

การพัฒนาเทคนิคการหาขอบแบบใหม่ด้วยพาทิเคอลสวอมออปติมัยเซชัน
สำหรับตรวจสอบรอยเชื่อมบัดกรี

DEVELOPMENT OF NEW EDGE-DETECTION FILTER BASED ON PARTICLE
SWARM OPTIMIZATION: AN APPLICATION SOLDERING

ปารวี ศรีรัตน์
PARAWEE SRIRAT

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมระบบและข้อมูล
วิทยาลัยนวัตกรรมการผลิตขั้นสูง
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2559

KMITL-2016-AMI-M-001-002

การพัฒนาเทคนิคการหาขอบแบบใหม่ด้วยพาทิโคลสวอมออปติมัยเซชัน
สำหรับตรวจสอบรอยเชื่อมบัดกรี

DEVELOPMENT OF NEW EDGE-DETECTION FILTER BASED ON PARTICLE
SWARM OPTIMIZATION: AN APPLICATION SOLDERING

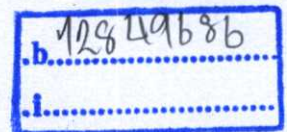


T147127

ปารวี ศรีรัตน์

PARAWEE SRIRAT

เลขหมู่.....
เลขทะเบียน..... I47127
วัน,เดือน,ปี 3 ก.ค. 2560



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชาวิศวกรรมระบบและข้อมูล

วิทยาลัยนวัตกรรมการผลิตขั้นสูง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2559

KMITL-2016-AMI-M-001-002

DEVELOPMENT OF NEW EDGE-DETECTION FILTER BASED ON PARTICLE
SWARM OPTIMIZATION: AN APPLICATION SOLDERING

PARAWEE SRIRAT

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF INFORMATION AND SYSTEM ENGINEERING
COLLEGE OF ADVANCED MANUFACTURING INNOVATION
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2016

KMITL-2016-AMI-M-001-002

COPYRIGHT 2016

COLLEGE OF ADVANCED MANUFACTURING INNOVATION

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

วิทยาลัยนวัตกรรมการผลิตขั้นสูง
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ใบรับรองวิทยานิพนธ์

หัวข้อวิทยานิพนธ์ การพัฒนาเทคนิคการหาขอบแบบใหม่ด้วยพาทิเคอลสวอมมออปติมัยเซชัน
สำหรับตรวจสอบรอยเชื่อมบัดกรี

Thesis Title DEVELOPMENT OF NEW EDGE-DETECTION FILTER BASED ON
PARTICLE SWARM OPTIMIZATION: AN APPLICATION SOLDERING

นักศึกษา นางสาวปารวี ศรีรัตน์


รหัสประจำตัว 55690309

ปริญญา วิศวกรรมศาสตรมหาบัณฑิต

สาขาวิชา วิศวกรรมระบบและข้อมูล

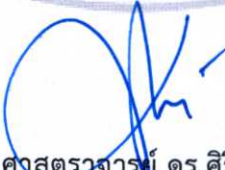
อาจารย์ที่ปรึกษาวิทยานิพนธ์ ผศ.ดร.อนรรฆพล แสนทน

หมายเลขวิทยานิพนธ์ KMITL-2016-AMI-M-001-002

คณะกรรมการสอบวิทยานิพนธ์		ลายมือชื่อ
รองศาสตราจารย์ ดร.สมยศ	เกียรติวินชิวไล	
รองศาสตราจารย์ ดร.ปิติเขต	สุรักษา	
ผู้ช่วยศาสตราจารย์ ดร.ถวิดา	มณีวรรณ	
ผู้ช่วยศาสตราจารย์ ดร.ชัยวัฒน์	หนูทอง	
ผู้ช่วยศาสตราจารย์ ดร.อนรรฆพล	แสนทน	

วัน/เดือน/ปี ที่สอบ 27 มิถุนายน 2559 เวลา 13.00 - 15.00 น.
สถานที่สอบ อาคารเฉลิมพระเกียรติ 55 พรรษา สมเด็จพระเทพรัตนราชสุดาฯ สยามบรมราชกุมารี

วิทยาลัยนวัตกรรมการผลิตขั้นสูง รับรองแล้ว


(ผู้ช่วยศาสตราจารย์ ดร.ศิริเดช บุญแสง)
คณบดี วิทยาลัยนวัตกรรมการผลิตขั้นสูง

วันที่ 22 กรกฎาคม พ.ศ. 2559

หัวข้อวิทยานิพนธ์	การพัฒนาเทคนิคการหาขอบแบบใหม่ด้วยพาทิคอลสวอม ออปติไมซ์เซชันสำหรับตรวจสอบรอยเชื่อมบัดกรี
นักศึกษา	นางสาวปารวี ศรีรัตน์
รหัสนักศึกษา	55690309
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมระบบและข้อมูล
พ.ศ.	2559
อาจารย์ที่ปรึกษาวิทยานิพนธ์	ผศ.ดร.อนรรฆพล แสนทน

บทคัดย่อ

วงจรไฟฟ้าประกอบด้วยชิ้นส่วนอิเล็กทรอนิกส์ขนาดเล็กหลายชนิดมาบัดกรีรวมกันบนแผงวงจร การตรวจวัดขนาดของการบัดกรีสามารถตรวจวัดได้ 2 วิธี วิธีแรกคือการตรวจวัดแบบทำลายโดยใช้กล้องจุลทรรศน์ ซึ่งเป็นวิธีที่มีความแม่นยำที่สุดแต่ชิ้นงานไม่สามารถนำกลับมาใช้ใหม่ วิธีที่สองใช้ภาพเอกซเรย์สำหรับตรวจวัดข้อดีคือไม่ต้องทำลายชิ้นงาน แต่มีปัญหาเรื่องความถูกต้องในการตรวจวัด เนื่องจากความคลุมเครือของภาพเอกซเรย์ทำให้ยากต่อการหาขอบเขตวัตถุในรูปภาพ ดังนั้นงานวิจัยนี้จึงสนใจการพัฒนาตัวกรองตรวจหาขอบสำหรับตรวจวัดรอยเชื่อมบัดกรีร่วมกับวิธีการพาทิคอลสวอมออปติไมซ์เซชัน (Particle Swarm Optimization: PSO) เพื่อเพิ่มความถูกต้องของการวัดขนาดผลการทดลองจะแสดงประสิทธิภาพของเทคนิคการนำเสนอซึ่งให้ผลลัพธ์ที่ดีกว่าเทคนิคการหาขอบแบบทั่วไป

Thesis Title	Development of new edge-detection filter based on particle swarm optimization: an application soldering
Student	Ms.Parawee Srirat
Student ID.	55690309
Degree	Master of Engineering
Program	Information and System Engineering
Year	2016
Thesis Advisor	Asst. Prof. Dr. Anakkapon Saenthon

Abstract

The electrical circuit is composed of very small parts. It is an important process for soldering joint connecting electronic components together. The size of soldering joint can be measured by two ways. The first method is a destructive measurement using of the microscope, which gives the best accuracy, but the sample could not be reused. The second method uses X-ray image for detection. However, the accuracy of measurement is the problem due to the ambiguity of X-ray image, which is difficult to find borders. Therefore, this research is focusing on developing a new edge detection filter for measuring solder joint by using particle swarm optimization (PSO) method to increase the accuracy of the measured size. The experiment results will show the performance of the proposed technique. This technique is better in finding a common edge.

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 จุดมุ่งหมายและวัตถุประสงค์ของการศึกษา	2
1.3 สมมุติฐานของการศึกษา.....	2
1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย	2
1.5 ขอบเขตการวิจัย.....	3
1.6 ขั้นตอนการศึกษา.....	3
บทที่ 2 ทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้อง	4
2.1 แผงวงจรพิมพ์	4
2.2 การบัดกรี.....	8
2.3 เอกซเรย์.....	10
2.4 การประมวลผลรูปภาพ	13
2.5 พาทิเคิลสวอมมออปติไมเซชัน	18
2.6 งานวิจัยที่เกี่ยวข้อง.....	21
บทที่ 3 การออกแบบและขั้นตอนการทดลอง	23
3.1 เครื่องมือวิจัย.....	23
3.2 ขั้นตอนการทำวิจัย	23
บทที่ 4 ผลการทดลอง และการวิเคราะห์ผลการทดลอง	30
4.1 ผลการแยกขอบเขตรอยบัดกรีจากพื้นหลัง	30
4.2 ผลการวัดความกว้างและความยาวของรอยบัดกรี.....	31
4.3 วิเคราะห์ผลการทดลอง	33

สารบัญ (ต่อ)

	หน้า
บทที่ 5 สรุปผลการวิจัย และข้อเสนอแนะ	36
บรรณานุกรม	37
ภาคผนวก	39
ภาคผนวก ก ภาพผลการทดลอง	40
ภาคผนวก ข ผลการวัดขนาดรอยบัดกรี	46
ภาคผนวก ค ค่าความคลาดเคลื่อนจากการวัดขนาดรอยบัดกรี.....	49
ภาคผนวก ง Source Code.....	51
ประวัติผู้เขียน	73

สารบัญตาราง

ตารางที่	หน้า
4.1 แสดงผลการวัดค่าความกว้างความยาวด้วยPSO-Edge	31
4.2 ค่าความคลาดเคลื่อนด้านกว้างของPSO-Edge แคนนี่ พรีวิต และโซเบล	32
4.3 ค่าความคลาดเคลื่อนด้านยาวของPSO-Edge แคนนี่ พรีวิต และโซเบล	32
4.4 ค่าความคลาดเคลื่อนรวมของPSO-Edge แคนนี่ พรีวิต และโซเบล.....	32
4.5 เวลาในการประมวลผลด้วยPSO-Edge แคนนี่ พรีวิต และโซเบล.....	32
4.6 แสดงภาพที่ได้จากการวิเคราะห์ขอบเขตด้วยวิธีต่างๆ	33

สารบัญรูปร่างภาพ

รูปที่	หน้า
2.1 ตัวอย่างการนำสายไฟมาบัดกรีเชื่อมต่อกับขั้วของอุปกรณ์แต่ละตัวเข้าด้วยกัน.....	4
2.2 แผ่นวงจรพิมพ์ที่ใช้กับวงจรวิทยุ.....	5
2.3 แผงวงจรพิมพ์แบบพินอลิก	6
2.4 แผงวงจรพิมพ์แบบกลาสอีพอกซี.....	6
2.5 แผงวงจรพิมพ์แบบเคมวัน.....	7
2.6 แผงวงจรสำหรับกล่องทวนสัญญาณในไมโครเวฟ.....	7
2.7 แผ่นวงจรพิมพ์แบบยืดหยุ่น.....	8
2.8 การบัดกรีในงานอุตสาหกรรม.....	8
2.9 ตัวอย่างการตรวจวัดชิ้นงานแบบทำลายโดยกล้องจุลทรรศน์.....	9
2.10 ภาพเอกซเรย์ภาพแรก	10
2.11 ภาพแสดงความแตกต่างของระดับความเข้มของเส้นกับพื้นหลังจากด้านซ้ายไปขวา	14
2.12 ขั้นตอนการทำงานของตัวดำเนินการแค่นี้.....	16
2.13 ขั้นตอนการทำงานของพาทิเคิลสวอมมอปติไมเซชัน	19
2.14 ตัวอย่างผลการทดลองใช้ PSO และ CNN ในการวิเคราะห์ขอบภาพ	22
3.15 ตัวอย่างภาพที่ใช้ในงานวิจัย.....	23
3.16 ผังขั้นตอนการทำงาน.....	24
3.17 ภาพก่อนและหลังผ่านการปรับปรุงคุณภาพ.....	25
3.18 แสดงการวัดความกว้างความยาวผ่านโปรแกรม MATLAB.....	27
3.19 ตัวอย่างการวัดความกว้างความยาว.....	28
4.20 ภาพตัวอย่างต้นฉบับและภาพจาก PSO-Edge	30
4.21 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนด้านกว้างของPSO-Edge แค่นี้ พรีวิต และโซเบล.....	34
4.22 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนด้านยาวของPSO-Edge แค่นี้ พรีวิต และโซเบล.....	34
4.23 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนรวมของPSO-Edge แค่นี้ พรีวิต และโซเบล	35

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

เนื่องจากประเทศไทยอยู่ในกลุ่มประเทศกำลังพัฒนา จึงเต็มไปด้วยการลงทุนการค้าและอุตสาหกรรมต่างๆมากมาย ด้วยตำแหน่งที่ตั้งอยู่ศูนย์กลางของเอเชียตะวันออกเฉียงใต้ (ASIAN) จึงก่อให้เกิดการขยายตัวในภาคอุตสาหกรรม โรงงานอุตสาหกรรมมากมายได้ก่อตั้งฐานการผลิตในประเทศไทยเพื่อเป็นศูนย์กลางผลิตและกระจายสินค้าขยายตัวในภาคอุตสาหกรรมส่งผลให้มีการพัฒนาเทคโนโลยีการผลิตเพื่อเพิ่มกำลังการผลิตลดระยะเวลาในการผลิต ลดต้นทุน แต่ได้ผลิตภัณฑ์เท่าเดิมหรือมากขึ้น การทำงานส่วนใหญ่ภาคอุตสาหกรรมจึงใช้เครื่องจักรกลเป็นหลักแทนที่การทำงานโดยใช้แรงงานมนุษย์เนื่องจากมีกำลังในการผลิตสูงกว่าและสามารถทำงานที่มีความเสี่ยงได้ ถึงแม้ว่าการผลิตโดยใช้เครื่องจักรกลจะมีกำลังการผลิตสูง ความผิดพลาดน้อย แต่ความผิดพลาดก็มีโอกาสเกิดขึ้นได้ ทั้งจากวัตถุดิบ เครื่องจักร หรือแม้แต่วางขั้นตอนการผลิต ดังนั้นกำลังการผลิตที่มากขึ้นก็ไม่ได้หมายความว่าผลิตภัณฑ์ที่ผลิตได้นั้นจะมีมาตรฐานทุกชิ้นในทุกภาคอุตสาหกรรมจึงมีกระบวนการตรวจสอบคุณภาพผลิตภัณฑ์ในทุกขั้นตอนการผลิต จึงมีการพัฒนาเทคโนโลยีการตรวจสอบและควบคุมคุณภาพการผลิตควบคู่กับเทคโนโลยีการผลิต

การตรวจสอบคุณภาพของผลิตภัณฑ์ที่มีขนาดเล็กต้องการความละเอียดแม่นยำและความชำนาญเป็นพิเศษซึ่งกระบวนการตรวจสอบคุณภาพนี้หากใช้มนุษย์ในการดำเนินงานจะต้องใช้เวลามากกว่าที่ควรจึงมีการศึกษาค้นคว้าวิจัยวิธีการตรวจสอบที่สามารถนำมาพัฒนาประสิทธิภาพในการตรวจสอบชิ้นงานให้ได้มาตรฐาน ปรับปรุงกระบวนการในการทำงานให้สามารถลดเวลาในขั้นตอนการตรวจสอบและสามารถลดต้นทุนเมื่อเทียบกับผลิตภัณฑ์ที่ได้การตรวจสอบข้อผิดพลาดในทุกขั้นตอนของกระบวนการผลิต สามารถช่วยในการนำข้อผิดพลาดของแต่ละกระบวนการไปปรับปรุงพัฒนากระบวนการผลิตในแต่ละฝ่าย เพื่อให้ได้ผลิตภัณฑ์ที่มีคุณภาพได้มาตรฐานสูงสุด ลดการสูญเสียวัตถุดิบ ทั้งยังทำให้การผลิตในแต่ละส่วนสามารถทำงานได้อย่างสอดคล้องกันมากยิ่งขึ้นและแก้ปัญหาคอขวดในกระบวนการผลิต กระบวนการตรวจสอบข้อผิดพลาดที่ถูกต้องแม่นยำและตรงจุดนั้นจะช่วยให้สามารถวิเคราะห์หาสาเหตุของปัญหาเพื่อดำเนินการแก้ไขและพัฒนากระบวนการผลิตให้ดีขึ้น ดังนั้นการที่จะทำได้ผลผลิตที่มีคุณภาพเพิ่มมากขึ้นจึงจำเป็นต้องมีการพัฒนากรรมวิธีในการตรวจสอบข้อผิดพลาดของผลิตภัณฑ์ที่รวดเร็วและถูกต้องแม่นยำนั่นเอง

แต่ด้วยกระบวนการที่แตกต่างกันในแต่ละสายงานการผลิต จึงจำเป็นต้องมีวิธีการเฉพาะในการตรวจสอบชิ้นงานนั้นๆ การนำเอาเทคโนโลยีมาปรับใช้เพื่อเพิ่มประสิทธิภาพการผลิตที่สูงที่สุด ลดส่วนเสียหายให้น้อยที่สุด การตรวจจับหาสิ่งผิดปกติ หรือข้อบกพร่องในชิ้นงานจึงเป็นเรื่องที่มีความสำคัญมากเป็นอันดับต้นๆในอุตสาหกรรม เพราะนั่นหมายถึงชื่อเสียงคุณภาพของผลิตภัณฑ์

1.5 ขอบเขตของการวิจัย

การตรวจสอบฟลิปชิป (Flip Chip) โดยใช้ภาพจากกล้องเอกซเรย์ซึ่งนำตัวดำเนินการแค่นี้
ฟรีวิต และโซเบลนำมาวิจัยเพิ่มเติมด้วยวิธีการพาทิเคิลสวอมออปติไมเซชันเพื่อใช้ในการวัดขนาด
กว้างยาวของรอยบัดกรี

1.6 ขั้นตอนการศึกษา

นำภาพตัวอย่างจำนวนหนึ่งมาคำนวณหาค่าแม่แบบสำหรับตรวจหาขอบด้วยพาทิเคิลสวอม
ออปติไมเซชัน ขั้นตอนต่อมา นำแม่แบบที่ได้มาวิเคราะห์ภาพรอยบัดกรี วัดขนาดขอบเขตของวัตถุใน
ภาพในหน่วยพิกเซลแล้วแปลงกลับเป็นนิ้ว แล้วนำมาประเมินความถูกต้องจากค่าที่วัดจริง

และความไว้วางใจของผู้บริโภค ซึ่งแต่ละอุตสาหกรรมก็จะมีกรรมวิธีในการตรวจสอบที่แตกต่างกันตามผลิตภัณฑ์ของอุตสาหกรรมนั้นๆ ซึ่งกระบวนการตรวจสอบคุณภาพหรือตรวจวัดที่นำมาใช้แทนสายตามนุษย์คือเทคโนโลยีการประมวลผลภาพที่ใช้การวิเคราะห์ข้อมูลจากรูปภาพดิจิทัล และสาเหตุที่เทคโนโลยีนี้เป็นที่นิยมในระบบอุตสาหกรรมเนื่องจากสามารถตรวจวัดรายละเอียดได้รวดเร็วและถูกต้องแม่นยำ คุ่มค่ากับการลงทุนมากกว่าการใช้แรงงานมนุษย์ ทั้งนี้เทคโนโลยีการประมวลผลภาพที่แต่เดิมใช้สายตาของมนุษย์ในการตรวจสอบ แน่แน่นอนว่าชิ้นส่วนที่ผิดพลาดนั้นไม่สามารถถูกคัดแยกออกมาได้อย่างที่ควร เนื่องจากความผิดพลาดที่เกิดขึ้นมีขนาดเล็กเกินสายตามนุษย์มองเห็นได้หรือความผิดพลาดเกิดขึ้นภายในโรงงาน เป็นต้น จึงจำเป็นต้องใช้หลักการของการวิเคราะห์รูปภาพและปัญญาประดิษฐ์มาช่วยในการวิเคราะห์ข้อมูลรูปภาพแทนการใช้สายตา โดยคัดแยกได้ถูกต้องกว่าและใช้เวลาใช้น้อยกว่านั่นเอง

1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

- 1.2.1 เพื่อศึกษาหาวิธีการวิเคราะห์ขอบของรูปภาพที่ดีขึ้น
- 1.2.2 เพื่อหาวิธีตรวจสอบรอยเชื่อมบัดกรีที่ได้ประสิทธิภาพสูงสุด
- 1.2.3 เพื่อเพิ่มอัตราการตรวจสอบข้อผิดพลาดให้ได้ประสิทธิภาพสูงขึ้น
- 1.2.4 เพื่อให้เกิดโอกาสในความร่วมมือทำวิจัยระหว่างมหาวิทยาลัยและภาคอุตสาหกรรม
- 1.2.5 เพื่อสร้างงานวิจัยที่สามารถแก้ไขปัญหาในอุตสาหกรรมได้จริง

1.3 สมมุติฐานของการศึกษา

การประยุกต์วิธีการหาขอบเขตภาพร่วมกับพาทิเคิลสวอมมออปติไมเซชันจะสามารถช่วยให้การตรวจรอยเชื่อมบัดกรีทำได้ดีขึ้นหรือไม่ ใช้เวลาน้อยลงหรือไม่ สัดส่วนการตรวจจับได้เป็นเท่าไร

1.4 ทฤษฎีหรือแนวความคิดที่ใช้ในการวิจัย

จากการศึกษางานวิจัยก่อนหน้าจะพบปัญหาความคมชัดของภาพเอกซเรย์ ทำให้ยากต่อการกำหนดขอบเขตของวัตถุในภาพ เมื่อนำวิธีการปัจจุบันมาใช้วิเคราะห์ขนาดของรอยเชื่อมบัดกรีที่วัดได้จากงานทดลองมาเปรียบเทียบกับค่าที่วัดจริงยังคงแตกต่างกันมากจนกระทั่งมีการพัฒนาแม่แบบโดยนำตัวดำเนินการโซเบลมาทำการวิจัยเพิ่มเติมร่วมกับอัลกอริทึมพันธุกรรม (Genetic Algorithm) ซึ่งผลลัพธ์ที่ได้เป็นที่น่าพอใจ และยังมีหลายงานวิจัยที่นำพาทิเคิลสวอมมออปติไมเซชัน (Particle Swarm Optimization) มาสร้างแม่แบบใหม่ทั้งนี้ผลที่ได้สำหรับรูปภาพที่มีลักษณะใกล้เคียงกันที่สามารถวิเคราะห์ได้ถูกต้อง ดังนั้นในงานวิจัยนี้จึงนำพาทิเคิลสวอมมออปติไมเซชันมาประยุกต์ใช้ร่วมกับการประมวลผลภาพสำหรับการตรวจวัดขนาดรอยเชื่อมบัดกรีจากภาพเอกซเรย์

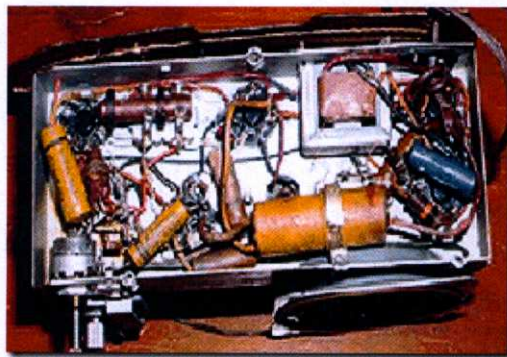
บทที่ 2

ทฤษฎีพื้นฐานและงานวิจัยที่เกี่ยวข้อง

เนื่องจากการทดลองนี้เป็นการศึกษารอยการบัดกรีบนแผงวงจรขนาดเล็กในรูปแบบภาพเอกซเรย์มาวิเคราะห์ จึงจำเป็นต้องมีความรู้ความเข้าใจในชิ้นส่วนและวิธีการต่างๆเพื่อนำมาประยุกต์ใช้ดังนี้

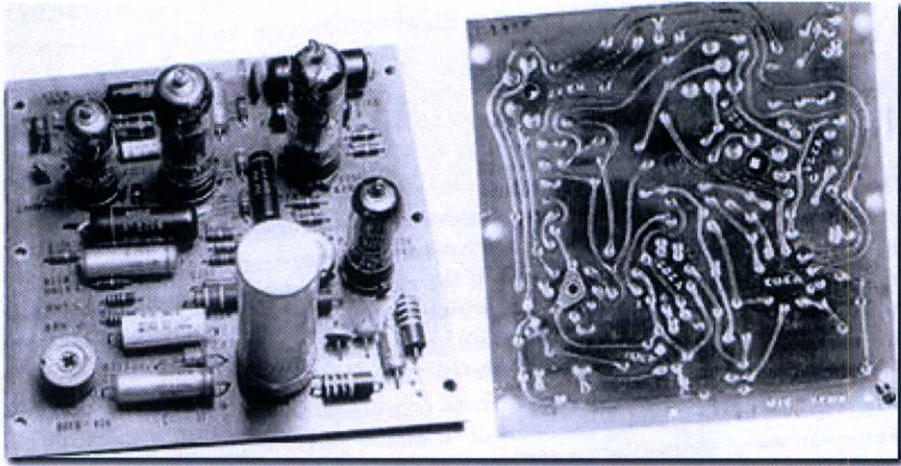
2.1 แผงวงจรพิมพ์

คือส่วนประกอบพื้นฐานที่สำคัญของวงจรอิเล็กทรอนิกส์ เป็นทางเดินสัญญาณไฟฟ้าของอุปกรณ์อิเล็กทรอนิกส์ต่าง ๆ ที่อยู่บนแผงวงจร ทำให้อุปกรณ์ต่าง ๆ เชื่อมต่อกันได้ และสามารถทำงานได้อย่างถูกต้องตามที่ได้ออกแบบไว้ และที่เรียกว่าPrint Circuit Board (PCB) เนื่องจากในยุคแรกของการคิดประดิษฐ์แผงวงจรพิมพ์ได้ใช้หมึกที่มีคุณสมบัตินำไฟฟ้า เช่น หมึกที่เป็นผงคาร์บอนด์พิมพ์ลงบนแผ่นฉนวน เพื่อการเชื่อมต่อกันทางไฟฟ้าของตัวอุปกรณ์แต่ละตัวที่อยู่บนแผ่นฉนวนที่นำมาใช้ สามารถพบเห็นได้ในวิทยุสมัยก่อน แม้ว่าปัจจุบันนี้ใช้แผ่นทองแดงบางๆที่เป็นตัวนำมาเคลือบบนฉนวนแทน ดังนั้นจึงเป็นที่มาของแผงวงจร และแผ่นวงจรพิมพ์แบบยืดหยุ่น (Flexible printed circuit: FPC) ที่เกิดขึ้นมาในภายหลัง



รูปที่ 2.1 ตัวอย่างการนำสายไฟมาบัดกรีเชื่อมต่อกับขั้วของอุปกรณ์แต่ละตัวเข้าด้วยกัน

ประวัติความเป็นมาของแผ่นวงจรพิมพ์นั้นเริ่มต้นตั้งแต่ ค.ศ. 1936 ยุคสมัยก่อนสงครามโลกครั้งที่สอง ได้เริ่มมีการใช้งานเครื่องวิทยุคอนข้างแพร่หลาย ขณะนั้นในส่วนของภาควงจรวิทยุยังใช้วิธีการนำสายไฟมาบัดกรีเชื่อมต่อกับขั้วของอุปกรณ์แต่ละตัวเข้าด้วยกัน ในช่วงเวลาที่Eisler P. นักประดิษฐ์ชาวออสเตรเลียซึ่งทำงานเกี่ยวกับการออกแบบวงจรวิทยุในประเทศอังกฤษได้มีแนวคิดที่จะพยายามลดขนาดของวิทยุ และเพิ่มประสิทธิภาพในการผลิต จึงได้ริเริ่มคิดทำแผ่นวงจรพิมพ์ขึ้นมาใช้สำหรับวงจรวิทยุ

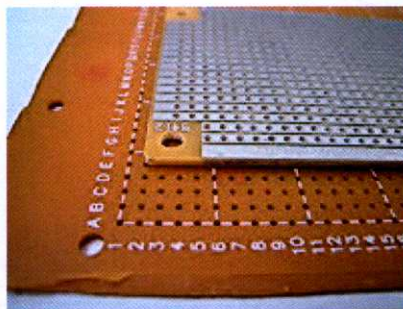


รูปที่ 2.2 แผ่นวงจรพิมพ์ที่ใช้กับวงจรวิทยุ

จากนั้นในช่วงประมาณ ค.ศ. 1950 หลังสงครามโลกครั้งที่สอง แผ่นวงจรพิมพ์ได้ถูกนำมาใช้อย่างแพร่หลาย ซึ่งขณะนั้นยังเป็นแผ่นวงจรพิมพ์แบบด้านเดียว (Single-Sided Board) โดยลายวงจรแต่ละด้านถูกออกแบบด้วยวิธีการแกะสติ๊กเกอร์ด้วยมือและใช้กรดกำจัดแผ่นทองแดงส่วนที่ไม่ต้องการออกไปให้เหลือเพียงส่วนของลายวงจรเท่านั้น ในช่วงระหว่าง ค.ศ. 1960 - 1970 เทคโนโลยีในการเคลือบทองแดงลงบนผนังภายในรูเจาะของแผ่นวงจรพิมพ์ได้ถูกพัฒนาจนกระทั่งสามารถสร้างแผ่นวงจรพิมพ์แบบสองด้าน (Double-Sided Board) ต่อจากนั้นในช่วงประมาณ ค.ศ. 1970 - 1980 การออกแบบวงจรอิเล็กทรอนิกส์ได้เริ่มมีความซับซ้อนมากยิ่งขึ้น จึงได้มีการริเริ่มพัฒนาเทคโนโลยีด้านการออกแบบโดยใช้คอมพิวเตอร์ (CAD) และการสร้างแผ่นวงจรพิมพ์แบบมัลติเลเยอร์ขึ้นมาใช้งาน ซึ่งในปัจจุบันนี้สามารถทำได้มากกว่า 32 เลเยอร์

PCB ประกอบไปด้วยแผ่นฐานหรือซับสเตรท (substrate) ที่ทำจากแผ่นฉนวนบาง ๆ อัดยึตรวมกันด้วยพลาสติกประเภทเทอร์โมเซตติง (Thermosetting plastic) เพื่อรองรับแผ่นตัวนำที่ใช้เชื่อมต่อสัญญาณไฟฟ้าระหว่างอุปกรณ์ ส่วนวัสดุที่ใช้ทำแผ่นฐานที่นิยม โดยทั่วไปจะแยกประเภทตามวัสดุที่นำมาเป็นฐานของทองแดง เช่น ฟีนอลิก (Phenolic, FR-2) กลาสอีพอกซี (Glass Epoxy, FR-4) เทฟลอน (Teflon) เคมวัน (CEM1) อีกแบบคือ เฟล็กซ์อิเบิ้ล (Flexible) และแบ่งตามตามลักษณะการใช้งาน เช่น แบบหน้าเดียว (Single Side) แบบสองหน้าธรรมดา (Double Side) แบบสองหน้าเพลททรูโฮล (Double Side Plate Through Hole) และแบบมัลติเลเยอร์ (Multi-Layer) รายละเอียดของแผ่นวงจรพิมพ์แบบต่างๆดังนี้

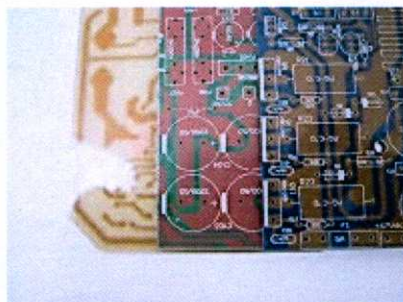
2.1.1 แผงวงจรพิมพ์แบบฟีนอลิก (Phenolic)



รูปที่ 2.3 แผงวงจรพิมพ์แบบฟีนอลิก(Phenolic)

แบบฟีนอลิก เราจะเห็นได้ทั่วไปในอุปกรณ์ต่างๆ เช่นวิทยุกระเป๋าหิ้ว ที่วี การผลิตทำได้รวดเร็วในจำนวนมาก เพราะเนื้อฟีนอลิกนั้นนิ่มและตัดเจาะง่าย ในโรงงานมักจะทำแบบสำหรับผลิตพร้อมทั้งส่วนที่เป็นรูและช่องต่างๆบนแผ่นเนื้อของฟีนอลิกจะเป็นสีน้ำตาลลักษณะวัสดุมีเนื้อที่อ่อนสามารถตัดด้วยมีดคัตเตอร์ได้ โค้งงอเมื่อถูกความร้อน ติดไฟง่าย และมีราคาถูก

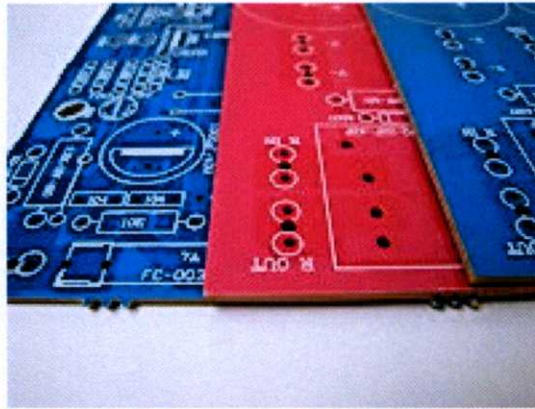
2.1.2 แผงวงจรพิมพ์แบบกลาสอีพอกซี (Glass Epoxy)



รูปที่ 2.4 แผงวงจรพิมพ์แบบกลาสอีพอกซี (Glass Epoxy)

แผงวงจรพิมพ์แบบนี้ใช้ฉนวนซึ่งเป็นฐานของตัวนำคือแผ่นทองแดงบางๆ ทำมาจากเส้นใยแก้วถักทอกันอัดเป็นแผ่นด้วยส่วนผสมของอีพอกซีเรซิน แล้วจึงมาเคลือบแผ่นทองแดงบางๆ ติดกันด้วยกาวอีกครั้งทั้งด้านเดียวและสองด้าน ตามการใช้งาน ดังภาพด้านล่าง สามารถพบเห็นแผงวงจรพิมพ์ชนิดนี้ได้ทั่วไป มีทั้งแบบหน้าเดียว สองหน้า สองหน้าเพลททรูโฮล(Double Side Plate Through Hole) และแบบมัลติเลเยอร์(Multi-Layer) เช่นแผงวงจรในคอมพิวเตอร์

2.1.3 แผงวงจรพิมพ์แบบเคมวัน (CEM1)



รูปที่ 2.5 แผงวงจรพิมพ์แบบเคมวัน (CEM1)

แบบเคมวันมีราคาอยู่ระหว่างแบบฟีนอลิกและอีพอกซี ลักษณะเนื้อสีขาวขุ่นหรือค่อนข้างขาว แข็งแรงมากกว่าแบบฟีนอลิกแต่น้อยกว่าแบบอีพอกซี พบได้ในผลิตภัณฑ์จำพวก UPS ที่ผลิตในประเทศไทย เพราะผลิตได้ง่ายเช่นเดียวกับแบบฟีนอลิกมีราคาถูกกว่าอีพอกซี

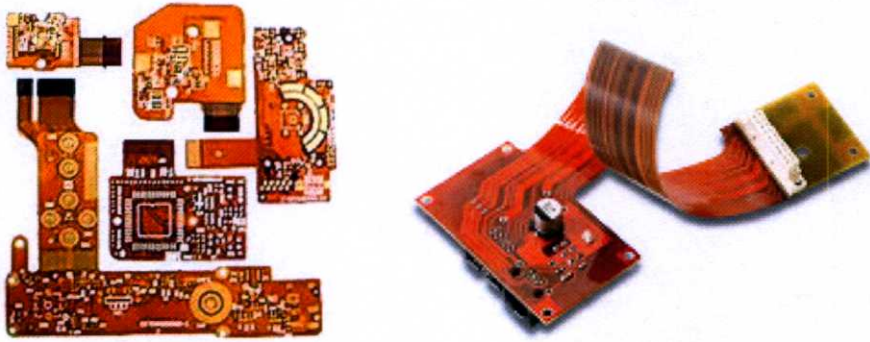
2.1.4 แผงวงจรพิมพ์แบบเทฟลอน (Teflon)



รูปที่ 2.6 แผงวงจรสำหรับกล่องทวนสัญญาณในไมโครเวฟ

แผงวงจรประเภทนี้ มีคุณสมบัติที่ดีทนต่อความร้อน แข็งแรง ไม่มีขायทั่วไป ต้องสั่งเป็นกรณีพิเศษเท่านั้น จะถูกนำไปใช้งานในอากาศยานและในอุปกรณ์ที่อยู่ในช่วงความถี่ระดับ GHz เช่น ความถี่ไมโครเวฟ

2.1.5 แผ่นวงจรพิมพ์แบบยืดหยุ่น (Flexible printed circuit, FPC)

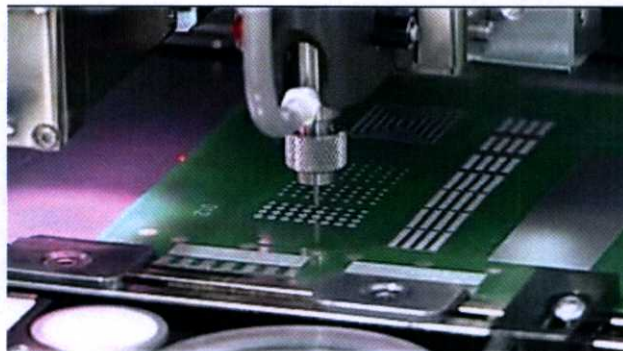


รูปที่ 2.7 แผ่นวงจรพิมพ์แบบยืดหยุ่น (Flexible printed circuit, FPC)

แผ่นวงจรพิมพ์ชนิดนี้เป็นแบบที่มีฐานที่เป็นฉนวนที่บางมาก สามารถดัดไปมาให้อยู่ในช่องแคบได้ สามารถพบเห็นได้ในอุปกรณ์อิเล็กทรอนิกส์ขนาดเล็กเช่นกล้องถ่ายรูป โทรศัพท์เคลื่อนที่

2.2 การบัดกรี

บัดกรีคือการเชื่อมโลหะเข้าด้วยกันโดยใช้โลหะที่มีจุดหลอมเหลวต่ำ (เช่นตะกั่วผสมกับโลหะอื่น) ให้ความร้อนที่ 180 ถึง 190 เซลเซียส (360 ถึง 370 ฟาเรนไฮต์) ด้วยหัวแร้งนิยมนำมาใช้ในการเชื่อมอุปกรณ์อิเล็กทรอนิกส์โดยแรงงานมนุษย์ แต่ในงานอุตสาหกรรมที่ต้องผลิตชิ้นงานจำนวนมากจะใช้โลหะบัดกรีเป็นโลหะหลอมเหลวที่นิยมนำมาใช้ในการประกอบ PCB หรือโลหะบัดกรีแบบกึ่งของเหลวของแข็ง (Solder paste) ซึ่งนิยมนำมาใช้ในการผลิต FPC แบบ SMT (Surface mount technology) ที่แสดงดังรูปที่ 2.8 โดยเมื่อพิมพ์ Solder paste ลงบนตำแหน่งที่ต้องการ แล้วจึงวางไอซีหรือตัวเก็บประจุลงบน Solder paste จากนั้นจึงให้ความร้อนเพื่อให้ Solder paste หลอมตัวและประสานยึดติดไอซีหรือตัวเก็บประจุกับแผ่นวงจรพิมพ์



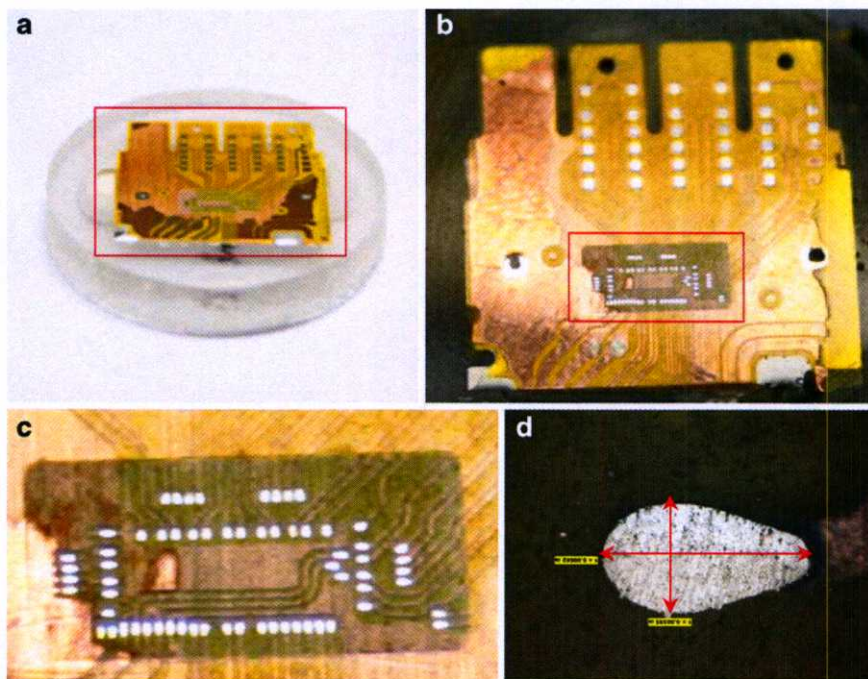
รูปที่ 2.8 การบัดกรีในงานอุตสาหกรรม

ถึงแม้ Solder paste เป็นส่วนประกอบเพียงเล็กน้อยแต่มีความสำคัญอย่างมากต่ออายุการใช้งานของอุปกรณ์อิเล็กทรอนิกส์นั้นๆ โดยทั่วไป Solder paste เป็นวัสดุที่ประกอบด้วยโลหะ สารอินทรีย์ และสารอินทรีย์หลายชนิดเพื่อให้ได้สมบัติขณะการปฏิบัติงานและสมบัติการยึดติดที่ต้องการ

Solder paste เป็นวัสดุที่มีลักษณะคล้ายยาสีฟัน โดยมีส่วนประกอบที่เป็นผงโลหะผสมกับฟลักซ์ (Flux) ซึ่งเป็นส่วนที่ทำให้วัสดุผสมนี้มีลักษณะคล้ายครีม ตัวแปรที่มีผลต่อสมบัติของ Solder paste ได้แก่ ส่วนประกอบของโลหะผสม ขนาดและการกระจายตัวของอนุภาคของโลหะผสม สัดส่วนของผงโลหะ ชนิดของฟลักซ์ (Flux)

2.2.1 วิธีการตรวจสอบรอยบัดกรี

การตรวจวัดขนาดของการบัดกรีสามารถตรวจวัดได้ 2 วิธี วิธีแรกคือการตรวจวัดแบบทำลายโดยใช้กล้องจุลทรรศน์ จึงเป็นวิธีที่มีความแม่นยำที่สุดแต่ชิ้นงานไม่สามารถนำกลับมาใช้ใหม่วิธีที่สองใช้ภาพเอกซเรย์สำหรับตรวจวัด ข้อดีคือไม่ต้องทำลายชิ้นงานแต่ก็มีปัญหาเรื่องความถูกต้องในการตรวจวัด เนื่องจากความคลุมเครือของภาพเอกซเรย์ทำให้ยากต่อการหาขอบ



รูปที่ 2.9 ตัวอย่างการตรวจวัดชิ้นงานแบบทำลายโดยกล้องจุลทรรศน์

2.3 เอกซเรย์



รูปที่ 2.10 ภาพเอกซเรย์ภาพแรก

เอกซเรย์คือ รังสีหรือแสงชนิดหนึ่งที่เราไม่สามารถมองเห็นได้ด้วยตาเปล่า เช่นเดียวกับแสงสว่างธรรมดา เอกซเรย์นี้มีลักษณะเป็นทั้งคลื่นและอนุภาค ของแม่เหล็กไฟฟ้า ที่มีช่วงคลื่นสั้นมาก ความยาวช่วงคลื่นตั้งแต่ 0.04-1000 อังสตรอม (Angstrom) (อังสตรอม คือ หน่วยวัดความยาวของคลื่น 1 อังสตรอม (A) เท่ากับ 10^{-7} เซนติเมตร) หรือ อยู่ระหว่างรังสีแกมมา กับรังสีอัลตราไวโอเล็ต คุณสมบัติของเอกซเรย์ คล้ายคลึงกับแสงสว่างธรรมดา เป็นส่วนใหญ่ แต่คุณสมบัติพิเศษ ของมันคือ มีอำนาจทะลุทะลวงผ่านวัตถุต่าง ๆ ได้มากบ้างน้อยบ้าง ขึ้นอยู่กับ ความแน่นทึบ และน้ำหนักอะตอมของวัตถุที่มันผ่าน นอกจากนั้น ยังทำให้เกิดการเปลี่ยนแปลง ทั้งทางเคมีชีวและอื่น ๆ อีกด้วย

2.3.1 ประวัติการค้นพบเอกซเรย์

ผู้ที่ค้นพบเอกซเรย์เป็นคนแรก คือ นักฟิสิกส์ชาวเยอรมัน ชื่อ Wilhelm Conrad Roentgen การพบนี้เกิดขึ้นในตอนเย็นของวันที่ 8 พฤศจิกายน ค.ศ.1895 ภายในห้องทดลอง ณ มหาวิทยาลัยวูร์ซบวร์ก (Wurzburg) ประเทศเยอรมัน เรินท์เก้นได้พบเอกซเรย์โดยบังเอิญ คือ ขณะที่เขากำลังทำการทดลองเกี่ยวกับเรื่อง "Absorption of cathode rays" โดยใช้หลอดทดลองที่เรียกว่า Crookes' tube เขาสังเกตเห็นว่า Cathodes rays ที่ออกมาจากหลอดทดลอง ทำให้กระดาษแข็งที่ฉาบด้วย แบเรียมปลาตินไซยาไนด์ (Barium platinocyanide) เกิดเรืองแสง (Fluoresce) ระยระยับที่สว่างที่สุดที่ยังมีการเรืองแสงบนกระดาษแข็ง คือ 120 เซนติเมตร ในขณะที่เดียวกันเขาสังเกตเห็นอีกว่า ตัวอักษร A ที่ทำด้วยแบเรียมปลาตินไซยาไนด์ อยู่ห่างออกไปเกือบสิบฟุตเกิดเรืองแสงขึ้นด้วย ทั้ง ๆ ที่ไม่อยู่ในระยะของ Cathode rays เรินท์เก้นจึงคิดว่า เขาได้ค้นพบรังสีชนิดใหม่ขึ้นแล้ว และให้ชื่อว่า "เอกซเรย์" และใช้เวลาอีกหลายสัปดาห์ต่อมาทำการสังเกตถึงการทะลุทะลวง (Penetration) ของเอกซเรย์ผ่านกระดาษผ่านโลหะ และแม้กระทั่งผ่านเนื้อหนังของคน และเขาได้ถ่ายภาพรังสี ของมือของภรรยาเขาไว้ด้วย ในที่สุดเขาจึงประกาศให้โลกรู้ว่า เขาได้ค้นพบเอกซเรย์เป็นคนแรก

2.3.2 ลักษณะและคุณสมบัติของเอกซเรย์

2.3.2.1 เป็นรังสีคลื่นแม่เหล็กไฟฟ้ามีช่วงคลื่นสั้นมาก คืออยู่ในช่วงตั้งแต่ 0.04-1000 A หรือ อยู่ในช่วงระหว่างรังสีแกมมาที่รังสีอัลตราไวโอเล็ต

2.3.2.2 มีคุณสมบัติเหมือนแสงสว่างธรรมดา เช่น เดินทางเป็นเส้นตรง ถ้าเดินทางในสุญญากาศแล้ว เดินทางด้วยความเร็วเท่ากับแสงคือ 186,000 ไมล์ต่อวินาที หรือ 3×10^{10} เซนติเมตร/วินาที นอกจากนี้ยังมีการสะท้อนกลับ หักเห และ เบี่ยงเบนได้ เช่นเดียวกับแสงสว่างธรรมดา

2.3.2.3 ไม่หักเหโดยสนามแม่เหล็กหรือสนามไฟฟ้า

2.3.2.4 เกิดจากการที่อนุภาคอิเล็กตรอนที่มีความเร็วสูงวิ่งไปชนเป้า(Target) อิเล็กตรอนดังกล่าวนี้ จะไปชนอิเล็กตรอนตัวอื่นๆที่อยู่ในวงโคจรของอะตอมของเป้า ให้หลุดกระเด็นออกนอกวงโคจร

2.3.2.5 ทำให้เกิดการเรืองแสง (Fluorescence และ Phosphorescence) ในสารพิเศษบางอย่าง

2.3.2.6 ดูดกลืน (Absorbed) โดยสสาร (Matter) ทุกชนิดมากขึ้นขึ้นอยู่กับความหนาแน่น และน้ำหนักของอะตอมของสสารนั้น

2.3.2.7 ทำให้เกิดการปล่อยประจุไฟฟ้า (Ionization) เมื่อผ่านไปใ้อากาศหรือก๊าซ

2.3.2.8 ทำให้เกิดการเปลี่ยนแปลงทางเคมี เช่น เมื่อเอกซเรย์ไปถูกฟิล์มถ่ายรูปจะทำให้ฟิล์มนั้นดำ จึงนำผลอันนี้มาใช้ในการบันทึกภาพรังสีลงบนแผ่น ฟิล์มเอกซเรย์

2.3.2.9 ทำให้เกิดการเปลี่ยนแปลงทางชีววิทยาทำให้เซลล์ของร่างกายเปลี่ยนแปลง หรือเกิดการผ่าเหล่า (Genetic mutation) ถ้าได้รับรังสีเป็นจำนวนมากและนานพอ

2.3.2.10 มีอำนาจในการทะลุทะลวง (Penetration) สูง สามารถทะลุผ่านเนื้อหนังของมนุษย์และสัตว์ได้แต่ไม่สามารถทะลุผ่านแผ่นตะกั่วหรือคอนกรีตหนาๆได้

2.3.3 การเกิดเอกซเรย์

เอกซเรย์เกิดขึ้นโดยที่อนุภาคของอิเล็กตรอนที่มีความเร็วสูงไปชนเป้า(Target) ผลทำให้เกิดเป็นเอกซเรย์และความร้อน การเกิดของเอกซเรย์นี้อาศัยองค์ประกอบสำคัญ 5 ประการ คือ

2.3.3.1 ทำให้เกิดอนุภาคอิเล็กตรอนหรือการแยกอิเล็กตรอนนอกจากอะตอมของโลหะอิเล็กตรอนจะเกิดขึ้น หรือแยกจากอะตอมของโลหะที่ทำให้เป็นไส้หลอดเอกซเรย์ โดยการที่ผ่านกระแสไฟฟ้าเข้าไปในไส้หลอด (Filament) จนกระทั่งไส้หลอดร้อนขึ้น ประมาณ 2,000 C หรือมากกว่านั้น จะมีผลทำให้เกิดเทอร์มิออนิกเอมิชชั่น (Thermionic Emission) คืออิเล็กตรอนหลุดออกจากอะตอมหรือวงโคจรของมัน เมื่อถูกความร้อนอิเล็กตรอนที่หลุดออกมานี้จะมารวมกันอยู่รอบๆ ผิวหน้าของโลหะ จนกลายเป็นกลุ่ม (Cloud) เรียกว่า Space charge สาเหตุที่อิเล็กตรอนไม่สามารถหลุดพ้นไปจากผิวหน้าโลหะเพราะว่าแรงดึงดูดระหว่างอิเล็กตรอนและอะตอมของโลหะยังมีอยู่

2.3.3.2 ทำให้อนุภาคอิเล็กตรอนหลุดและเคลื่อนที่ไปด้วยความเร็วสูง วิธีที่จะทำให้อนุภาคอิเล็กตรอนหลุดจากอะตอมไปได้ คือต้องหาสิ่งที่มีพลังงานหรือแรงดึงดูด มากกว่าแรงดึงดูดระหว่างอิเล็กตรอนกับอะตอมของโลหะมาดึงดูดอิเล็กตรอนสิ่งนั้นก็คือไฟฟ้าแรงสูง (High voltage) ไฟฟ้าแรงสูงนั้นนอกจากจะทำให้อิเล็กตรอนหลุดจากผิวหน้าของโลหะแล้ว ยังทำให้อิเล็กตรอนเคลื่อนที่หรือวิ่งไปด้วยความเร็วสูง กล่าวอีกอย่างหนึ่งคือต้องทำให้ไฟฟ้าระหว่างขั้วบวกและขั้วลบให้ต่างกัน ความต่างศักย์นี้ยิ่งมีค่ามากขึ้นเท่าใดก็ยิ่งทำให้อิเล็กตรอนหลุดจากผิวหน้าของไส้หลอดได้มากและวิ่งไปด้วยความเร็วสูงขึ้น ความต่างศักย์ 100 Kilo-voltage จะทำให้อิเล็กตรอนมีความเร็วเท่ากับประมาณ 165 กิโลเมตรต่อวินาที

2.3.3.3 การทำให้เส้นทางที่อิเล็กตรอนวิ่งผ่านไม่ต้องไม่มีสิ่งกีดขวางจึงทำให้อิเล็กตรอนวิ่งไปด้วยความเร็วสูงดังนั้นภายในหลอดเอกซเรย์จึงเป็นสุญญากาศ เพราะถ้ามีอากาศหรือก๊าซอยู่ภายในหลอดแม้เพียงเล็กน้อยก็ตามความเร็วของอิเล็กตรอนจะลดลง เนื่องจากเกิดไอออนไนเซชัน (Ionization)

2.3.3.4 การทำให้อิเล็กตรอนมีความเข้มข้น (Concentration of electron) ต้องมีวิธีการที่จะให้อิเล็กตรอนเคลื่อนที่ไปในแนวทิศทางเดียวกัน คือพุ่งไปหาจุดโฟกัสของเป้าด้วยปริมาณหรือความเข้มข้นที่มากพอสมควร ซึ่งทำได้โดยการใช้เครื่องมือที่เรียกว่า "Electron focusing device" คอยควบคุมให้อิเล็กตรอนส่วนใหญ่ไปตกในบริเวณจุดโฟกัสของเป้า

2.3.3.5 การทำให้อิเล็กตรอนหยุดวิ่งในทันทีทันใดการที่จะทำให้อิเล็กตรอนที่วิ่งมาด้วยความเร็วสูงหยุดวิ่งในทันที ทำได้โดยหาวัตถุมาถนัดหรือขวางอิเล็กตรอนวัตถุนั้นก็คือเป้า (Target) ผลที่เกิดขึ้นคือเกิดการชนหรือกระทบกันระหว่างอิเล็กตรอนกับเป้าอย่างแรง อิเล็กตรอนดังกล่าวจะไปชนอิเล็กตรอนที่อยู่ในวงโคจรของอะตอมในเป้านั้นให้หลุดกระเด็นออกนอกวงโคจร และจะมีอิเล็กตรอนจากวงโคจรอื่นๆที่อยู่ถัดไปวิ่งเข้ามาแทน แต่เนื่องจากพลังงานของอิเล็กตรอนในแต่ละวงโคจรจะไม่เท่ากัน ดังนั้นจึงมีการคายพลังงานส่วนเกินออกมาในรูปของเอกซเรย์และความร้อน (ส่วนใหญ่จะเป็นความร้อนมากกว่า) กล่าวคือจากพลังงานของอิเล็กตรอนทั้งหมด ที่วิ่งไปสู่เป้านั้น 99.8% จะเปลี่ยนเป็นความร้อน และ 0.2 % เป็นเอกซเรย์

2.4 การประมวลผลรูปภาพ (Image Processing)

เป็นการประยุกต์ใช้งานการประมวลผลสัญญาณบนสัญญาณ 2 มิติ เช่น ภาพนิ่ง (ภาพถ่าย) หรือภาพวีดิทัศน์ (วิดีโอ) และยังรวมถึงสัญญาณ 2 มิติอื่นๆ ที่ไม่ใช่ภาพด้วยแนวความคิดและเทคนิคในการประมวลผลสัญญาณสำหรับสัญญาณ 1 มิติสามารถปรับมาใช้กับภาพได้ไม่ยาก การประมวลผลภาพมีเทคนิคและแนวความคิดที่เฉพาะ เช่น connectivity และ rotation invariance ซึ่งจะมีความหมายกับสัญญาณ 2 มิติเท่านั้น แต่เทคนิคบางอย่างใช้การประมวลผลสัญญาณใน 1 มิติ จะค่อนข้างซับซ้อนเมื่อนำมาใช้กับ 2 มิติ เมื่อหลายสิบปีก่อนการประมวลผลภาพนั้น จะอยู่ในรูปของการประมวลผลสัญญาณแอนะล็อก (analog) โดยใช้อุปกรณ์ปรับแต่งแสง (optics) ซึ่งวิธีเหล่านั้นก็ไม่ได้หายสาบสูญหรือเลิกใช้ไป ยังมีใช้เป็นส่วนสำคัญสำหรับการประยุกต์ใช้งานบางอย่าง เช่น ฮอโลกราฟี (holography) แต่เนื่องจากอุปกรณ์คอมพิวเตอร์ในปัจจุบันราคาถูกลงและเร็วขึ้นมาก การประมวลผลภาพดิจิทัล (digital image processing) จึงได้รับความนิยมมากกว่าเพราะการประมวลผลที่ได้ซับซ้อนแม่นยำและง่ายในการลงมือปฏิบัติแบ่งเป็น 5 แขนงดังนี้

การบีบอัดข้อมูลภาพ (Image Compression) จะแปลงรูปภาพให้มีขนาดเล็กลงโดยทั่วไปแล้วจะแบ่งออกเป็น 2 ประเภทดังต่อไปนี้

- การบีบอัดแบบไม่มีการสูญเสียรายละเอียดข้อมูล (Lossless compression) ซึ่งค่าความสว่างของแต่ละจุดภาพจะยังคงอยู่เหมือนเดิมหรือไม่มีการเปลี่ยนแปลงค่าของแต่ละจุดภาพ ซึ่งการบีบอัดวิธีนี้จะอาศัยเทคนิคการจัดเก็บข้อมูลเชิงเลขในการลดขนาดของข้อมูล ตัวอย่างเช่น Lempel–Ziv–Welch (LZW), Run-Length Encoding (RLE) หรือ PackBits เป็นต้น
- การบีบอัดแบบสูญเสียรายละเอียดข้อมูล (Lossy compression) วิธีการนี้จะมีการเปลี่ยนแปลงค่าความสว่างของจุดภาพวิธีการนี้ไม่เหมาะสมสำหรับข้อมูลภาพที่ต้องมีการจำแนกข้อมูล (Classification) ตัวอย่างของการบีบอัดประเภทนี้ได้แก่ JPEG, MrSID หรือ ECW เป็นต้น

การปรับปรุงคุณภาพของภาพ (Image Enhancement) คือ กระบวนการปรับปรุงภาพให้ดีขึ้นเพื่อประโยชน์การแปลภาพด้วยการมองด้วยตา (Visual Interpretation) โดยที่ไม่มีการเปลี่ยนแปลงเนื้อหาภาพ สามารถแบ่งได้เป็น 2 Domain ได้แก่ Spatial Domain และ Frequency Domain ส่วน Noise คือการแปรปรวนโดยไม่ได้คาดหวังของความเข้มของแสง (Gray Scale) ของภาพ Histogram คือกราฟแสดงการกระจายของความเข้มของแสง (Gray Scale) ของภาพในทางสถิติ การกระจายข้อมูลของ Histogram ที่สมบูรณ์ควรเป็นรูประฆังคว่ำที่มีสมมาตรของการกระจายแบบปกติ (Normal Distribution)

การคืนภาพ(Image Restoration)เป็นกระบวนการปรับปรุงภาพออกมาให้เหมือนภาพตั้งต้น ข้อแตกต่างระหว่าง Image restoration กับ Image enhancement คือ Image restoration เป็นการทำให้ภาพเหมือนภาพเดิมมากที่สุดแต่ Image enhancement คือการปรับปรุงคุณภาพของภาพ

การวิเคราะห์รูปภาพ(Image Analysis)คือการนำภาพมาปรับแต่งให้ได้ข้อมูลที่ต้องการ การสังเคราะห์ภาพ(Image Synthesis)การรวมเอาสิ่งต่าง ๆ มาสร้างเป็นภาพ หรือโมเดล

2.4.1 วิธีกรองข้อมูลโดยใช้มัธยฐาน (Median Filter)

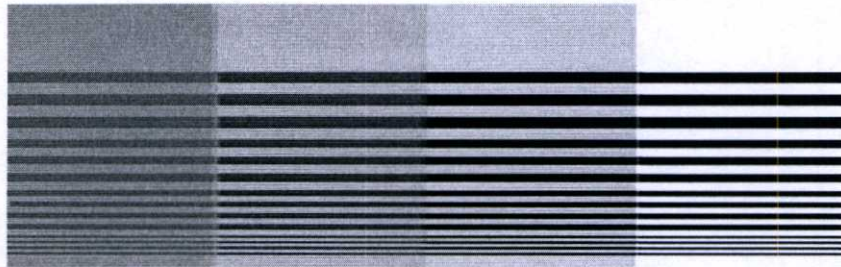
วิธีการกรองข้อมูลโดยใช้มัธยฐานเป็นการแทนค่าระดับสีเทาในจุดภาพที่ต้องการ โดยนำจุดภาพข้างเคียงของจุดภาพที่ต้องการมาเรียงลำดับข้อมูลแล้วจึงเลือกค่ามัธยฐานจากข้อมูลที่ทำกรเรียงลำดับแล้ว มาใช้แทนค่าในจุดภาพที่ต้องการนั้นเมื่อใช้กับข้อมูลรูปภาพจะใช้คำนวณทุกจุดภาพโดยเลื่อนไปที่ละจุดภาพ

$$f(x,y) = \text{median}_{(s,t) \in S_{xy}} \{g(s,t)\} \quad (2.1)$$

วิธีกรองข้อมูลโดยใช้มัธยฐานนี้เป็นวิธีที่รู้จักและถูกใช้อย่างแพร่หลายเพราะสามารถลดสัญญาณรบกวนได้ดีและทำให้ภาพเกิดความพริ้วม้วนน้อยกว่าการกรองข้อมูลแบบเชิงเส้น (Linear Smoothing Filter) ในขนาดภาพที่เท่ากัน

2.4.2 วิธีกรองข้อมูลโดยใช้การเพิ่มความคมชัด (Unsharp Filter)

วิธีการกรองข้อมูลโดยใช้การเพิ่มความคมชัด เป็นการปรับปรุงภาพโดยใช้หลักการปรับแต่ง Contrast ที่มีความเปรียบต่างสูงจะจำแนกวัตถุได้ชัดกว่าความเปรียบต่างต่ำดังภาพตัวอย่าง



รูปที่ 2.11 ภาพแสดงความแตกต่างของระดับความเข้มของเส้นกับพื้นหลังจากด้านซ้ายไปขวา

$$g(x,y) = f(x,y) - f_{\text{smooth}}(x,y) \quad (2.2)$$

$$f_{\text{unsharp}}(x,y) = f(x,y) + k * g(x,y) \quad (2.3)$$

การทำงานคือ เพิ่มความคมชัด โดยลดความสว่างตรงขอบในส่วนที่เข้มไปจากเดิมเล็กน้อย และเพิ่มความสว่างตรงขอบในส่วนที่สว่างไปอีกเล็กน้อย

2.4.3 การวิเคราะห์ขอบภาพ (Edge Detection)

เป็นส่วนหนึ่งของการประมวลผลรูปภาพการหาขอบภาพคือการตรวจสอบว่าเส้นขอบลากผ่านหรือใกล้เคียงกับจุดใด โดยวัดจากการเปลี่ยนแปลงของความเข้มในตำแหน่งที่ใกล้เคียงกับจุดดังกล่าว ซึ่งวิธีการหาขอบนั้นมีด้วยกันหลายวิธี แต่อย่างไรก็ตามสามารถแบ่งได้เป็น 2 กลุ่มหลัก คือ Gradient method และ Laplacian method โดยในแต่ละวิธีมีรายละเอียดดังต่อไปนี้

2.4.3.1 Gradientmethod วิธีนี้จะหาขอบโดยการหาจุดต่ำสุดและจุดสูงสุดในรูปของอนุพันธ์อันดับหนึ่งของภาพ โดยจุดที่เป็นขอบจะอยู่ในส่วนที่เหนือค่าเทรชโฮลด์(threshold)จึงอาจทำให้เส้นขอบที่ได้มีลักษณะหนา ตัวอย่างวิธีการหาขอบของกลุ่มนี้ เช่น Roberts, Prewitt, Sobel และ Canny เป็นต้น

1. ตัวดำเนินการโซเบล (Sobel Operator)

ถูกนำมาใช้อย่างหลากหลายโดยเฉพาะอย่างยิ่งอัลกอริทึมในการตรวจหาขอบ การคำนวณเกรเดียนต์ (Gradient) จากความเข้มของภาพ โดยแต่ละจุดของรูปภาพนั้นมีผลที่ได้จากตัวดำเนินการโซเบล(x,y)เทคนิคของแต่ละตัวดำเนินการก็แตกต่างกันออกไป ค่าวนจากจุดข้างเคียงจุดศูนย์กลางที่อยู่ใน3x3ใน 2 ทิศทาง เพื่อลดสัญญาณรบกวนจึงให้น้ำหนักโดยบริเวณจุดศูนย์กลางจะมีค่าสูงและส่วนที่ห่างออกไปจะมีค่าน้อย สามารถอธิบายเป็นสมการเกรเดียนต์ได้ดังนี้

$$G_x = \{f(x+1,y-1)+2f(x+1,y)+f(x+1,y+1)\} - \{f(x-1,y-1)+2f(x-1,y)+f(x-1,y+1)\} \quad (2.4)$$

$$G_y = \{f(x-1,y+1)+2f(x,y+1)+f(x+1,y+1)\} - \{f(x-1,y-1)+2f(x,y-1)+f(x+1,y-1)\} \quad (2.5)$$

ขนาดของเกรเดียนต์โดยทั่วไป

$$g(x,y) = \sqrt{G_x^2 + G_y^2} \quad (2.6)$$

สามารถนำมาใช้ในแบบเดียวกัน

$$g(x,y) = |G_x| + |G_y| \quad (2.7)$$

รูปแบบแม่แบบตัวดำเนินการคอนโวลูชัน

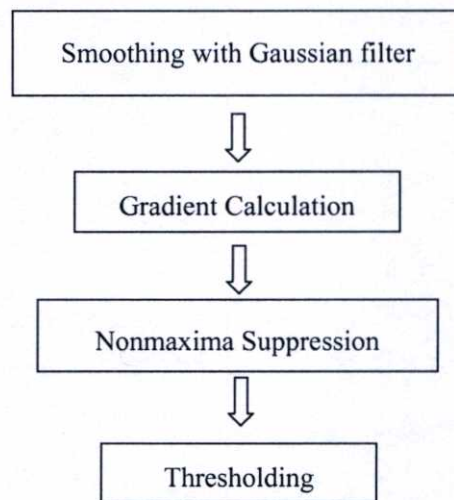
$$T_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (2.8)$$

$$T_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2.9)$$

ตัวดำเนินการโซเบลในการตรวจหาขอบของรูปภาพ M แม่แบบแนวนอน T_x และแม่แบบแนวตั้ง T_y มาคอนโวลูชัน (Convolution) กับรูปภาพยกเว้นส่วนกรอบของรูป เมื่อคอนโวลูชันเสร็จจะได้เมตริกซ์เกรเดียนต์ 2 เมตริกซ์ที่มีขนาดเท่ากับภาพตั้งต้นคือ M_1 และ M_2 นำค่าเมตริกซ์ทั้ง 2 มาบวกกันเพื่อหาค่า G แล้วได้ภาพขอบออกมาด้วยค่าเทรชโฮลด์ (Threshold)

2. ตัวดำเนินการแคนนี่ (Canny Operator)

ขั้นตอนการหาขอบโดยวิธีของแคนนี่ ประกอบด้วย 4 ขั้นตอนดังนี้



รูปที่ 2.12 ขั้นตอนการทำงานของตัวดำเนินการแคนนี่

การทำงานของ Canny edge detection นั้นเริ่มต้นจากการปรับภาพให้เรียบ (Smoothing) ด้วยตัวกรองเกาส์เซียน (Gaussian filter) เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นคำนวณค่าขนาด (magnitude) และทิศทาง (orientation) ของ gradient โดยใช้การหาอนุพันธ์อันดับหนึ่งต่อมาใช้ nonmaxima suppression กับ gradient magnitude เพื่อทำให้ได้ขอบที่บางลง และในขั้นตอนสุดท้ายใช้ double thresholding algorithm เพื่อระบุพิกเซลที่เป็นขอบและช่วยเชื่อมต่อขอบ ในแต่ละขั้นตอนมีรายละเอียดดังต่อไปนี้

Smoothing ในขั้นตอนแรกของการหาขอบโดยอัลกอริทึมนี้จะต้องกำจัดสัญญาณรบกวนออกก่อนโดยใช้ตัวกรองเกาส์เซียนซึ่งสามารถคำนวณได้จากการใช้แม่แบบขนาดเล็ก ขนาดของแม่แบบนี้หากมีขนาดกว้างจะมีผลทำให้ลดสัญญาณรบกวนได้มาก แต่ถ้ากว้างมากเกินไปจะมีผลทำให้ขอบย่อยๆ ที่เป็นส่วนรายละเอียดนั้นหายไป สำหรับการคำนวณหาภาพที่ได้จากการใช้ตัวกรองเกาส์เซียนเป็นดังสมการ (1)

$$S[i, j] = G[i, j, \sigma] * I[i, j] \tag{2.10}$$

กำหนดให้

$I[i, j]$ เป็นภาพที่ต้องการหาขอบ

$G[i, j, \sigma]$ เป็น Gaussian smoothing filter

σ เป็น spread of the Gaussian (ควบคุมระดับของการ smoothing)

Gradient Calculation ในขั้นแรกนำ smoothing image $S[i, j]$ มาสร้าง x, y partial derivatives $P[i, j]$ และ $Q[i, j]$ ตามลำดับดังสมการที่ (2.11) และ (2.12)

$$P[i, j] \approx (S[i, j+1] - S[i, j] + S[i+1, j+1] - S[i+1, j]) / 2 \tag{2.11}$$

$$Q[i, j] \approx (S[i, j] - S[i+1, j] + S[i, j+1] - S[i+1, j+1]) / 2 \tag{2.12}$$

หลังจากนั้นนำค่า x, y partial derivatives มาคำนวณด้วยสูตรมาตรฐานสำหรับการแปลงรูปแบบจาก rectangular ไปเป็น polar (rectangular-to-polar conversion) เพื่อหาขนาดและทิศทางของ gradient ตามสมการที่ (2.13)

$$M[i, j] = \sqrt{P[i, j]^2 + Q[i, j]^2} \tag{2.13}$$

$$\theta[i, j] = \arctan(Q[i, j], P[i, j]) \tag{2.14}$$

จากสมการข้างต้นจะสามารถหาค่ามุม θ ออกมาได้เมื่อแทนค่าตัวแปรในฟังก์ชัน $\arctan(x, y)$

Nonmaxima Suppression สำหรับการหาขอบด้วยแคนน์จุดที่ถือเป็นเส้นขอบได้นั้นต้องเป็นจุดที่ให้ค่าสูงสุดเฉพาะที่และเป็นทิศทางเดียวกับเกรเดียนต์ (gradient) ด้วยซึ่งด้วยวิธี

ดังกล่าวนี้ทำให้ได้ขอบที่บางเพียง 1 พิกเซล ภาพที่ได้หลังการทำ Nonmaxima Suppression จะให้ค่าเป็นศูนย์ในทุกจุดยกเว้นจุดที่เป็น local maxima points ซึ่งจะยังคงค่าเดิมไว้

Thresholding แม้ว่าภาพจะผ่านการ smoothing ในขั้นตอนแรกแล้วก็ตาม ภาพที่ได้อาจยังมีเส้นขอบที่ไม่ใช่ขอบที่แท้จริงปรากฏอยู่เนื่องมาจากสัญญาณรบกวนหรือลักษณะของวัตถุในภาพเป็นพื้นผิวที่มีลวดลายหรือมีรายละเอียดภายในมาก ดังนั้นเพื่อลดปัญหาดังกล่าวจึงได้มีการกำหนดค่าเทรชโฮลด์ขึ้นมา 2 ค่า คือ ค่าเทรชโฮลด์สูง (T1) และ ค่าเทรชโฮลด์ต่ำ (T2) โดยพิกเซลที่มีค่ามากกว่า T1 จะถูกปรับเป็น 1 (เป็นพิกเซลที่เป็นขอบ) แต่ถ้าน้อยกว่า T2 จะถูกปรับเป็น 0 ส่วนค่าที่อยู่ระหว่างค่าเทรชโฮลด์ทั้งสอง การปรับเป็นค่า 0 หรือ 1 นั้นขึ้นอยู่กับพิกเซลที่อยู่รอบข้าง หากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ (ค่า > T1) มีค่ามากกว่า T2 แล้ว จะปรับค่าพิกเซลดังกล่าวให้มีค่าเป็น 1 และถือเป็นสมาชิกหนึ่งในภาพขอบด้วยเช่นกัน

3. ตัวดำเนินการพรีวิต(Prewitt Operation)

การหาขนาดของเกรเดียนต์แบบไม่ต่อเนื่องโดยวิธีของพรีวิตสามารถคำนวณจากสมการ เช่นเดียวกับตัวดำเนินการโซเบลแต่เลือกค่าพารามิเตอร์เท่ากับ 1 สามารถหาแม่แบบสำหรับคอนโวลูชันได้ มีความไวต่อสัญญาณรบกวนต่ำการเกิดโอเวอร์โฟลว์ต่ำกว่าตัวดำเนินการโซเบล

2.4.3.2 Laplacian method จะหาขอบโดยใช้อนุพันธ์อันดับ 2 โดยใช้จุดที่ค่า y เป็น 0 (Zerocrossing) ซึ่งวิธีนี้จะใช้เวลาในการคำนวณมากกว่า Gradient method ตัวอย่างวิธีการหาขอบของกลุ่มนี้ เช่น Laplacian of Gaussian และ Marrs-Hildreth เป็นต้น

2.5 พาทิคอลสวอมมออปติมิซเซชัน(Particle Swarm Optimization: PSO)

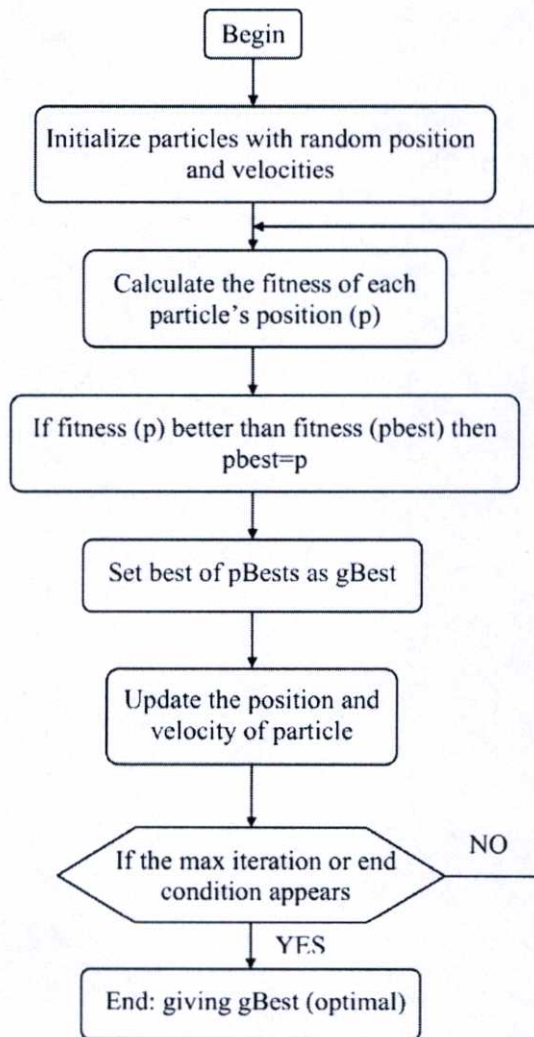
พาทิคอลสวอมมออปติมิซเซชันเป็นเทคนิคการหาค่าเหมาะสมที่สุดแบบกลุ่มอนุภาค ซึ่งได้รับการพัฒนาโดย J.Kennedy และ R.Eberthart โดยมีแนวคิดมาจากการจำลองวิธีการหาค่าเหมาะสมที่สุดจากพฤติกรรมการบินหรือการหาอาหารของฝูงสัตว์ โดยเฉพาะฝูงนกฝูงปลาอย่างเช่น มีนกฝูงหนึ่งกำลังบินสู่มหาอาหารอยู่พื้นที่หนึ่ง ซึ่งมีอาหารอยู่เพียงชั้นเดียวในบริเวณนั้น นกทุกตัวไม่รู้ว่าอาหารอยู่ที่ตำแหน่งไหน แต่สัญชาตญาณของฝูงนกจะรู้ว่ามันอยู่ห่างจากอาหารเท่าไร โดยเมื่อฝูงนกบินไปในแต่ละช่วง ฝูงนกจะหาระยะห่างของมันจากแหล่งอาหาร และนกทั้งฝูงจะเลือกบินตามนกตัวที่อยู่ใกล้อาหารมากที่สุด จากนั้นฝูงนกก็จะบินช่วงต่อไป โดยที่นกทั้งฝูงจะทำซ้ำเช่นนี้ไปเรื่อยๆ จนฝูงนกบินมาถึงแหล่งอาหาร จากวิธีการหาอาหารของฝูงนกเราจะเห็นว่า ในการกำหนดตำแหน่งระยะห่างของตัวมันจากแหล่งอาหาร และการตัดสินใจบินตามนกตัวที่บินอยู่ใกล้แหล่งอาหารมากที่สุด ในแต่ละช่วงจะใช้เวลาอันน้อยมาก จนดูเหมือนบินเกาะกลุ่มกันไปแบบต่อเนื่อง

พาทิคอลสวอมมออปติมิซเซชันเป็นการจำลองการหาอาหารของฝูงนก ในการหาค่าเหมาะสมที่สุดแบบนี้ นกแต่ละตัวในฝูงจะถูกแทนด้วยอนุภาค(particle) ในอนุภาคแต่ละตัวจะมีค่า ฟิตเนส

(fitness value) ที่บอกถึงระยะห่างของตัวมันจากแหล่งอาหาร โดยอนุภาคทั้งหมดจะบินตามอนุภาคที่มีค่าฟิตเนสที่ดีที่สุดในแต่ละช่วง(รอบ/iteration)

แนวคิดของพาทิคอลสวอมมออปติมิซเซชันจะเริ่มต้นด้วยการสุ่มหาตำแหน่งของอนุภาค (ซึ่งตำแหน่งต่างๆ ของอนุภาคเหล่านั้นก็คือคำตอบที่เป็นไปได้)ขึ้นมาชุดหนึ่งจากนั้นก็หาค่าเหมาะสมที่สุดด้วยการปรับปรุงค่าในแต่ละรอบของการตัดสินใจ โดยที่อนุภาคแต่ละตัวจะมีการปรับปรุงค่าด้วยการเปลี่ยนตำแหน่งตามค่าที่ดีที่สุด ซึ่งมี 2 ค่า ได้แก่

1. ค่าที่ดีที่สุดของอนุภาค (pbest: particle best) คือค่าตำแหน่งที่ดีที่สุดของการเคลื่อนที่ที่ผ่านมาของตัวมันเอง
2. ค่าที่ดีที่สุดของสากล (gbest : global best)คือค่าที่เป็นค่าที่ดีที่สุดของการเคลื่อนที่ที่ผ่านมาของทั้งกลุ่ม (ฝูง)



รูปที่ 2.13 ขั้นตอนการทำงานของพาทิคอลสวอมมออปติมิซเซชัน

การทำงานของพาทิคอล์สวอมมออปติมายเซชัน เป็นกระบวนการที่ทำงานเป็นรอบ (Iteration) ซึ่งในแต่ละรอบของการทำงานความเร็วของอนุภาคแต่ละตัวจะถูกปรับปรุงโดยมีตัวแปรที่สำคัญ 3 ตัว คือ

1. ความเร็วปัจจุบันของอนุภาคนั้น (velocity)
2. ข้อมูลที่อนุภาคมีอยู่ (pbest)
3. ข้อมูลรวมของอนุภาคทั้งกลุ่ม (gbest)

หลังจากนั้นอนุภาคแต่ละตัวจะปรับตำแหน่งของมันโดยใช้ความเร็วใหม่ที่คำนวณได้ โดยพิจารณาจากสมการดังต่อไปนี้

คำนวณค่าความเร็วใหม่ของอนุภาคตามสมการ (2.15)

$$v_{id}(t+1) = w(t)v_{id}(t) + c_p u(p_{id} - x_{id}(t)) + c_g u(p_{gd} - x(t)) \quad (2.15)$$

ปรับปรุงค่าตำแหน่งใหม่ของอนุภาคตามสมการ (2.16)

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1) \quad (2.16)$$

โดยที่

i	คือ ลำดับของอนุภาค
d	คือ มิติของเวกเตอร์
t	คือ รอบของการวนซ้ำ
$x_{id}(t)$	คือ ค่าของตำแหน่งของมิติที่ d ของอนุภาคตัวที่ i
$x(t)$	คือ ค่าสามารถแปลงเป็นคำตอบของปัญหา
v_{max}	คือ ความเร็วสูงสุด
$v_{id}(t)$	คือ ค่าความเร็วไม่สามารถมีค่าออกนอกช่วง $[-v_{max}, v_{max}]$
$w(t)$	คือ น้ำหนักแรงเฉื่อยใช้ในการควบคุมผลกระทบของความเร็วที่การวนซ้ำก่อนหน้า
p_{gd}	คือ ตำแหน่งที่ดีที่สุดที่ถูกพบโดยอนุภาคทั้งหมดที่มีมิติที่ d
p_{id}	คือ ตำแหน่งที่ดีที่สุดส่วนตัวของตัวพาร์ทิเคิลตัวที่ i ที่มีมิติที่ d
c_p	คือ ค่าคงที่อัตราเร่งของตำแหน่งที่ดีที่สุดส่วนตัว
c_g	คือ ค่าคงที่อัตราเร่งของตำแหน่งที่ดีที่สุดสากล

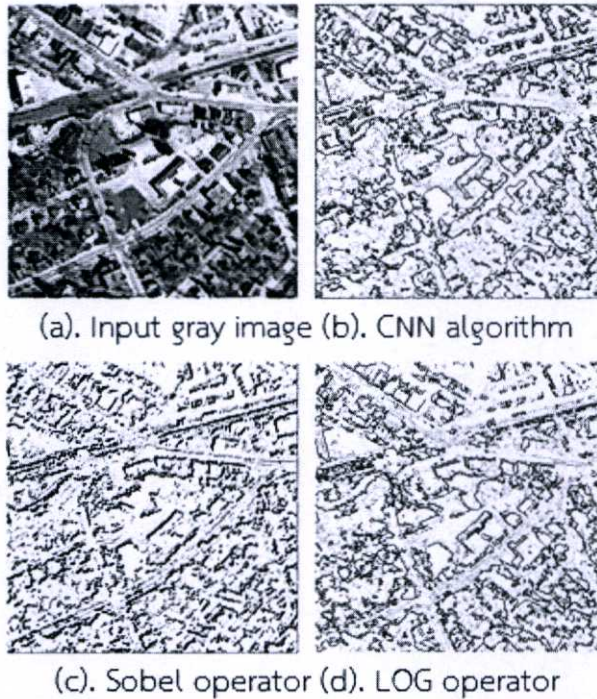
2.6 งานวิจัยที่เกี่ยวข้อง

การตรวจหาขอบของวัตถุที่ต้องการโดยใช้วิธีการของโซเบลแค่นี้ และพรีวิดิอีกทั้งนำพาทีเคลสวอมออปติไมเซชันมาวิเคราะห์ข้อมูล เพื่อประยุกต์สำหรับหาขอบด้วยวิธีต่างๆ โดยมีงานวิจัยที่เกี่ยวข้องดังนี้

Development of new edge-detection filter based on genetic algorithm: an application to a soldering joint inspection งานวิจัยชิ้นนี้ใช้สำหรับหาตรวจหาขอบการบัดกรีมาเปรียบเทียบกับผลการหาขอบจริงโดยใช้วิธีการอื่นๆเปรียบเทียบกับแม่แบบที่ใช้อัลกอริทึมเชิงพันธุกรรม (Genetic Algorithm) ในการสร้างหน้าต่างสำหรับหาขอบโดยนำหน้าต่างของตัวดำเนินการโซเบลมาประยุกต์ สำหรับการหาค่ารอยบัดกรีบนแผ่นชิพ และนำวิธีการมาเปรียบเทียบกับวิธีการอื่นๆ โดยเปรียบเทียบค่าความผิดพลาดของทุกๆวิธี

Edge Detection of Image Based on Improved Sobel Operator and Genetic Algorithms งานวิจัยชิ้นนี้ใช้การพัฒนาตัวดำเนินการโซเบลร่วมกับอัลกอริทึมเชิงพันธุกรรมในขั้นตอนการหาค่าเทรสโฮสต์เพื่อใช้หาค่าเทรสโฮสต์ที่เหมาะสมของรูปภาพเกรเดียนต์ และนำเสนออัลกอริทึมใหม่สำหรับหาค่าเทรสโฮสต์ที่เหมาะสม ซึ่งสามารถแก้ไขข้อบกพร่องของตัวดำเนินการโซเบลแบบเดิม ดังเช่น การแบ่งส่วนภาพและความไวต่อสัญญาณรบกวน ผลการทดลองจากการเปรียบเทียบแสดงให้เห็นว่าการประมวลผลมีความเร็วเพิ่มมากขึ้นและสามารถกำจัดสัญญาณรบกวนได้ดีขึ้น สิ่งเหล่านี้คือจุดแข็งของอัลกอริทึมนี้ อย่างไรก็ตามการตรวจหาเส้นขอบบางส่วนยังไม่ชัดเจนและจำเป็นต้องพัฒนาตัวดำเนินการให้ดีขึ้น

A Novel Algorithm for Edge Detection of Remote Sensing Image Base on CNN and PSO ทำการวิจัยการหาขอบของรูปภาพจากการสำรวจระยะไกลซึ่งจะเป็นภาพถ่ายจากดาวเทียมทำให้ภาพที่ได้มีแสงมารายละเอียดภาพขนาดเล็กขาดความคมชัดจึงใช้ปัญญาประดิษฐ์ในการวิเคราะห์ขอบของรูปภาพ ในการวิจัยนี้เลือกใช้อัลกอริทึม Cellular Neural Network ในการวิเคราะห์แบบคู่ขนานและใช้พาทีเคลสวอมออปติไมเซชันออกแบบแม่แบบสำหรับการหาขอบ ซึ่งเมื่อนำผลจากการวิจัยด้วยวิธีนี้มาเปรียบเทียบกับวิธีการหาขอบแบบโซเบลและ LOG สามารถเห็นความแตกต่างได้อย่างชัดเจน



รูปที่ 2.14 ตัวอย่างผลการทดลองใช้ PSO และ CNN ในการวิเคราะห์ขอบภาพ

ซึ่งจากภาพด้านบนเห็นได้ชัดว่า วิธีการนี้ช่วยให้ได้เส้นขอบที่ชัดเจนและต่อเนื่องขึ้น แต่มีส่วนผิดพลาดในการแยกขอบ สังเกตส่วนที่มีเงาพาดบนถนน วิธีการนี้ยังไม่สามารถแยกถนนออกจากเงาได้

Improve Edge Detection Using Particle Swarm Optimization การวิจัยนี้ได้พัฒนาการหาขอบโดยนำพาทิกเคิลสวอมมออปติไมเซชันมาช่วยในการเลือกทิศทางของส่วนที่เป็นขอบ และแบ่งแยกจากความเข้มของพิกเซลที่ต่างกับพิกเซลรอบข้างแล้วหาค่าเทรสโฮสต์ที่เหมาะสมสำหรับแบ่งแยกส่วนขอบกับเนื้อหาวิธีการนี้สามารถหาขอบภาพที่ชัดเจนถึงแม้ภาพต้นฉบับจะมีจุดสีรบกวนมาก แต่วิธีการนี้ซับซ้อนมากเกินไป ต้องใช้ทรัพยากรสูงในการคำนวณ

Edge detection for Medial Image based on PSO Algorithm งานวิจัยนี้ได้้นำการวิเคราะห์ขอบภาพมาประยุกต์กับทางการแพทย์ โดยนำพาทิกเคิลสวอมมออปติไมเซชันมาหาค่าที่เหมาะสมสำหรับแม่แบบโดยนำค่าที่ได้มาประยุกต์กับแม่แบบของตัวดำเนินการโซเบล วิธีการคือนำรูปภาพจากกล้องเอกซเรย์ที่เป็นภาพสีมาแปลงเป็นเกรเดียน แล้วปรับความเข้มของภาพ แล้วนำมาวิเคราะห์ขอบโดยใช้แม่แบบที่สร้างไว้ ซึ่งผลที่ได้เมื่อเปรียบเทียบภาพจะเห็นได้ชัดว่า วิธีการนี้ยังให้เส้นขอบไม่สมบูรณ์เท่าที่ควร เมื่อเทียบกับวิธีการแบบแคนนี่

บทที่ 3

การออกแบบและขั้นตอนการทดลอง

หลังจากการศึกษาเอกสารและงานวิจัยที่เกี่ยวข้องตามที่กล่าวไว้ในบทที่ 2 บทนี้จะกล่าวถึงวิธีดำเนินการวิจัยจากการศึกษาในบทที่ 2 ร่วมกับขั้นตอนวิธีที่ผู้วิจัยได้พัฒนาขึ้นเอง สำหรับวิธีดำเนินการวิจัยประกอบด้วยเครื่องมือวิจัย ขั้นตอนการทำวิจัย และการประเมินผล จะเป็นขั้นตอนวิธีดำเนินการวิจัย โดยมีรายละเอียดดังนี้

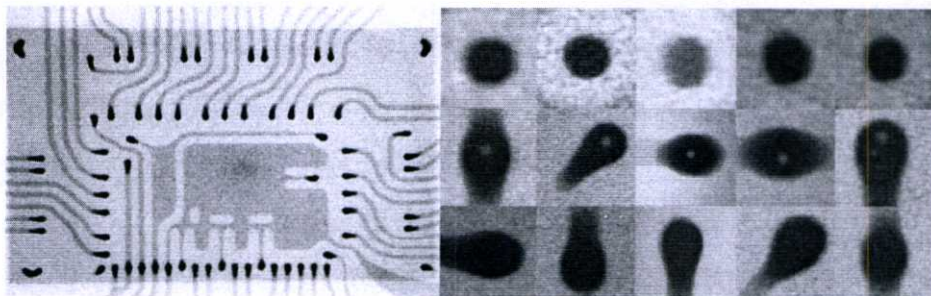
3.1 เครื่องมือวิจัย

- เครื่องคอมพิวเตอร์ CPU Intel Core i5 2.5 GHz, RAM 4 GB DDR3, Hard disk 500 GB
- ภาพถ่ายเอกซเรย์บัตรกรีที่อยู่ข้างในแผ่นวงจรขนาดเล็ก(ฟิลิปชิป)
- ระบบปฏิบัติการ Macintosh OS X, Windows 7
- โปรแกรม Matlab
- โปรแกรมสำหรับตัดแต่งภาพ Microsoft Picture Manager Tool

3.2 ขั้นตอนการทำวิจัย

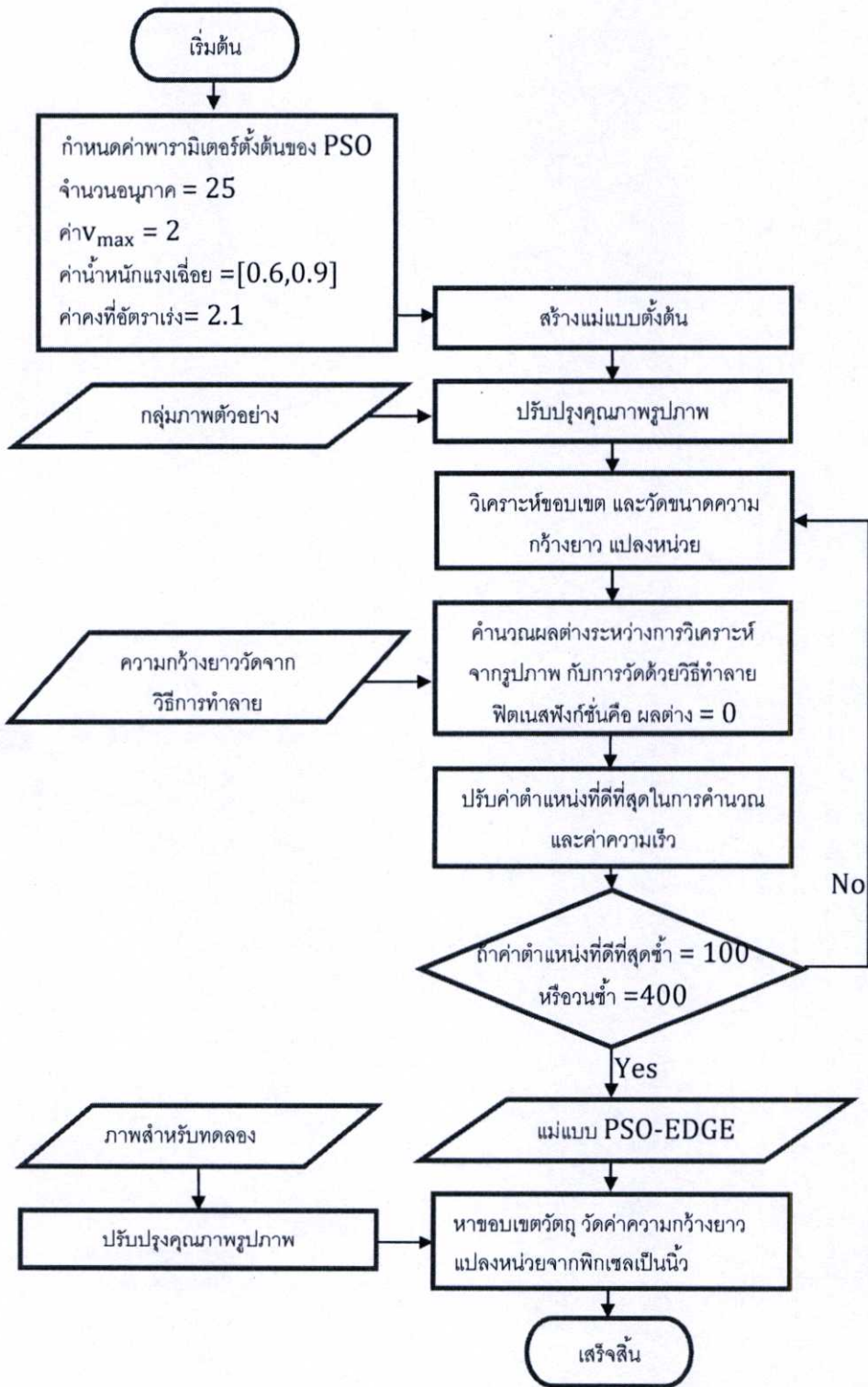
เริ่มจากการนำแผ่นฟิลิปชิปมาถ่ายภาพเอกซเรย์แล้วตัดภาพเป็นภาพย่อยๆ ได้ 129ภาพเสร็จแล้วก็นำแผ่นฟิลิปชิปมาขีดผิวเคลือบออกจนถึงตำแหน่งที่บัตรกรี แล้วใช้กล้องจุลทรรศน์ ส่องวัดขนาดความกว้าง ความยาวออกมาในหน่วยเป็นนิ้ว

จากนั้นนำภาพเอกซเรย์รอยบัตรกรีแบ่งเป็น 2 กลุ่ม กลุ่มแรกจำนวน 43 ภาพ สำหรับขั้นตอนการสร้างแม่แบบด้วยพาทิเคิลสวอมมออปติไมเซชันและกลุ่ม 2 จำนวน 86 ภาพ ใช้ในกระบวนการทดสอบ โดยรูปภาพในแต่ละกลุ่มนั้นมีขนาดรูปร่างและทิศทางใกล้เคียงกัน



รูปที่ 3.15 ตัวอย่างภาพที่ใช้ในงานวิจัย

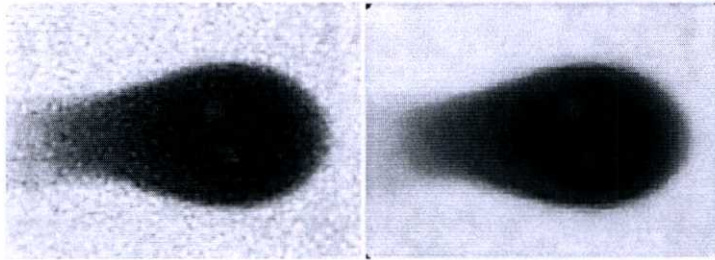
การทดลองนี้ใช้กระบวนการหาค่าแม่แบบที่เหมาะสมด้วยพาทิเคิลสวอมมอปติไมเซชัน และนำแม่แบบที่ได้มาวิเคราะห์ขอบเขตรูปภาพแล้วตรวจวัดขนาดสามารถอธิบายรายละเอียดขั้นตอนต่างๆ เป็นแผนผังการทำงานรูปที่ 3.16



รูปที่ 3.16 ผังขั้นตอนการทำงาน

3.2.1 กระบวนการปรับปรุงคุณภาพของรูปภาพ

ใช้การปรับความเข้มของแสงและการเปรียบเทียบ(contrast) เพื่อปรับให้ภาพมีความเปรียบต่างของค่าความเข้มเพิ่มขึ้นแยกจุดที่สำคัญของภาพให้ชัดเจนขึ้นและใช้ฟังก์ชันSharp Filtering และ 2-D median Filtering จากโปรแกรม MATLAB สำหรับการปรับปรุงคุณภาพรูปภาพดังรูปภาพตัวอย่างด้านล่าง



รูปที่ 3.17 ภาพก่อนและหลังผ่านการปรับปรุงคุณภาพ

3.2.2 กระบวนการหาแม่แบบที่เหมาะสมกับรูปภาพด้วยพาทิเคิลสวอมมออปติไมเซชัน

กระบวนการหาค่าความเหมาะสมใช้วิธีการพาทิเคิลสวอมมออปติไมเซชันและตัวดำเนินการโซเบลนำมาใช้ในการกำหนดรูปแบบของแม่แบบในที่จะทำการคำนวณ เนื่องจากตัวดำเนินการโซเบลที่มีการให้น้ำหนักในจุดใกล้เคียงศูนย์กลางมากที่สุด ทำให้บริเวณที่ขอบของวัตถุในภาพเด่นชัดขึ้นมาประยุกต์โดยทำการเพิ่มขนาดแม่แบบจากเดิมที่ใช้เมตริกซ์ขนาด 3×3 จำนวน 2 ทิศทางเป็นเมตริกซ์ขนาด 5×5 และเพิ่มจำนวนทิศทางการวิเคราะห์เป็น 4 ทิศทาง ได้แก่ แนวอน แนวตั้ง แนวทแยงบวก 45 องศา และแนวทแยงลบ 45 องศา

เนื่องจากภาพทดสอบของรอยบัดกรีเป็นภาพที่มีค่าการเปรียบเทียบต่ำมีสัญญาณรบกวนในภาพสูงวัตถุในภาพมีลักษณะเป็นวงกลมผสมกับวงรีการเพิ่มขนาดแม่แบบจะช่วยเก็บรายละเอียดเพื่อวิเคราะห์ขอบภาพส่วนการเพิ่มทิศเฉียงเพื่อช่วยในการวิเคราะห์ขอบภาพในด้านที่โค้งหรือเฉียงซึ่งรูปแบบของแม่แบบตามสมการที่ (3.15)-(3.18)

$$P_x = \begin{bmatrix} P_1 & P_2 & P_3 & P_4 & P_5 \\ P_6 & P_7 & P_8 & P_9 & P_{10} \\ 0 & 0 & 0 & 0 & 0 \\ -P_6 & -P_7 & -P_8 & -P_9 & -P_{10} \\ -P_1 & -P_2 & -P_3 & -P_4 & -P_5 \end{bmatrix} \quad (3.15)$$

$$P_y = \begin{bmatrix} P_1 & P_6 & 0 & -P_6 & -P_1 \\ P_2 & P_7 & 0 & -P_7 & -P_2 \\ P_3 & P_8 & 0 & -P_8 & -P_3 \\ P_4 & P_9 & 0 & -P_9 & -P_4 \\ P_5 & P_{10} & 0 & -P_{10} & -P_5 \end{bmatrix} \quad (3.16)$$

$$P_u = \begin{bmatrix} 0 & P_6 & P_2 & P_1 & P_3 \\ -P_6 & 0 & P_7 & P_8 & P_5 \\ -P_2 & -P_7 & 0 & P_9 & P_4 \\ -P_1 & -P_8 & -P_9 & 0 & P_{10} \\ -P_3 & -P_5 & -P_4 & -P_{10} & 0 \end{bmatrix} \quad (3.17)$$

$$P_v = \begin{bmatrix} -P_3 & -P_5 & -P_4 & -P_{10} & 0 \\ -P_1 & -P_8 & -P_9 & 0 & P_{10} \\ -P_2 & -P_7 & 0 & P_9 & P_4 \\ -P_6 & 0 & P_7 & P_8 & P_5 \\ 0 & P_6 & P_2 & P_1 & P_3 \end{bmatrix} \quad (3.18)$$

$$G = \sqrt{P_x^2 + P_y^2 + P_u^2 + P_v^2} \quad (3.19)$$

โดยที่

$P_1 - P_{10}$	คือ	ตัวแปรแทนค่าพารามิเตอร์ในแม่แบบ	
P_x, P_y, P_u, P_v	คือ	ค่าความแตกต่างในแนวนอน, แนวตั้ง, แนวเฉียง	45 องศา, แนวเฉียง-45 องศา
G	คือ	ค่าขนาดของฟิกเซลนั้นๆ	

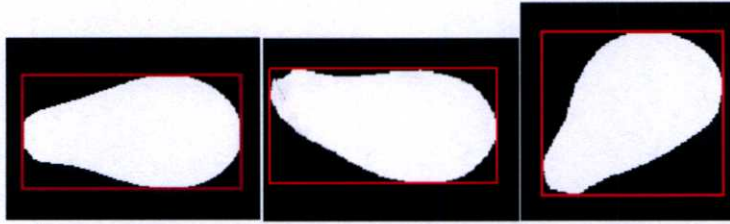
จากสมการข้างต้นเป็นสมการสำหรับคำนวณหาแม่แบบให้กับภาพรอยบัดกรี ด้วยพาทิเคิล สวอมออปติไมเซชัน มีการควบคุมจำนวนรอบในการประมวลผลและจำนวนอนุภาคสำหรับใช้ในการประมวลผลสามารถกำหนดในขั้นตอนพาทิเคิลสวอมออปติไมเซชันโดยการ Execute ไฟล์ demopsobehavior.m(Source code ในภาคผนวก)

ใช้อุณหภูมิ 15 จำนวนอนุภาคที่เหมาะสมประมาณ 24 -30 แต่ที่ใช้จำนวน 15 เนื่องจากติดข้อจำกัดทางด้านทรัพยากรเครื่องที่ใช้ระยะเวลาเพิ่มขึ้นเมื่ออนุภาคมีจำนวนมากขึ้นจากสมการ (2.15) และ (2.16) เลือกค่า v_{max} เป็น 2 หากมากกว่าจะทำให้การกระจายกว้างเกินไปไม่ได้ค่าที่เหมาะสม แต่หากน้อยกว่าก็จะเพิ่มจำนวนรอบในการหาค่าค่าน้ำหนักแรงเฉื่อย ($w(t)$) มีค่าระหว่าง 0.6 ถึง 0.9 และกำหนดค่าคงที่อัตราเร่ง (c_p, c_g) เป็น 2.1 สำหรับคำนวณค่าพารามิเตอร์ของแม่แบบที่ใช้ในการวิเคราะห์ให้เหมาะสมกับรูปภาพนั้นจะได้รับการคำนวณ โดยนำค่าเกรเดียนต์ของรูปภาพที่ผ่านการปรับปรุงคุณภาพพาทิเคิลสวอมออปติไมเซชันจะคำนวณค่าพารามิเตอร์ แล้วทำการคอนโวลูชันกับแม่แบบที่คำนวณมาได้ ผลลัพธ์ที่ได้คือเกรเดียนต์ของรูปภาพ

ต่อมาใช้ค่าเทรซโฮลด์แบ่งส่วนที่เป็นขอบภาพออกจากภาพพื้นหลัง โดยที่ให้จุดที่มีค่ามากกว่ามีค่าเทรซโฮลด์เป็น 0 และจุดที่น้อยกว่าเป็น 255 กำหนดค่าเทรซโฮลด์เป็น 200 เนื่องจากภาพมีแสงน้อยส่วนใหญ่ของขนาดของฟิกเซลที่คำนวณได้จะมีค่าสูง

จากนั้นวัดขนาดกว้างและยาวของรอยบัดกรีในรูปภาพด้วยฟังก์ชัน regionprops ของโปรแกรม Matlab โดยเลือกใช้พื้นที่ (Area) และจับขอบเขตเป็นตารางสี่เหลี่ยม (Bounding Box) เงื่อนไขในการเลือกความกว้างความยาวของรูปคือ มีค่าพื้นที่มากที่สุดในการวัด เมื่อได้ชุดข้อมูลที่มี

พื้นที่มากที่สุดแล้ว ก็นำค่าความกว้างและความยาวจากชุดข้อมูลนั้นมาทำการแปลงหน่วย โดยนำค่าความกว้างและความยาวคูณกับค่า 0.000045 เพื่อแปลงหน่วยจากพิกเซลเป็นนิ้ว



รูปที่ 3.18 แสดงการวัดความกว้างความยาวผ่านโปรแกรม MATLAB

ตามการวิเคราะห์ขอบเขตที่ได้จากแม่แบบตามค่าพารามิเตอร์ แล้วนำค่าความกว้างความยาวที่แปลงหน่วยเป็นนิ้วมาคำนวณร่วมกับค่าที่วัดจริงในแต่ละอนุภาค เพื่อหาค่าฟิตเนสโดยใช้สมการที่ (3.20)

$$\text{Fitness} = \sqrt{(W_{\text{pso}} - W_{\text{real}})^2 + (H_{\text{pso}} - H_{\text{real}})^2} \quad (3.20)$$

โดยที่

Fitness	คือ ค่าความผิดพลาดรวมทั้งความกว้างและความยาวของแต่ละอนุภาค
W_{pso}	คือ ค่าความกว้างที่วัดได้จากการหาขอบเขตในโปรแกรม
W_{real}	คือ ค่าความกว้างที่วัดได้จากวิธีการทำลาย
H_{pso}	คือ ค่าความยาวที่วัดได้จากการหาขอบเขตในโปรแกรม
H_{real}	คือ ค่าความยาวที่วัดได้จากวิธีการทำลาย

ซึ่งจะได้ค่าพารามิเตอร์ก็ต่อเมื่อฟิตเนสที่คำนวณได้นั้นน้อยที่สุดหรือเท่ากับศูนย์แต่หากค่าฟิตเนสนั้นมีค่าสูงอนุภาคนั้นจะถูกตัดทิ้งให้วนกลับไปคำนวณหาค่าพารามิเตอร์สำหรับแม่แบบใหม่จนกว่าจะได้ค่าฟิตเนสที่เข้าใกล้ศูนย์หรือให้ค่าพารามิเตอร์ที่ให้ค่าฟิตเนสน้อยที่สุดเมื่อคำนวณครบตามจำนวนรอบแล้วจากผลลัพธ์ครั้งนี้จากการวนซ้ำ 212 รอบค่าแม่แบบที่ได้จากพาทิเคิลสวมออปติไมเซชันดังต่อไปนี้

$$P_x = \begin{bmatrix} -0.053482 & -0.081306 & 0.13671 & -0.0099407 & -0.16567 \\ 0.021561 & 0.010715 & 0.068298 & -0.076705 & 0.013151 \\ 0 & 0 & 0 & 0 & 0 \\ -0.021561 & -0.010715 & -0.068298 & 0.076705 & -0.013151 \\ 0.053482 & 0.081306 & -0.13671 & 0.0099407 & 0.16567 \end{bmatrix} \quad (3.21)$$

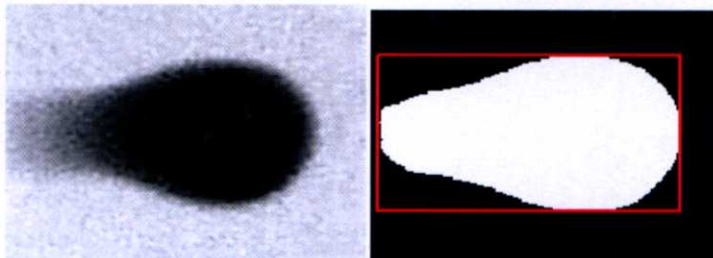
$$P_y = \begin{bmatrix} -0.053482 & 0.021561 & 0 & -0.021561 & 0.053482 \\ -0.081306 & 0.010715 & 0 & -0.010715 & 0.081306 \\ 0.13671 & 0.068298 & 0 & -0.068298 & -0.13671 \\ -0.0099407 & -0.076705 & 0 & 0.076705 & 0.0099407 \\ -0.16567 & 0.013151 & 0 & -0.013151 & 0.16567 \end{bmatrix} \quad (3.22)$$

$$P_u = \begin{bmatrix} 0 & 0.021561 & -0.081306 & -0.053482 & 0.13671 \\ -0.021561 & 0 & 0.010715 & 0.068298 & -0.16567 \\ 0.081306 & -0.010715 & 0 & -0.076705 & -0.0099407 \\ 0.053482 & -0.068298 & 0.076705 & 0 & 0.013151 \\ -0.13671 & 0.16567 & 0.0099407 & -0.013151 & 0 \end{bmatrix} \quad (3.23)$$

$$P_v = \begin{bmatrix} -0.13671 & 0.16567 & 0.0099407 & -0.013151 & 0 \\ 0.053482 & -0.068298 & 0.076705 & 0 & 0.013151 \\ 0.081306 & -0.010715 & 0 & -0.076705 & -0.0099407 \\ -0.021561 & 0 & 0.010715 & 0.068298 & -0.16567 \\ 0 & 0.021561 & -0.081306 & -0.053482 & 0.13671 \end{bmatrix} \quad (3.24)$$

3.2.3 กระบวนการวิเคราะห์ขอบเขตและวัดขนาด

เมื่อได้ค่าพารามิเตอร์สำหรับแม่แบบจากกระบวนการก่อนหน้า สร้างแม่แบบขึ้นมา 4 ทิศทางตามสมการ แล้วนำมาคอนโวลูชันกับเกรเดียนต์ นำเกรเดียนต์ที่ได้มาแบ่งส่วนภาพด้วยค่าเทรสโฮลด์สำหรับการทดลองนี้กำหนดค่าเทรสโฮลด์เป็น 200 จะแยกส่วนที่เป็นพื้นหลัง กับส่วนรอยบัตกรีในภาพ เมื่อเสร็จขั้นตอนนี้จะได้ภาพขาวดำ



รูปที่ 3.19 ตัวอย่างการวัดความกว้างความยาว

วัดขนาดกว้างและยาวของรอยบัตกรีด้วยฟังก์ชัน regionprops ของโปรแกรม Matlab โดยเลือกใช้พื้นที่และจับขอบเขตเป็นตารางสี่เหลี่ยม (Bounding Box) เลือกข้อมูลมีค่าพื้นที่มากที่สุดในการวัด นำค่าความกว้างและความยาวจากชุดข้อมูลนั้นทำการแปลงหน่วยจากพิกเซลเป็นนิ้ว และนำมาเปรียบเทียบกับขนาดที่วัดได้จากกล้องจุลทรรศน์ในวิธีการทำลาย ค่าผลต่างที่ได้มานี้จะใช้เป็นค่าผิดพลาดจากการวิเคราะห์รูปภาพ

3.2.4 กระบวนการประเมินความถูกต้อง

ใช้รูปภาพชุดเดียวกัน มาคำนวณหาขอบเขตด้วยตัวดำเนินการโซเบลแคนี้และพรีวิ
ตวัดขนาดกว้างยาวของรอยบัตกรี แล้วทำการแปลงหน่วยจากพิกเซลเป็นนิ้วด้วยการนำค่าคงที่
0.000045 มาคูณกับค่าที่วัดได้ในหน่วยพิกเซลและคำนวณค่าความผิดพลาดจากแต่ละวิธีเทียบกับ
ขนาดที่วัดได้จริง

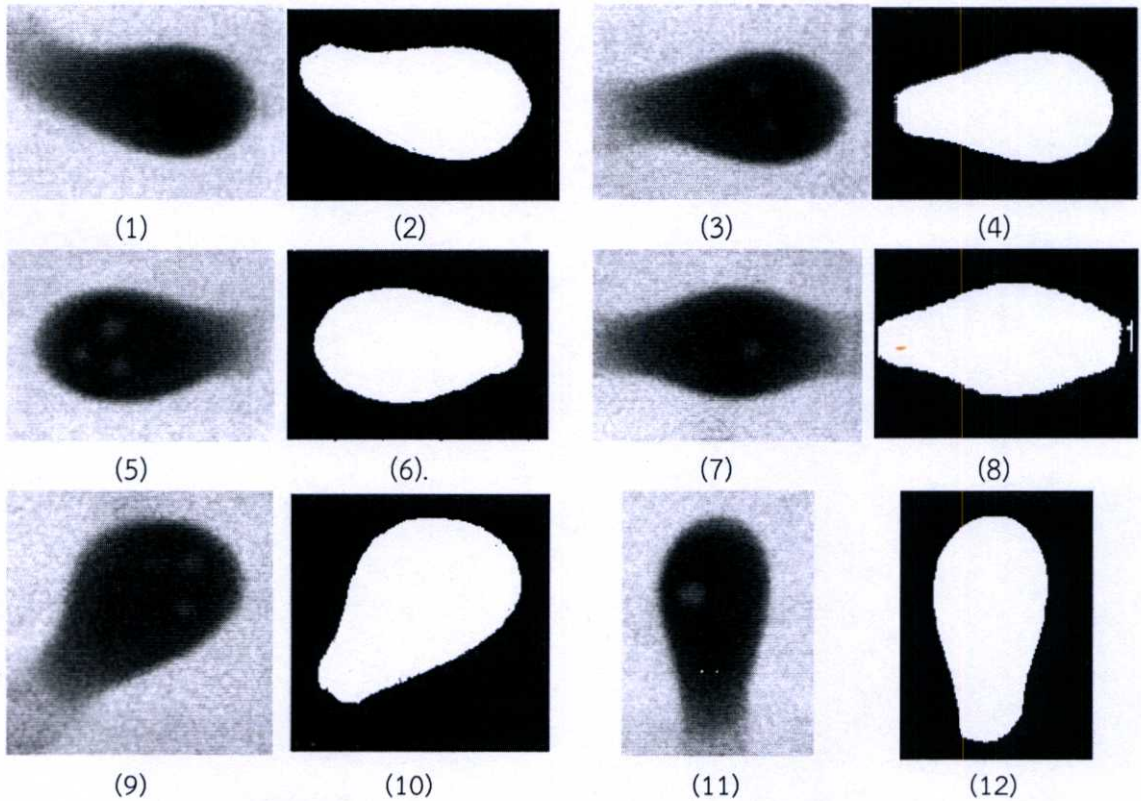
บทที่ 4

ผลการทดลองและการวิเคราะห์ผลการทดลอง

ใช้ภาพเอกซเรย์รอยบดกรีกกลุ่มที่ 2 ที่เตรียมไว้สำหรับทดลองนำมาวัดขนาดความกว้างและความยาวของรูป พร้อมทั้งเปรียบเทียบผลการทดลองกับภาพที่วิเคราะห์จากตัวดำเนินการโซเบล ตัวดำเนินการแคนนี่ และตัวดำเนินการพีวีวีดี

4.1 ผลการแยกขอบเขตรอยบดกรีกจากพื้นหลัง

ภาพรอยบดกรีกที่ผ่านการแยกขอบเขตด้วยแม่แบบจากพาทิเคิลสวอมมอปติไมเซชัน เรียกว่า "PSO-Edge" จะสามารถแยกส่วนขอบของภาพที่ค่อยๆ จางลงได้ดีอาจมีเรียบเนียนบริเวณตรงส่วนปลายที่มีสีไล่ระดับแต่สามารถแยกได้ทั้งหมดโดยไม่ต้องมีการหมุนรูปภาพ



รูปที่ 4.20 (1) (3) (5) (7) (9) (11) เป็นภาพตัวอย่างต้นฉบับ และ (2) (4) (6) (8) (10) (12) ภาพจาก PSO-Edge

4.2 ผลการวัดความกว้างและความยาวของรอยบัดกรี

ผลการวัดขนาดรอยบัดกรีด้านกว้างด้านยาวด้วยภาพที่ได้จาก PSO-Edge นำมาเปรียบเทียบกับขนาดรอยบัดกรีที่วัดจริงจากการทดลอง ตารางผลการทดลองของตัวอย่าง 30 ตัวอย่าง

ตารางที่ 4.1 แสดงผลการวัดค่าความกว้างความยาวด้วย PSO-Edge

ที่	ค่าที่วัดจริง(นิ้ว)		ค่าที่วัดจาก PSO-Edge(นิ้ว)		ผิดพลาด (นิ้ว)	
	กว้าง	ยาว	กว้าง	ยาว	กว้าง	ยาว
1	0.006580	0.003500	0.006705	0.003465	0.000125	0.000035
2	0.006560	0.003530	0.006480	0.003510	0.000080	0.000020
3	0.006490	0.003570	0.006660	0.003510	0.000170	0.000060
4	0.006730	0.003520	0.006885	0.003465	0.000155	0.000055
5	0.006020	0.003540	0.006165	0.003555	0.000145	0.000015
6	0.006620	0.003550	0.006750	0.003465	0.000130	0.000085
7	0.006430	0.003600	0.006525	0.003555	0.000095	0.000045
8	0.003560	0.006750	0.003465	0.006660	0.000095	0.000090
9	0.003480	0.006830	0.003465	0.006840	0.000015	0.000010
10	0.003470	0.006980	0.003420	0.007020	0.000050	0.000040
11	0.003460	0.006700	0.003465	0.006750	0.000005	0.000050
12	0.003690	0.006130	0.003645	0.006210	0.000045	0.000080
13	0.003610	0.006750	0.003510	0.006840	0.000100	0.000090
14	0.003630	0.006830	0.003600	0.006840	0.000030	0.000010
15	0.003700	0.006540	0.003690	0.006705	0.000010	0.000165
16	0.007940	0.003330	0.007785	0.003375	0.000155	0.000045
17	0.007440	0.003430	0.007470	0.003375	0.000030	0.000055
18	0.007590	0.003420	0.007515	0.003465	0.000075	0.000045
19	0.007840	0.003440	0.007785	0.003510	0.000055	0.000070
20	0.007710	0.003470	0.007785	0.003510	0.000075	0.000040
21	0.007870	0.003510	0.007785	0.003555	0.000085	0.000045
22	0.008020	0.003450	0.007785	0.003465	0.000235	0.000015
23	0.003800	0.006330	0.003645	0.006390	0.000155	0.000060
24	0.003560	0.007160	0.003465	0.007155	0.000095	0.000005
25	0.003720	0.006700	0.003645	0.006705	0.000075	0.000005
26	0.003790	0.006290	0.003690	0.006390	0.000100	0.000100
27	0.003560	0.006960	0.003510	0.006930	0.000050	0.000030
28	0.003500	0.006200	0.003465	0.005940	0.000035	0.000260
29	0.003340	0.006870	0.003375	0.007110	0.000035	0.000240
30	0.007820	0.003230	0.007605	0.003195	0.000215	0.000035

ค่าที่วัดได้จากการวิเคราะห์ด้วย PSO-Edge ทำให้ใกล้เคียงการวัดจริงมากต่างกันในระดับ 0.0001 นิ้วซึ่งมีความแม่นยำสูงกว่าวิธีการพื้นฐานอื่นๆมากโดยคำนวณค่าความคลาดเคลื่อนจากผลต่างระหว่างค่าที่วัดจริงผ่านวิธีการตรวจวัดแบบทำลายกับค่าที่วัดได้จากแอปพลิเคชันในแต่ละรูป ดังตารางด้านล่างนี้

ตารางที่ 4.2 ค่าความผิดพลาดด้านกว้างของ PSO-Edge แคนนี่ พรีวิต และโซเบล

ค่าผิดพลาด(นิ้ว)	PSO-Edge	แคนนี่	พรีวิต	โซเบล
ต่ำสุด	0.000005	0.000105	0.002080	0.001880
สูงสุด	0.000235	0.002980	0.006295	0.006295
เฉลี่ย	0.000091	0.000709	0.003814	0.003775

ตารางที่ 4.3 ค่าความผิดพลาดด้านยาวของ PSO-Edge แคนนี่ พรีวิต และโซเบล

ค่าผิดพลาด(นิ้ว)	PSO-Edge	แคนนี่	พรีวิต	โซเบล
ต่ำสุด	0.000005	0.000010	0.002285	0.002235
สูงสุด	0.000260	0.005480	0.005225	0.005495
ค่าเฉลี่ย	0.000063	0.000996	0.003495	0.003570

ตารางที่ 4.4 ค่าความผิดพลาดรวมของ PSO-Edge แคนนี่ พรีวิต และโซเบล

ค่าผิดพลาด(นิ้ว)	PSO-Edge	แคนนี่	พรีวิต	โซเบล
ต่ำสุด	0.000025	0.000115	0.004525	0.004225
สูงสุด	0.000295	0.008460	0.009535	0.009535
เฉลี่ย	0.000154	0.001705	0.007309	0.007345

จากตารางที่ 4.2-4.4แสดงค่าผิดพลาดต่ำสุดสูงสุด และเฉลี่ยของผลการทดลอง ในแต่ละวิธีการที่ทดลองเพื่อให้เห็นถึงประสิทธิภาพในการวิเคราะห์ภาพรอยบัดกรีที่ถูกต้องยิ่งขึ้น โดยที่สามารถดูผลการทดลองทั้งหมดได้ที่ภาคผนวก







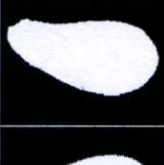
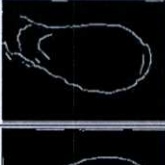
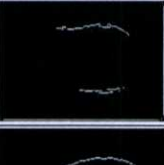

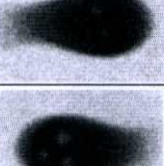
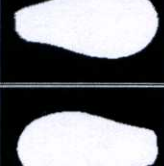
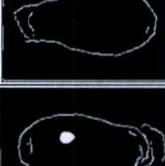


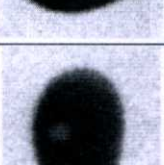
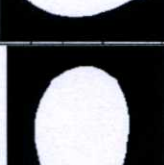
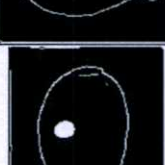


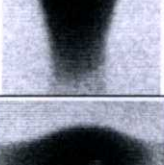


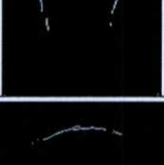



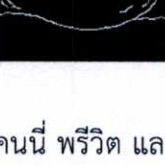
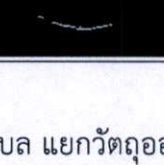

ตารางที่ 4.5 เวลาในการประมวลผลด้วย PSO-Edge แคนนี่ พรีวิต และโซเบล

วิธีการ	PSO-Edge	แคนนี่	พรีวิต	โซเบล
เวลา (วินาที)	0.358121	0.030670	0.036655	0.024860

4.3 วิเคราะห์ผลการทดลอง

นำภาพตั้งต้นมาเปรียบเทียบกับภาพที่วิเคราะห์วัตถุด้วย PSO-Edge ตัวดำเนินการโซเบล แคนนี่ และพรีวิต

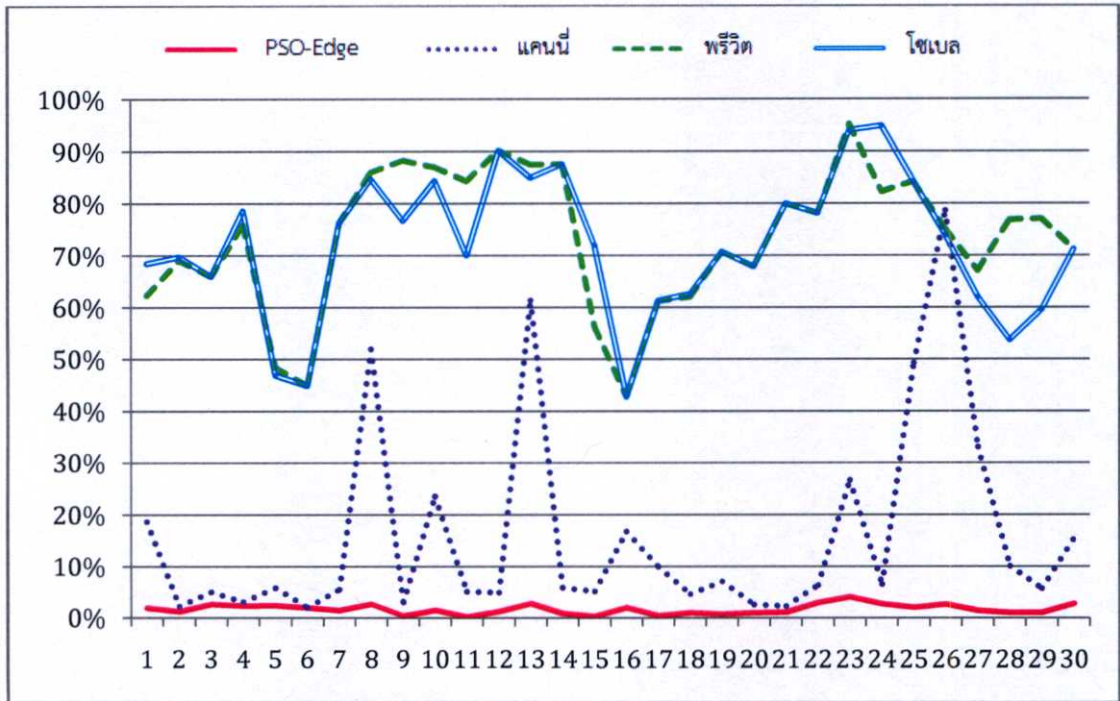
ตารางที่ 4.6 แสดงภาพที่ได้จากการวิเคราะห์ขอบเขตด้วยวิธีต่างๆ

ภาพต้นฉบับ	PSO-Edge	แคนนี่	พรีวิต	โซเบล
				
				
				
				
				
				

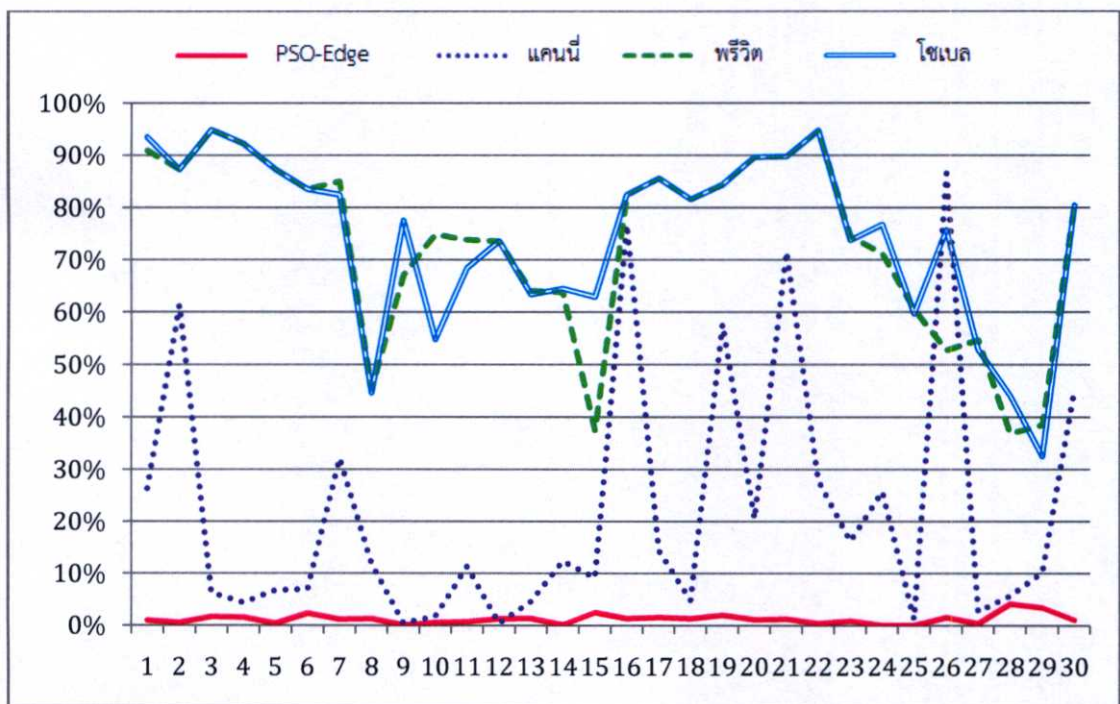
เห็นได้ชัดเจนว่าตัวดำเนินการแคนนี่ พรีวิต และโซเบล แยกวัตถุออกมาได้จากพื้นหลังแต่ ส่วนขอบของวัตถุหายไปโดยเฉพาะอย่างยิ่งส่วนทางที่ยังคลุมเครือแต่ภาพวัตถุที่ได้จาก PSO-Edge มีขอบเขตที่ชัดเจนและยังเก็บรายละเอียดของบริเวณที่ยังคลุมเครือได้ดี

ทำผลการวัดขนาดจาก PSO-Edge ตัวดำเนินการแคนนี่ พรีวิต และโซเบลมาเปรียบเทียบกับค่าความคลาดเคลื่อนจากขนาดของรอยบดกรีที่วัดผ่านแอปพลิเคชันที่ได้จากแต่ละวิธีการมาเปรียบเทียบกับขนาดรอยบดกรีที่วัดได้จริงจากรอยบดกรี 30 ภาพ โดยคำนวณหาค่าความคลาดเคลื่อนตามสมการที่ 4.24 และแสดงค่าให้ดูในรูปแบบกราฟในรูปที่ 3.17 – 3.20

$$\% \text{ ความคลาดเคลื่อน} = \frac{|\text{ค่าที่วัดจริง} - \text{ค่าที่วัดจากโปรแกรม}|}{\text{ค่าที่วัดจริง}} \times 100 \quad (4.25)$$

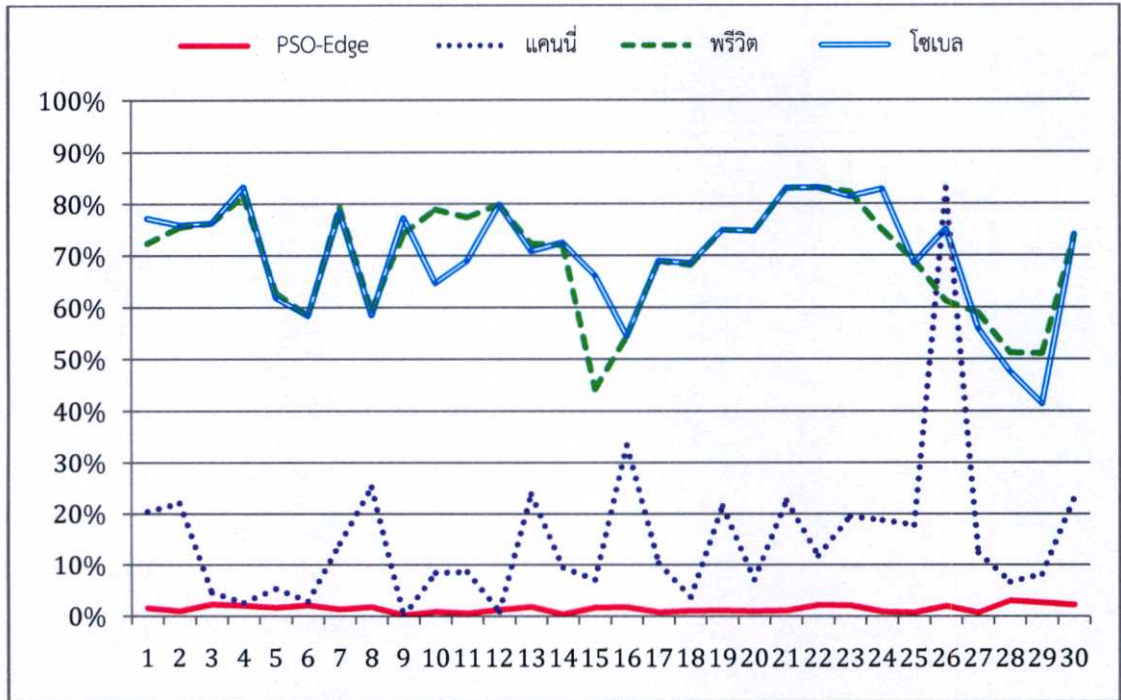


รูปที่ 4.21 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนด้านกว้างของ PSO-Edge แคนนี่ ฟรีวิต และโซเบล



รูปที่ 4.22 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนด้านกว้างของ PSO-Edge แคนนี่ ฟรีวิต และโซเบล

จากรูปที่ 4.20 และ 4.21 เห็นได้ชัดว่าตัวดำเนินการฟรีวิตและโซเบลมีความคลาดเคลื่อนสูงมากดูได้จากค่าเปอร์เซ็นต์ความคลาดเคลื่อนขอบเขตที่หาได้อาจเพราะเป็นภาพที่มีสัญญาณรบกวนมาก และภาพเอกซเรย์มีค่าความเข้มใกล้เคียงกันมาก ทำให้ยากต่อการแยกส่วนที่เป็นขอบของวัตถุภายในภาพ



รูปที่ 4.23 กราฟแสดงเปอร์เซ็นต์คลาดเคลื่อนรวมของ PSO-Edge แคนนี่ ฟรีวิต และโซเบล

จากรูปที่ 4.20 และรูปที่ 4.21 เป็นการแสดงเปอร์เซ็นต์ความคลาดเคลื่อนของแต่ละวิธีการตามด้านกว้างและด้านยาวของรูปภาพ จากกราฟจะเห็นได้ว่าบางรูปภาพตัวดำเนินการแคนนี่สามารถวิเคราะห์ได้ผิดพลาดน้อยใกล้เคียง PSO-Edge โดยเห็นได้ชัดในด้านกว้างที่มีเปอร์เซ็นต์ความคลาดเคลื่อนต่ำมาก ค่าผิดพลาดสูงสุดของด้านกว้าง คือ 0.000235 และค่าผิดพลาดสูงสุดของด้านยาว คือ 0.000260 ตัวอย่างตามรูปที่ 4.20 เมื่อเปรียบเทียบโดยรวมแล้ว PSO-Edge ทำได้ดีที่สุดตามรูปที่ 4.22

บทที่ 5

บทสรุปและข้อเสนอแนะ

บทสรุป

PSO-Edge สามารถคำนวณขนาดของรอยบัดกรีได้ถูกต้องกว่าตัวแคนนี่ พรีวิต และโซเบล โดยเห็นได้จากค่าความคลาดเคลื่อนที่ต่ำกว่าทุกวิธีการ ทั้งด้านกว้างและด้านยาว แม้ตัวดำเนินการแคนนี่จะวิเคราะห์ค่าขอบเขตได้ดีในบางรูปภาพ และเหตุผลที่ทำให้ PSO-Edge สามารถวิเคราะห์รูปภาพได้ดีกว่านั้นเนื่องจากใช้แม่แบบที่มีขนาดใหญ่กว่า เพราะตัวดำเนินการแคนนี่ พรีวิต และโซเบลนั้นใช้แม่แบบที่มีขนาด 3x3 ส่วนใน PSO-Edge นี้ใช้แม่แบบขนาด 5x5 ทำให้สามารถเปรียบเทียบความเข้มกับพิกเซลใกล้เคียงกว้างกว่า เหตุผลต่อมาคือเพราะเพิ่มทิศทางในการคำนวณจากที่มีเพียง 2 แม่แบบ มีทิศทางในแนวตั้งและแนวนอน แต่ PSO-Edge ใช้แม่แบบแนวเฉียงเพิ่มขึ้นมาอีก 2 ทิศทาง คือ 45 และ -45 องศาทำให้เปรียบเทียบการเปลี่ยนแปลงความเข้มของส่วนขอบภาพที่โค้งมนได้ดีกว่าและที่สำคัญที่สุด คือ มีการคำนวณค่าพารามิเตอร์ในแม่แบบที่เหมาะสมสำหรับภาพเอกซเรย์รอยบัดกรีด้วยพาทิเคิลสวอมมอปติไมเซชันที่เลือกค่าพารามิเตอร์ที่เหมาะสมให้กับแต่ละตำแหน่งเพื่อให้หาค่าขอบได้ถูกต้องมากที่สุด ด้วยเหตุผลทั้งหมดที่ได้กล่าวมาทำให้ผลที่ได้มีความถูกต้องมากกว่าวิธีการต่างๆ ที่นำมาทดลองเปรียบเทียบ แต่เนื่องจากขนาดและทิศทางที่เพิ่มมากกว่าส่งผลต่อความเร็วในการประมวลผลโดยตรง ทำให้ PSO-Edge ใช้เวลาในการประมวลผลนานกว่าตัวแคนนี่ พรีวิต และโซเบล

ข้อเสนอแนะ

ส่วนขอบภาพด้านละ 4 พิกเซล จะไม่ถูกนำมาคำนวณ เนื่องจากแม่แบบที่ใช้ในการวิเคราะห์ขอบเขตมีขนาด 5x5 อาจทำให้ขนาดที่วัดได้มีความผิดพลาดเพราะส่วนของวัตถุที่อยู่ในบริเวณพื้นที่ขอบของรูปภาพจะไม่ได้รับการวิเคราะห์

วิธีการ PSO-Edge ใช้เวลานานกว่าทุกกระบวนการที่นำมาเปรียบเทียบ เนื่องจากขนาดแม่แบบมีขนาดใหญ่กว่า และทิศทางที่เพิ่มขึ้นมา อาจใช้เทคนิคทางด้านโปรแกรมเพื่อเพิ่มประสิทธิภาพในการประมวลผล

บรรณานุกรม

- ชาติรี กอบัวแก้ว. 2550. "การจำแนกพระผงดโดยการเปรียบเทียบลักษณะพิเศษ." วิทยานิพนธ์ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร.
- บุญนที ศักดิ์บุญญารัตน์. 2548. "ระบบรู้จำภาพเจดีย์ไทย." วิทยานิพนธ์ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร.
- ศิริพร บุญเปลี่ยนพล. 2548. "การประยุกต์วิธีโมติฟายเจเนอรัลไลพ์ฮัฟทรานส์ฟอร์ม สำหรับรู้จำภาพจิตรกรรมฝาผนัง กรณีศึกษา:วัดชนะสงครามราชวรมหาวิหาร กรุงเทพมหานคร." วิทยานิพนธ์ปริญญาวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคอมพิวเตอร์ บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร.
- อรรถนิตติ วงศ์จักร์, กฤตวัน ศิริบุรณ์ และบุญธีร์ เครือตราชู. 2554. "การเพิ่มความสามารถในการค้นหาพื้นที่ใกล้เคียงของเจเนติก อัลกอริทึม โดยใช้เทคนิค พาดิเคิล สวอม ออปติไมเซชัน." วารสารเทคโนโลยีสารสนเทศ. ปีที่ 7 ฉบับที่ 14.
- Alipoor M., Imandoost S., Haddadnia J. 2010. "Designing edge detection filters using Particle Swarm Optimization." **Electrical Engineering (ICEE)**. 2010 : 548-552.
- A.Jevtić, D.Andina. 2010. "Adaptive artificial ant colonies for edge detection in digital images" **IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society**. 2010 : 2813-2816.
- A. Khunteta, D. Ghosh. 2013. "Edge detection via fuzzy rule-based edge strength estimation and optimal threshold selection using PSO." **Industrial and Information Systems (ICIIS) 8th IEEE International Conference**. 2013 :560-565.
- Noor Elaiza Abdul Khalid, MazaniManaf, MohdEzane Aziz. "Fusion of Fuzzy Heuristic and Particle Swarm Optimization as an edge detector." **Information Retrieval & Knowledge Management, (CAMP)**. 2010 : 250-254.
- Saenthon A, Kaitwanidvilai S. "Development of new edge-detection filter based on genetic algorithm: an application to a soldering joint inspection." **The International Journal of Advanced Manufacturing Technology**. Volume 46, Issue 9-12,(2010) : 1009-1019.
- Setayesh M, Mengjie Zhang, Johnston M. "Improving edge detection using particle swarm optimisation." **Image and Vision Computing New Zealand (IVCNZ), 25th International Conference, (2010) : 1-8.**

- Setayesh M, Mengjie Zhang, Johnston M. "Edge Detection Using Constrained Discrete Particle Swarm Optimisation in Noisy Image." 2011. **IEEE Congress of Evolutionary Computation (CEC)**, United States of America, (2011) : 246 –253.
- Setayesh M, Mengjie Zhang, Johnston M. "Effect of Static and Dynamic Topologies in Particle Swarm Optimisation for Edge Detection in Noisy Images." **IEEE Congress on Evolutionary Computation**. 10-15 June 2012, Australia:1–8.
- Setayesh M, Mengjie Zhang, Johnston M. "A novel particle swarm optimisation approach to detecting continuous, thin and smooth edges in noisy images." **Information Sciences**. Volume 246. 10 October 2013 : 28-51.
- Shelda Mohan and T.R. Mahesh. "Particle Swarm Optimization Based Contrast Limited Enhancement for Mammogram Images." **Intelligent Systems and Control (ISCO)**, 7th International Conference, 2013 : 384-388.
- SHI Zhen-gang, LI Qin-zi,"Edge detection for Medical Image based on PSO Algorithm." 2010. **Third International Conference on Intelligent Networks and Intelligent Systems IEEE**. Shenyang : 84-87.
- Etemada Ali S., Whiteb T. "An ant-inspired algorithm for detection of image edge features." **Applied Soft Computing**. Volume 11, Issue 8, December 2011: 4883–4893.
- TaghizadehM., HajipoorM. 2011. "A Hybrid Algorithm for Segmentation of MRI Images Based on Edge Detection" **International Conference of Soft Computing and Pattern Recognition (SoCPaR)**. Dalian : 107 – 111.
- Wang Jianlai, Yang Chunling, Sun Chao. 2009. "A Novel Algorithm for Edge Detection of Remote Sensing Image Based on CNN and PSO." **Image and Signal Processing**. CISP '09. 2nd International Congress onTianjin : 1 - 5.

ภาคผนวก

ภาคผนวก ก ภาพผลการทดลอง

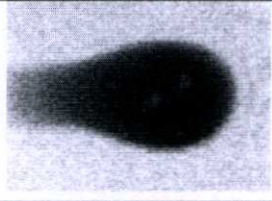

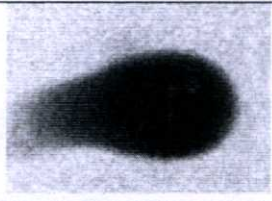

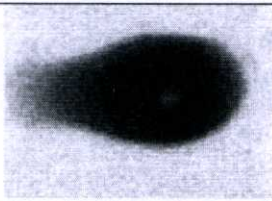

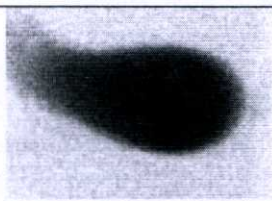

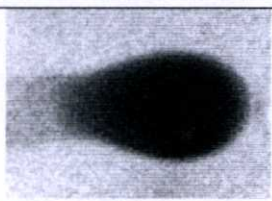

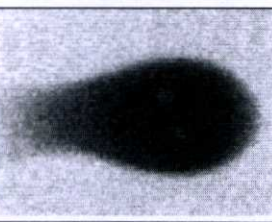

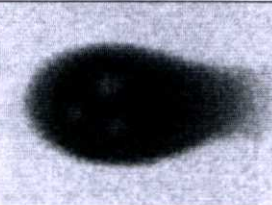

ภาคผนวก ข ผลการวัดขนาดรอยบัดกรี

ภาคผนวก ค ค่าผิดพลาดของแต่ละวิธีการ


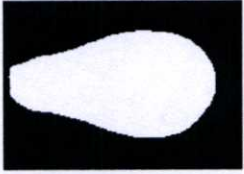


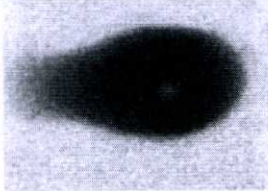

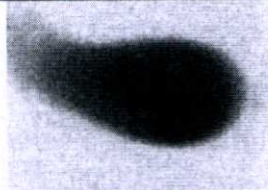

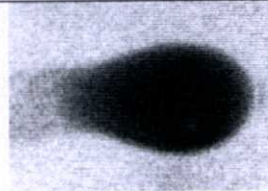

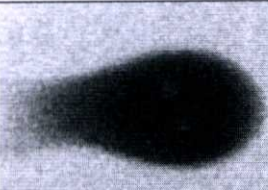



ภาคผนวก ง Source Code

ภาคผนวก ก
ภาพผลการทดลอง








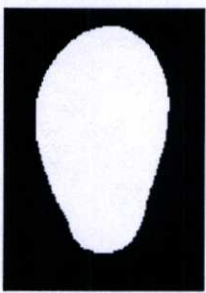


ภาพผลการทดลอง

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
1		
2		
3		
4		
5		
6		
7		





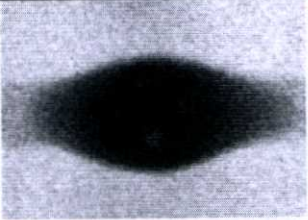

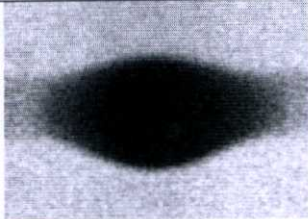




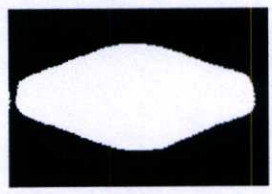
ภาพผลการทดลอง

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
1		
2		
3		
4		
5		
6		
7		

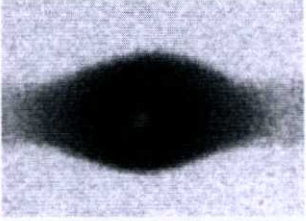

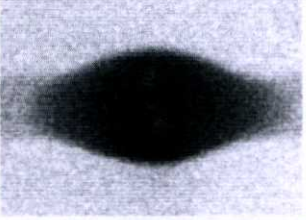

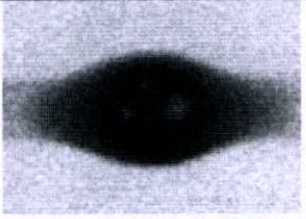
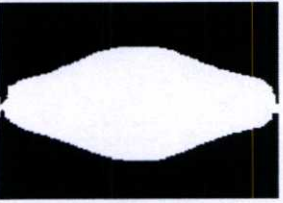






ภาพผลการทดลอง (ต่อ)

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
8		
9		
10		
11		
12		





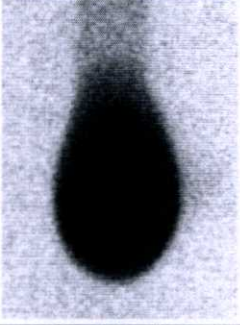

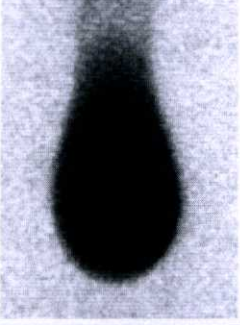



ภาพผลการทดลอง (ต่อ)

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
13		
14		
15		
16		
17		
18		

ภาพผลการทดลอง (ต่อ)

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
19		
20		
21		
22		
23		
24		

ภาพผลการทดลอง (ต่อ)

ที่	ภาพตั้งต้น	ภาพผลการทดลอง
25		
26		
27		
28		
29		

ภาคผนวก ข
ผลการวัดขนาดรอยบัดกรี

ผลการวัดความยาวด้านกว้างหน่วยเป็นนิ้ว

ที่	ค่าวัดได้จริง	PSO-Edge	แคมนี่	พรีวิต	โซเบล
1	0.00658	0.006705	0.005355	0.002475	0.00207
2	0.00656	0.00648	0.006705	0.002025	0.00198
3	0.00649	0.00666	0.006165	0.002205	0.002205
4	0.00673	0.006885	0.006525	0.00162	0.00144
5	0.00602	0.006165	0.00567	0.003105	0.003195
6	0.00662	0.00675	0.00675	0.003645	0.003645
7	0.00643	0.006525	0.006075	0.00153	0.00153
8	0.00356	0.003465	0.00171	0.000495	0.00054
9	0.00348	0.003465	0.003375	0.000405	0.00081
10	0.00347	0.00342	0.002655	0.00045	0.00054
11	0.00346	0.003465	0.003285	0.00054	0.001035
12	0.00369	0.003645	0.00351	0.00036	0.00036
13	0.00361	0.00351	0.001395	0.00045	0.00054
14	0.00363	0.0036	0.00342	0.00045	0.00045
15	0.0037	0.00369	0.00351	0.00162	0.001035
16	0.00794	0.007785	0.006615	0.004545	0.004545
17	0.00744	0.00747	0.006705	0.00288	0.00288
18	0.00759	0.007515	0.007245	0.00288	0.002835
19	0.00784	0.007785	0.00729	0.002295	0.002295
20	0.00771	0.007785	0.007515	0.002475	0.002475
21	0.00787	0.007785	0.007695	0.001575	0.001575
22	0.00802	0.007785	0.007515	0.001755	0.001755
23	0.0038	0.003645	0.00279	0.00018	0.000225
24	0.00356	0.003465	0.00333	0.00063	0.00018
25	0.00372	0.003645	0.00189	0.000585	0.000585
26	0.00379	0.00369	0.00081	0.000945	0.00099
27	0.00356	0.00351	0.002385	0.00117	0.00135
28	0.0035	0.003465	0.00315	0.00081	0.00162
29	0.00334	0.003375	0.00315	0.000765	0.00135
30	0.00782	0.007605	0.006615	0.00225	0.00225

ผลการวัดความยาวด้านยาวหน่วยเป็นนิ้ว

ที่	ค่าวัดได้จริง	PSO-Edge	แค่นี้	พรีวิต	โซเบล
1	0.00350	0.003465	0.002565	0.000315	0.000225
2	0.00353	0.00351	0.00135	0.00045	0.00045
3	0.00357	0.00351	0.00333	0.00018	0.00018
4	0.00352	0.003465	0.00369	0.00027	0.00027
5	0.00354	0.003555	0.003285	0.00045	0.00045
6	0.00355	0.003465	0.003285	0.000585	0.000585
7	0.00360	0.003555	0.00243	0.00054	0.00063
8	0.00675	0.00666	0.005895	0.00369	0.003735
9	0.00683	0.00684	0.00684	0.00225	0.00153
10	0.00698	0.00702	0.007155	0.001755	0.00315
11	0.00670	0.00675	0.005895	0.001755	0.002115
12	0.00613	0.00621	0.00612	0.00162	0.00162
13	0.00675	0.00684	0.00639	0.00243	0.002475
14	0.00683	0.00684	0.00594	0.002475	0.00243
15	0.00654	0.006705	0.005895	0.004095	0.00243
16	0.00333	0.003375	0.000765	0.000585	0.000585
17	0.00343	0.003375	0.002925	0.000495	0.000495
18	0.00342	0.003465	0.00324	0.00063	0.00063
19	0.00344	0.00351	0.00144	0.00054	0.00054
20	0.00347	0.00351	0.002745	0.00036	0.00036
21	0.00351	0.003555	0.00099	0.00036	0.00036
22	0.00345	0.003465	0.002475	0.00018	0.00018
23	0.00633	0.00639	0.005265	0.00162	0.001665
24	0.00716	0.007155	0.005265	0.00207	0.001665
25	0.00670	0.006705	0.00657	0.002655	0.0027
26	0.00629	0.00639	0.00081	0.00297	0.00153
27	0.00696	0.00693	0.0072	0.00315	0.003285
28	0.00620	0.00594	0.005805	0.003915	0.003465
29	0.00687	0.00711	0.00612	0.00423	0.004635
30	0.00323	0.003195	0.001755	0.00063	0.00063

ภาคผนวก ค
ค่าความคลาดเคลื่อนจากการวัดขนาดรอยบัดกรี

ค่าผิดพลาดจากการวัดรอยบัดกรี

ที่	ด้านกว้าง				ด้านยาว			
	วิธีใหม่	แค่นี้	พรีวิต	โซเบล	วิธีใหม่	แค่นี้	พรีวิต	โซเบล
1	0.000125	0.001225	0.004105	0.00451	0.000035	0.000935	0.003185	0.003275
2	0.00008	0.000145	0.004535	0.00458	0.00002	0.00218	0.00308	0.00308
3	0.00017	0.000325	0.004285	0.004285	0.00006	0.00024	0.00339	0.00339
4	0.000155	0.000205	0.00511	0.00529	0.000055	0.00017	0.00325	0.00325
5	0.000145	0.00035	0.002915	0.002825	0.000015	0.000255	0.00309	0.00309
6	0.00013	0.00013	0.002975	0.002975	0.000085	0.000265	0.002965	0.002965
7	0.000095	0.000355	0.0049	0.0049	0.000045	0.00117	0.00306	0.00297
8	0.000095	0.00185	0.003065	0.00302	0.00009	0.000855	0.00306	0.003015
9	0.000015	0.000105	0.003075	0.00267	0.00001	0.00001	0.00458	0.0053
10	0.00005	0.000815	0.00302	0.00293	0.00004	0.000175	0.005225	0.00383
11	0.000005	0.000175	0.00292	0.002425	0.00005	0.000805	0.004945	0.004585
12	0.000045	0.00018	0.00333	0.00333	0.00008	0.00001	0.00451	0.00451
13	0.0001	0.002215	0.00316	0.00307	0.00009	0.00036	0.00432	0.004275
14	0.00003	0.00021	0.00318	0.00318	0.00001	0.00089	0.004355	0.0044
15	0.00001	0.00019	0.00208	0.002665	0.000165	0.000645	0.002445	0.00411
16	0.000155	0.001325	0.003395	0.003395	0.000045	0.002565	0.002745	0.002745
17	0.00003	0.000735	0.00456	0.00456	0.000055	0.000505	0.002935	0.002935
18	0.000075	0.000345	0.00471	0.004755	0.000045	0.00018	0.00279	0.00279
19	0.000055	0.00055	0.005545	0.005545	0.00007	0.002	0.0029	0.0029
20	0.000075	0.000195	0.005235	0.005235	0.00004	0.000725	0.00311	0.00311
21	0.000085	0.000175	0.006295	0.006295	0.000045	0.00252	0.00315	0.00315
22	0.000235	0.000505	0.006265	0.006265	0.000015	0.000975	0.00327	0.00327
23	0.000155	0.00101	0.00362	0.003575	0.00006	0.001065	0.00471	0.004665
24	0.000095	0.00023	0.00293	0.00338	0.000005	0.001895	0.00509	0.005495
25	0.000075	0.00183	0.003135	0.003135	0.000005	0.00013	0.004045	0.004
26	0.0001	0.00298	0.002845	0.0028	0.0001	0.00548	0.00332	0.00476
27	0.00005	0.001175	0.00239	0.00221	0.00003	0.00024	0.00381	0.003675
28	0.000035	0.00035	0.00269	0.00188	0.00026	0.000395	0.002285	0.002735
29	0.000035	0.00019	0.002575	0.00199	0.00024	0.00075	0.00264	0.002235
30	0.000215	0.001205	0.00557	0.00557	0.000035	0.001475	0.0026	0.0026

ภาคผนวก ง
Source Code

```

% demopsobehavior.m
% demo of the pso.m function
% the pso tries to find the minimum of the f6 function, a standard
% benchmark
%
% on the plots, blue is current position, green is Pbest, and red is
Gbest

% Brian Birge
% Rev 3.0
% 2/27/06

clear all
close all
clc
help demopsobehavior
warning off

functnames =
{'ackley', 'alpine', 'DeJong_f2', 'DeJong_f3', 'DeJong_f4', ...
'Foxhole', 'Griewank', 'NDparabola', ...
'Rastrigin', 'Rosenbrock', 'f6', 'f6mod', 'tripod', ...
'f6_bubbles_dyn', 'f6_linear_dyn', 'f6_spiral_dyn', 'genEdge', 'genEdgeTh
ree'};
disp('Static test functions, minima don''t change w.r.t.
time/iteration:');
disp(' 1) Ackley');
disp(' 2) Alpine');
disp(' 3) DeJong_f2');
disp(' 4) DeJong_f3');
disp(' 5) DeJong_f4');
disp(' 6) Foxhole');
disp(' 7) Griewank');
disp(' 8) NDparabola (for this demo N = 2)');
disp(' 9) Rastrigin');
disp('10) Rosenbrock');
disp('11) Schaffer f6');
disp('12) Schaffer f6 modified (5 f6 functions translated from each
other)');
disp('13) Tripod');
disp(' ');
disp('Dynamic test functions, minima/environment evolves over
time/iteration:');
disp('14) f6_bubbles_dyn');
disp('15) f6_linear_dyn');
disp('16) f6_spiral_dyn');
disp('17) genEdge');
disp('18) genEdgeThree');

functchc=input('Choose test function ? ');
functname = functnames{functchc};

disp(' ');
disp('1) Intense graphics, shows error topology and surfing
particles');
disp('2) Default PSO graphing, shows error trend and particle
dynamics');
disp('3) no plot, only final output shown, fastest');
plotfcn=input('Choose plotting function ? ');
if plotfcn == 1

```

```

    plotfcn = 'goplotpso4demo';
    shw      = 1; % how often to update display
elseif plotfcn == 2
    plotfcn = 'goplotpso';
    shw      = 1; % how often to update display
else
    plotfcn = 'goplotpso';
    shw      = 0; % how often to update display
end

% set flag for 'dynamic function on', only used at very end for
tracking plots
dyn_on = 0;
if functchc==15 | functchc == 16 | functchc == 17 | functchc == 18
    dyn_on = 1;
end

%xrng=input('Input search range for X, e.g. [-10,10] ? ');
%yrng=input('Input search range for Y ? ');
xrng=[-2,2];
yrng=[-2,2];
disp(' ');
% if =0 then we look for minimum, =1 then max
disp('0) Minimize')
disp('1) Maximize')
minmax=0;%input('Choose search goal ? ');
% minmax=0;
disp(' ');
mvden = input('Max velocity divisor (2 is a good choice) ? ');
disp(' ');
ps     = input('How many particles (24 - 30 is common)? ');
disp(' ');
disp('0) Common PSO - with inertia');
disp('1) Trelea model 1');
disp('2) Trelea model 2');
disp('3) Clerc Type 1" - with constriction');
modl   = input('Choose PSO model ? ');
% note: if errgoal=NaN then unconstrained min or max is performed
if minmax==1
% errgoal=0.97643183; % max for f6 function (close enough for
termination)
    errgoal=NaN;
else
% errgoal=0; % min
    errgoal=NaN;
end
minx = xrng(1);
maxx = xrng(2);
miny = yrng(1);
maxy = yrng(2);

%-----
-----
    dims=10;
    varrange=[];
    mv=[];
for i=1:dims
    varrange=[varrange;minx maxx];
%mv=[mv;(varrange(i,2)-varrange(i,1))/mvden];

```

```

        mv = [mv; (varrange(i,2)-varrange(i,1))/mvden];
end

ac      = [2.1,2.1]; % acceleration constants, only used for modl=0
Iwt     = [0.9,0.6]; % inertia weights, only used for modl=0
epoch   = 400; % max iterations
wt_end  = 100; % iterations it takes to go from Iwt(1) to Iwt(2),
only for modl=0
errgrad = 1e-5; %1e-99 % lowest error gradient tolerance
errgraditer=100; % max # of epochs without error change >= errgrad
PSOseed = 0; % if=1 then can input particle starting positions,
if= 0 then all random
% starting particle positions (first 20 at zero, just for an example)
PSOseedValue = repmat([0],ps-10,1);

psoparams=...
[shw epoch ps ac(1) ac(2) Iwt(1) Iwt(2) ...
wt_end errgrad errgraditer errgoal modl PSOseed];

% run pso
% vectorized version
[pso_out,tr,te]=pso_Trelea_vectorized(funcname, dims,...
mv, varrange, minmax, psoparams,plotfcn,PSOseedValue);

%-----
% display best params, this only makes sense for static functions,
for dynamic
% you'd want to see a time history of expected versus optimized
global best
% values.
disp(' ');
disp(' ');
disp(['Best fit parameters: ']);
disp([' cost = ',funcname,'( [ input1, input2 ] )']);
disp(['-----']);
disp([' input1 = ',num2str(pso_out(1))]);
disp([' input2 = ',num2str(pso_out(2))]);
disp([' input3 = ',num2str(pso_out(3))]);
disp([' input4 = ',num2str(pso_out(4))]);
disp([' input5 = ',num2str(pso_out(5))]);
disp([' input6 = ',num2str(pso_out(6))]);
disp([' input7 = ',num2str(pso_out(7))]);
disp([' input8 = ',num2str(pso_out(8))]);
disp([' input9 = ',num2str(pso_out(9))]);
disp([' input10 = ',num2str(pso_out(10))]);
disp([' cost = ',num2str(pso_out(11))]);
disp([' mean cost = ',num2str(mean(te))]);
disp([' # of epochs = ',num2str(tr(end))]);

%% optional, save picture
%set(gcf,'InvertHardcopy','off');
%print -dmeta
%print('-djpeg',['demoPSOBehavior.jpg']);

% pso_Trelea_vectorized.m
% a generic particle swarm optimizer
% to find the minimum or maximum of any

```

```

% MISO matlab function
%
% Implements Common, Trelea type 1 and 2, and Clerc's class 1". It
will
% also automatically try to track to a changing environment (with
varied
% success - BKB 3/18/05)
%
% This vectorized version removes the for loop associated with
particle
% number. It also *requires* that the cost function have a single
input
% that represents all dimensions of search (i.e., for a function that
has 2
% inputs then make a wrapper that passes a matrix of ps x 2 as a
single
% variable)
%
% Usage:
% [optOUT]=PSO(funcname,D)
% or:
% [optOUT,tr,te]=...
%
PSO(funcname,D,mv,VarRange,minmax,PSOparams,plotfcn,PSOseedValue)
%
% Inputs:
% funcname - string of matlab function to optimize
% D - # of inputs to the function (dimension of problem)
%
% Optional Inputs:
% mv - max particle velocity, either a scalar or a vector of
length D
% (this allows each component to have it's own max
velocity),
% default = 4, set if not input or input as NaN
%
% VarRange - matrix of ranges for each input variable,
% default -100 to 100, of form:
% [ min1 max1
%   min2 max2
%   ...
%   minD maxD ]
%
% minmax = 0, funct minimized (default)
%         = 1, funct maximized
%         = 2, funct is targeted to P(12) (minimizes distance to
errgoal)
%
% PSOparams - PSO parameters
% P(1) - Epochs between updating display, default = 100. if 0,
%       no display
% P(2) - Maximum number of iterations (epochs) to train, default
= 2000.
% P(3) - population size, default = 24
%
% P(4) - acceleration const 1 (local best influence), default =
2
% P(5) - acceleration const 2 (global best influence), default =
2
% P(6) - Initial inertia weight, default = 0.9
% P(7) - Final inertia weight, default = 0.4

```

```

% P(8) - Epoch when inertial weight at final value, default =
1500
% P(9)- minimum global error gradient,
%         if abs(Gbest(i+1)-Gbest(i)) < gradient over
%         certain length of epochs, terminate run, default =
1e-25
% P(10)- epochs before error gradient criterion terminates run,
%         default = 150, if the SSE does not change over 250
epochs
%
%         then exit
% P(11)- error goal, if NaN then unconstrained min or max,
default=NaN
% P(12)- type flag (which kind of PSO to use)
%         0 = Common PSO w/intertia (default)
%         1,2 = Trelea types 1,2
%         3 = Clerc's Constricted PSO, Type 1"
% P(13)- PSOseed, default=0
%         = 0 for initial positions all random
%         = 1 for initial particles as user input
%
% plotfcn - optional name of plotting function, default
'goplotpso',
%         make your own and put here
%
% PSOseedValue - initial particle position, depends on P(13), must
be
%         set if P(13) is 1 or 2, not used for P(13)=0,
needs to
%         be nXm where n<=ps, and m<=D
%         If n<ps and/or m<D then remaining values are set
random
%         on Varrange
% Outputs:
% optOUT - optimal inputs and associated min/max output of
function, of form:
%         [ bestin1
%           bestin2
%           ...
%           bestinD
%           bestOUT ]
%
% Optional Outputs:
% tr      - Gbest at every iteration, traces flight of swarm
% te      - epochs to train, returned as a vector 1:endepoch
%
% Example: out=psos_Trelea_vectorized('f6',2)

% Brian Birge
% Rev 3.3
% 2/18/06

function [OUT,varargout]=psos_Trelea_vectorized(funcname,D,varargin)

rand('state',sum(100*clock));
if nargin < 2
    error('Not enough arguments. ');
end

% PSO PARAMETERS
if nargin == 2    % only specified funcname and D

```

```

VRmin=ones(D,1)*-100;
VRmax=ones(D,1)*100;
VR=[VRmin,VRmax];
minmax = 0;
P = [];
mv = 4;
plotfcn='goplotpso';
elseif nargin == 3 % specified functname, D, and mv
VRmin=ones(D,1)*-100;
VRmax=ones(D,1)*100;
VR=[VRmin,VRmax];
minmax = 0;
mv=varargin{1};
if isnan(mv)
    mv=4;
end
P = [];
plotfcn='goplotpso';
elseif nargin == 4 % specified functname, D, mv, Varrange
    mv=varargin{1};
if isnan(mv)
    mv=4;
end
VR=varargin{2};
minmax = 0;
P = [];
plotfcn='goplotpso';
elseif nargin == 5 % Functname, D, mv, Varrange, and minmax
    mv=varargin{1};
if isnan(mv)
    mv=4;
end
VR=varargin{2};
minmax=varargin{3};
P = [];
plotfcn='goplotpso';
elseif nargin == 6 % Functname, D, mv, Varrange, minmax, and
psoparams
    mv=varargin{1};
if isnan(mv)
    mv=4;
end
VR=varargin{2};
minmax=varargin{3};
P = varargin{4}; % psoparams
plotfcn='goplotpso';
elseif nargin == 7 % Functname, D, mv, Varrange, minmax, and
psoparams, plotfcn
    mv=varargin{1};
if isnan(mv)
    mv=4;
end
VR=varargin{2};
minmax=varargin{3};
P = varargin{4}; % psoparams
plotfcn = varargin{5};
elseif nargin == 8 % Functname, D, mv, Varrange, minmax, and
psoparams, plotfcn, PSOseedValue
    mv=varargin{1};
if isnan(mv)
    mv=4;

```

```

end
    VR=varargin{2};
    minmax=varargin{3};
    P = varargin{4}; % psoparams
    plotfcn = varargin{5};
    PSOseedValue = varargin{6};
else
    error('Wrong # of input arguments. ');
end

% sets up default pso params
Pdef = [100 2000 24 2 2 0.9 0.4 1500 1e-25 250 NaN 0 0];
Plen = length(P);
P = [P,Pdef(Plen+1:end)];

df = P(1);
me = P(2);
ps = P(3);
ac1 = P(4);
ac2 = P(5);
iw1 = P(6);
iw2 = P(7);
iwe = P(8);
ergrd = P(9);
ergrdep = P(10);
errgoal = P(11);
trelea = P(12);
PSOseed = P(13);

% used with trainpso, for neural net training
if strcmp(funcname,'pso_neteval')
    net = evalin('caller','net');
    Pd = evalin('caller','Pd');
    Tl = evalin('caller','Tl');
    Ai = evalin('caller','Ai');
    Q = evalin('caller','Q');
    TS = evalin('caller','TS');
end

% error checking
if ((minmax==2) & isnan(errgoal))
    error('minmax= 2, errgoal= NaN: choose an error goal or set
minmax to 0 or 1');
end

if ( (PSOseed==1) & ~exist('PSOseedValue') )
    error('PSOseed flag set but no PSOseedValue was input');
end

if exist('PSOseedValue')
    tmpsz=size(PSOseedValue);
    if D < tmpsz(2)
        error('PSOseedValue column size must be D or less');
    end
    if ps < tmpsz(1)
        error('PSOseedValue row length must be # of particles or
less');
    end
end
end

```

```

% set plotting flag
if (P(1))~=0
    plotflg=1;
else
    plotflg=0;
end

% preallocate variables for speed up
tr = ones(1,me)*NaN;

% take care of setting max velocity and position params here
if length(mv)==1
    velmaskmin = -mv*ones(ps,D); % min vel, psXD matrix
    velmaskmax = mv*ones(ps,D); % max vel
elseif length(mv)==D
    velmaskmin = repmat(forcerow(-mv),ps,1); % min vel
    velmaskmax = repmat(forcerow(mv),ps,1); % max vel
else
    error('Max vel must be either a scalar or same length as prob
dimension D');
end
posmaskmin = repmat(VR(1:D,1)',ps,1); % min pos, psXD matrix
posmaskmax = repmat(VR(1:D,2)',ps,1); % max pos
posmaskmeth = 3; % 3=bounce method (see comments below inside epoch
loop)

% PLOTTING
message = sprintf('PSO: %%g/%%g iterations, GBest = %%20.20g.\n',me);

% INITIALIZE INITIALIZE INITIALIZE INITIALIZE INITIALIZE INITIALIZE

% initialize population of particles and their velocities at time
zero,
% format of pos= (particle#, dimension)
% construct random population positions bounded by VR
pos(1:ps,1:D) = normmat(rand([ps,D]),VR',1);

if PSOseed == 1 % initial positions user input, see comments
above
    tmpsz = size(PSOseedValue);
    pos(1:tmpsz(1),1:tmpsz(2)) = PSOseedValue;
end

% construct initial random velocities between -mv,mv
vel(1:ps,1:D) = normmat(rand([ps,D]),...
[forcecol(-mv),forcecol(mv)]',1);

% initial pbest positions vals
pbest = pos;

% VECTORIZE THIS, or at least vectorize cost funct call
out = feval(funcname,pos); % returns column of cost values (1 for
each particle)
%-----

pbestval=out; % initially, pbest is same as pos

% assign initial gbest here also (gbest and gbestval)

```

```

if minmax==1
% this picks gbestval when we want to maximize the function
    [gbestval,idx1] = max(pbestval);
elseif minmax==0
% this works for straight minimization
    [gbestval,idx1] = min(pbestval);
elseif minmax==2
% this works when you know target but not direction you need to go
% good for a cost function that returns distance to target that can
be either
% negative or positive (direction info)
    [temp,idx1] = min((pbestval-ones(size(pbestval))*errgoal).^2);
    gbestval    = pbestval(idx1);
end

% preallocate a variable to keep track of gbest for all iters
bestpos      = zeros(me,D+1)*NaN;
gbest        = pbest(idx1,:); % this is gbest position
% used with trainpso, for neural net training
% assign gbest to net at each iteration, these interim assignments
% are for plotting mostly
if strcmp(funcname,'pso_neteval')
    net=setx(net,gbest);
end
%tr(1)      = gbestval;      % save for output
bestpos(1,1:D) = gbest;

% this part used for implementing Carlisle and Dozier's APSO idea
% slightly modified, this tracks the global best as the sentry
whereas
% their's chooses a different point to act as sentry
% see "Tracking Changing Extremea with Adaptive Particle Swarm
Optimizer",
% part of the WAC 2002 Proceedings, June 9-13, http://wacong.com
sentryval = gbestval;
sentry    = gbest;

if (trelea == 3)
% calculate Clerc's constriction coefficient chi to use in his form
kappa = 1; % standard val = 1, change for more or less
constriction
if ( (ac1+ac2) <=4 )
    chi = kappa;
else
    psi    = ac1 + ac2;
    chi_den = abs(2-psi-sqrt(psi^2 - 4*psi));
    chi_num = 2*kappa;
    chi     = chi_num/chi_den;
end
end

% INITIALIZE END INITIALIZE END INITIALIZE END INITIALIZE END
rstflg = 0; % for dynamic environment checking
% start PSO iterative procedures
cnt     = 0; % counter used for updating display according to df in
the options
cnt2    = 0; % counter used for the stopping subroutine based on
error convergence
iwt(1) = iw1;
for i=1:me % start epoch loop (iterations)

```

```

out          = feval(funcname, [pos;gbest]);
outbestval  = out(end,:);
out          = out(1:end-1,:);

tr(i+1)      = gbestval; % keep track of global best val
te          = i; % returns epoch number to calling program
when done
    bestpos(i,1:D+1) = [gbest,gbestval];

%assignin('base','bestpos',bestpos(i,1:D+1));
%-----
----
% this section does the plots during iterations
if plotflg==1
if (rem(i,df) == 0 ) | (i==me) | (i==1)
    fprintf(message,i,gbestval);
    cnt = cnt+1; % count how many times we display (useful for
movies)

    eval(plotfcn); % defined at top of script

end% end update display every df if statement
end% end plotflg if statement

% check for an error space that changes wrt time/iter
% threshold value that determines dynamic environment
% sees if the value of gbest changes more than some threshold value
% for the same location
    chkdyn = 1;
    rstflg = 0; % for dynamic environment checking

if chkdyn==1
    threshld = 0.05; % percent current best is allowed to change,
.05 = 5% etc
    letiter = 5; % # of iterations before checking environment,
leave at least 3 so PSO has time to converge
    outorng = abs( 1- (outbestval/gbestval) ) >= threshld;
    samepos = (max( sentry == gbest ));

if (outorng & samepos) & rem(i,letiter)==0
    rstflg=1;
% disp('New Environment: reset pbest, gbest, and vel');
%% reset pbest and pbestval if warranted
%     outpbestval = feval( funcname, [pbest] );
%     Poutorng    = abs( 1-(outpbestval./pbestval) ) > threshld;
%     pbestval    = pbestval.*~Poutorng + outpbestval.*Poutorng;
%     pbest       = pbest.*repmat(~Poutorng,1,D) +
pos.*repmat(Poutorng,1,D);

    pbest        = pos; % reset personal bests to current positions
    pbestval     = out;
    vel          = vel*10; % agitate particles a little (or a lot)

% recalculate best vals
if minmax == 1
    [gbestval,idx1] = max(pbestval);
elseif minmax==0
    [gbestval,idx1] = min(pbestval);

```

```

elseif minmax==2 % this section needs work
    [temp,idx1] = min((pbestval-
ones(size(pbestval))*errgoal).^2);
    gbestval    = pbestval(idx1);
end

    gbest  = pbest(idx1,:);

% used with trainpso, for neural net training
% assign gbest to net at each iteration, these interim assignments
% are for plotting mostly
if strcmp(funcname,'pso_neteval')
    net=setx(net,gbest);
end
end% end if outorng

    sentryval = gbestval;
    sentry    = gbest;

end% end if chkdyn

% find particles where we have new pbest, depending on minmax choice
% then find gbest and gbestval
%[size(out),size(pbestval)]
if rstflg == 0
if minmax == 0
    [tempi]          = find(pbestval>=out); % new min pbestvals
    pbestval(tempi,1) = out(tempi);      % update pbestvals
    pbest(tempi,:)  = pos(tempi,:); % update pbest positions

    [iterbestval,idx1] = min(pbestval);

if gbestval >= iterbestval
    gbestval = iterbestval;
    gbest    = pbest(idx1,:);
% used with trainpso, for neural net training
% assign gbest to net at each iteration, these interim assignments
% are for plotting mostly
if strcmp(funcname,'pso_neteval')
    net=setx(net,gbest);
end
end
elseif minmax == 1
    [tempi,dum]          = find(pbestval<=out); % new max pbestvals
    pbestval(tempi,1)   = out(tempi,1); % update pbestvals
    pbest(tempi,:)     = pos(tempi,:); % update pbest positions

    [iterbestval,idx1] = max(pbestval);
if gbestval <= iterbestval
    gbestval = iterbestval;
    gbest    = pbest(idx1,:);
% used with trainpso, for neural net training
% assign gbest to net at each iteration, these interim assignments
% are for plotting mostly
if strcmp(funcname,'pso_neteval')
    net=setx(net,gbest);
end
end
elseif minmax == 2 % this won't work as it is, fix it later
    egones          = errgoal*ones(ps,1); % vector of errgoals

```



```

elseif trelea == 1
% from Trelea's paper, parameter set 1
    vel = 0.600.*vel... % prev vel
        +1.700.*rannum1.*(pbest-pos)... %
independent
    +1.700.*rannum2.*(repmat(gbest,ps,1)-pos); % social
elseif trelea ==3
% Clerc's Type 1" PSO
    vel = chi*(vel... % prev vel
        +ac1.*rannum1.*(pbest-pos)... %
independent
    +ac2.*rannum2.*(repmat(gbest,ps,1)-pos)) ; % social
else
% common PSO algo with inertia wt
% get inertia weight, just a linear funct w.r.t. epoch parameter iwe
if i<=iwe
    iwt(i) = ((iw2-iw1)/(iwe-1))*(i-1)+iw1;
else
    iwt(i) = iw2;
end
% random number including acceleration constants
ac11 = rannum1.*ac1; % for common PSO w/inertia
ac22 = rannum2.*ac2;

    vel = iwt(i).*vel... % prev vel
        +ac11.*(pbest-pos)... %
independent
    +ac22.*(repmat(gbest,ps,1)-pos); % social
end

% limit velocities here using masking
    vel = ( (vel <= velmaskmin).*velmaskmin ) + ( (vel >
velmaskmin).*vel );
    vel = ( (vel >= velmaskmax).*velmaskmax ) + ( (vel <
velmaskmax).*vel );

% update new position (PSO algo)
    pos = pos + vel;

% position masking, limits positions to desired search space
% method: 0) no position limiting, 1) saturation at limit,
% 2) wraparound at limit , 3) bounce off limit
    minposmask_throwaway = pos <= posmaskmin; % these are psXD
matrices
    minposmask_keep = pos > posmaskmin;
    maxposmask_throwaway = pos >= posmaskmax;
    maxposmask_keep = pos < posmaskmax;

if posmaskmeth == 1
% this is the saturation method
    pos = ( minposmask_throwaway.*posmaskmin ) + (
minposmask_keep.*pos );
    pos = ( maxposmask_throwaway.*posmaskmax ) + (
maxposmask_keep.*pos );
elseif posmaskmeth == 2
% this is the wraparound method
    pos = ( minposmask_throwaway.*posmaskmax ) + (
minposmask_keep.*pos );
    pos = ( maxposmask_throwaway.*posmaskmin ) + (
maxposmask_keep.*pos );

```



```

if ((tr(i)<errgoal) & (gbestval>=errgoal)) | ((tr(i)>errgoal) ...
& (gbestval <= errgoal))
if plotflg == 1
    fprintf(message,i,gbestval);
    disp(' ');
    disp(['--> Error Goal reached, successful
termination!']);

        eval(plotfcn);
end
break
end
end% end if minmax==2
end% end ~isnan if

% % convert back to inertial frame
% pos = pos - repmat(gbestoffset,ps,1);
% pbest = pbest - repmat(gbestoffset,ps,1);
% gbest = gbest + gbestoffset;

end% end epoch loop

%% clear temp outputs
% evalin('base','clear temp_pso_out temp_te temp_tr;');

% output & return
OUT=[gbest';gbestval];
varargout{1}=[1:te];
varargout{2}=[tr(find(~isnan(tr)))];

return

```

Function template PSO

```

function [err] = genEdge(in)

disp(in)
P=in;
%P=in(:,1);
%err=P;

load out;

for temp=1:1:length(in(:,1))
%Initial value
p1 = P(temp,1);
p2 = P(temp,2);
p3 = P(temp,3);
p4 = P(temp,4);
p5 = P(temp,5);
p6 = P(temp,6);
p7 = P(temp,7);
p8 = P(temp,8);
p9 = P(temp,9);
p10 = P(temp,10);

```

```

% Generate Template
    mask1 = [ p1,p2,p3,p4,p5; p6,p7,p8,p9,p10;
              0,0,0,0,0; -p6,-p7,-p8,-p9,-p10;
              -p1,-p2,-p3,-p4,-p5 ];

mask2 = [
0,p6,p2,p1,p3;
-p6,0,p7,p8,p5;
-p2,-p7,0,p9,p4;
-p1,-p8,-p9,0,p10;
-p3,-p5,-p4,-p10,0
];

mask3 = [
-p3,-p5,-p4,-p10,0;
-p1,-p8,-p9,0,p10;
-p2,-p7,0,p9,p4;
-p6,0,p7,p8,p5;
0,p6,p2,p1,p3];

%Edge Detection and Measurement
for no=1:1:43;
    image=out(no).im; %Picture
    im=rgb2gray(image);
    [ht wd ch] = size(im);
    H = fspecial('unsharp');
    im = imfilter(im,H,'replicate');
    I=medfilt2(im,[10 10]);

    for x=4:1:(ht-4) %x=2:1:(ht-2)
        for j=4:1:(wd-4)%j=2:1:(wd-2)
            %Convolution
            %im(x,j)=FindEdge(I,x,j,mask);
            im(x,j)=FindEdge_five(I,x,j,mask1,mask2,mask3);
        end
    end

    Nim=im(4:ht-4,4:wd-4,:);

%Measurement
    s = regionprops(Nim,'Area','BoundingBox');
    [maxValue,index] = max([s.Area]);
    if(maxValue>0)
        %disp (s(index).Area)
        %figure, imshow(BWfinal), title('BoundingBox');
        out(no).Mw = s(index).BoundingBox(1,3)*0.000048;
        out(no).Mh = s(index).BoundingBox(1,4)*0.000048;
    else
        out(no).Mw = 0;
        out(no).Mh = 0;
    end

    out(no).Ew = abs(out(no).Rw-out(no).Mw);
    out(no).Eh = abs(out(no).Rh-out(no).Mh);
    %disp(out(no).Ew)
end

%Calculate Error
    sumW = sum([out.Ew]);
    sumH = sum([out.Eh]);

```

```

%meanW=mean([out.Ew]);
%meanH=mean([out.Eh]);
    err(temp,1) = sumW+sumH;
%disp(err(temp,1))
%    err=meanW+meanH;
end
%disp(err)

```

Convolution

```

function [y]=FindEdge_five(imT,i,j,kx,ka,kb)

% sobel kernel
ky=kx';

sx=0;
sy=0;
sa=0;
sb=0;
frac=0;
for xx=-2:2
for yy=-2:2
frac=double((imT(i+xx , j+yy))* kx(xx+3,yy+3));
%convolution
sx = double(sx+frac);

%kx(xx+2,yy+2)
%frac
%sx
end;
end;
frac=0;
for xx=-2:2
for yy=-2:2
frac=double((imT(i+xx , j+yy))* ky(xx+3,yy+3));
%ky(xx+2,yy+2)
sy=double(sy+frac);
%frac
%sy
end;
end;

frac=0;
for xx=-2:2
for yy=-2:2
frac=double((imT(i+xx , j+yy))* ka(xx+3,yy+3));
%ka(xx+2,yy+2)
sa=double(sa+frac);
%frac
%sa
end;
end;

frac=0;
for xx=-2:2
for yy=-2:2
frac=double((imT(i+xx , j+yy))* kb(xx+3,yy+3));
%kb(xx+2,yy+2)
sb=double(sb+frac);
%frac
%sb

```

```
end;  
end;
```

```
z=uint8(sqrt(sx^2+sy^2+sa^2+sb^2));  
%here threshold i defined threshold empirically . you can choose any  
method  
%adaptive thresholding  
if(z>=200)%70 //60  
y=0;  
else  
y=255;  
end;  
Y;
```

FindResult.m

```

clear all
close all
clc

out=struct();
    f1='Sample_All/TG/';
    f2='.jpg';
    fm='C:\Matlab R2012a\Workshop\MM.txt';
    fid=fopen(fm,'r');
    formatSpec = '%f %f';
    sizeA = [2 Inf];
    M = fscanf(fid,formatSpec,sizeA);
    fclose(fid);
%disp(M)
for i=1:1:86;
    out(i).imfile=[f1,num2str(i),f2];
    out(i).im=imread(out(i).imfile);
    out(i).Rw=M(1,i);
    out(i).Rh=M(2,i);
    out(i).Mw=0;
    out(i).Mh=0;
    out(i).Ew=0;
    out(i).Eh=0;
%disp(out(i).Rh)
end
    save out;

New method

load out;
fid = fopen('im_result/new/Result_New.txt','w+t');

%Initial value
p1 = -0.053482;
p2 = -0.081306;
p3 = 0.13671;
p4 = -0.0099407;
p5 = -0.16567;
p6 = 0.021561;
p7 = 0.010715;
p8 = 0.068298;
p9 = -0.076705;
p10 = 0.013151;

%Generate Template
mask1 = [ p1,p2,p3,p4,p5; p6,p7,p8,p9,p10;
          0,0,0,0,0; -p6,-p7,-p8,-p9,-p10;
          -p1,-p2,-p3,-p4,-p5 ];

mask2 = [
0,p6,p2,p1,p3;
-p6,0,p7,p8,p5;
-p2,-p7,0,p9,p4;
-p1,-p8,-p9,0,p10;
-p3,-p5,-p4,-p10,0 ];

mask3 = [

```

```

-p3,-p5,-p4,-p10,0;
-p1,-p8,-p9,0,p10;
-p2,-p7,0,p9,p4;
-p6,0,p7,p8,p5;
0,p6,p2,p1,p3];

fprintf(fid,'Mw,Mh,Ew,Eh\n');

tic

%Edge Detection and Measurement
for no=1:1:86;
    image=out(no).im; %Picture
    im=rgb2gray(image);
    [ht wd ch] = size(im);
    H = fspecial('unsharp');
    im = imfilter(im,H,'replicate');
    I=medfilt2(im,[10 10]);

    for x=4:1:(ht-4) %x=2:1:(ht-2)
        for j=4:1:(wd-4)%j=2:1:(wd-2)
            %Convolution
                im(x,j)=FindEdge_five(I,x,j,mask1,mask2,mask3);
        end
    end

    imI=imfill(im,'holes');
    Nim=imI(4:ht-4,4:wd-4,:);
    %Nim = imclearborder(imI,4);

    %Output Result
    %figure, imshowpair(I,Nim), title('Result');
    set(0,'DefaultFigureColor',[1 1 1]);
    figure, imshow(Nim);
    saveas(gcf,['im_result/new/','New_',num2str(no)],'png');

    %Measurement
    s = regionprops(Nim,'Area','BoundingBox');
    [maxValue,index] = max([s.Area]);
    if(maxValue>0)
        %disp (s(index).Area)

        out(no).Mw = s(index).BoundingBox(1,3)*0.000045;
        out(no).Mh = s(index).BoundingBox(1,4)*0.000045;
    else
        out(no).Mw = 0;
        out(no).Mh = 0;
    end

    out(no).Ew = abs(out(no).Rw-out(no).Mw);
    out(no).Eh = abs(out(no).Rh-out(no).Mh);
    %disp(out(no).Ew)

    %Output W,H,Error W,H
    %fprintf(fid,'%2.8f,%2.8f\n',out(no).Mw,out(no).Mh);

    fprintf(fid,'%2.8f,%2.8f,%2.8f,%2.8f\n',out(no).Mw,out(no).Mh,out(no)
    .Ew,out(no).Eh);
end
toc
fclose(fid);

```

FindResult Operation.m

```

load out;

%fid = fopen('im_result/canny/Result-canny.txt','w+t');
%fid = fopen('im_result/prewitt/Result-prewitt.txt','w+t');
fid = fopen('im_result/sobel/Result-sobel.txt','w+t');

fprintf(fid, 'Mw, Mh, Ew, Eh\n');

tic

%Edge Detection and Measurement
for no=1:1:86;
    image=out(no).im; %Picture
    im=rgb2gray(image);
    [ht wd ch] = size(im);
    H = fspecial('unsharp');
    im = imfilter(im,H,'replicate');
    I=medfilt2(im, [10 10]);

    %Edge detection
    %im = edge(I, 'canny');
    %im = edge(I, 'Prewitt');
    im = edge(I, 'sobel');

    imI=imfill(im, 'holes');
    Nim=imI(4:ht-4, 4:wd-4, :);
    %Nim = imclearborder(imI, 4);

    %Show output
    set(0, 'DefaultFigureColor', [1 1 1]);
    figure, imshow(Nim);

    %saveas(gcf, ['im_result/canny/', 'CN_', num2str(no)], 'png');
    %saveas(gcf, ['im_result/prewitt/', 'PW_', num2str(no)], 'png');
    saveas(gcf, ['im_result/sobel/', 'SB_', num2str(no)], 'png');

    %Measurement
    s = regionprops(Nim, 'Area', 'BoundingBox');
    [maxValue, index] = max([s.Area]);
    if(maxValue>0)
    %disp (s(index).Area)

        out(no).Mw = s(index).BoundingBox(1, 3)*0.000045;
        out(no).Mh = s(index).BoundingBox(1, 4)*0.000045;
    else
        out(no).Mw = 0;
        out(no).Mh = 0;
    end
    out(no).Ew = abs(out(no).Rw-out(no).Mw);
    out(no).Eh = abs(out(no).Rh-out(no).Mh);
    %disp(out(no).Ew)
    %fprintf(fid, '%2.8f ,%2.8f\n', out(no).Ew, out(no).Eh);

    fprintf(fid, '%2.8f, %2.8f, %2.8f, %2.8f\n', out(no).Mw, out(no).Mh, out(no).Ew, out(no).Eh);
end
toc
fclose(fid);

```

ประวัติผู้เขียน

ชื่อ-นามสกุล	นางสาวปารวี ศรีรัตน์
วัน เดือน ปีเกิด	16 กรกฎาคม 2532 ที่จังหวัดนราธิวาส
ที่อยู่	1305/473 คอนโดปาร์คแลนด์-วงศ์สว่างถนนกรุงเทพ-นนท์ อำเภอบางซื่อ จังหวัดกรุงเทพ 18000 โทร.0-974-245-997
ประวัติการศึกษา	2554วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมคอมพิวเตอร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี
สถานที่ทำงาน	บริษัท ไทยน้ำทิพย์ จำกัด ตำแหน่ง Business Solution Expert