

การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์
ด้วยเครื่องมือทางการประมวลผลภาพ

EXPLORING IMAGE PROCESSING TOOLS FOR
PRINTED CIRCUIT BOARD DEFECT INSPECTION

จักรกฤษ อ่อนชื่นจิตร

ดุษฎีนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
ปีการศึกษา 2564
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์
ด้วยเครื่องมือทางการประมวลผลภาพ

จักรกฤษ อ่อนชื่นจิตร

ดุษฎีนิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิศวกรรมศาสตรดุษฎีบัณฑิต สาขาวิชาวิศวกรรมไฟฟ้า
คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
ปีการศึกษา 2564
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี

วิทยานิพนธ์ฉบับนี้เป็นงานวิจัยที่เกิดจากการค้นคว้าและวิจัย ขณะที่ข้าพเจ้าศึกษาอยู่ในคณะ
วิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ดังนั้นงานวิจัยในวิทยานิพนธ์ฉบับนี้ถือเป็น
ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี และข้อความต่าง ๆ ในวิทยานิพนธ์ฉบับนี้ ข้าพเจ้า
ขอรับรองว่าไม่มีการคัดลอกหรือนำงานวิจัยของผู้อื่นมานำเสนอในชื่อของข้าพเจ้า

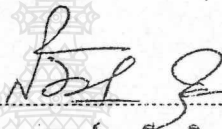
This thesis consists of research materials conducted at the Faculty of
Engineering, Rajamangala University of Technology Thanyaburi and hence the copyright
owner. I hereby certify that the thesis does not contain any forms of plagiarism.

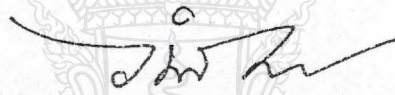
.....
(นายจักรกฤษ อ่อนชื่นจิตร)




หัวข้อวิทยานิพนธ์	การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ด้วยเครื่องมือทางการประมวลผลภาพ Exploring Image Processing Tools for Printed Circuit Board Defect Inspection
ชื่อ - นามสกุล	นายจักรกฤษ อ่อนชื่นจิตร
สาขาวิชา	วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์จักรี ศรีนนท์ฉัตร, Ph.D.
ปีการศึกษา	2564

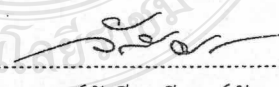
คณะกรรมการสอบวิทยานิพนธ์


..... ประธานกรรมการ
(รองศาสตราจารย์สมเกียรติ อุดมธรรษากุล, Ph.D.)



..... กรรมการ
(ผู้ช่วยศาสตราจารย์อำนาจ เรืองวารี, Dr.-Ing.)


..... กรรมการ
(อาจารย์วิเชียร อุปแก้ว, Ph.D.)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์นรเสฏฐ์ วิชัยพาณิชย์, วศ.ด.)


..... กรรมการ
(ผู้ช่วยศาสตราจารย์จักรี ศรีนนท์ฉัตร, Ph.D.)

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี อนุมัติวิทยานิพนธ์ฉบับนี้เป็น
ส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาตรีบัณฑิต


..... คณบดีคณะวิศวกรรมศาสตร์
(ผู้ช่วยศาสตราจารย์ศิวกร อ่างทอง, Ph.D.)
วันที่ 29 เดือน ตุลาคม พ.ศ. 2564

หัวข้อคุณนิพนธ์	การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ด้วยเครื่องมือทางการประมวลผลภาพ
ชื่อ – นามสกุล	นายจักรกฤษ อ่อนชื่นจิตร
สาขาวิชา	วิศวกรรมไฟฟ้า
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์จักรี ศรีนนท์ฉัตร, Ph.D.
ปีการศึกษา	2564

บทคัดย่อ

ระบบตรวจสอบแผ่นวงจรพิมพ์ (PCB) มีความสำคัญเป็นอย่างมากในอุตสาหกรรมการผลิตชิ้นส่วนอิเล็กทรอนิกส์ การออกแบบระบบตรวจสอบให้มีประสิทธิภาพที่ดีนอกจากจะช่วยให้ได้ PCB ที่มีคุณภาพดีแล้ว ยังช่วยเพิ่มความเร็วของกระบวนการผลิต และส่งผลให้ต้นทุนการผลิตลดลงด้วย ถึงแม้ว่าเทคโนโลยีการตรวจสอบ PCB ที่มีอยู่ในปัจจุบันจะสามารถตรวจจับข้อบกพร่องได้แล้ว แต่ก็ยังไม่สามารถจำแนกประเภทข้อบกพร่องทั้งหมดได้อย่างครบถ้วนและแม่นยำ งานวิจัยนี้จึงนำเสนอระบบตรวจสอบข้อบกพร่อง PCB ที่สามารถตรวจจับข้อบกพร่อง พร้อมทั้งระบุตำแหน่ง ขนาด รูปร่างลักษณะของข้อบกพร่อง และสามารถจำแนกประเภทของข้อบกพร่องที่เกิดขึ้นบน PCB เปลา่แบบหน้าเดียวครบทั้ง 14 ชนิด โดยมีค่าความผิดพลาดในการจำแนกประเภทไม่เกิน 10 เปอร์เซ็นต์

งานวิจัยนี้ใช้วิธีการจัดกลุ่มแบบ Fuzzy c-means สำหรับการแบ่งส่วนภาพก่อนที่จะตรวจจับข้อบกพร่องด้วยวิธีการลบภาพ ในส่วนของการจำแนกประเภทข้อบกพร่องจะใช้การดำเนินการทางคณิตศาสตร์และตรรกศาสตร์ ร่วมกับการแปลงฮัฟแบบวงกลม การสร้างใหม่ทางสัญญาณวิทยา และการติดฉลากส่วนประกอบที่เชื่อมต่อ

ผลการทดสอบพบว่าระบบตรวจสอบข้อบกพร่อง PCB ที่นำเสนอนี้มีประสิทธิภาพในการตรวจจับข้อบกพร่อง 100 เปอร์เซ็นต์ พร้อมทั้งสามารถระบุตำแหน่ง ขนาด รูปร่างลักษณะของข้อบกพร่องได้อย่างชัดเจน และสามารถจำแนกประเภทของข้อบกพร่องได้ครบทั้ง 14 ชนิด โดยมีความแม่นยำในการจำแนกประเภทของข้อบกพร่อง PCB เฉลี่ยอยู่ที่ประมาณ 99.05 เปอร์เซ็นต์

คำสำคัญ: การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ การแปลงฮัฟแบบวงกลม การสร้างใหม่ทางสัญญาณวิทยา การติดฉลากส่วนประกอบที่เชื่อมต่อ

Dissertation Title	Exploring Image Processing Tools for Printed Circuit Board Defect Inspection
Name – Surname	Mr.Jakkrit Onshaunjit
Program	Electrical Engineering
Dissertation Advisor	Assistant Professor Jakkree Srinonchat, Ph.D.
Academic Year	2021

ABSTRACT

Printed circuit board (PCB) inspection systems are essential in the electronics manufacturing industry. Designing an effective inspection system will not only help to get good quality PCBs but it also speeds up the production process and results in lower production costs. Although existing PCB inspection technology is capable of detecting defects, not all defects can be classified accurately and comprehensively. This research therefore proposes a PCB flaw detection system that can detect defects. It identifies the location, size, shape and nature of the defect and can identify 14 types of defects on a single-sided blank PCB with a classification error of no more than 10%.

This research uses fuzzy c-means clustering for image segmentation before defect detection via image subtraction. Arithmetic and logic operations, circle hough transform (CHT), morphological reconstruction (MR), and connected component labeling (CCL) are used in defect classification.

The research results showed that the proposed PCB flaw inspection system is 100% effective in detecting defects with the ability to clearly identify the location, size, shape, and nature of defects. All 14-known PCB defect types can also be classified, with an average PCB defect classification accuracy of about 99.05%.

Keywords: PCB defect inspection, circular hough transform, morphological reconstruction, connected component labeling

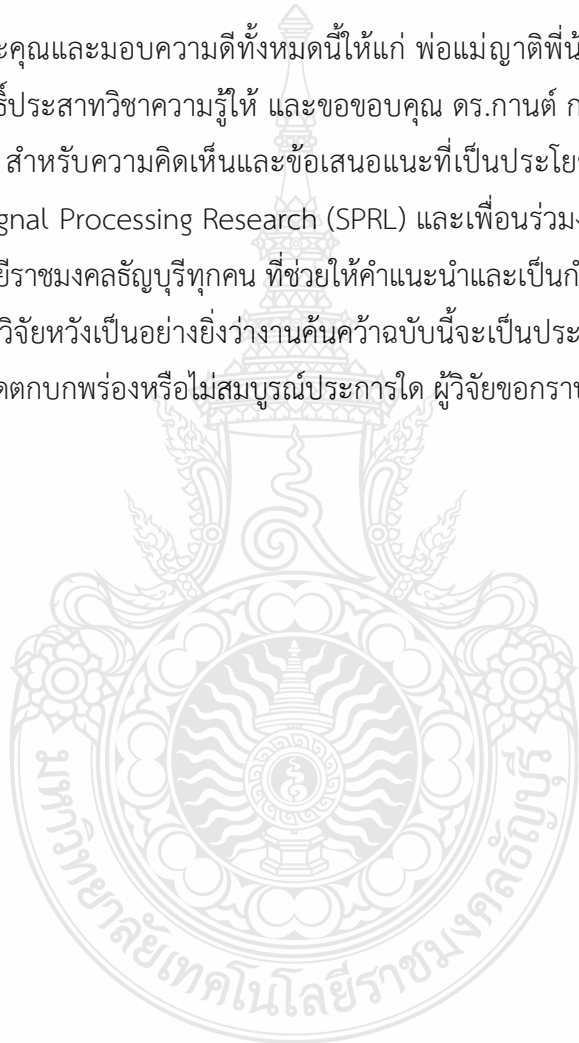
กิตติกรรมประกาศ

การจัดทำคุษฎีนิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยการสนับสนุนเป็นอย่างดีจากอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร.จักรี ศรีนนท์ฉัตร ที่ได้ช่วยให้คำแนะนำและติดตามการทำคุษฎีนิพนธ์ครั้งนี้อย่างใกล้ชิดตลอดมา นับตั้งแต่เริ่มต้นจนกระทั่งสำเร็จเรียบร้อยสมบูรณ์ ผู้วิจัยรู้สึกซาบซึ้งในความกรุณาของท่านเป็นอย่างยิ่ง

ขอขอบพระคุณและมอบความดีทั้งหมดนี้ให้แก่ พ่อแม่ญาติพี่น้องและคณาจารย์ ที่ให้การสนับสนุนและประสิทธิ์ประสาทวิชาความรู้ให้ และขอขอบคุณ ดร.กานต์ กาญจนะ จาก KB Systems (Thailand) Co., Ltd. สำหรับความคิดเห็นและข้อเสนอแนะที่เป็นประโยชน์มากมาย รวมทั้งสมาชิกร่วมห้องปฏิบัติการ Signal Processing Research (SPRL) และเพื่อนร่วมงานที่คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรีทุกคน ที่ช่วยให้คำแนะนำและเป็นกำลังใจที่ติดตลอดมา

สุดท้ายนี้ ผู้วิจัยหวังเป็นอย่างยิ่งว่างานชิ้นคว้ฉบับนี้จะเป็ประโยชน์สำหรับผู้สนใจ หากคุษฎีนิพนธ์ในครั้งนีัขาดตกบกพร่องหรือไม่สมบูรณ์ประการใด ผู้วิจัยขอกราบขออภัยมา ณ โอกาสนี้ด้วย

จักรกฤษ อ่อนชื่นจิตร



สารบัญ

	หน้า
บทคัดย่อภาษาไทย	(3)
บทคัดย่อภาษาอังกฤษ	(4)
กิตติกรรมประกาศ	(5)
สารบัญ	(6)
สารบัญตาราง	(8)
สารบัญรูป	(9)
คำอธิบายสัญลักษณ์และคำย่อ	(11)
บทที่ 1 บทนำ	12
1.1 ความเป็นมาและความสำคัญของปัญหา	12
1.2 วัตถุประสงค์การวิจัย	13
1.3 ขอบเขตของการวิจัย	13
1.4 ขั้นตอนการวิจัย	13
1.5 ประโยชน์ที่คาดว่าจะได้รับ	14
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	15
2.1 ข้อบกพร่องแผ่นวงจรพิมพ์	15
2.2 การมองเห็นของเครื่องจักรและการประมวลผลภาพ	22
2.3 เครื่องมือทางการประมวลผลภาพ	26
2.4 งานวิจัยที่เกี่ยวข้อง	29
บทที่ 3 วิธีดำเนินการวิจัย	35
3.1 การออกแบบระบบในภาพรวม	35
3.2 การเก็บรวบรวมข้อมูล	36
3.3 การประมวลผลภาพเบื้องต้น	36
3.4 การตรวจจับข้อบกพร่อง	42
3.5 การออกแบบอัลกอริทึม	43
3.6 วิธีการทดสอบงานวิจัย	47
บทที่ 4 ผลการทดสอบและการวิเคราะห์	51

สารบัญ (ต่อ)

	หน้า
4.1 ผลการทดสอบ Pre-processing	51
4.2 ผลการทดสอบ Image registration	54
4.3 ผลการทดสอบ Image segmentation	55
4.4 ผลการทดสอบ Defect detection	60
4.5 ผลการทดสอบวิธีการ CHT	62
4.6 ผลการทดสอบวิธีการ MR	63
4.7 ผลการทดสอบวิธีการ CCL	65
4.8 ผลการทดสอบประสิทธิภาพของอัลกอริทึม	67
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	72
5.1 สรุปผลการวิจัย	72
5.2 ข้อเสนอแนะ	73
บรรณานุกรม	74
ภาคผนวก	79
ภาคผนวก ก โปรแกรม	80
ประวัติผู้เขียน	109

สารบัญตาราง

	หน้า
ตารางที่ 2.1 การจำแนกประเภทของข้อบกพร่องตามกระบวนการและลักษณะการเกิดข้อบกพร่อง	17
ตารางที่ 4.1 ผลการแบ่งส่วนภาพด้วยวิธีการ Global Thresholding	55
ตารางที่ 4.2 ผลการแบ่งส่วนภาพด้วยวิธีการ Adaptive Thresholding	55
ตารางที่ 4.3 ผลการแบ่งส่วนภาพด้วยวิธีการ K-means Clustering	56
ตารางที่ 4.4 ผลการแบ่งส่วนภาพด้วยวิธีการ Fuzzy C-means Clustering	56
ตารางที่ 4.5 ตัวอย่างการเปรียบเทียบผลการแบ่งส่วนภาพ	57
ตารางที่ 4.6 ค่า SD ของแต่ละตัวชี้วัดในรูปที่ 4.2	60
ตารางที่ 4.7 ข้อมูลผลการทดลองที่ได้จากตัวอย่างในรูปที่ 4.9	61
ตารางที่ 4.8 ข้อมูลผลการทดลองที่ได้จากตัวอย่างในรูปที่ 4.10	62
ตารางที่ 4.9 จำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพ PCB ต้นแบบในรูปที่ 4.14	65
ตารางที่ 4.10 จำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพ PCB ต้นแบบในรูปที่ 4.15	66
ตารางที่ 4.11 ลักษณะการเปลี่ยนแปลงเมื่อเกิดข้อบกพร่องจากวิธีการ CCL	66
ตารางที่ 4.12 ผลการบันทึกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.16	68
ตารางที่ 4.13 ผลการบันทึกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.18	69
ตารางที่ 4.14 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่นำเสนอกับวิธีการอื่น	71

สารบัญรูป

	หน้า
รูปที่ 2.1 ตัวอย่างแผ่นวงจรพิมพ์	15
รูปที่ 2.2 ตัวอย่างกระบวนการผลิตแผ่นวงจรพิมพ์ในอุตสาหกรรม	16
รูปที่ 2.3 ตัวอย่างของข้อบกพร่องมาตรฐานบนแผ่นวงจรพิมพ์	17
รูปที่ 2.4 ลักษณะของข้อบกพร่อง Breakout	18
รูปที่ 2.5 ลักษณะของข้อบกพร่อง Pinhole	18
รูปที่ 2.6 ลักษณะของข้อบกพร่อง Open circuit	19
รูปที่ 2.7 ลักษณะของข้อบกพร่อง Under etch	19
รูปที่ 2.8 ลักษณะของข้อบกพร่อง Mouse bite	19
รูปที่ 2.9 ลักษณะของข้อบกพร่อง Missing conductor	20
รูปที่ 2.10 ลักษณะของข้อบกพร่อง Spur	20
รูปที่ 2.11 ลักษณะของข้อบกพร่อง Short circuit	20
รูปที่ 2.12 ลักษณะของข้อบกพร่อง Wrong size hole	21
รูปที่ 2.13 ลักษณะของข้อบกพร่อง Conductor too close	21
รูปที่ 2.14 ลักษณะของข้อบกพร่อง Spurious copper	21
รูปที่ 2.15 ลักษณะของข้อบกพร่อง Excessive short	21
รูปที่ 2.16 ลักษณะของข้อบกพร่อง Missing hole	22
รูปที่ 2.17 ลักษณะของข้อบกพร่อง Over etch	22
รูปที่ 2.18 ตัวอย่างการประมวลผลภาพ	23
รูปที่ 2.19 กระบวนการ Unsupervised Classification	25
รูปที่ 2.20 กระบวนการ Supervised Classification	26
รูปที่ 2.21 การแปลงฮัฟแมนแบบวงกลม	27
รูปที่ 2.22 หลักการของ Morphological reconstruction	27
รูปที่ 2.23 การประยุกต์ใช้ Morphological reconstruction กับภาพแบบไบนารี	28
รูปที่ 2.24 การประยุกต์ใช้ Morphological reconstruction กับภาพระดับสีเทา	28
รูปที่ 2.25 องค์ประกอบของภาพ	28
รูปที่ 2.26 การระบุหมายเลขให้กับองค์ประกอบของภาพโดยพิจารณาใน 4 ทิศทาง	29

สารบัญรูป (ต่อ)

	หน้า
รูปที่ 3.1 การออกแบบระบบในภาพรวม	35
รูปที่ 3.2 ขั้นตอนการประมวลผลภาพเบื้องต้น	36
รูปที่ 3.3 ตรีโกณมิติของสามเหลี่ยมมุมฉาก	39
รูปที่ 3.4 ขั้นตอนการตรวจจับข้อบกพร่อง	43
รูปที่ 3.5 ขั้นตอนการจำแนกชนิดข้อบกพร่อง	45
รูปที่ 3.6 โพลาร์ชาร์ตของอัลกอริทึมที่ออกแบบ	48
รูปที่ 4.1 ตัวอย่างภาพจากขั้นตอน ROI	51
รูปที่ 4.2 ตัวอย่างภาพจากขั้นตอน Gray scale conversion	52
รูปที่ 4.3 ตัวอย่างภาพจากขั้นตอน Noise reduction	52
รูปที่ 4.4 ตัวอย่างภาพจากขั้นตอน Image enhancement	53
รูปที่ 4.5 ตัวอย่างภาพจากขั้นตอน Image enhancement	53
รูปที่ 4.6 ตัวอย่างภาพจากขั้นตอน Image registration	54
รูปที่ 4.7 ตัวอย่างผลการแบ่งส่วนภาพ	57
รูปที่ 4.8 การเปรียบเทียบค่าเฉลี่ยของแต่ละตัวชี้วัด	59
รูปที่ 4.9 ตัวอย่างของการตรวจจับข้อบกพร่องจากภาพ PCB ต้นแบบที่ 1	60
รูปที่ 4.10 ตัวอย่างของการตรวจจับข้อบกพร่องจากภาพ PCB ต้นแบบที่ 2	61
รูปที่ 4.11 ตัวอย่างผลของการตรวจหา Hole pad ของลายวงจร	63
รูปที่ 4.12 ตัวอย่างผลของการตรวจหาช่องว่างขนาดเล็กของภาพตัวอย่างที่ 1	64
รูปที่ 4.13 ตัวอย่างผลของการตรวจหาช่องว่างขนาดเล็กของภาพตัวอย่างที่ 2	64
รูปที่ 4.14 ผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ต้นแบบ	65
รูปที่ 4.15 ผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ที่มีข้อบกพร่อง	66
รูปที่ 4.16 ตัวอย่างของการจำแนกข้อบกพร่องจากภาพ PCB ต้นแบบที่ 1	67
รูปที่ 4.17 ผลการจำแนกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.16	68
รูปที่ 4.18 ตัวอย่างของการจำแนกข้อบกพร่องจากภาพ PCB ต้นแบบที่ 2	69
รูปที่ 4.19 ผลการจำแนกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.18	70

คำอธิบายสัญลักษณ์และคำย่อ

AT	Adaptive Thresholding
CCL	Connected Component Labelling
CHT	Circular Hough Transform
CNN	Convolutional Neural Network
FCM	Fuzzy C-means Clustering
GT	Global Thresholding
IoU	Intersection over Union
KM	K-means Clustering
k-NN	k-Nearest Neighbors
MR	Morphological Reconstruction
PCA	Printed Circuit Assembly
PCB	Printed Circuit Board
PCBA	Printed Circuit Board Assembly
ROI	Region of Interest
SD	Standard Deviation



บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์ ระบบตรวจสอบข้อบกพร่องของแผ่นวงจรพิมพ์ (PCB Inspector) จะทำหน้าที่ป้องกันไม่ให้แผ่นวงจรพิมพ์ (Printed Circuit Board: PCB) ที่มีข้อบกพร่องก้าวไปสู่ขั้นตอนการผลิตต่อไปได้ เทคโนโลยีการตรวจสอบข้อบกพร่อง PCB ด้วยการประมวลผลภาพแบบอัตโนมัตินิยมนำมาใช้เพื่อตรวจจับ PCB ที่มีข้อบกพร่องมากขึ้นเนื่องจากประสิทธิภาพความแม่นยำและประสิทธิผลที่สูงขึ้นเมื่อเปรียบเทียบกับ การตรวจสอบด้วยสายตามนุษย์ หลักการตรวจสอบข้อบกพร่องของ PCB แบบอัตโนมัติสามารถแบ่งออกได้เป็น 3 วิธีการ ได้แก่ วิธีการอ้างอิง (Referential) วิธีการไม่อ้างอิง (Non-referential) และแบบไฮบริด (Hybrid) [1] สำหรับวิธีการอ้างอิง ข้อบกพร่องของ PCB จะถูกตรวจพบโดยการเปรียบเทียบภาพ PCB กับภาพ PCB แม่แบบ ส่วนวิธีการที่ไม่อ้างอิงจะใช้การกำหนดกฎทางคณิตศาสตร์ไว้ล่วงหน้าเพื่อตรวจจับข้อบกพร่องของ PCB และหลักการไฮบริดคือการผสมผสานของวิธีการอ้างอิงและไม่อ้างอิง อย่างไรก็ตามแม้ว่าเทคโนโลยีการตรวจสอบข้อบกพร่องที่มีอยู่จะสามารถตรวจจับข้อบกพร่องทั้งหมดได้ แต่ก็ยังไม่สามารถจำแนกชนิดของข้อบกพร่อง PCB ทั้งหมดได้ครบถ้วนและถูกต้องแม่นยำเพียงพอในการค้นหาสาเหตุที่แท้จริงของการเกิดข้อบกพร่อง PCB สำหรับกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์

ในงานวิจัยนี้จึงนำเสนออัลกอริทึมที่สามารถตรวจจับและจำแนกประเภทข้อบกพร่องของ PCB ที่รู้จักทั้ง 14 ชนิดที่เกิดขึ้นบนแผ่นวงจรพิมพ์เปล่า (Bare PCB) แบบหน้าเดียว (Single side PCB) โดยจะใช้วิธีการจัดกลุ่มแบบ Fuzzy C-means สำหรับการแบ่งส่วนภาพ (Image segmentation) ก่อนที่จะตรวจจับข้อบกพร่อง (Defects detection) ด้วยวิธีการการลบภาพ (Image subtraction) ในส่วนของการจำแนกประเภทข้อบกพร่อง (Defects classification) จะทำการออกแบบอัลกอริทึมโดยใช้เครื่องมือทางการประมวลผลภาพต่าง ๆ ได้แก่ การดำเนินการทางคณิตศาสตร์และตรรกศาสตร์ (Mathematical and logical operations) การแปลงฮัฟแบบวงกลม (Circular hough transform: CHT) การสร้างใหม่ทางสัณฐานวิทยา (Morphological reconstruction: MR) และการติดฉลากส่วนประกอบที่เชื่อมต่อ (Connected component labelling: CCL)

1.2 วัตถุประสงค์การวิจัย

ศึกษาและพัฒนาระบบตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ (PCB Defects Inspection) เพื่อตรวจหาตำแหน่ง ขนาด และรูปร่างลักษณะของข้อบกพร่องที่เกิดขึ้นจากกระบวนการเจาะและกัดลายวงจรทั้งหมดของแผ่นวงจรพิมพ์เปล่า และทำการจำแนกประเภทข้อบกพร่องของแผ่นวงจรพิมพ์ที่รู้จักทั้ง 14 ชนิดที่เกิดขึ้นบนแผ่นวงจรพิมพ์เปล่าแบบหน้าเดียว โดยใช้เครื่องมือทางการประมวลผลภาพต่าง ๆ

1.3 ขอบเขตของการวิจัย

1.3.1 ออกแบบและพัฒนาระบบการตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ด้วยเครื่องมือทางการประมวลผลภาพ

1.3.2 สามารถตรวจหาตำแหน่ง ขนาด และรูปร่างลักษณะของข้อบกพร่องแผ่นวงจรพิมพ์

1.3.3 สามารถจำแนกประเภทข้อบกพร่องของแผ่นวงจรพิมพ์ที่รู้จักทั้ง 14 ชนิดที่เกิดขึ้นบนแผ่นวงจรพิมพ์เปล่าแบบหน้าเดียว

1.3.4 ความผิดพลาดในการจำแนกข้อบกพร่องไม่เกิน 10 เปอร์เซ็นต์

1.4 ขั้นตอนการวิจัย

1.4.1 ศึกษาทฤษฎีการตรวจสอบแผ่นวงจรพิมพ์ ข้อบกพร่องของแผ่นวงจรพิมพ์ การมองเห็นของเครื่องจักรและการประมวลผลภาพ เครื่องมือทางการประมวลผลภาพ และรวบรวมวรรณกรรมที่เกี่ยวข้อง

1.4.2 ออกแบบระบบการตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ในภาพรวม

1.4.3 ออกแบบระบบการเก็บรวบรวมข้อมูลภาพ การประมวลผลภาพเบื้องต้น และการตรวจจับข้อบกพร่อง

1.4.4 วิเคราะห์หาตำแหน่ง ขนาด และรูปร่างลักษณะข้อบกพร่องของแผ่นวงจรพิมพ์

1.4.5 สร้างอัลกอริทึมสำหรับจำแนกประเภทข้อบกพร่องของแผ่นวงจรพิมพ์

1.4.6 ทดสอบอัลกอริทึมและวิเคราะห์ผลการทดสอบ

1.4.7 จัดทำรูปเล่มดุษฎีนิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 เพื่อเป็นข้อมูลในการศึกษาหลักการประมวลผลภาพแบบดิจิทัล

- 1.5.2 เพื่อเป็นแนวทางในการออกแบบระบบประมวลผลภาพ
- 1.5.3 เพื่อเป็นแนวทางในการตรวจหาข้อบกพร่องที่เกิดขึ้นบนแผ่นวงจรพิมพ์
- 1.5.4 เพื่อเป็นแนวทางในการพัฒนาอัลกอริทึมสำหรับจำแนกชนิดข้อบกพร่องของแผ่นวงจรพิมพ์
- 1.5.5 เพื่อพัฒนาเทคโนโลยีไปสู่การใช้งานจริงในอุตสาหกรรม



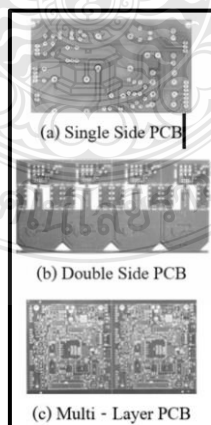
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้จะกล่าวถึงหลักการและทฤษฎีที่ใช้ในการทำดัชนีพจนานุกรม โดยเนื้อหาจะประกอบด้วย ข้อบกพร่องแผ่นวงจรพิมพ์ การมองเห็นของเครื่องจักรและการประมวลผลภาพ การประมวลผลขั้นต้น การแบ่งส่วนภาพ การจำแนกภาพ เครื่องมือทางการประมวลผลภาพ การแปลงฮัฟแบบวงกลม การสร้างใหม่ทางสัญญาณวิทยา และการติดตามลักษณะประกอบที่เชื่อมต่อ และในหัวข้อสุดท้ายจะเป็นงานวิจัยที่เกี่ยวข้อง

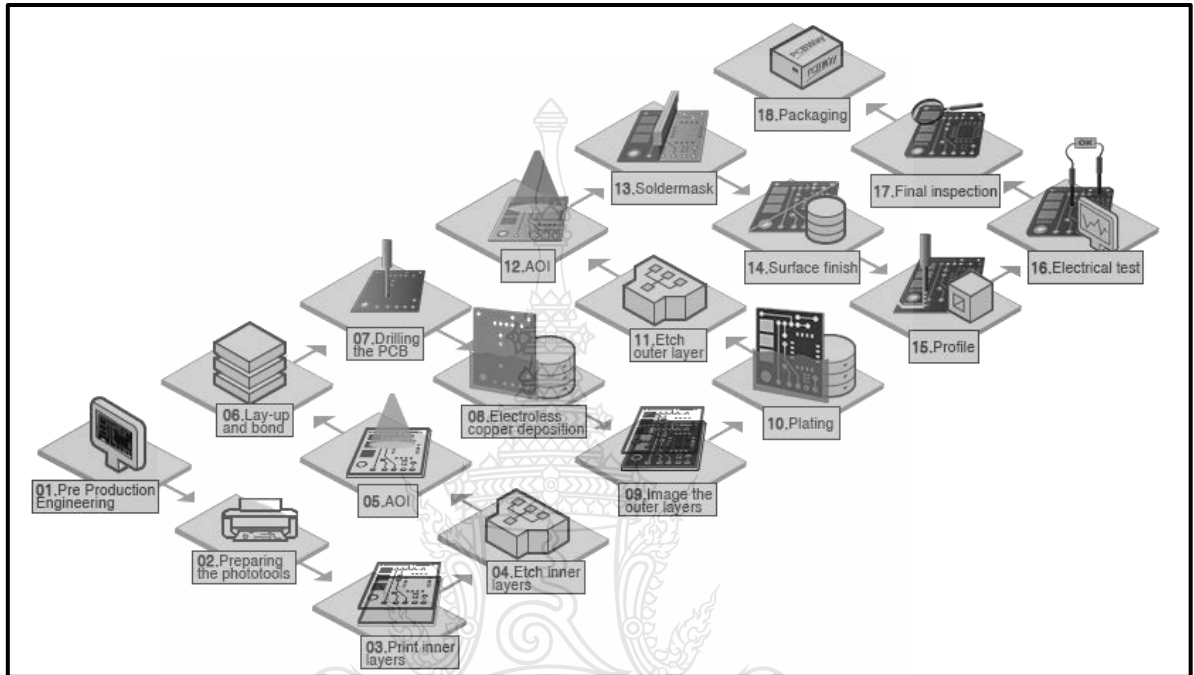
2.1 ข้อบกพร่องแผ่นวงจรพิมพ์

แผ่นวงจรพิมพ์ หรือ PCB เป็นส่วนประกอบพื้นฐานที่สำคัญของวงจรอิเล็กทรอนิกส์ โดย PCB มีหน้าที่หลักในการยึดอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ เข้าไว้ด้วยกัน มีตัวนำซึ่งเป็นลายทองแดง ทำหน้าที่เป็นทางเดินของสัญญาณไฟฟ้าจากอุปกรณ์ตัวหนึ่งไปสู่ตัวอื่น ๆ ตามวงจรที่ได้ออกแบบไว้ โดยมีการบัดกรีตะกั่วที่บริเวณขาของอุปกรณ์ที่จุดบัดกรี เพื่อเชื่อมต่อขาของอุปกรณ์เหล่านี้เข้ากับลายทองแดงของวงจรและเพื่อใช้ยึดอุปกรณ์เข้ากับ PCB ไปในตัวอีกด้วย PCB จะมีแผ่นทองแดงบาง ๆ เคลือบอยู่ตลอดแผ่น ในการใช้งานจำเป็นต้องกัดลายทองแดงบางส่วนออกไป ด้วยน้ำยาหรือกรดกัดแผ่นวงจรพิมพ์ โดย PCB จะมีอยู่หลายแบบ เช่น แบบหน้าเดียว (Single Side PCB), แบบ 2 หน้า (Double Side PCB), แบบ 2 หน้าเชื่อมต่อกัน (Double Side Plate Trough Hole PCB), แบบหลายชั้น (Multi - Layer PCB) เป็นต้น



รูปที่ 2.1 ตัวอย่างแผ่นวงจรพิมพ์ [1]

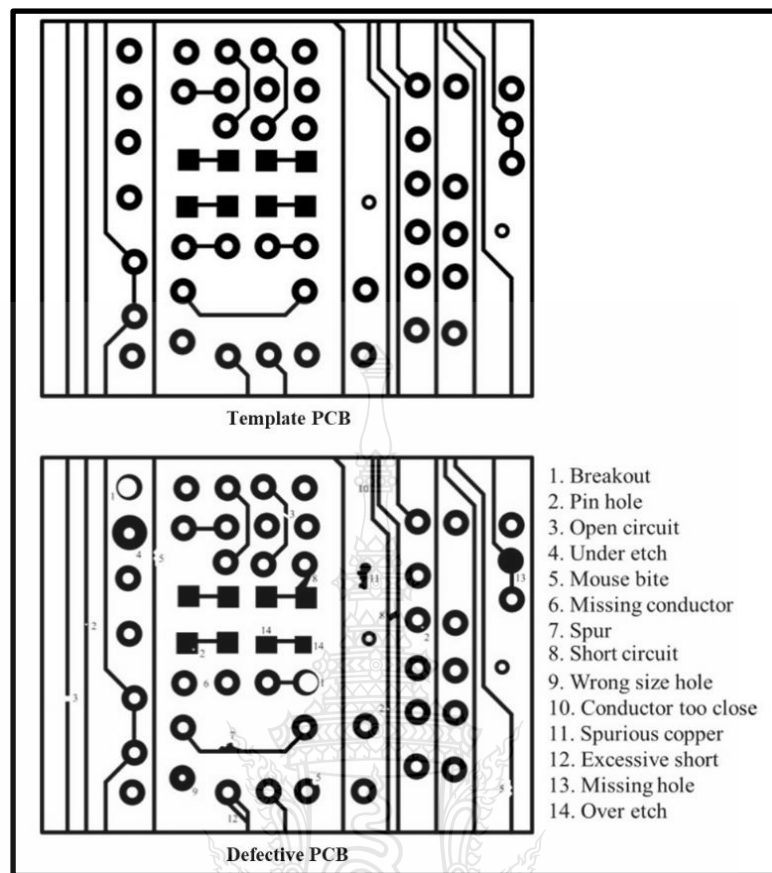
สำหรับการผลิตแผ่นวงจรพิมพ์จำนวนมาก ๆ นั้นต้องให้โรงงานอุตสาหกรรมเป็นผู้ผลิต ซึ่งประกอบด้วยกระบวนการผลิตในหลายขั้นตอน เช่น การออกแบบลายวงจร การกัดลายทองแดง การเจาะรู การเคลือบ และการตรวจสอบคุณภาพ เป็นต้น ซึ่งตัวอย่างของกระบวนการต่าง ๆ สำหรับผลิตแผ่นวงจรพิมพ์ในโรงงานอุตสาหกรรม แสดงในรูปที่ 2.2



รูปที่ 2.2 ตัวอย่างกระบวนการผลิตแผ่นวงจรพิมพ์ในอุตสาหกรรม [2]

ข้อบกพร่องของแผ่นวงจรพิมพ์ (PCB Defects) สามารถแบ่งได้เป็น 2 กลุ่มใหญ่ๆ คือ ข้อบกพร่องแบบ Functional และข้อบกพร่องแบบ Cosmetic ข้อบกพร่องแบบ Functional สามารถทำอันตรายต่อขั้นตอนการทำงานของวงจร ในขณะที่ข้อบกพร่องแบบ Cosmetic จะส่งผลต่อลักษณะที่ปรากฏของแผ่นวงจร และอาจส่งผลต่อประสิทธิภาพของวงจรในระยะยาว [3] ต่อมา M. Moganti และทีมวิจัยได้นิยามข้อบกพร่องมาตรฐานที่เกิดขึ้นบนแผ่น PCB เปล่าแบบด้านเดียวหลังจากผ่านกระบวนการกัดและเจาะในกระบวนการผลิต โดยจำแนกออกเป็นข้อบกพร่องที่รู้จัก 14 ชนิด ดังแสดงในรูปที่ 2.3 [4]

ข้อบกพร่องเหล่านี้เกิดจากกระบวนการที่แตกต่างกันคือกระบวนการเจาะและกระบวนการกัดรวมถึงมีลักษณะการเกิดข้อบกพร่องที่แตกต่างกันได้แก่ลักษณะทองแดงที่ขาดไป (Missing copper) และทองแดงที่เกินขึ้นมา (Excess copper) ดังแสดงในตารางที่ 2.1 [5]



รูปที่ 2.3 ตัวอย่างของข้อบกพร่องมาตรฐานบนแผ่นวงจรพิมพ์

ตารางที่ 2.1 การจำแนกประเภทของข้อบกพร่องตามกระบวนการและลักษณะการเกิดข้อบกพร่อง

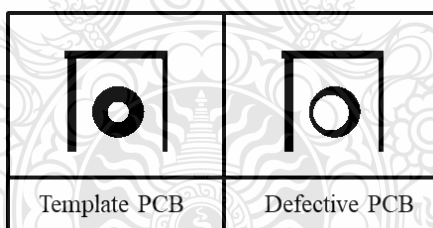
ลำดับ	ชนิดของข้อบกพร่อง	กระบวนการ		ลักษณะการเกิดข้อบกพร่อง	
		เจาะ	กัด	ทองแดงขาด	ทองแดงเกิน
1	Breakout	✓	-	-	-
2	Pin hole	-	✓	✓	-
3	Open circuit	-	✓	✓	-
4	Under etch	-	✓	-	✓
5	Mouse bite	-	✓	✓	-
6	Missing conductor	-	✓	✓	-
7	Spur	-	✓	-	✓
8	Short circuit	-	✓	-	✓
9	Wrong size hole	✓	-	-	-

ตารางที่ 2.1 การจำแนกประเภทของข้อบกพร่องตามกระบวนการและลักษณะการเกิดข้อบกพร่อง
(ต่อ)

ลำดับ	ชนิดของข้อบกพร่อง	กระบวนการ		ลักษณะการเกิดข้อบกพร่อง	
		เจาะ	กัด	ทองแดงขาด	ทองแดงเกิน
10	Conductor too close	-	✓	✓	✓
11	Spurious copper	-	✓	-	✓
12	Excessive short	-	✓	-	✓
13	Missing hole	✓	-	-	-
14	Over etch	-	✓	✓	-

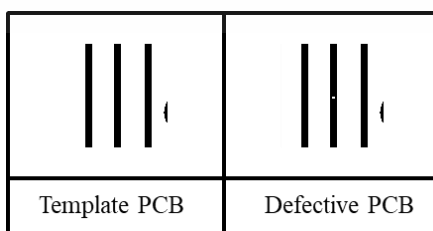
จากรูปที่ 2.3 และตารางที่ 2.1 ทำให้สามารถแสดงรายละเอียดของข้อบกพร่องที่รู้จักทั้ง 14 ชนิด ดังต่อไปนี้

1. ข้อบกพร่อง Breakout เกิดจากกระบวนการเจาะที่ผิดตำแหน่ง โดยเกิดขึ้นบริเวณ Hole pad รอบรูเจาะ ทำให้ทองแดงของ Hole pad บางส่วนขาดหายไปและบางส่วนเกินออกมา ลักษณะของข้อบกพร่อง Breakout แสดงในรูปที่ 2.4



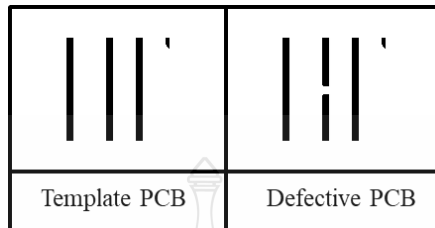
รูปที่ 2.4 ลักษณะของข้อบกพร่อง Breakout

2. ข้อบกพร่อง Pinhole เกิดจากกระบวนการกัด โดยเกิดขึ้นได้ทั้งบริเวณ Hole pad และบนลายวงจร ทำให้ทองแดงบางส่วนขาดหายไปเหมือนกับรูเข็มขนาดเล็ก ลักษณะของข้อบกพร่อง Pinhole แสดงในรูปที่ 2.5



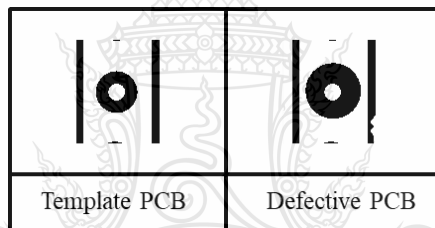
รูปที่ 2.5 ลักษณะของข้อบกพร่อง Pinhole

3. ข้อบกพร่อง Open circuit เกิดจากกระบวนการกัด โดยเกิดขึ้นบนลายวงจร ทำให้ทองแดงบางส่วนขาดหายไปและส่งผลให้ลายวงจรขาดออกจากกัน ลักษณะของข้อบกพร่อง Open circuit แสดงในรูปที่ 2.6



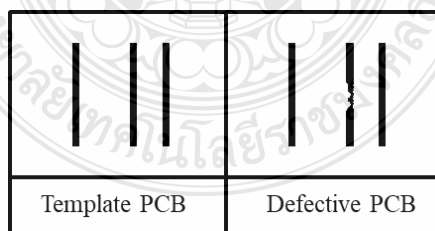
รูปที่ 2.6 ลักษณะของข้อบกพร่อง Open circuit

4. ข้อบกพร่อง Under etch เกิดจากกระบวนการกัดที่น้อยเกินไป ทำให้มีทองแดงเหลืออยู่ด้านข้างของลายวงจรและส่งผลให้ลายวงจรมีขนาดใหญ่ขึ้น ลักษณะของข้อบกพร่อง Under etch แสดงในรูปที่ 2.7



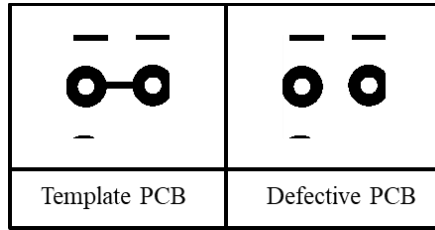
รูปที่ 2.7 ลักษณะของข้อบกพร่อง Under etch

5. ข้อบกพร่อง Mouse bite เกิดจากกระบวนการกัด ทำให้ทองแดงบางส่วนขาดหายไปเหมือนกับรอยหนูแทะ โดยมีรูปร่างที่ไม่สมมาตรกันทั้งแนวตั้งและแนวนอน ลักษณะของข้อบกพร่อง Mouse bite แสดงในรูปที่ 2.8



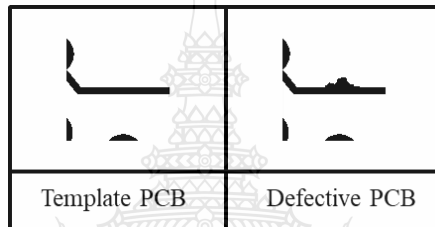
รูปที่ 2.8 ลักษณะของข้อบกพร่อง Mouse bite

6. ข้อบกพร่อง Missing conductor เกิดจากกระบวนการกัด ทำให้ทองแดงในส่วนของลายวงจรบางเส้นขาดหายไปและส่งผลให้ลายวงจรขาดออกจากกันคล้ายกับข้อบกพร่อง Open circuit แต่มีขนาดใหญ่กว่า ลักษณะของข้อบกพร่อง Missing conductor แสดงในรูปที่ 2.9



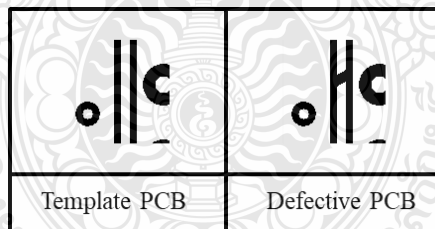
รูปที่ 2.9 ลักษณะของข้อบกพร่อง Missing conductor

7. ข้อบกพร่อง Spur เกิดจากกระบวนการกัด ทำให้ทองแดงบางส่วนเกินออกมาเหมือนกับการเกิดสปอร์ โดยมีรูปร่างที่ไม่สมมาตรกันทั้งแนวตั้งและแนวนอน ลักษณะของข้อบกพร่อง Spur แสดงในรูปที่ 2.10



รูปที่ 2.10 ลักษณะของข้อบกพร่อง Spur

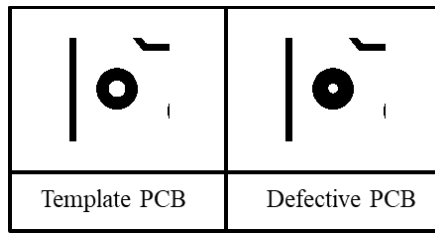
8. ข้อบกพร่อง Short circuit เกิดจากกระบวนการกัด ทำให้ทองแดงบางส่วนเกินออกมาและไปเชื่อมต่อกับลายวงจรส่วนอื่น ส่งผลให้เกิดการลัดวงจร ลักษณะของข้อบกพร่อง Short circuit แสดงในรูปที่ 2.11



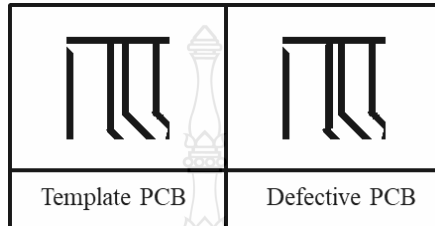
รูปที่ 2.11 ลักษณะของข้อบกพร่อง Short circuit

9. ข้อบกพร่อง Wrong size hole เกิดจากกระบวนการเจาะที่ผิดขนาด โดยเกิดขึ้นบริเวณ Hole pad รอบรูเจาะ ทำให้ทองแดงของ Hole pad บางส่วนขาดหายไปหรือเกินออกมา ลักษณะของข้อบกพร่อง Wrong size hole แสดงในรูปที่ 2.12

10. ข้อบกพร่อง Conductor too close เกิดจากกระบวนการกัด ทำให้ลายวงจรบางส่วนเคลื่อนที่ไปจากตำแหน่งเดิม ลักษณะของข้อบกพร่อง Conductor too close แสดงในรูปที่ 2.13

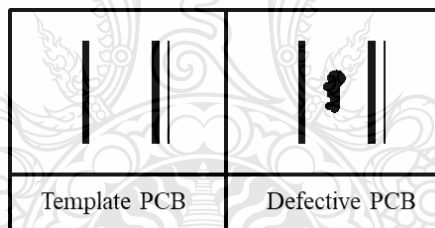


รูปที่ 2.12 ลักษณะของข้อบกพร่อง Wrong size hole



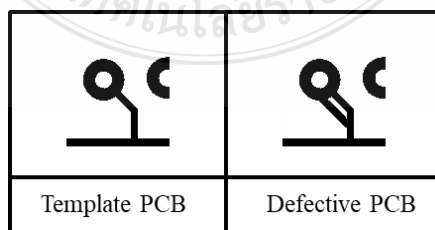
รูปที่ 2.13 ลักษณะของข้อบกพร่อง Conductor too close

11. ข้อบกพร่อง Spurious copper เกิดจากกระบวนการกัด ทำให้ทองแดงบางส่วนเกินออกมา โดยเกิดขึ้นในบริเวณที่ไม่ใช่ลายวงจร ถึงแม้ว่าจะไม่เกิดผลเสียต่อการนำ PCB ไปใช้งานแต่ก็ทำให้ลายวงจรไม่สวยงาม ลักษณะของข้อบกพร่อง Spurious copper แสดงในรูปที่ 2.14



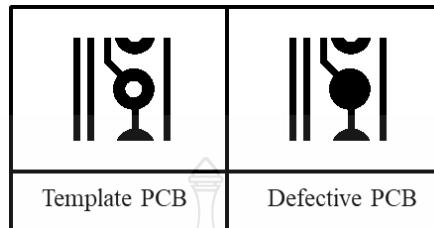
รูปที่ 2.14 ลักษณะของข้อบกพร่อง Spurious copper

12. ข้อบกพร่อง Excessive short เกิดจากกระบวนการกัด ทำให้ทองแดงบางส่วนเกินออกมาและไปเชื่อมต่อกับลายวงจรส่วนอื่นที่เป็นเส้นเดียวกัน ถึงแม้ว่าจะไม่เกิดผลเสียต่อการนำ PCB ไปใช้งานแต่ก็ทำให้ลายวงจรไม่สวยงาม ลักษณะของข้อบกพร่อง Excessive short แสดงในรูปที่ 2.15



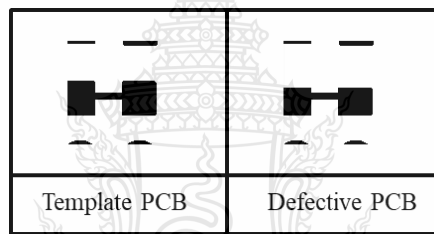
รูปที่ 2.15 ลักษณะของข้อบกพร่อง Excessive short

13. ข้อบกพร่อง Missing hole เกิดจากกระบวนการเจาะที่ผิดพลาดหรือไม่ได้ทำการเจาะรู โดยเกิดขึ้นบริเวณ Hole pad ทำให้ทองแดงของ Hole pad ที่ตำแหน่งรูเจาะเกินออกมา ลักษณะของ ข้อบกพร่อง Missing hole แสดงในรูปที่ 2.16



รูปที่ 2.16 ลักษณะของข้อบกพร่อง Missing hole

14. ข้อบกพร่อง Over etch เกิดจากกระบวนการกัดที่มากเกินไป ทำให้ทองแดงบางส่วนขาดหายไปและส่งผลให้ลายวงจรมีขนาดเล็กลง ลักษณะของข้อบกพร่อง Over etch แสดงในรูปที่ 2.17



รูปที่ 2.17 ลักษณะของข้อบกพร่อง Over etch

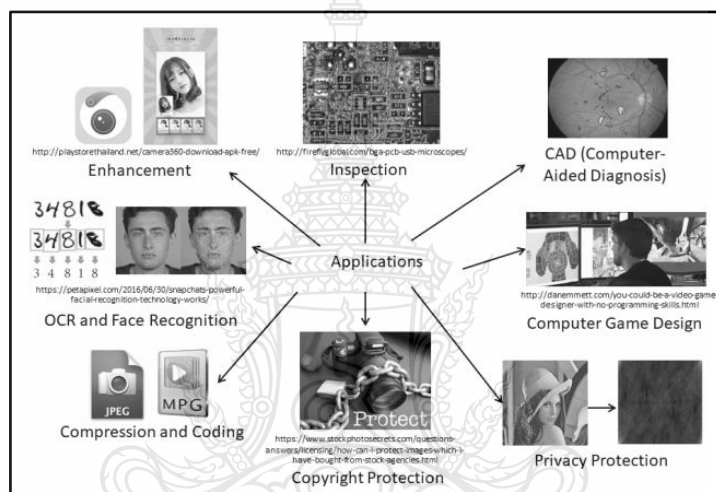
ในการตรวจจับและจำแนกข้อบกพร่องของ PCB เหล่านี้สามารถทำได้ทั้งวิธีการตรวจสอบด้วยสายตามนุษย์ (Manual inspection) และการตรวจสอบแบบอัตโนมัติ (Automatic inspection) แต่เนื่องจากข้อดีหลาย ๆ ประการของการตรวจสอบด้วยสายตามนุษย์ เช่น ความล่าช้า ความคลาดเคลื่อน ความยุ่งยากของลายวงจร รวมไปถึงค่าใช้จ่ายที่ค่อนข้างสูง ทำให้อุตสาหกรรมการผลิตชิ้นส่วนอิเล็กทรอนิกส์หันมาใช้ในการตรวจสอบแบบอัตโนมัติแทน และหนึ่งในการตรวจสอบแบบอัตโนมัติที่นิยมใช้ก็คือระบบการมองเห็นของเครื่องจักรโดยใช้หลักการประมวลผลภาพ

2.2 การมองเห็นของเครื่องจักรและการประมวลผลภาพ

ระบบการมองเห็นของเครื่องจักร (Machine vision) หรืออาจเรียกว่าระบบวิชัน (Vision System) เป็นระบบที่ถูกออกแบบขึ้นมาเพื่อที่จะใช้ในการตรวจสอบคุณภาพการผลิต เนื่องจากระบบ Machine Vision นี้สามารถตรวจเช็คได้เที่ยงตรงและแม่นยำกว่าการใช้คนตรวจสอบความผิดพลาดของผลิตภัณฑ์ และมีความละเอียดแม่นยำกว่าสายตามนุษย์ สามารถควบคุมคุณภาพได้สม่ำเสมอต่างจากสายตามนุษย์ที่อาจมีความเหนื่อยล้า เป็นสาเหตุทำให้เกิดความผิดพลาดได้ นอกจากนี้ระบบ Machine

Vision ยังสามารถตรวจเช็คคุณภาพชิ้นงานได้ตั้งแต่ช่วงแรกๆ ของกระบวนการผลิตซึ่งบางกระบวนการ คนไม่สามารถเข้าไปในพื้นที่นั้น ๆ ได้ ทำให้สามารถคัดแยกชิ้นงานที่เสียออกแต่เนิ่น ๆ ไม่ต้องปล่อยให้เสร็จสิ้นกระบวนการถึงจะมีการตรวจเช็ค

การประมวลผลภาพ (Image processing) เป็นวิธีการที่นิยมใช้ในการแยกแยะและจดจำวัตถุต่างๆ โดยเฉพาะในระบบ Machine Vision ที่ใช้ในงานอุตสาหกรรม การประมวลผลภาพเป็นการนำภาพมาประมวลผลหรือคิดคำนวณด้วยคอมพิวเตอร์เพื่อให้ได้ข้อมูลที่เราต้องการทั้งในเชิงคุณภาพและปริมาณ โดยมีวิธีการที่ใช้กันอยู่หลายรูปแบบมาก อย่างไรก็ตามในหัวข้อนี้จะนำเสนอเทคโนโลยีที่ใช้หรือเกี่ยวข้องกับงานวิจัยนี้เท่านั้น ได้แก่ การประมวลผลขั้นต้น การแบ่งส่วนภาพ และการจำแนกภาพ



รูปที่ 2.18 ตัวอย่างการประมวลผลภาพ [6]

2.2.1 การประมวลผลขั้นต้น

การประมวลผลขั้นต้น (Pre-processing) หมายถึง กระบวนการที่กระทำกับข้อมูลดิบที่ได้รับมาเพื่อปรับเปลี่ยนข้อมูลให้อยู่ในรูปแบบที่เหมาะสมที่จะนำไปวิเคราะห์ต่อไป โดยมีขั้นตอนต่าง ๆ ที่สำคัญ ดังนี้ การได้มาของภาพ (Image acquisition) เป็นการนำเอาภาพจากเซนเซอร์ (Sensor) และแปลงสัญญาณภาพให้เป็นภาพดิจิทัล (Digital Image) การเลือกบริเวณภาพที่สนใจ (Region of Interest: ROI) เป็นการเลือกบริเวณภาพซึ่งอาจจะเป็นบริเวณใดภายในภาพก็ได้ เพื่อนำภาพเฉพาะส่วนดังกล่าวมาประมวลผลหรือเปลี่ยนแปลงภาพตามต้องการ การแปลงเป็นระดับสีเทา (Gray scale conversion) จะทำการเปลี่ยนภาพที่รับเข้ามาในขั้นตอนแรกซึ่งเป็นภาพที่อยู่ในระบบปริภูมิสีแบบ RGB ให้เป็นภาพระดับสีเทา (Grayscale) เพื่อทำให้สามารถวิเคราะห์ภาพได้ง่ายขึ้น การลดสัญญาณรบกวน (Noise reduction) คือการลดสัญญาณรบกวนบนภาพทำได้โดยการออกแบบตัวกรองเพื่อกำจัดสัญญาณรบกวน แล้วใช้ตัวกรองที่ออกแบบมากระทำกับภาพที่รับเข้ามาผ่านกระบวนการประมวลผล

ภาพ การกรองภาพทำให้ภาพมีคุณภาพดีขึ้นก่อนที่จะนำไปใช้ในการประมวลผลต่อไป การปรับปรุงภาพ (Image enhancement) เป็นกระบวนการในการเปลี่ยนแปลงข้อมูลภาพเพื่อที่จะสร้างภาพที่เน้นรายละเอียดที่ต้องการ หรือปรับพิสัยของโทนแสงที่ต้องการเมื่อเปรียบเทียบกับข้อมูลหรือรายละเอียดอื่น ๆ ของภาพ การลงทะเบียนภาพ (Image registration) เป็นกระบวนการในการแปลงชุดข้อมูลที่แตกต่างกันให้เป็นระบบพิกัดเดียวกัน โดยข้อมูลอาจจะเป็นข้อมูลภาพหลาย ๆ ภาพที่ได้มาจากเซ็นเซอร์ที่ต่างกัน เวลาที่ต่างกัน ความลึกที่ต่างกัน หรือมุมมองที่แตกต่างกัน ด้วยสาเหตุนี้ทำให้การลงทะเบียนภาพเป็นสิ่งที่จำเป็นเพื่อให้สามารถเปรียบเทียบหรือรวมข้อมูลที่ได้จากการวัดที่แตกต่างกันเหล่านี้ สำหรับพื้นฐานของการลงทะเบียนภาพแบบ Linear transformation นั่นก็คือ การเลื่อนภาพ (Image translation) การหมุนภาพ (Image rotation) การย่อหรือขยายภาพ (Image scaling) ซึ่งอาจจะเรียกรวมกันว่าการแปลงสัมพรรค (Affine transform) โดยจะทำการประมวลผลในระบบพิกัด x และ y การใช้เทคนิคพื้นฐานเหล่านี้ภาพจะไม่มี การเปลี่ยนรูปแต่จะมีเพียงการเปลี่ยนแปลงลักษณะทางกายภาพ (Geometric) เท่านั้น

2.2.2 การแบ่งส่วนภาพ

การแบ่งส่วนภาพ (Image segmentation) จะทำให้สามารถแยกข้อมูลภาพของส่วนที่ต้องการออกมาได้ (ข้อมูลที่มีลักษณะเหมือนกับข้อมูลตัวอย่าง) วิธีการพื้นฐานสำหรับการแบ่งส่วนภาพคือการพิจารณา Image amplitude (ได้แก่การพิจารณาความสว่างของภาพสำหรับภาพแบบ Gray scale และความแตกต่างของสีสำหรับภาพสี) นอกจากนี้ขอบของภาพและลักษณะของ Texture ก็เป็นองค์ประกอบหนึ่งที่จะทำให้สามารถทำการแบ่งส่วนภาพได้สะดวกยิ่งขึ้น วิธีการแบ่งส่วนภาพมีอยู่หลากหลายวิธี ตัวอย่างเช่น การแบ่งส่วนภาพด้วยขนาด, การแบ่งส่วนภาพด้วยพื้นที่ และการแบ่งส่วนภาพด้วยขอบ เป็นต้น

2.2.2.1 การแบ่งส่วนภาพด้วยขนาด (Amplitude segmentation) เป็นวิธีการแยกองค์ประกอบของภาพโดยดูจากความเหมือนกันของคุณสมบัติของพิกเซลภายในพื้นที่เพียงอย่างเดียว เช่นวิธีการ Intensity thresholding เป็นการแยกแยะวัตถุจากฉากหลังโดยดูจาก Intensity ของพิกเซลเป็นหลัก ข้อดีของวิธีการนี้คือมีขั้นตอนในการทำงานที่ง่ายไม่ซับซ้อนทำงานได้รวดเร็ว แต่ข้อเสียคือไม่สามารถใช้กับภาพที่มีสัญญาณรบกวนมาก หรือภาพที่มีความสว่างไม่สม่ำเสมอ การแบ่งส่วนภาพด้วย Amplitude มีอยู่ด้วยกันหลายวิธี เช่น Bilevel Luminance Thresholding หรือ Multilevel Luminance Thresholding เป็นต้น

2.2.2.2 การแบ่งส่วนภาพด้วยพื้นที่ (Region segmentation) เป็นวิธีการแยกองค์ประกอบของภาพโดยดูจากตำแหน่งของพิกเซลและความเหมือนกันของคุณสมบัติของพิกเซลภายในพื้นที่เป็นหลัก โดยถ้าพิกเซลที่อยู่ติดกันและมีคุณสมบัติเหมือนกันจะถูกจัดให้เข้ากลุ่มเดียวกัน ข้อดีของการทำ

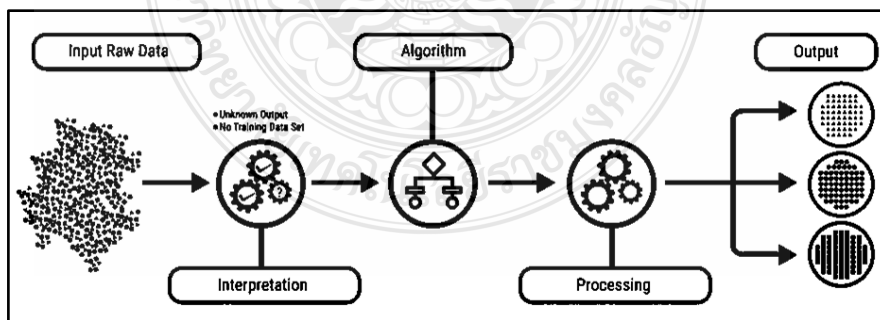
เช่นนี้จะได้พื้นที่ต่อเนื่องกัน ปัญหาอุปสรรคของวิธีการนี้คือ การกำหนดกฎเกณฑ์ที่เหมาะสมที่ใช้ตรวจสอบว่าพิกเซลนั้นที่มีคุณสมบัติเหมือนกัน เพื่อใช้ในการรวมกลุ่มเข้าด้วยกันทำได้ยาก และวิธีการเหล่านี้อาศัยอัลกอริทึมและโครงสร้างข้อมูลที่ซับซ้อนกันทำให้การประมวลผลใช้เวลานาน การแบ่งส่วนภาพด้วย Region มีอยู่ด้วยกันหลายวิธี เช่น Region Growing, Merge region, Split region หรือ Split and merge เป็นต้น

2.2.2.3 การแบ่งส่วนภาพด้วยขอบ (Edge segmentation) เป็นวิธีการแยกองค์ประกอบของภาพ โดยอาศัยความไม่ต่อเนื่องของคุณสมบัติของพิกเซลที่บริเวณขอบของวัตถุ ดังนั้นวิธีการนี้จึงมุ่งที่จะตรวจหาขอบของวัตถุเป็นข้อดีของวิธีการนี้คือมีความรวดเร็วในการประมวลผล เพราะวิธีการนี้ใช้เฉพาะข้อมูลบริเวณขอบของวัตถุเท่านั้นส่วนข้อเสียของวิธีการนี้ คือผลลัพธ์ที่ได้จะอยู่ในรูปของ “ขอบ” ของวัตถุซึ่งอาจจะต้องผ่านกระบวนการอื่นอีกต่อหนึ่ง จึงสามารถใช้งานได้ นอกจากนี้ขอบของวัตถุที่ได้อาจไม่ต่อเนื่องถ้าวัตถุมีสีที่ไม่สม่ำเสมอ ตัวอย่างเช่น วิธีการ Edge Linking เป็นต้น [7]

2.2.3 การจำแนกภาพ

การจำแนกภาพ (Image Classification) เป็นการวิเคราะห์เชิงสถิติเพื่อคัดแยกกลุ่มของข้อมูล (จุดภาพ) ที่มีคุณสมบัติทางแสงที่เหมือนกันหรือใกล้เคียงกันให้เป็นกลุ่มข้อมูลประเภท (Class) เดียวกัน และคัดแยกกลุ่มข้อมูลที่มีคุณสมบัติทางแสงต่างกันให้ออกจากกัน โดยการจำแนกภาพสามารถแบ่งได้เป็น 2 วิธีการคือ Unsupervised Classification และ Supervised Classification

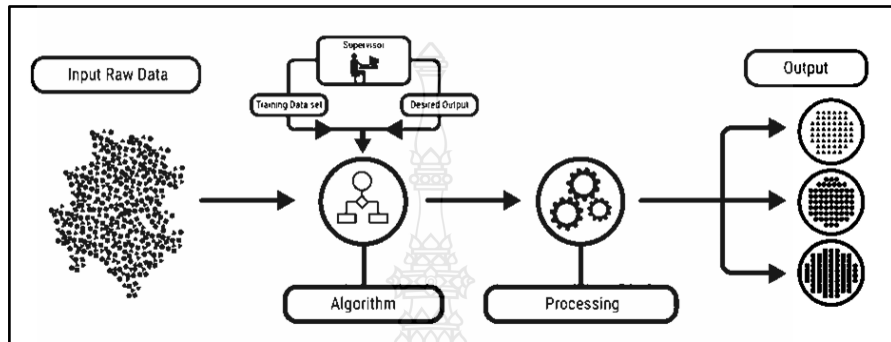
2.2.3.1 การจำแนกข้อมูลแบบไม่กำกับดูแล (Unsupervised Classification) เป็นการจำแนกข้อมูลจากการหาความสว่างของจุดภาพที่ใกล้เคียงกันซึ่งไม่ต้องใช้ความรู้ในการจำรูปแบบเลยทำให้การแบ่งแยกรายละเอียดเป็นกลุ่ม ๆ (Cluster) จึงขึ้นอยู่กับค่าความสว่างของภาพ ตัวอย่างเช่น Clustering, Dimensionality Reduction, Association Analysis, Hidden Markov Model เป็นต้น



รูปที่ 2.19 กระบวนการ Unsupervised Classification [8]

2.2.3.2 การจำแนกข้อมูลแบบกำกับดูแล (Supervised Classification) เป็นการจำแนกรายละเอียดที่ต้องการรู้คุณสมบัติของวัตถุว่าเป็นเช่นใดเพื่อนำมาใช้เลือกกลุ่มข้อมูลตัวอย่างที่เป็นค่า

ความสว่างของจุดภาพซึ่งเป็นตัวแทนของรายละเอียดนั้นในการจำแนกข้อมูล โดยจุดภาพที่เป็นตัวแทนของกลุ่มตัวอย่างนั้นเรียกว่า กลุ่มตัวอย่าง (Training Fields, Training Areas หรือ Spectral Signatures) ดังนั้นจะเห็นได้ว่าวิธีการจำแนกข้อมูลแบบกำกับดูแลจะเป็นการกำหนดกลุ่มตัวอย่างที่เป็นตัวแทนของรายละเอียดก่อนการจำแนก ตัวอย่างเช่น Classification, Regression, Decision Trees, Random Forests เป็นต้น [12]



รูปที่ 2.20 กระบวนการ Supervised Classification [8]

สำหรับการจำแนกประเภทข้อบกพร่อง PCB ในงานวิจัยนี้จะประยุกต์ใช้วิธีการจำแนกข้อมูลแบบกำกับดูแลเพื่อออกแบบอัลกอริทึมใหม่โดยใช้เครื่องมือทางการประมวลผลภาพต่าง ๆ

2.3 เครื่องมือทางการประมวลผลภาพ

จุดประสงค์หลักของการประมวลผลภาพคือการปรับปรุงเปลี่ยนแปลงลักษณะของภาพ ดังนั้นเครื่องมือทางการประมวลผลภาพ (Image processing tools) จึงได้ถูกสร้างขึ้นจำนวนมากเพื่อช่วยเหลือนักวิจัยในการวิเคราะห์ชุดปพลิเคชัน ชุดเครื่องมือ ฟังก์ชัน และแอปพลิเคชันสำหรับการประมวลผลภาพที่ครอบคลุม อย่างไรก็ตามในหัวข้อนี้จะกล่าวถึงเครื่องมือทางการประมวลผลภาพที่ใช้หรือเกี่ยวข้องกับงานวิจัยนี้เท่านั้น ได้แก่ การแปลงฮัฟแบบวงกลม การสร้างใหม่ทางสัญญาณวิทยา และการตัดฉากส่วนประกอบที่เชื่อมต่อ

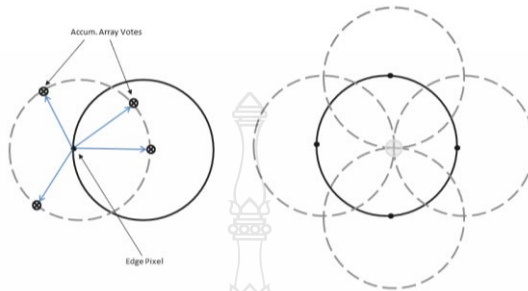
2.3.1 การแปลงฮัฟแบบวงกลม

การแปลงฮัฟแบบวงกลม (Circular Hough Transform: CHT) เป็นวิธีการตรวจจับภาพวัตถุที่เป็นวงกลมเพื่อค้นหาตำแหน่งและขนาดของวงกลมโดยใช้แต่ละพิกเซล x_i, y_i ในขอบภาพของวัตถุแทนตำแหน่งของจุดศูนย์กลางวงกลมที่มีรัศมี r_i ดังสมการที่ 2.1 และผลจากการวาดวงกลมรัศมี r_i ผ่านพิกัด (x, y) จะถูกสะสมค่าในเมทริกซ์สามมิติ $A(x, y, r_i)$ ซึ่งบ่งบอกถึงจำนวนเส้นรอบวงกลมขนาดต่าง ๆ ที่วาดผ่านพิกัดใด ๆ ภายในภาพ และสามารถระบุตำแหน่งของวัตถุที่เป็นวงกลมได้จากพิกัดที่เกิดจากการซ้อนทับกันมากที่สุด [9]

$$(x - x_i)^2 + (y - y_i)^2 = r_i^2 \quad (2.1)$$

เมื่อ $x = x_i + r_i \cos(\theta)$ และ $y = y_i + r_i \sin(\theta)$

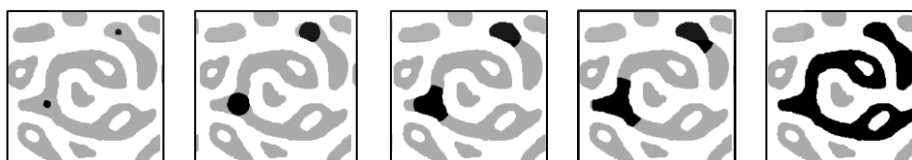
โดยที่ x_i และ y_i เป็นพิกัดของจุดศูนย์กลางวงกลม, r_i คือรัศมีของวงกลม และมุม θ กวาดผ่านเต็มช่วง 360 องศา รอบจุด (x, y) ตามเส้นรอบวงของวงกลม



รูปที่ 2.21 การแปลงฮัฟแบบวงกลม

2.3.2 การสร้างสัณฐานใหม่ทางวิทยา

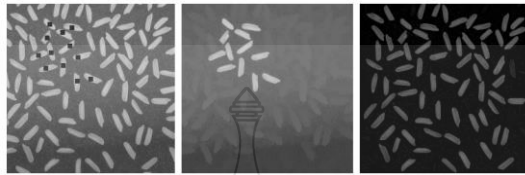
การสร้างสัณฐานใหม่ทางวิทยา (Morphological Reconstruction) เป็นพื้นฐานของอัลกอริธึมที่มีประโยชน์มากมาย ตัวอย่างเช่น การลบเส้นขอบ, การเติมช่องว่าง หรือการตรวจจับขอบเขตระดับต่ำสุดหรือสูงสุดในภาพระดับสีเทา หลักการของการสร้างสัณฐานวิทยาใหม่คือการทำซ้ำเงื่อนไขการขยาย (Dilation) หรือการกัดกร่อน (Erosion) จนกระทั่งเป็นไปตามเงื่อนไข Idempotence ซึ่งเงื่อนไข Idempotence เป็นสมบัติอย่างหนึ่งของการดำเนินการทางคณิตศาสตร์และวิทยาการคอมพิวเตอร์ซึ่งจะให้ผลลัพธ์เป็นค่าเดิมเสมอแม้ว่าจะกระทำการดำเนินการดังกล่าวกี่ครั้งก็ตาม เงื่อนไขการขยายเป็นผลจากการรวมกันของการขยายกับภาพหน้ากาก (Mask) โดยใช้ตัวดำเนินการทางตรรกศาสตร์ และกระทำซ้ำจนกว่าจะไม่มีอีกเกิดขึ้นอีก รูปที่ 2.22 แสดงขั้นตอนของการสร้างผิวใหม่ด้วยการขยายบนภาพแบบไบนารีคู่ ในภาพแรกหน้ากากแสดงด้วยส่วนที่เป็นสีเทาและมาร์กเกอร์ (Marker) แสดงด้วยส่วนที่เป็นสีดำ ต่อมาภาพที่สร้างใหม่ในแต่ละขั้นตอนจะแสดงด้วยส่วนที่เป็นสีดำ มาร์กเกอร์จะแพร่กระจายภายในหน้ากากเรื่อย ๆ จนกระทั่งเติมพื้นที่ที่เลือกจนเต็ม ตัวอย่างการประยุกต์ใช้งานแสดงในรูปที่ 2.23 และรูปที่ 2.24 [10,11]



รูปที่ 2.22 หลักการของ Morphological reconstruction



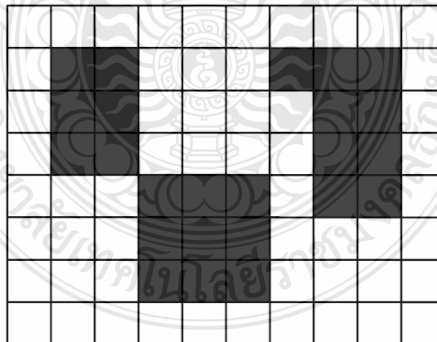
รูปที่ 2.23 การประยุกต์ใช้ Morphological reconstruction กับภาพแบบไบนารี



รูปที่ 2.24 การประยุกต์ใช้ Morphological reconstruction กับภาพระดับสีเทา

2.3.3 การติดฉลากส่วนประกอบที่เชื่อมต่อ

การติดฉลากส่วนประกอบที่เชื่อมต่อ (Connected Component Labeling: CCL) เป็นการพิจารณาว่าในภาพ ๆ หนึ่งอาจจะมีองค์ประกอบหรือวัตถุอยู่โดดเดี่ยวหรือมากกว่าหนึ่งวัตถุขึ้นไป โดยในแต่ละวัตถุอาจจะอยู่ในตำแหน่งที่ต่อเนื่องกัน ซ้อนทับกัน หรือแยกออกจากกัน ซึ่งในภาพที่เป็นแบบไบนารีจะมีการพิจารณาการเชื่อมต่อของวัตถุโดยพิจารณาจากพิกเซลที่เป็นส่วนของวัตถุกับพิกเซลรอบข้าง (Neighborhoods) ว่ามีการเชื่อมต่อกันหรือไม่ ซึ่งสามารถพิจารณาได้ทั้งใน 4 ทิศทาง (4-connected Neighborhoods) หรือใน 8 ทิศทาง (8-connected Neighborhoods) ซึ่งถ้าพิกเซลที่เป็นส่วนของวัตถุมีการเชื่อมต่อกันจะถือว่าเป็นวัตถุเดียวกัน แต่ถ้าพิกเซลที่เป็นส่วนของวัตถุแยกออกจากกันหรือไม่เชื่อมต่อกันจะกำหนดให้เป็นวัตถุที่ต่างกัน ดังแสดงในรูปที่ 2.25 [12]



รูปที่ 2.25 องค์ประกอบของภาพ

จากรูปที่ 2.25 นั้นกำหนดให้พิกเซลสีขาวเป็นส่วนของพื้นหลัง และพิกเซลสีดำเป็นส่วนของวัตถุ เมื่อนำภาพมาวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพทำให้สามารถระบุหมายเลข (Labeling of Connected Components) ให้กับองค์ประกอบของภาพได้ดังรูปที่ 2.26

0	0	0	0	0	0	0	0	0	0
0	1	1	0	0	0	3	3	3	0
0	1	1	0	0	0	0	3	3	0
0	1	1	0	0	0	0	3	3	0
0	0	0	2	2	2	0	3	3	0
0	0	0	2	2	2	0	0	0	0
0	0	0	2	2	2	0	0	0	0
0	0	0	0	0	0	0	0	0	0

รูปที่ 2.26 การระบุหมายเลขให้กับองค์ประกอบของภาพโดยพิจารณาใน 4 ทิศทาง

จากรูปที่ 2.26 เป็นผลการระบุหมายเลขให้กับองค์ประกอบของภาพ ซึ่งภาพจะประกอบไปด้วย 4 องค์ประกอบด้วยกัน โดยหมายเลข 0 แสดงส่วนของพื้นหลัง หมายเลข 1 แสดงวัตถุที่หนึ่ง หมายเลข 2 แสดงวัตถุที่สอง และหมายเลข 3 แสดงวัตถุที่สามของภาพ ซึ่งจากผลที่ได้ข้างต้นเป็นการวิเคราะห์องค์ประกอบที่เชื่อมต่อกันของภาพโดยใช้ 4 ทิศทาง แต่ถ้าเป็นการวิเคราะห์โดยใช้ 8 ทิศทาง จะทำให้ภาพข้างต้นประกอบไปด้วย 2 วัตถุเท่านั้น นั่นคือวัตถุที่หนึ่งและวัตถุที่สองเป็นวัตถุที่เชื่อมต่อกันในแนวทแยงจึงถือว่าเป็นวัตถุเดียวกัน

2.4 งานวิจัยที่เกี่ยวข้อง

ตั้งแต่อดีตจนถึงปัจจุบันมีการวิจัยเพื่อพัฒนาวิธีการตรวจสอบข้อบกพร่องของ PCB เป็นจำนวนมาก โดยส่วนใหญ่จะมุ่งเน้นไปทางการพัฒนาอัลกอริทึมในกระบวนการตรวจจับข้อบกพร่องและกระบวนการจำแนกประเภทข้อบกพร่องของ PCB สำหรับระบบตรวจสอบแบบอัตโนมัติโดยใช้หลักการประมวลผลภาพ ดังนั้นผู้วิจัยจึงได้ศึกษางานวิจัยที่เกี่ยวข้องเพื่อใช้เป็นแนวทางในการทำการวิจัย โดยมีรายละเอียดดังนี้

Wen-Yen Wu และคณะ (1996) นำเสนอการพัฒนากระบวนการมองเห็นอัตโนมัติสำหรับ PCB เพื่อทำการจำแนกชนิดของข้อบกพร่องที่ตรวจพบออกเป็น 7 ชนิด คือ Etching problem, Positive deformation, Short, Cut, Missing hole, Negative deformation และ Pin hole โดยใช้ดัชนี 3 ตัว ได้แก่ ประเภทของวัตถุที่ตรวจพบ, ความแตกต่างของหมายเลขวัตถุ และความแตกต่างในหมายเลขพื้นหลังระหว่างภาพที่ตรวจสอบและเทมเพลต ในขั้นตอนสุดท้ายได้ทำการทดลองกับภาพ PCB ขนาด 256 x 240 พิกเซล เพื่อแสดงประสิทธิภาพของระบบนี้ [3]

Madhav Moganti และคณะ (1996) ได้ทำการสำรวจอัลกอริทึมและเทคนิคสำหรับการตรวจสอบ PCB อัตโนมัติและนำเสนอแผนผังการจำแนกอัลกอริทึมเหล่านั้น โดยอัลกอริทึมต่าง ๆ จะถูกจัดกลุ่มตามการจำแนกประเภทนี้ ในการสำรวจจะเน้นที่การวิเคราะห์ภาพและกลยุทธ์การตรวจจับ

ข้อบกพร่องเป็นหลัก ซึ่งรวมไปถึงเทคนิคที่ล้ำสมัย นอกจากนี้ยังมีการนำเสนอบทสรุปของระบบการตรวจสอบ PCB เชิงพาณิชย์อีกด้วย [4]

Rudi Heriansyah และคณะ (2002) ทำการแบ่งส่วนภาพ PCB ออกเป็นรูปแบบพื้นฐาน เช่น Hole pad, Square pad, Rectangular pad และ Trace (Line) โดยใช้ตัวดำเนินการทางคณิตศาสตร์และทางสัญญาณวิทยา ในส่วนของการจำแนกประเภทจะทำการใส่รูปแบบเฉพาะลงในแต่ละ Window ซึ่งจะเรียกเทคนิคที่พัฒนาขึ้นนี้ว่า Windowing [5]

Noor Khaffah Khalid และคณะ (2008) ออกแบบอัลกอริทึมสำหรับจัดกลุ่มข้อบกพร่องมาตรฐาน 14 ชนิดที่พบบน PCB เปล่า โดยการใช้การดำเนินการทางประมวลผลภาพหลายอย่าง เช่น image subtraction, image adding, flood fill, ลอจิก XOR และลอจิก NOT จากผลการทดลองได้ข้อบกพร่องทั้งหมด 5 กลุ่ม ดังนี้ กลุ่มที่ 1 Missing hole และ Wrong size hole กลุ่มที่ 2 Conductor too close negative, Spurious copper, Excessive short, Underetch negative Spur, และ Short circuit กลุ่มที่ 3 Open circuit, Mouse bite, Overetch และ Conductor too close positive กลุ่มที่ 4 Underetch positive และกลุ่มที่ 5 Pinhole และ Breakout [13]

S.H. Indera Putera และ Z. Ibrahim (2010) พัฒนางานของ Khalid โดยเพิ่มจำนวนของการจัดกลุ่ม จาก 5 เป็น 7 กลุ่ม แต่ละกลุ่มประกอบด้วยข้อบกพร่อง 1 ถึง 4 ชนิด ด้วยการใช้อัลกอริทึมการแบ่งส่วนภาพทางสัญญาณวิทยาและทฤษฎีการประมวลผลภาพอย่างง่าย ผลการทดลองที่ได้สามารถจำแนกข้อบกพร่องทั้งหมด 13 จาก 14 ชนิดได้ [14]

Ajay Pal Singh Chauhan และ Sharat Chandra Bhardwaj (2011) ทำการตรวจจับข้อบกพร่องบน PCB เปล่า เริ่มตั้งแต่การเปรียบเทียบภาพ PCB ที่จะทำการตรวจสอบกับภาพ PCB มาตรฐานด้วยการใช้อัลกอริทึม Subtraction พื้นฐานเพื่อให้ได้บริเวณของข้อบกพร่อง จากนั้นใช้ฟังก์ชัน Particle Analysis ของ NI Vision Assistant วิเคราะห์ข้อบกพร่องในรูปแบบของพารามิเตอร์ เช่น พื้นที่, การวางแนว, พิกัด X และ Y เป็นต้น โดยมีข้อบกพร่องหลายชนิดที่สามารถตรวจจับได้ เช่น Etchings (Opens), Under-etchings (Shorts), Holes เป็นต้น [15]

Ismail Ibrahim และคณะ (2011) นำเสนออัลกอริทึมใหม่สำหรับตรวจจับและจำแนกชนิดของข้อบกพร่อง PCB เปล่า โดยการใช้การดำเนินการทั่วไปที่เป็นที่รู้จักกันดี เช่น image difference, image subtraction, image addition และ image comparison เพื่อจำแนกข้อบกพร่อง PCB 5 ชนิดที่แตกต่างกัน ได้แก่ Missing hole, Pin hole, Underetch, Short circuit และ Open circuit [16]

Ismail Ibrahim และคณะ (2012) นำเสนออัลกอริทึมสำหรับจำแนกชนิดของข้อบกพร่อง PCB โดยใช้ 7 ตัวดำเนินการทั่วไปที่เป็นที่รู้จักกันดีซึ่งประกอบด้วย image difference, image subtraction, image addition, counted image comparator, flood-fill และ labeling เพื่อจำแนก

ข้อบกพร่อง PCB 6 ชนิด ได้แก่ Missing hole, Wrong size, Open circuit, Short circuit, Pinhole, Underetch และ Mousebite ในอัลกอริทึมดังกล่าวได้ผสมผสานเทคนิค image registration และ thresholding เข้าไปด้วย เพื่อแก้ปัญหาการจัดตำแหน่งและความสว่างที่ไม่สม่ำเสมอ [17]

Siti Hazurah Indera Putera และคณะ (2012) พัฒนาการจัดกลุ่มข้อบกพร่องมาตรฐาน 14 ชนิด จาก 7 กลุ่มเป็น 11 กลุ่ม โดยใช้วิธีการวัดคุณสมบัติของภาพในการวัดขอบเขตภาพ และใช้เครื่องมือการประมวลผลภาพด้วย MATLAB [18]

Beant Kaur และคณะ (2014) นำเสนอเทคนิคในการตรวจจับและจำแนกชนิดของข้อบกพร่อง PCB โดยใช้วิธีการที่อยู่บนพื้นฐานของ image subtraction ในขั้นตอนแรกจะทำการตรวจจับข้อบกพร่อง และในขั้นตอนที่ 2 จะทำการจำแนกชนิดข้อบกพร่องที่แตกต่างกันด้วยการพัฒนาอัลกอริทึม handcraft ที่แยกกัน อัลกอริทึมนี้สามารถจำแนกข้อบกพร่องได้ 6 ชนิด ได้แก่ Missing hole, Over etching, Under etching, Wrong hole size, Missing conductor และ Break lines [19]

Prachi P. Londe และ S. A. Chavan (2014) นำเสนอระบบตรวจสอบข้อบกพร่อง PCB โดยอัลกอริทึมตรวจสอบหลักมุ่งเน้นไปที่การตรวจจับข้อบกพร่องและการจำแนกชนิดของข้อบกพร่อง โดยใช้เครื่องมือของ MATLAB ในขั้นตอนแรกทำการตรวจจับข้อบกพร่องโดยทำการเปรียบเทียบภาพ PCB มาจรรยาฐานกับภาพ PCB ที่ต้องการตรวจสอบด้วยการดำเนินการ image subtraction และขั้นตอนที่สองทำการจำแนกชนิดของข้อบกพร่องมาตรฐานทั้ง 14 ชนิดด้วยอัลกอริทึม K-NN ได้ความถูกต้องในการจำแนกอยู่ที่ 70-80 เปอร์เซ็นต์ [20]

Malge. P. S. และ Nadaf. R. S. (2014) ทำการจัดกลุ่มข้อบกพร่องมาตรฐาน 14 ชนิด ออกเป็น 7 กลุ่ม ด้วยการใช้อัลกอริทึมการแบ่งส่วนภาพทางสัญญาณวิทยาและทฤษฎีการประมวลผลภาพอย่างง่าย [21]

Shashikumar Vishwakarma และคณะ (2015) พัฒนาระบบการตรวจสอบ PCB โดยใช้การลงทะเบียนภาพทางเรขาคณิตเพื่อแปลงภาพที่ทำการวิจัยให้สอดคล้องกับภาพที่ใช้อ้างอิง ซึ่งช่วยให้ผลของการประมวลผลภาพมีความแม่นยำยิ่งขึ้น ใช้เทคนิค Minimum Threshold เพื่อกำหนดระดับ Threshold สำหรับภาพ Respective และใช้การกรองค่ามัธยฐานเพื่อแก้ปัญหาการส่องสว่างที่ไม่สม่ำเสมอ [22]

Mr.M. H. Thigale และคณะ (2017) ใช้วิธีการ normalized cross-correlation เพื่อตรวจจับข้อบกพร่อง และใช้อัลกอริทึมการแบ่งส่วนภาพทางสัญญาณวิทยาและทฤษฎีการประมวลผลภาพอย่างง่ายเพื่อจัดกลุ่มข้อบกพร่องมาตรฐาน 14 ชนิดออกเป็น 7 กลุ่ม อัลกอริทึมนี้ใช้การดำเนินการประมวลผลภาพที่เกี่ยวข้องกับ MATLAB เช่น image subtraction, ลอจิก XOR และ flood fill [23]

Ananthu Raj และ Sajeena A (2018) ได้เสนอวิธีการตรวจจับข้อบกพร่อง PCB โดยใช้การประมวลผลภาพสำหรับการใช้งานในอุตสาหกรรม เริ่มต้นด้วยการเปรียบเทียบภาพ PCB มาตรฐานกับภาพ PCB ที่จะทำการทดสอบ ด้วยการใช้อัลกอริทึม subtraction ง่าย ๆ เพื่อเน้นพื้นที่ที่มีปัญหาหลัก จากนั้นจะพิจารณาสัญญาณรบกวนในภาพ PCB เพื่อระดับความน่าพอใจในการตรวจจับข้อบกพร่องของวิธีการนี้ และสุดท้ายจะใช้การดำเนินการจำแนกข้อบกพร่องเพื่อวิเคราะห์แหล่งที่มาของข้อบกพร่องทั้งหมด 4 ชนิด ได้แก่ Missing hole, Pin hole, Short circuit และ Open circuit [24]

Juanhua Zhu และคณะ (2018) ศึกษาเทคโนโลยีการตรวจสอบข้อบกพร่องของ PCB โดยอาศัย wavelet denoising และการสกัดคุณลักษณะ เพื่อการตรวจจับและระบุข้อบกพร่องของ PCB อย่างรวดเร็ว ชนิดของข้อบกพร่องถูกกำหนดตามลักษณะและภาพข้างเคียงของข้อบกพร่อง ผลการวิจัยสามารถตรวจจับและระบุข้อบกพร่องได้ 5 ชนิด ได้แก่ Short circuit, Open circuit, Sag, Bulge และ Hole โดยพบว่าอัตราการตรวจจับข้อบกพร่องที่ได้คือ 100% และอัตราการรู้จำสูงกว่า 90% ซึ่งสามารถตอบสนองความต้องการในสายการผลิตของอุตสาหกรรมแบบเรียลไทม์ได้เป็นอย่างดี [25]

Peng Wei และคณะ (2018) พัฒนารูปแบบการจำแนกข้อบกพร่อง PCB โดยใช้โครงข่ายประสาทเทียม (CNN) ในการสอนโมเดลที่มีประสิทธิภาพ เริ่มต้นด้วยการใช้สัณฐานวิทยาในการแยกพื้นที่ของข้อบกพร่อง โดยทำการทดลองด้วยข้อมูล 1818 ภาพ ที่มีข้อบกพร่อง 6 ชนิด คือ Short circuit, Open circuit, Burr (Spur), Cavity (Pin hole), Deficit (Mouse bite) และ Missing hole หลังจากนั้นจึงทำการจำแนกชนิดของข้อบกพร่องด้วย CNN ผลลัพธ์ที่ได้ให้ความแม่นยำในการจำแนกสูงพอสมควร (95.7%) [26]

Linlin Zhang และคณะ (2018) นำเสนอแบบจำลอง multi-task CNN โดยเปลี่ยนการเรียนรู้แบบ multi-label ให้เป็นการเรียนรู้แต่ละ label หลาย ๆ งาน เพื่อทำการจำแนกแบบไบนารีด้วยการปรับแต่งฟังก์ชันการสูญเสีย และใช้สถาปัตยกรรม deep neural network เพื่อแก้ไขปัญหามulti-task นี้ ผลการทดลองด้วยวิธีการนี้สามารถจำแนกข้อบกพร่องได้ 6 ชนิด ได้แก่ Short circuit, Open circuit, Spurious, Copper, Mousebite และ Spur [27]

Weibo Huang และ Peng Wei (2018) ได้เผยแพร่ชุดข้อมูล PCB สังเคราะห์เพื่อใช้ในการลงทะเบียนภาพ, การตรวจจับภาพ และการจำแนกภาพ ซึ่งประกอบด้วย 1386 ภาพที่มีข้อบกพร่อง 6 ชนิด คือ Missing hole, Mouse bite, Open circuit, Short, Spur และ Spurious นอกจากนี้ยังนำเสนอวิธีการตรวจสอบข้อบกพร่องด้วยการใช้ end-to-end CNN ในการจำแนกชนิดของข้อบกพร่อง โดยวิธีการนี้จะทำการค้นหาข้อบกพร่องก่อนแล้วจึงจำแนกชนิดด้วย neural networks เพื่อแสดงประสิทธิภาพของชุดข้อมูล [28]

Li-Jie Zhao และคณะ (2019) นำเสนอวิธีการแบ่งส่วนของ Floc และ Filament อัตโนมัติ สำหรับภาพ Phase Contrast Microscopy (PCM) โดยใช้โครงสร้างการเรียนรู้เชิงลึก U-Net พร้อมกับการขยายข้อมูล และนำเสนอฟังก์ชันการสูญเสียที่รวมฟังก์ชัน Binary Cross Entropy (BCE) และค่าสัมประสิทธิ์ Dice เพื่อปรับปรุงความแม่นยำและความไวในการแบ่งส่วนภาพโดยใช้ตัวอย่างพื้นหน้า (Foreground) และพื้นหลัง (Background) ที่ไม่สมดุลกัน ประสิทธิภาพของอัลกอริทึมการแบ่งส่วนภาพได้รับการประเมินด้วยค่า Accuracy ค่า Precision ค่า Recall ค่า F-measure และค่า IoU ผลการทดลองเพื่อตรวจสอบวิธีการแบ่งส่วนภาพที่นำเสนอโมเดล U-Net พร้อมฟังก์ชันการสูญเสีย โดยรวมจะให้ผลลัพธ์ที่ดีกว่าโมเดล U-Net ที่มี BCE รวมไปถึงโมเดล FCN-VGG16 และวิธีการแบ่งส่วนภาพแบบดั้งเดิมด้วย [29]

M. H. Annaby และคณะ (2019) ทำการปรับปรุง Normalized Cross-Correlation (NCC) ที่มีความซับซ้อนต่ำเพื่อค้นหาตำแหน่งของวงจรรวม (ICs) ที่ขาดหายไปในการตรวจสอบแผ่นวงจรพิมพ์ (PCB) อัตโนมัติ โดยในโครงสร้างที่นำเสนอในภาพย่อ 2-D ในกระบวนการจับคู่จะถูกแปลงเป็นตัวบอกคุณสมบัติ 1-D ในขณะที่ภาพย่อที่กำลังประมวลผลอยู่จะถูกสแกนทั้งในแนวตั้งและแนวนอนร่วมกับการวัดทางสถิติเชิงพื้นที่ จากนั้นแปลงด้วย Discrete Cosine Transform (DCT) เพื่อสร้างตัวบอกคุณลักษณะ ผลการทดลองด้วย Template matching บนชุดข้อมูลภาพ PCB 32 ภาพ ทั้งบนเครื่องคอมพิวเตอร์และโทรศัพท์มือถือแสดงให้เห็นว่าโครงสร้างที่นำเสนอมีประสิทธิภาพที่ดีกว่า NCC แบบดั้งเดิมและดีกว่าวิธีการอื่น ๆ ที่เป็น 1-D ด้วย นอกจากนี้ค่า Signal-to-noise ที่วัดได้ยังแสดงให้เห็นถึงความแข็งแกร่งของเทคนิคที่นำเสนอต่อสัญญาณรบกวนอีกด้วย [30]

Runwei Ding และคณะ (2020) นำเสนอวิธีการ Multi-layer deep feature fusion เพื่อคำนวณหาความคล้ายคลึงกันระหว่างภาพ PCB ต้นแบบและภาพ PCB ที่มีข้อบกพร่อง แล้วทำการเปรียบเทียบวิธีการทั่วไปที่แบ่งการตรวจจับทั้งหมดออกเป็นสองส่วนอิสระต่อกันคือคุณลักษณะที่ออกแบบด้วยมือและการวัดความคล้ายคลึงกัน กับโมเดล end-to-end ของผู้เขียนที่ได้รับการออกแบบมาโดยรวมสองส่วนนี้เข้าด้วยกันเพื่อเสริมประสิทธิภาพของกันและกัน โดยขั้นตอนแรกจะใช้ Siamese network เป็นสถาปัตยกรรมหลักสำหรับการแยกคุณลักษณะของภาพทั้งคู่ จากนั้น Spatial pyramid pooling network จะรวมอยู่ในแผนผังคุณลักษณะของโมดูลการแปลงสัญญาณแต่ละโมดูลเพื่อหลอมรวมเวกเตอร์คุณลักษณะแบบหลายมาตราส่วน ในขั้นตอนสุดท้ายคุณลักษณะการฝังตัวและการวัดความคล้ายคลึงจะถูกรวมกันโดยใช้ Contrastive loss ในระหว่างกระบวนการฝึกสอน ผลการทดลองแสดงให้เห็นว่าแบบจำลองที่นำเสนอมีประสิทธิภาพที่ดีกว่าในการตรวจจับและค้นหาข้อบกพร่องที่ไม่รู้จักในภาพ PCB เปล่ามากกว่าวิธีการวัดความคล้ายคลึงแบบเดิม [31]

Bing Hu และ Jianhui Wang (2020) นำเสนอวิธีการตรวจจับภาพแบบเรียนรู้เชิงลึกสำหรับการตรวจจับข้อบกพร่องของ PCB วิธีนี้จะสร้างเครือข่ายใหม่บน Faster RCNN โดยใช้ ResNet50 ที่มี Feature Pyramid Networks เป็นวิธีการหลักสำหรับการแยกคุณสมบัติเพื่อให้ตรวจจับข้อบกพร่องขนาดเล็ก ๆ บน PCB ได้ดีขึ้น จากนั้นจะใช้ GARPN เพื่อทำนายจุดยึด (Anchor) และจุดผสาน (Merge) ของ Shuf eNetV2 ให้แม่นยำยิ่งขึ้น ผลการทดลองแสดงให้เห็นว่าวิธีนี้เหมาะสำหรับใช้ในการผลิตมากกว่าวิธีการตรวจจับข้อบกพร่องของ PCB อื่น ๆ [32]

แม้ว่านักวิจัยเหล่านี้ได้พัฒนาอัลกอริทึมต่าง ๆ ในการตรวจสอบและจำแนกข้อบกพร่องของ PCB ขึ้นมามากมาย แต่อย่างไรก็ตามอาจจะเนื่องมาจากข้อจำกัดของวิธีการต่าง ๆ ทำให้บางอัลกอริทึมสามารถจำแนกชนิดของข้อบกพร่องได้ไม่ครบทุกชนิด ในขณะที่เดียวกันบางอัลกอริทึมก็สามารถที่จะจำแนกประเภทข้อบกพร่องได้ทั้งหมดแต่มีความถูกต้องแม่นยำที่ไม่สูงนัก เช่นในงานวิจัยของ Prachi P. Londe และ S. A. Chavan [20] จะมีความแม่นยำอยู่ที่ 70-80 เปอร์เซ็นต์ เป็นต้น แต่เนื่องจากข้อบกพร่องของ PCB แต่ละประเภทเกิดขึ้นเนื่องจากสาเหตุและกระบวนการที่แตกต่างกัน การจำแนกประเภทข้อบกพร่องทั้งหมด 14 ชนิดด้วยความแม่นยำที่ค่อนข้างสูงจะสามารถช่วยระบุสาเหตุของข้อผิดพลาดที่เกิดขึ้นได้อย่างแท้จริง ด้วยเหตุผลเหล่านี้ส่งผลให้เทคโนโลยีตรวจสอบข้อบกพร่องของ PCB ยังไม่บรรลุประสิทธิภาพเพียงพอในการค้นหาสาเหตุที่แท้จริงของการเกิดข้อบกพร่อง PCB สำหรับกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์ในอุตสาหกรรมได้อย่างมีประสิทธิภาพ

งานวิจัยนี้จึงนำเสนออัลกอริทึมที่สามารถตรวจจับและจำแนกชนิดของข้อบกพร่องที่เกิดขึ้นบน PCB เปล่าแบบด้านเดียวหลังจากผ่านกระบวนการกัดและเจาะในกระบวนการผลิต โดยใช้วิธีการจัดกลุ่มแบบ Fuzzy C-means สำหรับการแบ่งส่วนภาพ ก่อนที่จะตรวจจับข้อบกพร่องทั้งหมดด้วยวิธีการ Image subtraction ในส่วนของการจำแนกประเภทข้อบกพร่องจะใช้การดำเนินการทางคณิตศาสตร์และตรรกศาสตร์ การแปลงฮัฟแบบวงกลม (CHT) การสร้างใหม่ทางสัญญาณวิทยา (MR) และการติดฉลากส่วนประกอบที่เชื่อมต่อ (CCL) โดยมีวัตถุประสงค์เพื่อจำแนกชนิดของข้อบกพร่องที่รู้จักทั้งหมด 14 ชนิด และให้ความแม่นยำอย่างน้อย 90 เปอร์เซ็นต์

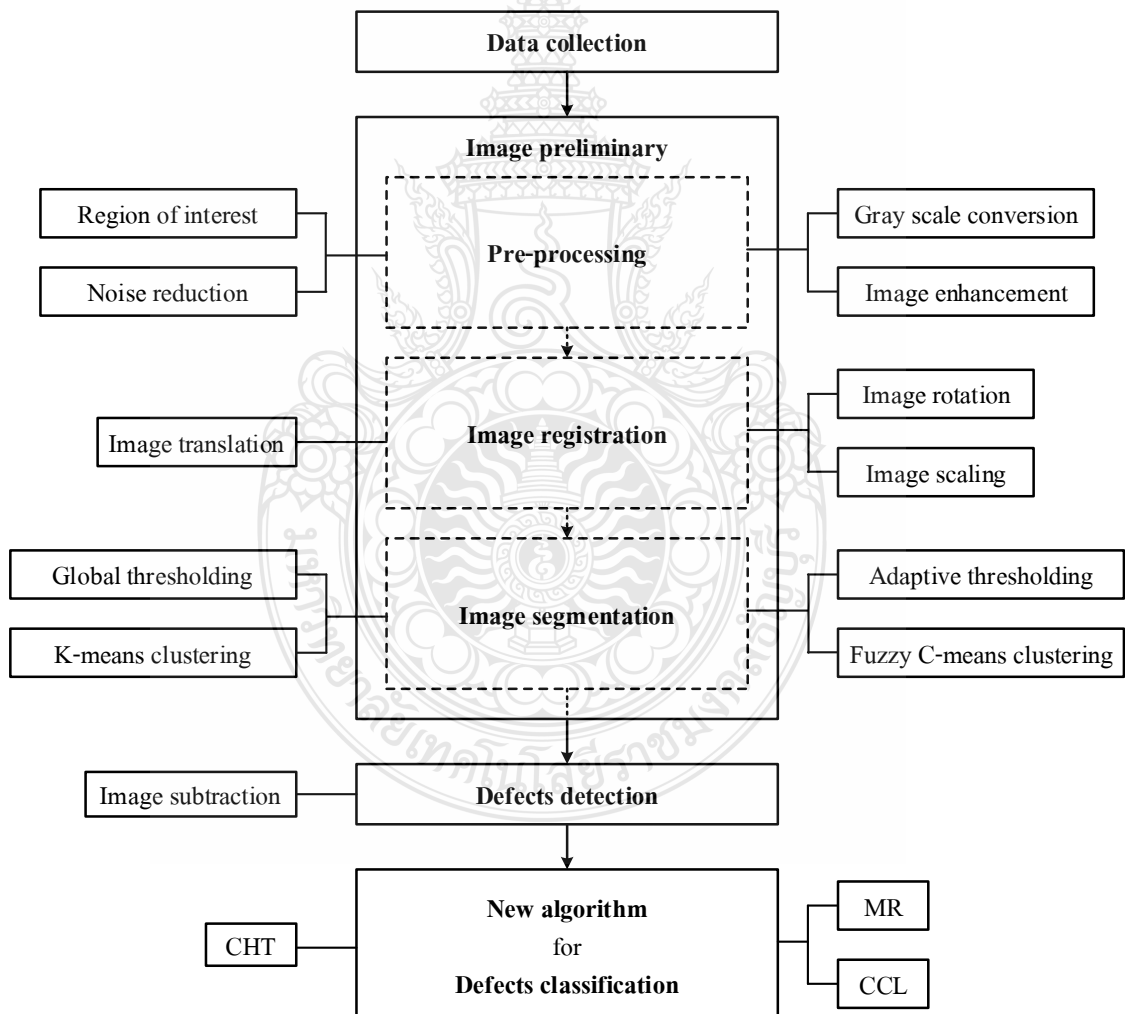
บทที่ 3

วิธีดำเนินการวิจัย

ในบทนี้จะกล่าวถึงขั้นตอนและวิธีดำเนินการวิจัยทั้งหมด ซึ่งประกอบด้วย การออกแบบระบบในภาพรวม การเก็บรวบรวมข้อมูล การประมวลผลภาพเบื้องต้น การตรวจจับข้อบกพร่อง การออกแบบอัลกอริทึมสำหรับจำแนกข้อบกพร่อง และหัวข้อสุดท้ายจะเป็นวิธีการทดสอบงานวิจัย

3.1 การออกแบบระบบในภาพรวม

ในการออกแบบระบบตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ของงานวิจัยนี้ แสดงในรูปที่ 3.1



รูปที่ 3.1 การออกแบบระบบในภาพรวม

3.2 การเก็บรวบรวมข้อมูล

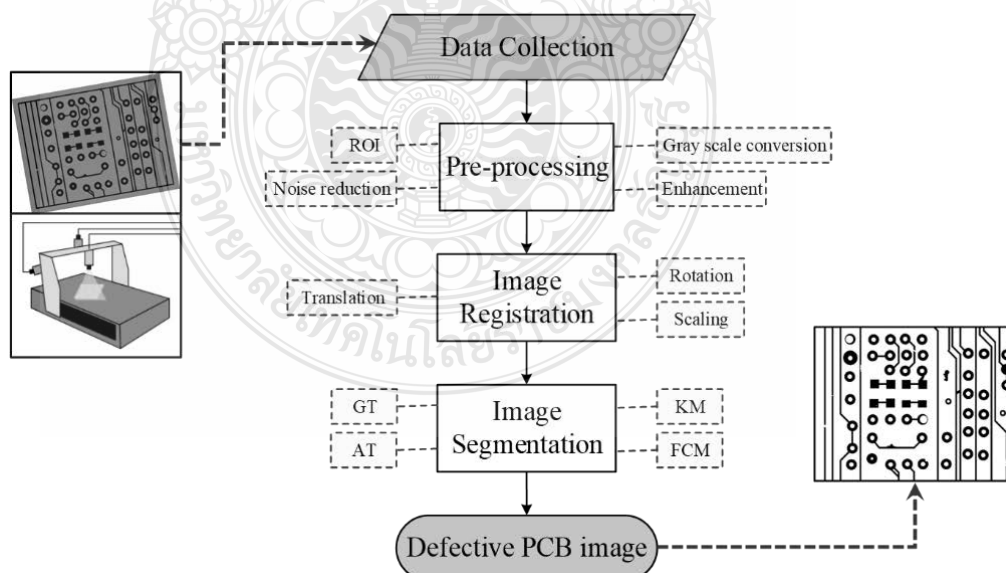
ในกระบวนการเก็บรวบรวมข้อมูล (Data collection) จะทำการเก็บและสร้างชุดของข้อมูลภาพขึ้นมาเพื่อใช้สำหรับการวิจัย จำนวน 3 ชุด โดยให้ชื่อว่า 1) InPCB 2) TemPCB และ 3) DefPCB

3.2.1 ชุดข้อมูล InPCB ได้จากการเก็บรวบรวมภาพแผ่น PCB เปล่าแบบ Single layer มาจากแหล่งข้อมูลต่าง ๆ เช่น ภาพถ่าย, ภาพในเอกสารและตำรา, ภาพจากสื่อออนไลน์ เป็นต้น จำนวน 1,000 ภาพ เพื่อใช้เป็นภาพ PCB อินพุตสำหรับทดสอบขั้นตอนการประมวลผลภาพเบื้องต้น

3.2.2 ชุดข้อมูล TemPCB ได้มาจากการเก็บรวบรวมภาพลายวงจรที่สร้างขึ้นด้วยโปรแกรมสำเร็จรูปต่าง ๆ จำนวน 100 ภาพ เพื่อนำมาใช้เป็นภาพ PCB ต้นแบบสำหรับกระบวนการตรวจจับข้อบกพร่อง

3.2.3 ชุดข้อมูล DefPCB จะได้มาจากการเก็บรวบรวมภาพลายวงจรที่มีข้อบกพร่องซึ่งเกิดจากกระบวนการกัดและเจาะ สำหรับแต่ละภาพ PCB ต้นแบบจะได้ภาพ PCB ที่มีข้อบกพร่อง จำนวน 100 ภาพ โดยบางภาพมีข้อบกพร่องชนิดเดียวและบางภาพมีข้อบกพร่องหลากหลายชนิดรวมอยู่ด้วยกัน เพื่อใช้เป็นภาพสำหรับทดสอบในกระบวนการตรวจจับข้อบกพร่องและกระบวนการจำแนกชนิดของข้อบกพร่อง รวมภาพ PCB ที่มีข้อบกพร่องทั้งหมด 10,000 ภาพ

3.3 การประมวลผลภาพเบื้องต้น



รูปที่ 3.2 ขั้นตอนการประมวลผลภาพเบื้องต้น

ภาพอินพุตในชุดข้อมูล InPCB ซึ่งเป็นภาพสี RGB จะถูกนำเข้าสู่ขั้นตอนของการประมวลผลภาพเบื้องต้น (Image Preliminary) แล้วทำการประมวลผลข้อมูลภาพที่รับเข้ามา เพื่อให้ได้ภาพ PCB ที่มีลักษณะทางกายภาพเหมือนภาพในชุดข้อมูล TemPCB โดยขั้นตอนการประมวลผลภาพเบื้องต้นจะประกอบไปด้วยขั้นตอนต่าง ๆ ได้แก่ การประมวลผลขั้นต้น (Pre-processing) การลงทะเบียนภาพ (Image registration) และการแบ่งส่วนภาพ (Image segmentation)

3.3.1 ขั้นตอนการประมวลผลขั้นต้น

ในขั้นตอนนี้จะทำการปรับปรุงข้อมูลภาพที่ได้รับมาให้อยู่ในรูปแบบที่เหมาะสมที่จะนำไปวิเคราะห์ต่อไป โดยแบ่งเป็นขั้นตอนย่อยดังต่อไปนี้

3.3.1.1 ขั้นตอนการเลือกบริเวณภาพที่สนใจ (ROI) จะทำการเลือกภาพเฉพาะในส่วนที่เป็นแผ่น PCB และตัดภาพในส่วนที่ไม่ต้องการทิ้งไปเพื่อให้ง่ายและมีความรวดเร็วในการประมวลผล

3.3.1.2 ขั้นตอนการแปลงเป็นภาพระดับสีเทา (Gray scale conversion) จะทำการแปลงภาพสี RGB แบบ 8 บิตที่รับเข้ามาให้เป็นภาพระดับสีเทาแบบ 8 บิต โดยใช้สมการที่ 3.1

$$Gray(x, y) = 0.299R(x, y) + 0.587G(x, y) + 0.114B(x, y) \quad (3.1)$$

เมื่อ $Gray(x, y)$ เป็นข้อมูลภาพระดับสีเทา, $R(x, y)$ เป็นข้อมูลภาพแสงสีแดง (Red), $G(x, y)$ เป็นข้อมูลภาพแสงสีเขียว (Green), $B(x, y)$ เป็นข้อมูลภาพแสงสีน้ำเงิน (Blue) และ (x, y) เป็นตำแหน่งของข้อมูลภาพ

3.3.1.3 ขั้นตอนการลดสัญญาณรบกวน (Noise reduction) จะทำการลดสัญญาณรบกวนในภาพด้วยการกรองแบบมัธยฐาน (Median filtering) ซึ่งเป็นการกรองแบบไม่เป็นเชิงเส้น (Nonlinear filtering) ในการกรองแบบมัธยฐานค่าของพิกเซลผลลัพธ์จะถูกกำหนดโดยค่ามัธยฐาน การหาค่ามัธยฐานทำได้โดยการนำค่าทั้งหมดในหน้าต่างการกรอง (window filtering) มาเรียงลำดับจากน้อยไปหามาก (หรือจากมากไปหาน้อยก็ได้) จากนั้นจะเลือกค่าที่อยู่ตรงกลางของลำดับเป็นค่าความเข้มแสงของจุดในภาพผลลัพธ์ หากจำนวนจุดในหน้าต่างเป็นจำนวนคู่ผลลัพธ์จะคำนวณได้จากการเฉลี่ยค่าระหว่างจุดกึ่งกลางทั้งสองจุด โดยในการกรองแบบมัธยฐานจะสามารถกำจัดสัญญาณรบกวนที่เป็นแบบสุ่มได้ดีและไม่ลดความคมชัดของภาพ ใช้งานได้ผลดีเป็นพิเศษกับสัญญาณรบกวนที่เรียกว่า Impulse noise (Salt and pepper noise) ที่มีลักษณะเป็นจุดสีขาวและสีดำกระจายบนภาพ [33]

$$g(x, y) = \text{median}\{f(m, n), (m, n \in W_{xy})\} \quad (3.2)$$

เมื่อ $f(m, n)$ เป็นข้อมูลภาพอินพุต, $g(x, y)$ เป็นข้อมูลภาพผลลัพธ์ และ W เป็นหน้าต่างการกรองขนาด $m \times n$ รอบตำแหน่ง (x, y) ซึ่งในงานวิจัยนี้จะใช้หน้าต่างการกรองขนาด 3×3

3.3.1.4 ขั้นตอนการปรับปรุงภาพให้ดีขึ้น (Image enhancement) จะทำการปรับปรุงภาพให้มีความเหมาะสมที่จะนำไปใช้ในการประมวลผลขั้นตอนต่อไป สำหรับงานวิจัยนี้จะทำการปรับความสว่างและความคมชัดของภาพ ด้วยวิธีการ Histogram equalization ซึ่งเป็นการปรับฮิสโตแกรมของภาพที่มีการกระจายตัวแบบต่าง ๆ ให้มีการกระจายตัวแบบปกติ โดยความสว่างของภาพ (Brightness: B) หาได้จากสมการที่ 3.3 และความคมชัดของภาพ (Contrast: C) หาได้จากสมการที่ 3.4

$$B = \frac{1}{NM} \sum_{y=1}^M \sum_{x=1}^N f(x, y) \quad (3.3)$$

$$C = \sqrt{\frac{1}{NM} \sum_{y=1}^M \sum_{x=1}^N [f(x, y) - B]^2} \quad (3.4)$$

3.3.2 ขั้นตอนการลงทะเบียนภาพ

เนื่องจากภาพอินพุตที่รับเข้ามาอาจจะมีขนาดและมุมมองที่ต่างไปจากภาพต้นแบบทำให้เกิดความยุ่งยากต่อการประมวลผลในขั้นตอนต่อไปจึงต้องทำการลงทะเบียนภาพให้ตรงกันเสียก่อน ในงานวิจัยนี้จะใช้วิธีการลงทะเบียนภาพแบบ Linear transformation ซึ่งเป็นวิธีการพื้นฐานที่เรียกว่าการแปลงสัมพรรค (Affine transform) โดยประกอบด้วยขั้นตอนการเลื่อนภาพ (Image translation) การหมุนภาพ (Image rotation) และการย่อหรือขยายภาพ (Image scaling) [34]

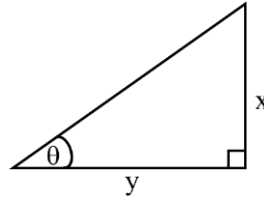
3.3.2.1 ขั้นตอนการเลื่อนภาพจะทำการตรวจหาจุดใดจุดหนึ่งที่เป็นลายวงจรรของภาพต้นแบบ เช่นจุดที่มีค่าตำแหน่ง (x, y) ต่ำสุด แล้วทำการเลื่อนภาพอินพุตนั้นทั้งภาพโดยให้จุดนั้นไปอยู่ในตำแหน่งเดียวกับภาพต้นแบบด้วยสมการที่ 3.5 และ 3.6

$$x' = x + Tx \quad (3.5)$$

$$y' = y + Ty \quad (3.6)$$

เมื่อ (x, y) เป็นตำแหน่งเดิมของภาพอินพุต, (x', y') เป็นตำแหน่งใหม่ของภาพอินพุต ส่วน Tx และ Ty เป็นระยะการจัดทางแนวแกน x และแกน y ตามลำดับ

3.3.2.2 ขั้นตอนการหมุนภาพจะทำการหามุม (θ) ในการหมุนภาพอินพุตให้ตรงกับภาพต้นแบบจากทฤษฎีตรีโกณมิติของสามเหลี่ยมมุมฉาก ตามรูปที่ 3.3 และสมการที่ 3.7 แล้วทำการหมุนภาพอินพุตในทิศทางทวนเข็มนาฬิการอบจุดหมุน (x, y) ด้วยมุม θ ด้วยสมการที่ 3.8 และ 3.9



รูปที่ 3.3 ตริโกณมิติของสามเหลี่ยมมุมฉาก

$$\theta = \tan^{-1}(x/y) \quad (3.7)$$

$$x' = x \cos \theta - y \sin \theta \quad (3.8)$$

$$y' = y \cos \theta + x \sin \theta \quad (3.9)$$

เมื่อ (x, y) เป็นตำแหน่งเดิมของภาพอินพุต, (x', y') เป็นตำแหน่งใหม่ของภาพอินพุต ส่วน θ เป็นมุมในการหมุนภาพ

3.3.2.3 ขั้นตอนการย่อหรือขยายภาพจะทำการคำนวณหา Scaling factor ระหว่างขนาดของภาพอินพุตกับภาพต้นแบบทั้งในแกน x (S_x) และแกน y (S_y) แล้วทำการย่อหรือขยายภาพอินพุตโดยใช้จุด (x, y) เป็นจุดหมุนด้วยสมการที่ 3.10 และ 3.11

$$x' = x \times S_x \quad (3.10)$$

$$y' = y \times S_y \quad (3.11)$$

3.3.3 ขั้นตอนการแบ่งส่วนภาพ

ในขั้นตอนการแบ่งส่วนภาพนี้จะทำการแยกข้อมูลภาพที่เป็นส่วนของลายวงจรรอกจากส่วนที่เป็นพื้นหลัง โดยในงานวิจัยนี้ได้เลือกใช้วิธีการแบ่งส่วนภาพทั้งหมด 4 วิธีการ ซึ่งทั้ง 4 วิธีการนี้เป็นวิธีการแบ่งส่วนภาพแบบ Pixel - Based Segmentation ที่นิยมใช้ในหมู่นักวิจัยเริ่มต้น เป็นวิธีการพื้นฐานที่ง่ายต่อการทำความเข้าใจ และเหมาะกับภาพที่ใช้ในงานวิจัยที่มีความซับซ้อนไม่มากนัก ได้แก่ วิธีการ Global Thresholding (GT), วิธีการ Adaptive Thresholding (AT), วิธีการ K-means Clustering (KM) และวิธีการ Fuzzy C-means Clustering (FCM) เพื่อนำผลลัพธ์ที่ได้มาเปรียบเทียบกันและเลือกวิธีการที่เหมาะสมไปใช้งานต่อไป

3.3.3.1 วิธีการ Global Thresholding เป็นเทคนิคที่ใช้ค่าเทรชโฮลด์ (Threshold) เพียงค่าเดียวเพื่อแบ่งส่วนของภาพให้ออกเป็น 2 กลุ่ม โดยใช้สมการที่ 3.12

$$g(x,y) = \begin{cases} 1 & \text{if } g(x,y) > T \\ 0 & \text{otherwise} \end{cases} \quad (3.12)$$

เมื่อ $g(x,y)$ คือข้อมูลภาพ ณ ตำแหน่งที่ (x, y)

T คือค่าเทรชโฮลด์

ขั้นตอนของ Global Thresholding แสดงได้ดังนี้ [35]

1. กำหนดค่าเทรชโฮลด์แบบอัตโนมัติด้วย Otsu's Threshold [36]
2. กำหนดให้ข้อมูลที่มีมากกว่าค่า T ให้มีค่าเป็น 1 นอกเหนือจากนั้นให้มีค่าเป็น 0

ขั้นตอนของการกำหนดค่าเทรชโฮลด์แบบอัตโนมัติ แสดงได้ดังนี้

1. กำหนดค่าเทรชโฮลด์เริ่มต้น (T_0) นิยมใช้ค่าเฉลี่ยหรือค่ากลางระหว่างค่าสูงสุดและต่ำสุด
2. แบ่งข้อมูลออกเป็น 2 กลุ่มโดยใช้เทรชโฮลด์ T_0 กำหนดให้เป็น G_1 และ G_2
3. หาค่าเฉลี่ยของ G_1 กำหนดให้เป็น u_1 และหาค่าเฉลี่ยของ G_2 กำหนดให้เป็น u_2
4. คำนวณหาค่าเทรชโฮลด์ใหม่ $T = (u_1+u_2) / 2$
5. ทำซ้ำข้อ 2 - 4 จนค่าเทรชโฮลด์ใหม่มีการเปลี่ยนแปลงไปจากค่าเทรชโฮลด์ก่อนหน้าด้วยจำนวนที่น้อยตามต้องการ

3.3.3.2 วิธีการ Adaptive Threshold เป็นเทคนิคการหาค่าเทรชโฮลด์ที่ปรับค่าตามข้อมูลที่อยู่ในวินโดวส์ (window) เทคนิคนี้เหมาะกับภาพที่มีความสว่างไม่สม่ำเสมอ โดยแต่ละบริเวณจะใช้เทรชโฮลด์ที่ต่างกัน ค่าเทรชโฮลด์ที่ได้เกิดจากการหาค่าเฉลี่ยของทุกพิกเซลภายใต้ Moving window ที่มีขนาด $m \times n$ และหาค่าเทรชโฮลด์เช่นนี้กับบริเวณที่ไม่ซ้ำกัน จนกระทั่งได้กำหนดค่าเทรชโฮลด์ครบในทุกวินโดวส์ ถ้าค่าระดับเทาของพิกเซลนั้นมีค่ามากกว่าค่าเทรชโฮลด์ของวินโดวส์นั้นจะกำหนดให้เป็น 1 แต่ถ้าค่าระดับเทาของพิกเซลนั้นมีน้อยกว่าค่าเทรชโฮลด์ของวินโดวส์นั้นจะกำหนดให้เป็น 0 ทำเช่นนี้จนครบทุกวินโดวส์ จะได้ผลลัพธ์เป็นภาพ Binary

ขั้นตอนของ Adaptive Thresholding แสดงได้ดังนี้ [35]

1. กำหนดขนาดของวินโดวส์และแบ่งข้อมูลภาพออกเป็นภาพย่อยตามขนาดของวินโดวส์
2. กำหนดค่าเทรชโฮลด์ของภาพย่อยด้วย Otsu's Threshold
3. กำหนดให้ข้อมูลที่มีมากกว่าค่า T ให้มีค่าเป็น 1 นอกเหนือจากนั้นให้มีค่าเป็น 0
4. ทำซ้ำข้อ 2 - 3 ไปจนครบทุกภาพย่อย

3.3.3.3 วิธีการจัดกลุ่มแบบ K-means เป็นเทคนิคการจำแนกข้อมูลออกเป็นกลุ่มย่อย (Cluster) โดยจัดให้ข้อมูลที่เหมือนกันอยู่ด้วยกัน ผู้ใช้จะต้องกำหนดจำนวนกลุ่มหรือจำนวนกลุ่มย่อยที่ต้องการด้วยตัวเองพร้อมทั้งกำหนดค่าเซนทรอยด์ (Centroid) เริ่มต้นของแต่ละกลุ่มย่อยและนำแต่ละข้อมูลจัดเข้ากลุ่มย่อยโดยดูระยะห่างจากค่าเซนทรอยด์ของแต่ละกลุ่มย่อย จากนั้นใช้การหาค่าเฉลี่ยของแต่ละกลุ่มย่อยใหม่ เพื่อกำหนดค่าเซนทรอยด์ของแต่ละกลุ่มย่อยใหม่ ทำซ้ำแบบนี้ไปจนข้อมูลไม่มีการ

เปลี่ยนกลุ่มย่อยอีก การกำหนดค่าเซนทรอยด์เริ่มต้นของแต่ละกลุ่มจะมีความสำคัญเป็นอย่างมาก เนื่องจากค่าเซนทรอยด์จะเป็นตัวกำหนดว่าข้อมูลควรจะอยู่ในกลุ่มย่อยใด ถ้าการกำหนดค่าเซนทรอยด์ออกมาต่างกันผลลัพธ์ที่ได้จากการแบ่งกลุ่มข้อมูลก็จะต่างกันด้วย และในการแบ่งกลุ่มด้วยวิธีการจัดกลุ่มแบบ K-means จะใช้ค่าระยะห่างรวมซึ่งเป็น Loss function ที่ต้องการ Minimize ในการทำ Clustering ซึ่งสามารถหาได้ด้วยสมการ Euclidean distance ดังแสดงในสมการที่ 3.13 [37]

$$J = \sum_{j=1}^k \sum_{i=1}^n \|x_i^{(j)} - c_j\|^2 \quad (3.13)$$

โดยที่ J คือค่าระยะห่างรวมซึ่งเป็น Loss function ที่ต้องการ Minimize ในการทำ Clustering

k คือจำนวนกลุ่มย่อย

$x_i^{(j)}$ คือข้อมูลที่ตำแหน่งที่ i ในกลุ่มที่ j

c_j คือค่าเซนทรอยด์ของกลุ่มที่ j และ n คือ จำนวนข้อมูล

ขั้นตอนของวิธีการจัดกลุ่มแบบ K-means เป็นดังนี้

1. กำหนดจำนวนกลุ่มย่อยให้เป็น K กลุ่ม
2. กำหนดค่าเซนทรอยด์เริ่มต้นของแต่ละกลุ่มย่อยด้วยวิธีที่เหมาะสมเพราะตำแหน่งเซน-ทรอยด์เริ่มต้นที่แตกต่างกันทำให้ได้ผลลัพธ์สุดท้ายแตกต่างกันด้วย
3. นำข้อมูลจัดเข้ากลุ่มย่อย โดยดูระยะห่างจากค่าเซนทรอยด์ของแต่ละกลุ่มย่อย
4. หาค่าเฉลี่ยของแต่ละกลุ่มย่อย และกำหนดให้เป็นค่าเซนทรอยด์ใหม่ของกลุ่มย่อยนั้น
5. ทำซ้ำข้อ 2 - 3 จนทุกข้อมูลไม่มีการเปลี่ยนกลุ่มย่อยอีก

3.3.3.4 วิธีการจัดกลุ่มแบบ Fuzzy C-means เป็นเทคนิคการจำแนกข้อมูลออกเป็นกลุ่มย่อย เช่นเดียวกับวิธีการจัดกลุ่มแบบ K-means เพียงแต่ข้อมูลของวิธีการจัดกลุ่มแบบ K-means จะต้องสังกัดกลุ่มย่อยใดกลุ่มย่อยหนึ่งเท่านั้นจึงไม่เหมาะกับข้อมูลที่มีความสัมพันธ์กัน (Correlation) ในส่วนของวิธีการจัดกลุ่มแบบ Fuzzy C-means ข้อมูลจะมีโอกาสเป็นสมาชิกของหลายกลุ่มย่อย แต่จะมีค่าการเป็นสมาชิกด้วยน้ำหนักที่แตกต่างกันไป โดยค่าการเป็นสมาชิกหาได้ตามสมการที่ 3.14 [38]

$$u_{ij} = \frac{1}{\sum_{q=1}^k \left(\frac{\|x_i - c_j\|}{\|x_i - c_q\|} \right)^{\frac{2}{m-1}}} \quad (3.14)$$

ค่าเซนทรอยด์ของแต่ละกลุ่มย่อย สามารถหาได้ตามสมการที่ 3.15

$$c_j = \frac{\sum_{i=1}^n (x_i u_{ij}^m)}{\sum_{i=1}^n (u_{ij}^m)} \quad (3.15)$$

ค่าระยะห่างรวมซึ่งเป็น Loss function ที่ต้องการ Minimize ในการทำ Clustering ซึ่งสามารถหาได้ด้วยสมการที่ 3.16

$$J = \sum_{j=1}^k \sum_{i=1}^n u_{ij}^m \|x_i^{(j)} - c_j\|^2 \quad (3.16)$$

โดยที่ J คือค่าระยะห่างรวมซึ่งเป็น Loss function ที่ต้องการ Minimize ในการทำ Clustering

u_{ij} คือค่าการเป็นสมาชิกของข้อมูล

k คือจำนวนกลุ่มย่อย

$x_i^{(j)}$ คือข้อมูลที่ตำแหน่งที่ i ในกลุ่มที่ j

c_j คือค่าเซนทรอยด์ของกลุ่มที่ j

n คือจำนวนข้อมูล

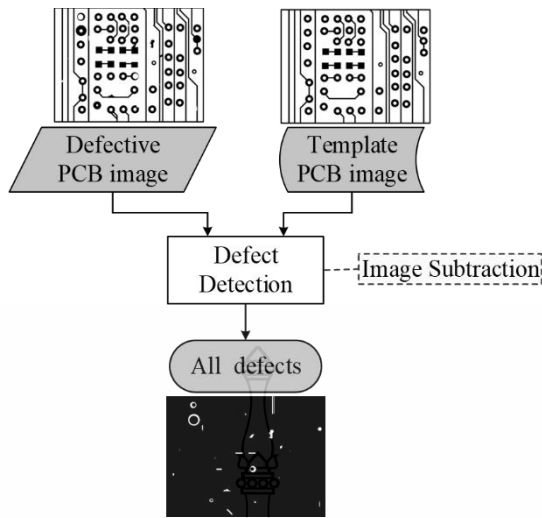
m คือจำนวนจริงใดๆ ที่มากกว่า 1

ขั้นตอนของวิธีการจัดกลุ่มแบบ Fuzzy C-means แสดงได้ดังนี้

1. กำหนดจำนวนกลุ่มย่อยให้เป็น K กลุ่ม
2. กำหนดค่าเซนทรอยด์เริ่มต้นของแต่ละกลุ่มย่อยด้วยวิธีที่เหมาะสม
3. นำข้อมูลจัดเข้ากลุ่มย่อย โดยดูระยะห่างจากค่าเซนทรอยด์ของแต่ละกลุ่มย่อย
4. หาค่าการเป็นสมาชิกของแต่ละข้อมูลในทุกกลุ่มย่อย
5. หาค่าเซนทรอยด์ใหม่ของแต่ละกลุ่มย่อย
6. นำข้อมูลจัดเข้ากลุ่มย่อยใหม่
7. ทำซ้ำข้อ 3-5 จนค่าการเป็นสมาชิกของแต่ละข้อมูลสัมฤทธิ์ผลตามที่ต้องการ

3.4 การตรวจจับข้อบกพร่อง

สำหรับขั้นตอนการตรวจจับข้อบกพร่องของ PCB (PCB Defects Detection) จะใช้ภาพอินพุตซึ่งเป็นภาพแบบไบนารีทั้งสองภาพ ได้แก่ 1) ภาพ PCB ที่มีข้อบกพร่องจากชุดข้อมูล DefPCB และ 2) ภาพ PCB ต้นแบบจากชุดข้อมูล TemPCB ในงานวิจัยนี้จะใช้วิธีการ Image subtraction สำหรับตรวจจับข้อบกพร่องทั้งหมดที่เกิดขึ้นโดยทำการเปรียบเทียบภาพ PCB ที่มีข้อบกพร่องกับภาพ PCB ต้นแบบ แบบพิกเซลต่อพิกเซล ผลลัพธ์ที่ได้จากการลบภาพตามสมการที่ 3.17 และ 3.18 จะสร้างภาพ



รูปที่ 3.4 ขั้นตอนการตรวจจับข้อบกพร่อง

ใหม่ขึ้นมา 2 ภาพ คือ 1) ภาพในส่วนที่มีทองแดงเกิน (Excess copper) หรือเรียกว่าภาพ Negative ($n(x, y)$) และ 2) ภาพในส่วนที่ทองแดงขาด (Missing copper) หรือเรียกว่าภาพ Positive ($p(x, y)$) และเมื่อรวมภาพ Negative และภาพ Positive เข้าด้วยกันก็จะได้ภาพซึ่งแสดงข้อบกพร่องทั้งหมดที่เกิดขึ้นบนแผ่น PCB ที่มีข้อบกพร่อง

$$n(x, y) = g(x, y) - f(x, y) \quad (3.17)$$

$$p(x, y) = f(x, y) - g(x, y) \quad (3.18)$$

เมื่อ $g(x, y)$ เป็นภาพ PCB ที่มีข้อบกพร่อง และ $f(x, y)$ เป็นภาพ PCB ต้นแบบ

โดยชนิดของข้อบกพร่องที่ทองแดงขาด ได้แก่ Breakout, Pin hole, Open circuit, Mouse bite, Missing conductor และ Over etch จะปรากฏในภาพ Positive ส่วนชนิดของข้อบกพร่องที่ทองแดงเกิน ได้แก่ Under etch, Spur, Short circuit, Wrong size hole, Spurious copper, Excessive short, และ Missing hole จะปรากฏในภาพ Negative ยกเว้นข้อบกพร่องชนิด Conductor too close ที่มีทั้งส่วนที่ทองแดงขาดและส่วนที่ทองแดงเกินก็เลยปรากฏอยู่ในทั้งสองภาพ

3.5 การออกแบบอัลกอริทึม

3.5.1 Circular hough transform (CHT)

ในการหาส่วนที่เป็น Hole pad ของลายวงจรจะใช้ฮัลกอริทึมจากพื้นฐานของ CHT เพื่อหาส่วนที่เป็นวงกลมในข้อมูลภาพ PCB ต้นแบบ โดยให้ชื่อว่าภาพ Hole pad: $hp(x, y)$ เริ่มต้นด้วยการตั้งสมมติฐานว่ากำลังมองหาวงกลมที่มีรัศมีเฉพาะ นั่นคือจะได้ค่า r โดยสมการของแต่ละวงกลม คือ

$$x = a + r \cos(\theta) \quad (3.19)$$

$$y = b + r \sin(\theta) \quad (3.20)$$

ดังนั้นทุกจุดใน xy space จะเทียบเท่ากับวงกลมใน ab space และเนื่องจากการจัดเรียงสมการใหม่สำหรับจุด (x_1, y_1) และ θ กวาดจาก 0 ถึง 360 องศา จะได้

$$a = x_1 + r \cos(\theta) \quad (3.21)$$

$$b = y_1 + r \sin(\theta) \quad (3.22)$$

ดังนั้นการขั้นตอนในการหาจุดศูนย์กลางของวงกลมจึงเป็นดังนี้

1. โหลดภาพ
2. ตรวจสอบขอบภาพ (edge) และสร้างภาพแบบไบนารี
3. สำหรับทุกพิกเซลของ 'edge' ให้สร้างวงกลมใน ab space
4. สำหรับทุกจุดบนวงกลมใน ab space ให้ใช้ 'votes' ในเซลล์ตัวสะสม
5. เซลล์ที่มีจำนวนของ 'votes' มากกว่าจะเป็นจุดศูนย์กลางของวงกลม

3.5.2 Morphological reconstruction (MR)

ในการเติมช่องว่างขนาดเล็กร่างต่าง ๆ ของภาพ PCB ให้เต็มจะใช้อัลกอริทึมจากพื้นฐานของ MR สำหรับข้อมูลภาพแบบไบนารีนั้นวิธีการนี้จะเปลี่ยนพิกเซลพื้นที่หลังที่เชื่อมต่อกับพิกเซลเบื้องหน้าจนกว่าจะถึงขอบเขตของวัตถุ โดย $f(x, y)$ และ $g(x, y)$ แสดงถึงภาพไบนารี และภาพผลลัพธ์คือ $fm(x, y)$ และ $gm(x, y)$ ซึ่งจะมีค่าของพิกเซลเป็น 0 ในทุกตำแหน่ง ยกเว้นตำแหน่งที่เป็นขอบภาพซึ่งกำหนดให้มีค่าเป็น $1 - f(x, y)$ และ $1 - g(x, y)$ ดังแสดงในสมการที่ 3.23 และ 3.24

$$fm(x, y) = \begin{cases} 1 - f(x, y), & \text{if } (x, y) \text{ on border of } f(x, y) \\ 0, & \text{Otherwise} \end{cases} \quad (3.23)$$

$$gm(x, y) = \begin{cases} 1 - g(x, y), & \text{if } (x, y) \text{ on border of } g(x, y) \\ 0, & \text{Otherwise} \end{cases} \quad (3.24)$$

ในงานวิจัยนี้จะใช้การเติมช่องว่างทั้งภาพ PCB ต้นแบบและภาพ PCB ที่มีข้อบกพร่องซึ่งเป็นภาพแบบไบนารี โดยผลลัพธ์ที่ได้จะเป็นภาพที่เรียกว่าภาพ Flood-fill ต้นแบบ: $fm(x, y)$ และภาพ Flood-fill ที่มีข้อบกพร่อง: $gm(x, y)$

3.5.3 Connected component labelling (CCL)

อัลกอริทึมจากพื้นฐานของ CCL จะถูกนำมาใช้ในการหาข้อมูลภาพที่เรียกว่าภาพ Labelling ซึ่งเป็นภาพทางสัญลักษณ์ โดยอัลกอริทึมจะทำการจับกลุ่มของพิกเซลที่เชื่อมต่อกันในภาพเข้าด้วยกัน

ในงานวิจัยนี้จะใช้วิธีการของ Binary CCL เพื่อจับกลุ่มข้อมูลภาพในส่วนที่เป็นลายวงจรที่ต่อเนื่องกันเข้าเป็นกลุ่มเดียวกัน โดยเลือกใช้อัลกอริทึม Faster-scanning ซึ่งเป็นวิธีการที่ไม่ยุ่งยากซับซ้อนและประมวลผลได้รวดเร็ว อัลกอริทึม Faster-scanning จะประกอบด้วยการทำ First pass และ Second pass ดังแสดงรายละเอียดของแต่ละขั้นตอนดังต่อไปนี้

ขั้นตอนการทำ First pass :

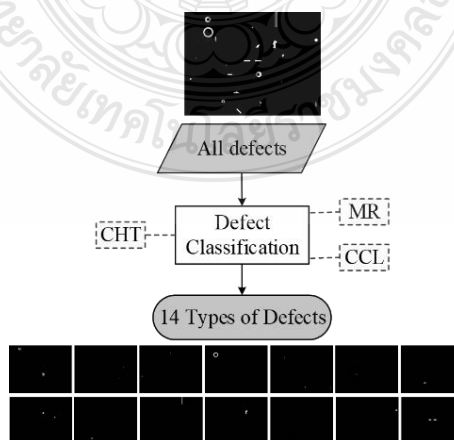
1. วนลูปเข้าไปในอิลิเมนต์ของข้อมูล แต่ละคอลัมน์ และแต่ละแถว (Raster Scanning)
2. ถ้าอิลิเมนต์นั้นไม่ใช่ Background
 - 2.1. ให้หาอิลิเมนต์ข้างเคียง (Neighboring Elements) ของอิลิเมนต์ปัจจุบัน
 - 2.2. ถ้าไม่มี Neighbors ให้ติดป้ายชื่ออิลิเมนต์ปัจจุบันด้วยป้ายชื่อที่ไม่ซ้ำกับป้ายชื่ออื่นและดำเนินการต่อ
 - 2.3. ถ้ามี Neighbors ให้หา Neighbor ที่มีป้ายชื่อน้อยที่สุด และกำหนดป้ายชื่อนั้นให้กับอิลิเมนต์ปัจจุบัน
 - 2.4. เก็บค่า Equivalence ของป้ายชื่อของพิกเซลปัจจุบันและป้ายชื่อของพิกเซลที่อยู่ข้างเคียง

ขั้นตอนการทำ Second pass :

1. วนลูปเข้าไปในอิลิเมนต์ของข้อมูล แต่ละคอลัมน์และแต่ละแถว
2. ถ้าอิลิเมนต์นั้นไม่ใช่ Background ให้กำหนดป้ายชื่อใหม่ให้กับอิลิเมนต์ด้วยป้ายชื่อที่เทียบเท่ากันที่มีค่าน้อยที่สุด

ซึ่งผลลัพธ์ที่ได้จากขั้นตอนนี้จะเป็นข้อมูลภาพที่มีการติดป้ายชื่อองค์ประกอบที่เชื่อมต่อกันเข้าด้วยกัน และเมื่อทำการล้อมรอบในแต่ละพื้นที่จะได้ภาพที่เรียกว่าภาพ Labelling

3.5.4 การออกแบบอัลกอริทึมสำหรับจำแนกข้อบกพร่อง



รูปที่ 3.5 ขั้นตอนการจำแนกชนิดข้อบกพร่อง

ในขั้นตอนนี้ผู้วิจัยจะทำการออกแบบอัลกอริทึมในการจำแนกชนิดของข้อบกพร่อง PCB ที่รู้จักทั้งหมด 14 ชนิดขึ้นมาเพื่อปรับปรุงประสิทธิภาพของวิธีการต่าง ๆ ที่พัฒนาขึ้นมาก่อนหน้านี้ โดยอัลกอริทึมที่ออกแบบขึ้นมาจะสามารถจำแนกชนิดของข้อบกพร่อง PCB เปล่าแบบหน้าเดียวหลังจากผ่านกระบวนการกัดและเจาะในกระบวนการผลิตได้ครบทั้งหมด 14 ชนิดด้วยความถูกต้องแม่นยำที่สูงเพียงพอสำหรับใช้ในกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์ในอุตสาหกรรม ในการออกแบบอัลกอริทึมจะใช้ข้อมูลภาพของข้อบกพร่องทั้งหมดจากขั้นตอนการตรวจจับข้อบกพร่องเป็นข้อมูลอินพุต ผ่านกรรมวิธีในการจำแนกชนิดของข้อบกพร่องโดยใช้เครื่องมือทางการประมวลผลภาพ ได้แก่ การดำเนินการทางคณิตศาสตร์และตรรกศาสตร์ CHT MR และ CCL สุดท้ายจะได้เอาท์พุตเป็นภาพของข้อบกพร่องแต่ละชนิดแยกออกจากกัน โดยมีขั้นตอนทั้งหมดดังต่อไปนี้

Input: ภาพ $f(x, y)$, $g(x, y)$, $n(x, y)$, $p(x, y)$, $hp(x, y)$, $fm(x, y)$ และ $gm(x, y)$

Outout: ภาพของข้อบกพร่อง PCB แต่ละชนิด

Step 1: หาข้อบกพร่องในภาพ Negative ที่ Hole pad จาก " $(n(x, y) \& hp(x, y))$ " จะได้ข้อบกพร่องชนิด Missing hole และ Wrong size hole

Step 2: ใช้อัลกอริทึม MR กับภาพใน Step 1 ถ้ากลุ่มไหนมีช่องว่างนั้นคือข้อบกพร่องชนิด Wrong size hole (D09) นอกจากนั้นจะเป็น Missing hole (D13)

Step 3: หาข้อบกพร่องในภาพ Positive ที่ Hole pad จาก " $(p(x, y) \& hp(x, y))$ " จะได้ข้อบกพร่องชนิด Breakout, Mouse bite และ Pin hole

Step 4: ใช้อัลกอริทึม MR กับภาพที่ได้จาก Step 3 และถ้ากลุ่มใดมีช่องว่างคือ Breakout (D01)

Step 5: หา Pin hole (D02) จาก " $D01 \oplus (p(x, y) \& (g(x, y) \oplus gm(x, y)))$ "

Step 6: ถ้ามีการเติมช่องว่างอื่นที่ติดกับข้อบกพร่องจะได้ Excessive short (D12)

Step 7: ใช้อัลกอริทึม CCL กับภาพ PCB ต้นแบบเพื่อหาจำนวนและขนาดของกลุ่ม

Step 8: เพิ่มแต่ละข้อบกพร่อง Negative ลงในภาพ PB ต้นแบบ และใช้อัลกอริทึม CCL เพื่อหาจำนวนและขนาดของกลุ่มอีกครั้ง

Step 9: จาก Step 8 ถ้าจำนวนกลุ่มลดลงข้อบกพร่องที่เพิ่มลงไปจะเป็น Short circuit (D08) แต่ถ้าจำนวนกลุ่มเพิ่มขึ้นจะเป็น Spurious copper (D11)

Step 10: ทำเหมือนใน Step 8 แต่เพิ่มข้อบกพร่อง Positive แทน

Step 11: จาก Step 10 ถ้าจำนวนกลุ่มเพิ่มขึ้น และกลุ่มที่เพิ่มขึ้นนั้นมีขนาดของน้อยกว่าความกว้างของเส้นลายวงจร (Trace) ยกกำลังสอง ข้อบกพร่องที่เพิ่มลงไปจะเป็น Open circuit (D03) แต่ถ้ากลุ่มที่เพิ่มขึ้นมีขนาดมากกว่าความกว้างของเส้นลายวงจรยกกำลัง

สอง ขั้วบขบพร่องที่เพิ่มลงไปจะเป็น Missing conductor (D06)

Step 12: จาก Step 8 และ 10 ถ้าจำนวนกลุ่มไม่เปลี่ยนแปลง จะพิจารณาชนิดของขั้วบขบพร่องที่เหลือตาม Step 13 - 17

Step 13: ถ้าขั้วบขบพร่องเกิดขึ้นในภาพ Negative และไม่สมมาตรกัน จะเป็น Spur (D07)

Step 14: เช่นเดียวกันกับ Step 13 แต่เกิดขึ้นในภาพ Positive จะเป็น Mouse bite (D05)

Step 15: ถ้าขั้วบขบพร่องเกิดขึ้นในกลุ่มเดียวกันทั้งในภาพ Negative และ Positive จะเป็น Conductor too close (D10)

Step 16: ถ้าขั้วบขบพร่องเกิดขึ้นในภาพ Negative และสมมาตรกัน จะเป็น Under etch (D04)

Step 17: เช่นเดียวกันกับ Step 16 แต่เกิดขึ้นในภาพ Positive จะเป็น Over etching (D14)

Step 18: ได้ภาพของขั้วบขบพร่องทั้ง 14 ชนิด แยกออกจากกันเป็น 14 ภาพ

จากอัลกอริทึมของการจำแนกประเภทขั้วบขบพร่องของ PCB ที่ออกแบบขึ้นมาใหม่นี้สามารถเขียนแทนด้วยโฟลว์ชาร์ต ดังแสดงในรูปที่ 3.6

3.6 วิธีการทดสอบงานวิจัย

ในการทดสอบงานวิจัยนี้ ผู้วิจัยจะทำการทดสอบในแต่ละขั้นตอนย่อยต่าง ๆ ทุกขั้นตอน และสุดท้ายจะทำการทดสอบประสิทธิภาพของอัลกอริทึมที่ออกแบบ โดยมีรายละเอียดดังนี้

3.6.1 ทดสอบ Pre-processing

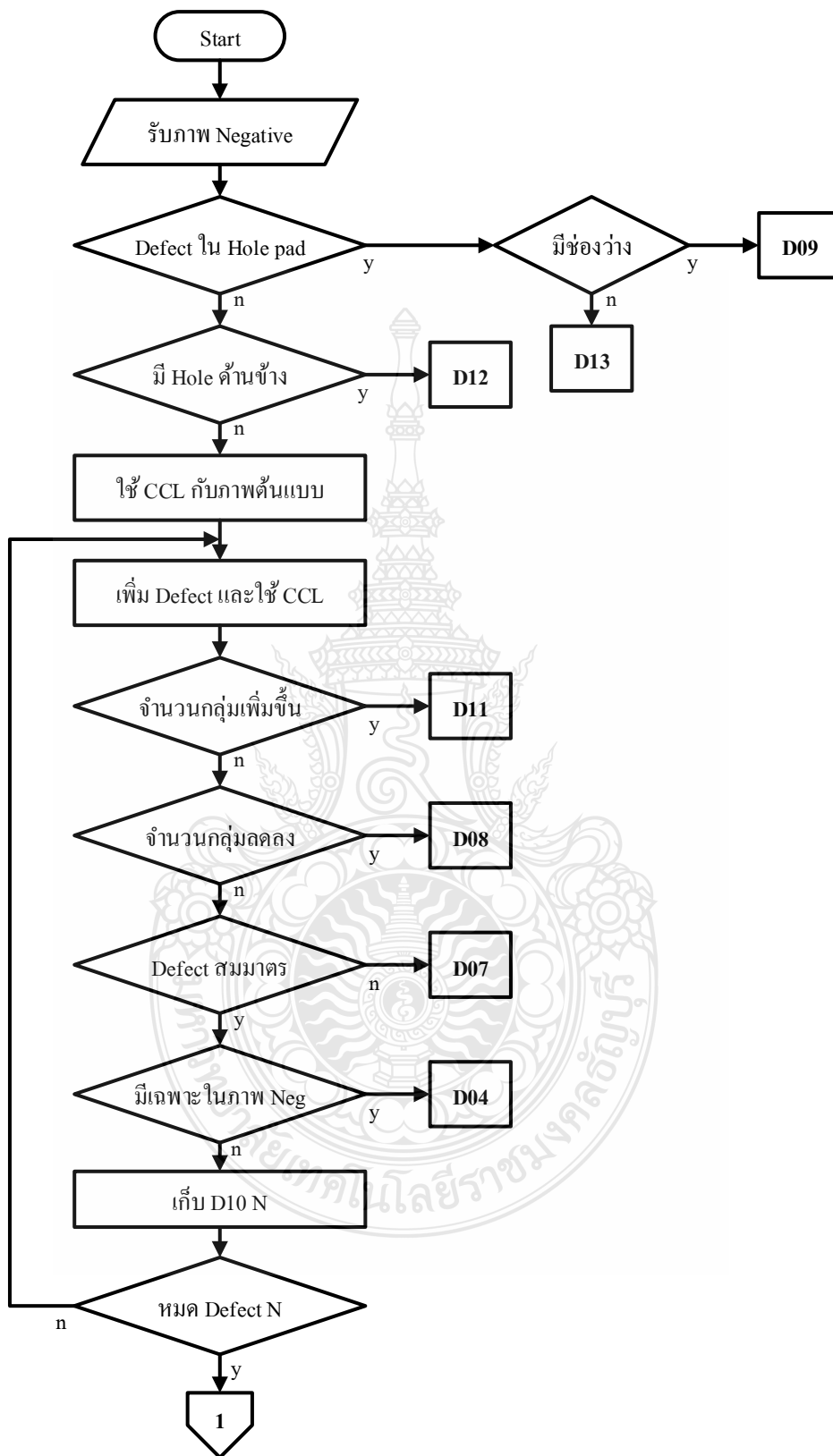
ในการทดสอบ Pre-processing จะนำข้อมูลภาพ PCB อินพุตในชุดข้อมูล InPCB จำนวน 1,000 ภาพ มาทำการเลือกบริเวณภาพที่สนใจ (ROI) แล้วแปลงภาพสี RGB แบบ 8 บิตที่รับเข้ามาให้เป็นเป็นภาพระดับสีเทาแบบ 8 บิต จากนั้นทำการลดสัญญาณรบกวนในภาพด้วย Median filtering และสุดท้ายจะทำการปรับความสว่างและความคมชัดของภาพ ด้วยวิธีการ Histogram equalization

3.6.2 ทดสอบ Image registration

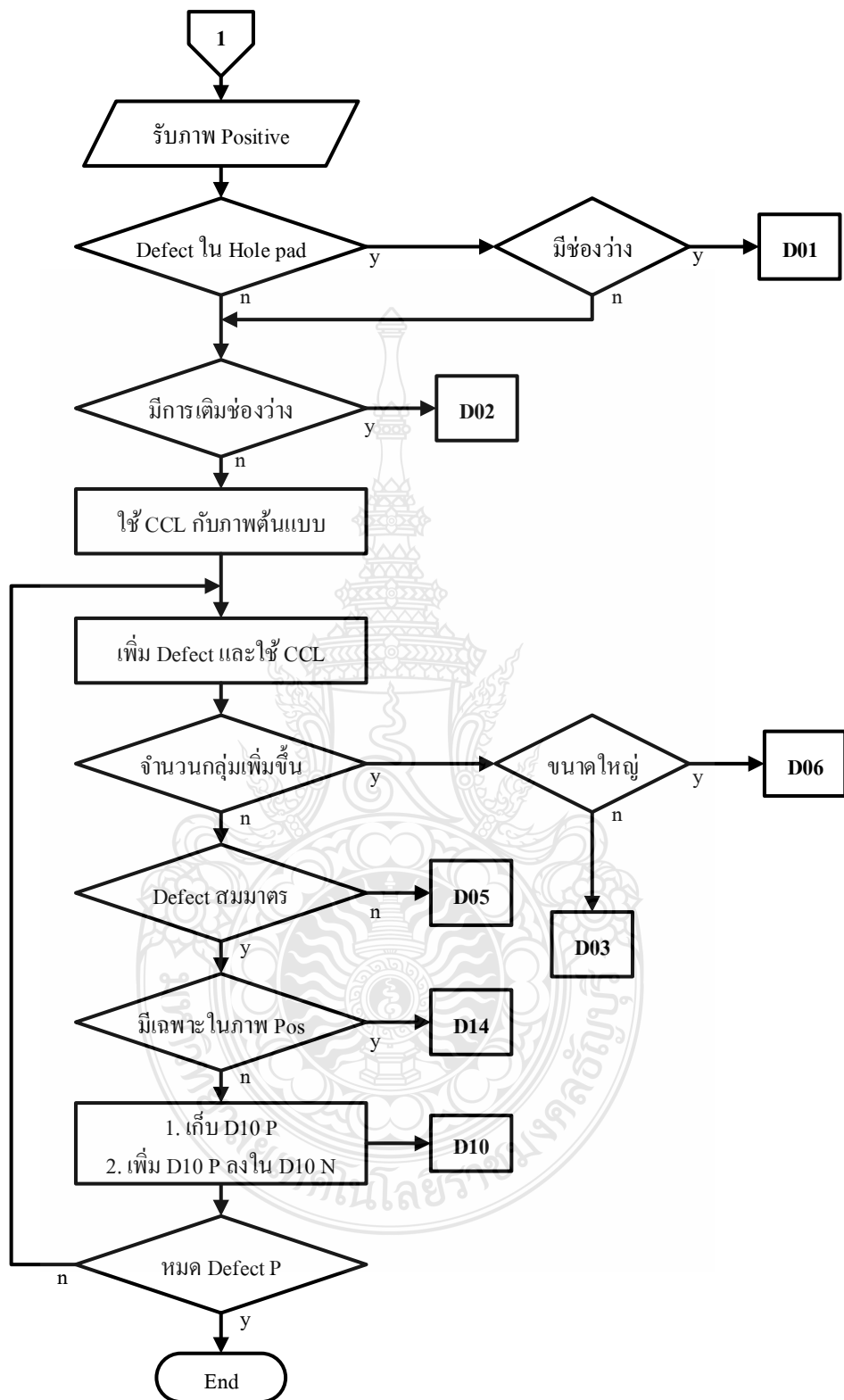
ในการทดสอบ Image registration จะนำข้อมูลภาพ PCB ที่ผ่านขั้นตอนการประมวลผลขั้นต้นแล้วมาทำการลงทะเบียนภาพแบบ Affine transform เพื่อปรับให้ภาพที่รับเข้ามามีขนาดและมุมมองเหมือนกับภาพต้นแบบ โดยใช้วิธีการเลื่อนภาพ การหมุนภาพ และการย่อหรือขยายภาพ

3.6.3 ทดสอบ Image segmentation

ในการทดสอบ Image segmentation จะนำข้อมูลภาพ PCB ที่ผ่านขั้นตอนการลงทะเบียนภาพแล้วมาทำการแบ่งส่วนภาพด้วยวิธีการทั้ง 4 วิธีการ ได้แก่ Global Thresholding, Adaptive Thresholding, K-means Clustering และ Fuzzy C-means Clustering เพื่อทำการแยกข้อมูลภาพ



รูปที่ 3.6 โพลีชาร์ตของอัลกอริทึมที่ออกแบบ



รูปที่ 3.6 โฟลว์ชาร์ตของอัลกอริทึมที่ออกแบบ (ต่อ)

ที่เป็นส่วนของลายวงจรออกจากส่วนที่เป็นพื้นหลัง ทำการคำนวณและเปรียบเทียบประสิทธิภาพในการแบ่งส่วนภาพของทั้ง 4 วิธีการดังกล่าว

3.6.4 ทดสอบ Defects detection

ในการทดสอบ Defects detection จะนำข้อมูลภาพ PCB ที่มีข้อบกพร่องในชุดข้อมูล DefPCB และข้อมูลภาพ PCB ต้นแบบในชุดข้อมูล TemPCB มาเปรียบเทียบกับด้วยวิธีการ Image subtraction เพื่อหาข้อบกพร่องทั้งหมดที่เกิดขึ้นบน PCB ที่มีข้อบกพร่องทั้งหมด 10,000 ภาพ พร้อมระบุตำแหน่ง, พื้นที่ และลักษณะการเกิดข้อบกพร่องที่ตรวจจับได้

3.6.5 ทดสอบวิธีการ CHT

ในส่วนของการทดสอบ Circular hough transform จะทำการตรวจหาส่วนที่เป็น Hole pad ของลายวงจรซึ่งมีลักษณะเป็นรูวงกลมในข้อมูลภาพ PCB ต้นแบบทั้งหมด 100 ภาพ จากนั้นจะนำข้อมูลที่ได้ไปสร้างเป็นภาพ Hole pad ของแต่ละภาพ PCB ต้นแบบ

3.6.6 ทดสอบวิธีการ MR

ในส่วนของการทดสอบ Morphological reconstruction จะทำการเติมช่องว่างขนาดเล็กของภาพ PCB ต้นแบบและภาพ PCB ที่มีข้อบกพร่องให้เต็มโดยใช้การดำเนินการ Flood-fill จากนั้นจะนำข้อมูลที่ได้ไปสร้างเป็นภาพ Flood-fill ต้นแบบและภาพ Flood-fill ที่มีข้อบกพร่อง

3.6.7 ทดสอบวิธีการ CCL

ในส่วนของการทดสอบ Connected component labelling จะทำการจับกลุ่มของพิกเซลที่เชื่อมต่อกันในภาพเข้าด้วยกันโดยใช้วิธีการของ Binary CCL แบบ Faster-scanning กับทั้งข้อมูลภาพ PCB ต้นแบบและข้อมูลภาพ PCB ที่มีข้อบกพร่อง จากนั้นจะใช้จำนวนของกลุ่มและขนาดของกลุ่มเพื่อนำไปวิเคราะห์หาชนิดของข้อบกพร่อง PCB ในอัลกอริทึมต่อไป

3.6.8 ทดสอบประสิทธิภาพของอัลกอริทึม

สำหรับการทดสอบประสิทธิภาพของอัลกอริทึมในการจำแนกประเภทข้อบกพร่องของ PCB ที่ออกแบบขึ้นมาใหม่นี้ ผู้วิจัยจะนำข้อมูลภาพข้อบกพร่องจากขั้นตอน Defects detection พร้อมทั้งข้อมูลภาพอื่น ๆ ที่เกี่ยวข้องทั้งหมดเข้าสู่อัลกอริทึมที่ได้ออกแบบขึ้นมา ผลการทดสอบที่ได้จะแสดงเป็นภาพของข้อบกพร่อง PCB แต่ละชนิดที่แยกออกจากกันเป็น 14 ภาพของแต่ละข้อบกพร่อง

บทที่ 4

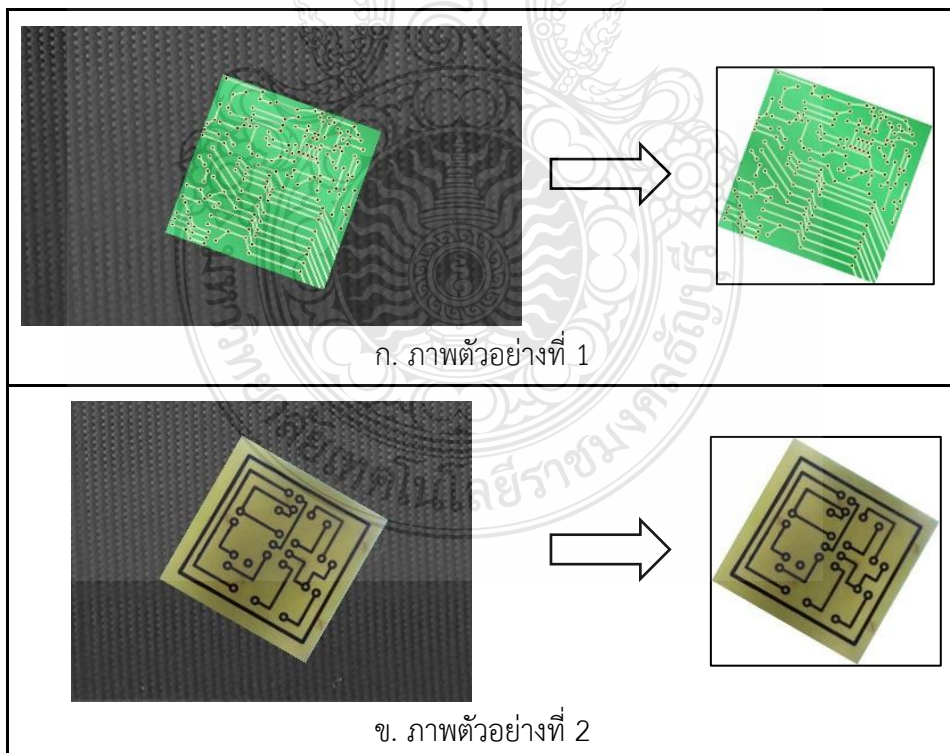
ผลการทดสอบและการวิเคราะห์ผล

ในบทนี้จะแสดงให้เห็นถึงผลการทดสอบงานวิจัยตามวิธีการทดสอบในหัวข้อที่ 3.6 และการวิเคราะห์ผล ซึ่งประกอบด้วย ผลการทดสอบ Pre-processing ผลการทดสอบ Image registration ผลการทดสอบ Image segmentation ผลการทดสอบ Defects detection ผลการทดสอบวิธีการ CHT ผลการทดสอบวิธีการ MR ผลการทดสอบวิธีการ CCL และผลการทดสอบประสิทธิภาพของอัลกอริทึมในการจำแนกประเภทข้อบกพร่องของ PCB ที่ออกแบบจะแสดงในหัวข้อสุดท้าย

4.1 ผลการทดสอบ Pre-processing

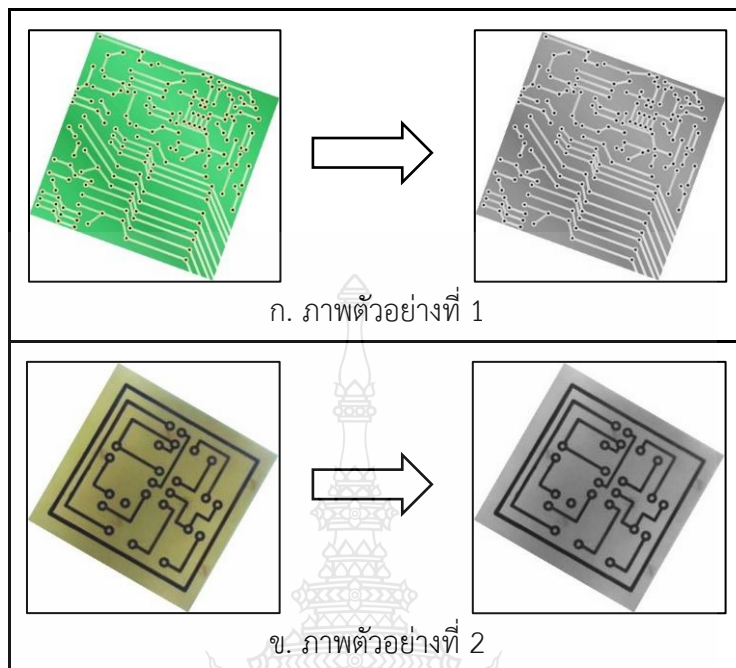
สำหรับขั้นตอนการประมวลผลขั้นต้นนี้ จะทำการปรับปรุงภาพอินพุตซึ่งเป็นภาพสี RGB ด้วยขั้นตอนย่อย ได้แก่ การเลือกบริเวณภาพที่สนใจ การแปลงเป็นภาพระดับสีเทา การลดสัญญาณรบกวน และการปรับปรุงภาพให้ดีขึ้น เพื่อให้ได้ภาพที่เหมาะสมจะนำไปวิเคราะห์ต่อไป

ผลการทดสอบที่ได้จากขั้นตอนการเลือกบริเวณภาพที่สนใจ แสดงในรูปที่ 4.1



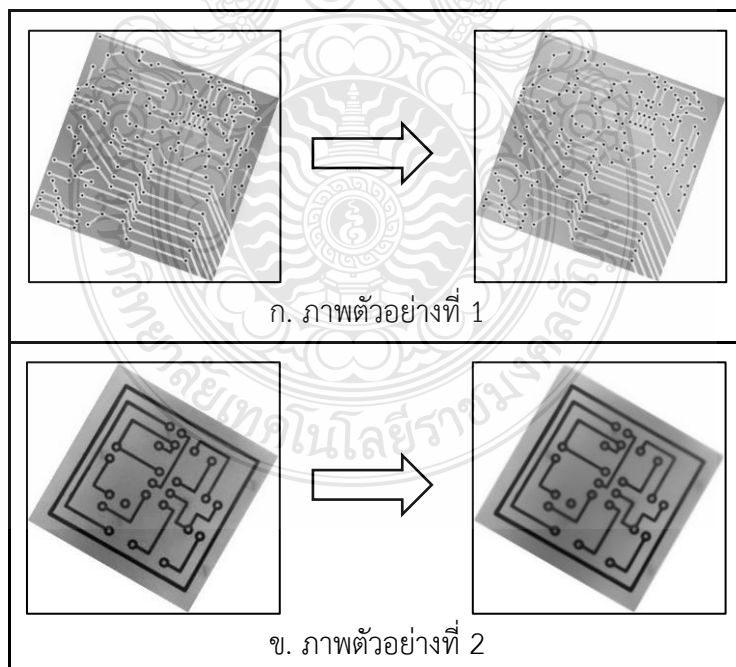
รูปที่ 4.1 ตัวอย่างภาพจากขั้นตอน ROI

ผลการทดสอบที่ได้จากขั้นตอนการแปลงเป็นภาพระดับสีเทา แสดงในรูปที่ 4.2



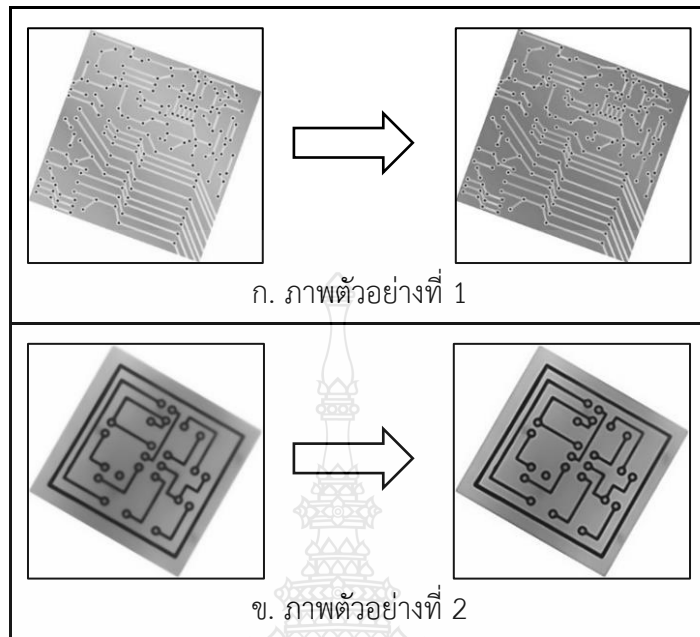
รูปที่ 4.2 ตัวอย่างภาพจากขั้นตอน Gray scale conversion

ผลการทดสอบที่ได้จากขั้นตอนการลดสัญญาณรบกวน แสดงในรูปที่ 4.3



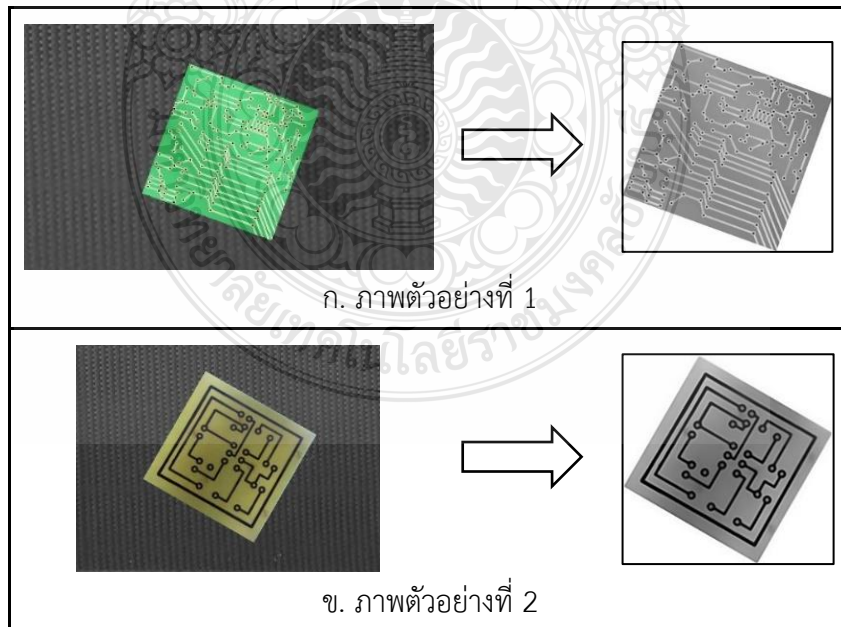
รูปที่ 4.3 ตัวอย่างภาพจากขั้นตอน Noise reduction

ผลการทดสอบที่ได้จากขั้นตอนการปรับปรุงภาพให้ดีขึ้น แสดงในรูปที่ 4.4



รูปที่ 4.4 ตัวอย่างภาพจากขั้นตอน Image enhancement

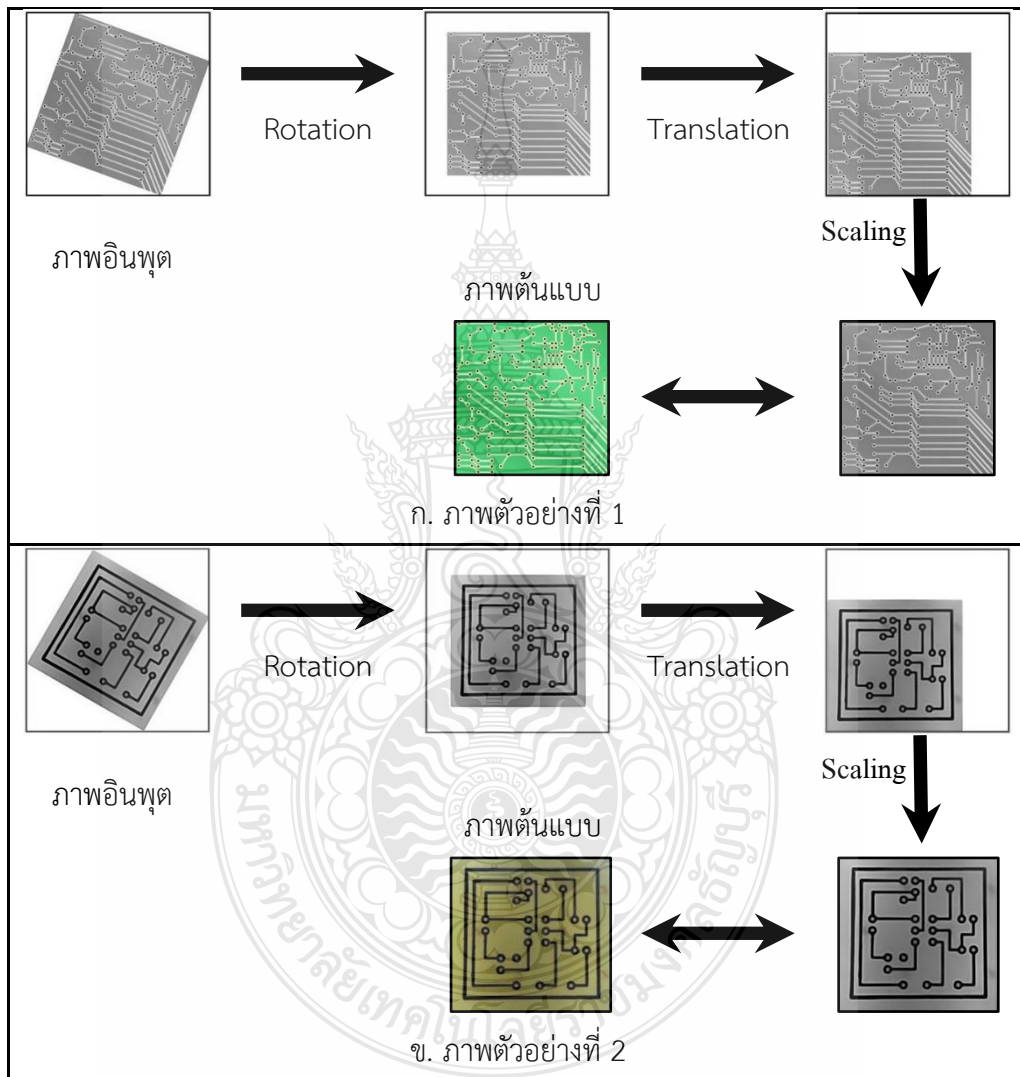
จากตัวอย่างผลการทดสอบแต่ละขั้นตอนย่อยที่ได้ในรูปที่ 4.1 ถึงรูปที่ 4.4 แสดงให้เห็นว่า ข้อมูลภาพที่ผ่านการประมวลผลขั้นต้นแล้วจะได้ภาพที่มีลักษณะตามที่เราต้องการเพื่อนำไปใช้ประมวลผลในขั้นตอนต่อไป โดยผลการทดสอบขั้นตอน Pre-processing ทั้งหมด แสดงในรูปที่ 4.5



รูปที่ 4.5 ตัวอย่างภาพจากขั้นตอน Image enhancement

4.2 ผลการทดสอบ Image registration

ในขั้นตอนการลงทะเบียนภาพนี้ผู้วิจัยได้ทำการปรับแต่งลักษณะทางกายภาพของภาพ PCB ที่มีข้อบกพร่องให้เหมือนกับภาพ PCB ต้นแบบ ด้วยวิธีการ Affine transform ซึ่งประกอบด้วย การหมุนภาพ การเลื่อนภาพ และการย่อหรือขยายภาพ โดยใช้ภาพ PCB ต้นแบบเป็นภาพอ้างอิง ผลการทดสอบที่ได้จากขั้นตอนนี้ แสดงตัวอย่างในรูปที่ 4.6



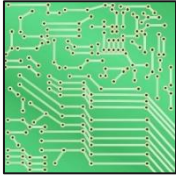
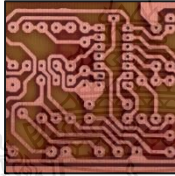
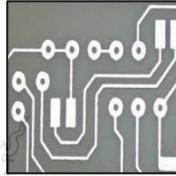
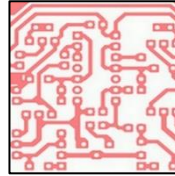
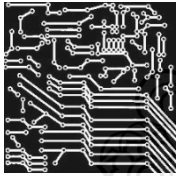
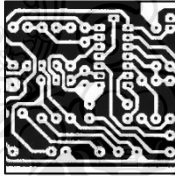
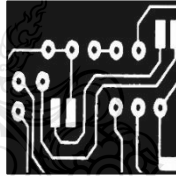
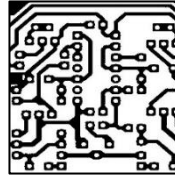
รูปที่ 4.6 ตัวอย่างภาพจากขั้นตอน Image registration

จากผลการทดสอบที่ได้ในรูปที่ 4.6 แสดงให้เห็นว่าเมื่อนำภาพ PCB ที่มีข้อบกพร่องซึ่งใช้เป็นภาพอินพุตผ่านขั้นตอน Image registration แล้ว ภาพอินพุตจะถูกเปลี่ยนแปลงให้เป็นภาพผลลัพธ์ซึ่งจะมีขนาด ทิศทาง และการวางแนวเหมือนกับภาพ PCB ต้นแบบทุกประการ


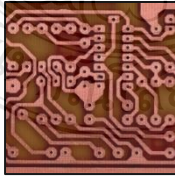
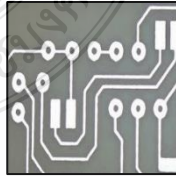
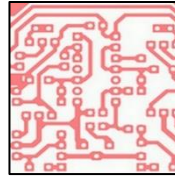
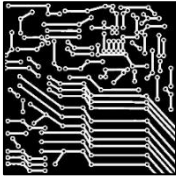
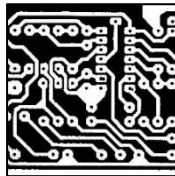
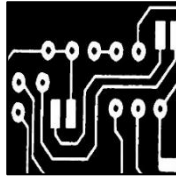
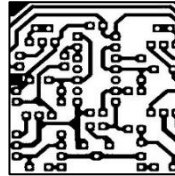
4.3 ผลการทดสอบ Image segmentation

สำหรับขั้นตอนการแบ่งส่วนภาพในงานวิจัยนี้ได้ทำการแยกข้อมูลภาพที่เป็นส่วนของลายวงจรออกจากส่วนที่เป็นพื้นหลัง ด้วยการทดสอบตามหัวข้อที่ 3.6.3 โดยใช้ข้อมูลภาพ PCB เปล่าแบบด้านเดียว ที่ผ่านขั้นตอนการประมวลผลขั้นต้นและขั้นตอนการลงทะเบียนภาพแล้วจำนวน 1,000 ภาพ และได้เลือกใช้วิธีการแบ่งส่วนภาพแบบ Pixel - Based Segmentation ทั้งหมด 4 วิธีการ ได้แก่ Global Thresholding, Adaptive Thresholding, K-means Clustering และ Fuzzy C-means Clustering ผลการแบ่งส่วนภาพที่ได้แสดงในตารางที่ 4.1 - 4.4

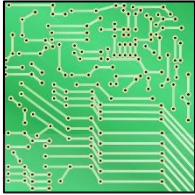
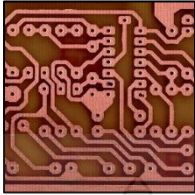
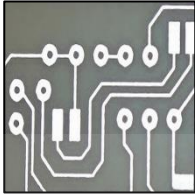
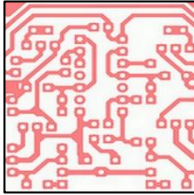
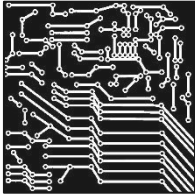
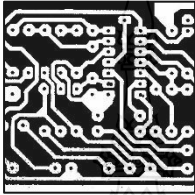
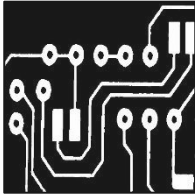
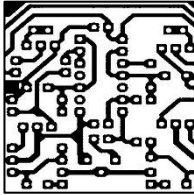
ตารางที่ 4.1 ผลการแบ่งส่วนภาพด้วยวิธีการ Global Thresholding

ภาพ	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
ต้นฉบับ				
ผลลัพธ์				

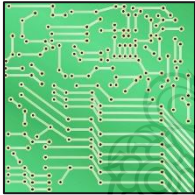
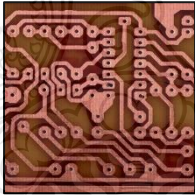
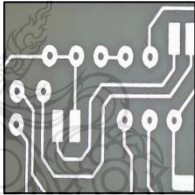
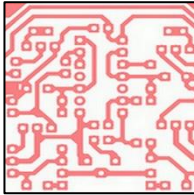
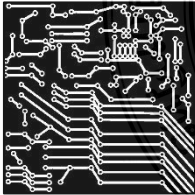
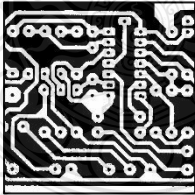
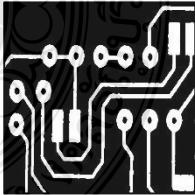
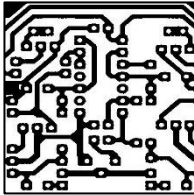
ตารางที่ 4.2 ผลการแบ่งส่วนภาพด้วยวิธีการ Adaptive Thresholding

ภาพ	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
ต้นฉบับ				
ผลลัพธ์				

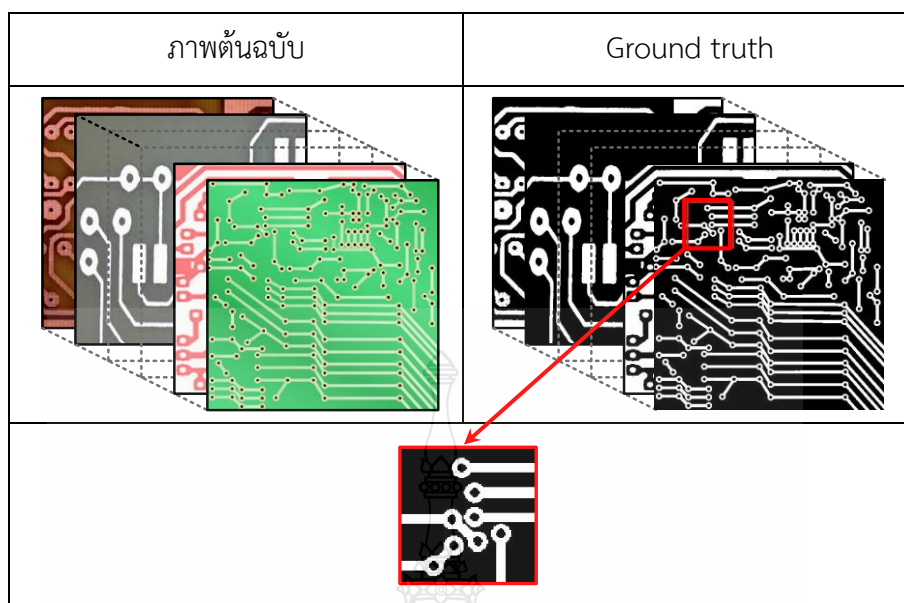
ตารางที่ 4.3 ผลการแบ่งส่วนภาพด้วยวิธีการ K-means Clustering

ภาพ	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
ต้นฉบับ				
ผลลัพธ์				

ตารางที่ 4.4 ผลการแบ่งส่วนภาพด้วยวิธีการ Fuzzy C-means Clustering

ภาพ	ตัวอย่างที่ 1	ตัวอย่างที่ 2	ตัวอย่างที่ 3	ตัวอย่างที่ 4
ต้นฉบับ				
ผลลัพธ์				

เพื่อให้เห็นผลการทดลองที่ชัดเจนยิ่งขึ้น ผู้วิจัยจึงได้ทำการขยายภาพบางส่วนให้ใหญ่ขึ้น ดังแสดงในรูปที่ 4.7 และตารางที่ 4.5



รูปที่ 4.7 ตัวอย่างผลการแบ่งส่วนภาพ

ตารางที่ 4.5 ตัวอย่างการเปรียบเทียบผลการแบ่งส่วนภาพ

	Ground truth	GT	AT	KM	FCM
ตัวอย่างภาพที่ 1					
ตัวอย่างภาพที่ 2					
ตัวอย่างภาพที่ 3					
ตัวอย่างภาพที่ 4					

จากรูปที่ 4.7 และตารางที่ 4.5 แสดงให้เห็นทั้งภาพที่เป็นภาพต้นฉบับ (Original) ภาพที่แบ่งส่วนด้วยสายตาหรือเรียกว่า Ground truth ภาพที่แบ่งส่วนด้วยวิธีการ Global Thresholding โดยใช้ค่าเทรชโฮลด์จาก Otsu's method ภาพที่แบ่งส่วนด้วยวิธีการ Adaptive Thresholding ภายใต้ Moving window ที่มีขนาด 100x100 และใช้ค่าเทรชโฮลด์จาก Otsu's method เช่นกัน ภาพที่แบ่งส่วนภาพด้วยวิธีการ K-means Clustering โดยกำหนดค่าเซนทรอยด์เริ่มต้นทั้งสองเป็นกึ่งกลางของค่าต่ำสุดกับค่าเฉลี่ยและกึ่งกลางของค่าเฉลี่ยกับค่าสูงสุด และรูปสุดท้ายเป็นภาพที่แบ่งส่วนภาพด้วยวิธีการ

จัดกลุ่มแบบ Fuzzy C-means โดยใช้สัมประสิทธิ์ $m = 2$ ทำการกำหนดค่าเซนทรอยด์เริ่มต้นทั้งสอง เป็นกึ่งกลางของค่าต่ำสุดกับค่าเฉลี่ยและกึ่งกลางของค่าเฉลี่ยกับค่าสูงสุดเช่นเดียวกับวิธีการจัดกลุ่มแบบ K-means clustering และกำหนดให้สมาชิกของแต่ละข้อมูลสัมฤทธิ์ผลเมื่อค่าเซนทรอยด์เปลี่ยนแปลงน้อยกว่า 10^{-3}

ในส่วนของตัวชี้วัด (Parameter) ที่จะใช้ในการประเมินประสิทธิภาพของเทคนิคในการแบ่งส่วนภาพทั้ง 4 วิธีการที่ได้ทำการทดลอง ได้แก่ ค่า Intersection over Union (IoU) ค่าความถูกต้อง ค่าความแม่นยำ ค่าความไว และเวลาที่ใช้ในการประมวลผล

1. ค่า Intersection over Union ซึ่งสามารถหาได้จากอัตราส่วนของพื้นที่ที่ทับซ้อน (Area of Overlap) ระหว่างบริเวณที่ทำนาย (Prediction Area) กับบริเวณค่าความจริง (Ground truth Area)หารด้วยพื้นที่รวมทั้งหมด (Area of Union) ระหว่างบริเวณที่ทำนายกับบริเวณค่าความจริง ดังแสดงในสมการที่ 4.1

$$IoU = \frac{\text{Ground Truth} \cap \text{Prediction}}{\text{Ground Truth} \cup \text{Prediction}} \quad (4.1)$$

2. ค่าความถูกต้อง (Accuracy) คือความใกล้เคียงของค่าจริงที่ตรวจพบกับค่าที่ตรวจพบทั้งหมด สามารถหาได้ตามสมการที่ 4.2

$$\text{Accuracy} = \frac{(TP) + (TN)}{(TP) + (FP) + (TN) + (FN)} \quad (4.2)$$

เมื่อ TP คือค่าจริงเป็นบวก แสดงว่าส่วนที่เป็นลายวงจรถูกตรวจพบว่าเป็นลายวงจรถ, TN คือค่าจริงเป็นลบ แสดงว่าส่วนที่ไม่ได้เป็นลายวงจรถูกตรวจพบว่าเป็นลายวงจรถ, FP คือค่าเท็จเป็นบวก แสดงว่าส่วนที่ไม่ได้เป็นลายวงจรถูกตรวจพบว่าเป็นลายวงจรถ และ FN คือค่าเท็จเป็นลบ แสดงว่าส่วนที่เป็นลายวงจรถูกตรวจพบว่าเป็นลายวงจรถ

3. ค่าความแม่นยำ (Precision) คือความใกล้เคียงของภาพวัตถุที่ตรวจพบจริงกับภาพวัตถุจริงทั้งหมด สามารถหาได้ตามสมการที่ 4.3

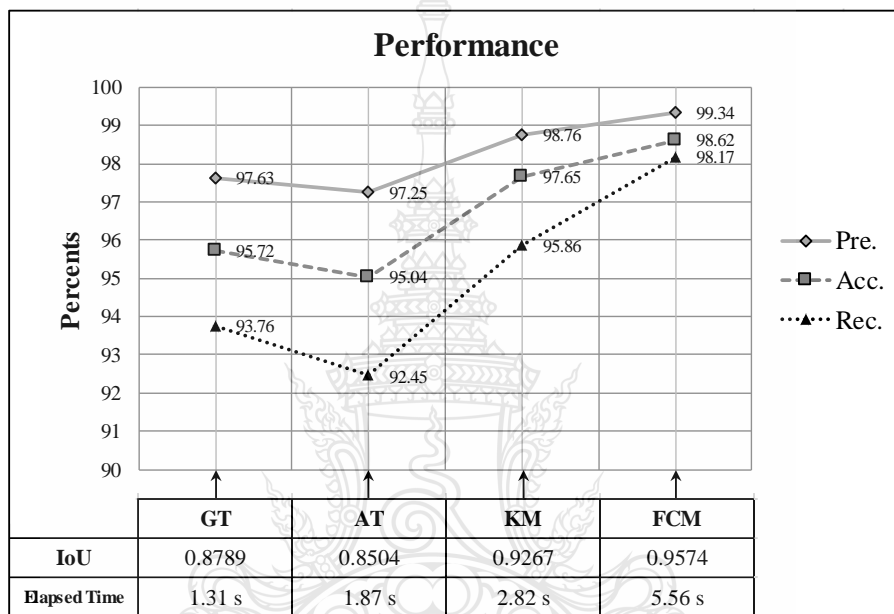
$$\text{Precision} = \frac{(TP)}{(TP) + (FP)} \quad (4.3)$$

4. ค่าความไว (Sensitivity หรือ Recall) คือสัดส่วนของภาพวัตถุที่ตรวจพบจริงกับภาพวัตถุที่ตรวจพบทั้งหมด สามารถหาได้ตามสมการที่ 4.4

$$\text{Recall} = \frac{(TP)}{(TP) + (FN)} \quad (4.4)$$

5. เวลาที่ใช้ (Elapse Time) จะพิจารณาจากค่าเฉลี่ยของเวลาในการประมวลผลบนเครื่องคอมพิวเตอร์ Laptop Acer Swift 3 ซึ่งมีชิปประมวลผลเป็น Intel Core i5-7200U ที่มีความเร็วในการประมวลผลอยู่ที่ 2.5 GHz หน่วยความจำขนาด 8 GB และใช้ระบบปฏิบัติการ Windows 10

เมื่อทำการเปรียบเทียบค่าที่ได้จากการแบ่งส่วนภาพทั้ง 4 วิธีการกับค่าที่ได้จาก Ground truth นำไปสู่ผลการประเมินประสิทธิภาพในการแบ่งส่วนภาพ โดยทำการทดลองกับภาพตัวอย่างจำนวน 1,000 ภาพ และทำการหาค่าเฉลี่ยของแต่ละตัวชี้วัด แสดงในรูปที่ 4.8



รูปที่ 4.8 การเปรียบเทียบค่าเฉลี่ยของแต่ละตัวชี้วัด

ทำการคำนวณหาค่าเบี่ยงเบนมาตรฐาน (SD) ของแต่ละตัวชี้วัดด้วยสมการที่ 4.5 และผลที่ได้แสดงในตารางที่ 4.6

$$SD = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (4.5)$$

เมื่อ x_i เป็นค่าข้อมูลทั้งหมดในชุดข้อมูล, \bar{x} เป็นค่าเฉลี่ยของชุดข้อมูล และ N เป็นจำนวนของข้อมูล

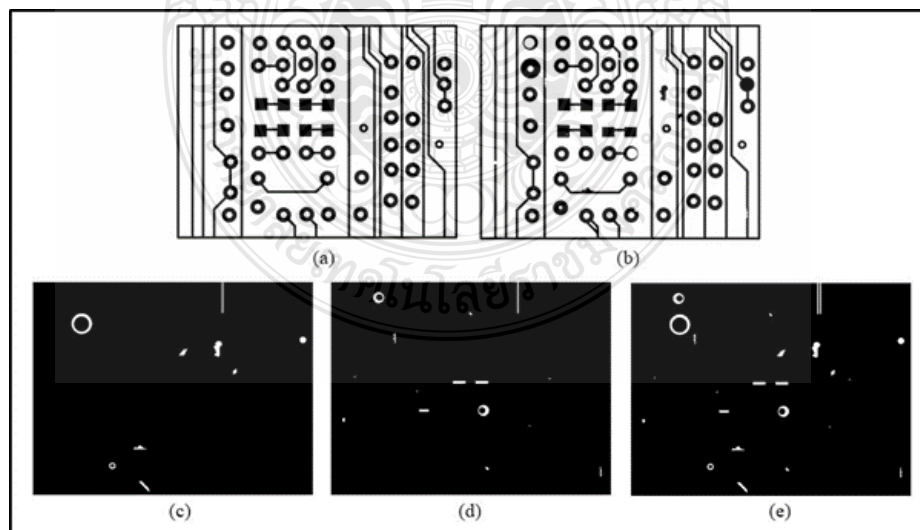
จากผลการทดลองในรูปที่ 4.8 แสดงให้เห็นว่าวิธีการจัดกลุ่มแบบ Fuzzy C-means จะมีประสิทธิภาพของการแบ่งส่วนภาพสูงกว่าวิธีการอื่น ๆ โดยมีค่าความแม่นยำ 99.34%, ค่าความถูกต้อง 98.62%, ค่าความไว 98.17% และค่า IoU เท่ากับ 0.9574 รองลงมาคือวิธีการจัดกลุ่มแบบ K-means แต่ในส่วนของเวลาที่ใช้ในการประมวลผล วิธีการจัดกลุ่มแบบ Fuzzy C-means ใช้เวลามากกว่าวิธีการอื่น และค่า SD ที่ต่ำในตารางที่ 4.6 ซึ่งให้เห็นว่าค่าตัวชี้วัดต่าง ๆ ใกล้เคียงกับค่าเฉลี่ยของแต่ละตัวชี้วัด

ตารางที่ 4.6 ค่า SD ของแต่ละตัวชี้วัดในรูปที่ 4.8

Methods	SD				
	IoU	Accuracy	Precision	Recall	Elapse Time
GT	1.236	1.201	0.416	5.309	3.546
AT	1.115	0.936	0.318	4.536	2.235
KM	1.302	1.103	0.028	3.256	2.673
FCM	1.374	1.332	0.084	2.362	4.164

4.4 ผลการทดสอบ Defect detection

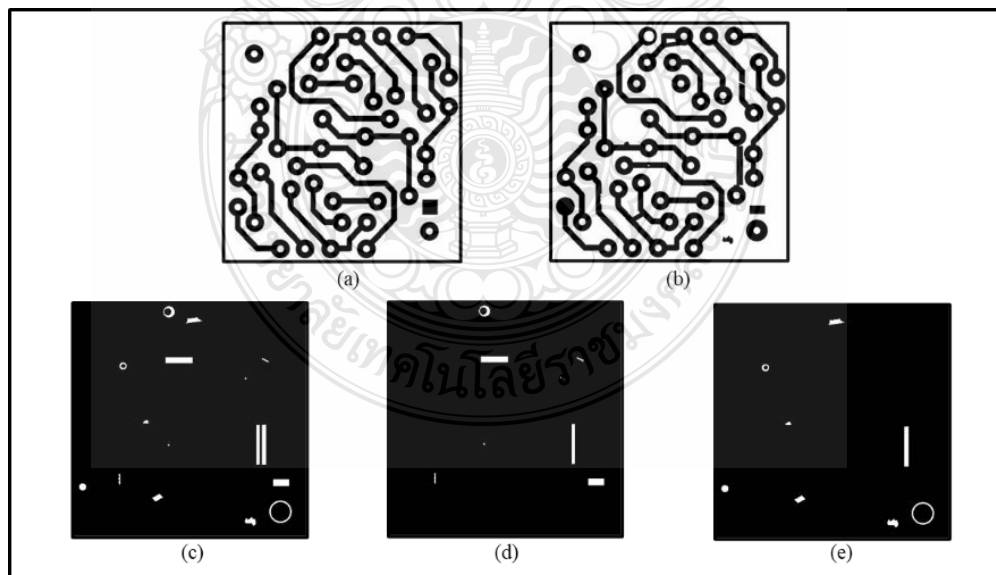
ในขั้นตอนการตรวจจับข้อบกพร่องแผ่นวงจรพิมพ์จะใช้วิธีการทดสอบตามหัวข้อที่ 3.6.4 โดยข้อบกพร่องทั้งหมดจะถูกตรวจจับจากภาพ PCB ที่มีข้อบกพร่องในชุดข้อมูล DefPCB โดยใช้อัลกอริทึม Image subtraction ทำการเปรียบเทียบแบบพิกเซลต่อพิกเซลกับภาพ PCB ต้นแบบในชุดข้อมูล TemPCB ข้อบกพร่องที่เป็นทองแดงเกินจะเกิดขึ้นเมื่อผลการลบภาพ PCB ที่มีข้อบกพร่องด้วยภาพ PCB ต้นแบบแล้วได้ผลเป็นบวก ในขณะที่ข้อบกพร่องที่เป็นทองแดงขาดจะเกิดขึ้นเมื่อผลการลบภาพ PCB ต้นแบบด้วยภาพ PCB ที่มีข้อบกพร่องแล้วได้ผลเป็นบวก ตัวอย่างผลการทดสอบหาข้อบกพร่องทั้งหมดที่เกิดขึ้นบนแผ่น PCB ที่มีข้อบกพร่อง แสดงในรูปที่ 4.9 และรูปที่ 4.10 ในขณะเดียวกันตารางที่ 4.7 และตารางที่ 4.8 จะแสดงข้อมูลของตำแหน่ง, พื้นที่ และลักษณะการเกิดข้อบกพร่องที่ตรวจจับได้



รูปที่ 4.9 ตัวอย่างของการตรวจจับข้อบกพร่องจากภาพ PCB ต้นแบบที่ 1

ตารางที่ 4.7 ข้อมูลผลการทดลองที่ได้จากตัวอย่างในรูปที่ 4.9

No.	Centroid		Area	Copper	No.	Centroid		Area	Copper
	X	Y				X	Y		
1	33	360	52	Missing	13	394	262	229	Missing
2	62	248	8	Missing	14	393	184	182	Excess
3	129	107	1001	Excess	15	400	335	414	Missing
4	123	41	410	Missing	16	408	487	46	Missing
5	167	145	81	Missing	17	483	173	516	Excess
6	208	480	138	Excess	18	488	41	243	Missing
7	225	285	8	Missing	19	495	41	241	Excess
8	243	335	151	Missing	20	519	376	8	Missing
9	279	433	174	Excess	21	527	234	94	Excess
10	292	532	197	Excess	22	571	252	9	Missing
11	335	262	227	Missing	23	705	151	220	Excess
12	364	83	32	Missing	24	704	496	71	Missing



รูปที่ 4.10 ตัวอย่างของการตรวจจับข้อบกพร่องจากภาพ PCB ต้นแบบที่ 2

ตารางที่ 4.8 ข้อมูลผลการทดลองที่ได้จากตัวอย่างในรูปที่ 4.10

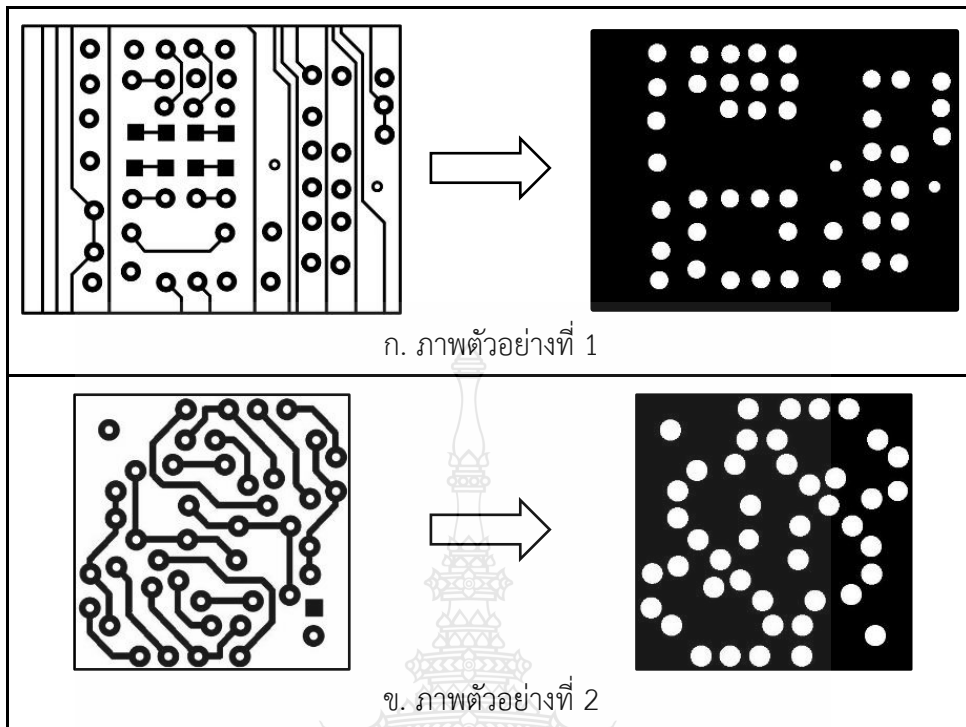
No.	Centroid		Area	Copper	No.	Centroid		Area	Copper
	X	Y				X	Y		
1	16	343	140	Excess	9	224	30	192	Excess
2	85	328	49	Missing	10	322	139	7	Missing
3	92	116	92	Excess	11	332	408	195	Excess
4	135	221	46	Excess	12	345	264	449	Missing
5	157	363	147	Excess	13	359	105	32	Missing
6	180	14	204	Missing	14	356	264	602	Excess
7	197	105	561	Missing	15	388	389	364	Excess
8	178	264	8	Missing	16	388	335	390	Missing

รูปที่ 4.9 และรูปที่ 4.10 แสดงผลของกระบวนการตรวจจับข้อบกพร่อง โดยในรูป (a) เป็นตัวอย่างของภาพ PCB ต้นแบบและรูป (b) เป็นตัวอย่างของภาพ PCB ที่มีข้อบกพร่องซึ่งทั้งสองภาพจะใช้เป็นอินพุตของกระบวนการนี้ รูป (c), (d) และ (e) แสดงข้อบกพร่องที่มีทองแดงเกิน, ข้อบกพร่องที่มีทองแดงขาด และข้อบกพร่องที่ตรวจพบทั้งหมด ตามลำดับ ในขณะที่ตารางที่ 4.7 และตารางที่ 4.8 จะแสดงข้อมูลของจำนวนของข้อบกพร่อง เช่น ภาพ PCB ในรูปที่ 4.9 มีข้อบกพร่อง 24 จุด และภาพ PCB ในรูปที่ 4.10 มีข้อบกพร่อง 16 จุด รวมไปถึงแสดงตำแหน่งของข้อบกพร่อง, ขนาดพื้นที่ของข้อบกพร่อง และลักษณะการเกิดข้อบกพร่องที่ตรวจจับได้ทั้งหมด

จากผลการตรวจจับข้อบกพร่องแผ่นวงจรพิมพ์ที่ได้แสดงให้เห็นว่ากระบวนการตรวจจับข้อบกพร่องโดยใช้วิธีการ Image subtraction นี้สามารถตรวจพบข้อบกพร่องทั้งหมดที่เกิดขึ้นบนภาพ PCB ที่มีข้อบกพร่อง รวมไปถึงสามารถระบุจำนวนของข้อบกพร่อง, ตำแหน่งของข้อบกพร่อง, ขนาดพื้นที่ของข้อบกพร่อง และลักษณะการเกิดข้อบกพร่องได้ ซึ่งคุณสมบัติเหล่านี้จะเป็นข้อมูลที่เป็นประโยชน์ในการจำแนกชนิดของข้อบกพร่องในกระบวนการต่อไป

4.5 ผลการทดสอบวิธีการ CHT

ในการตรวจหาข้อมูลภาพส่วนที่เป็น Hole pad ของลายวงจรซึ่งมีลักษณะเป็นรูปร่างกลมโดยใช้วิธีการ CHT กับข้อมูลภาพ PCB ต้นแบบ เพื่อใช้ประกอบในการจำแนกประเภทของข้อบกพร่องบางชนิดที่เกิดขึ้นภายใน Hole pad ซึ่งได้แก่ข้อบกพร่อง Breakout ข้อบกพร่อง Wrong size hole และข้อบกพร่อง Missing hole ตัวอย่างผลการทดลองที่ได้ แสดงในรูปที่ 4.11

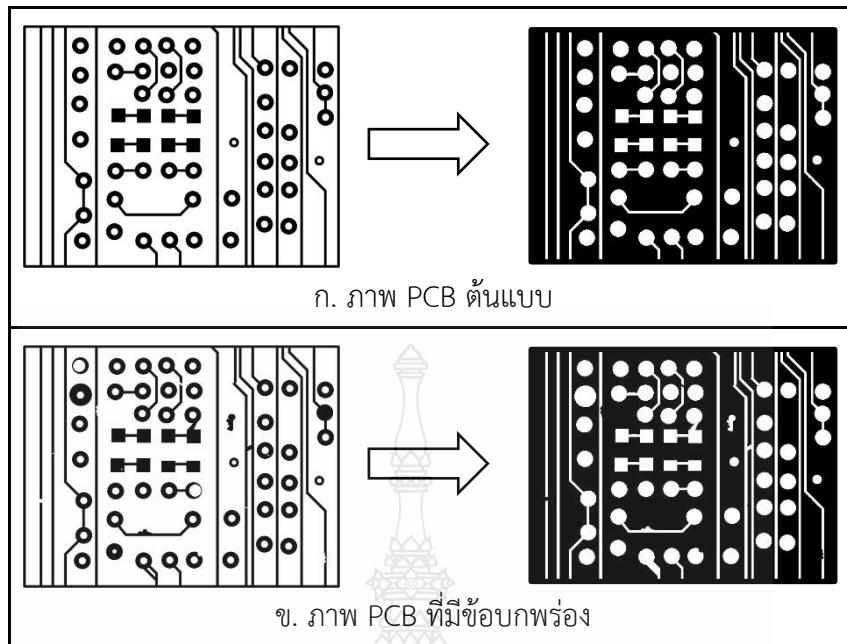


รูปที่ 4.11 ตัวอย่างผลของการตรวจหา Hole pad ของลายวงจร

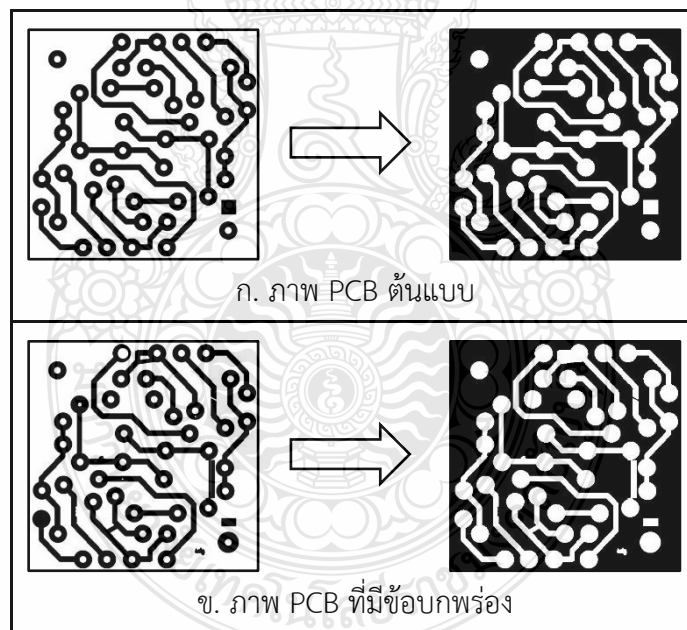
จากผลการทดสอบที่ได้ในรูปที่ 4.11 แสดงให้เห็นว่าในขั้นตอนการตรวจหา Hole pad ของลายวงจร วิธีการ CHT ที่ใช้ในงานวิจัยนี้สามารถตรวจพบ Hole pad ในภาพตัวอย่างที่ 1 จำนวน 46 Hole pad จากที่มีอยู่ 46 Hole pad และในภาพตัวอย่างที่ 2 ตรวจพบ 43 Hole pad จากที่มีอยู่ 43 Hole pad แสดงให้เห็นว่าวิธีการ CHT ที่ใช้ในงานวิจัยนี้สามารถตรวจพบ Hole pad ของลายวงจรในภาพ PCB ต้นแบบได้ครบถ้วนทั้งหมด

4.6 ผลการทดสอบวิธีการ MR

ในงานวิจัยนี้จะใช้การเติมช่องว่างขนาดเล็กโดยใช้การดำเนินการ Flood-fill ภายใต้พื้นฐานของวิธีการ MR กับข้อมูลภาพ PCB ต้นแบบและข้อมูลภาพ PCB ที่มีข้อบกพร่อง เพื่อใช้ประกอบในการจำแนกประเภทของข้อบกพร่องบางชนิดที่เกี่ยวข้องกับช่องว่างขนาดเล็กที่มีอยู่ในข้อมูลภาพของลายวงจร ซึ่งได้แก่ข้อบกพร่อง Breakout ข้อบกพร่อง Pin hole ข้อบกพร่อง Wrong size hole ข้อบกพร่อง Excessive short และข้อบกพร่อง Missing hole ตัวอย่างผลการทดลองที่ได้ แสดงในรูปที่ 4.12 และรูปที่ 4.13



รูปที่ 4.12 ตัวอย่างผลของการตรวจหาช่องว่างขนาดเล็กของภาพตัวอย่างที่ 1

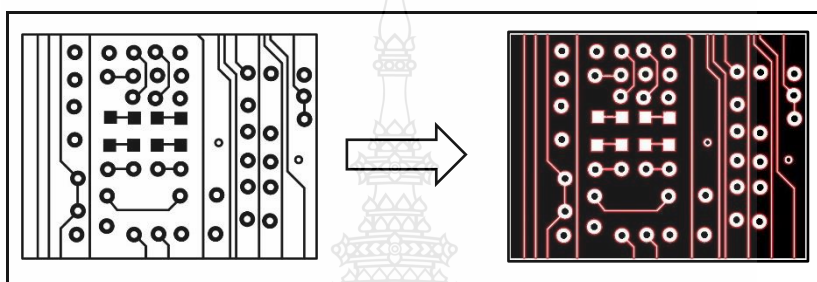


รูปที่ 4.13 ตัวอย่างผลของการตรวจหาช่องว่างขนาดเล็กของภาพตัวอย่างที่ 2

จากผลการทดสอบที่ได้ในรูปที่ 4.12 และรูปที่ 4.13 แสดงให้เห็นว่าวิธีการ MR ที่ใช้ใน งานวิจัยนี้สามารถตรวจหาช่องว่างขนาดเล็กได้อย่างครบถ้วน และทำการเติมช่องว่างให้เต็มโดยใช้การ ดำเนินการ Flood-fill จากนั้นจะนำข้อมูลที่ได้ไปสร้างเป็นภาพ Flood-fill ต้นแบบและภาพ Flood-fill ที่มีข้อบกพร่องเพื่อนำไปใช้งานต่อไป

4.7 ผลการทดสอบวิธีการ CCL

ในการจำแนกข้อบกพร่องบางชนิดของ PCB สามารถพิจารณาได้จากจำนวนกลุ่มและขนาดกลุ่มของข้อมูลที่เชื่อมต่อกันภายในภาพ ดังนั้นงานวิจัยนี้จึงเลือกใช้วิธีการของ Binary CCL แบบ Faster-scanning เพื่อจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ต้นแบบและข้อมูลภาพ PCB ที่มีข้อบกพร่อง แล้วนำจำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพทั้งสองมาเปรียบเทียบกัน ตัวอย่างผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ต้นแบบ แสดงในรูปที่ 4.14 และตารางที่ 4.9

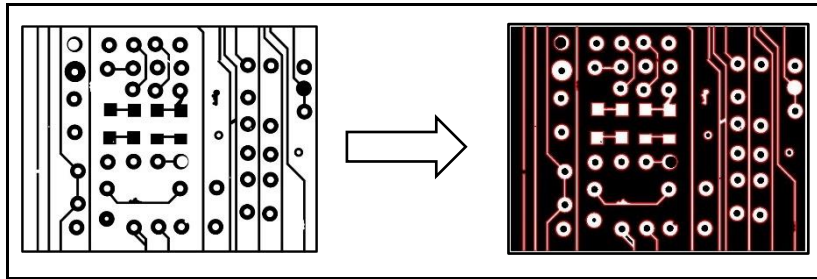


รูปที่ 4.14 ผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ต้นแบบ

ตารางที่ 4.9 จำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพ PCB ต้นแบบในรูปที่ 4.14

กลุ่มที่	1	2	3	4	5	6	7	8	9	10	11
ขนาด	3,330	3,885	5,083	1,009	1,006	999	1,006	1,004	3,885	1,011	3,107
กลุ่มที่	12	13	14	15	16	17	18	19	20	21	22
ขนาด	2,175	2,168	999	2,349	2,359	2,623	1,296	2,681	1,012	2,126	2,334
กลุ่มที่	23	24	25	26	27	28	29	30	31	32	33
ขนาด	2,334	1,292	1,008	1,000	1,003	1,009	3,904	1,011	1,002	329	3,943
กลุ่มที่	34	35	36	37	38	39	40	41	42	43	44
ขนาด	3,420	1,517	994	1,001	1,004	1,000	1,002	3,885	1,007	998	999
กลุ่มที่	45	46	47	48	49	50	51				
ขนาด	998	3,462	1,000	3,578	2,979	328	1,010				

จากรูปที่ 4.14 และตารางที่ 4.9 แสดงให้เห็นว่าวิธีการ Binary CCL สามารถจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ต้นแบบได้อย่างถูกต้อง โดยข้อมูลภาพ PCB ต้นแบบที่นำมาเป็นตัวอย่างมีจำนวนกลุ่มของข้อมูล 51 กลุ่ม และแต่ละกลุ่มมีขนาดพื้นที่ที่แตกต่างกัน ส่วนตัวอย่างผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ที่มีข้อบกพร่อง แสดงในรูปที่ 4.15 และตารางที่ 4.10



รูปที่ 4.15 ผลการจับกลุ่มของข้อมูลที่เชื่อมต่อกันภายในข้อมูลภาพ PCB ที่มีข้อบกพร่อง

ตารางที่ 4.10 จำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพ PCB ที่มีข้อบกพร่องในรูปที่ 4.15

กลุ่มที่	1	2	3	4	5	6	7	8	9	10	11
ขนาด	2,130	1,148	3,877	5,083	2,000	1,009	597	1,006	1,004	3,804	1,149
กลุ่มที่	12	13	14	15	16	17	18	19	20	21	22
ขนาด	3,281	2,175	1,011	999	2,341	2,359	2,623	1,006	1,494	1,233	1,416
กลุ่มที่	23	24	25	26	27	28	29	30	31	32	33
ขนาด	1,012	1,711	3,518	1,878	1,292	1,008	1,003	963	3,904	1,011	1,002
กลุ่มที่	34	35	36	37	38	39	40	41	42	43	44
ขนาด	516	329	7,440	1,517	994	992	1,004	1,000	1,002	3,885	1,007
กลุ่มที่	45	46	47	48	49	50	51	52	53		
ขนาด	998	999	998	3,462	1,000	3,507	3,199	328	1,010		

จากรูปที่ 4.15 และตารางที่ 4.10 จะเห็นว่าข้อมูลภาพ PCB ที่มีข้อบกพร่องที่นำมาเป็นตัวอย่างมีจำนวนกลุ่มของข้อมูล 53 กลุ่มซึ่งเพิ่มขึ้นจากข้อมูลภาพ PCB ต้นแบบ และขนาดพื้นที่ของแต่ละกลุ่มก็จะเปลี่ยนแปลงไปจากผลที่ได้จากข้อมูลภาพ PCB ต้นแบบในตารางที่ 4.9 เช่นกัน เมื่อนำจำนวนกลุ่มและขนาดกลุ่มของข้อมูลภาพ PCB ที่มีข้อบกพร่องมาทำการเปรียบเทียบกับข้อมูลภาพ PCB ต้นแบบจะพบว่าเมื่อ PCB เกิดข้อบกพร่องบางชนิดจะทำให้จำนวนกลุ่มของข้อมูลเปลี่ยนแปลงไป และเมื่อ PCB เกิดข้อบกพร่องบางชนิดจะส่งผลให้ขนาดของกลุ่มเปลี่ยนแปลงไป ดังแสดงรายละเอียดในตารางที่ 4.11

ตารางที่ 4.11 ลักษณะการเปลี่ยนแปลงเมื่อเกิดข้อบกพร่องจากวิธีการ CCL

ข้อบกพร่อง	ลักษณะการเปลี่ยนแปลง
Open circuit	จำนวนกลุ่มเพิ่มขึ้น
Under etch	จำนวนกลุ่มเท่าเดิม แต่ขนาดกลุ่มเพิ่มขึ้น


ตารางที่ 4.11 ลักษณะการเปลี่ยนแปลงเมื่อเกิดข้อบกพร่องจากวิธีการ CCL (ต่อ)

ข้อบกพร่อง	ลักษณะการเปลี่ยนแปลง
Mouse bite	จำนวนกลุ่มเท่าเดิม แต่ขนาดกลุ่มลดลง
Missing conductor	จำนวนกลุ่มเพิ่มขึ้น
Spur	จำนวนกลุ่มเท่าเดิม แต่ขนาดกลุ่มเพิ่มขึ้น
Short circuit	จำนวนกลุ่มลดลง
Spurious copper	จำนวนกลุ่มเพิ่มขึ้น
Over etch	จำนวนกลุ่มเท่าเดิม แต่ขนาดกลุ่มลดลง

4.8 ผลการทดสอบประสิทธิภาพของอัลกอริทึม

สำหรับการทดสอบประสิทธิภาพของอัลกอริทึมในการจำแนกประเภทข้อบกพร่องของ PCB ที่ออกแบบในหัวข้อที่ 3.5.4 จะใช้วิธีการทดสอบตามหัวข้อที่ 3.6.8 โดยจะนำเข้าข้อมูลภาพข้อบกพร่องที่ตรวจจับได้ทั้งหมดจากขั้นตอนการตรวจจับข้อบกพร่องเพื่อให้อัลกอริทึมค่อย ๆ จำแนกข้อบกพร่องแต่ละชนิดออกมา

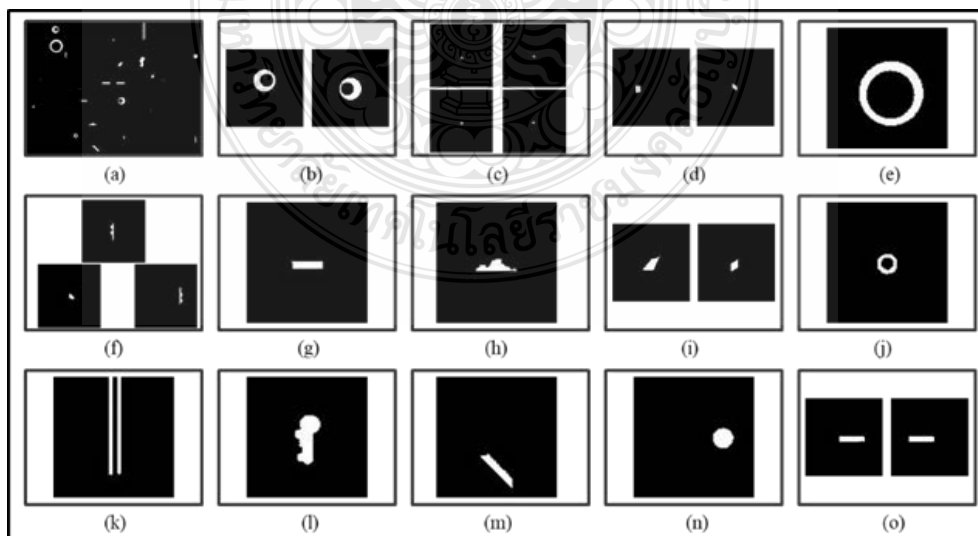
ตัวอย่างผลการทดสอบเมื่อนำเข้าข้อมูลภาพข้อบกพร่องตามรูปที่ 4.16 ที่ได้มาจากภาพ PCB ต้นแบบที่ 1 แสดงในตารางที่ 4.12 และรูปที่ 4.17

														
ข้อบกพร่อง	D01	D02	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14
จำนวน	2	4	2	1	3	1	1	2	1	1	1	1	1	2

รูปที่ 4.16 ตัวอย่างข้อมูลภาพข้อบกพร่องที่ได้มาจากภาพ PCB ต้นแบบที่ 1

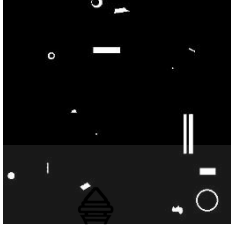
ตารางที่ 4.12 ผลการบันทึกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.16

	D01	D02	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14
Step 2	0	0	0	0	0	0	0	0	1	0	0	0	1	0
Step 4	2	0	0	0	0	0	0	0	1	0	0	0	1	0
Step 5	2	4	0	0	0	0	0	0	1	0	0	0	1	0
Step 6	2	4	0	0	0	0	0	0	1	0	0	1	1	0
Step 9	2	4	0	0	0	0	0	2	1	0	1	1	1	0
Step 11	2	4	2	0	0	1	0	2	1	0	1	1	1	0
Step 13	2	4	2	0	0	1	1	2	1	0	1	1	1	0
Step 14	2	4	2	0	3	1	1	2	1	0	1	1	1	0
Step 15	2	4	2	0	3	1	1	2	1	1	1	1	1	0
Step 16	2	4	2	1	3	1	1	2	1	1	1	1	1	0
Step 17	2	4	2	1	3	1	1	2	1	1	1	1	1	2



รูปที่ 4.17 ผลการจำแนกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.16

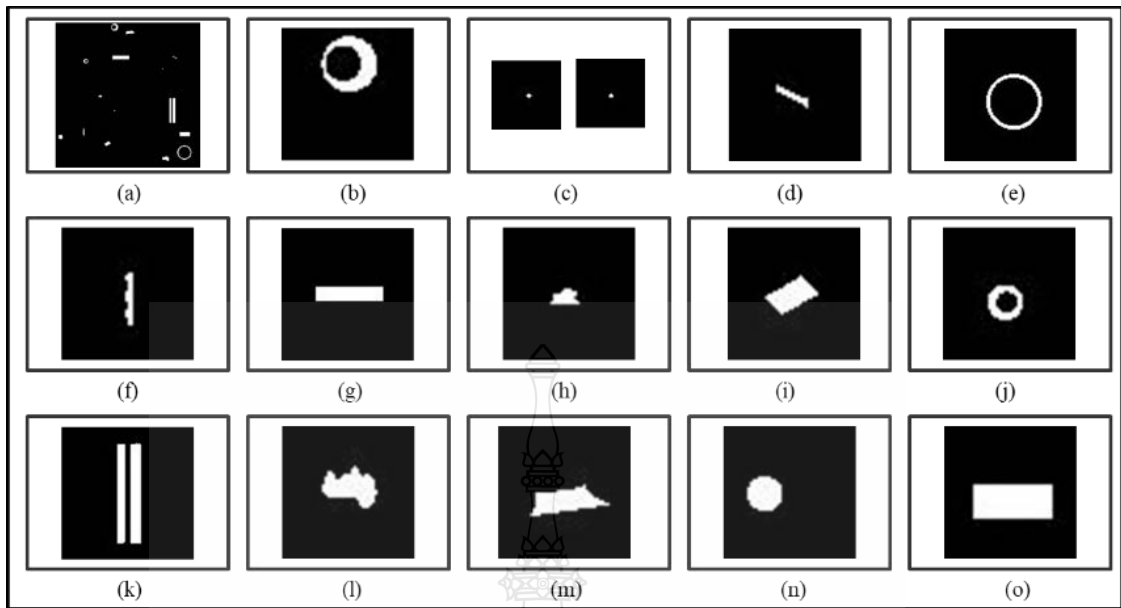
ตัวอย่างผลการทดสอบเมื่อนำเข้าข้อมูลภาพข้อบกพร่องตามรูปที่ 4.18 ที่ได้มาจากภาพ PCB ต้นแบบที่ 2 แสดงในตารางที่ 4.13 และรูปที่ 4.19

														
ข้อบกพร่อง	D01	D02	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14
จำนวน	1	2	1	1	1	1	1	1	1	1	1	1	1	1

รูปที่ 4.18 ตัวอย่างข้อมูลภาพข้อบกพร่องที่ได้มาจากภาพ PCB ต้นแบบที่ 2

ตารางที่ 4.13 ผลการบันทึกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.18

	D01	D02	D03	D04	D05	D06	D07	D08	D09	D10	D11	D12	D13	D14
Step 2	0	0	0	0	0	0	0	0	1	0	0	0	1	0
Step 4	1	0	0	0	0	0	0	0	1	0	0	0	1	0
Step 5	1	2	0	0	0	0	0	0	1	0	0	0	1	0
Step 6	1	2	0	0	0	0	0	0	1	0	0	1	1	0
Step 9	1	2	0	0	0	0	0	1	1	0	1	1	1	0
Step 11	1	2	1	0	0	1	0	1	1	0	1	1	1	0
Step 13	1	2	1	0	0	1	1	1	1	0	1	1	1	0
Step 14	1	2	1	0	1	1	1	1	1	0	1	1	1	0
Step 15	1	2	1	0	1	1	1	1	1	1	1	1	1	0
Step 16	1	2	1	1	1	1	1	1	1	1	1	1	1	0
Step 17	1	2	1	1	1	1	1	1	1	1	1	1	1	1



รูปที่ 4.19 ผลการจำแนกข้อบกพร่องจากภาพข้อบกพร่องในรูปที่ 4.18

ในรูปที่ 4.17 และรูปที่ 4.19 จะแสดงตัวอย่างของข้อบกพร่อง PCB ที่หลากหลาย โดยที่รูป (a) ข้อบกพร่องที่ตรวจจับได้ทั้งหมด, (b) ข้อบกพร่องชนิด Breakout, (c) ข้อบกพร่องชนิด Pin hole, (d) ข้อบกพร่องชนิด Open circuit, (e) ข้อบกพร่องชนิด Under etch, (f) ข้อบกพร่องชนิด Mouse bite, (g) ข้อบกพร่องชนิด Missing conductor, (h) ข้อบกพร่องชนิด Spur, (i) ข้อบกพร่องชนิด Short circuit, (j) ข้อบกพร่องชนิด Wrong size hole, (k) ข้อบกพร่องชนิด Conductor too close, (l) ข้อบกพร่องชนิด Spurious copper, (m) ข้อบกพร่องชนิด Excessive short, (n) ข้อบกพร่องชนิด Missing hole และ (o) ข้อบกพร่องชนิด Over etch จากผลการทดลองแสดงให้เห็นว่าการจำแนกประเภทข้อบกพร่องของ PCB ด้วยอัลกอริทึมที่นำเสนอนี้สามารถจำแนกชนิดของข้อบกพร่องที่เป็นที่รู้จัก 14 ชนิดซึ่งเกิดขึ้นบนแผ่น PCB เปล่าแบบหน้าเดียวหลังจากผ่านกระบวนการกัดและเจาะในกระบวนการผลิตได้อย่างครบถ้วนและมีความถูกต้องแม่นยำสูง

ผู้วิจัยได้ตรวจสอบประสิทธิภาพของรูปแบบอัลกอริทึมในการจำแนกประเภทข้อบกพร่องที่นำเสนอ โดยนำจำนวนชนิดของข้อบกพร่องที่อัลกอริทึมสามารถระบุได้ชัดเจนและอัตราส่วนความถูกต้องแม่นยำในการจำแนกด้วยอัลกอริทึมต่อการจำแนกด้วยสายตาไปเปรียบเทียบกับวิธีการที่พัฒนาขึ้นก่อนหน้านี้ ซึ่งประกอบไปด้วยอัลกอริทึมที่ใช้วิธีการดำเนินการทั่วไป ([16] [17] [19] และ [25]) อัลกอริทึมที่ใช้วิธีการ CNN ([27] และ [28]) และอัลกอริทึมที่ใช้วิธีการ K-NN ([20]) ได้ผลการเปรียบเทียบดังแสดงในตารางที่ 4.14

ตารางที่ 4.14 การเปรียบเทียบประสิทธิภาพของอัลกอริทึมที่นำเสนอกับวิธีการอื่น

Defect type	งานวิจัย	งานวิจัย	งานวิจัย	งานวิจัย	งานวิจัย	งานวิจัย	งานวิจัย	วิธีการที่
	[16]	[17]	[19]	[25]	[27]	[28]	[20]	นำเสนอ
Breakout	-	-	-	-	-	-	75%	100%
Pin hole	✓	✓	-	✓	-	-	75%	96.5%
Open circuit	✓	✓	✓	✓	97.34%	97.74%	75%	99.1%
Under etch	✓	✓	✓	-	-	-	80%	98.4%
Mouse bite	-	✓	-	✓	87.34%	97.94%	70%	97.6%
Missing conductor	-	-	✓	-	-	-	75%	99.1%
Spur	-	-	-	✓	83.34%	93.65%	70%	98.4%
Short circuit	✓	✓	-	✓	96.67%	99.48%	75%	100%
Wrong size hole	-	-	✓	-	-	-	80%	100%
Conductor too close	-	-	-	-	-	-	75%	100%
Spurious copper	-	-	-	-	88.67%	98.52%	75%	100%
Excessive short	-	-	-	-	-	-	80%	100%
Missing hole	✓	✓	✓	-	-	98.96%	75%	100%
Over etch	-	-	✓	-	-	-	75%	97.6%

จากข้อมูลในตารางที่ 4.14 แสดงให้เห็นว่าวิธีการในงานวิจัย [16], [17], [19], [25], [27] และ [28] สามารถระบุได้เฉพาะข้อบกพร่องบางชนิดเท่านั้น ในขณะที่วิธีการในงานวิจัย [20] และ อัลกอริทึมที่นำเสนอสามารถตรวจจับข้อบกพร่องได้ทั้งหมด 14 ชนิด เมื่อพิจารณาจากมุมมองของความถูกต้องแม่นยำจะพบว่าอัตราความถูกต้องของวิธีการในงานวิจัย [20] อยู่ที่ประมาณ 70-80% และของงานวิจัยที่นำเสนออยู่ที่ประมาณ 96.5-100% เนื่องจากวิธีการที่นำเสนอนั้นเป็นการใช้อัลกอริทึม CHT, MR และ CCL ร่วมกัน และจากคุณสมบัติที่ส่งเสริมกันของอัลกอริทึมเหล่านี้จึงช่วยให้ประสิทธิภาพของอัลกอริทึมที่นำเสนอสามารถจำแนกชนิดของข้อบกพร่องได้อย่างครอบคลุมและแม่นยำกว่าวิธีการอื่น

อย่างไรก็ตามในส่วนของความถูกต้องแม่นยำจากงานวิจัย [20] และงานวิจัยที่นำเสนอนั้นไม่สามารถนำมาเปรียบเทียบเพื่อชี้ชัดได้ว่าวิธีการที่นำเสนอมีเปอร์เซ็นต์ความถูกต้องสูงกว่าวิธีการในงานวิจัย [20] เนื่องจากข้อมูลภาพที่นำมาทดสอบเพื่อหาประสิทธิภาพความถูกต้องนั้นใช้ข้อมูลภาพอินพุตซึ่งมีความแตกต่างกันออกไป

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงสรุปผลการวิจัยที่ทำมาทั้งหมด รวมไปถึงข้อเสนอแนะในการนำงานวิจัยนี้ไปพัฒนาหรือประยุกต์ใช้ในอนาคตต่อไป

5.1 สรุปผลการวิจัย

งานวิจัยนี้นำเสนอระบบตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ที่สามารถตรวจจับและจำแนกชนิดของข้อบกพร่อง PCB ที่รู้จักทั้งหมด 14 ชนิดที่เกิดขึ้นบนแผ่นวงจรพิมพ์เปล่าแบบด้านเดียว หลังจากผ่านกระบวนการกัดและเจาะในกระบวนการผลิต เพื่อแก้ไขปัญหาที่เทคโนโลยีการตรวจสอบข้อบกพร่องที่มีอยู่ไม่สามารถจำแนกชนิดของข้อบกพร่อง PCB ทั้งหมดได้ครบถ้วนและถูกต้องแม่นยำเพียงพอในการค้นหาสาเหตุที่แท้จริงของการเกิดข้อบกพร่อง PCB สำหรับกระบวนการผลิตชิ้นส่วนอิเล็กทรอนิกส์ในอุตสาหกรรม โดยระบบที่นำเสนอนี้ประกอบด้วย 3 ขั้นตอนหลัก ได้แก่ 1) การประมวลผลภาพเบื้องต้น 2) การตรวจจับข้อบกพร่อง และ 3) การจำแนกประเภทข้อบกพร่อง ในขั้นตอนของการประมวลผลภาพเบื้องต้นจะทำการแปลงภาพสี RGB ของแผ่นวงจรพิมพ์ที่มีข้อบกพร่องให้เป็นภาพระดับสีเทา จากนั้นใช้การกรองค่ามัธยฐานเพื่อขจัดสัญญาณรบกวนออกจากภาพ ใช้การแปลง Affine เพื่อทำให้ลักษณะทางกายภาพของภาพ PCB ที่มีข้อบกพร่องอยู่ในรูปแบบเดียวกับภาพ PCB ต้นแบบ ในการแบ่งส่วนภาพจะใช้อัลกอริทึมการจัดกลุ่มแบบ Fuzzy C-Mean เพื่อแยกส่วนของภาพที่เป็นลายวงจรออกจากส่วนของภาพที่เป็นพื้นหลัง ในการตรวจจับข้อบกพร่องจะใช้วิธีการลบภาพโดยการเปรียบเทียบภาพ PCB ที่มีข้อบกพร่องกับภาพ PCB ต้นแบบ และจุดเด่นของงานวิจัยนี้อยู่ที่การออกแบบอัลกอริทึมเพื่อจำแนกประเภทของข้อบกพร่อง PCB ที่รู้จักทั้งหมด 14 ชนิดได้อย่างมีประสิทธิภาพโดยใช้การดำเนินการทางคณิตศาสตร์และตรรกศาสตร์ร่วมกับอัลกอริทึม CHT MR และ CCL

ผลการทดลองแสดงให้เห็นว่าระบบที่นำเสนอสามารถบรรลุประสิทธิภาพในการตรวจจับข้อบกพร่องได้ 100 เปอร์เซ็นต์ พร้อมทั้งสามารถระบุตำแหน่ง ขนาด และรูปร่างลักษณะของข้อบกพร่อง PCB ได้อย่างชัดเจน และบรรลุประสิทธิภาพในการจำแนกประเภทของข้อบกพร่อง PCB ที่รู้จักได้ครบทั้งหมด 14 ชนิด โดยมีความถูกต้องแม่นยำในการจำแนกประเภทอยู่ที่ประมาณ 99.05 เปอร์เซ็นต์ และเมื่อทำการเปรียบเทียบกับวิธีการตรวจสอบข้อบกพร่อง PCB อื่น ๆ ที่มีอยู่ ทำให้สรุปได้ว่าระบบตรวจสอบข้อบกพร่อง PCB ที่นำเสนอนี้มีประสิทธิภาพที่ดีกว่าวิธีการอื่น นอกจากนี้

ความสามารถและความถูกต้องแม่นยำของกระบวนการตรวจจับและการจำแนกข้อบกพร่องในงานวิจัยนี้จะมีส่วนช่วยในการปรับปรุงคุณภาพโดยรวมและความน่าเชื่อถือของผลิตภัณฑ์ รวมไปถึงความเป็นไปได้ที่จะลดเวลาสำหรับการผลิตแผ่นวงจรพิมพ์ในอุตสาหกรรม

5.2 ข้อเสนอแนะ

เนื่องด้วยสถาปัตยกรรมของแผ่นวงจรพิมพ์มีการพัฒนาอย่างต่อเนื่องและมีโครงสร้างที่ซับซ้อนมากขึ้น อาจจะทำให้ระบบตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ที่นำเสนอนี้ต้องมีการพัฒนาเพื่อให้รองรับกับแผ่นวงจรพิมพ์ที่พัฒนาขึ้น ดังนั้นงานในอนาคตจึงมีความคิดที่จะมุ่งเน้นไปที่การตรวจสอบข้อบกพร่องแผ่นวงจรพิมพ์ที่มีโครงสร้างและกระบวนการที่ซับซ้อนมากยิ่งขึ้นเพื่อให้ตอบสนองต่อความต้องการของอุตสาหกรรมการผลิตชิ้นส่วนอิเล็กทรอนิกส์ต่อไป



บรรณานุกรม

- [1] Agile Circuit Co., Ltd. (2018). **Basic Knowledge about Single Sided PCB, Double Sided PCB and Multi-layer PCB**. สืบค้นจาก <https://agipcb.blogspot.com/2013/10/pcb.html>
- [2] PCBWay. (2013). **Making a PCB - PCB Manufacture step by step**. สืบค้นจาก <https://www.pcbway.com/pcb-service.html>
- [3] W. Wu, M. J. Wang, & C. Liu. (1996). Automated Inspection of Printed Circuit Boards through Machine Vision. **Computers in Industry**, 28(2), pp. 103-111. doi:10.1016/0166-3615(95)00063-1
- [4] M. Moganti, F. Ercal, C. H. Dagli, & S. Tsunekawa. (1996). Automatic PCB Inspection Algorithms. **Computer vision and image understanding**, 63(2), pp. 287-313.
- [5] R. Heriansyah, S. A. R. Al-Attas, & M. M. i. A. Zabidi. (2002). Segmentation of PCB Images into Simple Generic Patterns using Mathematical Morphology and Windowing Technique. **The National Conference on Computer Graphic & Multimedia (CoGRAMM)**, Melaka, Malaysia, pp. 233-238.
- [6] The Sirindhorn International Thai-German Graduate School of Engineering. (2011). **Image Processing**. สืบค้นจาก <https://sse.tggs.kmutnb.ac.th/index.php/8-research-areas/116-image-processing.html>
- [7] มนตรี กาญจนเดชะ. (2001). **บทที่ 8 การแยกข้อมูลภาพออกเป็นส่วน ๆ**. สืบค้นจาก <http://fivedots.coe.psu.ac.th/~montri/Teaching/image/segment.doc>
- [8] Ronald van Loon. (2018). **Machine learning explained: Understanding supervised, unsupervised, and reinforcement learning**. สืบค้นจาก <https://bigdata-madesimple.com/machine-learning-explained-understanding-supervised-unsupervised-and-reinforcement-learning/>
- [9] N. Gothankar, C. Kambhamettu, & P. Moser. (2019). Circular Hough Transform Assisted CNN Based Vehicle Axle Detection and Classification. **2019 4th International Conference on Intelligent Transportation Engineering (ICITE)**, Singapore, pp. 217-221. doi:10.1109/ICITE.2019.8880232

บรรณานุกรม (ต่อ)

- [10] Y. Sharma & Y. K. Meghrajani. (2014). Brain tumor extraction from MRI image using mathematical morphological reconstruction. **2014 2nd International Conference on Emerging Technology Trends in Electronics, Communication and Networking**, Surat, pp. 1-4.
doi:10.1109/ET2ECN.2014.7044982
- [11] D. Legland & I. Arganda-Carreras. (2019). **Morphological reconstruction**. สืบค้นจาก https://imagej.net/MorphoLibJ#Morphological_reconstruction
- [12] A. Rakhmadi, N. Z. S. Othman, A. Bade, M. S. M. Rahim, & I. M. Amin. (2010). Connected Component Labelling Using Components Neighbours-Scan Labelling Approach. **Journal of Computer Science**, 6(10), pp. 1070-1078.
doi:10.3844/jcssp.2010.1099.1107
- [13] N. K. Khalid, Z. Ibrahim, & M. S. Z. Abidin. (2008). An Algorithm to Group Defects on Printed Circuit Board for Automated Visual Inspection. **International Journal of Simulation: Systems, Science, and Technology**, 9(2), pp. 1-10.
- [14] S. H. I. Putera & Z. Ibrahim. (2010). Printed circuit board defect detection using mathematical morphology and MATLAB image processing tools. **2010 Second International Conference on Education Technology and Computer (ICETC)**, Shanghai, China, vol. 5, pp. 359-363. doi:10.1109/ICETC.2010.5530052
- [15] A. P. S. Chauhan & S. C. Bhardwaj. (2011). Detection of Bare PCB Defects by Image Subtraction Method using Machine Vision. **Proceedings of the World Congress on Engineering**, 2, pp. 1597-1601.
- [16] I. Ibrahim, Z. Ibrahim, K. Khalil, M. M. Mokji, & S. A. B. S. A. Rahman. (2011). An Algorithm for Classification of Five Types of Defects on Bare Printed Circuit Board. **International Journal of Computer Sciences and Engineering Systems**, 5(3), pp. 201-208.

บรรณานุกรม (ต่อ)

- [17] I. Ibrahim, Z. Ibrahim, K. Khalil, M. M. Mokji, S. A. R. S. A. Bakar, N. Mokhtar, & W. K. W. Ahmad. (2012). An Improved Defect Classification Algorithm for Six Printing Defects and Its Implementation on Real Printed Circuit Board Images. **International Journal of Innovative Computing, Information and Control**, 8(5A), pp. 3239-3250.
- [18] S. H. I. Putera, S. F. Dzafaruddin, & M. Mohamad. (2012). MATLAB based defect detection and classification of printed circuit board. **2012 Second International Conference on Digital Information and Communication Technology and it's Applications (DICTAP)**, Bangkok, 2012, pp. 115-119. doi:10.1109/DICTAP.2012.6215366
- [19] B. Kaur, G. Kaur, & A. Kaur. (2014). Detection and Classification of Printed Circuit Board Defects using Image Subtraction Method. **2014 Recent Advances in Engineering and Computational Sciences (RAECS)**, Chandigarh, pp. 1-5. doi:10.1109/RAECS.2014.6799537
- [20] P. P. Londe & S. A. Chavan. (2014). Automatic PCB Defects Detection and Classification using Matlab. **International Journal of Current Engineering and Technology**, 4(3), pp. 2119-2123.
- [21] P. S. Malge & R. S. Nadaf. (2014). PCB Defect Detection, Classification and Localization using Mathematical Morphology and Image Processing Tools. **International Journal of Computer Applications**, 87(9), pp. 40-45. doi:10.5120/15240-3782
- [22] S. Vishwakarma, S. Tikke, C. Manurkar, & A. Thanekar. (2015). PCB Detection and Classification using Digital Image Processing. **International Journal of Advance Research In Science And Engineering**, 4(special 01), pp. 1486-1490.
- [23] M. H. Thigale, S. Gaikwad, P. Nangare, & N. Hule. (2017). Defect detection and classification of printed circuit board using MATLAB. **International Research Journal of Engineering and Technology (IRJET)**, 4(2), pp. 1812-1816.

บรรณานุกรม (ต่อ)

- [24] A. Raj & A. Sajeena. (2018). Defects Detection in PCB Using Image Processing for Industrial Applications. **2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)**, Chandigarh, pp. 1077-1079. doi:10.1109/ICICCT.2018.8473285
- [25] J. Zhu, A. Wu, & X. Liu. (2018). Printed circuit board defect visual detection based on wavelet denoising. **IOP Conference Series: Materials Science and Engineering**, **392**, pp. 1-6. doi:10.1088/1757-899X/392/6/062055
- [26] P. Wei, C. Liu, M. Liu, Y. Gao, & H. Liu. (2018). CNN-based Reference Comparison Method for Classifying Bare PCB Defects. **The Journal of Engineering**, **2018(16)**, pp. 1528-1533. doi:10.1049/joe.2018.8271
- [27] L. Zhang, Y. Jin, X. Yang, X. Li, X. Duan, Y. Sun, & Hong Liu. (2018). Convolutional Neural Network-based Multi-label Classification of PCB Defects. **The Journal of Engineering**, **2018(16)**, pp. 1612-1616. doi:10.1049/joe.2018.8279
- [28] W. Huang, & P. Wei. (2018). A PCB Dataset for Defects Detection and Classification. **Journal of LATEX Class Files**, **14(8)**, pp. 1-9. <https://arxiv.org/pdf/1901.08204>
- [29] L. Zhao, S. Zou, Y. Zhang, M. Huang, Y. Zuo, J. Wang, ... X. Liu. (2019). Segmentation of Activated Sludge Phase Contrast Microscopy Images Using U-Net Deep Learning Model. **Sensors and Materials**, **31(6)**, pp. 2013-2028.
- [30] M. H. Annaby, Y. M. Fouda, & M. A. Rushdi. (2019). Improved Normalized Cross-Correlation for Defect Detection in Printed-Circuit Boards. **IEEE Transactions on Semiconductor Manufacturing**, **32(2)**, pp. 199-211.
- [31] R. Ding, C. Zhang, Q. Zhu, & H. Liu. (2020). Unknown defect detection for printed circuit board based on multi-scale deep similarity measure method. **The Journal of Engineering**, **2020(13)**, pp. 388-393. doi:10.1049/joe.2019.1188
- [32] B. Hu & J. Wang. (2020). Detection of PCB Surface Defects with Improved Faster-RCNN and Feature Pyramid Network. **IEEE Access**, **8**, pp. 108335-108345. doi:10.1109/ACCESS.2020.3001349

บรรณานุกรม (ต่อ)

- [33] T. S. Huang, G. J. Yang, & G. Y. Tang. (1979). A fast two-dimensional median filtering algorithm. **IEEE Transactions on Acoustic, Speech, and Signal Processing**, **27**(1), pp. 13–18. doi:10.1109/TASSP.1979.1163188
- [34] I. Ibrahim, S. A. R. S. A. Bakar, M. M. Mokji, J. A. A. Mukred, Z. M. Yusof, Z. Ibrahim, ... M. S. Mohamad. (2011). A printed circuit board inspection system with defect classification capability. **2nd International Symposium on Innovative Management, Information & Production (IMIP)**, **3**(1), pp. 82-87.
- [35] P. Panuwantichakorn & J. Payakpate. (2014). Using Image Segmentation Technique on the Image of Orchids Paphiopedilum Native Species of Thailand. **Journal of Science and Technology Mahasarakham University**, Special Issue, pp. 277-284.
- [36] N. Otsu. (1979). A Threshold Selection Method from Gray-Level Histograms. **IEEE Transactions on Systems, Man, and Cybernetics**, **9**(1), pp. 62-66.
- [37] M. Saravanan, B. Kalaivani, & R. Geethamani. (2017). Image Segmentation Using K-means clustering Based Thresholding Algorithm. **International Journal of Advanced Technology in Engineering and Science (IJATES)**, **5**(2), pp. 173-183.
- [38] W. Wang, C. Wang, X. Cui, & A. Wang. (2008). Improving Fuzzy C-Means Clustering Based on Adaptive Weighting. **2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery**, Shandong, pp. 62-66. doi:10.1109/FSKD.2008.160
- [39] A. McAndrew, J. Wang, & C. Tseng. (2010). **Introduction to Digital Image Processing with MATLAB**, Asia Edition: Taiwan, Cengage Learning Asia Pte Ltd (Taiwan Branch).

ภาคผนวก



ภาคผนวก ก
โปรแกรม



ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น

```
clc; clear all;
I0 = imread('jk00.jpg');
gray = rgb2gray(I0);
gray = 255 - gray;
BW0 = im2bw(gray);
[w0,h0] = size(BW0);
figure(1); imshow(BW0);
rgb = imread('jkn45.jpg');
gray = rgb2gray(rgb);
gray = 255 - gray;
BW = im2bw(gray);
figure(2); imshow(BW);
w,h] = size(BW);
for i = h:-1:1
    for j = w:-1:1
        if BW(i,j) == 1
            x(2) = j; y(1) = i;
        end
    end
end
for i = 1:h
    for j = w:-1:1
        if BW(j,i) == 1
            x(4) = i; y(2) = j;
        end
    end
end
for i = h:-1:1
```

ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น (ต่อ)

```
for j = 1:w
    if BW(j,i) == 1
        x(1) = i; y(3) = j;
    end
end
end
for i = 1:h
    for j = 1:w
        if BW(i,j) == 1
            x(3) = j; y(4) = i;
        end
    end
end
end
figure(3); imshow(BW);
hold on
plot(x(2),y(1),'-r*');
plot(x(4),y(2),'-g*');
plot(x(1),y(3),'-m*');
plot(x(3),y(4),'-b*');
xmin = x(1);
ymin = y(1);
xw = x(4)-x(1);
yw = y(4)-y(1);
pos = [xmin ymin xw yw];
rectangle('Position',pos,'EdgeColor','y')
hold off
I2 = imcrop(BW,pos);
[m,n] = size(I2);
```

ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น (ต่อ)

```
cx = (x(4)-x(1))/2 + x(1);
cy = (y(4)-y(1))/2 + y(1);
thetar = atan((cy-y(2))/(x(4)-cx));
theta = ceil(45 - thetar*180/pi)
IL2 = imresize(IL2,5);
IL2 = imrotate(IL2,(theta),'loose', 'bilinear');
z1 = imresize(IL2,0.2);
[w2,h2] = size(z1);
for i = h2:-1:1
    for j = w2:-1:1
        if z1(i,j) == 1
            x(2) = j; y(1) = i;
        end
    end
end
for i = 1:h2
    for j = w2:-1:1
        if z1(j,i) == 1
            x(4) = i; y(2) = j;
        end
    end
end
for i = h2:-1:1
    for j = 1:w2
        if z1(j,i) == 1
            x(1) = i; y(3) = j;
        end
    end
end
```

ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น (ต่อ)

```
end
for i = 1:h2
    for j = 1:w2
        if z1(i,j) == 1
            x(3) = j; y(4) = i;
        end
    end
end
end
xmin = x(1);
ymin = y(1);
xw = x(4)-x(1);
yw = y(4)-y(1);
pos = [xmin ymin xw yw];
hold off
IL3 = imcrop(z1,pos);
IL3 = imresize(IL3, [w0 h0], 'nearest');
defect1 = xor(BW0,IL3);
d1 = sum(sum(defect1));
IR2 = imresize(I2,5);
IR2 = imrotate(IR2,(theta-90),'loose', 'bilinear');
z2 = imresize(IR2,0.2);
[w2,h2] = size(z1);
for i = h2:-1:1
    for j = w2:-1:1
        if z2(i,j) == 1
            x(2) = j; y(1) = i;
        end
    end
end
end
```

ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น (ต่อ)

```
end
for i = 1:h2
    for j = w2:-1:1
        if z2(j,i) == 1
            x(4) = i; y(2) = j;
        end
    end
end
end
for i = h2:-1:1
    for j = 1:w2
        if z2(j,i) == 1
            x(1) = i; y(3) = j;
        end
    end
end
end
for i = 1:h2
    for j = 1:w2
        if z2(i,j) == 1
            x(3) = j; y(4) = i;
        end
    end
end
end
xmin = x(1);
ymin = y(1);
xw = x(4)-x(1);
yw = y(4)-y(1);
pos = [xmin ymin xw yw];
hold off
```

ก.1 โปรแกรมการประมวลผลภาพเบื้องต้น (ต่อ)

```
IR3 = imcrop(z2,pos);
IR3 = imresize(IR3, [w0 h0], 'nearest');
defect2 = xor(BW0,IR3);
d2 = sum(sum(defect2));
if d1>d2
    I3 = IR3;
else
    I3 = IL3;
end
figure(4), imshow(I3,'InitialMagnification', 'fit');
```

ก.2 โปรแกรมการแบ่งส่วนภาพด้วย Global Thresholding

```
clc; clear all;
rgb = imread('pcb001.jpg');
figure(1); imshow(rgb);
gray = rgb2gray(rgb);
x = gray;
[m,n] = size(x);
[counts,binLocations] = imhist(x);
T0 = graythresh(gray)*256;
[M1,I1] = max(counts(1:round(T0)));
[M2,I2] = max(counts(round(T0):255));
I2 = I2+round(T0);
[M3,I3] = min(counts(I1:I2));
T = I3+I1-2;
for i = 1:m
    for j = 1:n
        if x(i,j) < T
```


ก.2 โปรแกรมการแบ่งส่วนภาพด้วย Global Thresholding (ต่อ)

```
        rgb(i,j,:) = 0;
    else
        rgb(i,j,:) = 255;
    end
end
end
figure(2); imshow(rgb);
```

ก.3 โปรแกรมการแบ่งส่วนภาพด้วย Adaptive Thresholding

```
clc; clear all;
rgb = imread('pcb001.jpg');
figure(1); imshow(rgb);
gray = rgb2gray(rgb);
x = gray;
[m,n] = size(x);
x1 = x(1:(m/4),:);
x2 = x((m/4)+1:(m/2),:);
x3 = x((m/2)+1:(3*m/4),:);
x4 = x((3*m/4)+1:m,:);
[m1,n1] = size(x1);
T = mean(mean(x1));
T0 = 0;
K1 = x1 <= T;
K2 = x1 > T;
S1 = sum(K1);
S2 = sum(K2);
T1 = sum(x1(K1))/sum(S1);
T2 = sum(x1(K2))/sum(S2);
```

ก.3 โปรแกรมการแบ่งส่วนภาพด้วย Adaptive Thresholding (ต่อ)

```
while abs(T-T0)>10^-8
    T0 = T;
    T = (T1+T2)/2;
    K1 = x1 <= T;
    K2 = x1 > T;
    S1 = sum(K1);
    S2 = sum(K2);
    T1 = sum(x1(K1))/sum(S1);
    T2 = sum(x1(K2))/sum(S2);
end
for i = 1:m1
    for j = 1:n1
        if x1(i,j) < T
            rgb(i,j,:) = 0;
        else
            rgb(i,j,:) = 255;
        end
    end
end
end
[m2,n2] = size(x2);
T = mean(mean(x2));
T0 = 0;
K1 = x2 <= T;
K2 = x2 > T;
S1 = sum(K1);
S2 = sum(K2);
T1 = sum(x2(K1))/sum(S1);
T2 = sum(x2(K2))/sum(S2);
```

ก.3 โปรแกรมการแบ่งส่วนภาพด้วย Adaptive Thresholding (ต่อ)

```
while abs(T-T0)>10^-8
    T0 = T;
    T = (T1+T2)/2;
    K1 = x2 <= T;
    K2 = x2 > T;
    S1 = sum(K1);
    S2 = sum(K2);
    T1 = sum(x2(K1))/sum(S1);
    T2 = sum(x2(K2))/sum(S2);
end
for i = 1:m2
    for j = 1:n2
        if x2(i,j) < T
            rgb(m1+i,j,:) = 0;
        else
            rgb(m1+i,j,:) = 255;
        end
    end
end
end
[m3,n3] = size(x3);
T = mean(mean(x3));
T0 = 0;
K1 = x3 <= T;
K2 = x3 > T;
S1 = sum(K1);
S2 = sum(K2);
T1 = sum(x3(K1))/sum(S1);
T2 = sum(x3(K2))/sum(S2);
```

ก.3 โปรแกรมการแบ่งส่วนภาพด้วย Adaptive Thresholding (ต่อ)

```
while abs(T-T0)>10^-8
    T0 = T;
    T = (T1+T2)/2;
    K1 = x3 <= T;
    K2 = x3 > T;
    S1 = sum(K1);
    S2 = sum(K2);
    T1 = sum(x3(K1))/sum(S1);
    T2 = sum(x3(K2))/sum(S2);
end
for i = 1:m3
    for j = 1:n3
        if x3(i,j) < T
            rgb(m1+m2+i,j,:) = 0;
        else
            rgb(m1+m2+i,j,:) = 255;
        end
    end
end
end
[m4,n4] = size(x4);
T = mean(mean(x4));
T0 = 0;
K1 = x4 <= T;
K2 = x4 > T;
S1 = sum(K1);
S2 = sum(K2);
T1 = sum(x4(K1))/sum(S1);
T2 = sum(x4(K2))/sum(S2);
```

ก.3 โปรแกรมการแบ่งส่วนภาพด้วย Adaptive Thresholding (ต่อ)

```
while abs(T-T0)>10^-8
    T0 = T;
    T = (T1+T2)/2;
    K1 = x4 <= T;
    K2 = x4 > T;
    S1 = sum(K1);
    S2 = sum(K2);
    T1 = sum(x4(K1))/sum(S1);
    T2 = sum(x4(K2))/sum(S2);
end
for i = 1:m2
    for j = 1:n
        if x4(i,j) < T
            rgb(m1+m2+m3+i,j,:) = 0;
        else
            rgb(m1+m2+m3+i,j,:) = 255;
        end
    end
end
end
figure(2); imshow(rgb);
```

ก.4 โปรแกรมการแบ่งส่วนภาพด้วย K-means Clustering

```
clc; clear all;
rgb = imread('pcb004.jpg');
figure(1); imshow(rgb);
gray = rgb(:,:,1);
figure(3); imshow(gray);
x = gray;
```

ก.4 โปรแกรมการแบ่งส่วนภาพด้วย K-means Clustering (ต่อ)

```
[m,n] = size(x);
x = double(x);
t1 = zeros(m,n);
t2 = zeros(m,n);
c1 = double((mean(mean(x)) + min(min(x)))/2);
c2 = double((mean(mean(x)) + max(max(x)))/2);
m1 = 0;
m2 = 0;
while ((abs(c1-m1)>10^-3) || (abs(c2-m2)>10^-3))
    m1 = c1;
    m2 = c2;
    k1 = 0;
    k2 = 0;
    for i = 1:m
        for j = 1:n
            err1 = double(abs(x(i,j)-c1));
            err2 = double(abs(x(i,j)-c2));
            if err1 <= err2
                t1(i,j) = x(i,j);
                t2(i,j) = 0;
                k1 = k1 + 1;
            else
                t1(i,j) = 0;
                t2(i,j) = x(i,j);
                k2 = k2 + 1;
            end
        end
    end
end
end
```

ก.4 โปรแกรมการแบ่งส่วนภาพด้วย K-means Clustering (ต่อ)

```
c1 = sum(sum(t1))/(k1);  
c2 = sum(sum(t2))/(k2);  
end  
for i = 1:m  
    for j = 1:n  
        if t2(i,j) > 0  
            rgb(i,j,:) = 255;  
        else  
            rgb(i,j,:) = 0;  
        end  
    end  
end  
end  
figure(2); imshow(rgb);
```

ก.5 โปรแกรมการแบ่งส่วนภาพด้วย Fuzzy C-means Clustering

```
clc; clear all;  
rgb = imread('pcb001.jpg');  
figure(1); imshow(rgb);  
gray = rgb2gray(rgb);  
x = gray;  
[m,n] = size(x);  
x = double(x);  
t1 = zeros(m,n);  
t2 = zeros(m,n);  
u1 = zeros(m,n);  
u2 = zeros(m,n);  
c1 = double((mean(mean(x)) + min(min(x)))/2);  
c2 = double((mean(mean(x)) + max(max(x)))/2);
```

ก.5 โปรแกรมการแบ่งส่วนภาพด้วย Fuzzy C-means Clustering (ต่อ)

```
m1 = 0;
m2 = 0;
while ((abs(c1-m1)>10^-3) || (abs(c2-m2)>10^-3))
    m1 = c1;
    m2 = c2;
    k1 = 0;
    k2 = 0;
    tt1 = zeros(m,n);
    tt2 = zeros(m,n);
    for i = 1:m
        for j = 1:n
            err1 = double(abs(x(i,j)-c1));
            err2 = double(abs(x(i,j)-c2));
            if err1 <= err2
                t1(i,j) = x(i,j);
                t2(i,j) = 0;
                k1 = k1 + 1;
                u1(i,j) = 1 / ((err1 / err1)^2 + (err1 / err2)^2);
                u2(i,j) = 1 / ((err2 / err1)^2 + (err2 / err2)^2);
            else
                t1(i,j) = 0;
                t2(i,j) = x(i,j);
                k2 = k2 + 1;
                u1(i,j) = 1 / ((err1 / err1)^2 + (err1 / err2)^2);
                u2(i,j) = 1 / ((err2 / err1)^2 + (err2 / err2)^2);
            end
        end
    end
end
end
```


ก.5 โปรแกรมการแบ่งส่วนภาพด้วย Fuzzy C-means Clustering (ต่อ)

```
for i = 1:m
for j = 1:n
    if u1(i,j) >= u2(i,j)
        tt1(i,j) = t1(i,j) * u1(i,j);
        u2(i,j) = 0;
    else
        tt2(i,j) = t2(i,j) * u2(i,j);
        u1(i,j) = 0;
    end
end
end
c1 = sum(sum(tt1))/sum(sum(u1));
c2 = sum(sum(tt2))/sum(sum(u2));
end
for i = 1:m
for j = 1:n
    if t2(i,j) > 0
        rgb(i,j,:) = 255;
    else
        rgb(i,j,:) = 0;
    end
end
end
end
figure(2); imshow(rgb);
```

ก.6 โปรแกรมคำนวณหาประสิทธิภาพของการแบ่งส่วนภาพ

```
clc; clear all;
GT = imread('j0100.jpg');
GT = im2bw(GT,0.5);
xx = imread('j0101.jpg');
xx = im2bw(xx,0.5);
[m,n] = size(GT);
TP = 0; TN = 0; FP = 0; FN = 0; In = 0;
for i = 1:m
    for j = 1:n
        if (GT(i,j) - xx(i,j) == 0)&&(GT(i,j)==1)
            TP = TP+1;
        elseif (GT(i,j) - xx(i,j) == 0)&&(GT(i,j)==0)
            TN = TN+1;
        elseif GT(i,j) - xx(i,j) > 0
            FN = FN+1;
        else
            FP = FP+1;
        end
    end
end
for i = 1:m
    for j = 1:n
        if (GT(i,j) - xx(i,j) == 0)
            In = In+1;
        end
    end
end
Un = TP + FP + TN +FN;
```

ก.6 โปรแกรมคำนวณหาประสิทธิภาพของการแบ่งส่วนภาพ (ต่อ)

```
Acc = (TP + TN) / (TP + FP + TN + FN) * 100;
IoU = In / Un;
Pre = TP / (TP + FP) * 100;
Rec = TP / (TP + FN) * 100;
GT = imread('j0100.jpg');
GT = im2bw(GT,0.5);
xx = imread('j0101.jpg');
xx = im2bw(xx,0.5);
[m,n] = size(GT);
err = zeros(m,n);
for i = 1:m
    for j = 1:n
        err(i,j) = (xx(i,j) - GT(i,j))^2;
    end
end
E = sum(sum(err))/(m*n);
MSE = sum(sum(err))/(m*n) * 100;
PSNR = 10*log10((255^2)/E);
```

ก.7 โปรแกรมการตรวจจับขอบภาพร่อง PCB

```
clc; clear;
I01 = imread('PCBgood.jpg');
[x0,y0,~] = size(I01);
A01 = I01(10:x0-10, 10:y0-10,:);
figure(1); imshow(A01); %% Good Image %%
A02 = rgb2gray(A01);
figure(2); imshow(A02); %% Good Image (Gray) %%
A03 = im2bw(255 - A02);
```

ก.7 โปรแกรมการตรวจจับข้อบกพร่อง PCB (ต่อ)

```
figure(3); imshow(A03); %% Good Image (Negative) %%
[x,y,~] = size(A03);
A04 = imfill(A03,'holes');
figure(4); imshow(A04); %% Flood-fill Good Image (Negative) %%
imwrite(A04,'fm.jpg');
A05 = xor(A03,A04);
figure(5); imshow(A05); %% Hold Image %%
[Z01,nZ01] = bwlabel(A05);
for i = 1:nZ01
    fZ01{i} = (Z01 == i);
    aZ01(i) = sum(sum(fZ01{i}));
end
bH = max(max(aZ01)); %% The biggest Hole size %%
rH = ceil(sqrt(bH)/pi); %% The biggest Hole radius %%
[centers,rHP] = imfindcircles(A04,[12 21]);
Blk1 = zeros(size(A03));
Blk2 = zeros(size(A03));
[reg,~] = size(centers);
for i=1:reg
    Blk1 = Blk1 + Blk2;
    Blk3 = [centers(i,1) centers(i,2) rHP(i)-1];
    Blk4 = insertShape(Blk1,'Filledcircle',Blk3,'color','white');
    Blk2 = rgb2gray(Blk4);
end
HolePad = im2bw(Blk2);
rmHP = ceil(max(max(rHP))); %% The biggest Hole Pad %%
TR = (rmHP - rH)/2; %% The biggest Trace size %%
I02 = imread('PCBdefectAll.jpg');
```

ก.7 โปรแกรมการตรวจจับข้อบกพร่อง PCB (ต่อ)

```
B01 = I02(10:x0-10, 10:y0-10,:);
figure(6); imshow(B01); %% Defect Image %%
B02 = rgb2gray(B01);
figure(7); imshow(B02); %% Defect Image (Gray) %%
B03 = im2bw(255 - B02);
figure(8); imshow(B03); %% Defect Image (Negative) %%
B04 = imfill(B03,'holes');
figure(9); imshow(B04); %% Flood-fill Defect Image (Negative) %%
imwrite(B04,'gm.jpg');
C01 = and(A03,not(B03));
C02 = zeros(x,y);
[Z02,nZ02] = bwlabel(C01);
for i = 1:nZ02 %% Filter %%
    fZ02{i} = (Z02 == i);
    aZ02(i) = sum(sum(fZ02{i}));
    if aZ02(i) <= 5
        C03{i} = zeros(x,y);
    else
        C03{i} = fZ02{i};
    end
    C02 = or(C02,C03{1,i});
end
figure(10); imshow(C02); %% Positive Defect Image %%
imwrite(C02,'C02.jpg');
C04 = and(B03,not(A03));
C05 = zeros(x,y);
[Z03,nZ03] = bwlabel(C04);
for i = 1:nZ03 %% Filter %%
```

ก.7 โปรแกรมการตรวจจับข้อบกพร่อง PCB (ต่อ)

```
fZ03{i} = (Z03 == i);
aZ03(i) = sum(sum(fZ03{i}));
if aZ03(i) <= 5
    C06{i} = zeros(x,y);
else
    C06{i} = fZ03{i};
end
C05 = or(C05,C06{1,i});
end
figure(11); imshow(C05); %% Negative Defect Image %%
imwrite(C05,'C05.jpg');
C07 = or(C02,C05);
figure(12); imshow(C07);
%% All Defect Image %% imwrite(C07,'All Defects.jpg');
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB

```
%% โปรแกรมจะต่อเนื่องจากโปรแกรมการตรวจจับข้อบกพร่อง PCB %%
C08 = and(C07,HolePad);
figure(13); imshow(C08); %% All Defect in Hole Pad %%
C09 = and(C02,HolePad);
figure(14); imshow(C09); %% Positive Defect in Hole Pad %%
C10 = and(C05,HolePad);
figure(15); imshow(C10); %% Negative Defect in Hole Pad %%
C11 = imfill(C10,'holes');
C12 = imfill(C09,'holes');
figure(100); imshow(C11);
D09 = zeros(x,y);
[Z04,nZ04] = bwlabel(C10);
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
[Z05,~] = bwlabel(C11);
for i = 1:nZ04 %% Number of Defect %%
    fZ04{i} = (Z04 == i);
    fZ05{i} = (Z05 == i);
    aZ04(i) = sum(sum(fZ04{i}));
    aZ05(i) = sum(sum(fZ05{i}));
    if aZ05(i) > aZ04(i)
        fZ06{i} = fZ04{i};
    else
        fZ06{i} = zeros(x,y);
    end
    D09 = or(D09,fZ06{1,i});
end
figure(16); imshow(D09);
9 Wrong size hole imwrite(D09,'D09.jpg');
D13 = xor(C10,D09);
figure(17); imshow(D13);
13 Missing hole imwrite(D13,'D13.jpg');
D01 = zeros(x,y);
[Z07,nZ07] = bwlabel(C09);
[Z08,~] = bwlabel(C12);
for i = 1:nZ07 %% Number of Defect %%
    fZ07{i} = (Z07 == i);
    fZ08{i} = (Z08 == i);
    aZ07(i) = sum(sum(fZ07{i}));
    aZ08(i) = sum(sum(fZ08{i}));
    if aZ08(i) > aZ07(i)
        fZ09{i} = fZ07{i};
    end
end
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
else
    fZ09{i} = zeros(x,y);
end
D01 = or(D01,fZ09{1,i});
end
figure(18); imshow(D01);
%%% 1 Breakout %%% imwrite(D01,'D01.jpg');
dHP = or(D09,or(D13,D01));
figure(19); imshow(dHP); %% Defects in Hole Pad (3 Types) %%
D02 = xor(D01,and(C02,xor(B03,B04)));
figure(20); imshow(D02);
%%% 2 Pin Hole %%% imwrite(D02,'D02.jpg');
C13 = or(D01,or(D02,or(D09,D13))); %% D01, D02, D09, D13 %%
C14 = xor(C13,(xor(A05,xor(B04,B03))));
C15 = or(C07,C14);
D12 = zeros(x,y);
[Z10,nZ10] = bwlabel(C07);
[Z11,~] = bwlabel(C15);
for i = 1:nZ10
    fZ10{i} = (Z10 == i);
    fZ11{i} = (Z11 == i);
    aZ10(i) = sum(sum(fZ10{i}));
    aZ11(i) = sum(sum(fZ11{i}));
    if aZ11(i) > aZ10(i)
        fZ12{i} = fZ10{i};
    else
        fZ12{i} = zeros(x,y);
    end
end
```


ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
D12 = or(D12,fZ12{1,i});
end
figure(21); imshow(D12);
%%% 12 Excessive short %%% imwrite(D12,'D12.jpg');
C16 = xor(C05,xor(D09,xor(D13,D12)));
figure(22); imshow(C16); %% D04, D07, D08, D11 + D10 %%
C17 = xor(C02,xor(D01,D02));
figure(23); imshow(C17); %% D03, D05, D06, D14 + D10 %%
%% Number of group and Area of each group for Good PCB %% [Z100,nZ100]
= bwlabel(A03);
for i = 1:nZ100
    fZ100{i} = (Z100 == i);
    aZ100(i) = sum(sum(fZ100{i}));
end
D08 = zeros(x,y);
D11 = zeros(x,y);
[Z13,nZ13] = bwlabel(C16);
for i = 1:nZ13
    fZ13{i} = (Z13 == i);
    dZ13{i} = xor(A03,fZ13{i});
    [Z14,nZ14] = bwlabel(dZ13{i});
    if nZ14 < nZ100
        D08 = or(D08,fZ13{i});
    elseif nZ14 > nZ100
        D11 = or(D11,fZ13{i});
    end
end
end
figure(24); imshow(D08);
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
%%% 8 Short circuit %%% imwrite(D08,'D08.jpg');
figure(25); imshow(D11);
%%% 11 Spurious copper %%% imwrite(D11,'D11.jpg');
C18 = xor(C16,xor(D08,D11));
figure(26); imshow(C18); %% D04, D07 + D10 %%
D03 = zeros(x,y);
D06 = zeros(x,y);
[Z15,nZ15] = bwlabel(C17);
for i = 1:nZ15
    fZ15{i} = (Z15 == i);
    aZ15(i) = sum(sum(fZ15{i}));
    dZ15{i} = xor(A03,fZ15{i});
    [Z16,nZ16] = bwlabel(dZ15{i});
    if nZ16 > nZ100
        if aZ15(i) < ceil(TR)^2
            D03 = or(D03,fZ15{i});
        else
            D06 = or(D06,fZ15{i});
        end
    end
end
end
figure(27); imshow(D03);
%%% 3 Open circuit %%% imwrite(D03,'D03.jpg');
figure(28); imshow(D06);
%%% 6 Missing Conductor %%% imwrite(D06,'D06.jpg');
C19 = xor(C17,xor(D03,D06));
figure(29); imshow(C19); %% D05, D14 + D10 %%
%% Find defect is not symmetry %%
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
D07 = zeros(x,y);
[Z23,nZ23] = bwlabel(C18);
for i = 1:nZ23
    fZ23{i} = (Z23 == i);
    [r1,c1] = find(Z23 == i);
    fZ24{i} = fZ23{i}(min(r1):max(r1),min(c1):max(c1));
    fZ25{i} = fliplr(fZ24{i});
    fZ26{i} = flipud(fZ24{i});
    fZ27{i} = xor(fZ24{i},fZ25{i});
    fZ28{i} = xor(fZ24{i},fZ26{i});
    are1(i) = (max(r1)-min(r1)+1)*(max(c1)-min(c1)+1);
    pix1(i) = sum(sum(fZ27{i})) + sum(sum(fZ28{i}));
    per1(i) = pix1(i)/are1(i)*100;
    if per1(i) > 25 %%% Error from folding %%%
        fZ29{i} = (Z23 == i);
        fZ30{i} = zeros(x,y);
        D07 = or(D07,fZ29{i});
    end
end
figure(29); imshow(D07);
%% 7 Spur %% imwrite(D07,'D07.jpg');
D05 = zeros(x,y);
[Z31,nZ31] = bwlabel(C19);
for i = 1:nZ31
    fZ31{i} = (Z31 == i);
    [r2,c2] = find(Z31 == i);
    fZ32{i} = fZ31{i}(min(r2):max(r2),min(c2):max(c2));
    fZ33{i} = fliplr(fZ32{i});
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
fZ34{i} = flipud(fZ32{i});
fZ35{i} = xor(fZ32{i},fZ33{i});
fZ36{i} = xor(fZ32{i},fZ34{i});
are2(i) = (max(r2)-min(r2)+1)*(max(c2)-min(c2)+1);
pix2(i) = sum(sum(fZ35{i})) + sum(sum(fZ36{i}));
per2(i) = pix2(i)/are2(i)*100;
if per2(i) > 25 %%% Error from folding %%%
    fZ37{i} = (Z31 == i);
    fZ38{i} = zeros(x,y);
    D05 = or(D05,fZ37{i});
end
end
figure(30); imshow(D05);
%% 5 Mouse bite %% imwrite(D05,'D05.jpg');
C20 = xor(C18,D07);
figure(31); imshow(C20); %% D04 + D10 %%
C21 = xor(C19,D05);
figure(32); imshow(C21); %% D14 + D10 %%
C22 = or(C20,C21);
figure(33); imshow(C22);
for i = 1:nZ100
    fx{i} = zeros(x,y);
end
zN = zeros(1,nZ100);
zP = zeros(1,nZ100);
[Z40,nZ40] = bwlabel(C22);
for i = 1:nZ40
    fZ40{i} = (Z40 == i);
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
dZ40{i} = xor(A03,fZ40{i});
[Z41,nZ41] = bwlabel(dZ40{i});
for j = 1:nZ100
    fZ41{j} = (Z41 == j);
    aZ41{j} = sum(sum(fZ41{j}));
    if aZ41{j} - aZ100(j) > 10
        fx{j} = xor(fx{j},fZ40{i});
        zN(j) = 1;
    elseif aZ100(j) - aZ41{j} > 10
        fx{j} = xor(fx{j},fZ40{i});
        zP(j) = 1;
    end
end
end
end
PN = zeros(1,nZ100);
D10 = zeros(x,y);
D04 = zeros(x,y);
D14 = zeros(x,y);
for i = 1:nZ100
    PN(i) = zN(i) + zP(i);
    if PN(i) == 2
        D10 = fx{i};
    elseif zN(i) == 1
        D04 = fx{i};
    elseif zP(i) == 1
        D14 = fx{i};
    end
end
end
```

ก.8 โปรแกรมการจำแนกชนิดข้อบกพร่อง PCB (ต่อ)

```
figure(34); imshow(D10);  
%%% 10 Conductor too close %%% imwrite(D10,'D10.jpg');  
figure(35); imshow(D04);  
%%% 4 Under etch %%% imwrite(D04,'D04.jpg');  
figure(36); imshow(D14);  
%%% 14 Over etch %%% imwrite(D14,'D14.jpg');
```



ประวัติผู้เขียน

ชื่อ-สกุล	นายจักรกฤษ อ่อนชื่นจิตร
วัน เดือน ปีเกิด	วันที่ 16 เดือน มิถุนายน พ.ศ. 2517
ที่อยู่	63/65 หมู่บ้านสีวลี 3 หมู่ที่ 7 ตำบลคลองสวนพูล อำเภอพระนครศรีอยุธยา จังหวัดพระนครศรีอยุธยา 13000
การศึกษา	- ปริญญาตรี วศ.บ.(วิศวกรรมไฟฟ้า - อิเล็กทรอนิกส์) คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี - ปริญญาโท วศ.ม.(วิศวกรรมไฟฟ้า) คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี
ประสบการณ์ทำงาน	อาจารย์ประจำภาควิชาวิศวกรรมอิเล็กทรอนิกส์และโทรคมนาคม คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลธัญบุรี ตั้งแต่ พ.ศ. 2539 จนถึงปัจจุบัน
เบอร์โทรศัพท์	08-0061-3450
อีเมล	jakkrit.o@en.rmutt.ac.th

