

**MICROCRACK DETECTION AND NOISE REDUCTION IN
INTEGRATED CIRCUIT PACKAGES**

KOH YE SHENG

**UNIVERSITI SAINS MALAYSIA
2018**

**MICROCRACK DETECTION AND NOISE REDUCTION IN
INTEGRATED CIRCUIT PACKAGES**

by

KOH YE SHENG

**Thesis submitted in partial fulfilment of the
requirements for the degree of
Bachelor of Engineering (Mechatronics Engineering)**

JUNE 2018

ACKNOWLEDGMENTS

I would like to take this opportunity to express my heartfelt gratitude towards those who had offered to help and advised me throughout the duration of this project. First of all, I would like to offer my most sincere gratitude towards my final year project supervisor, Professor Dr. Nor Ashidi bin Mat Isa. He has given me plenty of useful advise and guided me towards the right path for me to complete my project and thesis. His approach towards my problem and suggestions on how I could solve the problem at hand has been invaluable and I appreciate his patience in guiding me which empowered me to complete this project.

I would also like to give my thanks to ViTrox Corporation Sdn. Bhd. for offering me an opportunity to take up an industrial final year project title also my industrial mentor, Mr. Kim Kek Keong for his invaluable help and assistance along the path to the completion of this project.

Last but not least, I would like to say thanks to my friends in USM and also my family members for supporting me and offering their help to me when I needed it. It is with their support that I can complete this project successfully.

TABLE OF CONTENTS

Acknowledgments	ii
Table of Contents	iii
List of Figures	vi
List of Tables	x
List of Abbreviations	xi
Abstrak	xii
Abstract	xiv
CHAPTER 1: INTRODUCTION	1
1.1 Research Background	1
1.2 Problem Statement.....	2
1.3 Objectives	3
1.4 Scope of Project.....	3
1.5 Thesis Outline.....	3
CHAPTER 2: LITERATURE REVIEW	5
2.1 Introduction	5
2.2 Image Processing	5
2.2.1 Digital Image	5
2.2.2 Contrast Enhancement.....	6
2.2.3 Noise Filter	6
2.2.4 Edge Detection	7
2.3 Crack Detection using Image Processing	8
2.3.1 Crack Detection	8
2.3.2 State-of-the-art Crack Detection Methods.....	9
2.3.2.1 Percolation Theory	9
2.3.2.2 Thermography	12
2.3.2.3 Machine Learning	13
2.3.2.4 Edge Detection Techniques.....	15
2.3.2.4.1 Anisotropic Diffusion	15
2.3.2.4.2 Probability Based Thresholding	17
2.3.2.4.3 Histogram Equalization	20
2.3.2.5 Scanning Acoustic Microscopy (SAM)	21

2.4 Summary.....	22
CHAPTER 3: METHODOLOGY	24
3.1 Introduction	24
3.2 Proposed Method 1: Probability Based Thresholding Method	24
3.2.1 Method Overview	24
3.2.1.1 Image Crack Segmentation	25
3.2.1.2 Image Denoising	27
3.3 Proposed Method 2: Histogram Equalization Method	28
3.3.1 Method Overview	28
3.3.1.1 Image Contrast Enhancement.....	29
3.3.1.2 Image Crack Segmentation	30
3.3.1.3 Image Denoising	31
3.4 Proposed Method 3: Modified Perona-Malik’s Anisotropic Diffusion Method	32
3.4.1 Method Overview	32
3.4.1.1 Image Crack Enhancement.....	33
3.4.1.2 Image Denoising	34
3.4.1.3 Image Crack Segmentation	35
3.5 Data Samples and Analysis	35
3.6 Summary.....	36
CHAPTER 4: RESULTS AND DISCUSSION	37
4.1 Introduction	37
4.2 Results of Probability Based Thresholding Method.....	37
4.2.1 Results of Image Crack Segmentation	39
4.2.2 Results of Image Denoising.....	47
4.2.3 Summary of Probability Based Thresholding Method	56
4.3 Results of Histogram Equalization Method.....	57
4.3.1 Results of Image Contrast Enhancement.....	57
4.3.2 Results of Image Crack Segmentation	69
4.3.3 Results of Image Denoising.....	80
4.3.4 Summary of Histogram Equalization Method.....	84
4.4 Results of Modified Perona-Malik Anisotropic Diffusion Method.....	84
4.4.1 Results of Image Crack Enhancement.....	84
4.4.2 Results of Image Noise Reduction	99

4.4.3 Results of Image Crack Segmentation	100
4.4.4 Summary of Modified Perona-Malik's Anisotropic Diffusion Method..	102
4.5 Performance Comparison Among Proposed Methods	102
Chapter 5	104
5.1 Conclusion	104
5.2 Suggestions for Future Work.....	105
References	106

LIST OF FIGURES

Figure 1.1: Image of crack on IC package (Retrieved from http://www.mcdry.asia/html/main1.html Copyright 2007 by ERC Co., Ltd.).....	1
Figure 2.1: Example of an image in (a) RGB colour space (b) greyscale (Dawson- Howe, 2014)	6
Figure 2.2: Example of image (a)Original image with low contrast (b)Image with improved contrast after histogram equalization (Gonzalez and Woods, 2002)	6
Figure 2.3: (a)Original image (b)Original image with influence of Gaussian noise (c)Original image with salt and pepper noise (Dawson-Howe, 2014).....	7
Figure 2.4: (a)Original greyscale image (b)Image after being applied with Roberts operator (Dawson-Howe, 2014).....	8
Figure 2.5: Flowchart for percolation process	10
Figure 2.6: Output from percolation process (a) Background (b) Crack pixels (Tomoyuki <i>et al.</i> , 2008).....	12
Figure 2.7: Digital image of bridge crack (Huayong <i>et al.</i> , 2011).....	18
Figure 2.8: Pixel density distribution with respect to position (Huayong <i>et al.</i> , 2011) .	18
Figure 2.9: Pixel density distribution with respect to position (Huayong <i>et al.</i> , 2011) .	19
Figure 2.10: Resultant binarized image using threshold $\mu - 3\sigma$ (Huayong <i>et al.</i> , 2011)	19
Figure 2.11: Imaging differences for different SAM scans (Ma <i>et al.</i> , 2007)	21
Figure 3.1: Flowchart of probability thresholding method	25
Figure 3.2: Flowchart of image crack segmentation	25
Figure 3.3: Flowchart of image denoising	27
Figure 3.4: Flowchart of histogram equalization method	29
Figure 3.5: Flowchart of image contrast enhancement	29
Figure 3.6: Flowchart of image crack segmentation	30
Figure 3.7: Flowchart for overall process	32
Figure 3.8: Flowchart for image crack enhancement.....	33
Figure 3.9: Flowchart for image noise reduction	34
Figure 3.10: Flowchart for image crack segmentation.....	35
Figure 4.1: (a)Original image 1 with crack region bounded in red box (b)Original image 2 with crack region bounded in red box (c)Original image 3 with crack region bounded in red box (d)Original image 4 with crack region bounded in red box (e)Original image 5 with crack region bounded in red box	38
Figure 4.2: Segmented image crack regions and their respective RGB channel image.	39
Figure 4.3: Image crack region histogram distribution according to RGB channels.....	40
Figure 4.4: (a)Image 1 crack region with red reference line to indicate 0 th and centre of image rows (b)Image 1 pixel density plot for blue channel (c) Image 1 pixel density plot for green channel (d) Image 1 pixel density plot for red channel.....	42

Figure 4.5: (a)Image 2 crack region with red reference line to indicate 0 th and centre of image rows (b)Image 2 pixel density plot for blue channel (c) Image 2 pixel density plot for green channel (d) Image 2 pixel density plot for red channel.....	43
Figure 4.6: (a)Image 3 crack region with red reference line to indicate 0 th and centre of image rows (b)Image 3 pixel density plot for blue channel (c) Image 3 pixel density plot for green channel (d) Image 3 pixel density plot for red channel.....	44
Figure 4.7: (a)Image 4 crack region with red reference line to indicate 0 th and centre of image rows (b)Image 4 pixel density plot for blue channel (c) Image 4 pixel density plot for green channel (d) Image 4 pixel density plot for red channel.....	45
Figure 4.8: (a)Image 5 crack region with red reference line to indicate 0 th and centre of image rows (b)Image 5 pixel density plot for blue channel (c) Image 5 pixel density plot for green channel (d) Image 5 pixel density plot for red channel.....	46
Figure 4.9: Binarization results using different thresholding values at $\mu - \sigma$, $\mu - 2\sigma$, and $\mu - 3\sigma$ for Image 1	47
Figure 4.10: Binarization results using different thresholding values at $\mu - \sigma$, $\mu - 2\sigma$, and $\mu - 3\sigma$ for Image 2	48
Figure 4.11: Binarization results using different thresholding values at $\mu - \sigma$, $\mu - 2\sigma$, and $\mu - 3\sigma$ for Image 3	49
Figure 4.12: Binarization results using different thresholding values at $\mu - \sigma$, $\mu - 2\sigma$, and $\mu - 3\sigma$ for Image 4	50
Figure 4.13: Binarization results using different thresholding values at $\mu - \sigma$, $\mu - 2\sigma$, and $\mu - 3\sigma$ for Image 5	51
Figure 4.14: Resultant image after morphology closing operation	54
Figure 4.15: Images of crack for (a)Original Image 1 (b)Resultant Image 1 after morphology closing (c)Original Image 2 (d)Resultant Image 2 after morphology closing (e)Original Image 4 (f)Resultant Image 4 after morphology closing (g)Original Image 5 (h)Resultant Image 5 after morphology closing.....	56
Figure 4.16: Separation of original images into RGB channels after masking and cropping operations	58
Figure 4.17: Image 1 RGB channels after conventional histogram equalization and CLAHE.....	59
Figure 4.18: Image 2 RGB channels after conventional histogram equalization and CLAHE.....	60
Figure 4.19: Image 3 RGB channels after conventional histogram equalization and CLAHE.....	61
Figure 4.20: Image 1 RGB channels after conventional histogram equalization and CLAHE.....	62
Figure 4.21: Image 1 RGB channels after conventional histogram equalization and CLAHE.....	62
Figure 4.22: Image 1 histogram distribution after conventional histogram equalization and CLAHE.....	63
Figure 4.23: Image 2 histogram distribution after conventional histogram equalization and CLAHE.....	64

Figure 4.24: Image 3 histogram distribution after conventional histogram equalization and CLAHE.....	65
Figure 4.25: Image 4 histogram distribution after conventional histogram equalization and CLAHE.....	66
Figure 4.26: Image 5 histogram distribution after conventional histogram equalization and CLAHE.....	67
Figure 4.27: Images of IC packages' pads circled in red.....	68
Figure 4.28: Grey level differences for Image 1.....	69
Figure 4.29: Grey level differences for Image 2.....	70
Figure 4.30: Grey level differences for Image 3.....	70
Figure 4.31: Grey level differences for Image 4.....	71
Figure 4.32: Grey level differences for Image 5.....	71
Figure 4.33: Grey level differences histogram distribution for Image 1.....	72
Figure 4.34: Grey level differences histogram distribution for Image 2.....	73
Figure 4.35: Grey level differences histogram distribution for Image 3.....	74
Figure 4.36: Grey level differences histogram distribution for Image 4.....	75
Figure 4.37: Grey level differences histogram distribution for Image 5.....	76
Figure 4.38: Binarization of RGB channels for Image 1.....	77
Figure 4.39: Binarization of RGB channels for Image 2.....	77
Figure 4.40: Binarization of RGB channels for Image 3.....	78
Figure 4.41: Binarization of RGB channels for Image 4.....	78
Figure 4.42: Binarization of RGB channels for Image 5.....	79
Figure 4.43: Resultant image from bitwise operation for image 1.....	80
Figure 4.44: Resultant image from bitwise operation for image 2.....	81
Figure 4.45: Resultant image from bitwise operation for image 3.....	82
Figure 4.46: Resultant image from bitwise operation for image 4.....	83
Figure 4.47: Resultant image from bitwise operation for image 5.....	83
Figure 4.48: Separation of original images into RGB channels.....	85
Figure 4.49: Resultant image from before and after anisotropic diffusion (AD) for each RGB channel image.....	87
Figure 4.50: Resultant images from different parameters of K and iteration in anisotropic diffusion for Image 1.....	89
Figure 4.51: Resultant images from different parameters of K and iteration in anisotropic diffusion for Image 2.....	90
Figure 4.52: Resultant images from different parameters of K and iteration in anisotropic diffusion for Image 3.....	91
Figure 4.53: Resultant images from different parameters of K and iteration in anisotropic diffusion for Image 4.....	92
Figure 4.54: Resultant images from different parameters of K and iteration in anisotropic diffusion for Image 5.....	93
Figure 4.55: Resultant image from subtraction of anisotropic diffused image with original image for each RGB channel image.....	94
Figure 4.56: Resultant image from summation of grey level differences images from previous step.....	96

Figure 4.57: Histogram distribution from summed images (a)Image 1 (b)Image 2 (c)Image 3 (d)Image 4 (e)Image 5	97
Figure 4.58: Resultant image from conventional binarization.....	99
Figure 4.59: Resultant image after morphology opening and median filtering	100
Figure 4.60: Resultant image after filtering of blobs by area.....	101

LIST OF TABLES

Table 2.1: Comparison of advantages and disadvantages of each method	22
Table 4.1: Comparison of Performance Among Proposed Methods.....	102

LIST OF ABBREVIATIONS

AD	- Anisotropic Diffusion
AFM	- Atomic Force Microscopy
CLAHE	- Contrast Limited Adaptive Histogram Equalization
HE	- Histogram Equalization
IC	- Integrated Circuit
SAM	- Scanning Acoustic Microscopy

PENGESANAN RETAKAN MIKRO DAN PENGURANGAN HINGAR DALAM PAKEJ LITAR BERSEPADU

ABSTRAK

Peningkatan penggunaan produk elektronik pada kebelakangan ini telah mengakibatkan peningkatan pengeluaran litar bersepadu (IC) untuk memenuhi keperluan pengguna. Oleh itu, ia adalah amat penting bahawa setiap IC diperiksa untuk kecacatan yang mampu menjejaskan tahap kualitinya. Ini memastikan bahawa tiada IC yang cacat digunakan dalam pembuatan produk elektronik yang boleh menjejaskan prestasi dan jangka hayat produk. Antara kecacatan yang terbabit ialah retakan mikro pada pakej IC. Pemprosesan imej digunakan untuk mengesan kewujudan retakan mikro pada pakej IC dan kaedah yang digunakan untuk mencapai matlamat ini ialah menerusi lingkaran dengan pelbagai inti dengan tatarajah yang berlainan. Kaedah ini memakan masa yang banyak kerana banyak tatarajah yang perlu ditala dan ia juga mudah terjejas dengan gangguan bunyi dari imej yang mengurangkan ketepatan pengesanan retakan mikro. Algoritma yang lebih baik diperlukan untuk meningkatkan prestasi pengesanan dari segi masa dan ketepatan. Tiga algoritma telah diuji dan dinilai dari segi pengesanan retakan mikro dan pengurangan gangguan bunyi dalam imej iaitu pengembangan berdasarkan kemungkinan, penyamaan histogram, dan resapan tak isotropi Perona-Malik yang diubah suai. Untuk algoritma yang pertama, kaedah pengembangan berdasarkan kemungkinan dibahagikan kepada dua bahagian iaitu (i) bahagian peruasan retakan dalam imej di mana bahagian retakan dalam imej akan dianalisa untuk mendapatkan nilai yang sesuai untuk pengembangan, dan (ii) bahagian pengurangan hingar dalam imej di mana imej tersebut akan dikenakan morfologi penutupan. Bagi algoritma yang kedua, kaedah penyamaan histogram, ia dibahagikan kepada tiga bahagian iaitu (i) peningkatan kontras imej menerusi penyamaan histogram, (ii) bahagian peruasan retakan dalam imej mengambil nilai piksel imej yang telah dikenakan penyamaan histogram dan ditolakkan dengan nilai piksel imej sebelum proses penyamaan histogram bagi setiap saluran dan gabungkan imej dari setiap saluran yang terhasil untuk menjadi satu imej baru, dan (iii) bahagian pengurangan hingar dalam imej menggunakan morfologi pembukaan. Dalam algoritma yang ketiga, kaedah resapan tak isotropi Perona-Malik yang diubah suai dibahagikan kepada tiga bahagian, iaitu (i) bahagian menonjolkan retakan dalam imej yang akan memisahkan imej kepada saluran merah, biru, dan hijau dan meningkatkan ciri-ciri

retakan menggunakan resapan tak isotropi Perona-Malik yang diubah suai, (ii) peruasan imej retakan yang mengambil nilai piksel imej yang telah diresap dan ditolakkan dengan nilai piksel imej yang belum diresap bagi setiap saluran dan gabungkan imej dari setiap saluran yang terhasil untuk menjadi satu imej baru, dan (iii) bahagian pengurangan hingar dalam imej di mana imej tersebut akan dikenakan morfologi pembukaan dan seterusnya penapisan median. Imej yang diproses menerusi kaedah resapan tak isotropi Perona-Malik yang diubah suai tersebut mempunyai hingar yang lebih kurang daripada kaedah pengambangan berdasarkan kemungkinan dan kaedah penyamaan histogram. Kaedah tersebut juga telah mengesan retakan yang wujud dalam tiga sampel daripada lima sampel yang diuji. Kaedah resapan tak isotropi Perona-Malik yang diubah suai telah dibuktikan mempunyai prestasi yang lebih baik secara relatifnya jika dibandingkan dengan dua lagi kaedah yang diuji.

MICROCRACK DETECTION AND NOISE REDUCTION IN INTEGRATED CIRCUIT PACKAGES

ABSTRACT

The rise in consumption of electronic products in the recent years have subsequently led to an increase in manufacturing of integrated circuits (ICs) to meet consumers' demands. Thus, it is vital that each IC is inspected for defects that compromises its quality and usability. This ensures that no defective ICs are used in conjunction with the manufacturing of electronic products which may severely impact the end product's performance and lifespan. One of the common defects is microcrack on the IC's package. Image processing is used to detect the presence of microcracks on the IC and the method currently employed to achieve this is by convolution with multiple kernels with different configurations. However, this method is time consuming due to the multiple configurations needed to be tuned and is also susceptible to image noise which lowers the accuracy of the detected microcracks. Therefore, a better algorithm is desired to improve the detection performance in terms of time and accuracy. Three algorithms are tested and evaluated in terms of microcrack detection and noise reduction which are probability based thresholding, histogram equalization, and modified Perona-Malik's anisotropic diffusion methods. The first algorithm, probability based thresholding method consists of two stages, (i) image crack segmentation where the crack regions are analysed to obtain a suitable thresholding value, and (ii) image denoising where morphological closing is performed on the image. For the second algorithm, histogram equalization method has three stages, (i) image contrast enhancement through histogram equalization, (ii) image crack segmentation which subtracts the histogram equalized image with the image before histogram equalization process before merging the images using bitwise operation, and (iii) image denoising using morphological opening. The third algorithm, modified Perona-Malik's anisotropic diffusion method consists of three stages, (i) image crack enhancement which separates the image into its red, green, and blue channels and enhances the crack features using modified Perona-Malik's anisotropic diffusion, (ii) image crack segmentation which subtracts the diffused image with the pre-diffused image before summing the grey values of the images together, and (iii) image denoising using morphological opening and median filter. Images processed using modified Perona-Malik's anisotropic diffusion method produces images

with less noise compared to probability based thresholding method and histogram equalization method. The method has detected cracks present in three samples out of the five samples tested. The modified Perona-Malik's anisotropic diffusion method is thus proven to produce relatively better performance compared to the other tested methods.

CHAPTER 1

INTRODUCTION

1.1 Research Background

As more electronic based applications require the usage of ICs, the size of ICs have subsequently increased. The increase in size requires more space but the package dimensions that houses the ICs have been maintained or even reduced to become smaller and thinner. Widespread usage of ICs have also resulted in an expectation of lower manufacturing cost for the chips. To fulfil the needs for a lower cost, plastic packaging is used to house the ICs which are inexpensive packaging solutions (Hong and Lai, 2011). However, the restriction of IC package dimensions have resulted in an increase in mechanical stresses and may cause cracking in the development stage and popcorn effect which stresses the plastic packaging during fabrication resulting in external package cracks (Hong and Lai, 2011). These cracked packages will be rejected as the compromised mechanical integrity of an IC package may cause it to fail prematurely or immediately with no indication of degradation in electrical performance (Ma *et al.*, 2007).

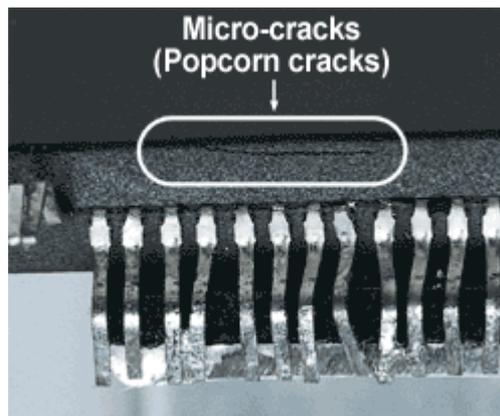


Figure 1.1: Image of crack on IC package (Retrieved from <http://www.mcdry.asia/html/main1.html> Copyright 2007 by ERC Co., Ltd.)

Figure 1.1 illustrates an example of an IC package with a crack on the plastic package. These cracks are undesirable and a method that utilizes a software approach to detect it is desired.

1.2 Problem Statement

In general, a method is needed to detect the presence of microcracks on IC packages. There have been several researches done in microcrack detections with some employing different image acquisition techniques only while some also use image acquisition with image processing and others are non-imaging based (Abdelhamid *et al.*, 2014; Trautmann *et al.*, 2012). Imaging based image acquisition produces an image of the surface being inspected and an image that is sharp and clear is desired to ensure that the detection of microcracks (if any) is accurate. Optical transmission which involves the usage of a camera and a light source (Abdelhamid *et al.*, 2014; Ke *et al.*, 2010) and luminescence imaging which captures light emissions from nonthermal energy sources are examples of imaging based image acquisition (Abdelhamid *et al.*, 2014; Teo and Abdullah, 2016). Non-imaging based crack detections involves mechanical bending such as twist-test to detect the presence of cracks (Trautmann *et al.*, 2012) or via ultrasonic inspections (Abdelhamid *et al.*, 2014; Ma *et al.*, 2007). However, all images used in this project are captured via optical transmission. Non-imaging methods become unsuitable as it requires specialized hardware to detect the cracks.

After image acquisition, image processing is needed to obtain the microcrack features. Some of the methods that can be used to achieve this are the usage of percolation-based image processing (Yamaguchi *et al.*, 2008), thermography (Cheng and Tian, 2011; El-amiri *et al.*, 2016; Liu *et al.*, 2013; Noethen *et al.*, 2010; Zhao *et al.*, 2017), machine learning (Choudhary and Dey, 2012; Gaith *et al.*, 2015; Kim *et al.*, 2001; Maazi *et al.*, 2008; Saar and Talvik, 2010), and edge detection techniques that utilizes histogram equalization (Ke *et al.*, 2010; Parveen and Sathik, 2009; Qiu *et al.*, 2012; Zhuang *et al.*, 2004), anisotropic diffusion filters (Anwar and Abdullah, 2014; Oliveira and Correia, 2010; Tang and Gu, 2013; Tsai *et al.*, 2010), and probability based thresholding (Wu *et al.*, 2011). However, these researches are based on cracks on solar cells, silicon wafer cells, concrete surfaces such as walls and bridges, and pavements. Researches on implementation of image processing methods for crack detection of IC packages is limited.

1.3 Objectives

The objectives of this research are as follows:

- 1) To investigate a suitable method that reduces noise in acquired image without losing significant features for microcrack detection
- 2) To analyse a method that can detect microcracks on IC packages
- 3) To evaluate the performance of tested methods in microcrack detection

1.4 Scope of Project

This project focuses on the image processing application in crack detection. It will focus on the image processing portions with crack segmentation and noise reduction as the main concern.

This research will be conducted using OpenCV imaging library version 2.4.13.4 and will be developed in a C++ environment using Microsoft Visual Studio 2017. The imaging library contains a variety image processing functions that will be useful in the image processing areas of the project.

Image acquisition is not a concern in this project and all images used are provided by ViTrox Corporation Sdn. Bhd. The provided images are coloured images with the image file type of TIFF (.tiff).

1.5 Thesis Outline

The thesis is arranged into five chapters in the following order: Introduction, Literature Review, Methodology, Results and Discussion, and Conclusion. In Chapter 1, the research background, problem statement, objectives, and scope of the project are defined and explained.

In Chapter 2, a literature review of digital image processing and crack detection methods is presented. The concepts of digital image processing on contrast enhancement, noise filtering, and edge detection are reviewed in this chapter. Five existing methods on crack detection are discussed and a comparison of their advantages and disadvantages is evaluated.

In Chapter 3, the methodology of this project to achieve the objectives from Chapter 2 is presented. Three methods are tested and the overall flow of each method is presented. An explanation about the implementation of the techniques will be provided. Explanation about the data samples and analysis of the project are presented as well.

In Chapter 4, the results from the methods presented in Chapter 3 are presented and evaluated. Three different sets of results from three different methods are presented. A discussion of their performance and effectiveness will be included as well.

In Chapter 5, the conclusion of the work presented in this project is presented along with suggestion for future works.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Image processing is a method that analyses and processes image to extract features in that image. It can also be used to enhance an image in terms of contrast or noise reduction. There are many applications to image processing and in this project, the focus will be towards crack detection in IC packages.

In Section 2.2, the focus will be on image processing where the concepts of digital image, contrast enhancement, noise filter, and edge detection will be presented. This is followed by a review of applications of crack detections using image processing in Section 2.3 where five different methods in crack detections are reviewed. A summary of this chapter is presented in Section 2.4.

2.2 Image Processing

Image processing is used to manipulate an image for the purpose of enhancing an image or extracting certain features from an image. In general, these operations are performed on digital images.

2.2.1 Digital Image

An image is visualized by a computer as a matrix of numbers where each elements of the matrix corresponds to the pixel intensity or values of the image at the spatial domain. The image can be expressed in several different formats namely the red, green, blue (RGB), greyscale, and binary format (i.e. the common format), and hue, saturation, and lightness/value (HSL/HSV) colour spaces (i.e. the uncommon format). A typical RGB image is visualized as a 3-dimensional matrix where the red, green, and blue colour spaces have their own $n \times m$ matrices where n and m are the size of the image in terms of length and height. The elements contained in these matrices will correspond to the value of red, green, and blue in the image and the combination of these three colours produces the coloured images. A conversion of coloured images to greyscale will convert the image into a 2-dimensional $n \times m$ matrix instead. An example of a RGB colour image and its corresponding greyscale image is shown in Figure 2.1.

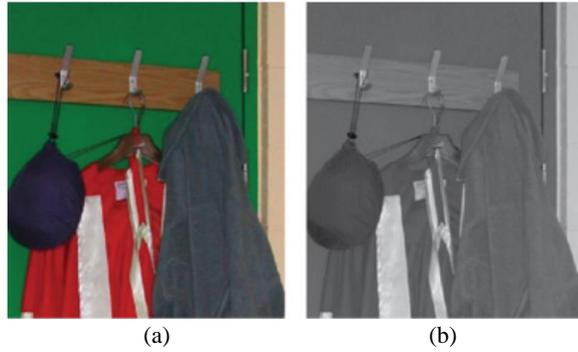


Figure 2.1: Example of an image in (a) RGB colour space (b) greyscale (Dawson-Howe, 2014)

2.2.2 Contrast Enhancement

Contrast enhancement aims to improve the image's quality by enhancing the features of an image in terms of brightness. An example of an image with poor contrast and improved contrast is as shown in Figure 2.2.

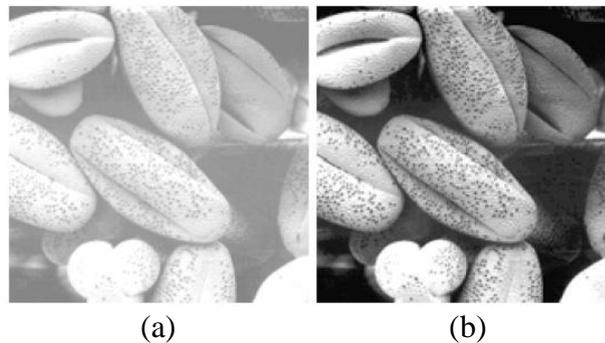


Figure 2.2: Example of image (a)Original image with low contrast (b)Image with improved contrast after histogram equalization (Gonzalez and Woods, 2002)

Two types of contrast enhancement are explored in this project which are histogram equalization and contrast limiting adaptive histogram equalization (CLAHE).

2.2.3 Noise Filter

Noise filtering is done to remove pixels from the image that have obscure useful information in an image. These noise pixels typically occur due to sensor noise or electronic noise. Some examples on the types of noise are shown in Figure 2.3.

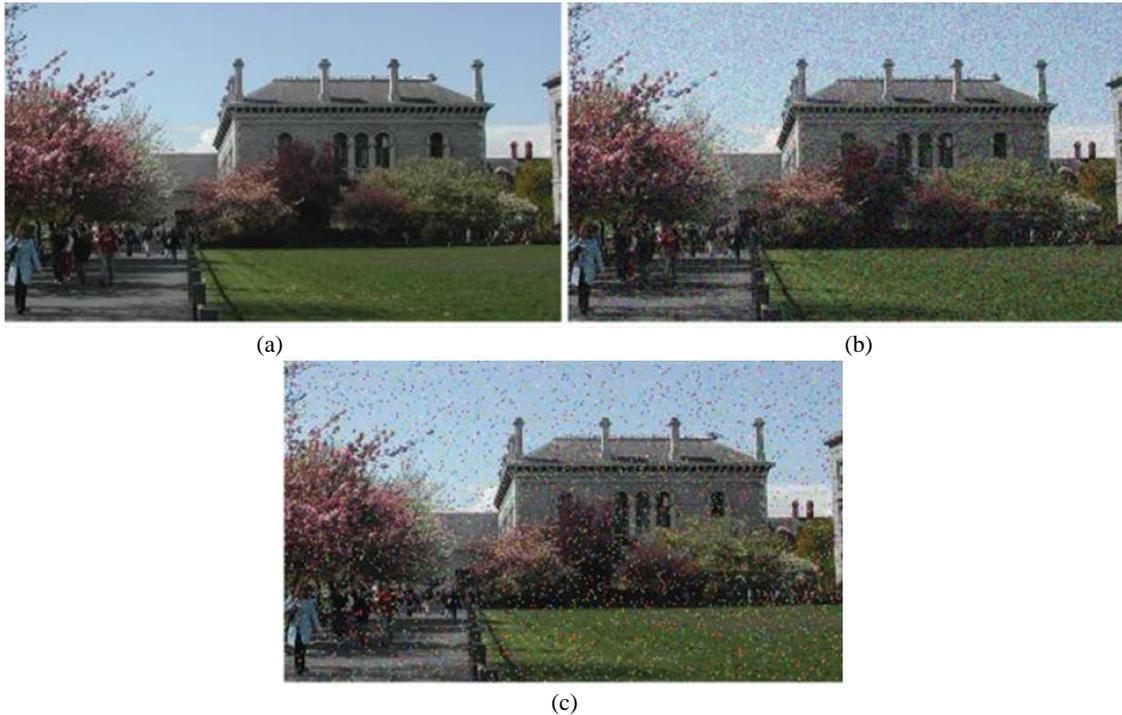


Figure 2.3: (a)Original image (b)Original image with influence of Gaussian noise (c)Original image with salt and pepper noise (Dawson-Howe, 2014)

Noise are typically removed using noise filters. Some examples of noise filters are median filter, Gaussian filter, and mean filter. These filters are masks of $n \times n$ sizes which are applied onto the image which modifies the value of pixel at the mask's centre.

2.2.4 Edge Detection

Edge detection is used to identify regions where edges appear. This usually occurs in regions where there are sharp changes in gradient or regions where image brightness changes abruptly. Edge detection is typically classified into two categories which are first derivative and second derivative (Dawson-Howe, 2014). For first derivatives, it calculates a local maximum at the edge which is where rate of change is the highest while the second derivative results in a value that changes from positive to negative at the edge (Dawson-Howe, 2014).

An example of a first derivative edge detection is Roberts operator as shown in Figure 2.4. The edges present in Figure 2.4 (a) are detected by Roberts operator and this is useful to obtain features in an image that occurs at places where there are changes in gradient.



(a) (b)
Figure 2.4: (a)Original greyscale image (b)Image after being applied with Roberts operator (Dawson-Howe, 2014)

2.3 Crack Detection using Image Processing

Image processing in crack detections are widely applied in crack detections for roads, pavements, bridges and walls. Some of the methods used by these detection techniques include anisotropic diffusion, histogram equalizations, probability based distribution, and percolation methods. However, none of these methods employ crack detection on IC packages which have smaller cracks in general.

The current method being used to obtain the microcrack features involves convolving the image with several different configured kernels to detect the microcracks. However, this method is slow and consumes a lot of time due to the multiple configurations needed to be tuned for the kernels. Another drawback of this method is that it is more susceptible to image noise thus causing it to be inaccurate as it may detect image noise as cracks.¹

2.3.1 Crack Detection

For crack detection in general, there are three approaches that can be used to detect cracks, namely image processing, the image acquisition, and integration of both methods. Typically image processing methods are purely software based where algorithms are used to enhance or extract the desired features. On image acquisition, this typically deals with hardware where different imaging techniques as stated in the paper (Abdelhamid *et al.*, 2014) may be used such as photoluminescence or electroluminescence imaging that

¹ The information pertaining to the convolution method to obtain crack features was provided by ViTrox Corporation Berhad's Machine Vision Systems - Standard (MVS-S) Department

obtains images from photon emissions. Another image acquisition method would be thermography which images the material under test with heat profiles (Cheng and Tian, 2011; El-amiri et al., 2016; Liu et al., 2013; Noethen et al., 2010). SAM is also an image acquisition method which uses ultrasonic waves to quantify defects and anomalies (Ma et al., 2007; Sirivathanant et al., 2007). A mixture of both methods would process the acquired image further to interpret the detected features.

2.3.2 State-of-the-art Crack Detection Methods

Several methods have been used on the aspect of crack detection. The following sections explain five methods that have been conducted to detect cracks using image processing based solutions, hardware solutions, and an integration of both.

2.3.2.1 Percolation Theory

Tomoyuki and Shuji (2006), and Tomoyuki *et al.* (2008) used a percolation model to detect cracks on concrete structures. The percolation model is modelled based on liquid permeation and can describe the spread of epidemics, forest fires, and distribution of oil in oil fields (Stauffer and Aharony, 1994; Yamaguchi *et al.*, 2008). In crack detection, this method attempts to reconstruct the crack features in the image from a focal pixel in the image. One of the methods proposed by them is seeding the image to obtain arbitrary pixels as focal pixels and percolate until a termination condition is reached in which the process repeats itself with another arbitrary pixel not bounded within the previously percolated regions (Yamaguchi and Hashimoto, 2006). Another method employs fast template matching and active search which serves to reduce computational burden by determining whether the focal pixel is a background pixel or crack pixel before beginning percolation (Yamaguchi *et al.*, 2008). A similar approach was used by Praveen and Reshmi S (2014) to detect cracks on concrete wall surfaces while Yasuhiro *et al.* (2017) used it to detect cracks on road pavements. On the other hand, Bo *et al.* (2013) uses a percolation mask which utilizes canny edge detector to determine regions of pixels to be percolated to reduce computational burden which is employed in solar cells crack detection. All of the approaches (Feng *et al.*, 2013; Kawasaki *et al.*, 2017; Sekhar and Bhooshan, 2014) implement the same percolation process which is summarized in Figure 2.5.

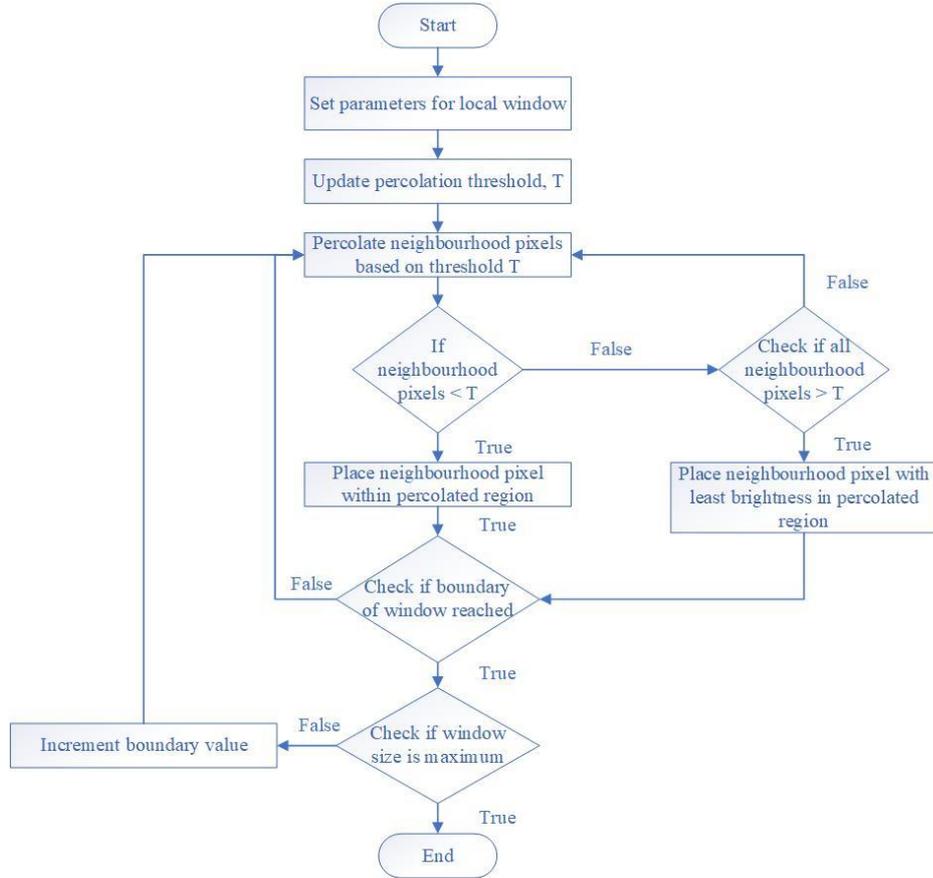


Figure 2.5: Flowchart for percolation process

2.3.2.1.1 Set parameters for local window

The percolation process first defines a local window of size $N \times N$ with a maximum window size after percolation of $M \times M$. (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006) The pixel located at the center of the $N \times N$ window is known as the focal pixel or initial pixel, p_s . p_s belongs to the percolated region defined as D_p . (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006)

2.3.2.1.2 Update percolation threshold, T

Initially, T will be set to the brightness value, I of p_s . (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006) During the subsequent iterations of percolation, the threshold will be updated according to Equation (2.1). (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006)

$$T = \max\left(\max_{p \in D_p}(I(p)), T\right) + w \quad (2.1)$$

where

w is acceleration parameter to accelerate percolation.

2.3.2.1.3 Percolate neighbourhood pixels based on threshold T

From the position of pixel p_s , the brightness for eight neighbouring pixels are evaluated to determine whether they are within the threshold or exceeding it. Any neighbouring pixels with brightness value less than threshold will be included into the region D_p . If no pixels have a brightness value below the threshold, the pixel with the lowest brightness value will be included instead (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006).

2.3.2.1.4 Check if window size is maximum

During each iteration, the size of windows, D_p increases and the size is checked against the boundary of the local window, $N \times N$ (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006). If the boundary has been reached but it has not reach the maximum boundary $M \times M$, the iteration continues by incrementing the size of boundary by 2, i.e $N + 2$ (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006). This process is repeated until the maximum boundary is reached in which then the process is terminated (Yamaguchi *et al.*, 2008; Yamaguchi and Hashimoto, 2006).

Figure 2.6 shows the resultant image after it has undergone the percolation process. Figure 2.6(a) is the resultant image when the focal pixel is a background pixel and Figure 2.6(b) is the resultant image when the focal pixel is a crack pixel.

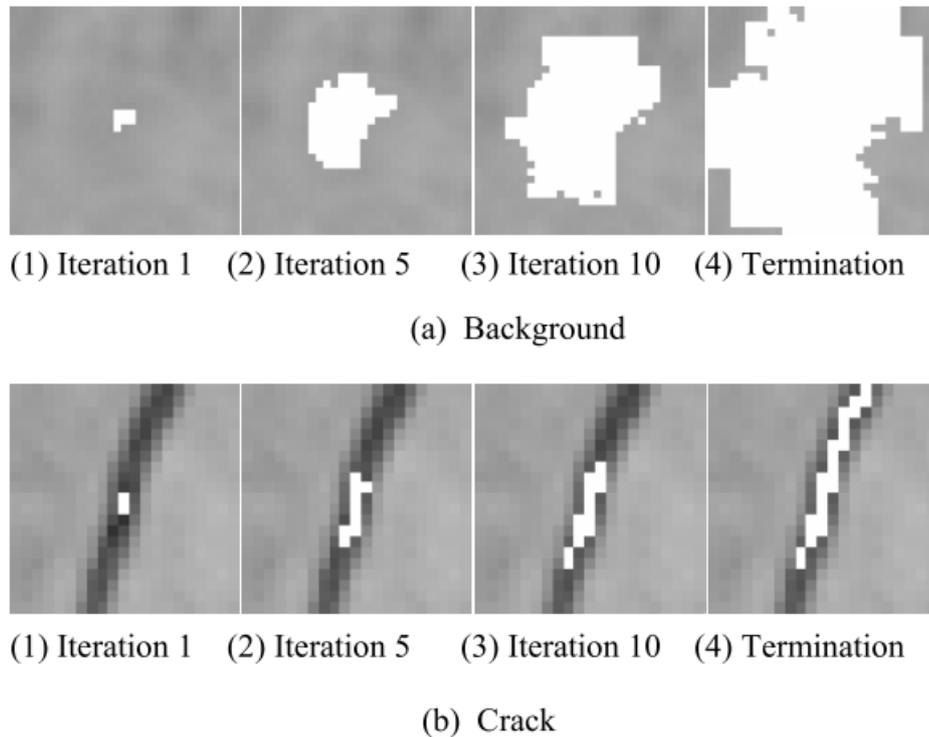


Figure 2.6: Output from percolation process (a) Background (b) Crack pixels
(Tomoyuki *et al.*, 2008)

2.3.2.2 Thermography

Thermography is a method of obtaining images that are produced by infrared radiation and is a non-destructive testing (NDT) method in diagnostics (Gao *et al.*, 2016). In crack detection, several papers discuss the usage of different types of thermography for crack detection.

In papers by Cheng and Tian (2011), Liu *et al.* (2013), Noethen *et al.* (2010), and Zhao *et al.* (2017), variations of eddy current (EC) thermography is used for crack detection. In paper by Liu *et al.* (2013), eddy current thermography is deployed via induction coils which are used to generate continuous alternating magnetic fields which causes eddy current to flow and rapidly heats up the material under inspection in a short amount of time. A thermographic camera with high resolution and frequency is then used to capture the temperature distribution and cracks are said to be present when there is a disturbance in the induced current flow which affects the temperature profile around the crack area (Liu *et al.*, 2013).

In papers by Cheng and Tian (2011), Noethen *et al.* (2010), and Zhao *et al.* (2017), pulsed EC thermography is used instead which differs by heating the material under inspection in pulses. The advantages of using pulsed EC thermography as opposed to

continuous EC thermography lies in the fact that pulsed methods can operate in multi frequency levels while continuous methods operate in a narrow frequency bandwidth. In pulsed modes, it consumes significantly less energy compared to continuous modes operating at the same intensity. The presence of cracks is also identified by changes in temperature profiles around crack regions similarly with continuous EC thermography. However, Jian *et al.* (2017) proposed an additional procedure that collects magnetic flux using ferrite core and injects it into the sample along with eddy current. Anomalies are detected if there is a change in temperature and magnetic field which makes it more sensitive to presence of defects.

Another thermography approach was proposed by El-amiri *et al.* (2016) which is known as pulsed thermography. Unlike pulsed EC thermography, there are no EC present in this setup. The material under test is heated up using lamps, flashes, or lasers (El-amiri *et al.*, 2016). Similarly to EC thermography, defects are characterised by the changes in temperature profiles (El-amiri *et al.*, 2016).

Although both thermography and EC thermography is able to detect cracks and defects, EC thermography is much more efficient for geometrically complex materials and shows more crack indication in addition of being faster in revealing anomalies compared to conventional thermography (Gao *et al.*, 2016). However, EC thermography only works in conductive materials (Gao *et al.*, 2016).

2.3.2.3 Machine Learning

Machine learning methods such as neural networks have been used to identify crack features after training a model with inputs such as features extracted from cracks. It is essentially a data modelling system that can represent the linear or nonlinear relationships between inputs and outputs (Maazi *et al.*, 2008).

Choudhary and Dey (2012), Gaith *et al.* (2015), Kim *et al.* (2001), Maazi *et al.* (2008), and Saar and Talvik (2010) have adapted neural networks to identify cracks by providing different types of inputs for the neural network with different results. Maazi *et al.* (2008) had trained a neural network using backpropagation method with 12 sample images that contains flaws and another 12 sample images with no flaws. The images with flaws consists of different types of flaws present in dielectric materials.

A similar approach was used by Kim *et al.* (2001) to identify the presence of cracks, delaminations, and cracks and delaminations in semiconductor devices using a

backpropagation neural network. Two different approaches were used for the inputs of the neural network. The first approach uses Kohonen's self organizing map as a dimensionality reduction for the image and the second uses image processing consisting of image filtering, binarization, and edge detection. The dimensionality reduction via image processing is concluded to be much faster and accurate compared to self organizing map.

Cloudhary and Dey (2012) compared fuzzy logic and neural network techniques to detect cracks on concrete surfaces. Edge detection is performed on the images of concrete surfaces to be used as inputs for the neural network and fuzzy logic model. The fuzzy logic model used trapezoidal membership functions with the parameters area and ratio of objects in the image used as input variables while the output is the class of the object (noise or crack). However, two approaches were used for neural networks which are image based and object based. In image based approach, the neural network is given the area and ratio of objects in an image and getting a binary output depending on whether there are cracks in the image. The second approach using object involves feeding the neural network with the area and ratio of one object at a time to obtain a single binary output which classifies it as either noise or crack. From their research, the object approach outperforms the image approach while both outperforms the fuzzy logic models.

Saar and Talvik (2010) also used a neural network that takes five input images. The first input are sub-images of the entire original image that may or may not contain crack features. The second and third inputs are median and maximum values of the summation of image matrix values column- and row-wise which corresponds to overall number of smaller defects and length of cracks respectively. The fourth and fifth inputs are binary images convolved with two 11 x 11 matrix where one has a horizontal line in the middle and the other vertical line in the middle to suppress noise. The resultant model is used to identify cracks on asphalt pavements.

Another approach utilizes measurements of natural frequencies to be used with neural networks to predict the presence of cracks (Gaith *et al.*, 2015). The neural network is trained using data obtained from ANSYS software which uses finite element techniques to generate data for both cracked and uncracked beams. This method measures natural frequencies obtained when subjecting cantilever beams to three different vibration modes which is then fed into the neural network for analysis.

2.3.2.4 Edge Detection Techniques

Edge detection techniques are methods that target regions of the image where there is a sudden change in intensity of pixel values. Section 2.3.2.4.1 to Section 2.3.2.4.4 explains different edge detection techniques which are anisotropic diffusion, probability based thresholding, and histogram equalization respectively.

2.3.2.4.1 Anisotropic Diffusion

Perona and Malik (1990) proposed a nonlinear method to reduce noise without affecting edges compared to conventional linear methods such as Gaussian (Tang and Gu, 2013) and low-pass filtering. The traditional Perona-Malik anisotropic diffusion filter smoothens each pixel according to the gradient magnitude (Tang, 2009; Tsai *et al.*, 2010). Regions with high gradient (i.e. crack regions or edges) are relatively unaffected while a strong smoothing effect will occur at regions with low gradient (crack or edge free regions) (Anwar and Abdullah, 2014; Tang, 2009). This is achieved by convolving the image with an isotropic Gaussian filter for several iterations. The resultant image is the sum of the convolved image with the resultant image in the previous iteration. and is expressed as Equation (2.2) (Perona and Malik, 1990).

$$I_{i,j}^{t+1} = I_{i,j}^t + \lambda [c_N \cdot \nabla I_N + c_S \cdot \nabla I_S + c_E \cdot \nabla I_E + c_W \cdot \nabla I_W]_{i,j}^t \quad (2.2)$$

where

$$\nabla I_{N(i,j)} = I_{(i-1,j)} - I_{(i,j)} \quad (2.3)$$

$$\nabla I_{S(i,j)} = I_{(i+1,j)} - I_{(i,j)} \quad (2.4)$$

$$\nabla I_{E(i,j)} = I_{(i,j+1)} - I_{(i,j)} \quad (2.5)$$

$$\nabla I_{W(i,j)} = I_{(i,j-1)} - I_{(i,j)} \quad (2.6)$$

$$c_D = g(\nabla I) = e^{-\left(\frac{|\nabla I|}{K}\right)^2} \quad (2.7)$$

$$c_D = g(\nabla I) = \frac{1}{1 + \left(\frac{|\nabla I|}{K}\right)^2} \quad (2.8)$$

where

$I_{i,j}^t$ is brightness value of the image at coordinate i and j during the iteration t.

$\nabla I_{D(i,j)}$ is nearest-neighbour differences where the subscript D denotes the direction of diffusion and is given as north (N), south (S), east (E), and west (W).

λ is conduction coefficient and must lie between the values of 0 to 0.25 for stability purposes.

c_D is diffusion coefficients function where the subscript D denotes the direction of diffusion and is given as north (N), south (S), east (E), and west (W).

∇I is brightness gradient of the image.

K is threshold that affects the blurring of discontinuities and sharpening of edges.

Equation (2.7) and Equation (2.8) are diffusion coefficient functions that favour high contrast edges over low contrast ones and wide regions over smaller ones respectively (Perona and Malik, 1990).

Oliveira and Correia (2010) used anisotropic diffusion to smooth random textures on pavement surfaces as the random textures makes it difficult to detect cracks easily using simple segmentation methods such as thresholding. Two smoothing models are tested where the first model favours high contrast edges over low contrast ones while the second model favours wide regions over smaller ones which is given as Equation (2.7) and Equation (2.8).

Tang and Gu (2013) used an adaptive anisotropic diffusion filter that uses a different brightness gradient in the diffusion coefficient functions to detect cracks in roads. The brightness gradient in Equation (2.7) and Equation (2.8) is replaced with a different operator that is sensitive to noise which may produce a weak edge, which is given by:

$$\nabla I = \sqrt{\frac{|\nabla I_N(i,j)|^2 + |\nabla I_S(i,j)|^2 + |\nabla I_E(i,j)|^2 + |\nabla I_W(i,j)|^2}{4}} \quad (2.9)$$

Tsai *et al.* (2010), and Anwar and Abdullah (2014) have used a modified model of Perona-Malik anisotropic diffusion to detect microcracks in solar cells. The approach used by both researches modifies the anisotropic diffusion filter to smoothen pixels at crack regions while maintaining the pixel values at regions that are crack free. The resulting image is then subtracted with the original grey image which leaves the pixels at crack regions intact.

Both researches proposed a different model for the modified Perona-Malik anisotropic diffusion coefficient function which is given by Equation (2.10) and Equation (2.12).

$$g(\nabla I, \mathbf{u}, \mathbf{v}) = \frac{1}{1 + \left[\frac{\prod_i u_i}{\prod_j v_j} \cdot \left(\frac{K}{|\nabla I|} \right)^2 \right]} \quad (2.10)$$

where

$\nabla I = \nabla I_t^i(\mathbf{x}, \mathbf{y})$, is the gradient of the pixel at coordinate \mathbf{x} and \mathbf{y} . $i = 1, 2, 3$, and 4 representing the gradients of four Laplacian neighbours in the north, south, east,

and west directions respectively. t represents the current iteration for the anisotropic diffusion (Anwar and Abdullah, 2014; Tsai *et al.*, 2010).

$\mathbf{u} = \{u_i\}$ and u_i are pixels with defects with feature values smaller than the defect-free pixels (Tsai *et al.*, 2010)

$\mathbf{v} = \{v_i\}$ and v_i are pixels with defects with feature values greater than the defect-free pixels (Tsai *et al.*, 2010)

K is edge stopping threshold where a value of K that is too small may result in no smoothing effect or too much smoothing if K is too large (Anwar and Abdullah, 2014)

$$g(\nabla I) = 1 - \frac{1}{1 + \left(\frac{|\nabla I|}{K}\right)^2} \quad (2.11)$$

which is then modified to Equation (2.11). Both equations perform the same function but Equation 2.11 is more sensitive when the threshold is low for the same gradient.

$$g(\nabla I) = 1 - \frac{1}{1 + \left(\frac{|\nabla I|}{h}\right)^2} \quad (2.12)$$

h is from Equation (2.12) and is defined as an edge stopping matrix that adaptively generates a unique threshold for each image pixel at coordinate x and y using input image grey values and is given in Equation (2.12) (Anwar and Abdullah, 2014):

$$h(x, y) = \frac{255}{1 + e^{-b(I_0(x, y) - \varepsilon)}} \quad (2.13)$$

where

b is the gradient ramp in the transfer function.

ε is the threshold value where the intensity of $I_0(x, y)$ is mapped to the center of the greyscale range.

In comparison, both approaches achieve the same result which smoothens the pixels with microcracks while the defect-free pixels remains unchanged. In contrast, Tsai *et al.* (2010) approach requires a trial and error in obtaining a good K parameter while Anwar and Abdullah (2014) approach adaptively generates a unique threshold for each pixel. The adaptive approach is more robust as it adapts to the image but it is slower to implement compared to the trial and error method.

2.3.2.4.2 Probability Based Thresholding

Huayong *et al.* (2011) proposed a method to detect cracks on a bridge deck as shown in Figure 2.8. The method attempts to capture as many crack pixels as possible against the background during binarization by modelling pixel density distribution with respect to position and frequency as shown in Figure 2.9 and Figure 2.10 respectively. It can be

seen that the distribution in Figure 2.10 conforms to a Normal distribution, $(N(\mu, \sigma))$ (Wu *et al.*, 2011).



Figure 2.7: Digital image of bridge crack (Huayong *et al.*, 2011)

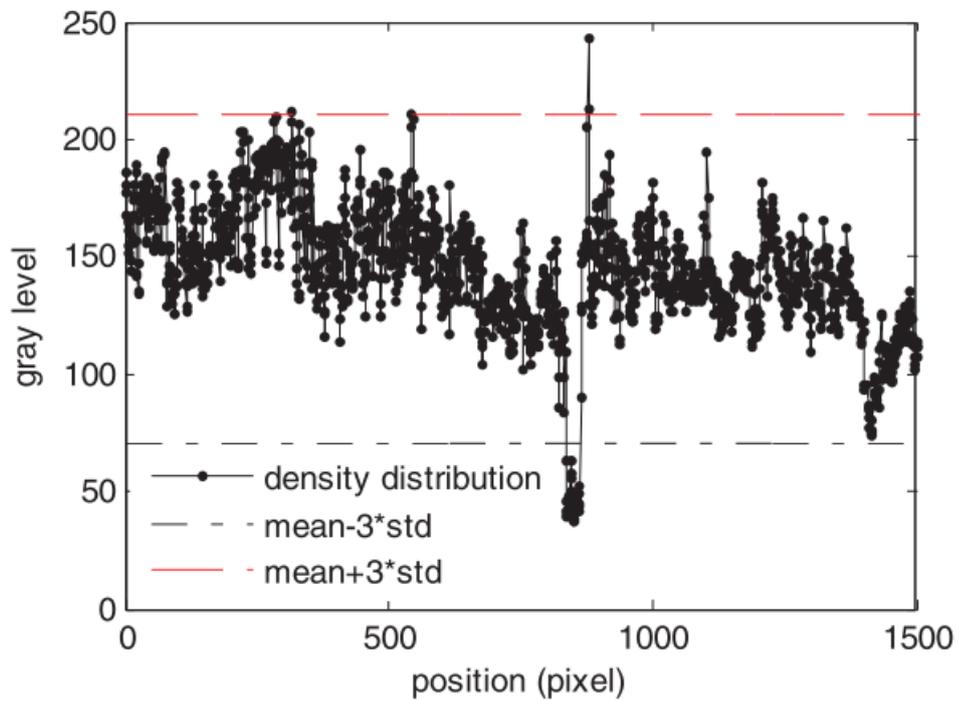


Figure 2.8: Pixel density distribution with respect to position (Huayong *et al.*, 2011)

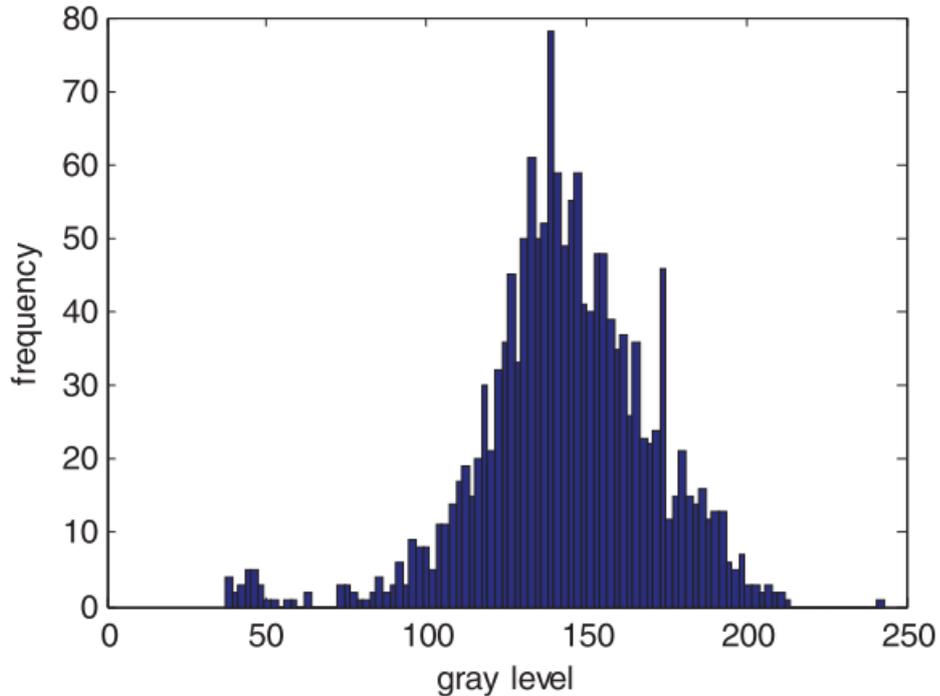


Figure 2.9: Pixel density distribution with respect to position (Huayong *et al.*, 2011)

From Huayong *et al.*'s (2011) (Wu *et al.*, 2011) research, there is a 99.74% confidence that any pixels with grey values within the range of $\mu - 3\sigma$ to $\mu + 3\sigma$ are background pixels and any pixels with values outside of that range has a high possibility of being a crack. A grey level of $\mu - 3\sigma$ is used as the binarization threshold to capture as many crack pixels as possible (Wu *et al.*, 2011). The binarization results is shown in Figure 2.11.

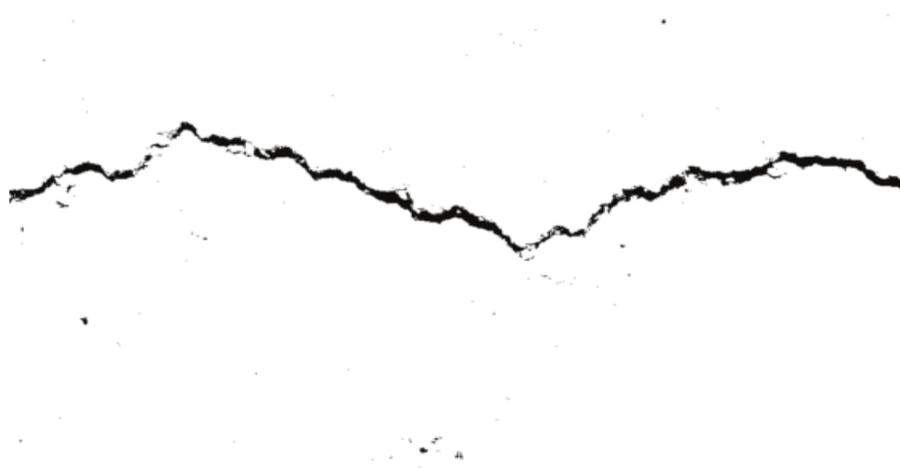


Figure 2.10: Resultant binarized image using threshold $\mu - 3\sigma$ (Huayong *et al.*, 2011)

After binarization, noise removal is employed using block filter algorithm which is proposed by Wu *et al.* (2011) to remove salt and pepper noise and isolated islands which are groups of noise pixels. Although the same effect can be achieved by enlarging the mask for a median filter, the enlarged mask will also destroy the crack details extracted. A block filter works by creating a 3 x 3 mask and calculates the number of noise pixels covered by the mask. On the same pixel, a bigger mask with the size increment from the original 3 x 3 by $6N$ where N is the number of iterations this process will be performed is used and the number of noise pixels is calculated. In the event where the number of noise pixels from adjacent steps are the same, it is deduced that the pixels covered by the mask are isolated islands and all pixels covered by the mask is then set to the background value. On the other hand, if the number of noise pixel keeps increasing, then it is defined as part of the target crack.

2.3.2.4.3 Histogram Equalization

Histogram equalization is a contrast enhancement method that produces a brighter image compared to the original (Yadav *et al.*, 2016). This is achieved by redistributing the greyscale pixels in the image evenly (Dawson-Howe, 2014). Two methods of histogram equalization are reviewed here which are the conventional histogram equalization and contrast limited adaptive histogram equalization (CLAHE). Conventional equalization is automatic and requires no parameters set while CLAHE works by dividing the original image into a number of tiles and apply contrast enhancement on each tile before recombination into the original image (Yadav *et al.*, 2016).

Parveen and Sathik (2009) tried conventional histogram equalization and CLAHE to enhance the bone fracture images. From their research, it was concluded that CLAHE performs better in enhancing the fracture features in the X-ray bone image.

Ke *et al.* (2010) used histogram equalization to enhance the image of solar wafers for microcrack detection. In their research, the histogram equalized image is smoothed and filtered for particles before subtracted with the original greyscale image to segment the crack and this is then followed by binarization.

Qiu *et al.* (2012) used Laplace operator and histogram equalization to enhance the images of crack in casting billets. The Laplace operator sharpens the image while histogram equalization enhances the contrast for crack segmentation.

In Zhuang *et al.* (2004), an inverse histogram equalization operation was used to centralize the grey values to enhance the crack regions followed by binarization and Gauss-Laplacian transform. The Gauss-Laplacian transform combines Gauss filter with Laplacian operator. The former is used to smoothen the image while the latter detects steep edges (where there is a sharp change in brightness) (Choudhary and Dey, 2012) in the image (Zhuang *et al.*, 2004). A lowpass filter is a necessary step before Laplacian as it is susceptible to noise (Dawson-Howe, 2014; Zhuang *et al.*, 2004).

2.3.2.5 Scanning Acoustic Microscopy (SAM)

SAM is a non-destructive failure analysis of IC packages and works by imaging the IC package by firing an acoustic pulse of energy to the sample and analyses the returned echoes (Ma *et al.*, 2007). Any anomalies detected along the propagation paths will modify the amplitude and polarity of the returned ultrasonic waves. The waves are typically produced by a piezoelectric transducer equipped with acoustic lens to focus waves in a spot and moves mechanically to scan the entire sample. Both transducer and sample being scanned needs to be immersed in a fluid coupling system which are typically de-ionized water if scanning is conducted for IC packages as acoustic waves become attenuated quickly in air (Abdelhamid *et al.*, 2014). There are two modes available in SAM which are reflective mode which focuses the ultrasonic beam on a plane inside the object and through transmission mode which shows the internal features of the object (Ma *et al.*, 2007).

From that study, it is found that C-SAM where C represents the imaging in x-y plane at a specific depth is a good method to detect cracks, voids, delaminations, and regions of poor adhesion in IC packages. The different imaging methods are as shown in Figure 2.12.

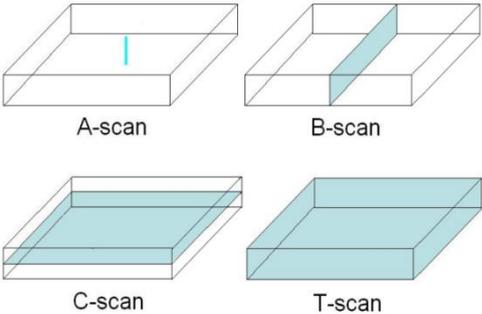


Figure 2.11: Imaging differences for different SAM scans (Ma *et al.*, 2007)

A-SAM scans performs scanning at a specific x-y coordinate and increments in the z-direction which is useful to supply information about a specific point at a specific depth (Ma et al., 2007). B-SAM scans in the x-direction and increments in the z-direction which makes it suitable for observing crack propagation and location of voids in packages. For C-SAM scans, the scan is performed in the x-direction and increments in the y-direction at a specific planar depth while T-scan measures transmitted sound loss which is useful for detecting delaminations and popcorn cracks in IC packages.

Abdul *et al.* (2000) and Abdul *et al.* (2003) also used SAMs to evaluate the quality of IC packages by correlating the simulated samples with reflection coefficients.

2.4 Summary

Table 2.1: Comparison of advantages and disadvantages of each method

Methods	Advantages	Disadvantages
Percolation	<ul style="list-style-type: none"> • Less susceptible to noise • Able to reconstruct entire crack feature 	<ul style="list-style-type: none"> • Computationally taxing and slow
Machine learning	<ul style="list-style-type: none"> • High accuracy in identifying crack features • Can be used to classify different types of cracks 	<ul style="list-style-type: none"> • Training beforehand may require few hundreds of images before getting a good model
Thermography	<ul style="list-style-type: none"> • Sensitive 	<ul style="list-style-type: none"> • Other defects and anomalies also result in disturbance in measurements • Requires specialized hardware • May damage sensitive IC packages
Edge detection techniques	<ul style="list-style-type: none"> • Usually less computationally expensive 	<ul style="list-style-type: none"> • Usually susceptible to noise and requires denoising procedures
SAM	<ul style="list-style-type: none"> • Sensitive • Widely used to detect defects in IC packages 	<ul style="list-style-type: none"> • Requires specialized hardware

The methods reviewed for crack detection can be loosely classified into three different approaches which are image processing approach, image acquisition approach,

and machine learning. In image acquisition approach, it requires the usage of specialized hardware or tools such as thermography and SAM. For an image processing approach, the focus typically lies in image enhancement, segmentation, and noise reduction for crack detection purposes. Machine learning would focus on identifying key features of the desired features and train a model to identify these features.

In the hardware approach, it is typically more sensitive compared to an image processing approach as any anomalies or defects in the measured material will result in an obvious disturbance in the measured readings. However, in the case of thermography, it is not suitable for crack detection in IC packages as it utilizes EM waves which may damage sensitive ICs. In addition, the method heats the material under testing in rapid succession which may shorten the lifespan of the IC. On the other hand, SAM is very suitable for detection of defects in IC packages including cracks but it requires specialized hardware to be able to do so.

On the image processing approach, percolation-based crack detection is a good algorithm to detect cracks as it is less susceptible to noise but is computationally taxing compared to edge detection techniques. In contrast, percolation-based method does not implement algorithms that enhances crack features when compared to edge detection techniques for Perona-Malik anisotropic diffusion model and histogram equalization. In probability based thresholding, the algorithm only works well if the image is not significantly noisy or textured. Histogram equalization on the other hand enhances the contrast of the image which helps to differentiate the background and foreground (crack regions) further. Perona-Malik's anisotropic diffusion on the other is specific and target edges. It is useful to reduce noise in image without blurring significant features.

Another approach, machine learning is proven to be very good at identifying crack features. However, this method requires training a model beforehand which may require few hundreds of images to have a good performance.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In microcrack detection, it is necessary that the algorithm be sensitive enough to detect the features of the crack without too much interference from noise in the image. Three methods are proposed and evaluated to determine their performance in crack detection and noise reduction. The first method has two stages which are image crack segmentation and image denoising. In the second method, there are three stages which are image contrast enhancement, image crack segmentation, and image denoising. For the third method, it is separated into three stages which are image crack enhancement, image denoising, and image crack segmentation. The input images for the system are the images of IC packages with cracks and the output images for the system are the segmented crack regions.

Microsoft Visual Studio 2017 with OpenCV imaging library is used to build the algorithm of the system. No additional hardware is required for this system as it is purely software based.

3.2 Proposed Method 1: Probability Based Thresholding Method

Section 3.2.1 provides an overview of probability based thresholding method. An explanation of the stages in this method is provided in Section 3.2.1.1 and Section 3.2.1.2.

3.2.1 Method Overview

This method performs crack detection by analyzing the histogram distribution of the image. By using the theory behind probability distribution for a normal distribution, cracks are segmented by determining the regions where pixels are most likely to belong to crack pixels. The flowchart of this method is as shown in Figure 3.1.