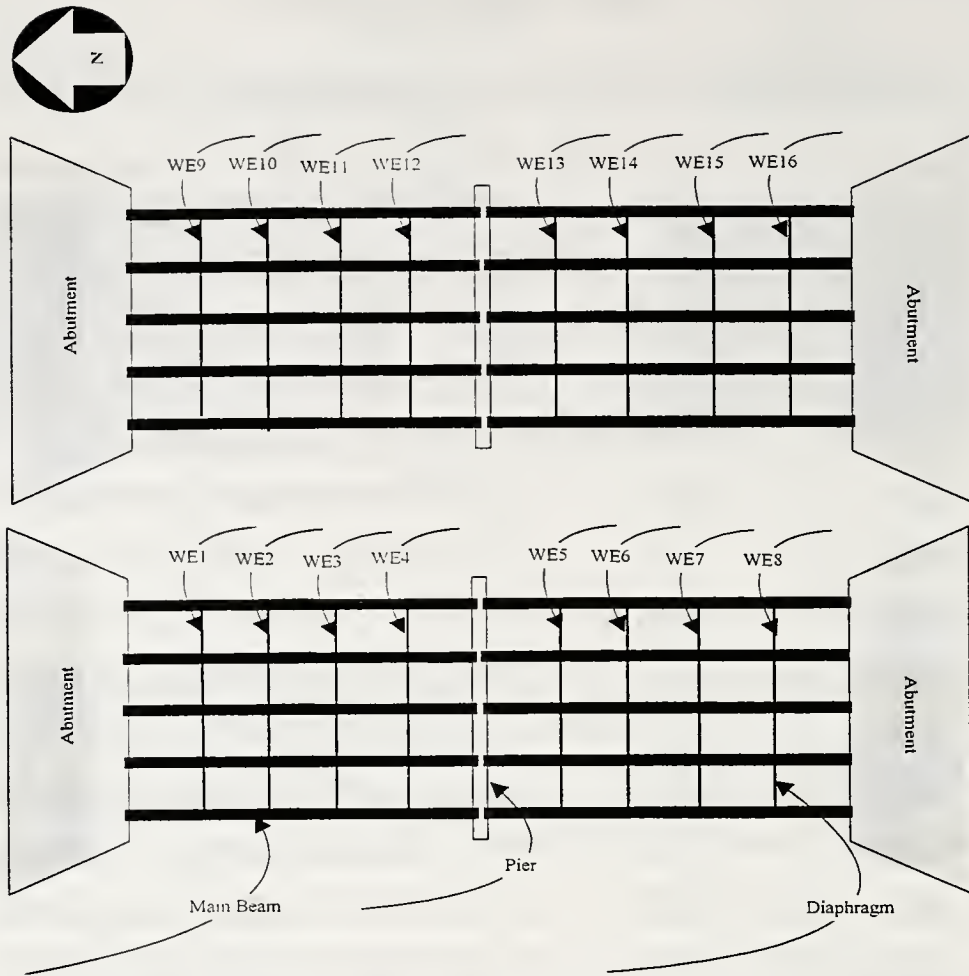
Figure 6.17 Plan of Example Bridge on Highway US-41

Table 6.4 illustrates the assessment summary for the example bridge on highway US-41. One image was acquired for each steel beam in the bridge and the exact image's location was selected randomly. Every image was labeled whether it represents a main beam or a diaphragm with the corresponding beam code, according to the coding scheme described in section 6.2.3. The rust persentage area in each image was calculated and every sample was considered defective if the image rust percentage area exceeded 1% and non-defective otherwise.

Table 6.4 Bridge Assessment Summary (Bridge on Highway US-41)

Number	Beam Code	Main Beam	Rust Percentage	Assessment
1	NS1	X	0	Non-defective
2	NS2	X	0.23	Non-defective
3	NS3	X	0.9	Non-defective
4	NS4	X	0	Non-defective
5	NS5	X	0	Non-defective
6	NS6	X	0	Non-defective
7	NS7	X	4.7	Defective
8	NS8	X	0.81	Non-defective
9	NS9	X	0	Non-defective
10	NS10	X	11.4	Defective
11	NS11	X	0	Non-defective
12	NS12	X	0	Non-defective
13	NS13	X	0	Non-defective
14	NS14	X	0	Non-defective
15	NS15	X	0	Non-defective
16	NS16	X	0.62	Non-defective
17	NS17	X	0	Non-defective
18	NS18	X	0	Non-defective
19	NS19	X	0	Non-defective
20	NS20	X	0	Non-defective
21	WE1		0	Non-defective
22	WE2		5.11	Defective
23	WE3		0	Non-defective
24	WE4		0	Non-defective
25	WE5		6.34	Defective
26	WE6		0	Non-defective
27	WE7		0.56	Non-defective
28	WE8		0	Non-defective
29	WE9		0	Non-defective
30	WE10		0.85	Non-defective
31	WE11		0	Non-defective
32	WE12		0	Non-defective
33	WE13		0	Non-defective
34	WE14		7.5	Defective
35	WE15		0	Non-defective
36	WE16		0	Non-defective

The statistical acceptance plan is applied for the decision making of whether to accept or reject the steel bridge coating job. For this example, the lot size or total population N is 36, producer's risk α =5%, owner's risk β = 5%, and sample size n_1=10. By selecting, randomly, 10 observations from the total population of 36 there were 2 defective units d_1. Another sample of size n_2 =10 was taken and the number of defective neurons d_2 was found to be 2. The total number of defective units in both samples (d_1+d_2) = 4. Table 6.5 illustrates the final assessment summary.

Table 6.5 Final Steel Bridge Coating Assessment Summary (Bridge on Highway US-41)

Bridge Number	Facility Carried	Coating Date	Last Inspection Date	Sample Size	Number of Detectives		Accept	
					d_1	d_1+d_2	Yes	No
041-23-02359	US41	1987	8/4/97	10	2	4	x	

In conclusion, this hybrid model makes the coating assessment procedure more objective and consistent. Using computer image processing and neural networks enables the system to recognize and measure defects on a quantitaive and consistent bases which is an exteremely diffuicult task to achieve depending only on visual assessment.

CHAPTER 7 IMPLEMENTATION

7.1 Implementation Suggestions

In view of the accuracy and efficiency on bridge surface coating inspection provided by the developed hybrid model, it is suggested that the Indiana Department of Transportation (INDOT) can gradually implement this optical imaging method on bridge coating inspection. The optical imaging method can be implemented on bridges under warranty clauses or be used for regular maintenance plans. The developed hybrid model will be implemented in late 1999 for assessing two bridges under warranty clauses. The detailed procedures of implementation are described sequentially in the following sections.

7.2 Image Acquisition

For image acquisition a Kodak DC-260 digital camera was used. The camera is shown in Figure 7.1 and the camera specifications are shown in Appendix C.
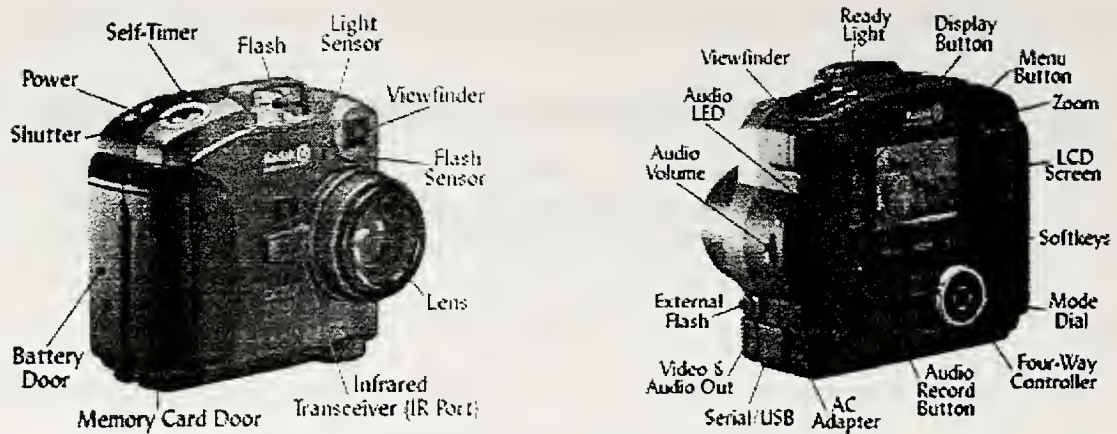
Figure 7.1 The Digital Camera

In the integrated sampling plan, the exact location for each sample image should be determined beforehand to ensure randomness. A standard coding system for the steel bridge coating assessment is developed for efficient sampling and record keeping. Steel beams are numbered according to their orientation starting at the top left corner. For example, Figures 7.2 and 7.3 illustrate typical steel bridges with the coding system and the north point direction shown for two possible orientations. The coding starts at the top left corner and the first beam is coded (NS1) because it's running North-South starting from left and it's beam number 1. The second beam is coded (NS2) and so forth. Beams running in the opposite direction are coded in the same manner.
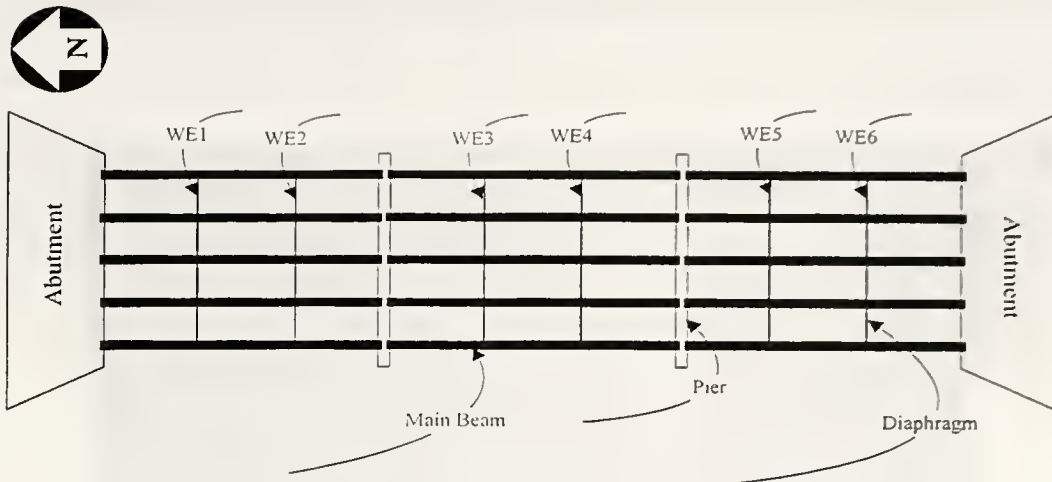
Figure 7.2 Statistical Plan Bridge Labeling Sequence (Bridge Plan View I)
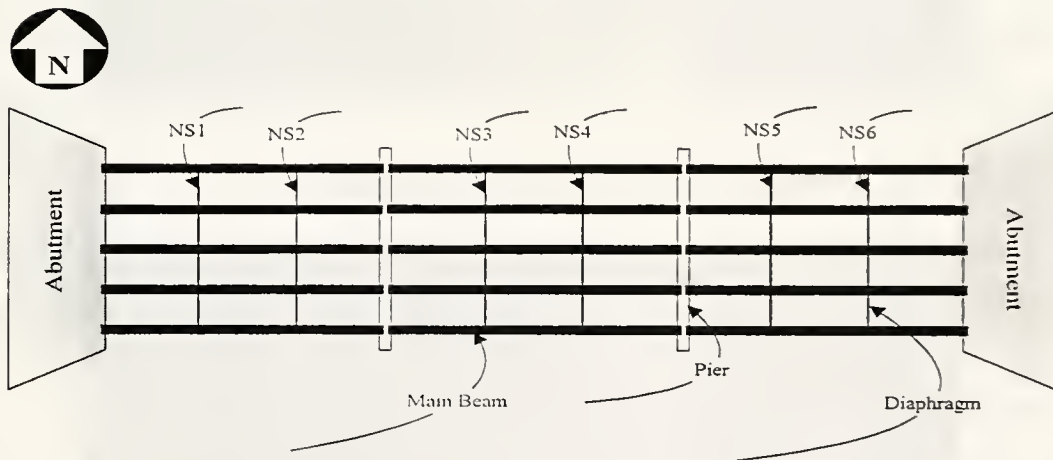
Figure 7.3 Statistical Plan Bridge Labeling Sequence (Bridge Plan View II)

Moreover, the distance between the camera and objects should be standardized. The distance between the camera and the beam or area to be assessed should be close to 3-4 feet. The image acquired should be of a continuous area of coating as much as possible, as shown in Figure 7.4. Nevertheless, during image analysis particular areas of interest in the image

can be selected for analysis as in Figure 7.5. In order to do that, all accessibility problems

need to be resolved.
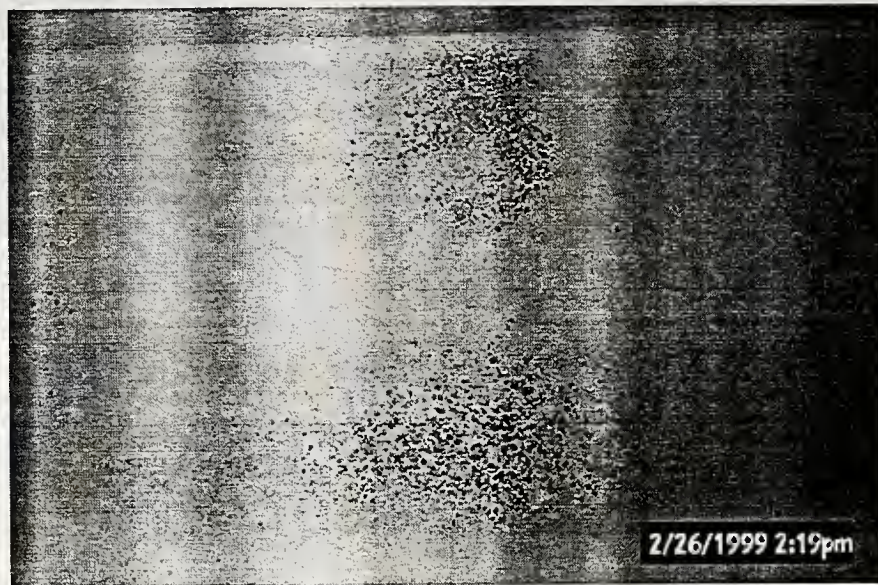
Figure 7.4 Image of a Continuous Area

Figure 7.5 Selected Region of Interest in an Image

The digital camera has a built-in flash, which is activated when there is low light in the area. For the purpose of this research, effort was made to acquire images during reasonable lighting condition using daylight and the built-in camera flash. No images were acquired at night and a mix of sunny and cloudy days was used for image's acquisition and training.

7.3 Images Transfer and Storage

The digital camera used has three resolutions: 1536 x 1024 pixels, 1152 x 768 pixels, 768 x 512 pixels. Moreover, the camera has three image compression settings: Best, Better, and Good. A resolution of 1152 x 768 pixels together with Best compression setting was used for images' acquisition. The digital camera has a flash memory card of 32 MB capacity which can store around 207 images using the above-mentioned setting. The images can be transferred to the computer on site or back at the office via a Universal Serial Bus USB Port, a serial port, or with a wireless communication via an infra red port.

A working computerized framework for images database was developed. The database includes comprehensive information for all steel bridges coating images such as dates and times each image was taken, weather condition, lighting condition, bridge number, facility carried, coating contract number, image location code, beam type, measured rust area, assessment decision, and action to be taken. A screen shot of the database is shown in Figure 7.6. The database also includes a hyperlink to the corresponding image. The database functions as a record-keeping tool for future reference. Patterns of deterioration for a bridge or parts of a bridge can be easily studied by referring to the database.

Figure 7.6 Images Database

7.4 Image Analysis

Matrox Inspector software is used for analysis. The first step is to convert the image like the one shown in Figure 7.4 to a gray-scale image. Figure 7.7 shows a screen shot of the original image and the converted gray-scale image.
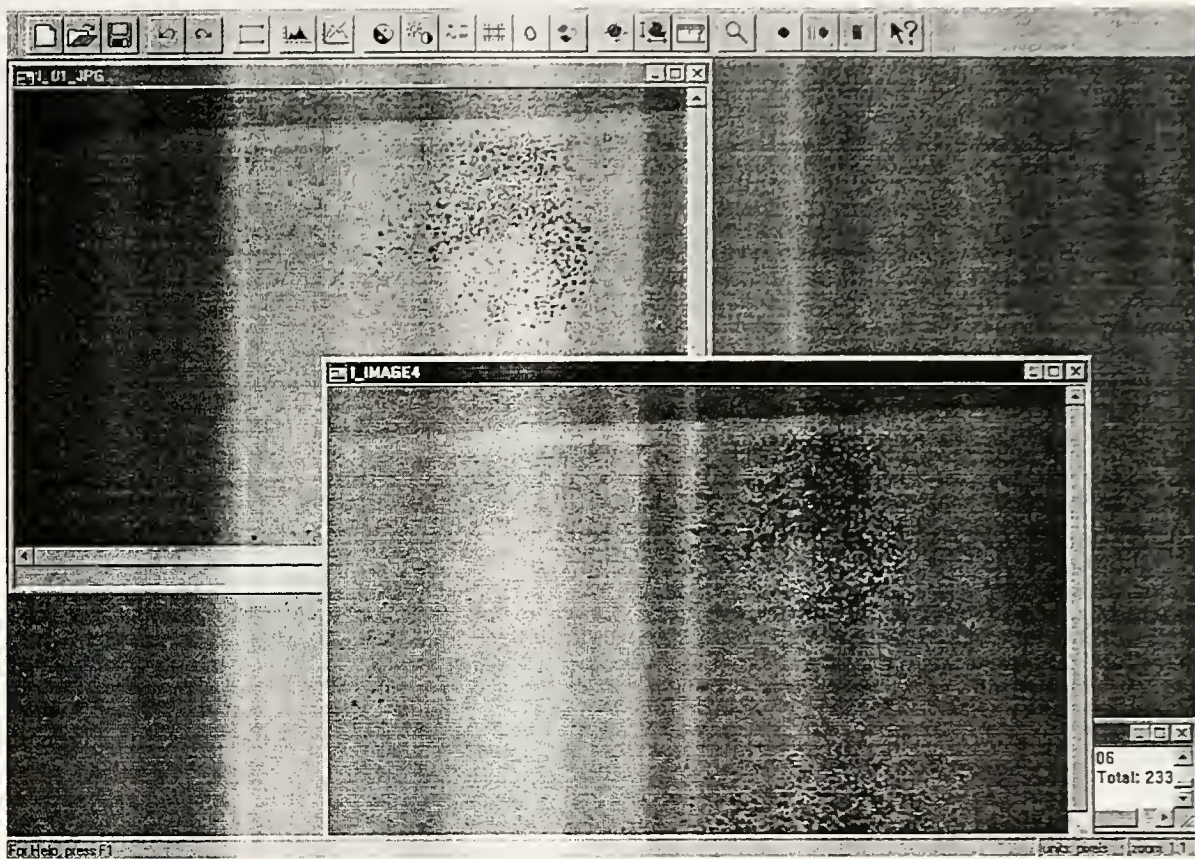
Figure 7.7 Image Conversion

The next step is to apply the threshold as described in Chapter 6. Figure 7.8 illustrates a screen shot of the threshold operation.
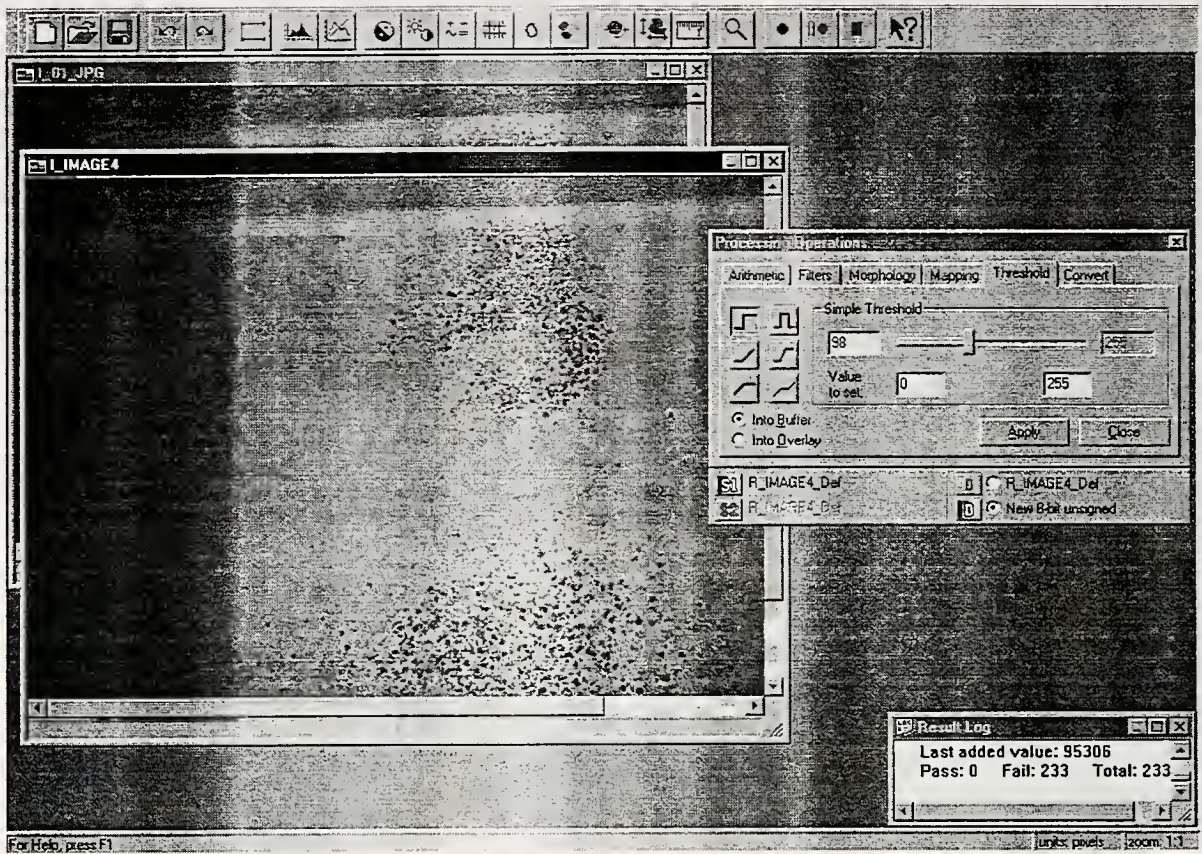
Figure 7.8 Screen Shot of Threshold Operation

The final step is to use the blob analysis operation to calculate the rust areas in the image. Figure 7.9 shows a screen shot of the operation.
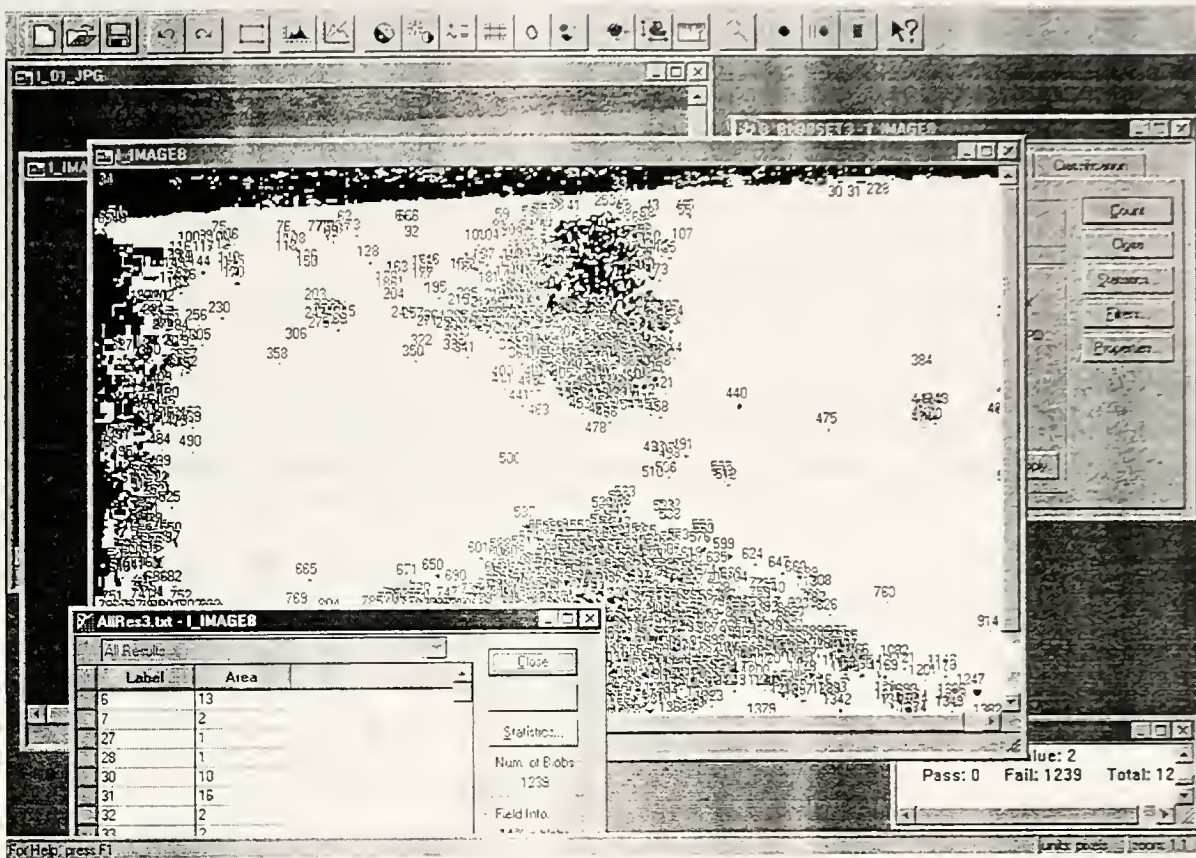
Figure 7.9 Screen Shot of Rust Area Calculation

For the model performance measurement, there was no reliable benchmark comparison data. For example, the system measures a rust area in a steel bridge coating as a percentage and the available source to compare that measurement to, is inspectors' estimate of that percentage which is a subjective estimate. ASTM has a template of 9 example rust percentage areas for visual comparison as shown in Figure 7.10.
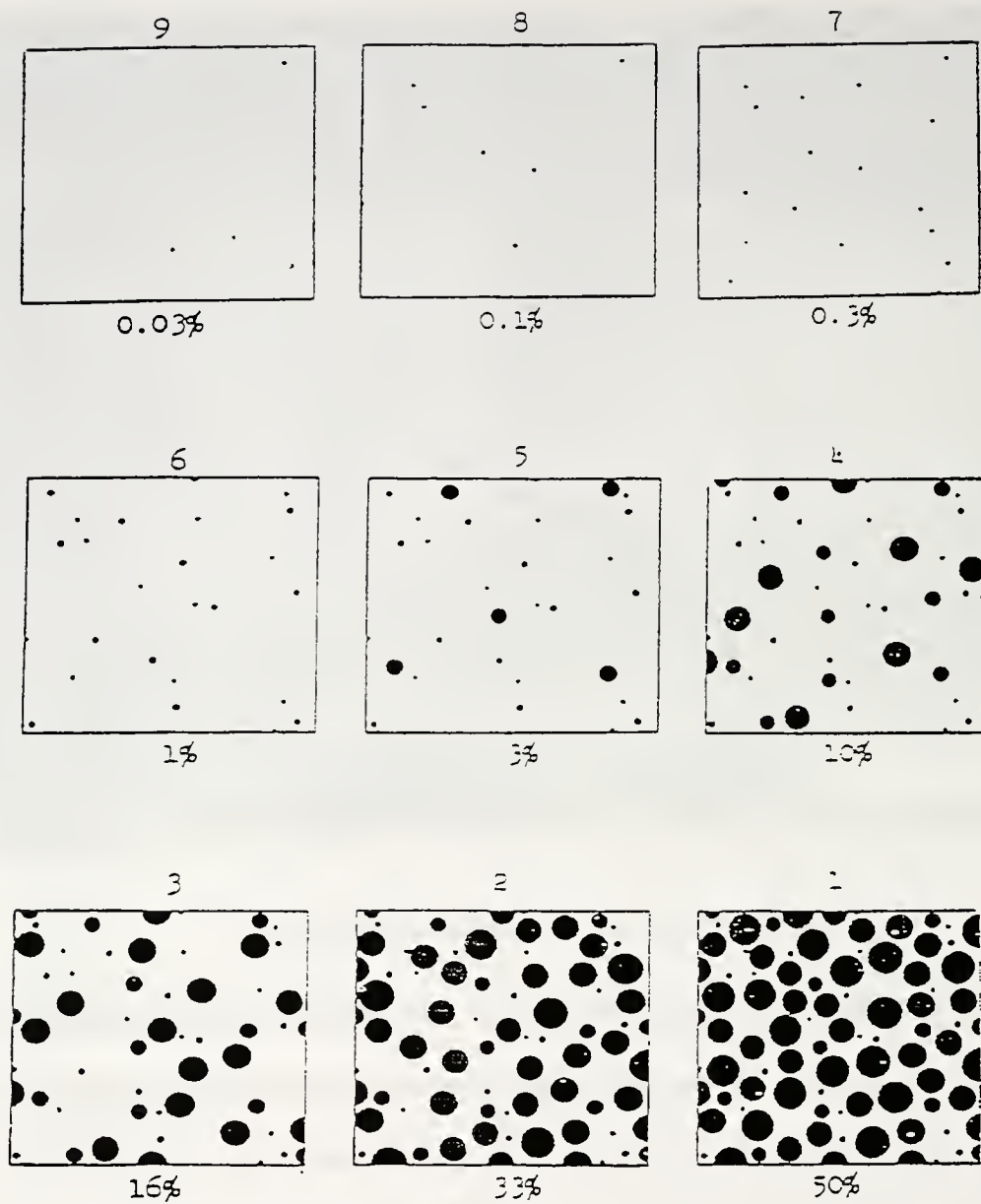
FIG. 1 Examples of Area Percentages

Figure 7.10 ASTM Rust Area Template

The model can be used in conjunction with the ASTM template by actually measuring the rust areas in the image and converting that to a performance rating in accordance with the template as shown in Table 7.1.

Table 7.1 Rust Performance Rating

Rating	10	9	8	7	6	5	4	3	2	1	0
Rust Percentage	0-0.01%	.02-0.03	0.04-0.1%	0.11-0.3%	0.31-1%	1.1-3%	3.1-10%	10.1-16%	16.1-33%	33.1-50%	51-100%

Table 7.2 illustrates a comparison of the actually measured rust areas from the ASTM Template.

Table 7.2 Comparison with ASTM Template

Rating	9	8	7	6	5	4	3	2	1
ASTM Rust Percentage	0.03%	0.10%	0.30%	1%	3%	10%	16%	33%	50%
System Measured Rust	0.05%	0.20%	0.30%	1%	3%	10%	17%	33%	50%

The model implementation in steel bridge coating focused only on identifying rust. Rust is considered the main factor in the deterioration of steel bridges' condition. However, other assessment parameters such as cracking, blistering, and scratching might need completely different techniques and equipment. Using image processing enables surface quality assessment but can not be used for depth analysis for steel bridge coating. For instance, infrared imaging is more appropriate for depth quality assessment.

CHAPTER 8 CONCLUSIONS AND RECOMMENDATIONS

8.1 Conclusions of the Research

This research proposed a hybrid computerized model for highway steel bridge quality assessment and specifically assessment of surface coatings The model employs image processing and neural learning for defects recognition and measurement. The model is supplemented with a statistical assessment plan for proper assessment sampling procedure. The main objective is to propose a system that is objective, quantitative, consistent, and reliable.

This hybrid system applies concepts from the fields of neural learning and image processing. The system overcomes the subjectivity and inconsistency of human visual assessment by analyzing digital images with computers. Unlike the human eye, the computer can recognize and distinguish millions of colors and 256 shades of gray. By analyzing different parameters and characteristics for each pixel in any given image, the model can recognize defect patterns undetectable by humans. Moreover, the system can measure the extent of defect with a reasonable accuracy.

Nevertheless, human expertise can be integrated in the system, specifically during images' threshold selection for classification to accommodate for external factors such as

images' quality and the existence of dirt or drippings on coating. Neural networks enable the system to learn from examples in order to, automatically, perform the recognition task after training.

The basic concept of recognizing defects is assigning all the parts or "pixels" of an image to different classes according to their optical characteristics. The number of classes varies according to the application. For the recognition of rust in steel bridges coating, two classes were used: rust areas and non-rust areas and all images' pixels were classified into these two classes.

This system is expected to improve the quality assessment process by making the process more objective, quantitative, consistent, and accurate. The system can also reduce the assessment time.

8.2 Recommendations for Future Work

♦ A hybrid model for construction quality assessment is presented. However, that model needs to be integrated into a complete computerized quality control system. The integrated computerized system will substantially improve the efficiency and quality of construction assessment. The system will include a central image and technical information database, statistical assessment plan, and remote communication system between office and construction site.

For instance, in applying the hybrid model to steel bridge coating, the integrated computerized system will be composed of four basic modules:

1. A searchable database of all state bridges with digital images and technical information included. Information and images can be retrieved or added from any location in the system, i.e. office or job site.

2. A detailed statistical sampling plan for quality assessment. The plan will determine how to select images numbers, lighting conditions, distances, site conditions and other logistics factors, and images locations. The plan will also determine acceptance and rejection parameters.

3. The hybrid model to provide quantitative measurements for assessment by identifying and measuring defects.

4. Remote communication scheme between offices and sites via an Intranet. An Intranet is a local limited access network using the Internet backbone. Communications with job sites is done using computers (laptop) and cellular phones to provide paperless, fast, and efficient communication scheme.

♦   Other fields of artificial intelligence could be explored to enhance the system performance. Fuzzy logic could be incorporated in the model to formulate a neuro-fuzzy model for assessment. The linguistic to numerical translation capability of fuzzy logic systems may address the decision-making problem more appropriately. Furthermore, fuzzy logic could give more flexibility in classifying objects into classes. For example, the process of threshold selection could be improved by integrating a fuzzy logic system with the image processing algorithm or by using a fuzzy estimator with the neural network.

◆ Moreover, other assessment techniques could be integrated with the hybrid model to improve the assessment process. For example, an infrared imaging module could be integrated with the image processing module and have each module feed a separate neural network and compile both outputs for final assessment. Different outputs can be weighted according to importance, which primarily depends on the specific application.

◆ In some instances, the construction assessment process can be hazardous or in areas hard to access. It might be feasible in the future to integrate the data acquisition module with some robotic apparatus. Robots can reach difficult and risky access areas and they perform tasks more consistently. Integrating the assessment model with a robotic system might improve the model reliability and enhance the potential for application in different kinds of assessment.

LIST OF REFERENCES

Abraham, D., Iseley, T., Prasanth, R., and Wirahadikusumah, R. (1997). "Integrating Sensing Technologies for Underground Utilities Assessment." *Infrastructure Condition Assessment: Art, Science, and Practice. Proceedings of the conference sponsored by the Facilities Management Committee of the Urban Transportation Division of the American Society of Civil Engineers*, August 25-27, 1997, Boston, MA, 316-325.

ASCE (1997). *Artificial Neural Networks for Civil Engineers: Fundamentals and Applications.* American Society of Civil Engineers, New York, NY, 19-43.

ASCE (1988). *Manual of Professional Practice: Quality in the Constructed Project: A Guideline for Owners, Designers, and Constructors.* American Society of Civil Engineers, New York, NY, 17-22.

Bhanu, B. and Lee, S. (1994). *Genetic Learning for Adaptive Image segmentation*, Kluwer Academic Publishers, Boston , MA.

Brumel, H. (1996). "Construction quality in America: A current status report." *Annual Quality Congress Transactions. ASQC,* 1996, Milwaukee, WI, 192-193.

Calder, D. (1997). "Construction quality auditing." *Journal of Management in engineering.* 13(6)26-28.

Chang, L. and Hsie, M. (1995). "Developing acceptance sampling methods for quality construction." *Journal of Construction Engineering and management,* 121(2), 246-253.

Cox, R. and Issa, R. (1996). "Mobile field data acquisition for construction quality control and ISO 9000 certification." *Proceedings of the Third Congress held in conjunction with A/E/C Systems '96 by Journal of Computing in Civil Engineering*, June 17-19, 1996, Anaheim, CA, 1041-1046.

Croall, I. F., and Mason, J. P. (1992). *Industrial Applications of Neural Networks*, Springer-Verlag New York Inc., New York, NY.

De La Blanca, N. (1992). *Pattern recognition and Image Analysis.* World Scientific, River Edge, NJ.

Devijver, P. A. and Kittler, J. (1986). *Pattern Recognition Theory and Applications*, Springer-Verlag New York Inc., New York, NY.

ENR (1998). "Second Quarterly Cost Report." *Engineering News Record*, June 29, 1998, 31-42.

Dry, C., and Mcmillan, W. (1996). "Crack and damage assessment in concrete and polymer matrices using liquids released internally from hollow optical fibers." *Proceedings of SPIE – The International Society for Optical Engineering, Society of Photo-Optical Instrumentation Engineers*, 1996, Bellingham, WA, 448-451.

Fu, H., Xu, Y. (1998). "Multilinguistic handwritten character recognition by Bayesian decision-based neural networks." *IEEE Transactions on Signal Processing*, 46(10),2781-2789.

Garris, M., Wilson, C., and Blue, J. (1998). "Neural network-based systems for handprint OCR applications." *IEEE Transactions of Image Processing*. 7(8), 1097-1112.

Haykin, S. (1999), *Neural Networks: A Comprehensive Foundation*, Prentice Hall, Upper Saddle River, NJ.

Hsie, M. (1994). "Computer Aided Acceptance Planning: Generating Acceptance Parameters and Stratified Sampling Plans through Neural Network Learning Ability and CAD Modeling." *Ph.D. Thesis*. Purdue University, West Lafayette, IN. 1-2.

Hunt, V., Helmicki, A., and Aktan, E. (1997). "Instrumented monitoring and nondestructive evaluation of highway bridges." *Infrastructure Condition Assessment: Art, Science, and Practice. Proceedings of the conference sponsored by the Facilities Management Committee of the Urban Transportation Division of the American Society of Civil Engineers*, August 25-27, 1997, Boston, MA, 121-130.

Kandel, A., and Langholz, G. (1992). *Hybrid Architectures for Intelligent systems*, CRC Press, Boca Raton, FL.

Kosko, B. (1992). *Neural Networks and Fuzzy Systems*, Prentice-Hall, Inc., Englewood Cliffs, NJ.

Korr, R., Buba, J., and Kogut, G. (1983). "Bridge rehabilitation programming by using infrared techniques, bridge inspection and rehabilitation." *Transportation Research Record 899, National Academy of Sciences*.

Lang, J., Voigt, H., Burkhardt, S., and Goebel, R. (1998). "Intelligent Inspection Engine - a real-time real-world visual classifier system." *Conference Proceedings, IEEE World Congress on Computational Intelligence, IEEE International Conference on Neural Networks*, 1998, Piscataway, NJ, 1810-1815.

Liu, J. (1997). "Quality prediction for concrete manufacturing." *Automation in Construction*. 5(6), 491-499.

Looney, C. (1997). *Pattern Recognition using Neural Networks*. Oxford University Press, New York, NY.

Montgomery, D. (1991). *Statistical Quality Control*, John Wiley and Sons, New York, NY.

Navabi, H. and Agarwal, A. (1998). " Adaptive response organizer network for space-time patterns in low level vision." *Neural Networks*, 11(5),825-836.

Ripley, B. D. (1996). *Pattern Recognition and Neural Networks*, Cambridge University Press, New York, NY.

Russ, J.C. (1995). *The Image Processing Handbook*, CRC Press, Inc., Boca Raton, FL

Shubinsky, G. (1994). "Application of Optical Imaging Method for Bridge Maintenance and Inspection." *ITI Technical Report No. 4*, Northwestern University, Evanston, IL.

Steel Structures Painting Manual (1989). *Good Painting Practice*, Steel Structures Painting Council. 1, 280-291, 490-519.

Tsoukalas, H., and Uhrig, E. (1997). *Fuzzy and Neural Approaches in Engineering*, John Wiley & Sons, Inc., New York, NY.

Wang, K., Elliott, R., and Zaniewski, J. (1997). "Automated Survey of Pavement Distress: Current technologies and New Approaches." *Infrastructure Condition Assessment: Art, Science, and Practice. Proceedings of the conference sponsored by the Facilities Management Committee of the Urban Transportation Division of the American Society of Civil Engineers*, August 25-27, 1997, Boston, MA, 514-524.

Webster, A. (1992). *Applied Statistics for Business and Economics*, IRWIN, Boston MA.

Weeks, A. (1996). *Fundamentals of Electronic Image Processing*, SPIE Optical Engineering Press, Bellingham, WA, and IEEE Press, Piscataway, NJ.

Yu, W. (1996). "A Neuro-Fuzzy Knowledge-Based Multi-Criterion Decision Model for Constructability Analysis and Improvement of Construction Technologies" *Ph.D. thesis*, Purdue University, West Lafayette. IN.

Appendix A. Statistical Sampling

A.1 Single Sampling Plan

In a single sampling plan, a predetermined number of samples are taken once and the acceptance or rejection decision is based on data from those samples only.

Notation

N = Lot size

n = sample size

c = acceptance number

d = number of defective

p = fraction of defective items in the lot

The probability of observing d defectives:

$$P(d) = f(d) = \frac{n!}{d!(n-d)!} p^d (1-p)^{n-d} \tag{A.1}$$

The probability of acceptance is the probability that d is less than or equal to c:

$$P_a = P\{d \le c\} = \sum_{d=0}^{c} \frac{n!}{d!(n-d)!} p^d (1-p)^{n-d} \tag{A.2}$$

For example, if $n = 89$, $c = 2$, and $p = 1\%$

The probability of acceptance:

$$P_a = P\{d \le 2\} = \sum_{d=0}^{2} \frac{89!}{d!(89-d)!}(0.01)^d (0.99)^{89-d} = 0.9397$$

Several probabilities of acceptance P_a for different percentages of defective units p are calculated to formulate the Operating Characteristic (OC) curve. The OC curve is the curve that represents the probability of acceptance of a lot (Montgomery 1991). Figure A.1 shows a typical OC curve, which is developed by evaluating the probability of acceptance for various values of fracture of defectives as shown in Table A.1.

Table A.1 Probability of Acceptance for Different p

Fraction Defective p	Probability of Acceptance P_a
0.005	0.9897
0.01	0.9397
0.02	0.7366
0.03	0.4985
0.04	0.3042
0.05	0.1721
0.06	0.0919
0.07	0.0468
0.08	0.023
0.09	0.0109
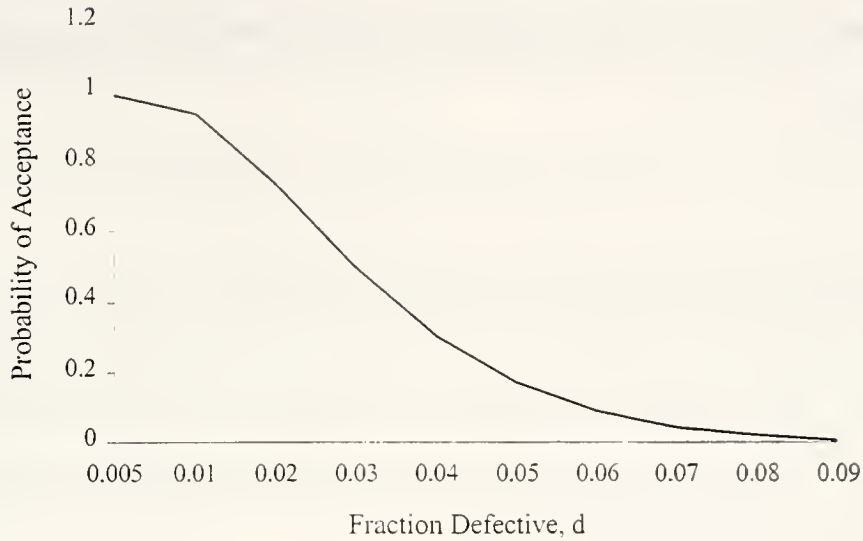
Figure A.1 Operating Characteristic Curve

Moreover, a sampling plan can be designed with a specified OC curve where a probability of acceptance is $(1 - \alpha)$ for lots with fraction defective p_1, and the probability of acceptance β for lots with fraction defective p_2.

$$1-\alpha = \sum_{d=0}^{c} \frac{n!}{d!(n-d)!} p_1^{\,d} (1-p)^{n-d} \tag{A.3}$$

$$\beta = \sum_{d=0}^{c} \frac{n!}{d!(n-d)!} p_2^{\,d} (1-p)^{n-d} \tag{A.4}$$

Commonly, α is referred to as the producer's risk or the contractor's risk of rejecting good quality product and β as owner's risk or the owner's risk of accepting defective products. Statistical nomographs for sampling plans could be used to solve these equations simultaneously and obtain the values of the sample size n and acceptance number c

(Montgomery 1991). In this research, the parameters for the sampling plan are adopted from previously developed attribute double sampling plan for coating quality assessment (Chang and Hsie 1995).

<div align="center">A.2 Attribute Double Sampling Plan</div>

Double sampling plan utilizes two sample sizes. If the information from the first sampling clearly implies acceptance or rejection, the decision will be based on one sampling. Contrarily, if the information gathered from the first sampling is between acceptance and rejection, a second sampling will be performed.

Notation

N = Lot size

n_1 = sample size on the first sample

n_2 = sample size on the second sample

c_1 = acceptance number on the first sample

c_2 = acceptance number on the second sample

d_1 = number of defective on the first sample

d_2 = number of defective on the second sample

p = fraction of defective items in the lot

The probability of acceptance for the combined sample

$$P_a = P_a^1 + P_a^2 \tag{A.5}$$

For example, if $n_1=50$, $c_1=1$, $n_2=100$, $c_2=3$, and $p=5\%$

$$P_a^1 = P\{d \leq 1\} = \sum_{d_1=0}^{1} \frac{50!}{d!(50-d_1)!}(0.05)^{d_1}(0.95)^{50-d_1} = 0.279$$

The second sample is drawn only if there are 2 or 3 defectives on the first sample or:

$$c_1 \leq d_1 \leq c_2 \tag{A.6}$$

The lot is accepted in the second sample only if:

$d_1 = 2$ and $d_2 = 0$ or 1

$$P\{d_1 = 2, d_2 \leq 1\} = P\{d_1 = 2\}.P\{d_2 \leq 1\}$$

$$= \frac{50!}{2!48!}(0.05)^2(0.95)^{48} * \sum_{d_2=0}^{1} \frac{100!}{d_2!(100-d_2)!}(0.05)^{d_2}(0.95)^{100-d_2} = 0.009$$

or if:

$d_1 = 3$ and $d_2 = 0$

$$P\{d_1 = 3, d_2 = 0\} = P\{d_1 = 3\}.P\{d_2 = 0\}$$

$$= \frac{50!}{2!47!}(0.05)^3(0.95)^{47} \frac{100!}{0!100!}(0.05)^0(0.95)^{100} = 0.001$$

The probability of acceptance for the second sample is

$$P_a^2 = P\{d_1 = 2, d_2 \leq 1\} + P\{d_1 = 3, d_2 = 0\}$$

$$= 0.009 + 0.001 = 0.010$$

The probability of acceptance for the lot:

$$P_a = P_a^1 + P_a^2 = 0.279 + 0.010 = 0.289$$

Other probabilities of acceptance for different fraction defective are calculated similarly to plot the OC curve as described in section A.1.

Appendix B. Input and Output

B.1 Training Images Set
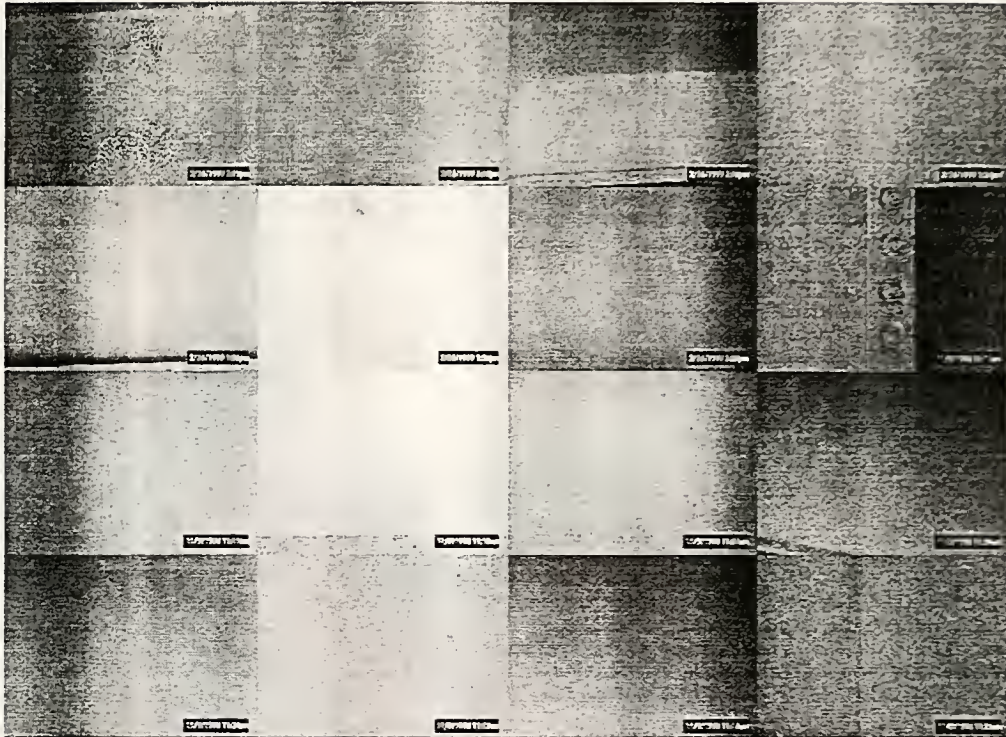
Figure B.1 Images 'Montage'

B.2 Example Image Input Data (Histogram Format)

Intensity	Count gray
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0
21	69
22	0
23	0
24	0
25	0
26	0
27	0
28	0
29	0
30	468
31	0
32	0
33	0
34	0
35	895
36	0
37	0

38	0
39	3593
40	0
41	86
42	0
43	96
44	0
45	0
46	0
47	0
48	0
49	421
50	134
51	0
52	0
53	0
54	4810
55	0
56	0
57	700
58	0
59	0
60	0
61	0
62	0
63	0
64	1013
65	786
66	0
67	0
68	0
69	352
70	702
71	0
72	256
73	0
74	1546
75	0
76	0
77	1305
78	0
79	3250
80	1214
81	2349
82	0

83	1337
84	0
85	2293
86	731
87	769
88	0
89	0
90	1220
91	3229
92	0
93	0
94	2472
95	1206
96	739
97	2172
98	3928
99	391
100	945
101	1888
102	3818
103	245
104	3070
105	0
106	959
107	3611
108	1241
109	437
110	1335
111	2330
112	4183
113	402
114	2689
115	1144
116	491
117	826
118	3669
119	3297
120	2741
121	2236
122	2802
123	0
124	3268
125	3317
126	0
127	1723

128	2151
129	3318
130	2481
131	4228
132	145
133	8243
134	0
135	0
136	0
137	4186
138	5430
139	565
140	0
141	5737
142	7299
143	1719
144	6672
145	3907
146	0
147	2115
148	4877
149	313
150	4081
151	0
152	655
153	2974
154	1649
155	5246
156	0
157	1691
158	91
159	5688
160	2780
161	3088
162	0
163	1094
164	2385
165	2237
166	0
167	0
168	574
169	5417
170	0
171	0
172	5555

173	0
174	0
175	678
176	0
177	0
178	0
179	3149
180	4381
181	1840
182	0
183	7055
184	0
185	0
186	0
187	2741
188	2806
189	526
190	0
191	1525
192	690
193	0
194	0
195	0
196	0
197	5618
198	355
199	47
200	0
201	0
202	2108
203	0
204	0
205	639
206	0
207	0
208	0
209	0
210	0
211	0
212	0
213	0
214	0
215	0
216	0
217	0

218	0
219	0
220	0
221	0
222	0
223	0
224	0
225	0
226	0
227	0
228	0
229	0
230	0
231	0
232	52
233	0
234	0
235	0
236	0
237	0
238	0
239	0
240	0
241	0
242	0
243	0
244	0
245	0
246	0
247	0
248	0
249	0
250	0
251	0
252	0
253	0
254	0
255	0

B.3 Example Image Output Data (Histogram Format)

Intensity	Count	Mapped	Count
0	0	0	0
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
10	0	0	0
11	0	0	0
12	0	0	0
13	0	0	0
14	0	0	0
15	0	0	0
16	0	0	0
17	0	0	0
18	0	0	0
19	0	0	0
20	0	0	0
21	69	0	69
22	0	0	0
23	0	0	0
24	0	0	0
25	0	0	0
26	0	0	0
27	0	0	0
28	0	0	0
29	0	0	0
30	468	0	468
31	0	0	0
32	0	0	0
33	0	0	0
34	0	0	0
35	895	0	895
36	0	0	0
37	0	0	0
38	0	0	0
39	3593	0	3593

40	0	0	0
41	86	0	86
42	0	0	0
43	96	0	96
44	0	0	0
45	0	0	0
46	0	0	0
47	0	0	0
48	0	0	0
49	421	0	421
50	134	0	134
51	0	0	0
52	0	0	0
53	0	0	0
54	4810	0	4810
55	0	0	0
56	0	0	0
57	700	0	700
58	0	0	0
59	0	0	0
60	0	0	0
61	0	0	0
62	0	0	0
63	0	0	0
64	1013	0	1013
65	786	0	786
66	0	0	0
67	0	0	0
68	0	0	0
69	352	0	352
70	702	0	702
71	0	0	0
72	256	0	256
73	0	0	0
74	1546	0	1546
75	0	0	0
76	0	0	0
77	1305	0	1305
78	0	0	0
79	3250	0	3250
80	1214	0	1214
81	2349	0	2349
82	0	0	0
83	1337	0	1337
84	0	0	0

85	2293	0	2293
86	731	0	731
87	769	0	769
88	0	0	0
89	0	0	0
90	1220	0	1220
91	3229	0	3229
92	0	0	0
93	0	0	0
94	2472	0	2472
95	1206	0	1206
96	739	0	739
97	2172	0	2172
98	3928	0	3928
99	391	0	391
100	945	0	945
101	1888	0	1888
102	3818	0	3818
103	245	0	245
104	3070	0	3070
105	0	0	0
106	959	0	959
107	3611	0	3611
108	1241	0	1241
109	437	0	437
110	1335	0	1335
111	2330	0	2330
112	4183	0	4183
113	402	0	402
114	2689	0	2689
115	1144	0	1144
116	491	0	491
117	826	0	826
118	3669	0	3669
119	3297	0	3297
120	2741	0	2741
121	2236	0	2236
122	2802	0	2802
123	0	0	0
124	3268	0	3268
125	3317	0	3317
126	0	0	0
127	1723	0	1723
128	2151	0	2151
129	3318	1	3318

130	2481	1	2481
131	4228	1	4228
132	145	1	145
133	8243	1	8243
134	0	1	0
135	0	1	0
136	0	1	0
137	4186	1	4186
138	5430	1	5430
139	565	1	565
140	0	1	0
141	5737	1	5737
142	7299	1	7299
143	1719	1	1719
144	6672	1	6672
145	3907	1	3907
146	0	1	0
147	2115	1	2115
148	4877	1	4877
149	313	1	313
150	4081	1	4081
151	0	1	0
152	655	1	655
153	2974	1	2974
154	1649	1	1649
155	5246	1	5246
156	0	1	0
157	1691	1	1691
158	91	1	91
159	5688	1	5688
160	2780	1	2780
161	3088	1	3088
162	0	1	0
163	1094	1	1094
164	2385	1	2385
165	2237	1	2237
166	0	1	0
167	0	1	0
168	574	1	574
169	5417	1	5417
170	0	1	0
171	0	1	0
172	5555	1	5555
173	0	1	0
174	0	1	0

175	678	1	678
176	0	1	0
177	0	1	0
178	0	1	0
179	3149	1	3149
180	4381	1	4381
181	1840	1	1840
182	0	1	0
183	7055	1	7055
184	0	1	0
185	0	1	0
186	0	1	0
187	2741	1	2741
188	2806	1	2806
189	526	1	526
190	0	1	0
191	1525	1	1525
192	690	1	690
193	0	1	0
194	0	1	0
195	0	1	0
196	0	1	0
197	5618	1	5618
198	355	1	355
199	47	1	47
200	0	1	0
201	0	1	0
202	2108	1	2108
203	0	1	0
204	0	1	0
205	639	1	639
206	0	1	0
207	0	1	0
208	0	1	0
209	0	1	0
210	0	1	0
211	0	1	0
212	0	1	0
213	0	1	0
214	0	1	0
215	0	1	0
216	0	1	0
217	0	1	0
218	0	1	0
219	0	1	0

220	0	1	0
221	0	1	0
222	0	1	0
223	0	1	0
224	0	1	0
225	0	1	0
226	0	1	0
227	0	1	0
228	0	1	0
229	0	1	0
230	0	1	0
231	0	1	0
232	52	1	52
233	0	1	0
234	0	1	0
235	0	1	0
236	0	1	0
237	0	1	0
238	0	1	0
239	0	1	0
240	0	1	0
241	0	1	0
242	0	1	0
243	0	1	0
244	0	1	0
245	0	1	0
246	0	1	0
247	0	1	0
248	0	1	0
249	0	1	0
250	0	1	0
251	0	1	0
252	0	1	0
253	0	1	0
254	0	1	0
255	0	1	0

B.4 Example Thresholded Image
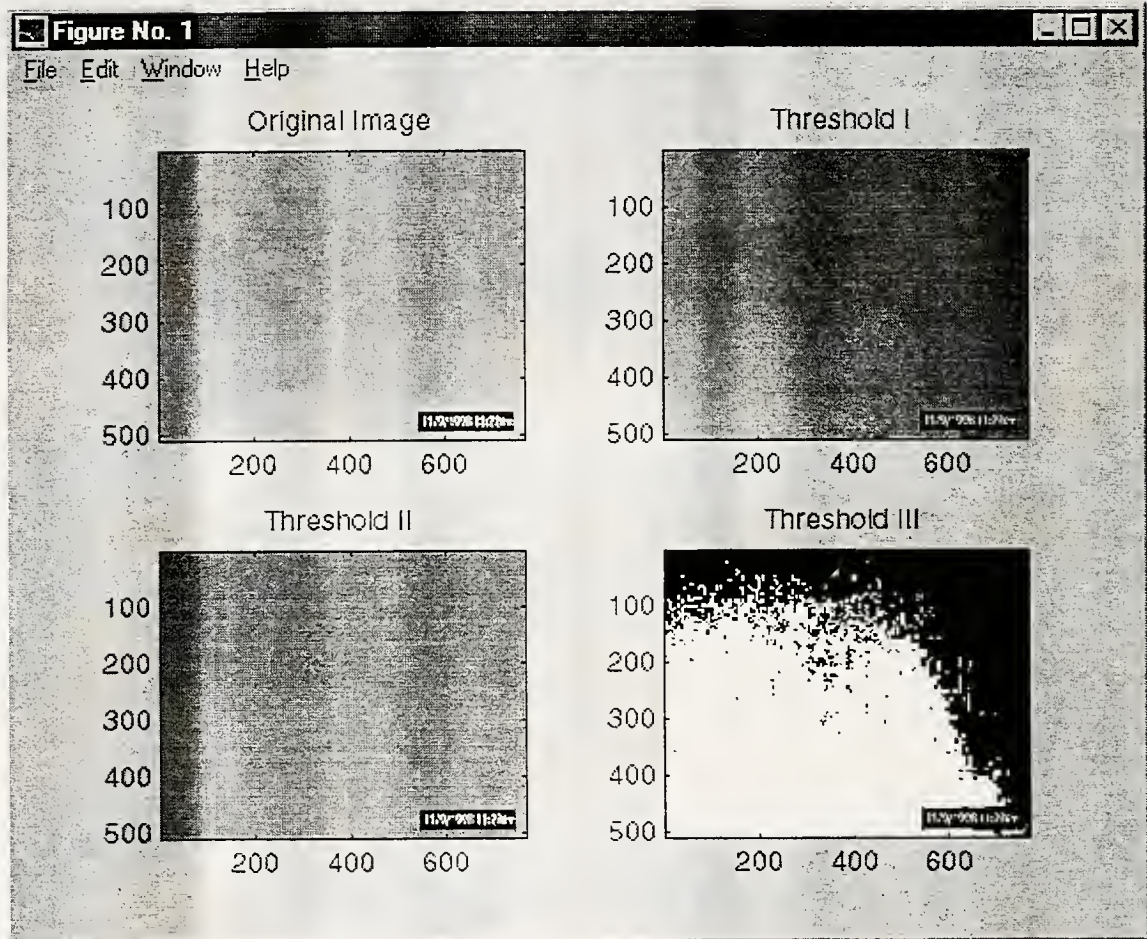
Figure B.2 Example Image Threshold (MatLab Screen)

B.5 Example Training Results
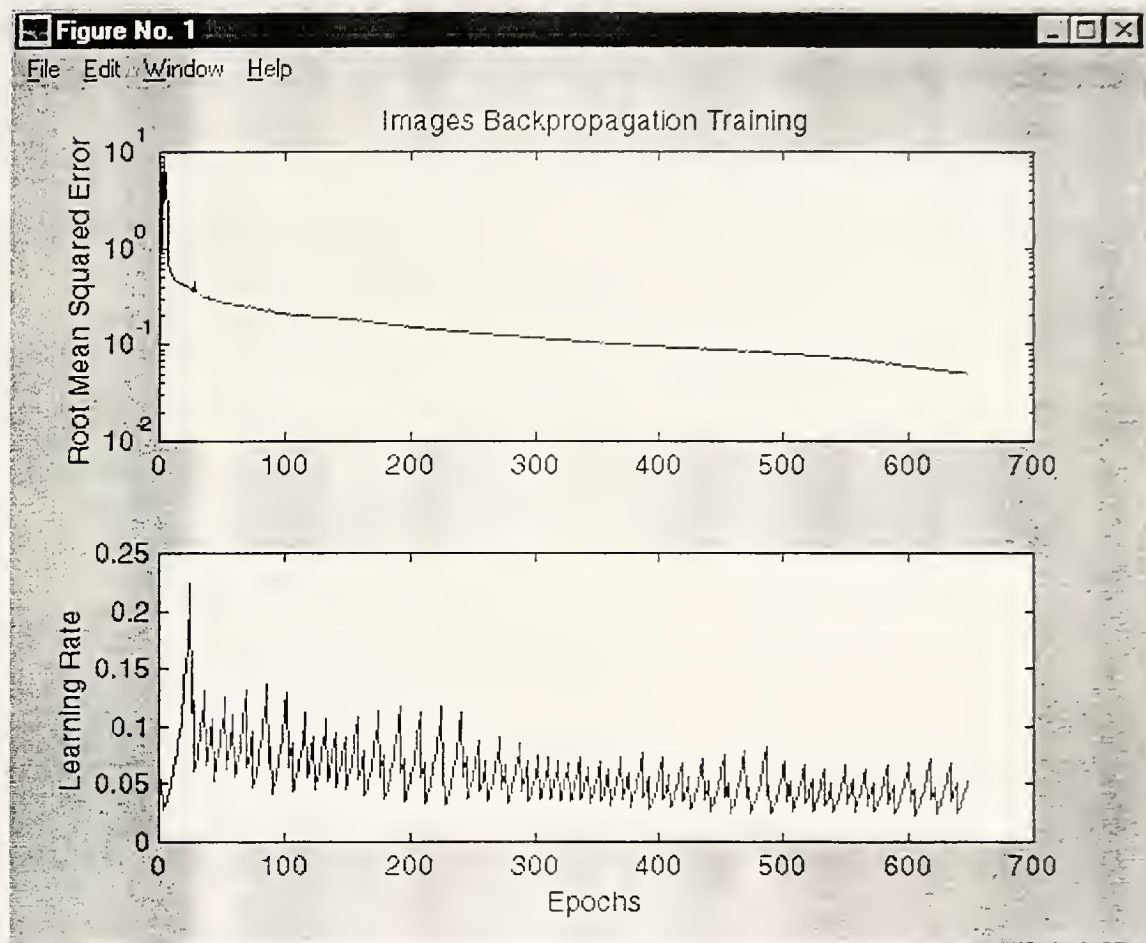
Figure B.3 Images Neural Training (MatLab Screen)

Appendix C. Digital Camera

C.1 System Requirements

WINDOWS Systems:
- WINDOWS 95, 98 or NT 4.0 Software
- PC with PENTIUM 90 MHz or greater processor
- 16 MB or more of RAM
- 70 MB hard disk space available
- CD-ROM drive for software installation
- 256-color monitor (16-bit color recommended)
- Available serial port, USB port, IrDA connection or PC card reader

MACINTOSH Systems:
- Power MACINTOSH System, MacOS 7.6 or later (serial); MacOS 8.5 or later (USB)
- 32 MB or more of RAM
- 70 MB hard disk space available
- CD-ROM drive for software installation
- 256-color monitor (16-bit color recommended)
- Available serial port, USB port, or PC card reader

KODAK Picture Easy Software:
- PC with PENTIUM 90 MHz or greater processor
- WINDOWS 98 with 24 MB of RAM, or WINDOWS 95 with 16 MB of RAM, or WINDOWS NT 4.0 with 32 MB of RAM
- minimum of 70 MB of free disk space before Picture Easy Software installation
- CD-ROM drive for software installation
- 256-color monitor (16-bit color recommended)
- 800 x 600 screen resolution recommended

C.2 Specifications

- **CCD Resolution**
  1548 x 1032 pixels
- **Image Resolution**
  1536 x 1024 pixels, 1152 x 768 pixels, 768 x 512 pixels
- **Image Compression**
  Best, better, good
- **Image Storage**
  8 MB KODAK Picture Card included. Stores up to 95 pictures
- **Viewfinder**
  2.0" TFT color LCD for review and preview, plus real-image optical viewfinder
- **Lens**
  Auto focus 3X zoom
- **Digital Enhancement**
  2X digital zoom
- **Lens Focal Length**
  38 mm to 115 mm equivalent

- **Focus Range**
  12" (0.3 m) to infinity
- **Exposure**
  Auto or manual exposure (+/- 2 EV in 0.5-EV increments) with automatic white balance
- **Shutter Speed**
  1/4 to 1/400 second
- **Aperture Range**
  Wide: f/3.0 to f/14.0;
  Tele: f/4.7 to f/22.0
- **ISO Equivalent**
  100
- **Scripting**
  Digita text-based language, extends functionality by automating camera
- **Burst Capture**
  Max 2 pictures at high and medium res, max 8 pictures at standard res, frame rate selectable from 0.1 to 3 frames per second
- **Time Lapse**
  Set to capture shot at pre-defined intervals; play back as movie on camera
- **Orientation Sensor**
  Automatically rotates image right-side-up at LCD and host
- **Picture Overlay**
  Time/date stamp, text, logo/graphic; user-selectable location
- **Albums**
  In-camera albums keep pictures organized. Albums transfer to Picture Easy Software
- **Self-Timer**
  10 seconds
- **Tripod Mount**
  Standard
- **Flash**
  Strobe flash (auto, red-eye, fill, off); range up to 9.8' (3.0 m); optional flash sync cable supports external flash
- **File Formats**
  JPEG (EXIF), FPX
- **User Interface**
  Graphical, menu-driven, easy to navigate
- **Video Out**
  NTSC, PAL: live view
- **Audio**
  Record and playback
- **Interface**
  Serial, IrDA (camera-to-camera or camera-to-PC), USB, PC Card
- **Power**
  4 AA batteries, AC adapter (included in box)
- **Dimensions**
  4.6"(w) x 2.2"(d) x 4.2"(h);
  118 mm (w) x 57 mm (d) x 106 mm (h)
- **Certifications**
  VCCI, CE, FCC Class B, C-Tick, ICES-003 Class B, CCIB
- **Weight**
  1.2 lb. (525 g) without batteries
- **Warranty**
  One year

Appendix D. Implementation Procedures

In practice, the implementation can be categorized into two groups: sampling and image processing. Sampling includes the image acquisition and the associated sampling plan; image processing analyzes the image taken from the bridge to be inspected and helps with the rating and acceptance decision.
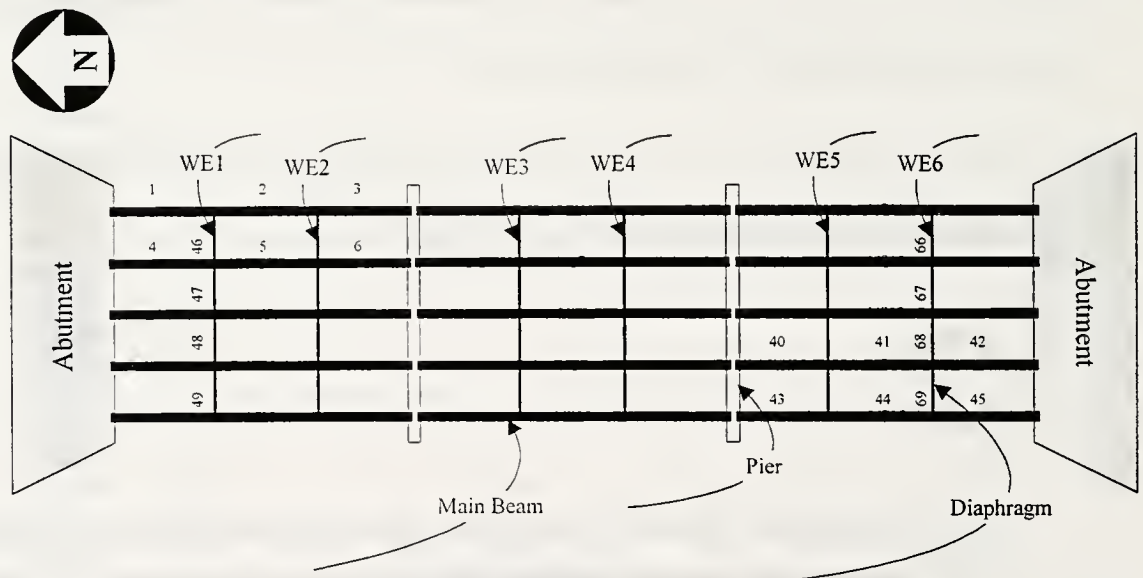
D.1 Sampling

Sampling Option I:

1) *Details of the bridge to be inspected:* The numbers of beams and diaphragms should be known before the start of sampling procedure.

2) *Coding:* As mentioned in 6.2.3, steel beams and diaphragms are numbers according to their geographical direction starting at the top left corner.
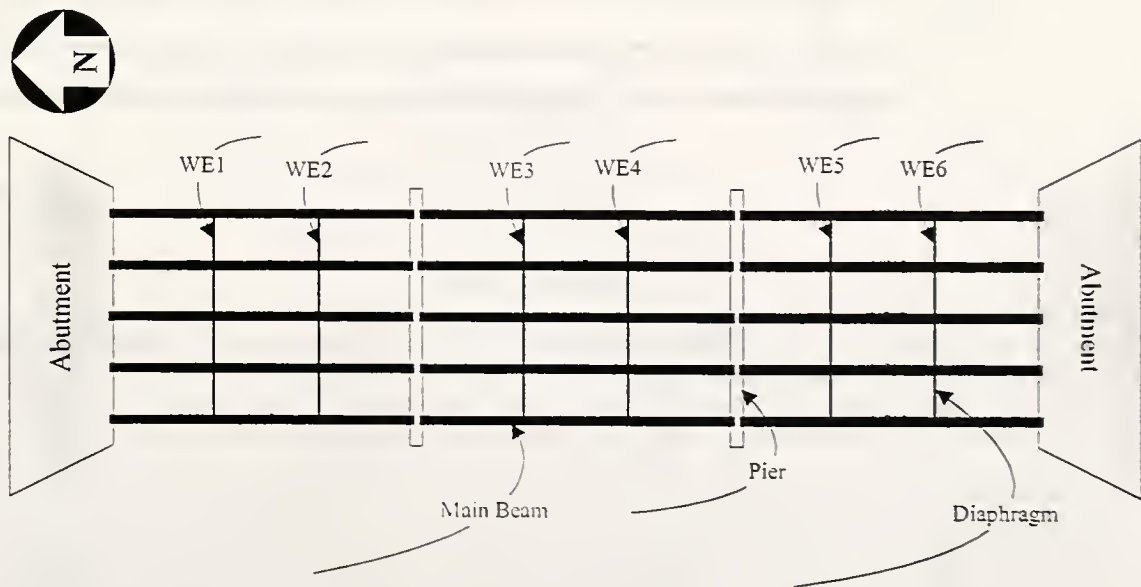
3) *Sampling table:* After the coding is finished, a detailed sampling table should be made for the convenience of random sampling. In the sampling table, all coded beams and diaphragms are numbered sequentially based on "sections". In the picture shown, there are three sections in each beam and four sections in each diaphragm. The way of numbering sections also starts from the top left corner with diaphragms numbered after beams. In the picture, there are 15 beams and 6 diaphragms. Therefore, the numbers of beam sections start from 1 to 45, and the numbers of diaphragm sections are from 46 to 69. To make a more accurate sampling table, users can number both sides of each section and the total number will double.

Sampling Table

Number	Beam/Diaphragm	Number	Beam/Diaphragm
1	NS1	46	WE1
2	NS1	47	WE1
3	NS1	48	WE1
4	NS2	49	WE1
5	NS2	50	WE2
.	.	.	.
42	NS14	66	WE6
43	NS15	67	WE6
44	NS15	68	WE6
45	NS15	69	WE6

4) *Sampling plan:* The double sampling plan is adopted for acceptance assessment. The flow chart is as follows. First, take 10 samples and count the number of defects. If the defect number is 0 or 1, the painting work of the bridge is accepted. If the defect number is equal to or larger than 3, the painting work is rejected. If the defect number is equal to 2, a second set of 10 samples should be taken. In the second set of samples, if the defect number is 0, 1, or 2, the painting work is accepted. If the defect number is equal to or larger than 3, the painting work is rejected.

```
              ┌─────────────────────────────┐
              │ Take 10 measurements and    │
              │ Count the Number of Defect X1│
              └─────────────────────────────┘
                            │
         ┌──────────────────┼──────────────────┐
         │                  │                  │
   ┌──────────┐       ┌──────────┐       ┌──────────┐
   │ X1=3, 4,…│       │  X1=2    │       │ X1=0, 1  │
   └──────────┘       └──────────┘       └──────────┘
         │                  │                  │
   ┌──────────┐    ┌─────────────────┐   ┌──────────┐
   │  Reject  │    │ Take the second │   │  Accept  │
   └──────────┘    │ 10 measurements │   └──────────┘
                   │ and Count the   │
                   │ Number of Defect X2│
                   └─────────────────┘
                            │
                 ┌──────────┴──────────┐
                 │                     │
           ┌─────────────┐      ┌─────────────┐
           │ X2=3, 4, 5,…│      │ X2=0, 1, 2  │
           └─────────────┘      └─────────────┘
                 │                     │
           ┌──────────┐         ┌──────────┐
           │  Reject  │         │  Accept  │
           └──────────┘         └──────────┘
```
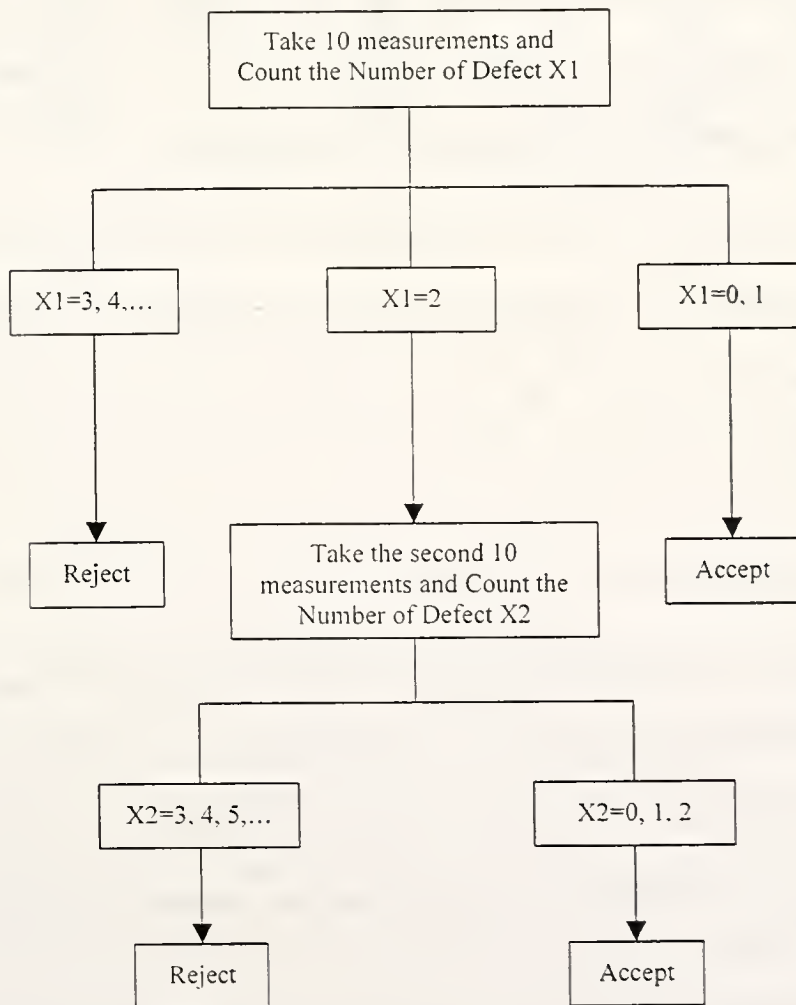
5) *Random sampling:* The random number generator is used for random sampling. It evenly generates random numbers between 0 and 1. (A random number could be 0 but has to be less than 1.) A random number could be converted to a section number by the following equation:

$$Section\ Number = [\ Random\ Number * 100 * (\ N\ /\ 100)\ ]$$
$$= [Random\ Number * N\ ]$$

where [ ] is a rounding operator, and N indicates the total section number.

In this case, N is equal to 69. The first set of 10 samples can be selected by taking the first 10 different section numbers generated by the random number generator. If a second set is required, it can be generated with the same way.

6) *Image acquisition:* After the section numbers are determined, images could be taken with a digital camera three feet from the sections with the flash on. To ensure all the image samples have similar brightness, the flash should be switched on throughout the image-taking process.

Sampling Option II:

1) *Selecting lanes.* The first step in sampling option II is to select the lanes. Based on the centerline of the bridge and the lane lines on both sides, the whole bridge can be broken into four sections, sections I, II, III and IV. Sections I and III belong to one group, and sections II and IV belong to the other one. Images can be taken randomly from either I and III, or II and IV.
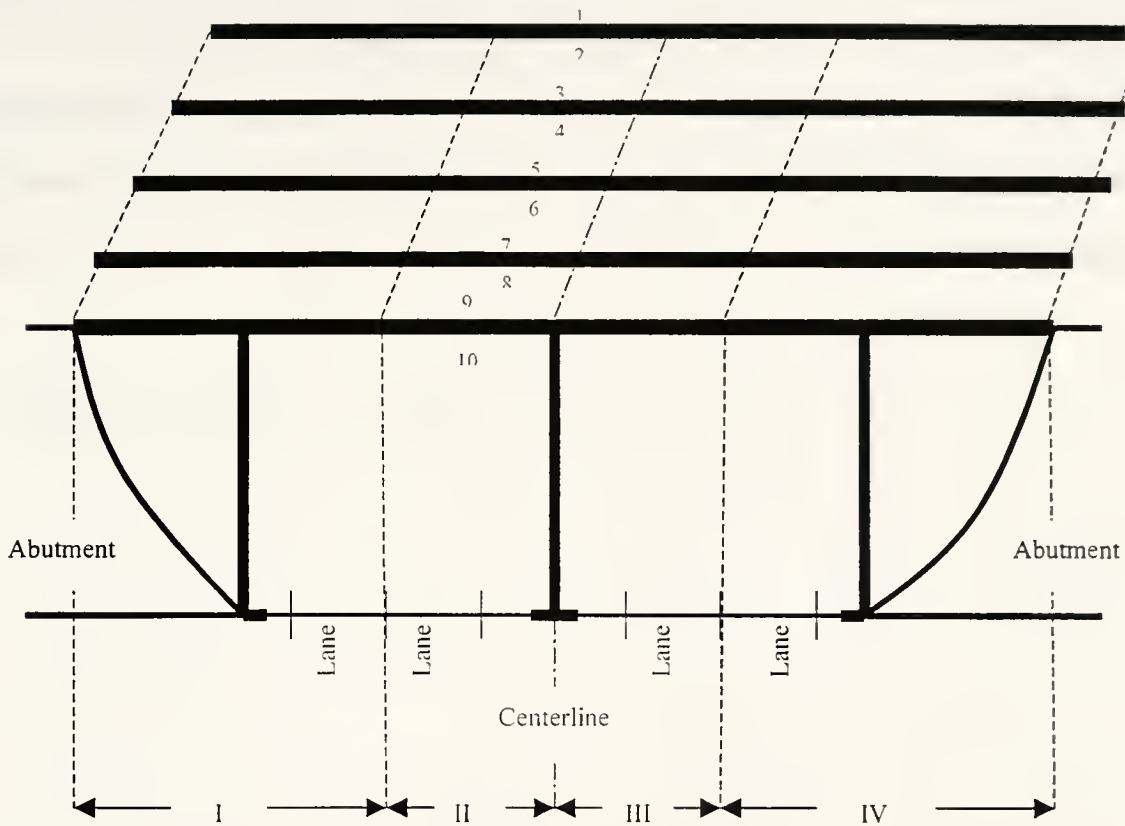
2) *Deciding lots*. After the group of sections is selected, lots from where the samples are taken can be decided. Each side of a beam in a section is defined as a lot (as shown in the above figure). Thus, there are two lots for each beam within a selected section. If there are four beams in a section, eight lots are contained in this section. For the same reason, if seven beams are in a section, it has fourteen lots in this section. Ten lots are selected from each of the two sections in the group chosen. If the number of lots in a section is larger than ten, ten lots are randomly selected from this section. If the number of lots is ten, images are taken directly from the ten lots without random selection process. If the number of lots in a section is less than ten, all lots are chosen plus randomly selecting the difference between ten and

the lot number from these lots. For example, if eight lots are in a section, all eight lots are selected as well as randomly selecting two lots out of the eight.

3) *Taking images.* Each lot is broken into five locations and three images are taken from each location. The following figure depicts the five locations. There are fifteen images taken from each lot, and thirty images taken from a beam in a section.
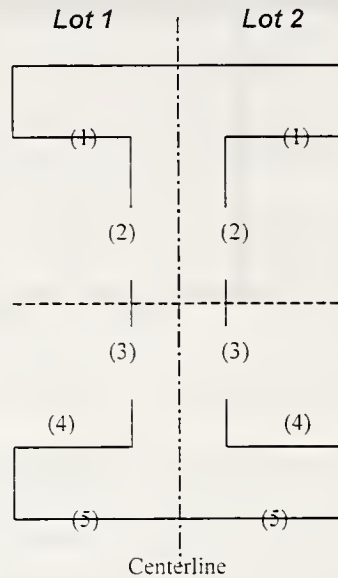
4) *Acceptance/Rejection.* The acceptance or rejection is made based on beams. After taking thirty images from a beam, send them for image processing and get the corresponding rust percentages. Take the average of the thirty rust percentages. If the average is larger than 1%, the beam is rejected. Otherwise, the beam is accepted.

$$Summation(\% \ Rust) / 30 > 1\% \rightarrow Reject \ the \ beam$$

$$Summation(\% \ Rust) / 30 \leq 1\% \rightarrow Accept \ the \ beam$$

D.2 Image Processing

After images are acquired, it is ready to go through the image analysis. Before the image analysis starts, the images should be transferred from the digital camera to the computer that runs the image analysis. Matrox Inspector software is used for the image analysis. After the image analysis is complete, the defect number can be counted. Acceptance, rejection, or the requirement of a second set of image samples could then be determined. The step-by-step procedure for using Matrox Inspector is presented below.

Step-by-Step Procedures for the Use of Matrox Inspector

1. Start the Matrox Inspector program.
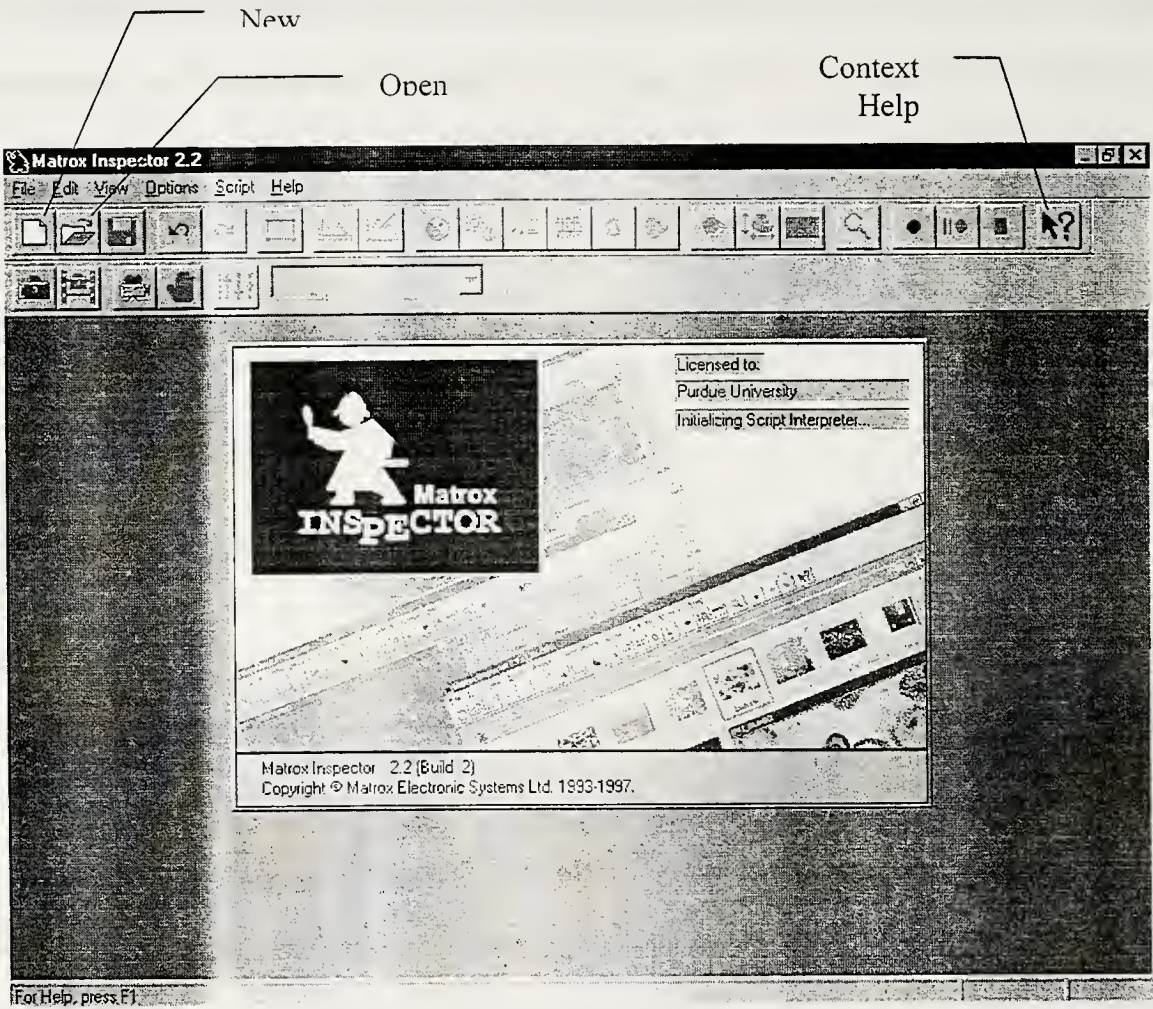
2. Click the "Open" button to launch the "Open" window.
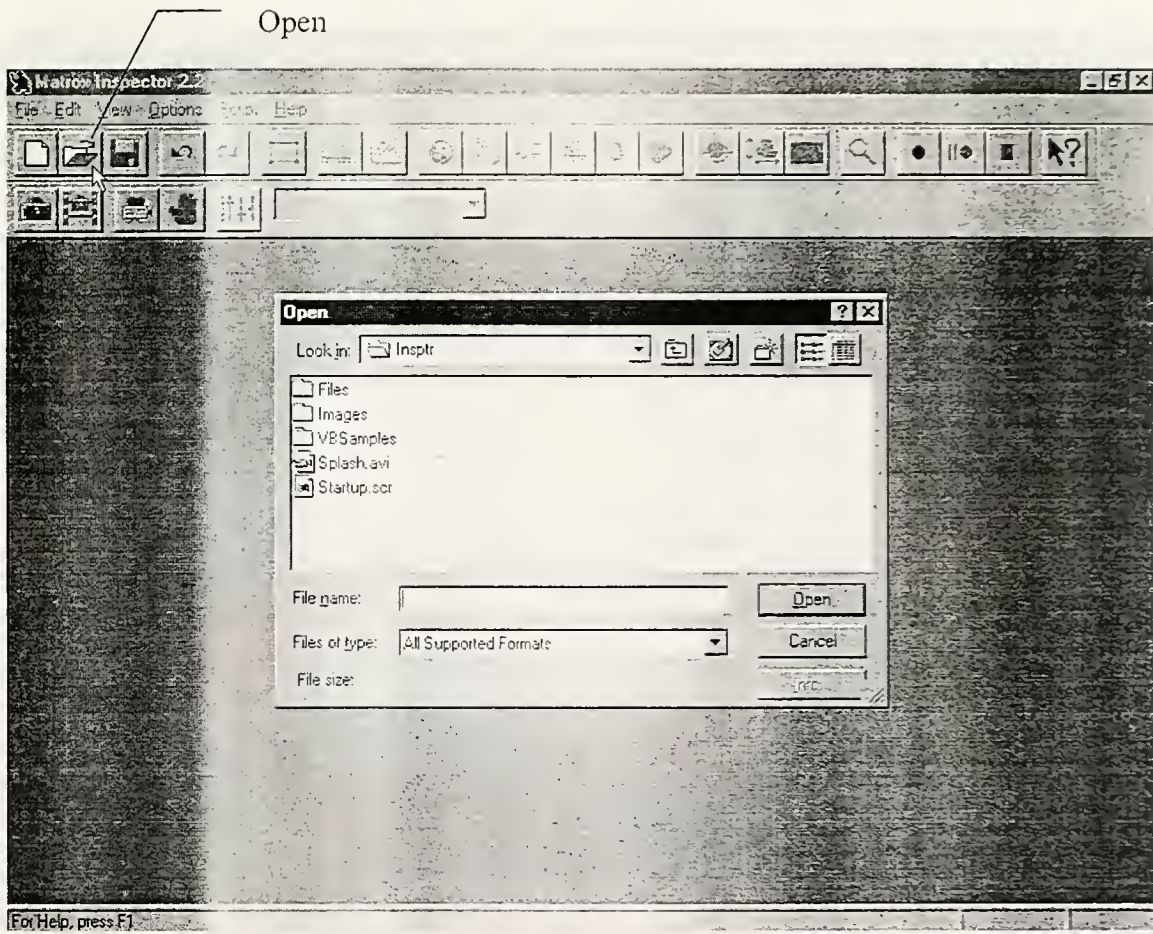
Open

3. Select the image to be analyzed and then click the "Open" button in the "Open" window.
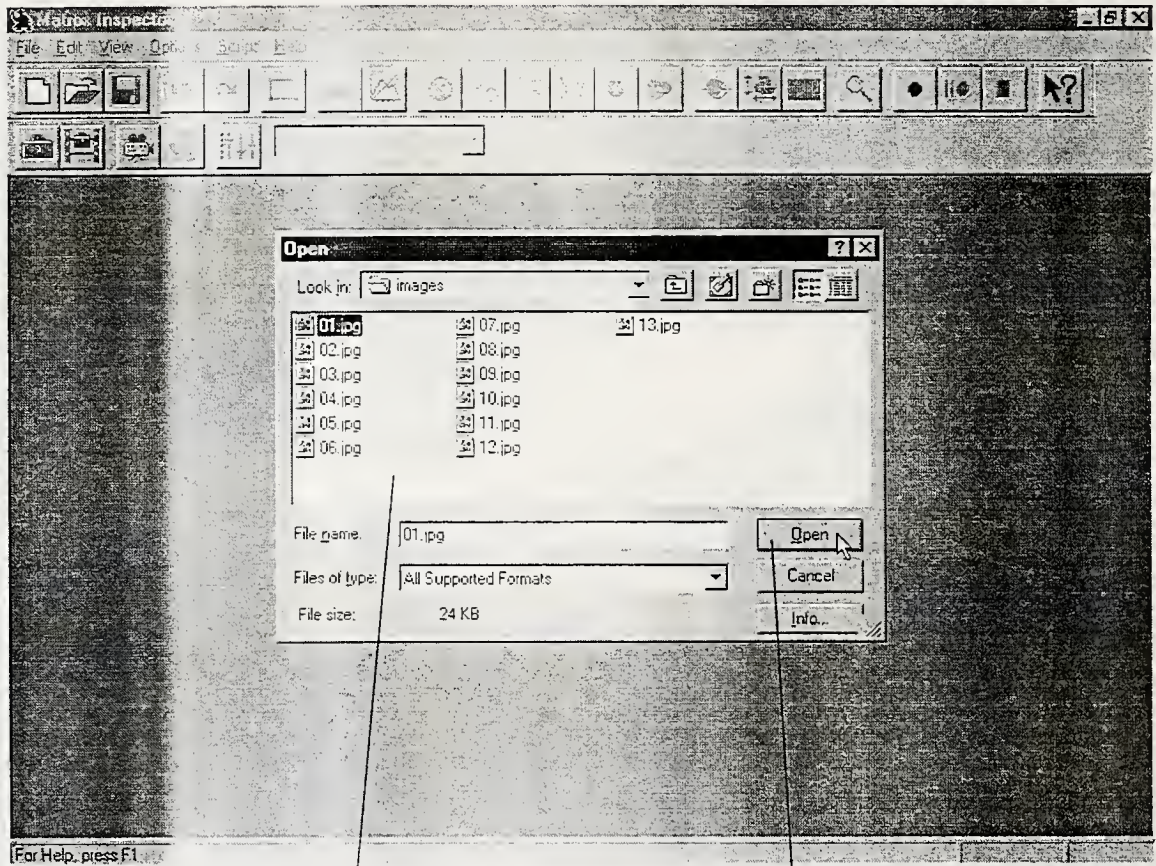
Select the
image to be
analyzed

Click on this
button to open
the image

4. Select "Conversion" under the "Tools" header to convert the image to a new grayscale one.

Conversion button;
Users can also select
Conversion by
clicking on this button
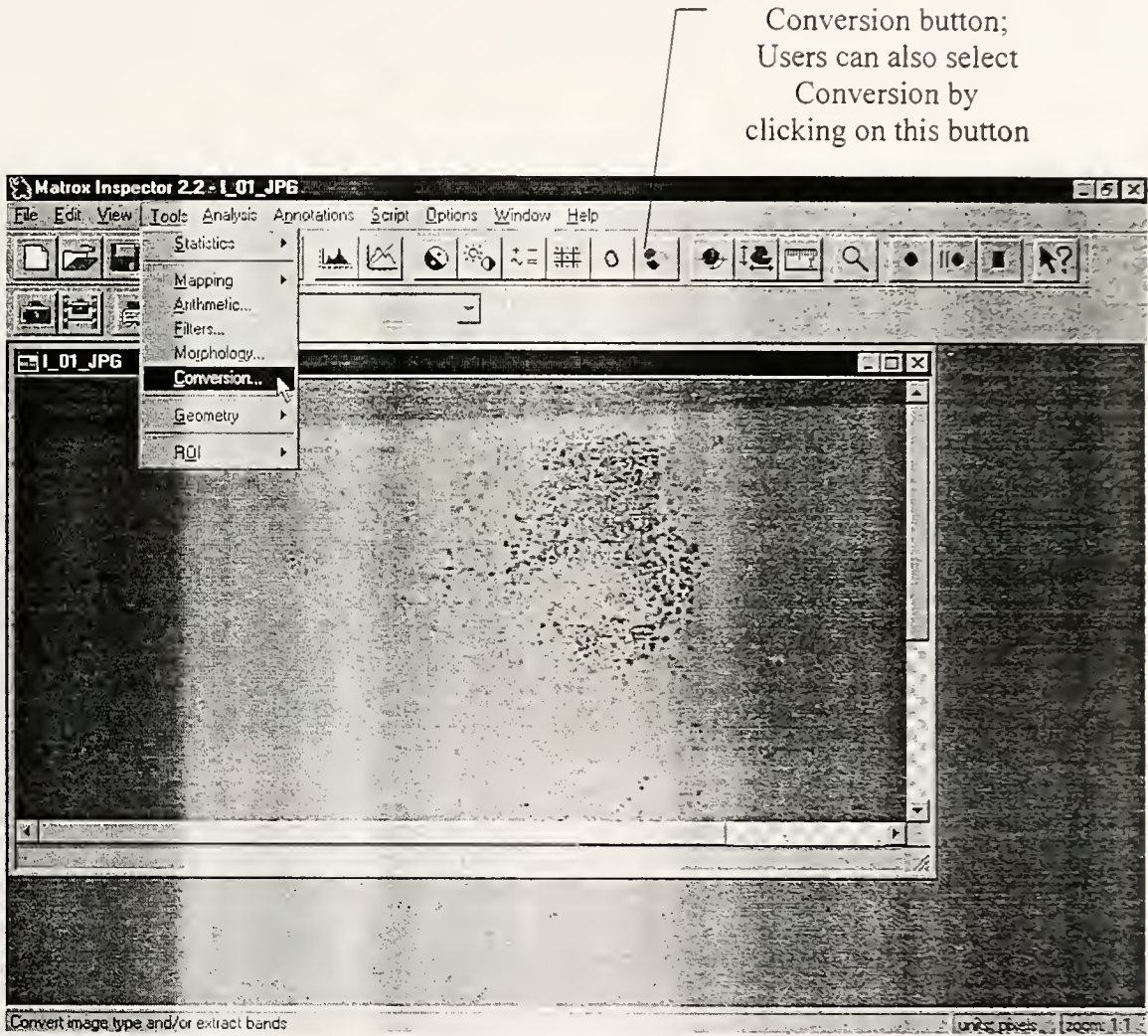
5. After "Conversion" is chosen, the "Processing Operations" window will pop up. Left-click on

the "D" button as indicated by the cursor, and a list of options will appear.
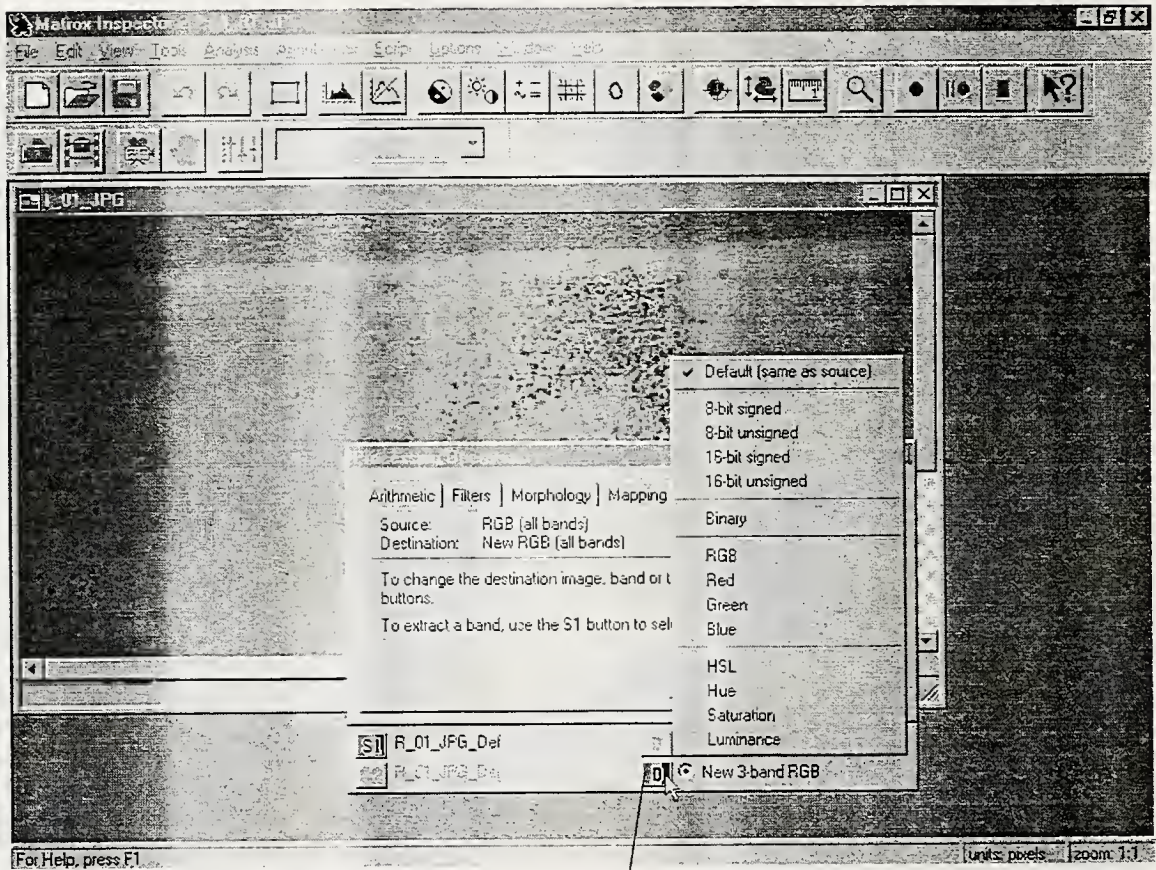
Clicking on this button
pops up a list of options
that could convert the
original image to a new
monochrome/RGB one

6. Select the "8-bit unsigned" option on the list.
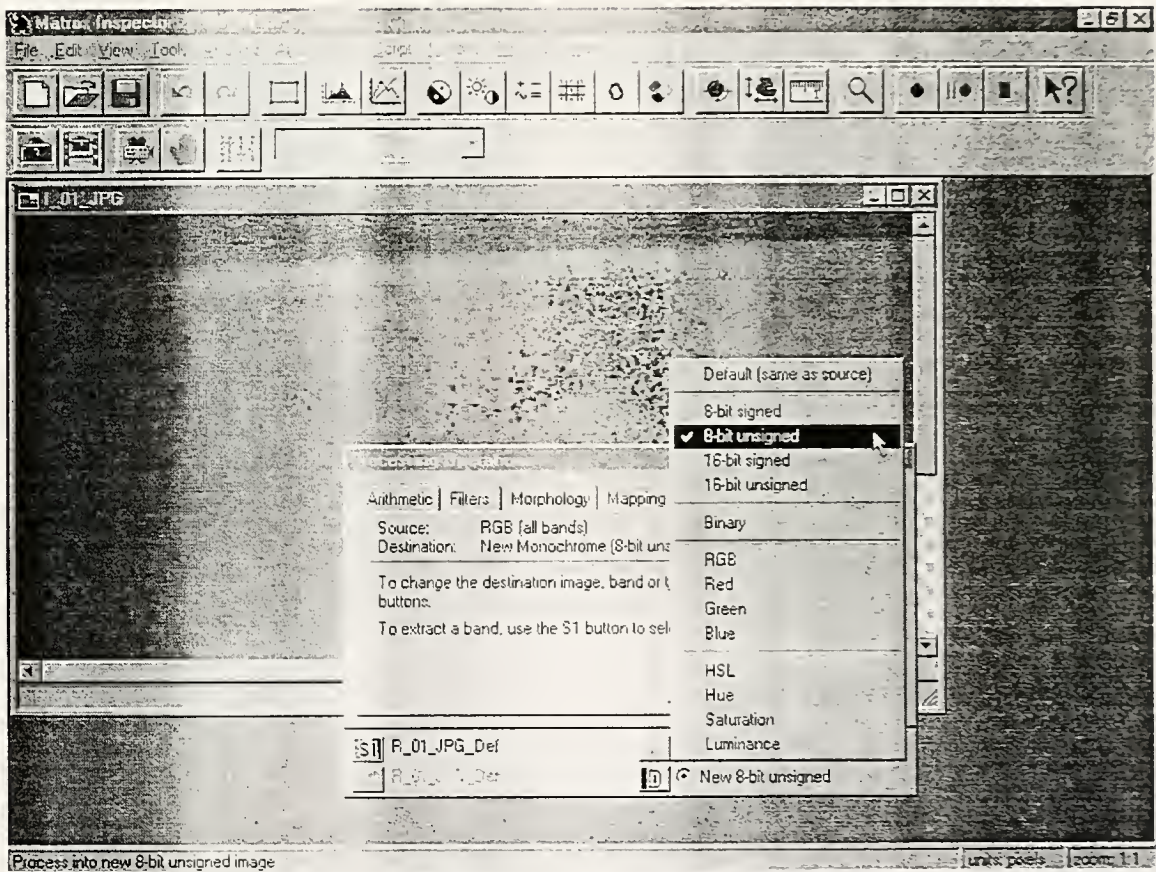
7. Click on the "Apply" button and a new grayscale (monochrome) image will be generated.
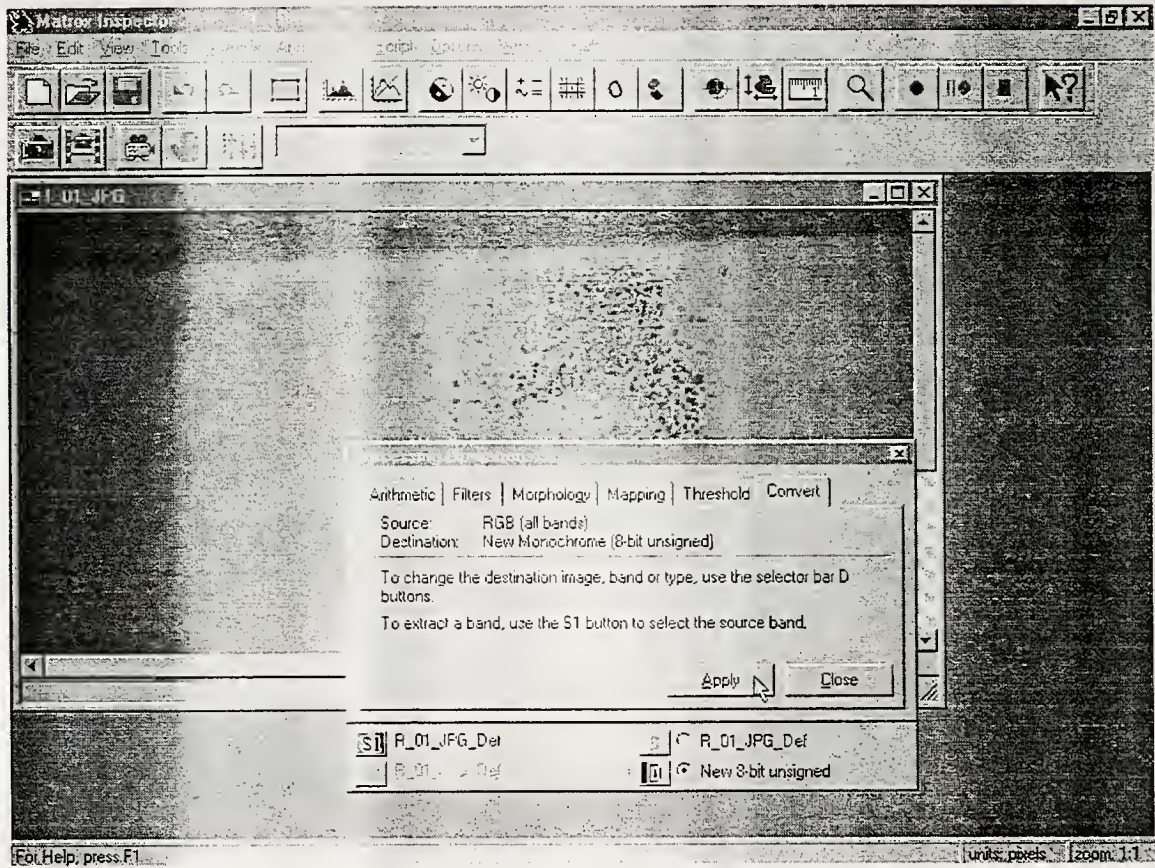
8. Click on the "Close" button to close the "Processing Operations" window.
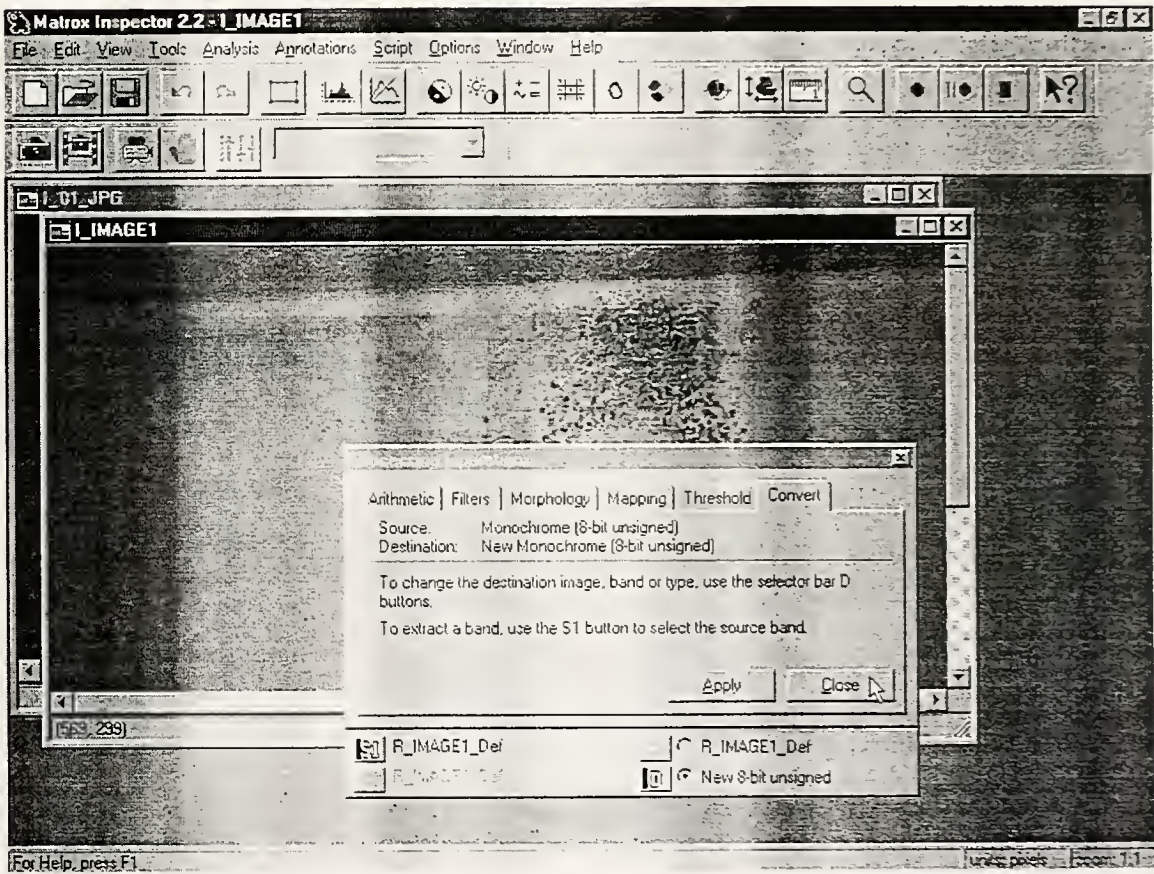
9. Select the "Mapping" item under the "Tools" header and choose "Brightness/ Contrast".

Brightness/Contrast button;
Users can also select
Brightness/Contrast by
clicking on this button
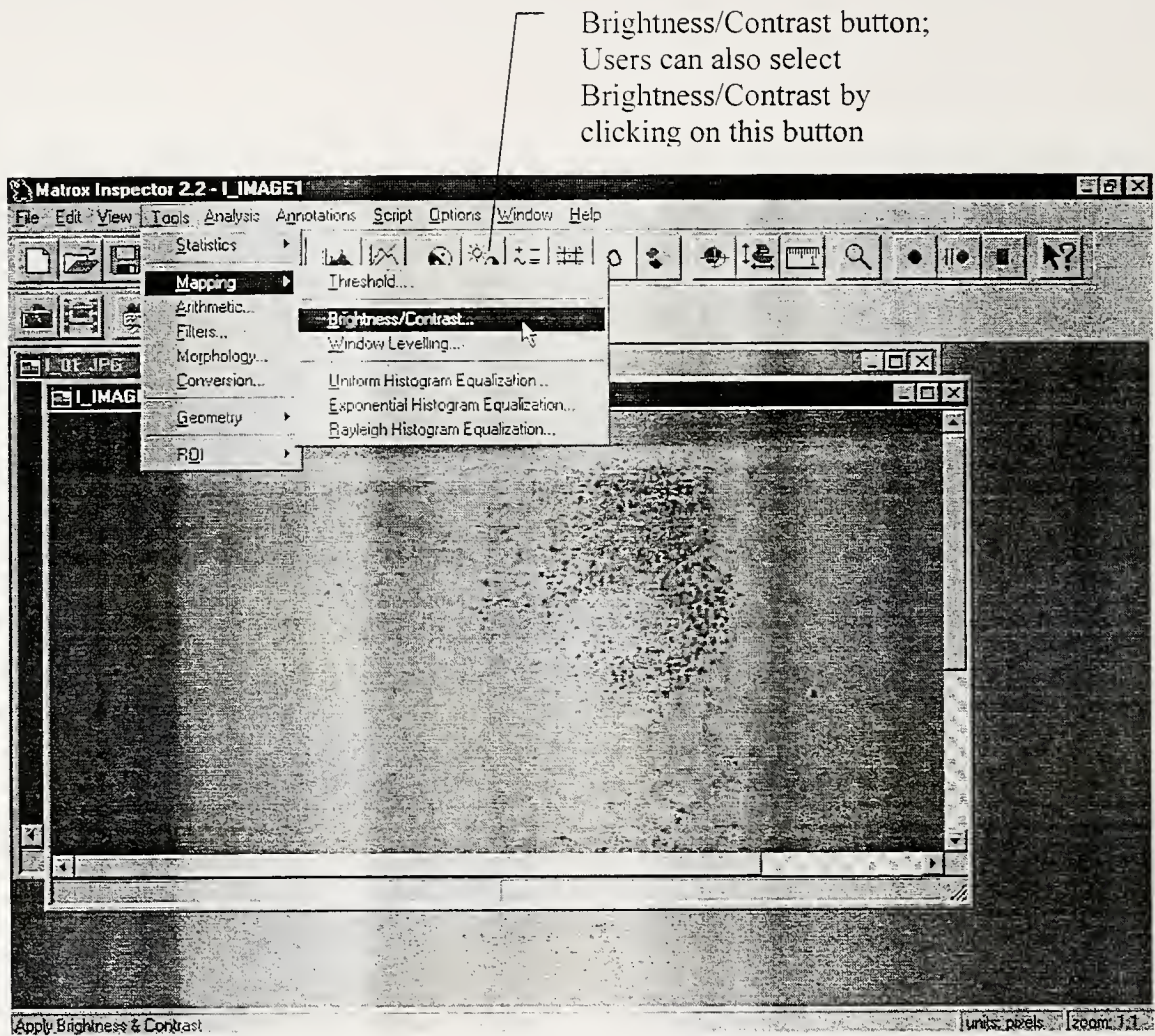
10. As the "Brightness/ Contrast" item is selected, the "Processing Operations" window will open with the 'Mapping" tab on. Use the slide bars to appropriately adjust the brightness and contrast of the grayscale image. (Usually the Brightness/Contrast need not be adjusted if the pictures are taken three feet from the objects with the flash on.)
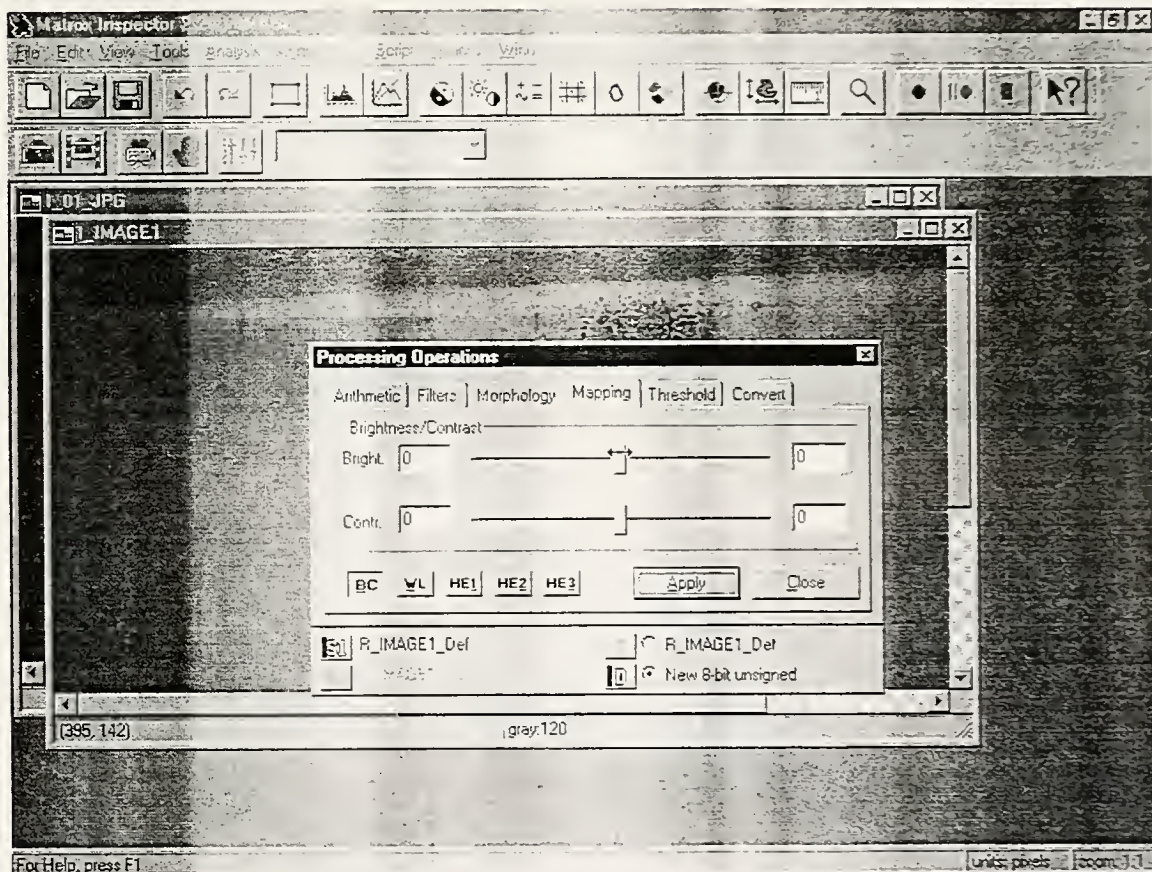
11. When the adjustment of brightness and contrast is done, click on the "Apply" button to generate a new image with the adjusted brightness and contrast.
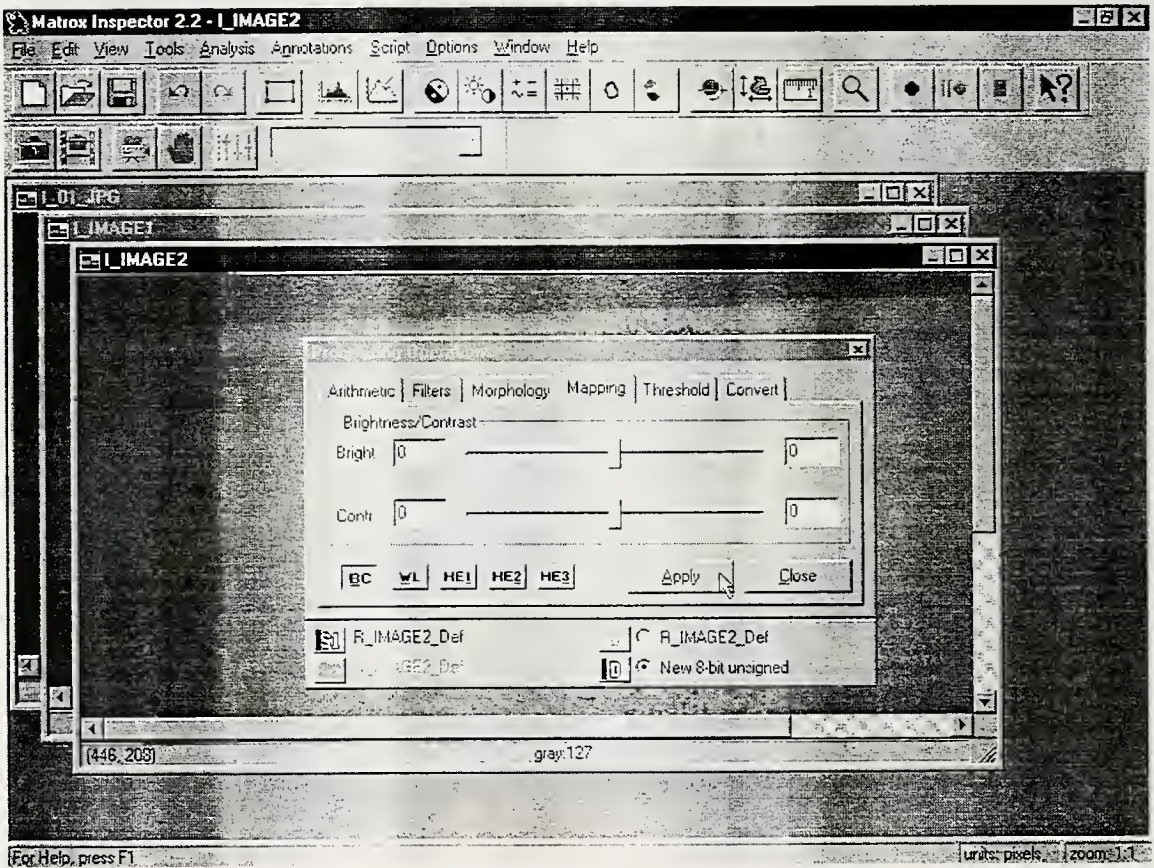
12. Select the "Threshold" tab in the "Processing Operations" window for threshold adjustment.

Threshold button;
Users can also select
Threshold by clicking
on this button
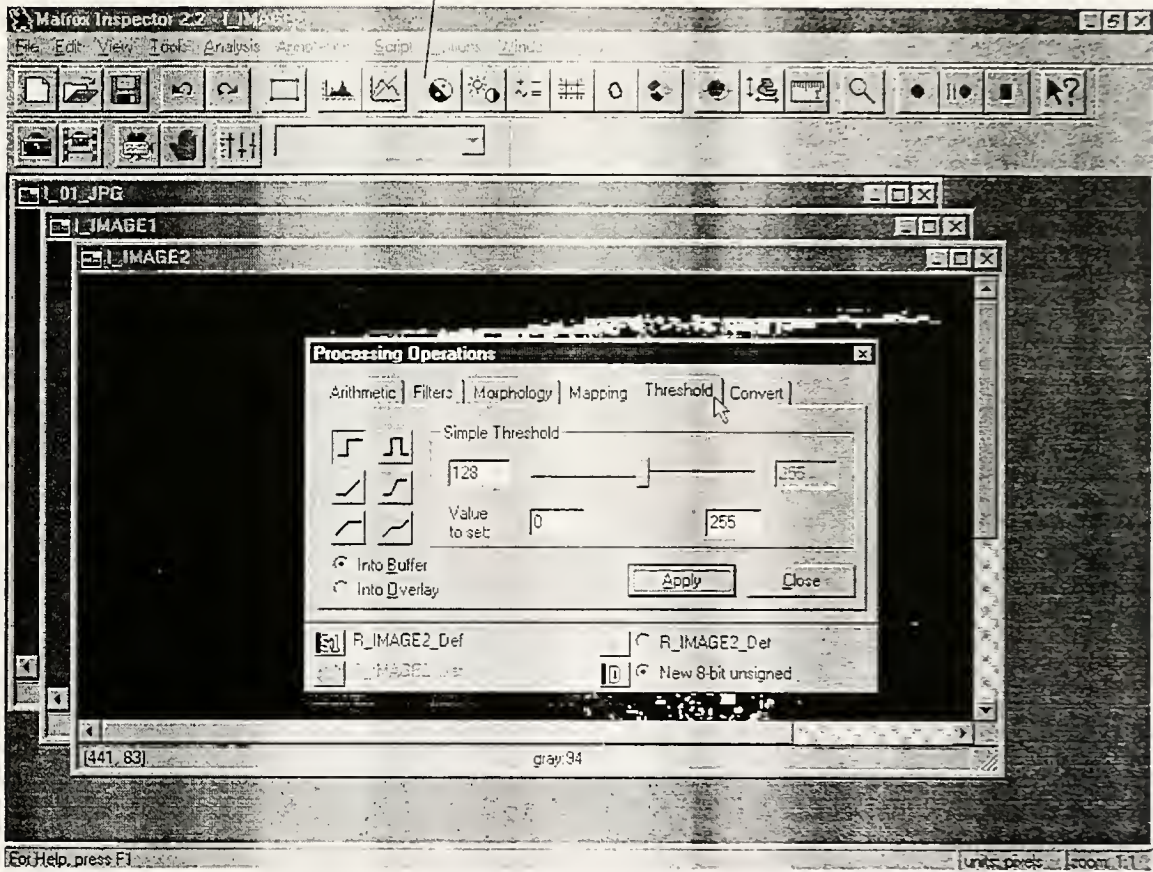
13. Use the slide bar to adjust the threshold value of Simple Threshold. (The optimum threshold value with neural network training is 106.) The image will change with the move of the slide bar. Users can also type in the threshold value directly in the blank to the left of the slide bar.
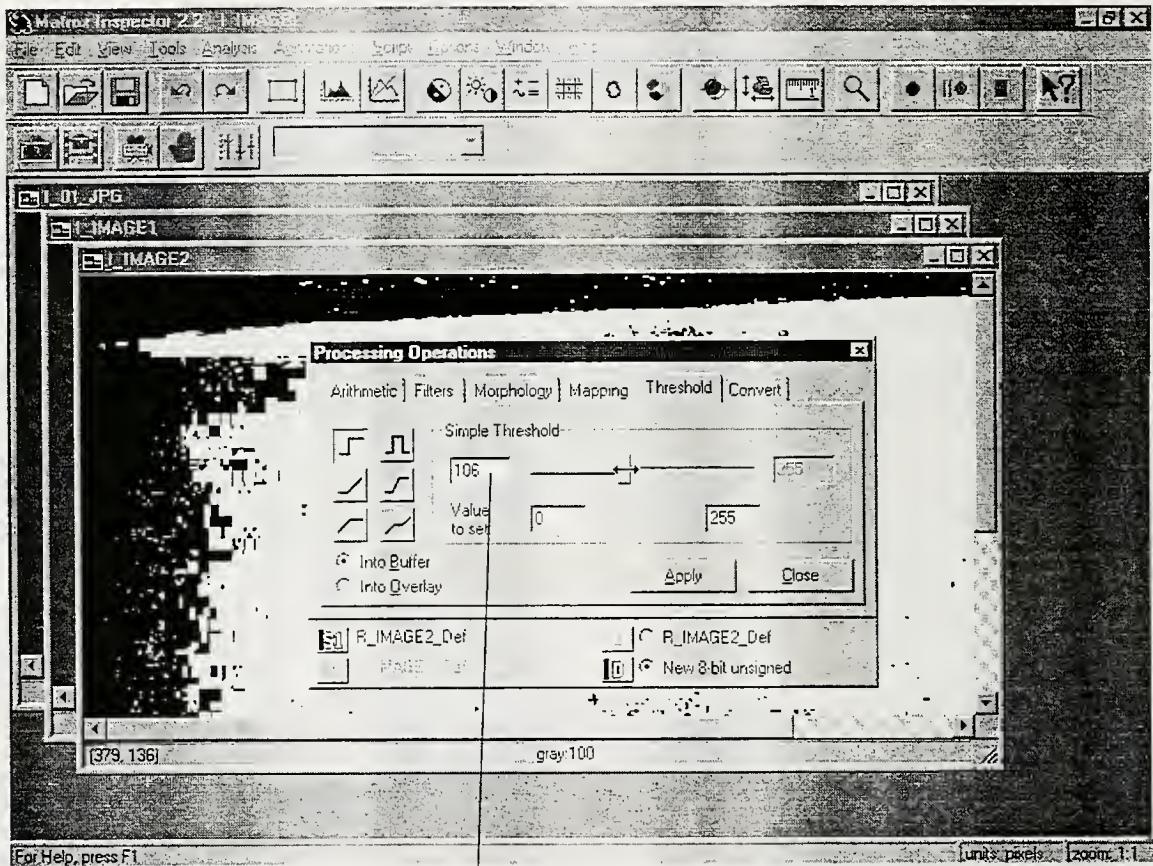
Users can also directly
input the threshold value
here

14. Click on the "Apply" button to generate a new image with the customized threshold value.
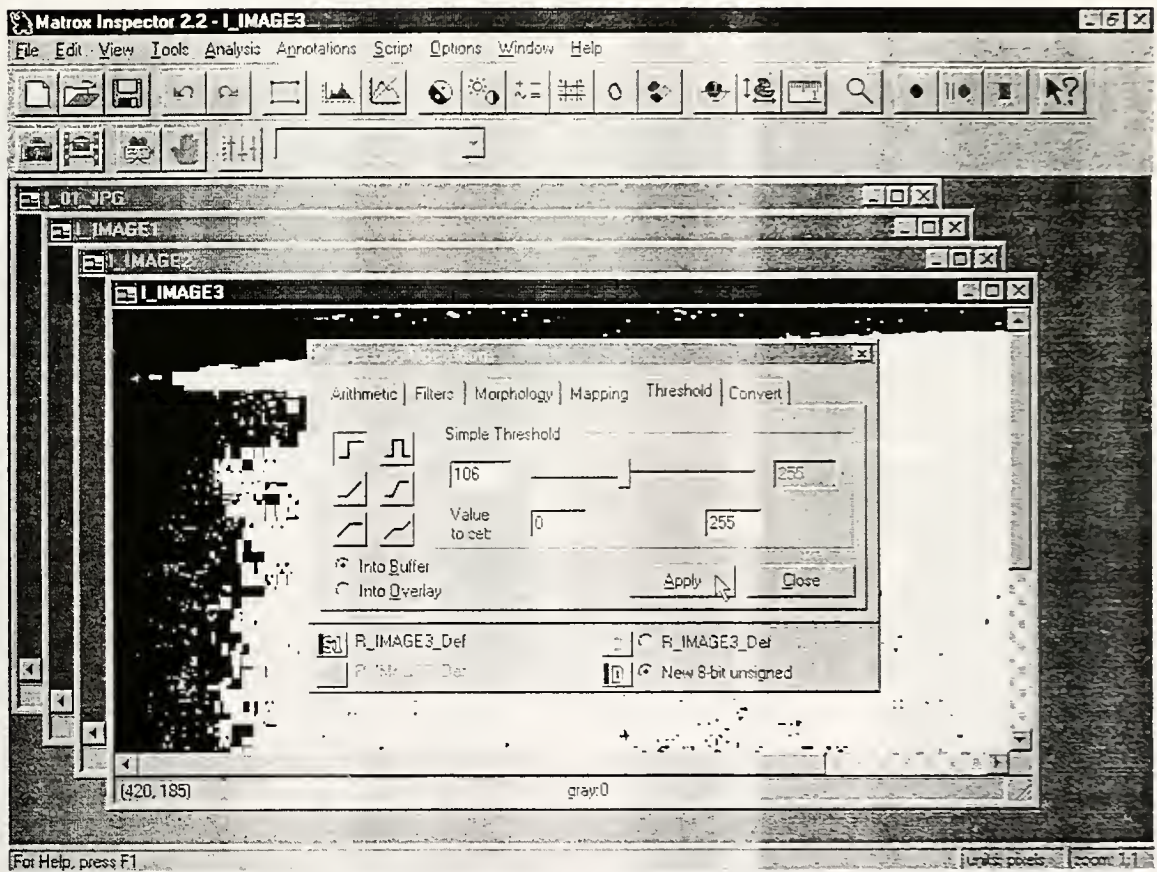
15. Click on the "Close" button to close the "Processing Operations" window.
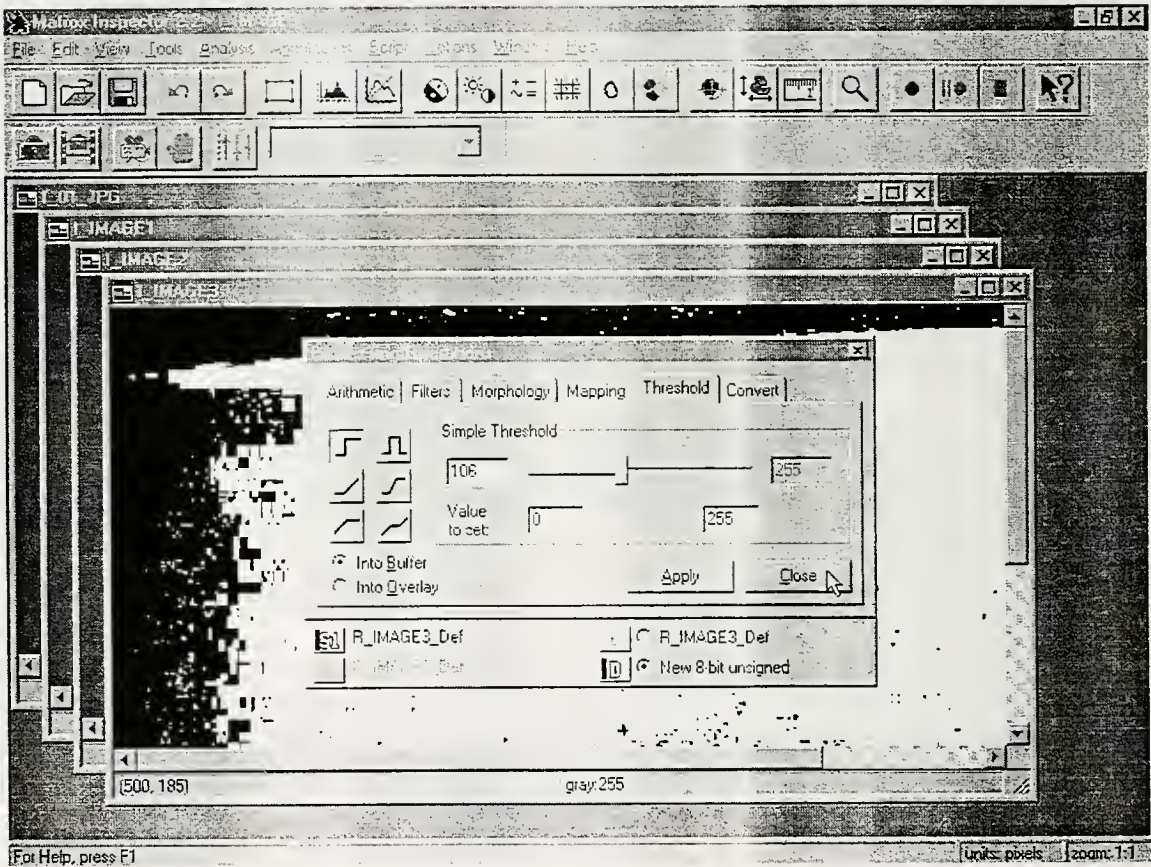
16. Select "Blob" under "Analysis" header and choose "New", and the "B_BLOBSET" window
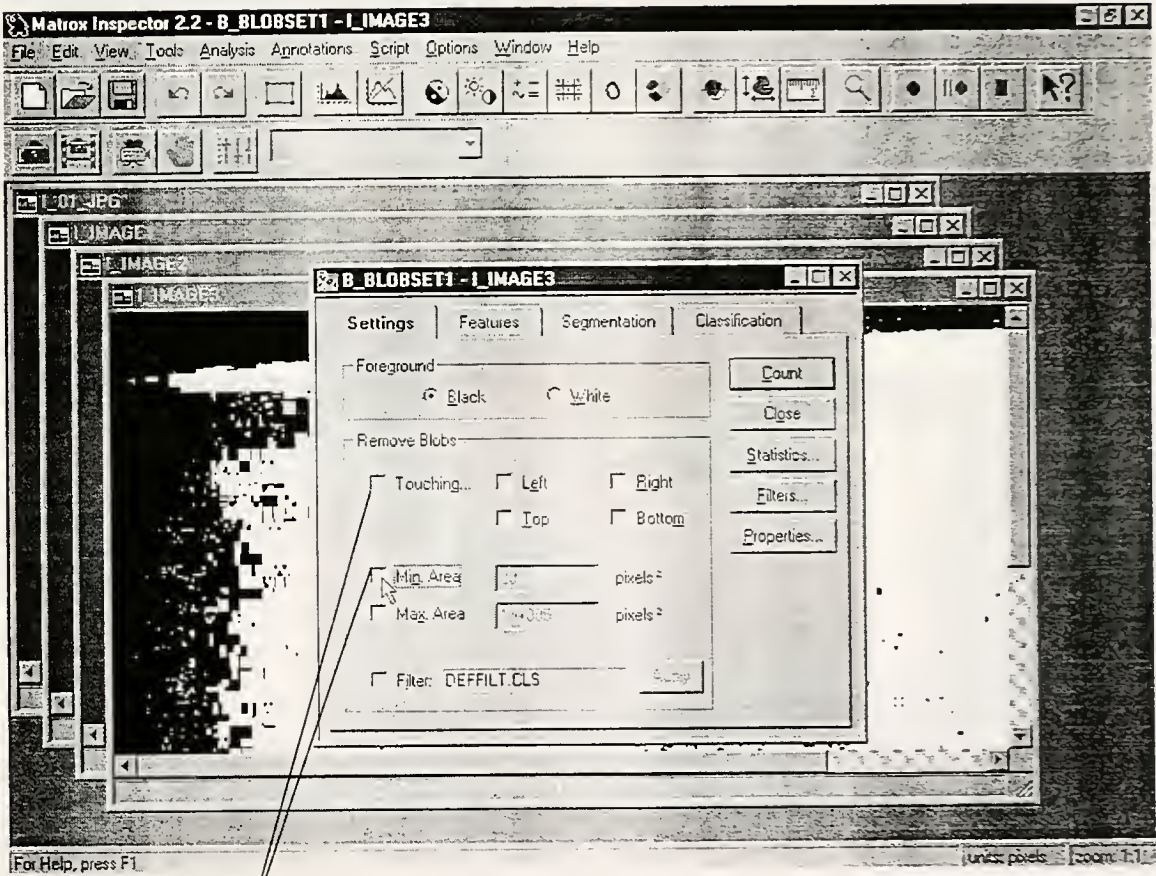
will pop up.

Blob Analysis button; By
clicking on this button the
"B_BLOBSET" window will
also show up

17. Change the "Foreground" to Black, which indicates the possible rusts in this case.
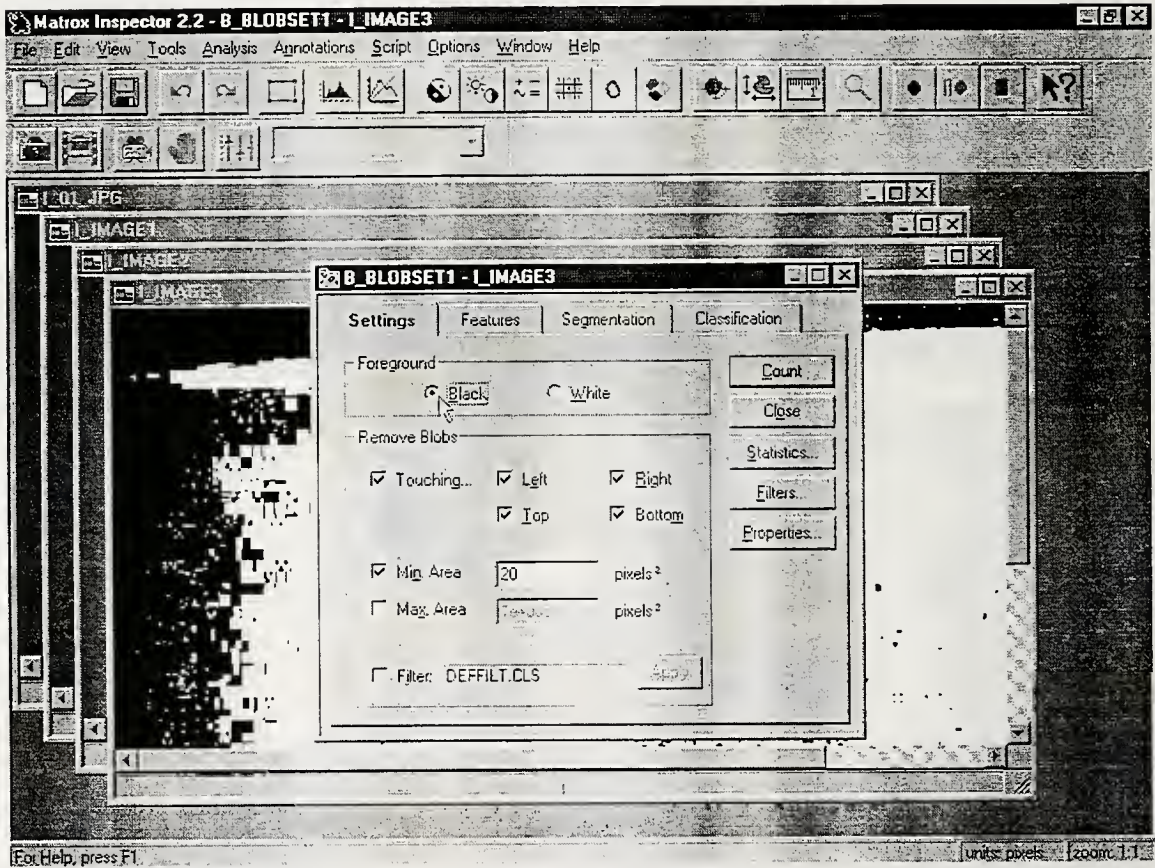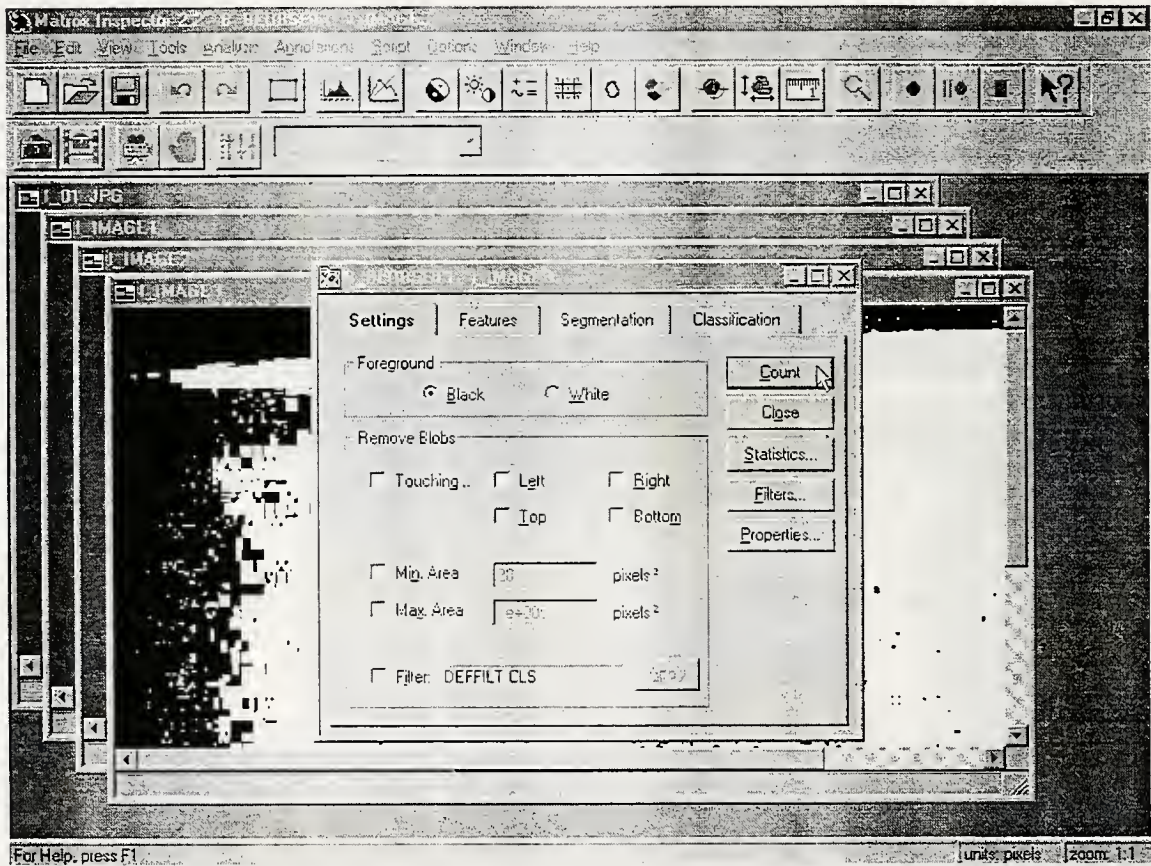
18. Remove the "Touching" and "Min. Area" selections by left-clicking on the two items.
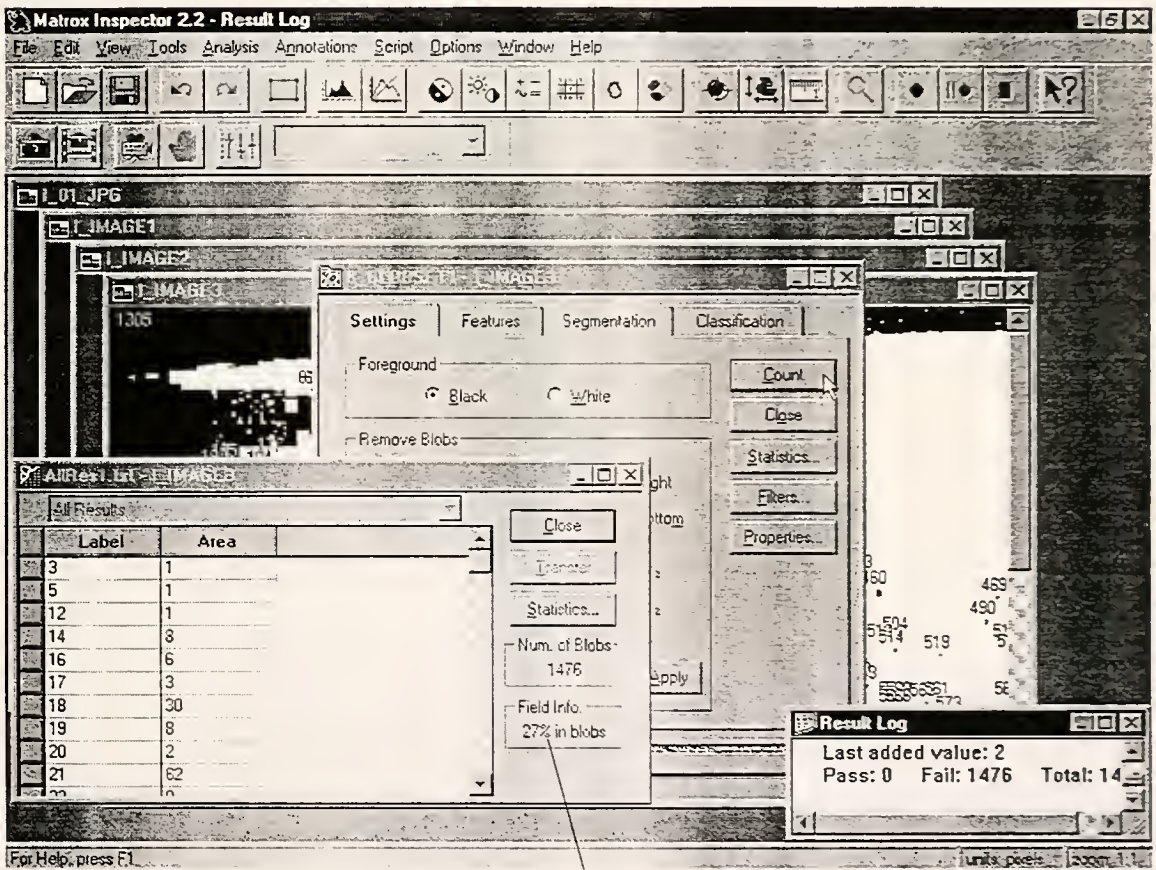

Remove these
two selections

19. Click on the "Count" button and the program will start computing the area percentage of blobs.

20. After the computation of the area percentage is done, the Label-Area List and the Result Log will be generated. In the Label-Area List, the number of blobs and the percentage in blobs are shown.

This indicates
the percentage
of rusts