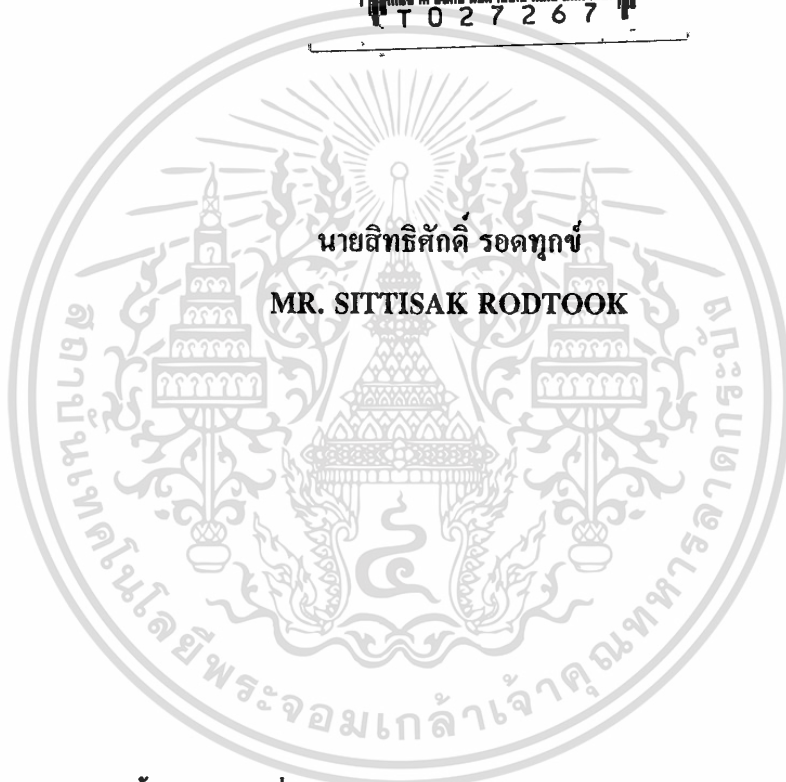


การทำธรดโซลด์ภาพสำหรับงานประยุกต์ที่ต้องการแยกภาพตัวอักษร
IMAGE THRESHOLDING FOR APPLICATIONS REQUIRING
CHARACTER IMAGE EXTRACTION



นายสิทธิศักดิ์ รอดทุกข์
MR. SITTISAK RODTOOK

วิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ

บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

พ.ศ. 2539

ISBN 974-621-759-3

ลิขสิทธิ์ของบัณฑิตวิทยาลัย สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

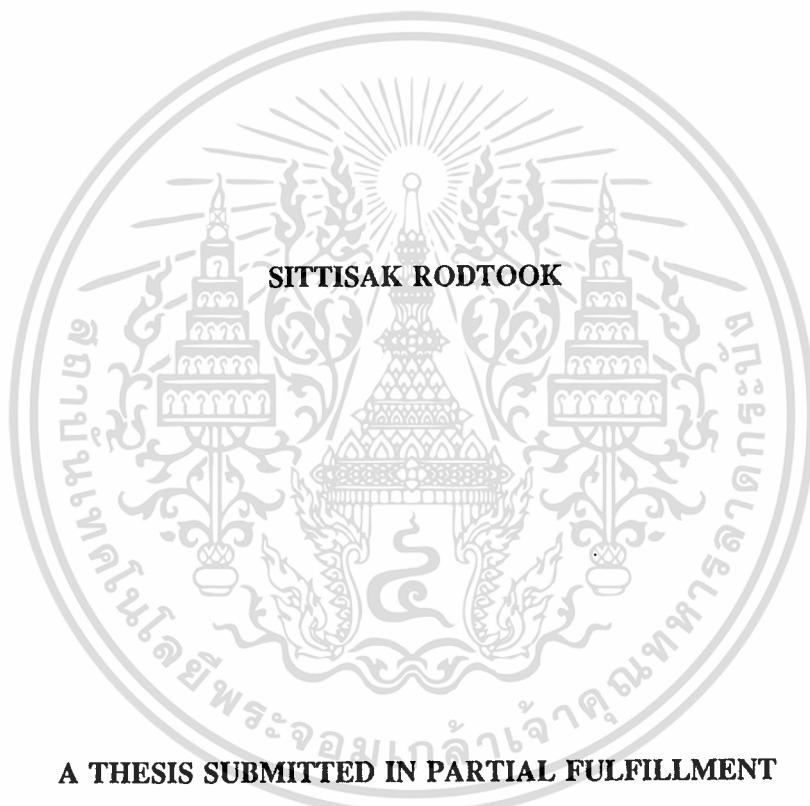
เลขหมึก.....

เลขทะเบียน..... 27267

วัน, เดือน, ปี..... 1.8 ค.ศ. 2540

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่าจะโดยทางใดก็ตาม หากมีข้อสงสัย กรุณาติดต่อขอข้อมูลเพิ่มเติม และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**IMAGE THRESHOLDING FOR APPLICATIONS REQUIRING
CHARACTER IMAGE EXTRACTION**



SITTISAK RODTOOK

**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE
MASTER OF SCIENCE PROGRAM IN COMPUTER SCIENCE
AND INFORMATION TECHNOLOGY
- SCHOOL OF GRADUATE STUDIE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

1996

ISBN 974-621-759-3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การทำเรคโครดภาพสำหรับงานประยุกต์ที่ต้องการแยกภาพตัวอักษร
นักศึกษา	นายสิทธิศักดิ์ รอดทุกซ์
อาจารย์ผู้ควบคุมวิทยานิพนธ์	ดร.บุทรพงษ์ รังสรรค์เสรี
ระดับการศึกษา	วิทยาศาสตร์มหาบัณฑิต
ภาควิชา	สาขาวิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ
พ.ศ.	คณิตศาสตร์และวิทยาการคอมพิวเตอร์
	2539

บทคัดย่อ

การแยกภาพตัวอักษรออกจากภาพเอกสารระดับเทาใด ๆ ที่มีความซับซ้อนสูงในส่วนพื้นหลังของภาพเพื่อเก็บเป็นภาพไบนารี ถือเป็นขั้นตอนการประมวลผลภาพที่สำคัญขั้นตอนหนึ่งในงานวิเคราะห์ภาพเอกสาร วิทยานิพนธ์นี้จึงนำเสนอเทคนิคสำหรับการแยกภาพตัวอักษรออกจากภาพเอกสาร ซึ่งเทคนิคเหล่านี้ขึ้นอยู่กับพื้นฐานของการเซกเมนต์ภาพโดยการทำเรคโครด และทำการประเมินเทคนิคเหล่านี้ด้วยเกณฑ์ที่กำหนด ผลที่ได้จากการประเมินและผลการทดลองจะถูกนำมาใช้ในการวิเคราะห์รูปแบบของเทคนิค ซึ่งทำให้ทราบถึงประสิทธิภาพของแต่ละเทคนิค และจากขั้นตอนการวิเคราะห์พบว่าเทคนิคที่นำเสนอ นั้นเป็นเทคนิคที่มีความยืดหยุ่นสูงพอที่ใช้สำหรับการแยกภาพตัวอักษรออกจากเอกสารที่มีความซับซ้อนในส่วนพื้นหลัง และให้ภาพไบนารีที่มีความถูกต้อง

Thesis Title	Image Thresholding for Applications Requiring Character Image Extraction
Student	Sittisak Rodtook
Thesis Advisor	Dr. Yuttapong Rangsanseri
Level of Study	Master of Science program in Computer Science and Information Technology
Department	Mathematics and Computer Science, Faculty of Science King Mongkut's Institute of Technology Ladkrabang
Year	1996

Abstract

The extraction of binary character images from grayscale document images with complex background is a common critical image processing operation particularly for document image analysis. This thesis present the character image extraction from complex document images by different techniques. Each of techniques is based on image segmentation by thresholding. We systematically evaluate new techniques with respect to aspects. Each of these techniques has been analysed by evaluation and experimental results. The analysis will also show that the new techniques are efficient and are most flexible for complex document images. Experimental results of new techniques shows that it can be correctly binarized.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี เนื่องจากได้รับความอนุเคราะห์ และความช่วยเหลือจากบุคคลหลายท่าน ขอกราบขอบพระคุณเป็นอย่างสูงในความกรุณาของ ดร. ยุทธพงษ์ รั้งสรรค์เสรี อาจารย์ที่ปรึกษา ผู้ซึ่งทุ่มเทแรงกายแรงใจในการควบคุมงานวิจัย ให้คำแนะนำในการแก้ปัญหาและข้อบกพร่องต่าง ๆ เป็นอย่างดี

ขอขอบพระคุณคณะกรรมการการสอบวิทยานิพนธ์ทุกท่าน ที่ได้ช่วยแก้ไขวิทยานิพนธ์ฉบับนี้ให้สมบูรณ์ยิ่งขึ้น

ขอขอบคุณ เพื่อน ๆ ร่วมรุ่น 4 สามัญ ภาควิชาวิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ และเพื่อนร่วมห้อง 224 สำนักวิจัยและบริการคอมพิวเตอร์ ที่ให้ความช่วยเหลือด้านต่าง ๆ และให้กำลังใจที่ดีตลอดมา

ขอขอบคุณ พี่บุญช่วย ชาติทอง และเจ้าหน้าที่คณะเทคโนโลยีสารสนเทศทุกท่าน ที่ช่วยประสานงานและให้ความสะดวกในทุก ๆ ด้าน ตลอดระยะเวลาที่ผู้เขียนศึกษา ณ สถาบันแห่งนี้

สุดท้ายนี้ ขอขอบพระคุณเป็นอย่างสูงสำหรับ คุณพ่อ คุณแม่ ครู อาจารย์ ที่อบรมสั่งสอน และให้ความช่วยเหลือและสนับสนุนด้านการศึกษามาโดยตลอด

สิทธิศักดิ์ รอดทุกข์

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญภาพประกอบ.....	VII
บทที่	
1. บทนำ.....	1
ความเป็นมาและความสำคัญของงานวิจัย.....	1
วัตถุประสงค์ของงานวิจัย.....	2
ขอบเขตของงานวิจัย.....	3
2. ทฤษฎีเบื้องต้น.....	4
การทำเรดโซลด์ภาพ.....	6
การหาขอบภาพ.....	12
การกำจัดสัญญาณรบกวนภาพ.....	26
3. อัลกอริทึมการแยกภาพตัวอักษร.....	32
อัลกอริทึมฟังก์ชันร่วม.....	34
อัลกอริทึมการทำเรดโซลด์ตามคุณลักษณะของวัตถุ.....	41
อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียน....	45
อัลกอริทึมการประเมินผลต่างของภาพ.....	50
อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพ	
ในแนวราสเตอร์สแกน.....	57
อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มจุดภาพ.....	61
อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวน	
ของระดับความเข้มจุดภาพ.....	66

สารบัญ(ต่อ)

บทที่	หน้า
อัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง.....	70
4. ผลการทดลอง.....	75
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมฟังก์ชันร่วม.....	75
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการทำเรดโซลด์ตามคุณลักษณะของวัตถุ.....	82
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามเครื่องหมาย จุดภาพของภาพลาปลาเซียน.....	85
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการประเมินผลต่างของภาพ.....	90
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับ ความเข้มของจุดภาพในแนวราสเตอร์สแกน.....	94
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ย ระดับความเข้มจุดภาพ.....	97
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ย และค่าความแปรปรวนของระดับความเข้มจุดภาพ.....	100
ผลการแยกภาพตัวอักษร โดยอัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้ม ของจุดภาพรอบข้าง.....	101
การวิเคราะห์ผลการทดลองและรูปแบบอัลกอริธึมการแยกภาพตัวอักษร.....	102
การประเมินรูปแบบอัลกอริธึมการแยกภาพตัวอักษร.....	110
5. สรุปและข้อเสนอแนะ.....	118
สรุปงานวิจัย.....	118
ข้อเสนอแนะ.....	119
เอกสารอ้างอิง.....	120
ภาคผนวก.....	122
ภาคผนวก ก เพิ่มข้อมูลกราฟฟิกรูปแบบ TIFF Version 5.0.....	123
ภาคผนวก ข การคำนวณหาค่าเรดโซลด์โดยอัลกอริธึมของ OTSU.....	130
ภาคผนวก ค การแยกภาพตัวอักษรโดยอัลกอริธึม Local Contrast.....	132
ภาคผนวก ง บทความวิชาการ.....	135

สารบัญตาราง

ตารางที่

หน้า

1. ตารางแสดงผลการประเมินรูปแบบของอัลกอริทึม โดยใช้เกณฑ์ที่กำหนด..... 112



สารบัญญภาพ

หน้า

1. แสดงขั้นตอนการแปลงภาพเอกสารเริ่มต้นที่ได้จากการสแกนให้อยู่ในรูปของภาพไบนารีโดยใช้แอฟพลีเคชันการแยกภาพตัวอักษร.....	2
2. การแทนสัญญาณอานาล็อกด้วยสัญญาณเชิงตัวเลข.....	5
3. แสดงฮิสโตแกรมระดับความเข้มของภาพ $g(x,y)$ ที่องค์ประกอบภาพส่วนที่เป็นวัตถุอ่อนข้างมีคอบนส่วนของพื้นหลังที่สว่าง.....	7
4. แสดงผลของการเชกเมนต์ภาพโดยวิธีการทำเรดโซลด์แบบครอบคลุม.....	9
5. แสดงฮิสโตแกรมระดับความเข้มของภาพ $g(x,y)$ ที่มีความไม่สม่ำเสมอของระดับความเข้มในองค์ประกอบของภาพ.....	10
6. แสดงการแบ่งข้อมูลภาพออกเป็นภาพย่อย ๆ และหาค่าเรดโซลด์ในแต่ละภาพย่อย ๆ...	10
7. แสดงผลของการเชกเมนต์ภาพโดยวิธีการทำเรดโซลด์แบบปรับค่า.....	11
8. แสดงขอบภาพชนิดต่างๆ.....	13
9. แสดงวินโดว์สำหรับคำนวณหาค่ากราเดียนท์.....	15
10. แสดงวินโดว์ขนาด 2×2 ที่ใช้สำหรับค่ากราเดียนท์.....	16
11. แสดงวินโดว์ขนาด 2×2 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของ Robert.....	16
12. แสดงวินโดว์ขนาด 3×3 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของ Sobel.....	17
13. แสดงวินโดว์ขนาด 3×3 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของ Prewitt.....	18
14. แสดงผลของการหาขอบภาพโดยตัวดำเนินการต่าง ๆ.....	19
15. แสดงความสัมพันธ์ระหว่างการหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่หนึ่งกับการหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่สอง.....	20
16. แสดงตัวดำเนินการหาขอบภาพโดยวิธีของ Laplacian.....	21
17. แสดงผลของการหาขอบภาพโดยใช้ตัวดำเนินการ Laplacian.....	22
18. แสดงตัวดำเนินการหาขอบภาพของ Marr และ Hildreth ที่ $\sigma = 1.4$	24
19. แสดงการประมาณค่า $\nabla^2 G$ โดยใช้หลักการของ DOG.....	25
20. แสดงตัวกรองสัญญาณรบกวนภาพแบบเชิงเส้น ขนาด 3×3	26
21. แสดงขั้นตอนการกรองสัญญาณรบกวน.....	28

สารบัญญภาพ (ต่อ)

	หน้า
22. แสดงผลลัพธ์ของการประมวลผลภาพเพื่อกำจัดสัญญาณรบกวนที่มีสองชนิด คือ ยูนิฟอร์ม และ เกาส์เซียน.....	29
23. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมฟังก์ชันร่วม.....	36
24. แสดงวินโดว์ขนาด 3x3 สำหรับหา G_x และ G_y	37
25. แสดงวินโดว์ขนาด 3x3 ที่ใช้สำหรับหาภาพลาปลาเซียน.....	37
26. ตัวอย่างภาพ $S(x,y)$	38
27. แสดงทิศทางในการตรวจสอบจุดภาพ.....	40
28. แผนภาพแสดงขั้นตอนการทำงานของ OAT.....	43
29. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรด โซลด์แบบปรับค่า ตามเครื่องหมายจุดภาพของภาพลาปลาเซียน.....	46
30. แสดงภาพเครื่องหมายลาปลาเซียน $L(x,y)$ ที่นิยามด้วยหลักการของ DOG.....	48
31. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการหาผลต่างของภาพ.....	52
32. แสดงตำแหน่งของจุดภาพที่ใช้ในขั้นตอนการปรับปรุงภาพ ไบนารี $b_1(x,y)$ และ ขั้นตอนการประมาณค่าระดับความเข้มที่มีคุณสมบัติของส่วนพื้นหลังที่พิกัด (x,y) ของภาพ $b_1(x,y)$	56
33. แสดงแนวทางการหาค่าเฉลี่ยระดับความเข้มจุดภาพในแนว 1 มิติ.....	58
34. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรด โซลด์แบบปรับค่า ตามค่าเฉลี่ยระดับความเข้มของจุดภาพ ในแนวราสเตอร์สแกน.....	59
35. แสดงพิกัดจุดภาพของภาพเอกสาร.....	60
36. แสดงแนวทางการหาค่าเฉลี่ยระดับความเข้มจุดภาพในแนว 2 มิติ.....	61
37. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรด โซลด์แบบปรับค่า ตามค่าเฉลี่ยระดับความเข้มของจุดภาพ.....	62
38. แสดงแนวทางของค่าผลลัพธ์ที่ได้จากฟังก์ชันการปรับค่า.....	65
39. แสดงแนวทางการหาค่าเฉลี่ยระดับความเข้มจุดภาพในแนว 2 มิติในบล็อกจตุรัส.....	66
40. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรด โซลด์แบบปรับค่า ตามค่าเฉลี่ยและค่าความแปรปรวนระดับความเข้มของจุดภาพ.....	67

สารบัญญภาพ (ต่อ)

หน้า

41. แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการประเมินค่าเฉลี่ยระดับ ความเข้มของจุดภาพในพิกัดรอบข้าง.....	71
32. แสดงพิกัด $P_i, P'_i, P_{i+1}, P'_{i+1}$ ที่ใช้สำหรับหาค่าเฉลี่ยระดับความเข้ม.....	73
43. แสดงข้อมูลภาพเริ่มต้นที่ใช้เป็นอินพุตสำหรับอัลกอริธึมแยกภาพตัวอักษร.....	77
44. แสดงภาพโมดูลกราเดียนท์.....	78
45. แสดงฮิสโตแกรมระดับความเข้มของภาพ โมดูลกราเดียนท์พร้อมทั้งค่าเรด โซลด์.....	79
46. แสดงภาพเครื่องภาพที่มี 3 สถานะ $S(x,y)$	80
47. แสดงภาพผลลัพธ์โดยอัลกอริธึมฟังก์ชันร่วม.....	81
48. แสดงฮิสโตแกรมระดับความเข้มของภาพเอกสารพร้อมทั้งค่าเรด โซลด์.....	83
49. แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรด โซลด์ตามคุณลักษณะของวัตถุ.....	84
50. แสดงภาพเครื่องหมายลาปลาเซียนที่นิยามโดยหลักการของ DOG.....	86
51. กราฟแสดงระดับความเข้มของจุดภาพในแนวนอนเปรียบเทียบกับค่าเรด โซลด์.....	87
52. กราฟแสดงระดับความเข้มของจุดภาพในแนวตั้งเปรียบเทียบกับค่าเรด โซลด์.....	88
53. แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรด โซลด์แบบปรับค่าตามเครื่องหมาย จุดภาพของภาพลาปลาเซียน.....	89
54. แสดงภาพที่ได้จากขั้นตอนการกรองหาจุดภาพพื้นหลัง.....	90
55. แสดงภาพไบนารีที่ได้รับการปรับปรุง.....	91
56. แสดงภาพพื้นหลัง.....	92
57. แสดงภาพผลลัพธ์โดยอัลกอริธึมการประเมินผลต่างของภาพ.....	93
58. กราฟแสดงระดับความเข้มของภาพเริ่มต้นเปรียบเทียบกับค่าเรด โซลด์ที่เปลี่ยน ไปตามข้อมูล.....	95
59. แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรด โซลด์แบบปรับค่าตามค่าเฉลี่ยระดับ ความเข้มของจุดภาพในแนวราสเตอร์สแกน.....	96
60. กราฟแสดงค่าเรด โซลด์ของจุดภาพพร้อมทั้งค่าระดับความเข้มของจุดภาพใน พิกัดย้อนหลัง.....	98
61. แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรด โซลด์แบบปรับค่าตามค่าเฉลี่ย ระดับความเข้มจุดภาพ.....	99

สารบัญภาพ (ต่อ)

	หน้า
62. แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรดโชลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ.....	100
63. แสดงภาพผลลัพธ์โดยอัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง.....	101
64. ฝั่งงานของ scanned image ในงาน Desktop Publishing.....	122
65. ตัวอย่าง Fixed-file format และ ตัวอย่าง โครงสร้างทั่วไปของ TIFF.....	124
66. TIFF Header.....	125
67. แสดง Image File Directory.....	126
68. แสดง 12 bytes of a TIFF tag.....	126
69. แสดงพิกัดจุดภาพที่อ้างอิงในอัลกอริธึม Local Contrast.....	132
70. แสดงภาพผลลัพธ์โดยอัลกอริธึม Local Contrast.....	133

บทที่ 1

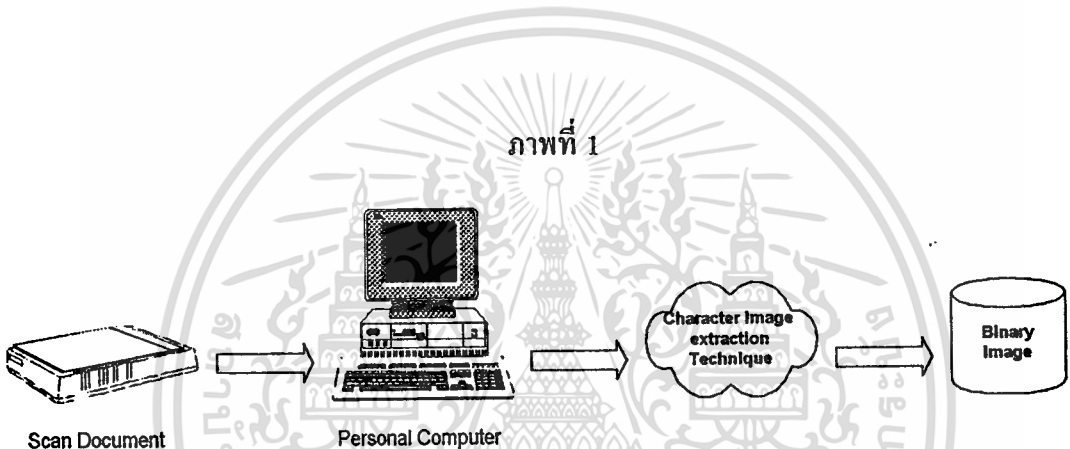
บทนำ

ความเป็นมาและความสำคัญของงานวิจัย

ในสังคมสารสนเทศทุกวันนี้เอกสารจัดได้ว่าเป็นสื่อสารสนเทศเบื้องต้นที่ใช้กันอยู่ ซึ่งสัญลักษณ์ที่ใช้แสดงข้อมูลข่าวสารในเอกสารต่างๆ นั้น จะได้แก่ ตัวอักษรลายมือเขียน หรือตัวอักษรพิมพ์ และสัญลักษณ์ทางกราฟฟิก และเนื่องด้วยประโยชน์ของเครื่องคอมพิวเตอร์, วีซีโอ, ออฟติคัลดิสก์, เครื่องโทรสาร, เครื่องพิมพ์เลเซอร์ และเครื่องสแกนเนอร์ ทำให้สัญลักษณ์ต่าง ๆ ที่กล่าวมาข้างต้นนั้นสามารถถูกสร้าง, ส่ง, เก็บไว้, แสดงผล, พิมพ์ออกมา และนำกลับเข้ามาใช้ใหม่ได้อีกครั้ง เช่นเดียวกับระบบการวิเคราะห์ภาพเอกสารที่สามารถจะจดจำสัญลักษณ์ต่าง ๆ บนเอกสาร และสามารถจัดกลุ่มของสัญลักษณ์เหล่านั้นได้ โดยซอฟต์แวร์ที่ใช้บนเครื่อง PC สำหรับการจดจำตัวอักษรและอาจจะรวมถึงสัญลักษณ์กราฟฟิกในระบบเวกเตอร์ที่ใช้ในงานทางด้านวิศวกรรม ซึ่งในปัจจุบันระบบการวิเคราะห์ภาพเอกสารนี้จะถูกนำไปใช้ในทางการศึกษาหรืองานในแวดวงอื่นๆ ทั่วไป

ในการนำระบบการวิเคราะห์ภาพเอกสารดังกล่าวข้างต้นมาใช้จะมีข้อจำกัดอยู่ที่ว่า ภาพเอกสารที่ใช้ต้องมีคุณภาพที่ดี แต่ในทางปฏิบัติพบว่าเอกสารที่มีคุณภาพดีนั้นหาได้ยาก ซึ่งปัญหาที่มักพบในเอกสารที่ด้อยคุณภาพจะ ได้แก่ ภาพเอกสารที่มีลักษณะของการเน้นข้อความด้วยสีต่าง ๆ, ภาพเอกสารที่มีรอยเลอะเทอะเปื้อน, ภาพเอกสารที่ส่วนของพื้นหลังมีรูปแบบที่ไม่แน่นอน, ภาพเอกสารที่มีการเขียนไม่ชัดเจนหรือการพิมพ์ที่ไม่มีคุณภาพ และ เงาที่เกิดขึ้นกับภาพขณะสแกนภาพเอกสารด้วยเครื่องสแกนเนอร์ ซึ่งปัญหาต่าง ๆ เหล่านี้จะเป็นปัญหาเบื้องต้นและปัญหาหลักในของระบบการวิเคราะห์เอกสาร

ดังนั้นในงานวิจัยชิ้นนี้จะนำเสนอแนวความคิดที่ใช้สำหรับการแก้ปัญหาที่เกิดขึ้นดังกล่าว โดยการการสแกนภาพเอกสารด้วยสแกนเนอร์เข้ามาเก็บในรูปแบบของข้อมูลภาพระดับความเข้มสีเทา (Grayscale) หลังจากนั้นก็ใช้เทคนิคการแยกภาพที่มีหลักการและรูปแบบที่แน่นอนสำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิกเหล่านั้นออกมาเก็บในรูปแบบของภาพไบนารี ซึ่งภาพไบนารีนั้น จุดภาพสีดำจะนิยามถึงสัญลักษณ์ คือ ตัวอักษรและภาพกราฟฟิก และจุดภาพสีขาวนิยามถึงพื้นหลัง และนำภาพไบนารีที่ได้ไปใช้ในการงานการวิเคราะห์ภาพเอกสารในขั้นตอนอื่น ๆ ต่อไป ซึ่งจะมีประสิทธิภาพมากกว่าการใช้ภาพระดับความเข้มสีเทา



แสดงขั้นตอนการแปลงภาพเอกสารเริ่มต้นที่ได้จากการสแกนให้อยู่ในรูปแบบของภาพไบนารี
โดยใช้แอปพลิเคชันการแยกภาพตัวอักษร

วัตถุประสงค์ของงานวิจัย

1. เพื่อศึกษารูปแบบการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยมุ่งเน้นถึงเทคนิคการแยกภาพตัวอักษรที่มีความยืดหยุ่นสูงเหมาะสำหรับภาพเอกสารหลากหลาย
2. เพื่อศึกษารูปแบบการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยมุ่งเน้นถึงเทคนิคการแยกภาพตัวอักษรที่มีความเหมาะสมกับลักษณะเฉพาะของภาพเอกสารนั้น
3. เพื่อสร้างภาพไบนารีที่มีความถูกต้องสูง เหมาะที่จะนำไปใช้ในการประมวลผลภาพเอกสารในส่วนอื่น ๆ ต่อไปได้อย่างมีประสิทธิภาพ เช่น การจดจำตัวอักษร หรือ การบีบอัดข้อมูลภาพ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขอบเขตของการวิจัย

เนื้อหาของงานวิจัยนี้จะมุ่งเน้นถึงการนำเสนอหลักการในรูปแบบต่าง ๆ ที่ใช้สำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิคออกจากภาพเอกสารเริ่มต้น พร้อมทั้งทำการวิเคราะห์หลักการที่ใช้และวิเคราะห์ผลที่เกิดขึ้นในแต่ละหลักการที่นำเสนอในงานวิจัยชิ้นนี้

ลำดับของการนำเสนองานวิจัย

ในงานวิจัยนี้ได้นำเสนอเทคนิคการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยรายละเอียดของการวิจัยและเนื้อหาแยกออกได้เป็น 5 บท ดังต่อไปนี้

บทที่ 1 บทนำ กล่าวถึงความเป็นมาและความสำคัญของงานวิจัย วัตถุประสงค์ของการทำวิจัย และขอบเขตของการวิจัย

บทที่ 2 ทฤษฎีเบื้องต้น ในบทนี้จะกล่าวถึงทฤษฎีพื้นฐานต่างๆ ที่นำมาใช้ในขั้นตอนการแยกภาพตัวอักษร เช่น การทำเรดโซลด์, การหาขอบภาพ, การกำจัดสัญญาณรบกวนภาพ เป็นต้น

บทที่ 3 อัลกอริทึมการแยกภาพตัวอักษร ในบทนี้จะนำเสนออัลกอริทึมในรูปแบบต่างๆ ที่ใช้สำหรับการแยกภาพตัวอักษรออกจากภาพเอกสาร ซึ่งแต่ละอัลกอริทึมก็จะมีหลักการที่แตกต่างกันออกไป

บทที่ 4 ผลการทดลอง ในบทนี้จะประกอบด้วย 2 ส่วนหลักกัน คือ ส่วนแรกจะกล่าวถึงผลที่ได้จากการทดลองแยกภาพตัวอักษรออกจากภาพเอกสารตัวอย่างที่กำหนดไว้ โดยใช้อัลกอริทึมแยกภาพตัวอักษรในรูปแบบต่างๆ ที่กล่าวไว้ในบทที่ 3 พร้อมทั้งทำการวิเคราะห์ผลการทดลองและหลักการในแต่ละอัลกอริทึมนั้น ในส่วนที่สองจะกล่าวถึงผลที่ได้จากการประเมินรูปแบบอัลกอริทึมแยกภาพตัวอักษร โดยใช้เกณฑ์ที่กำหนด

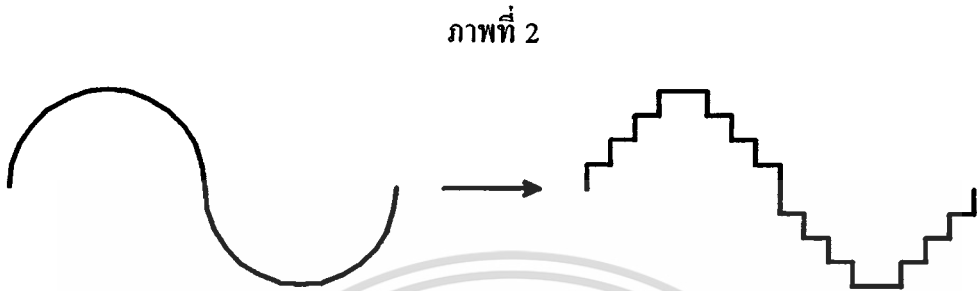
บทที่ 5 บทสรุป จะกล่าวถึงผลสรุปของการวิจัย และข้อเสนอแนะของแนวทางการพัฒนาต่อไป

บทที่ 2

ทฤษฎีเบื้องต้น

การเกิดภาพโดยใช้เลนส์เป็นวิธีการพื้นฐานและพบทั่วไปในการถ่ายภาพ แสงที่กระทบวัตถุ และสะท้อนกลับมาที่เลนส์จะถูกรวบรวมที่เลนส์และนำมาแสดงยังจุดที่สอดคล้องกันกับวัตถุ ดังนั้นกระบวนการถ่ายภาพจึงเป็นการเปลี่ยนข้อมูลของวัตถุ 3 มิติ มาเป็นข้อมูลภาพ 2 มิติ นอกจากนี้ยังสามารถบันทึกหรือประมวลผลรูปแบบของแสงที่สะท้อนเข้ามาจากวัตถุโดยการใส่เซนเซอร์ลงไป โดยที่เซนเซอร์จะให้สัญญาณไฟฟ้าในรูปอนุภาคออกมา ในขั้นตอนนี้เซนเซอร์จะทำการสแกนหรือทำการวัดผลรวมความเข้มของแสงที่จุดเล็กๆ ที่ละจุดไปเรื่อยๆ ตามแนวทางที่กำหนดไว้ หรือแนวราสเตอร์ (RASTER SCAN) ซึ่งปกติจะไล่จากซ้ายไปขวาและจากบนลงล่าง ค่าที่เซนเซอร์วัดได้นี้ถือว่าเป็นค่าความเข้มของภาพที่จุดนั้น กระบวนการถ่ายภาพทั้งหมดสามารถจัดให้เสมือนเป็นการทำคอนโวลูชันของความเข้มภาพ $I(x,y)$ กับช่วงวัดที่จัดแบ่งเป็นลำดับ และภาพผลลัพธ์จะได้จากสุ่มผลของการทำคอนโวลูชัน ดังนั้นข้อมูลภาพจึงเป็นค่าฟังก์ชัน 2 มิติของความเข้มแสง คือ $f(x,y)$ ดังนั้นค่าของฟังก์ชันที่พิกัด Spatial (x,y) จะแสดงความเข้มของภาพ ณ พิกัดนั้น แต่เนื่องจากภาพที่ได้จากการสแกนเป็นภาพแบบอนาล็อก โดยค่าความเข้ม $f(x,y)$ จะมีความต่อเนื่อง (Continuous) ดังนั้นในการที่จะนำภาพมาประมวลผลโดยคอมพิวเตอร์จึงจำเป็นต้องทำให้ภาพลักษณะต่อเนื่องกลายเป็นภาพดิจิทัลหรือภาพเชิงตัวเลขเสียก่อน โดยการทำดิจิทัล (Digitization) ซึ่งเป็นการแปลงฟังก์ชันต่อเนื่อง $f(x,y)$ ให้เป็นฟังก์ชันไม่ต่อเนื่อง $g(x,y)$ โดยการแบ่ง $f(x,y)$ ออกเป็นช่วง ๆ (Quantized) ซึ่งผลลัพธ์ที่ได้ก็คือ ค่าความเข้มของจุดภาพที่มีอยู่ L ระดับ รวมทั้งค่าพิกัด (x,y) ซึ่งค่าเหล่านี้จะเป็นจำนวนเต็มที่ไม่เป็นลบ โดยทั่วไปจุดภาพแต่ละจุดจะเป็นสมาชิกของเมตริกซ์ที่มีขนาด M แถว N คอลัมน์ ($M \times N$) เพราะฉะนั้น x, y จะมีค่าอยู่ในช่วง $(1 \leq y \leq M, 1 \leq x \leq N)$ และจำนวนช่วงระดับความเข้มของจุดภาพ L จะบ่งบอกถึงความละเอียดของภาพเชิงตัวเลข (Digital Image) โดยทั่วไปแล้ว L จะมีระดับความเข้มเท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าระดับความเข้มของจุดภาพอยู่ในช่วง $[0-255]$ โดยใช้ 8 บิต ($2^8 = 256$) สำหรับเก็บข้อมูลภาพในแต่ละจุด ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับความเข้มสูง ๆ L อาจต้องการจำนวน

บิตสำหรับเก็บข้อมูลมากกว่า 8 บิต คือ อาจเป็น 16 หรือ 24 บิต โดยค่าระดับความเข้มของจุดภาพ จะเท่ากับ 2^{16} หรือ 2^{24} ตามลำดับ



การแทนสัญญาณอนาล็อกด้วยสัญญาณเชิงตัวเลข

จากการที่นำภาพเชิงตัวเลขไปใช้ในการประมวลผลในรูปแบบต่าง ๆ มากมายนั้น ทำให้สามารถที่จะแบ่งรูปแบบการประมวลผลภาพเหล่านั้นออกเป็น 2 ระดับด้วยกัน คือ การประมวลผลภาพในระดับต่ำ (Low-level Image Processing) และการประมวลผลภาพในระดับสูง (High-level Image Processing)

การประมวลผลภาพในระดับต่ำ เป็นการประมวลผลเชิงตัวเลขเกือบทั้งหมดเพื่อหาตัวแปรต่าง ๆ มาอธิบายข้อมูลภาพ และมีจุดประสงค์ที่จะนำตัวแปรเหล่านี้ไปใช้ในการประมวลผลภาพในระดับสูงต่อไป ซึ่งการประมวลผลภาพในระดับสูง คือ การทำให้คอมพิวเตอร์รู้จักและเข้าใจภาพได้ ดังเช่น การจดจำรูปแบบของตัวอักษร สำหรับการประมวลผลภาพในระดับต่ำโดยทั่วไปจะประกอบไปด้วย การประมวลผลภาพก่อน (Image Pre-processing), การกำจัดสัญญาณรบกวน, การทำให้ภาพคมชัด, การหาขอบภาพ, การเซกเมนต์ภาพหรือการแบ่งแยกวัตถุภายในภาพ, การจำแนกภาพ, การสร้างภาพไบนารี เป็นต้น

ความแตกต่างของการประมวลผลภาพในระดับต่ำและระดับสูง คือ ข้อมูลที่นำมาใช้ในการประมวลผลภาพ ซึ่งการประมวลผลในระดับต่ำจะใช้ค่าความสว่างหรือระดับความเข้มของจุดภาพโดยตรง ส่วนการประมวลผลภาพในระดับสูงข้อมูลที่นำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ โดยสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่าง ๆ ที่อยู่ในภาพ และใช้ตัวแปรที่ได้จากประมวลผลในระดับต่ำมาอธิบายถึงสัญลักษณ์เหล่านี้ ซึ่งจะเห็นได้ว่าการประมวลผลภาพในระดับต่ำนั้นมีความสำคัญมากสำหรับที่จะทำให้คอมพิวเตอร์สามารถเข้าใจและรู้จักภาพได้ นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

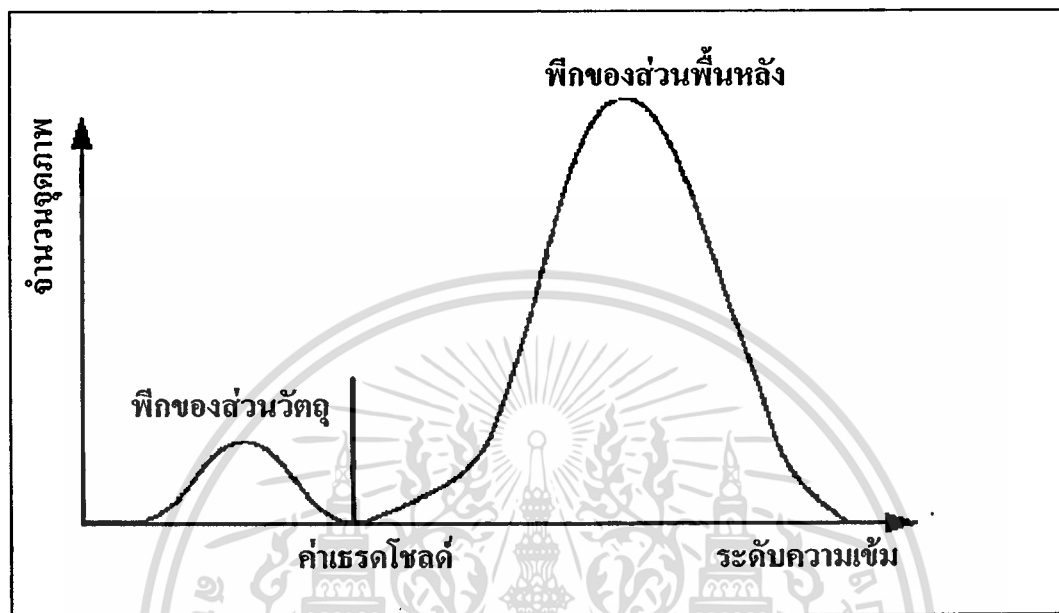
การทำแธรชโวลด์

การทำแธรชโวลด์ถือว่าเป็นเทคนิคที่สำคัญในการประมวลผลภาพในส่วนของการเซกเมนต์ภาพ ซึ่งจุดประสงค์ของการเซกเมนต์ภาพ คือ การแยกองค์ประกอบของภาพออกไปเป็นส่วนประกอบย่อย ๆ ที่มีสัมพันธ์กันทางกายภาพของภาพนั้น และส่วนประกอบที่ถูกแยกออกมานั้นอาจถูกนำไปประมวลผลในส่วนอื่นต่อไป ซึ่งการเซกเมนต์ภาพจะมีหลักการทำงานในแนวเดียวกันกับสายตาของคน คือ สามารถแยกลักษณะเด่นออกมาจากภาพที่มองเห็นได้ และเทคนิคการทำแธรชโวลด์ (Thresholding technique) ถือว่าเป็นเทคนิคในการแบ่งแยกองค์ประกอบภาพที่ง่ายเทคนิคหนึ่ง ซึ่งอาศัยหลักการที่ว่าจุดภาพที่มีคุณสมบัติของภาพอยู่ในช่วงหนึ่งจะถูกจัดเป็นกลุ่มๆหนึ่ง โดยที่ระดับความเข้มหนึ่งสามารถที่จะแบ่งกลุ่มของจุดภาพออกเป็น 2 กลุ่มอย่างชัดเจน คือ กลุ่มของวัตถุ (Object) กับกลุ่มของส่วนพื้นหลัง (Background) ดังเช่นภาพที่ (3) ซึ่งแสดงฮิสโตแกรมระดับความเข้มของภาพ $g(x,y)$ ที่มีองค์ประกอบส่วนที่เป็นวัตถุค่อนข้างมีคบนส่วนพื้นหลังที่สว่าง ดังนั้นการแยกกลุ่มทั้งสองออกจากกันอย่างชัดเจนสามารถทำได้โดยเลือกค่าแธรชโวลด์ที่มีค่าระดับความเข้มอยู่ระหว่างกลุ่มทั้งสองบนฮิสโตแกรมระดับความเข้มของภาพ และทำการตรวจสอบแต่ละจุดภาพ ถ้าค่า $g(x,y)$ น้อยกว่าค่าแธรชโวลด์ ($g(x,y) < Thr$) ถือว่าเป็นจุดภาพของวัตถุ และถ้ามากกว่าหรือเท่ากับค่าแธรชโวลด์ ($g(x,y) \geq Thr$) ถือว่าเป็นจุดภาพของส่วนพื้นหลัง ดังนั้นข้อมูลภาพ $g_{thr}(x,y)$ ที่ผ่านการทำแธรชโวลด์สามารถนิยามด้วยสมการดังต่อไปนี้

$$g_{thr}(x,y) = \begin{cases} 1 & \text{if } g(x,y) < Thr \\ 0 & \text{if } g(x,y) \geq Thr \end{cases} \quad (2.1)$$

จุดภาพที่นิยามด้วย 1 คือ จุดภาพของส่วนที่เป็นวัตถุ และจุดภาพที่นิยามด้วย 0 จะเป็นจุดภาพของพื้นหลังของภาพ

ภาพที่ 3



แสดงฮิสโตแกรมระดับความเข้มของภาพ $g(x,y)$ ที่องค์ประกอบภาพ ส่วนที่เป็นวัตถุค่อนข้างมีค่าน้อยกว่าส่วนของพื้นหลังที่สว่าง

การเซกเมนต์ภาพโดยใช้เทคนิคการทำแธรชโหดเพื่อให้ได้ภาพผลลัพธ์ที่มีความถูกต้องเหมาะสมนั้นสิ่งสำคัญที่สุด คือ ค่าแธรชโหดที่ใช้ เนื่องจากถ้าเลือกค่าแธรชโหดที่ไม่เหมาะสม ภาพผลลัพธ์ที่ได้อาจไม่ถูกต้อง ดังนั้นปัญหาของการเซกเมนต์ภาพโดยวิธีการทำแธรชโหนดนี้ก็คือ ทำอย่างไรจึงจะสามารถคำนวณหาค่าแธรชโหนดที่เหมาะสมสำหรับแต่ละภาพที่นำมาทำการเซกเมนต์ ซึ่งได้มีผู้เสนอวิธีการในการคำนวณหาค่าแธรชโหนดหลายวิธี โดยแต่ละวิธีอาจจะเหมาะสมกับภาพเฉพาะอย่างที่แตกต่างกันออกไป

ส่วนรูปแบบในการทำการแธรชโหนดกับจุดภาพนั้น ภาพที่จุดภาพของส่วนที่เป็นวัตถุกับจุดภาพของส่วนพื้นหลังในแต่ละส่วนมีระดับความเข้มที่ความสม่ำเสมอ และระดับความเข้มระหว่างทั้ง 2 ส่วนแตกต่างกันอย่างชัดเจนทั่วทั้งภาพ ซึ่งภาพลักษณะดังกล่าวนี้สามารถใช้ค่าแธรชโหนดเพียงค่าเดียวทำแธรชโหนดกับแต่ละจุดภาพทั่วทั้งภาพได้ เรียกการทำแธรชโหนดในลักษณะนี้ว่า การทำแธรชโหนดแบบครอบคลุม (Global Thresholding) แต่ถ้าภาพที่มีระดับความเข้มไม่สม่ำเสมอ เกิดขึ้นในส่วนของวัตถุหรือส่วนของพื้นหลังหรือทั้งสองส่วน ค่าโอบอลแธรชโหนดเพียงค่าเดียวไม่เหมาะสมสำหรับการทำแธรชโหนดกับแต่ละจุดภาพทั่วทั้งภาพนี้ ค่าแธรชโหนดที่ดีควรมีการปรับเปลี่ยนค่า

เอกส...
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าตามตำแหน่งของจุดภาพนั้น คือ การใช้ค่าเรดโซลด์ที่ต่างกันสำหรับจุดภาพที่ตำแหน่งต่างกัน และเรียกการทำเรดโซลด์ในลักษณะดังกล่าวว่า การทำเรดโซลด์แบบปรับค่า (Adaptive Thresholding)

การทำเรดโซลด์แบบครอบคลุม (Global Thresholding) สำหรับขั้นตอนการหาค่าเรดโซลด์ที่ครอบคลุมภาพทั้งภาพโดยอัตโนมัติ โดยทั่วไปแล้วจะมีพื้นฐานของการดำเนินการอยู่บนฮิสโตแกรมระดับความเข้มของจุดภาพ ซึ่งฮิสโตแกรมระดับความเข้มสามารถคำนวณจากจำนวนของจุดภาพในแต่ละระดับความเข้มของภาพ และทำการหาค่าเรดโซลด์ในรูปแบบต่าง ๆ ที่สามารถแบ่งฮิสโตแกรมระดับของเข้มของภาพออกเป็น 2 ส่วนได้อย่างถูกต้อง คือ ส่วนที่เป็นระดับความเข้มของวัตถุ กับส่วนที่เป็นระดับความเข้มของพื้นหลัง ตัวอย่างเช่น กรณีของภาพที่มีอัตราความแตกต่างของระดับความเข้มระหว่างส่วนที่เป็นวัตถุกับส่วนของพื้นหลังสูง และระดับความเข้มที่เกิดขึ้นในแต่ละส่วนมีความสม่ำเสมอ กรณีนี้สามารถใช้ระดับความเข้มที่มีจำนวนของจุดภาพที่ต่ำที่สุดและอยู่ระหว่างกลุ่มระดับความเข้ม (Peak) ทั้ง 2 กลุ่มบนฮิสโตแกรมเป็นค่าเรดโซลด์ดังเช่นตัวอย่างในภาพที่ 4 หรือในกรณีทั่ว ๆ ไปค่าเรดโซลด์อาจจะพิจารณาจากค่าระดับความเข้มที่สามารถแบ่งฮิสโตแกรมออกเป็น 2 กลุ่มแล้วทำให้ความแปรปรวนที่เกิดขึ้นระหว่างกลุ่มมีค่ามากที่สุด และความแปรปรวนที่เกิดขึ้นภายในกลุ่มมีค่าต่ำสุด หลังจากนั้นนำค่าเรดโซลด์ที่คำนวณหาได้ไปทำเรดโซลด์กับแต่ละจุดภาพทั่วทั้งภาพเพื่อให้ได้ภาพผลลัพธ์

การทำเรดโซลด์แบบปรับค่า (Adaptive Thresholding) กรณีที่ข้อมูลภาพภาพมีความไม่สม่ำเสมอเกิดขึ้นในส่วนของวัตถุหรือส่วนของพื้นหลัง หรือทั้งสองส่วน ซึ่งภาพในลักษณะเช่นนี้ฮิสโตแกรมระดับความเข้มของภาพจะมีลักษณะดังเช่นภาพที่ 5 การใช้ค่าเรดโซลด์แบบครอบคลุมเพียงค่าเดียวกับภาพทั้งภาพอาจทำให้ภาพผลลัพธ์ที่ได้ไม่ถูกต้อง จากปัญหาที่เกิดขึ้นสามารถแก้ไขได้โดยแบ่งข้อมูลภาพทั้งภาพออกเป็นภาพย่อย ๆ ที่แสดงดังภาพที่ 6 และแต่ละภาพย่อยก็จะมีการหาค่าเรดโซลด์ในรูปแบบที่กำหนด เพื่อให้ได้ค่าเรดโซลด์ที่เหมาะสมสำหรับภาพย่อยนั้น และใช้ค่าเรดโซลด์นั้นทำเซกเมนเตชันกับแต่ละภาพย่อย ขั้นตอนสุดท้ายนำแต่ละภาพย่อยที่ผ่านการเซกเมนเตชันมารวมกันตามพิกัดเดิม

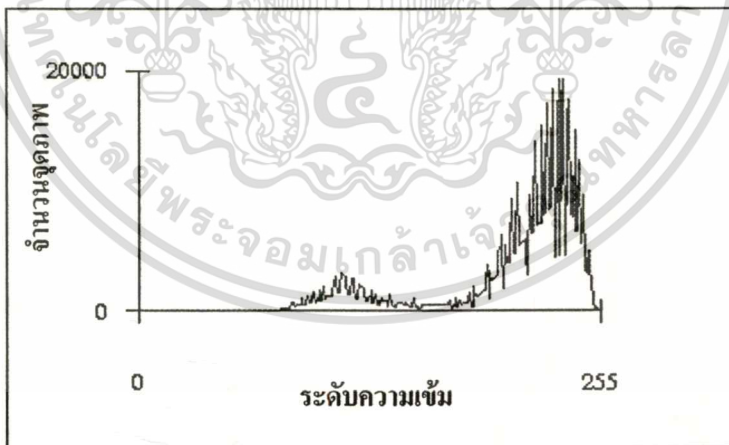
ภาพที่ 4

เพราะความยิ่งใหญ่ของ
คาต้ามะชั้น ชั้น ๓ อิมพีเรียล
เวสต์ ลาคพรวัว ไม่ว่าใครที่
ได้มาสัมผัส ต่างตื่นตาใน
ความแปลกใหม่ ทันสมัย
บนพื้นที่กว้างกว่า 1,000
ตารางเมตร ตื่นใจในความ
หลากหลายและครบครันของ
สินค้า ที่มีให้เลือกมากมาย
ตรงใจที่สุด

(ก)

เพราะความยิ่งใหญ่ของ
คาต้ามะชั้น ชั้น ๓ อิมพีเรียล
เวสต์ ลาคพรวัว ไม่ว่าใครที่
ได้มาสัมผัส ต่างตื่นตาใน
ความแปลกใหม่ ทันสมัย
บนพื้นที่กว้างกว่า 1,000
ตารางเมตร ตื่นใจในความ
หลากหลายและครบครันของ
สินค้า ที่มีให้เลือกมากมาย
ตรงใจที่สุด

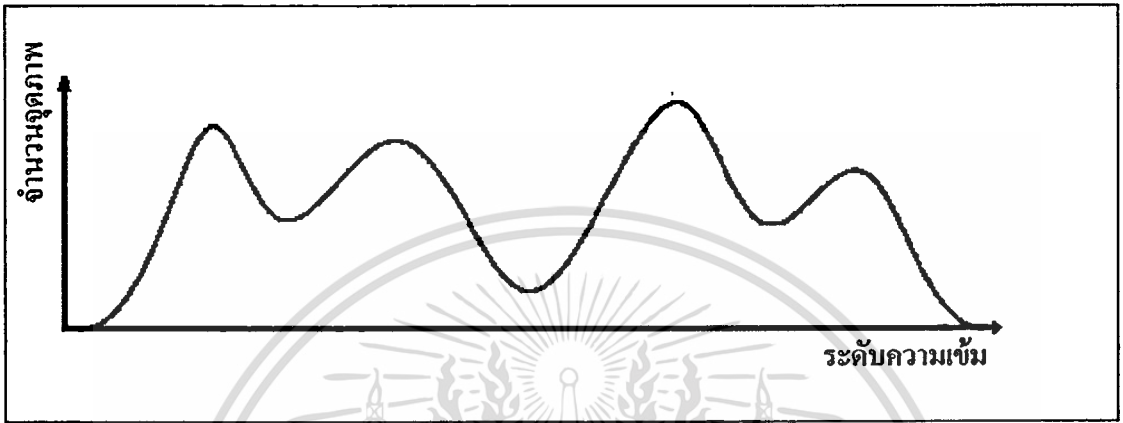
(ข)



(ค)

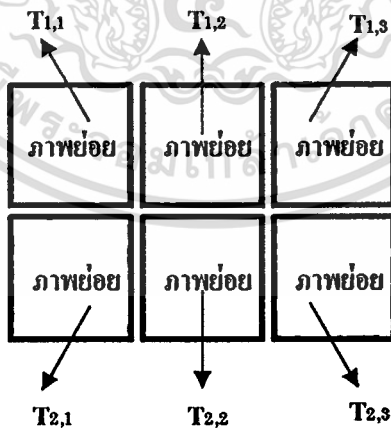
แสดงผลของการเชกเมนต์ภาพโดยวิธีการทำเรคโวลด์แบบครอบคลุม (ก) ภาพอินพุท
(ข) ภาพผลลัพธ์ (ค) ฮิสโตแกรมระดับความเข้มของภาพอินพุท

ภาพที่ 5



แสดงฮิสโตแกรมระดับความชัดของภาพ $g(x,y)$ ที่มีความไม่สม่ำเสมอของระดับความชัดในองค์ประกอบของภาพ

ภาพที่ 6



แสดงการแบ่งข้อมูลภาพออกเป็นภาพย่อย ๆ และหาค่าเรดโซลต์ในแต่ละภาพย่อย ๆ

ภาพที่ 7

จากมอนิเตอร์สีที่ขายดีติดอันดับในยุโรปและอเมริกา วันนี้...แชนทรอน พร้อมให้คุณสัมผัสแล้ว ด้วยเทคโนโลยีระดับแนวหน้า สีสันนุ่มนวลดูสบายตา อีกทั้งรักษาสิ่งแวดล้อม พร้อมช่วยถนอมสุขภาพสายตา และด้วยคุณสมบัติพิเศษมากมาย

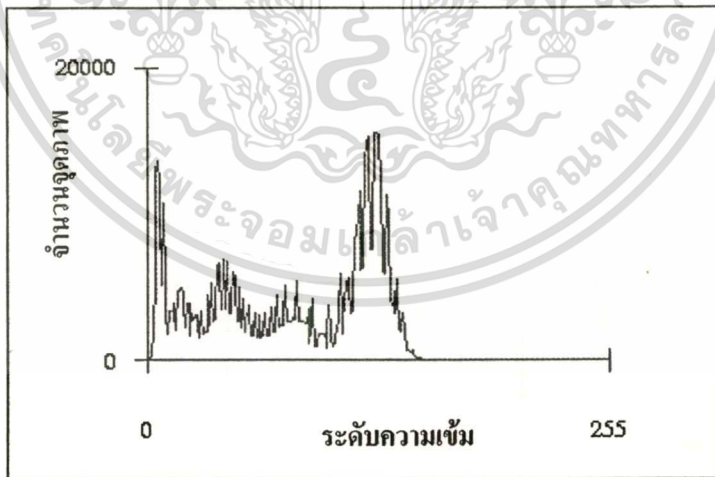
แชนทรอน จึงตอบสนองความต้องการได้ตรงจุด เพื่องานคุณภาพระดับมืออาชีพ...เช่นคุณ

จากมอนิเตอร์สีที่ขายดีติดอันดับในยุโรปและอเมริกา วันนี้...แชนทรอน พร้อมให้คุณสัมผัสแล้ว ด้วยเทคโนโลยีระดับแนวหน้า สีสันนุ่มนวลดูสบายตา อีกทั้งรักษาสิ่งแวดล้อม พร้อมช่วยถนอมสุขภาพสายตา และด้วยคุณสมบัติพิเศษมากมาย

แชนทรอน จึงตอบสนองความต้องการได้ตรงจุด เพื่องานคุณภาพระดับมืออาชีพ...เช่นคุณ

(ก)

(ข)



(ค)

แสดงผลของการเชกเมนต์ภาพโดยวิธีการทำเรดโซลต์แบบปรับค่า (ก) ภาพอินพุต

(ข) ภาพผลลัพธ์ (ค) ฮิสโตแกรมระดับความเข้มของภาพอินพุต

การหาขอบภาพ

การหาขอบภาพเป็นการหาขอบเขตของวัตถุภายในภาพ ซึ่งขอบเขตของวัตถุจะเป็นคุณสมบัติที่เด่นและมีความสำคัญมากที่จะนำไปรู้จักวัตถุนั้น ๆ โดยคอมพิวเตอร์ จะเห็นได้ว่าการหาขอบภาพในที่นี้ก็คือ การดึงคุณลักษณะโครงร่างที่เด่นของวัตถุออกมา ลักษณะเด่นของวัตถุที่เรามองเห็นโดยทั่วไปก็คือ ส่วนที่เป็นสันหรือขอบของวัตถุ เมื่อส่วนนี้มีแสงมาตกกระทบก็就会有ความสว่างมาก ซึ่งลักษณะเช่นนี้ก็เกิดขึ้นในภาพเชิงตัวเลขที่นำมาประมวลผลเช่นกัน

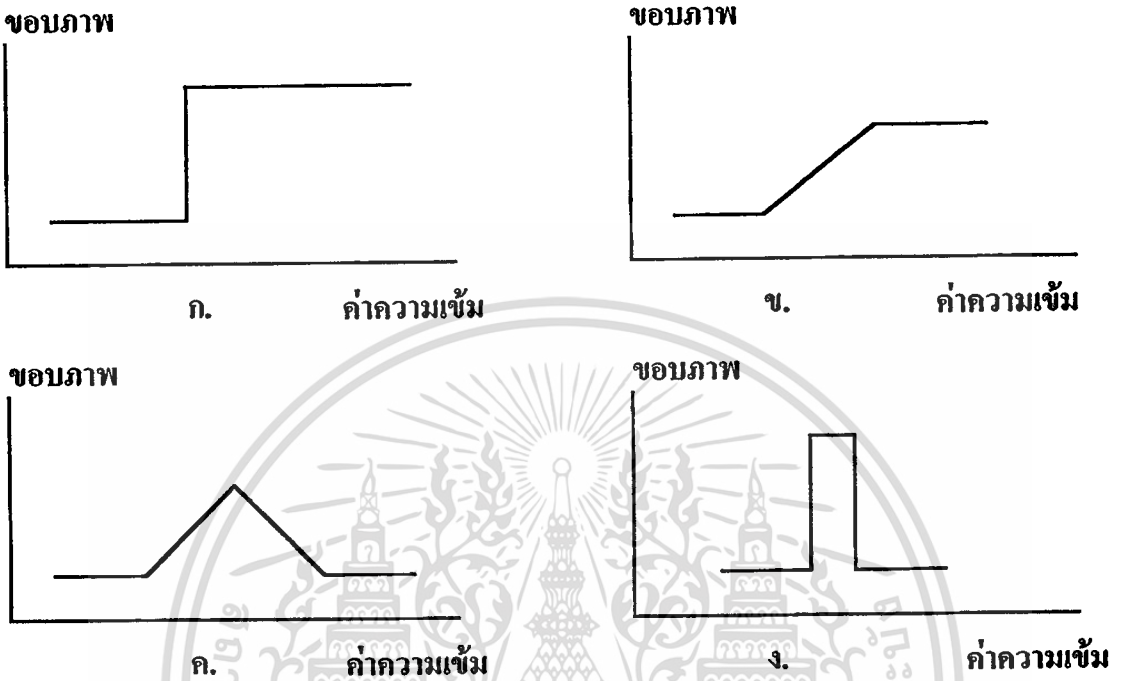
ขอบภาพเชิงตัวเลข คือ ฟังก์ชันค่าความเข้มของจุดภาพที่มีการเปลี่ยนแปลงค่าความเข้มอย่างทันทีทันใด และจุดที่เป็นขอบภาพของวัตถุจะแบ่งพื้นผิวของวัตถุหนึ่งออกจากวัตถุอีกชิ้นหนึ่ง หรืออาจจะแบ่งแยกวัตถุออกจากสีของพื้นหลัง โดยทั่วไปลักษณะของขอบภาพที่ดีจะต้องมีลักษณะดังต่อไปนี้

1. มีความบาง คือ ส่วนที่เป็นขอบภาพจะต้องมีความกว้างเพียงจุดเดียว
2. มีความต่อเนื่อง โดยจุดของขอบภาพในวัตถุเดียวกันควรมีความต่อเนื่องกัน ถ้าจุดของขอบภาพมีอยู่เพียงจุดเดียว ไม่ต่อเนื่องกับจุดใดในย่านใกล้เคียงเลยอาจเป็นขอบภาพที่ไม่สมบูรณ์ หรืออาจเป็นสัญญาณรบกวนก็ได้

รูปแบบของขอบภาพที่ปรากฏในรูปภาพโดยทั่วไป จะประกอบด้วยขอบภาพหลายชนิด ดังแสดงในภาพที่ 8 ซึ่งภาพที่ 8 (ก) เป็นขอบภาพที่มีการเปลี่ยนแปลงอย่างทันทีทันใด ขอบภาพชนิดนี้ส่วนมากจะอยู่ในภาพที่มนุษย์จำลองขึ้นมา ส่วนขอบภาพในรูปภาพทั่วไปจะเป็นขอบภาพที่มีการเปลี่ยนแปลงค่าความเข้มที่ละน้อย ดังแสดงในภาพที่ 8 (ข), ภาพที่ 8 (ค) เป็นขอบภาพที่ค่าความเข้มค่อย ๆ เพิ่มขึ้นหรือลดลงทีละน้อยในลักษณะของรูปหน้าจั่ว และภาพที่ 8 (ง) แสดงขอบภาพที่เป็นเส้น

การเปลี่ยนแปลงหรือความไม่ต่อเนื่องในฟังก์ชันค่าความเข้มของจุดภาพที่เกิดจากการส่องสว่าง หรือลักษณะทางกายภาพของตัววัตถุเอง (เช่น ลักษณะของพื้นผิว, รูปทรง) คุณสมบัตินี้จะเป็นปัจจัยพื้นฐานที่สำคัญของรูปภาพที่จะบอกถึงขอบเขตทางกายภาพของวัตถุที่อยู่ในภาพนั้น ฟังก์ชันจุดภาพ $g(x,y)$ ที่เป็นขอบภาพจะมีการเปลี่ยนแปลงจากค่าความเข้มหนึ่งไปยังอีกค่าความ

ภาพที่ 8



(ก) - (ง) แสดงขอบภาพชนิดต่างๆ

ค่าความเข้มในที่นี้มีตัววัดมากมายหลายวิธี และตัววัดความแตกต่างเหล่านี้จะเรียกว่าตัวดำเนินการหาขอบภาพซึ่งจะอธิบายแต่ละวิธีดังนี้

การหาขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่หนึ่ง

การหาขอบภาพเป็นขั้นตอนที่สำคัญในการสืบหาบริเวณที่มีการเปลี่ยนแปลงระดับความเข้มของจุดภาพเกิดขึ้น โดยการหาค่ากราเดียนท์บนข้อมูลภาพซึ่งกราเดียนท์เป็นฟังก์ชันที่วัดอัตราการเปลี่ยนแปลงระดับความเข้มที่เกิดขึ้นของจุดภาพ ซึ่งอยู่บนพื้นฐานการหาอนุพันธ์เชิงเส้นอันดับที่หนึ่ง และกราเดียนท์ $\nabla f(x, y)$ ที่เป็นฟังก์ชันต่อเนื่องในระนาบสองมิติสามารถกำหนดอยู่ในรูปของเวกเตอร์ได้ดังสมการต่อไปนี้

$$\nabla f(x, y) = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \partial f / \partial x \\ \partial f / \partial y \end{bmatrix} \quad (2.2)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาดของกราเดียนท์ (Magnitude of Gradient) จะเท่ากับ อัตราการเปลี่ยนแปลงระดับความเข้มของจุดภาพในทิศทาง G ที่กำหนด (ทิศทาง G ในที่นี้ คือ ทิศทางในแนวนอน และแนวตั้ง) ซึ่งขนาดของกราเดียนท์สามารถหาได้จากสมการดังต่อไปนี้

$$\nabla f(x, y) = \sqrt{G_x^2 + G_y^2} \quad (2.3)$$

ขนาดกราเดียนท์โดยประมาณ (Approximate Magnitude Of Gradient) สามารถหาได้จากสมการค่าสัมบูรณ์ดังต่อไปนี้

$$\nabla f(x, y) \approx |G_x| + |G_y| \quad (2.4)$$

ทิศทางของกราเดียนท์จะเท่ากับ

$$\alpha(x, y) = \tan^{-1} \left(\frac{G_x}{G_y} \right) \quad (2.5)$$

สำหรับภาพเชิงตัวเลขการหาค่ากราเดียนท์ของฟังก์ชันสองมิติ $g(x, y)$ ซึ่งเป็นฟังก์ชันไม่ต่อเนื่องที่พิกัด (x, y) สามารถกระทำในรูปของสมการดังต่อไปนี้

$$G_x = g(x+1, y) - g(x, y) \quad (2.6)$$

$$G_y = g(x, y) - g(x, y+1) \quad (2.7)$$

จากสมการที่ (2.6) และ (2.7) สามารถพิจารณาในรูปของการนำวินโดว์ขนาด 1×2 และ 2×1 ที่แสดงดังภาพที่ 9 มาคอนโวลูชันหรือการประสานงาน (Convolution) กับข้อมูลภาพเชิงตัวเลขเพื่อใช้สำหรับคำนวณหาค่า G_x และ G_y ได้ตามลำดับ

ภาพที่ 9

-1	1
----	---

(ก)

1
-1

(ข)

วินโดว์สำหรับการคำนวณหาค่ากราเดียนต์ (ก) วินโดว์ขนาด 1×2 ที่ใช้สำหรับหาค่า G_x
 (ข) วินโดว์ขนาด 2×1 ที่ใช้สำหรับหาค่า G_y

จากขั้นตอนการนำวินโดว์ขนาด 1×2 และ 2×1 ข้างต้นมาคอนโวลูชันกับข้อมูลภาพเชิงตัวเลขเพื่อหาค่ากราเดียนต์ในทิศทาง x และ y ตามลำดับนั้น พบว่าค่า G_x ที่คำนวณได้นั้นเปรียบเสมือนค่าที่เกิดจากการอินโทโปรเลตหรือการทำนายเชิงเส้น (interpolate) ที่พิกัดจุดภาพ $(x + \frac{1}{2}, y)$ โดยอ้างอิงระดับความเข้มของจุดภาพข้างเคียงที่มีพิกัดสอดคล้องกับพิกัดบนวินโดว์ ส่วนค่า G_y ที่คำนวณได้นั้นเปรียบเสมือนค่าที่เกิดจากการอินโทโปรเลตที่พิกัดจุดภาพ $(x, y + \frac{1}{2})$ โดยอ้างอิงระดับความเข้มของจุดภาพข้างเคียงที่มีพิกัดสอดคล้องกับพิกัดบนวินโดว์เช่นกัน นอกจากนี้ยังสามารถที่นำวินโดว์ขนาด 2×2 ดังภาพที่ 10 (ก) และ (ข) มาคอนโวลูชันกับภาพเชิงตัวเลขเพื่อหาค่า G_x และ G_y ตามลำดับ และขนาดของกราเดียนต์ที่เกิดขึ้นจะสูงกว่าการคอนโวลูชันด้วยวินโดว์ขนาด 1×2 และ 2×1 และจากการคอนโวลูชันข้อมูลภาพด้วยวินโดว์ขนาด 2×2 พบว่าค่ากราเดียนต์ที่คำนวณได้ทั้งในทิศทาง x และ ทิศทาง y เปรียบเสมือนค่าที่เกิดจากการอินโทโปรเลตที่พิกัดจุดภาพ $(x + \frac{1}{2}, y + \frac{1}{2})$ โดยอ้างอิงระดับความเข้มของจุดภาพรอบข้างที่มีพิกัดสอดคล้องกับพิกัดบนวินโดว์ หรืออาจจะใช้วินโดว์ขนาด 3×3 มาคำนวณหากราเดียนต์ของจุดภาพที่พิกัดศูนย์กลางของวินโดว์ก็ได้

1. การหาขอบภาพโดยวิธีของ Roberts การหาขนาดของกราเดียนต์แบบไม่ต่อเนื่อง (Magnitude of Gradient) โดยวิธีของ Roberts สามารถคำนวณจากสมการดังต่อไปนี้

$$\nabla g(x,y) = |g(x,y) - g(x+1,y+1)| + |g(x,y+1) - g(x+1,y)| \quad (2.8)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดก็ตาม อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 10

-1	1
-1	1

(ก)

1	1
-1	-1

(ข)

วินโดว์ขนาด 2x2 ที่ใช้สำหรับหาค่ากราเดียนท์

(ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

จากสมการการที่ (2.8) สามารถพิจารณาในรูปของการคอนโวลูชันข้อมูลภาพเชิงตัวเลขด้วยวินโดว์ โดยวิธีของ Robert จะใช้วินโดว์ขนาด 2x2 ดังภาพที่ 11 (ก) และ (ข) สำหรับการหากราเดียนท์ของภาพในทิศทาง x และ y ตามลำดับ

ภาพที่ 11

1	0
0	-1

(ก)

0	-1
1	0

(ข)

วินโดว์ขนาด 2x2 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของ Robert

(ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

2. การหาขอบภาพโดยวิธีของ Sobel สำหรับการหาขนาดของกราเดียนท์แบบไม่ต่อเนื่องโดยวิธีของ Sobel สามารถคำนวณจากสมการดังต่อไปนี้

$$\nabla g(x,y) = \left(\begin{array}{l} ((g(x-1,y+1)+c*g(x,y+1)+g(x+1,y+1))-(g(x-1,y-1)+c*g(x,y-1)-g(x+1,y-1))) \\ ((g(x-1,y-1)+c*g(x-1,y)+g(x-1,y+1))-((g(x+1,y-1)+c*g(x+1,y)-g(x+1,y+1))) \end{array} \right) \quad 2.9$$

โดยที่ค่าคงที่ c เท่ากับ 2

เอกสารฉบับนี้จัดทำขึ้นเพื่อใช้ในการเรียนการสอนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการการหาขนาดกราเดียนท์ข้างต้น สามารถพิจารณาในรูปของการคอนโวลูชัน ข้อมูลภาพด้วยวินโดว์ โดยวิธีของโซเบลจะใช้วินโดว์ขนาด 3×3 ดังภาพที่ 12 (ก) และ (ข) สำหรับการหากราเดียนท์ของภาพในทิศทาง x และ y ตามลำดับ

ภาพที่ 12

-1	0	1
-2	0	2
-1	0	1

(ก)

-1	-2	-1
0	0	0
1	2	1

(ข)

วินโดว์ขนาด 3×3 ที่ใช้สำหรับหาค่ากราเดียนท์โดยวิธีของ Sobel

(ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

3. การหาขอบภาพโดยวิธีของ Prewitt สมการที่ใช้ในการคำนวณหาขนาดกราเดียนท์โดยวิธีของ Prewitt จะเป็นสมการเดียวกับสมการที่ใช้หาขนาดของกราเดียนท์โดยวิธีของ Sobel แต่ต่างกันตรงที่สมการของ Prewitt ค่าคงที่ C เท่ากับ 1 ดังนั้นวินโดว์ที่ใช้ในการหากราเดียนท์ของภาพ ทั้งในทิศทาง x และ y แสดงดังภาพที่ 13

จากการหาขอบภาพโดยวิธีต่าง ๆ ที่กล่าวข้างต้น พบว่าวินโดว์ที่นำมาใช้คอนโวลูชันกับข้อมูลภาพในแต่ละวิธีจะมีค่าถ่วงน้ำหนักต่างไปจากการหากราเดียนท์ของสมการเริ่มต้น คือ สมการที่ (2.7) และ (2.8) ซึ่งมีค่าถ่วงน้ำหนักเพียงสองระดับเท่านั้น (คือ 1,-1) ส่วนการหาขอบภาพโดยวิธีของ Roberts และ Prewitt จะใช้ค่าถ่วงน้ำหนักถึง 3 ระดับ (คือ 1,0,1) และวิธีของ Sobel จะใช้ค่าถ่วงน้ำหนักถึง 5 ระดับ (คือ -2,-1,0,1,2)

ภาพที่ 13

-1	0	1
-1	0	1
-1	0	1

(ก)

1	1	1
0	0	0
-1	-1	-1

(ข)

วินโดว์ขนาด 3x3 ที่ใช้สำหรับหาค่ากราเดียนต์โดยวิธีของ Prewitt

(ก) สำหรับหาค่า G_x (ข) สำหรับหาค่า G_y

การหาขอบภาพโดยใช้ออนุพันธ์เชิงเส้นอันดับที่สอง

การหาขอบภาพโดยวิธีนี้อาศัยหลักการที่ว่า ที่ตำแหน่งของจุดภาพใดมีการเปลี่ยนแปลงระดับความเข้มอย่างทันทีทันใด และเมื่อหาอนุพันธ์เชิงเส้นอันดับที่สองที่ตำแหน่งจุดภาพนั้นจะพบว่า มีซีโรครอสซึ่งเกิดขึ้น (ดังภาพที่ 15) ซึ่งขั้นตอนการหาขอบภาพที่อยู่บนพื้นฐานของหลักการดังกล่าวมีอยู่หลายวิธีดังที่กล่าวจะต่อไปนี้

1. ตัวดำเนินการ Laplacian ตัวดำเนินการหาขอบภาพ Laplacian ของฟังก์ชันสองมิติ $g(x,y)$ ซึ่งเป็นฟังก์ชันไม่ต่อเนื่องที่พิกัด (x,y) ได้มาจากการประมาณอนุพันธ์เชิงเส้นอันดับสองของฟังก์ชันต่อเนื่องดังสมการที่ (2.10)

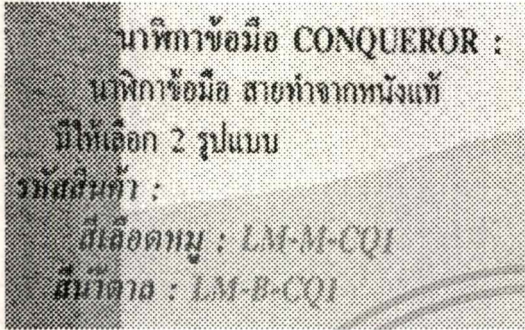
$$\nabla^2 f(x,y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.10)$$

สำหรับในกรณีของ $\nabla^2 g(x,y)$ ที่ได้จากการประมาณสมการที่ (2.10) เพื่อใช้ในการหาขอบภาพในภาพเชิงตัวเลข สามารถกำหนดได้ดังนี้

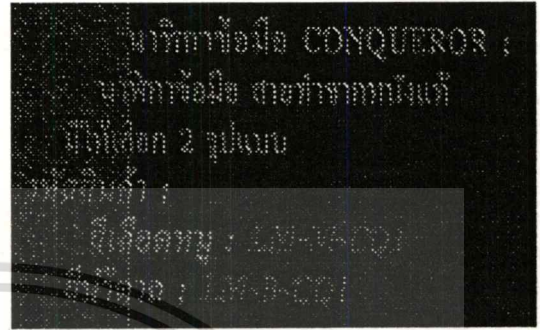
$$\nabla^2 g(x,y) = 4g(x,y) - ((g(x+1,y) + g(x-1,y) + g(x,y+1) + g(x,y-1))) \quad (2.11)$$

ไม่ว่ากรณีใดก็ตาม ทั้งสี่ข้างห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 14



(ก)



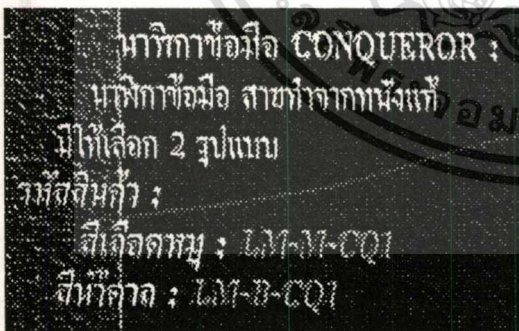
(ข)



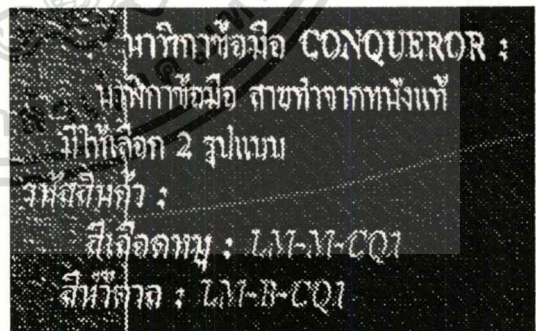
(ค)



(ง)



(จ)



(ฉ)

แสดงผลของการหาขอบภาพโดยตัวดำเนินการต่าง ๆ (ก) ภาพอินพุท

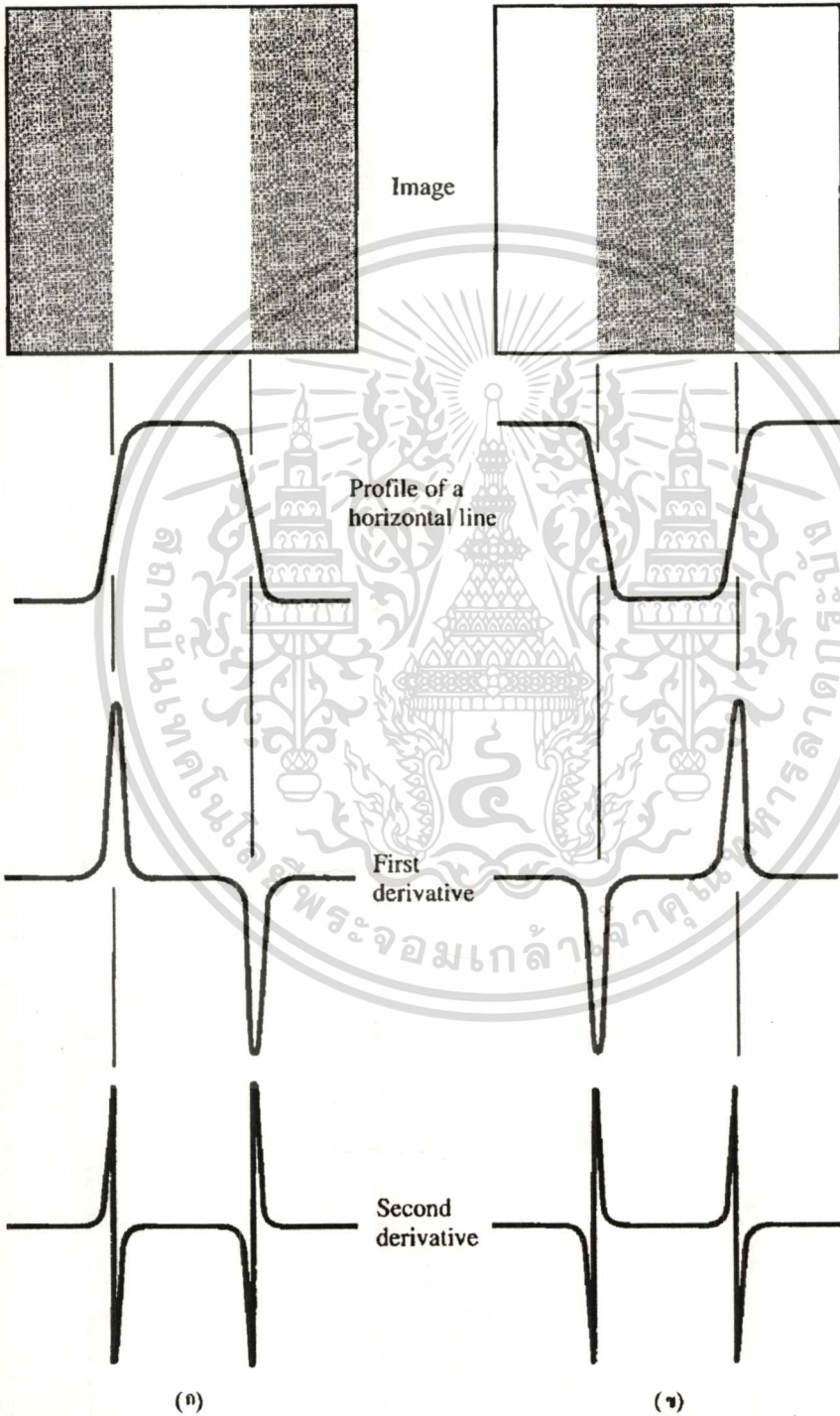
(ข) ภาพผลลัพธ์ที่ได้จากการนำวินโดว์ขนาด 1x2 และ 2x1 มาคอนโวลูชันกับภาพอินพุท

(ค) ภาพผลลัพธ์ที่ได้จากการนำวินโดว์ขนาด 2x2 มาคอนโวลูชันกับภาพอินพุท

(ง) ภาพผลลัพธ์ที่ได้จากวิธีของ Robert (จ) ภาพผลลัพธ์ที่ได้จากวิธีของ Prewitt

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ (ฉ) ภาพผลลัพธ์ที่ได้จากวิธีของ Sobel ญาติให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 15



แสดงความสัมพันธ์ระหว่างการหาขอบขอบภาพโดยใช้อนุพันธ์เชิงเส้นอันดับที่หนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้เพื่อวัตถุประสงค์ในการศึกษาเท่านั้น ไม่ควรนำเอกสารนี้ไปใช้ประโยชน์ด้านการค้า
ถ้าหากมีการนำเอกสารนี้ไปใช้โดยไม่ได้รับอนุญาตจากศูนย์บริการวิชาการ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากสมการการที่ (2.11) สามารถพิจารณาในรูปของการคอนโวลูชันข้อมูลภาพเชิงตัวเลขด้วยวินโดว์ขนาด 3×3 โดยสัมประสิทธิ์ในวินโดว์ที่ใช้จุดอยู่ที่พิกัดที่ต้องการหาขอบภาพจะต้องมีค่าเป็นบวกเสมอ ส่วนสัมประสิทธิ์ในตำแหน่งอื่นๆ อาจมีค่าเป็นบวกหรือลบก็ได้ เนื่องจากอนุพันธ์ของ Laplacian ผลรวมของสัมประสิทธิ์ทั้งหมดจะต้องเป็นศูนย์เสมอ ซึ่งในกรณีนี้ถ้าจุดภาพที่อยู่ในพื้นที่ขนาด 3×3 มีค่าเท่ากันหมด เมื่อนำมาคอนโวลูชันกับตัวดำเนินการนี้ผลลัพธ์ที่ได้ก็คือ ศูนย์ ซึ่งแสดงว่าจุดนี้ไม่ใช่ขอบภาพอย่างแน่นอน ในภาพที่ 16 แสดงวินโดว์ของ Laplacian ในรูปแบบต่าง ๆ ทั้งที่ใช้จุดภาพในบริเวณใกล้เคียง 4 จุด หรือ 8 จุด ดังรูป ซึ่งจะมีผลของขอบภาพต่างกัน

ภาพที่ 16

0	-1	0
-1	4	-1
0	-1	0

(ก)

-1	-1	-1
-1	8	-1
-1	-1	-1

(ข)

1	-2	1
-2	4	-2
1	-2	1

(ค)

(ก) - (ค) แสดงตัวดำเนินการหาขอบภาพโดยวิธีของ Laplacian

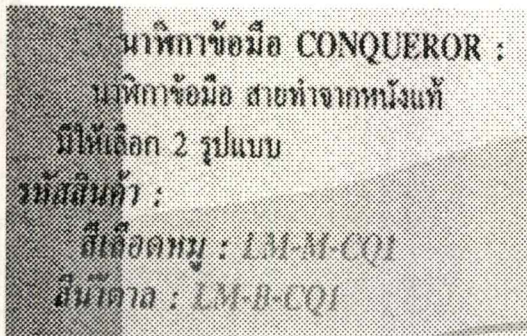
2. Marr และ Hildreth จากการหาขอบภาพโดยใช้อนุพันธ์เชิงอันดับที่สองจะพบว่า การหาขอบภาพโดยวิธีนี้จะมีควมไวต่อสัญญาณรบกวนมาก ดังนั้น Marr และ Hildreth จึงได้เสนอแนวทางที่จะแก้ปัญหานี้ โดยการนำภาพเริ่มต้นมากรองสัญญาณรบกวนก่อน หลังจากนั้นนำภาพที่ผ่านการกรองสัญญาณรบกวนมาหาขอบโดยใช้ตัวดำเนินการ Laplacian ซึ่งเรียกหลักการนี้ว่า Laplacian of Guassian (LoG)

สำหรับการหาขอบภาพโดยวิธีนี้จะใช้ตัวกรองสัญญาณรบกวนชนิดที่ให้ความถี่ต่ำผ่าน (Low Pass) ซึ่งใช้ฟังก์ชันการแจกแจงแบบเกาส์เซียน (Gaussian) ในระนาบสองมิติ ดังสมการที่ (2.12)

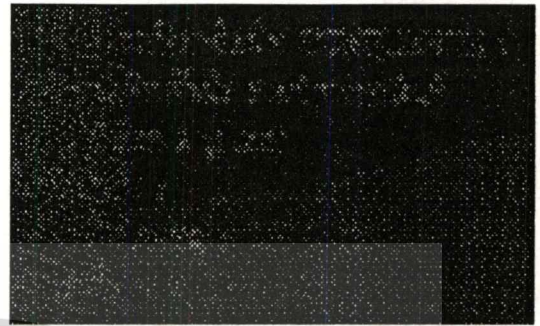
$$w(i,j) = \frac{1}{2\pi\sigma^2} e^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2.12)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 17



(ก)



(ข)



(ค)



(ง)

- แสดงผลของการหาขอบภาพโดยใช้ตัวดำเนินการ Laplacian (ก) ภาพอินพุต
- (ข) ภาพผลลัพธ์ที่ได้จากการนำวินโดว์ดังภาพ 13 (ก) มาคอนโวลูชันกับภาพอินพุต
- (ค) ภาพผลลัพธ์ที่ได้จากการนำวินโดว์ดังภาพ 13 (ข) มาคอนโวลูชันกับภาพอินพุต
- (ง) ภาพผลลัพธ์ที่ได้จากการนำวินโดว์ดังภาพ 13 (ค) มาคอนโวลูชันกับภาพอินพุต

จากหลักการ LoG ที่กล่าวข้างต้น Marr และ Hildreth จึงนำมาสร้างเป็นตัวนำเดินการหาขอบภาพโดยการหาจุดที่เป็นซีโรครอสซิงของอนุพันธ์เชิงเส้นอันดับที่สองของค่าความเข้มข้อมูลภาพที่ผ่านการกรองสัญญาณรบกวนแบบเกาส์เซียน ซึ่งสามารถเขียนเป็นสมการได้ดังนี้

$$h(x,y) = \nabla^2 [w(i,j) \otimes g(x,y)] \tag{2.13}$$

โดยที่ \otimes เป็นตัวดำเนินการคอนโวลูชัน
ดังนั้น

$$h(x,y) = [\nabla^2 w(i,j)] \otimes g(x,y) \tag{2.14}$$

ที่

$$\nabla^2 w(i,j) = \frac{1}{\pi\sigma^4} \left(\frac{i^2 + j^2 - 2\sigma^2}{\sigma^2} \right) e^{-\frac{(i^2+j^2)}{2\sigma^2}} \quad (2.15)$$

ค่า i และ j จะขึ้นอยู่กับขนาดของวินโดวที่นำมาใช้ในการหาขอบภาพ เช่น ถ้าใช้วินโดวขนาด 3×3 ค่า i และ j จะอยู่ในช่วง -1 ถึง 1 และถ้าใช้วินโดวขนาด 11×11 ค่า i และ j จะมีค่าอยู่ในช่วง -5 ถึง 5 เป็นต้น และจากสมการที่ (2.15) สามารถนำมาใช้ในการกำหนดวินโดวเพื่อใช้ในการหาขอบภาพได้ ซึ่งรูปร่างทางกราฟฟิกของวินโดวขนาด 11×11 สามารถแสดงดังภาพที่ 18

3. หลักการของ Differential Of Guassian (DOG)

จากฟังก์ชันการแจกแจงแบบเกาส์เซียนในระนาบสองมิติ (สมการที่ 2.12) ค่าผลต่างของฟังก์ชันการแจกแจงแบบเกาส์เซียนในกรณีที่ ค่า σ ต่างกัน จะเท่ากับ

$$DOG(\sigma_e, \sigma_i) = \frac{1}{2\pi\sigma_e^2} e^{-\frac{(i^2+j^2)}{2\sigma_e^2}} - \frac{1}{2\pi\sigma_i^2} e^{-\frac{(i^2+j^2)}{2\sigma_i^2}} \quad (2.16)$$

โดยที่ $\sigma_e = \sigma$ และ $\sigma_i = \sigma + \delta$ จากสมการ (2.16) สามารถเขียนใหม่ในรูปของสมการดังต่อไปนี้

$$\frac{1}{2\pi} \left[\frac{1}{\sigma^2} e^{-\frac{(i^2+j^2)}{2\sigma^2}} - \frac{1}{(\sigma+\delta)^2} e^{-\frac{(i^2+j^2)}{2(\sigma+\delta)^2}} \right] = \frac{\delta}{2\pi} \left[\frac{d(1/\sigma^2 * e^{-\frac{(i^2+j^2)}{2\sigma^2}})}{d\sigma} \right] \quad (2.17)$$

โดยค่าอนุพันธ์เชิงเส้นจะเท่ากับ $\frac{2}{\sigma^4} * \frac{(i^2 + j^2 - 2\sigma^2)}{\sigma^2} e^{-\frac{(i^2+j^2)}{2\sigma^2}}$ ดังนั้นจึงสามารถที่จะใช้

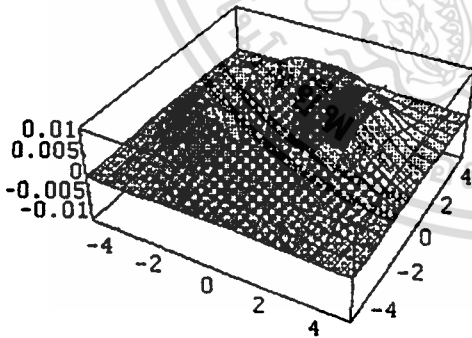
หลักการของ DOG สำหรับการประมาณค่าลาปลาเซียน ($\nabla^2 G(x,y)$) แทนการใช้หลักการของ LOG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

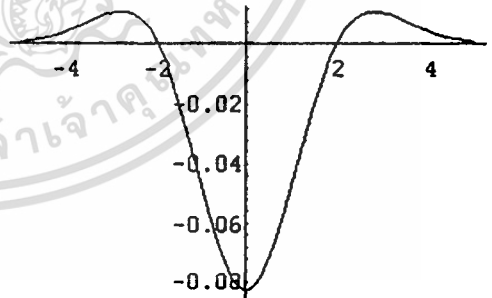
ภาพที่ 18

2.81e-06	2.25e-05	1.09e-04	3.25e-04	6.14e-04	7.57e-04	6.14e-04	3.25e-04	1.09e-04	2.25e-05	2.81e-06
2.25e-05	1.69e-04	7.57e-04	2.07e-03	3.61e-03	4.31e-03	3.61e-03	2.07e-03	7.57e-04	1.69e-04	2.25e-05
1.09e-04	7.57e-04	3.01e-03	6.96e-03	1.00e-02	1.08e-02	1.00e-02	6.96e-03	3.31e-03	7.57e-04	1.09e-04
3.25e-04	2.07e-03	6.96e-03	1.12e-02	6.37e-03	6.09e-04	6.37e-03	1.12e-02	6.96e-03	2.07e-03	3.25e-04
6.14e-01	3.61e-03	1.00e-02	6.37e-03	-2.44e-02	-4.78e-02	-2.44e-02	6.37e-03	1.00e-02	3.61e-03	6.14e-04
7.57e-04	4.31e-03	1.08e-02	6.09e-04	-4.78e-02	-8.28e-02	-4.78e-02	6.09e-04	1.08e-02	4.31e-03	7.57e-04
6.14e-04	3.61e-03	1.00e-02	6.37e-03	-2.44e-02	-4.78e-02	-2.44e-02	6.37e-03	1.00e-02	3.61e-03	6.14e-04
3.25e-04	2.07e-03	6.96e-03	1.12e-02	6.37e-03	6.09e-04	6.37e-03	1.12e-02	6.96e-03	2.07e-03	3.25e-04
1.09e-04	7.57e-04	3.31e-03	6.96e-03	1.00e-02	1.08e-02	1.00e-02	6.96e-03	3.01e-03	7.57e-04	1.09e-04
2.25e-05	1.69e-04	7.57e-04	2.07e-03	3.61e-03	4.31e-03	3.61e-03	2.07e-03	7.57e-04	1.69e-04	2.25e-05
2.81e-06	2.25e-05	1.09e-04	3.25e-04	6.14e-04	7.57e-04	6.14e-04	3.25e-04	1.09e-04	2.25e-05	2.81e-06

(ก)



(ข)



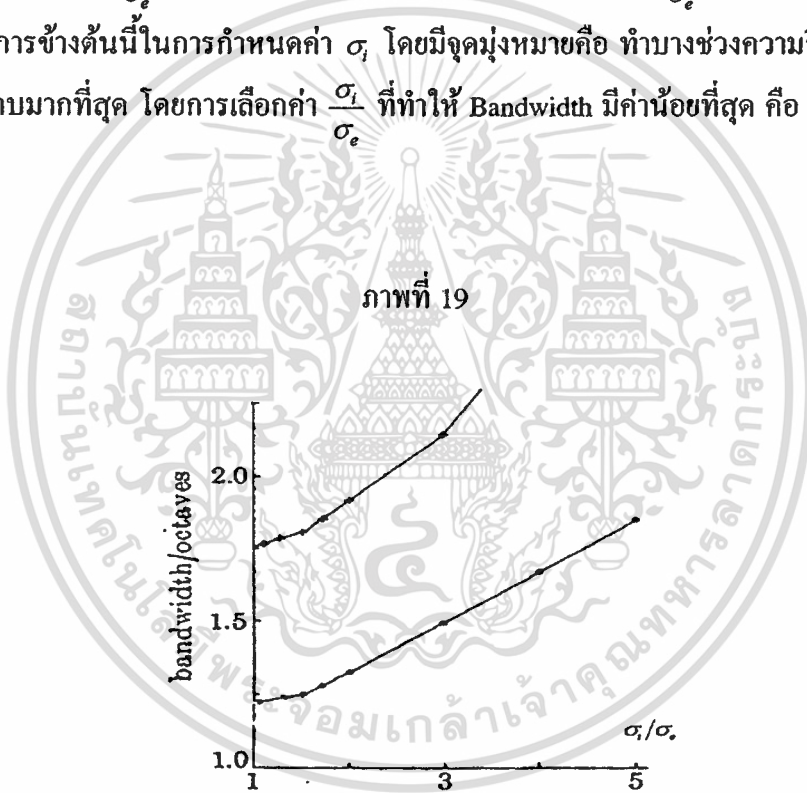
(ค)

แสดงตัวดำเนินการหาขอบภาพของ Marr และ Hildreth ที่ $\sigma = 1.4$

(ก) วินโดว์ขนาด 11x11 (ข) แสดงรูปร่างทางกราฟฟิคของรูป (ก)

(ค) เป็นรูปด้านข้างของรูป (ก)

ข้อสำคัญในการที่ใช้หลักการ DOG ในการประมาณค่า $\nabla^2 G$ คือ การกำหนดค่า σ เพราะต้องการให้การกรองสัญญาณแบบเกาส์เซียนมี Bandwidth น้อยพอสำหรับความถี่บางค่าที่ผ่านไปได้ ซึ่งที่ half sensitivity และ half power จะพบว่าค่า Bandwidth จะขึ้นอยู่กับอัตราส่วนระหว่างค่า $\frac{\sigma_i}{\sigma_e}$ (ดังภาพที่ 19 (ก)) โดยค่า Bandwidth ที่ half sensitivity จะมีอัตราการเพิ่มขึ้นเพียงเล็กน้อยบริเวณที่ $\frac{\sigma_i}{\sigma_e} \approx 1.6$ และมีการอัตราการเพิ่มอย่างรวดเร็วที่ $\frac{\sigma_i}{\sigma_e} \approx 3.0$ ดังนั้นจึงสามารถที่จะใช้หลักการข้างต้นนี้ในการกำหนดค่า σ โดยมีจุดมุ่งหมายคือ ทำบางช่วงความถี่ (Band-pass) ให้มีความแคบมากที่สุด โดยการเลือกค่า $\frac{\sigma_i}{\sigma_e}$ ที่ทำให้ Bandwidth มีค่าน้อยที่สุด คือ ค่า $\frac{\sigma_i}{\sigma_e} \leq 1.6$



กราฟแสดงความสัมพันธ์ระหว่างค่า $\frac{\sigma_i}{\sigma_e}$ กับ Bandwidth ที่ Half-sensitivity (+) และ ที่ Half-power (•) สำหรับการประมาณค่า $\nabla^2 G$ โดยใช้หลักการของ DOG

การกำจัดสัญญาณรบกวน

การกำจัดสัญญาณรบกวนภาพ ถือว่าเป็นการประมวลผลภาพในขั้นตอนการประมวลผลก่อน (Image Preprocessing) ซึ่งเป็นขั้นตอนที่ใช้ในการกำจัดสัญญาณรบกวนและสิ่งที่ไม่พึงประสงค์ที่มีขนาดเล็กออกจากภาพ ก่อนที่จะนำภาพนั้นไปประมวลผลในขั้นตอนอื่นต่อไป สำหรับสัญญาณรบกวนที่ปะปนอยู่ในภาพส่วนใหญ่จะเกิดจากตัวรับสัญญาณไฟฟ้า การขยายสัญญาณภาพ และความผิดพลาดที่เกิดขึ้นของช่องรับสัญญาณ ซึ่งการกำจัดสัญญาณรบกวนภาพนั้นมีวิธีต่าง ๆ ดังต่อไปนี้

1 การกำจัดสัญญาณรบกวนด้วยตัวกรองสัญญาณเชิงเส้น สัญญาณรบกวนโดยทั่วไปแล้วจะมีความถี่สูงกว่าสัญญาณภาพปกติ สัญญาณรบกวนประเภทนี้สามารถกำจัดออกโดยใช้ตัวกรองสัญญาณที่ยอมให้ความถี่ต่ำผ่านไปได้ ซึ่งสามารถกำหนดได้ดังภาพต่อไปนี้

ภาพที่ 20

$$w = (1/(b+2))^2 * \begin{array}{|c|c|c|} \hline 1 & b & 1 \\ \hline b & b^2 & b \\ \hline 1 & b & 1 \\ \hline \end{array}$$

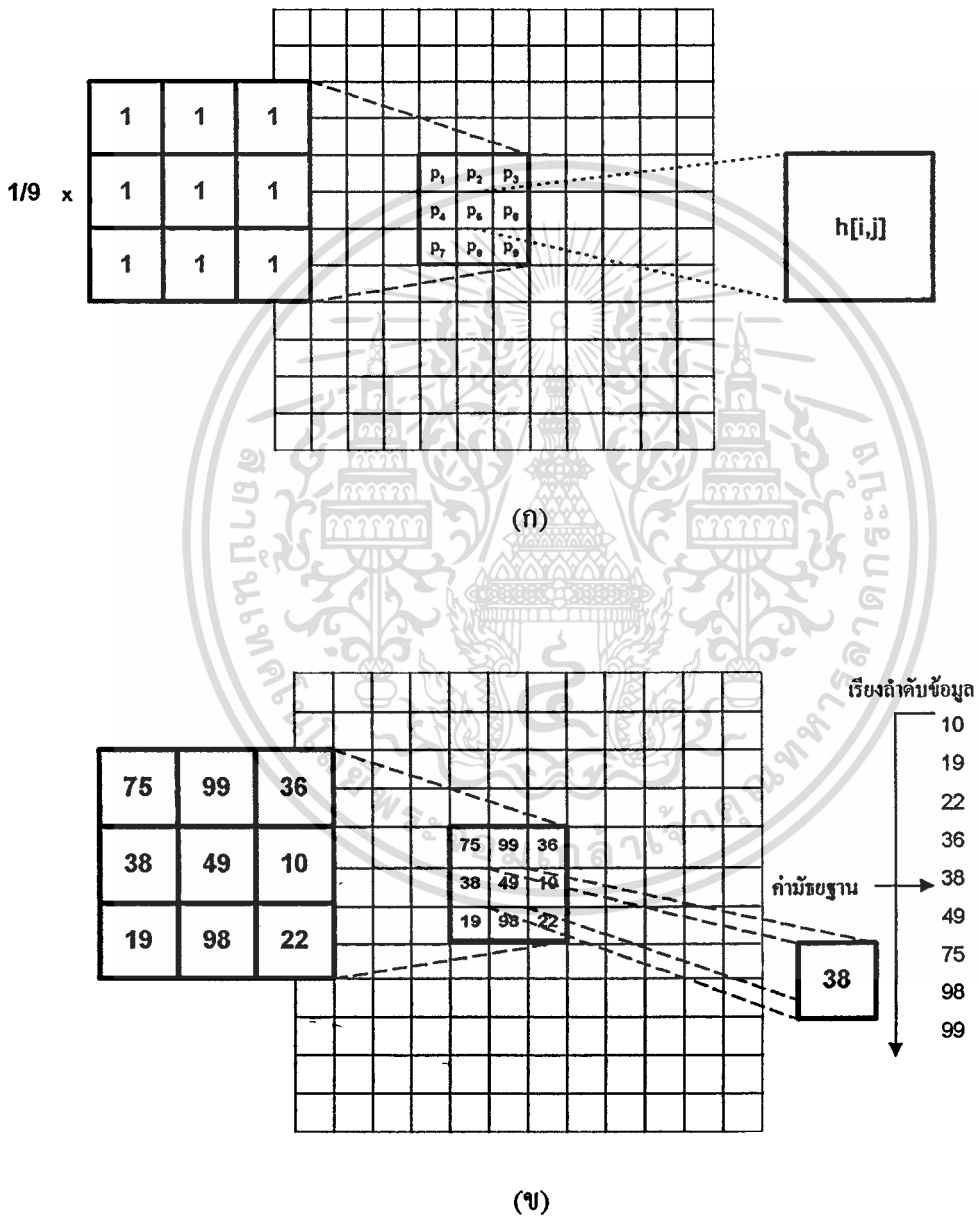
แสดงตัวกรองสัญญาณรบกวนภาพแบบเชิงเส้น ขนาด 3x3

วินโดว์ดังภาพที่ 20 เป็นวินโดว์ขนาด 3x3 โดยที่ b เป็นจำนวนจริงที่มากกว่าศูนย์ สำหรับกรณีที่ $b=1$ เมื่อนำตัวกรองสัญญาณไปคอนโวลูชันกับภาพก็เหมือนกับการทำการเฉลี่ยข้อมูลภาพภายในวินโดว์ขนาด 3x3 ที่แสดงดังภาพที่ 21 (ก) ซึ่งเรียกตัวกรองสัญญาณรูปแบบนี้ว่า ตัวกรองสัญญาณแบบเฉลี่ยข้อมูล (Mean Filter) ขนาด 3x3 และในกรณีที่ $b=2$ ตัวกรองสัญญาณนี้จะถูกเรียกว่า ตัวกรองสัญญาณแบบเกาส์เซียน ซึ่งการกำจัดสัญญาณรบกวนด้วยตัวกรองสัญญาณที่เป็นเชิงเส้นนี้จะใช้ได้ผลกับสัญญาณรบกวนที่มีความต่อเนื่อง เช่น สัญญาณรบกวนแบบเกาส์เซียน หรือสัญญาณรบกวนที่มีการแจกแจงแบบยูนิฟอร์ม

2. การกำจัดสัญญาณรบกวนด้วยวิธีที่ไม่เป็นเชิงเส้น เทคนิคอันนี้จะลดสัญญาณรบกวน และยังคงรายละเอียดของภาพได้ดีกว่าการกำจัดสัญญาณรบกวนด้วยตัวกรองสัญญาณเชิงเส้น โดยเฉพาะสัญญาณรบกวนประเภท เกลือและพริกไทย (Salt and Pepper) สำหรับการกำจัดสัญญาณรบกวนด้วยวิธีที่ไม่เป็นเชิงเส้นที่รู้จักกันดีก็คือ การกำจัดสัญญาณรบกวนด้วยตัวกรองสัญญาณค่ามัธยฐาน (Median Filter)

3. จากการวิเคราะห์รูปแบบการกำจัดสัญญาณรบกวนภาพด้วยวิธีต่าง ๆ พบว่า การกำจัดสัญญาณรบกวนด้วยวิธีที่เป็นเชิงเส้นนั้นจะพิจารณาจากผลของการถ่วงน้ำหนักค่าผลบวกของระดับความเข้มที่มีพิสัยสอดคล้องกับพิสัยคบนวินโดว์ที่นำมาคอนโวลูชันกับข้อมูลภาพ ดังเช่น การกรองสัญญาณรบกวนภาพด้วยตัวกรองสัญญาณแบบเฉลี่ยข้อมูลที่แสดงในภาพที่ 21 (ก) การกรองสัญญาณรบกวนด้วยวิธีที่เป็นเชิงเส้นแต่ต่างประเภทกัน ก็จะมีค่าถ่วงน้ำหนักที่แตกต่างกันออกไป สำหรับการกรองสัญญาณภาพด้วยวิธีที่ไม่เป็นเชิงเส้นนั้น จะไม่พิจารณาจากผลของการถ่วงน้ำหนักค่าผลบวกระดับความเข้มดังเช่นวิธีที่เป็นเชิงเส้น แต่จะพิจารณาจากตำแหน่งของค่าระดับความเข้มแทน ดังเช่น การกรองสัญญาณรบกวนภาพแบบค่ามัธยฐาน ที่แสดงในภาพที่ 21 (ข)

ภาพที่ 21



แสดงขั้นตอนการกรองสัญญาณรบกวน

(ก) ตัวกรองสัญญาณแบบเฉลี่ยข้อมูล (ข) ตัวกรองสัญญาณค่ามัธยฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้วยเทคโนโลยีที่ก้าวไกลของทวีป
วันนี้..... วัฒนธรรมาไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความดีเป็นไทย

ยูนิฟอร์ม

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้วยเทคโนโลยีที่ก้าวไกลของทวีป
วันนี้..... วัฒนธรรมาไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความดีเป็นไทย

เกาส์เขียน

(๑)

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้วยเทคโนโลยีที่ก้าวไกลของทวีป
วันนี้..... วัฒนธรรมาไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความดีเป็นไทย

ยูนิฟอร์ม

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้วยเทคโนโลยีที่ก้าวไกลของทวีป
วันนี้..... วัฒนธรรมาไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความดีเป็นไทย

เกาส์เขียน

(๑)

ภาพที่ 22

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
 อุดหนุนภูมิและความชื้น ด้วยเทคโนโลยีที่ทั่วโลกยอมรับ
 วันนี้..... วัฒนธรรมไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
 เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความเป็นไทย

(ก)

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
 อุดหนุนภูมิและความชื้น ด้วยเทคโนโลยีที่ทั่วโลกยอมรับ
 วันนี้..... วัฒนธรรมไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
 เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความเป็นไทย

(ข)

ช่วยในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
 อุดหนุนภูมิและความชื้น ด้วยเทคโนโลยีที่ทั่วโลกยอมรับ
 วันนี้..... วัฒนธรรมไทย จะไม่เหลือไว้เพียงแค่ตำนาน.....
 เพราะ.....คนไทยยังรักษาไว้ซึ่งคุณค่าของความเป็นไทย

(ค)

ชายในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้านเทคโนโลยีที่ทั่วโลกยอมรับ
วันนี้ วัฒนธรรมไทย ปลอดภัยไว้ก่อน
เพราะ คนไทยอีกรักขา ไร้สิ่งตกค้างของ ความเป็นไทย

ยูนิฟอร์ม

ชายในการบันทึกข้อมูลได้อย่างมั่นใจ แม่นยำในทุกสภาวะแวดล้อม
อุณหภูมิและความชื้น ด้านเทคโนโลยีที่ทั่วโลกยอมรับ
วันนี้ วัฒนธรรมไทย ปลอดภัยไว้ก่อน
เพราะ คนไทยอีกรักขา ไร้สิ่งตกค้างของ ความเป็นไทย

เกาส์เซียน

(ก)

แสดงผลลัพธ์ของการประมวลผลภาพเพื่อกำจัดสัญญาณรบกวนที่มีสองชนิดคือ ยูนิฟอร์ม และ เกาส์เซียน (ก) ภาพอินพุตที่ไม่มีสัญญาณรบกวน (ข) ภาพที่เพิ่มสัญญาณรบกวนยูนิฟอร์ม ด้วย ความแปรปรวนเท่ากับ 25 (ค) ภาพที่เพิ่มสัญญาณรบกวนเกาส์เซียนด้วยค่าความแปรปรวนเท่ากับ 25 (ง. - จ.) เป็นการกำจัดสัญญาณรบกวนด้วยตัวกรองเชิงเส้น โดยที่ $b = 1$ และ 2 ตามลำดับ (ฉ.) เป็นการกำจัดสัญญาณรบกวนด้วยตัวกรองสัญญาณมีขยฐานซึ่งเป็นตัวกรองที่ไม่เป็นเชิงเส้น

สรุป

การประมวลผลภาพเชิงตัวเลขจะประกอบด้วยการประมวลผลในระดับต่ำและการประมวลผลในระดับสูง การประมวลผลในระดับต่ำจะเป็นการประมวลผลกับข้อมูลภาพเชิงตัวเลขโดยตรง เพื่อที่จะคำนวณหาค่าตัวแปร ซึ่งจะนำไปใช้ในการวิเคราะห์ภาพโดยการประมวลผลในระดับสูงต่อไป

จากหัวข้อต่างๆ ที่ได้กล่าวมาแล้วนั้นเป็นการประมวลผลภาพในระดับต่ำ ซึ่งประกอบไปด้วยการแยกองค์ประกอบของภาพโดยเทคนิคการทำเรคโพลด์, การหาขอบภาพ, และการกำจัดสัญญาณรบกวนภาพ สำหรับจุดประสงค์การแยกองค์ประกอบของภาพ คือ การแยกองค์ประกอบของภาพออกไปเป็นส่วนประกอบย่อย ๆ ที่มีความสัมพันธ์กันทางกายภาพ เพื่อที่จะนำส่วนประกอบที่แยกออกมานั้น ไปใช้ประมวลผลในส่วนอื่นต่อไป ส่วนการหาขอบภาพเป็นขั้นตอนที่สำคัญของการวิเคราะห์ภาพเชิงตัวเลขเพื่อที่จะหาขอบเขตของวัตถุแต่ละชิ้นและเป็นการแบ่งแยกวัตถุด้วย ส่วนการกำจัดสัญญาณรบกวนเป็นขั้นตอนที่ใช้ในการปรับปรุงภาพ ซึ่งการปรับปรุงภาพจะส่งผลถึงขั้นตอนการวิเคราะห์ภาพ โดยคอมพิวเตอร์ด้วย คือ การปรับปรุงภาพสามารถจะช่วยให้การวิเคราะห์มีความถูกต้องยิ่งขึ้น เนื่องจากภาพที่ผ่านการปรับปรุงแล้วจะมีสัญญาณรบกวนน้อยลง แต่การปรับปรุงภาพบางครั้งอาจจะทำให้ข้อมูลที่สำคัญของภาพเกิดการสูญหายได้เหมือนกันถ้าไม่มีการพิจารณาเลือกวิธีให้เหมาะสมกับแต่ละภาพ

บทที่ 3

อัลกอริทึมสำหรับแยกภาพตัวอักษร

ในบทนี้ของงานวิจัยจะนำเสนอหลักการที่ใช้สำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิคออกจากภาพเอกสาร เพื่อนำมาเก็บในรูปแบบของภาพไบนารี ซึ่งโดยปกติแล้วเราสามารถสร้างภาพไบนารีจากภาพเอกสารเริ่มต้นได้โดยการสแกนภาพเอกสารด้วยเครื่องสแกนเนอร์แล้วกำหนดการทำงานในโหมดของภาพไบนารี ซึ่งหลักการทำงานในโหมดนี้ของซอฟต์แวร์ที่ใช้กันอยู่ทั่วไป คือ การเปรียบเทียบระดับความเข้มของแต่ละจุดภาพกับค่าขีดโหดซึ่งเป็นค่าคงที่สำหรับตัดสินว่าจุดภาพนั้นจะเป็นสีขาว (คือ พื้นหลัง) หรือสีดำ (คือ ตัวอักษร) ซึ่งภาพไบนารีที่ได้จากวิธีดังกล่าวจะมีคุณภาพที่ดีก็ต่อเมื่อภาพเอกสารเริ่มต้นมีคุณภาพที่ดีเท่านั้น หรืออาจจะสร้างภาพไบนารีโดยการสแกนภาพเอกสารด้วยเครื่องสแกนเนอร์แล้วกำหนดการทำงานในโหมดของภาพระดับความเข้มสีเทา แล้วนำข้อมูลภาพที่ได้ไปประมวลผลโดยซอฟต์แวร์ที่ชื่อว่า โฟโตสไตลเลอร์เวอร์ชัน 2.0 (Photo Styler Version 2.0) ที่เลือกการทำงานในโหมดของการทำขีดโหด ซึ่งการทำงานในโหมดนี้จะให้ผู้ใช้เป็นผู้กำหนดค่าขีดโหดให้กับภาพ และใช้ผลของการทำขีดโหดมาใช้เป็นแนวทางสำหรับการปรับค่าขีดโหดให้มีความเหมาะสมกับภาพ แต่สำหรับหลักการที่ใช้สำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารที่จะนำเสนอในบทนี้ จะเป็นการคำนวณหาข้อมูลของจุดภาพในรูปแบบต่าง ๆ เพื่อนำค่าข้อมูลนั้นมาใช้พิจารณาตัดสินคุณสมบัติของจุดภาพโดยอัตโนมัติ ซึ่งข้อมูลของจุดภาพส่วนใหญ่จะอยู่ในรูปของค่าสถิติ ซึ่งบทนี้ได้นำเสนออัลกอริทึมที่ใช้สำหรับการแยกภาพตัวอักษรทั้งหมด 8 อัลกอริทึม ดังนี้

1. อัลกอริทึมฟังก์ชันร่วม
2. อัลกอริทึมการทำขีดโหดตามคุณลักษณะของวัตถุ
3. อัลกอริทึมการทำขีดโหดแบบปรับค่าตามหมายจุดภาพของภาพลาปลาเซียน
4. อัลกอริทึมการประเมินผลต่างภาพ
5. อัลกอริทึมการทำขีดโหดแบบปรับค่าตามค่าเฉลี่ยระดับความเข้มจุดภาพในแนว

ราสเตอร์สแกน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มจุดภาพ
7. อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ
8. อัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง

อัลกอริทึมแต่ละอัลกอริทึมที่นำเสนอมีหลักการที่ใช้สำหรับการแยกภาพตัวอักษรที่แตกต่างกันออกไป แต่อัลกอริทึมเหล่านี้จะอยู่บนพื้นฐานของการทำเรดโซลด์กับจุดภาพ และบางรูปแบบอัลกอริทึมมีการนำหลักการทางอิมเมจโปรเซสซึ่งมาประยุกต์ใช้ หรือบางรูปแบบอัลกอริทึมมีการนำค่าพารามิเตอร์ เช่น ค่าผลต่างของระดับความเข้มระหว่างพื้นหลังกับตัวอักษร หรือค่าความกว้างของเส้นตัวอักษร มาร่วมใช้ในการพิจารณาคูสมบัติของจุดภาพ ดังรายละเอียดที่จะได้กล่าวถึงต่อไป สำหรับข้อมูลภาพที่จะใช้ในการประมวลผลกับอัลกอริทึมเหล่านี้จะอยู่ในรูปของภาพระดับความเข้มสีเทา 256 ระดับ และภาพผลลัพธ์ที่ได้จะอยู่ในรูปของภาพไบนารี โดยจุดภาพสีดำจะนิยามถึงจุดภาพของตัวอักษร และจุดภาพสีขาวจะนิยามถึงจุดภาพของพื้นหลัง

การแยกภาพตัวอักษรโดยอัลกอริทึมฟังก์ชันร่วม

อัลกอริทึมฟังก์ชันร่วม (Integrated Function Algorithm)[1] เป็นอัลกอริทึมรูปแบบหนึ่งที่มีการนำเอาโอเปอเรเตอร์พื้นฐานทางอิมเมจโปรเซสซึ่ง เช่น การหาขอบภาพ การนิยามภาพลาปลาเซียน และการคำนวณหาค่าเรดโซลด์ของภาพ มาประยุกต์ใช้ร่วมกันในขั้นตอนการแยกภาพตัวอักษรออกจากภาพเอกสารเพื่อเก็บอยู่ในรูปของภาพไบนารี และอัลกอริทึมฟังก์ชันร่วมนี้ได้พัฒนามาจากอัลกอริทึมการแยกภาพตัวอักษรของ J.M. White, G.D. Rohrer [5]

หลักการที่นำเสนอ

หลักการที่นำเสนอสำหรับการแยกภาพตัวอักษรในอัลกอริทึมรูปแบบนี้เริ่มจาก การนิยามขอบภาพโดยใช้โอเปอเรเตอร์ที่อยู่บนพื้นฐานของการหาอนุพันธ์เชิงเส้นอันดับที่หนึ่ง ผลที่ได้จะถูกนำมาใช้คำนวณหาค่าเรดโซลด์แบบครอบคลุม (Global Threshold) เพื่อนำไปใช้ในขั้นตอนการนิยามขอบของตัวอักษร หลังจากนั้นก็จะทำการนิยามตำแหน่งของจุดภาพในบริเวณที่เป็นขอบ แต่จุดภาพจะถูกพิจารณาว่าเป็นขอบทางด้านมืดหรือขอบทางด้านสว่างโดยใช้โอเปอเรเตอร์ที่อยู่

บนพื้นฐานของการหาอนุพันธ์เชิงเส้นอันดับที่สอง และขั้นตอนสุดท้ายทำการนิยามภาพผลลัพธ์ คือ ภาพไบนารี โดยจุดภาพที่ไม่ได้อยู่ในขอบเขตของขอบจะถูกนิยามให้เป็นจุดภาพของพื้นหลัง ส่วนจุดภาพที่อยู่ภายในขอบเขตของขอบจะถูกตัดสินโดยใช้ความกว้างของเส้นตัวอักษรมาเป็นเกณฑ์สำคัญในการตัดสินใจ ถ้าระยะห่างระหว่างขอบเขตของขอบสัมพันธ์กับความกว้างของเส้นตัวอักษร จุดภาพที่อยู่ภายในขอบเขตของขอบจะถูกนิยามให้เป็นจุดภาพของตัวอักษร มิฉะนั้นแล้วจุดภาพนั้นจะถูกนิยามให้เป็นจุดภาพของพื้นหลัง สำหรับการทำงานของอัลกอริทึมด้วยหลักการดังกล่าวสามารถแสดงในรูปของแผนภาพการทำงานดังภาพที่ 23

วิธีการที่เสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการประมวลผลออกเป็น ขั้นตอนย่อย ๆ ทั้งหมด 5 ขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 การนิยามขอบภาพ

คำนวณหากราเดียนต์ของภาพเอกสารเริ่มต้น เพื่อทำการนิยามจุดภาพที่อยู่ในบริเวณที่เป็นขอบ ซึ่งอัลกอริทึมนี้ได้ใช้วิธีของโซเบล (Sobel Operator) ในการหากราเดียนต์ของภาพ โดยการนำวินโดว์ขนาด 3*3 ที่แสดงดังภาพที่ 24 (ก) และ (ข) มาคอนโวลูชันกับข้อมูลภาพเชิงตัวเลข เพื่อการหากราเดียนต์ของภาพในแนวนอนและแนวตั้งตามลำดับ

ขนาดกราเดียนต์ของภาพโดยประมาณ (Approximated Magnitude of Gradient) จะเท่ากับ

$$\nabla g(x,y) = |G_x| + |G_y| \quad (3.1)$$

หลังจากหากราเดียนต์ของภาพแล้ว จะได้ภาพใหม่ que แสดงลักษณะความเป็นขอบ ซึ่งเป็นทั้งขอบของตัวอักษรและขอบของสิ่งไม่พึงประสงค์ที่อยู่ในส่วนของพื้นหลัง

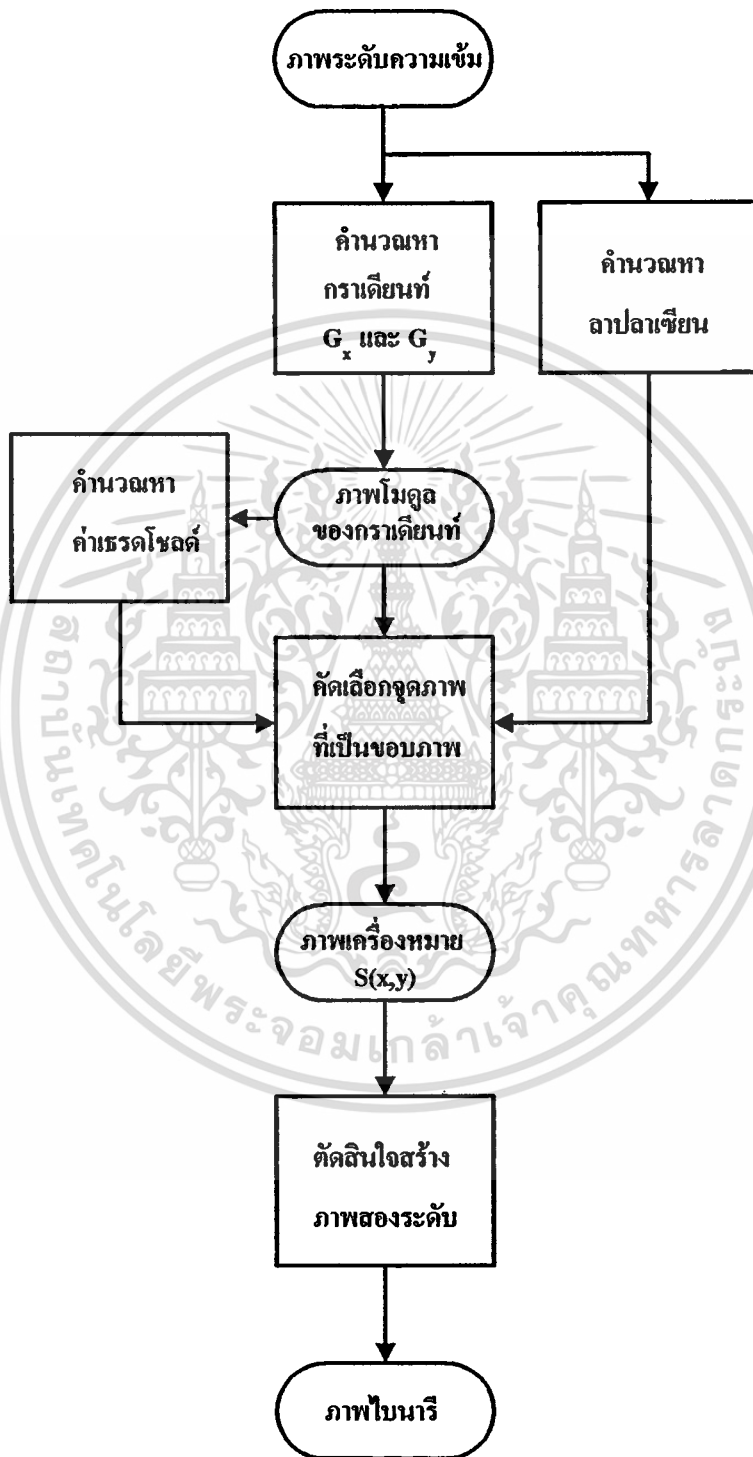
ขั้นตอนที่ 2 การคำนวณหาแตรด โซลด์แบบครอบคลุมจากภาพโมดูลของกราเดียนต์

นำภาพที่ผ่านขั้นตอนการหากราเดียนต์มาคำนวณหาค่าแตรด โซลด์ เพื่อนำค่าแตรด โซลด์ นั้นมาใช้ในการสืบหาเฉพาะจุดภาพที่เป็นขอบของตัวอักษร ซึ่งขั้นตอนการคำนวณหาค่า

แตรด โซลด์ในอัลกอริทึมนี้ประยุกต์มาจากวิธี การทำแตรด โซลด์ตามคุณลักษณะของวัตถุ (Object Attribute Threshold) [2] โดยวัตถุในที่นี้ก็คือ ขอบของตัวอักษรนั่นเอง

ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 23



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริทึมฟังก์ชันร่วม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 24

-1	0	1
-2	0	2
-1	0	1

(ก)

-1	-2	-1
0	0	0
1	2	1

(ข)

แสดงวินโดว์ขนาด 3x3 (ก) สำหรับหา G_x (ข) สำหรับหา G_y

ขั้นตอนที่ 3 การนิยามจุดภาพให้อยู่ในรูปของเครื่องหมาย

ทำการนิยามจุดภาพทั้งหมดของภาพเอกสารระดับความเข้มเริ่มต้นให้อยู่ในรูปของเครื่องหมายโดยใช้ตัวดำเนินการลาปลาเซียน (Laplacian Operator) ซึ่งคำนวณได้โดยใช้วินโดว์ขนาด 3x3 ดังภาพที่ 25 มาคอนโวลูชันกับข้อมูลภาพเริ่มต้น ผลลัพธ์ที่ได้คือ $\nabla^2 g(x,y)$ ซึ่งจะถูกนำมาใช้กำหนดเครื่องหมายของข้อมูลแต่ละจุดภาพในขั้นตอนต่อไป

ภาพที่ 25

-1	-1	-1
-1	8	-1
-1	-1	-1

แสดงวินโดว์ขนาด 3x3 ที่ใช้สำหรับหาภาพลาปลาเซียน

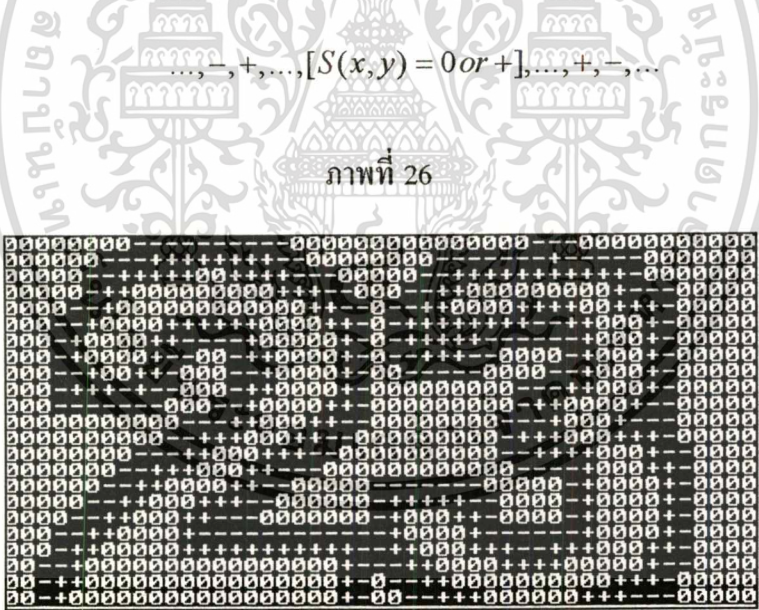
ขั้นตอนที่ 4 การนิยามภาพ $S(x,y)$ ที่มี 3 สถานะ

นำค่าทั้ง 3 คือ $\nabla g(x,y)$, Thr (ค่าเทรชโฮลด์), $\nabla^2 g(x,y)$ มาประยุกต์ใช้ร่วมกันสำหรับการตัดสินใจสร้างภาพใหม่ $S(x,y)$ ที่มี 3 สถานะโดยที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$s(x,y) = \begin{cases} 0 & \text{if } \nabla g(x,y) < Thr \\ + & \text{if } \nabla g(x,y) \geq Thr \text{ and } \nabla^2 g(x,y) > 0 \\ - & \text{if } \nabla g(x,y) \geq Thr \text{ and } \nabla^2 g(x,y) < 0 \end{cases} \quad (3.2)$$

สัญลักษณ์ 0,+,- ทั้งสามจะถูกนำมาใช้แทนระดับความเข้มของภาพ โดยจุดภาพทั้งหมดที่ไม่อยู่ในบริเวณที่เป็นขอบที่ต้องการ (ซึ่งตัดสินโดย $(\nabla g(x,y) < Thr)$) จะถูกนิยามเป็นศูนย์ ส่วนจุดภาพทั้งหมดที่เป็นขอบและอยู่ทางด้านมืดจะถูกนิยามโดยเครื่องหมายบวก และจุดภาพทั้งหมดที่เป็นขอบและอยู่ทางด้านสว่างจะถูกนิยามโดยเครื่องหมายลบ จากเงื่อนไขข้างต้นทำให้ขอบที่เกิดจากสิ่งที่ไม่พึงประสงค์บางส่วนจะถูกกำจัดทิ้งไปในภาพ $S(x,y)$ และนำภาพ $S(x,y)$ ไปใช้ในการสร้างภาพ 2 ระดับในขั้นตอนสุดท้าย ภาพที่ 26 แสดงตัวอย่างภาพ $S(x,y)$ เราสามารถสังเกตได้ว่าจุดภาพที่อยู่ภายในเส้นตัวอักษรจะถูกนิยามด้วยเครื่องหมาย + หรือ - ไม่ที่ 0 และมีขอบเขตที่จะแสดงตามลำดับดังต่อไปนี้



ตัวอย่างภาพ S(x,y)

ขั้นตอนที่ 5 การนิยามภาพ 2 ระดับ

ทำการสร้างภาพผลลัพธ์ที่มีเพียง 2 ระดับ โดยจุดภาพที่นิยามด้วยเครื่องหมายลบทั้งหมดจะถูกทำให้เป็นสีขาว และจุดภาพที่นิยามด้วยศูนย์แต่อยู่นอกขอบเขตของขอบ (-,+,-) ก็จะถูกกำหนดให้เป็นสีขาว ส่วนการตัดสินใจสร้างภาพ 2 ระดับ คำขาว กับจุดภาพที่อยู่ในขอบเขตของ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาค้นคว้าเท่านั้น ไม่สามารถนำออกจำหน่ายหรือทำซ้ำโดยไม่ได้รับอนุญาตจากเจ้าของลิขสิทธิ์ หากมีข้อผิดพลาดประการใดขออภัยเป็นอย่างสูง

หาขอบเขตของขอบทั้งสองด้านและตรวจสอบระยะทางที่ได้ว่า ได้สัดส่วนกับค่าที่ต้องการหรือไม่ คือ ความกว้างของเส้นตัวอักษร ถ้าไม่ได้สัดส่วน จุดภาพนั้นจะถูกกำหนดให้เป็นสีขาว ถ้าได้สัดส่วน จุดภาพนั้นจะถูกกำหนดให้เป็นสีดำ ซึ่งการทดสอบในแต่ละจุดภาพจะถูกกำหนดไว้ 4 ทิศทางสัมพันธ์กันไป ดังอัลกอริทึมที่ 1 และจากการตรวจสอบในลักษณะดังกล่าวจะทำให้ส่วนของสิ่งที่ไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงหรือระดับเดียวกับตัวอักษรจะถูกจำกัดทิ้งไป ยกเว้นแต่สิ่งไม่พึงประสงค์นั้นจะมีขนาดใกล้เคียงกับขนาดของตัวอักษร ผลลัพธ์ที่ได้จากขั้นตอนนี้จะ เป็นภาพไบนารีที่มีตัวอักษรสีดำบนพื้นหลังสีขาว

อัลกอริทึมที่ 1

Begin

if (จุดภาพ $S(x,y)$ อยู่ในขอบเขตของขอบ $(-,+,...,+ -)$) *then*

 สืบหา d_R, d_L, d_U, d_D แสดงดังภาพที่ 27

 ค่าความกว้างโดยประมาณ (*Width*) เท่ากับ ค่าที่น้อยที่สุดระหว่างค่า $d_U + d_D$ กับ ค่า $d_R + d_L$

if ($(Width < W)$ หรือ $(Width > W)$) *then*

if ($Width < W$) *then* $b(x,y) = 0$

else

 สืบหาค่า $d_{UL}, d_{DR}, d_{DL}, d_{UR}$ แสดงดังภาพที่ 27

 ค่าความกว้างโดยประมาณ (*Width*) เท่ากับ ค่าที่น้อยที่สุดระหว่างค่า $d_{UL} + d_{DR}$ กับ

 ค่า $d_{DL} + d_{UR}$

if ($(Width < W)$ หรือ $(Width > W)$) *then* $b(x,y) = 0$

else $b(x,y) = 1$

else $b(x,y) = 1$

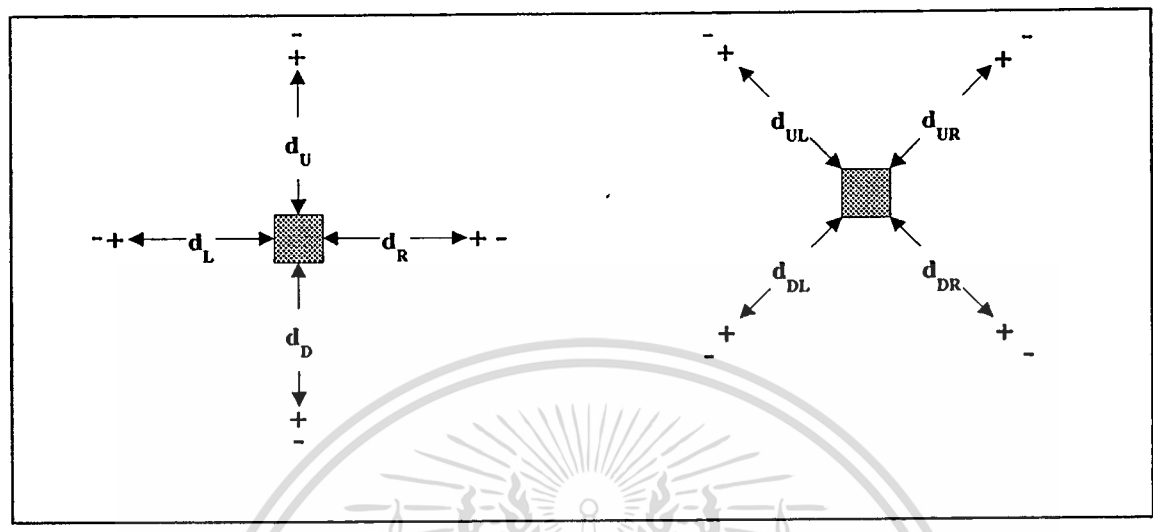
else $b(x,y) = 0$

End

โดยที่

$b(x,y)$ คือ ภาพไบนารี จุดภาพที่นิยามด้วย 1 แสดงถึงจุดภาพของตัวอักษร จุดภาพที่นิยามด้วย 0 แสดงถึงจุดภาพของพื้นหลัง

ภาพที่ 27



แสดงตำแหน่งของจุดภาพ $S(x,y)$ ที่อยู่ภายในขอบเขตของขอบ $(-+,...,+ -)$

ส่วนที่ทำการพัฒนาจากอัลกอริทึมของ J.M. White และ R.D. Rohrer

อัลกอริทึมฟังก์ชันร่วมเป็นอัลกอริทึมที่พัฒนามาจากอัลกอริทึมของ J.M. White และ R.D. Rohrer [5] ซึ่งสิ่งที่พัฒนาเพิ่มขึ้นมา คือ รูปแบบการคำนวณหาค่าเรดโซลด์จากภาพโมดูลกราเดียนท์ เพื่อนำค่าเรดโซลด์มาใช้ในการนิยามขอบของตัวอักษร ซึ่งในอัลกอริทึมของ White กับ Rohrer เดิม ค่าเรดโซลด์จะได้จากการกำหนดค่าโดยผู้ใช้ (Fix Threshold) สำหรับอีกส่วนหนึ่งที่ทำการพัฒนาขึ้นมาในอัลกอริทึมฟังก์ชันร่วม คือ รูปแบบการนิยามภาพสองระดับจากภาพ $S(x,y)$ ซึ่งอัลกอริทึมฟังก์ชันร่วมจะทำการพิจารณาคุณสมบัติของจุดภาพ $S(x,y)$ ที่อยู่ภายในขอบเขต $(-+,...,+ -)$ โดยการตรวจสอบจุดภาพใน 4 ทิศทาง แต่อัลกอริทึมของ White กับ Rohrer เดิม จะทำการตรวจสอบจุดภาพแค่ 2 ทิศทาง คือ ทิศทางในแนวตั้ง และทิศทางในแนวนอน

การแยกภาพตัวอักษรโดยการทำเรดโซลด์ตามคุณลักษณะของวัตถุ

อัลกอริทึมสำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารในรูปแบบที่จะนำเสนอต่อไปนี้ อยู่บนพื้นฐานของการเซกเมนต์ภาพโดยการทำเรดโซลด์แบบครอบคลุม (Global Thresholding) คือ การใช้ค่าเรดโซลด์เพียงค่าเดียวทำเรดโซลด์กับทุก ๆ จุดภาพของภาพเอกสาร แต่ขั้นตอนการคำนวณหาค่าเรดโซลด์แบบครอบคลุมในอัลกอริทึมนี้ได้แนะนำแนวความคิดบางส่วนของ การหา ค่าเรดโซลด์แบบปรับค่ามาประยุกต์ใช้ คือ การปรับค่าเรดโซลด์แบบครอบคลุมให้มีเหมาะสมกับภาพมากที่สุด และภาพผลลัพธ์ที่ได้จะมีความถูกต้องมากที่สุด

หลักการที่นำเสนอ

ที่มาของหลักการที่ต้องการจะนำเสนอในอัลกอริทึมรูปแบบนี้ คือ การคำนวณหา ค่าเรดโซลด์ที่สามารถแบ่งระดับความเข้มบนฮิสโตแกรมระดับความเข้มของภาพเอกสารออกเป็น 2 กลุ่มได้อย่างถูกต้อง คือ กลุ่มระดับความเข้มของตัวอักษร และกลุ่มระดับความเข้มของส่วนพื้นหลัง แต่จากการคำนวณหา ค่าเรดโซลด์โดยใช้อัลกอริทึมของ Otsu [6] พบว่าค่าเรดโซลด์ที่คำนวณได้จะเหมาะกับภาพเอกสารที่มีจำนวนพีก (Peak) ที่เกิดขึ้นบนฮิสโตแกรมระดับความเข้มแค่ 2 พีก ซึ่งค่าเรดโซลด์เหล่านี้จะทำให้ภาพผลลัพธ์ที่ได้มีความถูกต้องมากที่สุด แต่ในกรณีของภาพเอกสารที่มีลักษณะของความยุ่งยาก คือ ภาพเอกสารที่มีความไม่สม่ำเสมอของระดับความเข้มในส่วนของตัวอักษร หรือมีความไม่สม่ำเสมอของระดับความเข้มในส่วนของพื้นหลัง หรือมีความไม่สม่ำเสมอทั้งสองส่วน ซึ่งกรณีดังกล่าวจะเป็นสาเหตุทำให้เกิดความหลากหลายของจำนวนพีกบนฮิสโตแกรมระดับความเข้ม การคำนวณหา ค่าเรดโซลด์ด้วยอัลกอริทึมของ Otsu กับภาพในลักษณะดังกล่าวจะให้ค่าเรดโซลด์ที่ไม่มีความเหมาะสม ดังนั้นในอัลกอริทึมรูปแบบนี้จึงนำเสนอ ขั้นตอนการทำเรดโซลด์ในรูปแบบหนึ่งที่สามารถแก้ปัญหาที่เกิดขึ้นในขั้นตอนการแยกภาพตัวอักษรข้างต้น เรียกว่า การทำเรดโซลด์ตามคุณลักษณะของวัตถุ (Object Attribute Thresholding, OAT) [2] ซึ่งการทำเรดโซลด์ในรูปแบบนี้จะนำอัลกอริทึมของ Otsu [6] มาประยุกต์ใช้ ดังนั้นความยุ่งยากของภาพเอกสารที่มีสาเหตุขึ้นอันเนื่องมาจากความหลากหลายของจำนวนพีกบนฮิสโตแกรมระดับความเข้มจะถูกกำจัดลง โดยค่าเรดโซลด์จะถูกปรับค่าให้มีความเหมาะสมยิ่งขึ้น และใช้คุณลักษณะของวัตถุ (คือ ระดับความเข้มของตัวอักษร) มาพิจารณาความเหมาะสมของค่าเรดโซลด์ และค่าเรดโซลด์ที่ได้จากการปรับในแต่ละครั้งจะเข้าใกล้ค่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษรมากยิ่งขึ้น ซึ่งการทำงานด้วยหลักการดังกล่าวเปรียบเสมือนกับการทำซ้ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่ออนุญาตเห็นาใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Iterative Thresholding) จนกระทั่งได้ค่าเรดโซลด์ที่เหมาะสม สำหรับการทำงานของอัลกอริทึม ด้วยหลักการดังกล่าว สามารถแสดงในรูปของแผนภาพการทำงานในภาพที่ 28

วิธีการที่เสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการทำงานออกเป็นขั้นตอนย่อย ๆ ดังต่อไปนี้

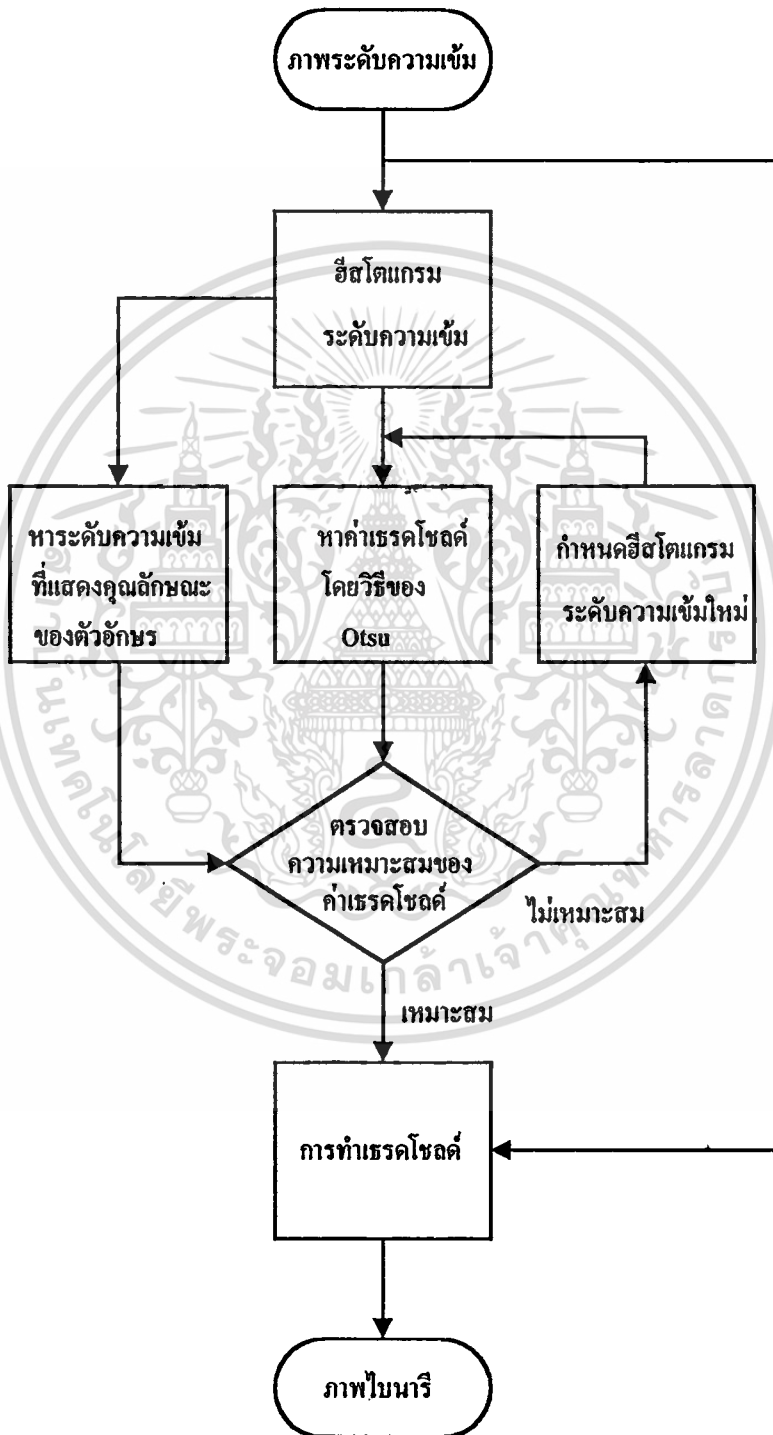
ขั้นตอนที่ 1 การคำนวณหาระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร

นำภาพเอกสารระดับความเข้มเริ่มต้นมาประมวลผลเพื่อหาระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร โดยมีขั้นตอนดำเนินการดังต่อไปนี้

1.1 การทำพรีเทสต์ (Pretest) กับภาพเอกสารเริ่มต้น ในรูปของการคำนวณหาค่าเรดโซลด์จากฮิสโตแกรมระดับความเข้มของภาพโดยใช้อัลกอริทึมของ Otsu [6] หลังจากนั้นทำการตรวจสอบลักษณะการกระจายของระดับความเข้มจุดภาพบนฮิสโตแกรม โดยดูจากค่าความน่าจะเป็นของแต่ละระดับความเข้มระหว่างช่วง “Black” ($Black \approx 0$) กับ ค่าเรดโซลด์ที่ได้จากการทำพรีเทสต์ (Pt_1) ค่าระดับความเข้มใดที่มีค่าความน่าจะเป็นสูงในช่วงนี้ หรือ มีอัตราการเพิ่มของค่าความน่าจะเป็นสูงเมื่อเทียบกับค่าความน่าจะเป็นที่คำนวณจากฮิสโตแกรมระดับเข้มทั้งหมด ค่าระดับความเข้มค่า A_1 จะมีความเป็นไปได้ที่จะเป็นระดับความเข้มที่มีคุณลักษณะของตัวอักษร ทำการตรวจสอบคุณสมบัติของ A_1 โดยการทดสอบว่าช่วงระหว่าง “Black” กับ “ A_1 ” บนฮิสโตแกรมระดับเข้ม สามารถแบ่งระดับความเข้มออกเป็นสองกลุ่มโดยอัลกอริทึมของ Otsu ได้หรือไม่ ถ้าไม่สามารถแบ่งได้แสดงว่าระดับความเข้ม A_1 เป็นระดับความเข้มที่มีคุณลักษณะของตัวอักษร แต่ถ้าสามารถแบ่งได้โดยค่าเรดโซลด์ (Pt_2) จะทำการตรวจสอบลักษณะการกระจายของระดับความเข้มบนฮิสโตแกรมระหว่างช่วง “Black” กับ “ Pt_2 ” เพื่อหา A_2 หลังจากนั้นก็จะทำการตรวจสอบคุณสมบัติของ A_2 ด้วยหลักการข้างต้น ทำจนกระทั่งได้ค่า A_i ที่ไม่สามารถแบ่งต่อไปได้อีก หรือ ค่า A_i ที่สามารถแบ่งต่อไปได้ แต่มีจำนวนจุดภาพมากกว่าจำนวนจุดภาพที่ A_{i+1} หนึ่งเท่า และจะอนุมานระดับความเข้มที่ A_i ว่าเป็นระดับความเข้มที่มีคุณลักษณะของตัวอักษร แต่บางภาพเอกสารจะพบว่ามีความไม่สม่ำเสมอของระดับความเข้มของตัวอักษรเกิดขึ้น คือ ตัวอักษรในภาพนั้นมีระดับความเข้มที่แตกต่างกัน ดังนั้นค่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษรจะเท่ากับค่าระดับความเข้มดังสมการที่ (3.3)

$$Attribute = A_i + t \quad (3.3)$$

ภาพที่ 28



แผนภาพแสดงขั้นตอนการทำงานของ OAT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 2 การคำนวณหาค่าเรคโพลด์ที่มีความเหมาะสมกับภาพเอกสาร

ค่าเรคโพลด์ที่คำนวณได้ต้องไม่น้อยกว่าค่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร ซึ่งขั้นตอนการปรับค่าเรคโพลด์ที่เหมาะสมสามารถกระทำดังต่อไปนี้

2.1 กำหนด $j = 1$

2.2 คำนวณฮิสโตแกรมระดับความเข้มของภาพ (G_j)

2.3 นำอัลกอริธึมของ Otsu [6] มาคำนวณหาค่าเรคโพลด์ t_j จาก G_j

2.4 คำนวณฮิสโตแกรมระดับความเข้มใหม่โดยให้ G_{j+1} เท่ากับ ส่วนของ G_j ที่อยู่ระหว่าง “Black” กับ t_j

2.5 กำหนดให้ $j = j + 1$ แล้วทำซ้ำข้อ (2.3) และ (2.4) จนกระทั่งค่าเรคโพลด์ที่ได้จากการปรับไม่อยู่ได้กฎเกณฑ์ที่กำหนดขึ้น คือ มีค่าน้อยกว่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร ดังนั้นค่าเรคโพลด์ที่ได้จากวิธีของ OAT นี้คือ ค่าเรคโพลด์ค่าสุดท้ายที่ได้จากการปรับและอยู่ภายใต้กฎเกณฑ์ที่กำหนด ซึ่งเป็นค่าที่มีความเหมาะสม

ขั้นตอนที่ 3 การทำเรคโพลด์แบบครอบคลุมกับภาพเอกสารเริ่มต้น

ทำเรคโพลด์แบบครอบคลุม (Global Thresholding) กับทุก ๆ จุดภาพของภาพเอกสารเริ่มต้น โดยใช้ค่าเรคโพลด์ที่มีความเหมาะสมที่คำนวณได้จากขั้นตอนที่ 2 ซึ่งภาพผลลัพธ์ที่ได้เป็นภาพไบนารี

การแยกภาพตัวอักษรโดยอัลกอริธึมการทำแธรชโวลด์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียน

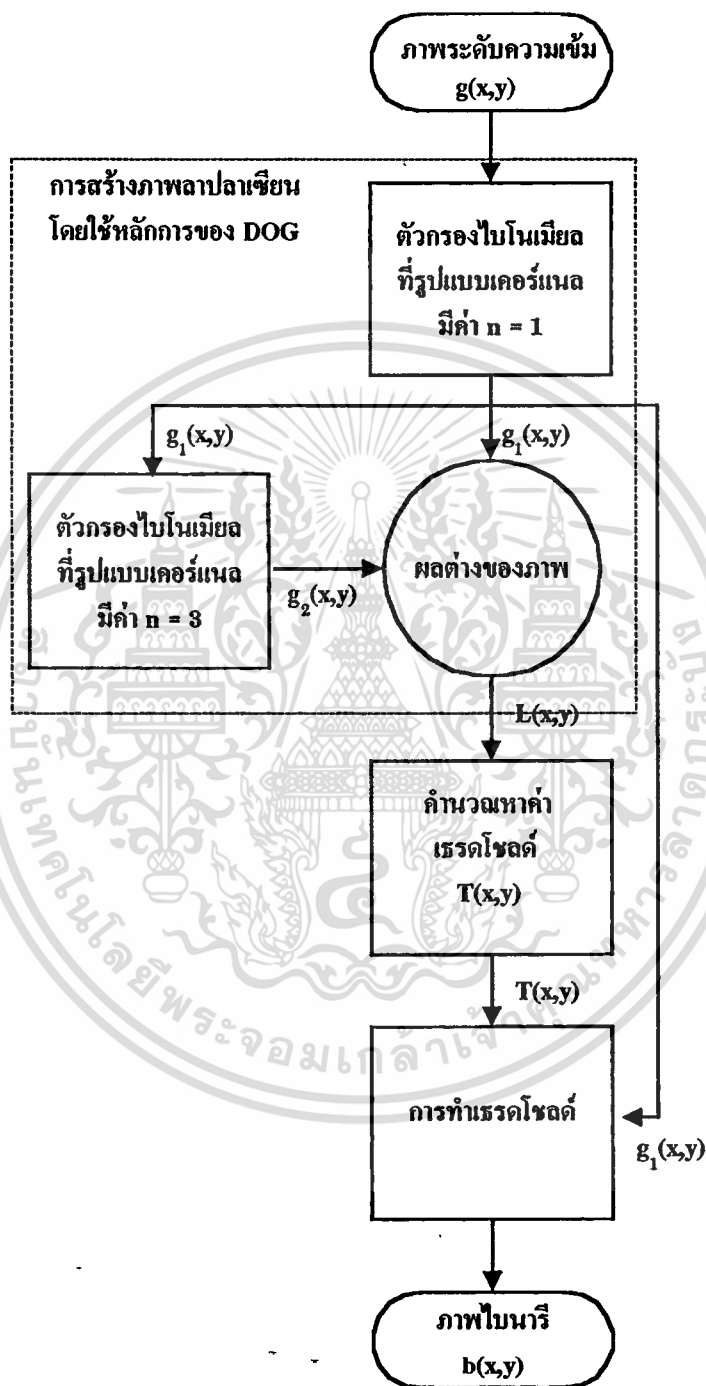
อัลกอริธึมสำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิกออกจากภาพเอกสารเริ่มต้นในรูปแบบที่จะนำเสนอต่อไปนี้อยู่บนพื้นฐานของการเชกเมนต์ภาพ โดยการทำแธรชโวลด์แบบปรับค่า (Adaptive Thresholding) โดยนำเอาภาพเครื่องหมายลาปลาเซียนมาประยุกต์ใช้ในขั้นตอนการปรับค่าแธรชโวลด์ ซึ่งภาพเครื่องหมายลาปลาเซียนในอัลกอริธึมรูปแบบนี้จะถูกนิยามโดยโอเปอเรเตอร์ที่อยู่บนพื้นฐานของการหาอนุพันธ์อันดับที่สอง

หลักการที่นำเสนอ

หลักการที่นำเสนอในอัลกอริธึมการแยกภาพตัวอักษรรูปแบบนี้ คือ การคำนวณหาค่าแธรชโวลด์ใหม่ทุกครั้งที่มีการเปลี่ยนแปลงเครื่องหมายจุดภาพของภาพลาปลาเซียนตามแนวสแกนบริเวณที่มีการเปลี่ยนแปลงเครื่องหมายจุดภาพของภาพลาปลาเซียน คือ บริเวณที่เป็นขอบภาพนั่นเอง ซึ่งเป็นบริเวณที่แสดงให้เห็นว่ามีการเปลี่ยนแปลงทางกายภาพของข้อมูลภาพเกิดขึ้น ดังเช่นการเปลี่ยนจากองค์ประกอบย่อยหนึ่งไปเป็นอีกองค์ประกอบย่อยหนึ่งภายในภาพ ดังนั้นค่าแธรชโวลด์ที่ดีควรปรับเปลี่ยนให้เหมาะสมกับจุดภาพในองค์ประกอบย่อยนั้น ไม่ใช่การใช้ค่าแธรชโวลด์เพียงค่าเดียวกับจุดภาพในทุก ๆ องค์ประกอบย่อยของภาพ สำหรับการทำแธรชโวลด์กับแต่ละจุดภาพของภาพเอกสารนั้น ถ้าระดับความเข้มของจุดภาพน้อยกว่าค่าแธรชโวลด์แสดงว่าเป็นจุดภาพของส่วนตัวอักษรหรือสัญลักษณ์กราฟฟิก แต่ถ้ามากกว่าค่าแธรชโวลด์แสดงว่าเป็นจุดภาพของพื้นหลัง สำหรับทำงานของอัลกอริธึมด้วยหลักการดังกล่าว สามารถแสดงในรูปของแผนภาพการทำงานในภาพที่ 29

สำหรับขั้นตอนการนิยามภาพเครื่องหมายพบว่า การนิยามภาพเครื่องหมายโดยใช้ตัวดำเนินการลาปลาเซียนเช่นอัลกอริธึมฟังก์ชันร่วมนั้น ภาพลาปลาเซียนที่ได้จะมีความไวต่อสัญญาณรบกวน (Noise) ไม่เหมาะสมที่จะนำมาประมวลผลกับอัลกอริธึมในรูปแบบนี้ ซึ่งคุณลักษณะที่ดีของภาพเครื่องหมายลาปลาเซียนนั้นควรมีลักษณะของความราบเรียบ (Smoothing) ดังนั้นในอัลกอริธึมนี้จึงแก้ไขข้อจำกัดที่เกิดขึ้นข้างต้น โดยนำหลักการของดิฟเฟอเรนเชียลของฟูกาส์เซียน (Differential Of Two Gaussian, DOG [4]) มาใช้สำหรับการนิยามภาพลาปลาเซียนในรูปของภาพเครื่องหมาย ภาพที่ได้จะมีลักษณะของความราบเรียบมากกว่า

ภาพที่ 29



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรตโซลด์
แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียน

วิธีการที่เสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการประมวลผลออกเป็น ขั้นตอนย่อย ๆ ตอนดังต่อไปนี้

ขั้นตอนที่ 1 นิยามภาพเครื่องหมายลาปลาเซียน

นำหลักการของ DOG [4] มาใช้ในการนิยามภาพเครื่องหมายลาปลาเซียน ซึ่งมีขั้นตอนการดำเนินงานดังต่อไปนี้

1.1 นำภาพเอกสารระดับความเข้ม $g(x,y)$ มากรองความถี่ด้วยตัวกรองสัญญาณแบบไบโนเมียลที่ได้รับการพิสูจน์ว่ามีคุณสมบัติใกล้เคียงกับตัวกรองสัญญาณแบบเกาส์เซียน [3] และมีรูปแบบเคอร์เนลของตัวกรองความถี่ดังต่อไปนี้

$$f(x,y,n) = \frac{1}{16^n} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}^{*n} \quad (3.4)$$

โดยที่พารามิเตอร์ n หมายถึง การคอนโวลูชันของตัวคอร์เนล n ครั้ง ซึ่งเทียบกับ σ ในฟังก์ชันเกาส์เซียนได้คือ

$$\sigma^2 = 0.5n \quad (3.5)$$

การปฏิบัติในขั้นตอนนี้ n จะมีค่าเท่ากับ 1 และให้ภาพผลลัพธ์ $g_1(x,y)$

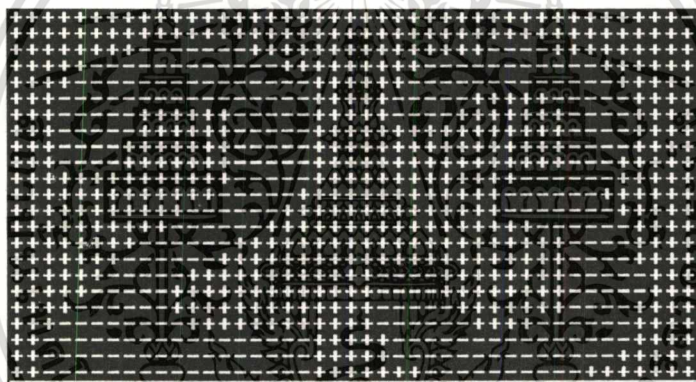
1.2 นำเคอร์เนลพื้นฐานขนาด 3×3 ดังกล่าว มาคาสเคด (cascade) จำนวน 3 ครั้ง กับภาพ $g_1(x,y)$ เพื่อปรับแบนด์วิธให้สอดคล้องกับหลักการของ DOG [4] และการปฏิบัติในขั้นตอนนี้จะให้ภาพผลลัพธ์ $g_2(x,y)$

1.3 คำนวณหาดิฟเฟอเรนเชียลของเกาส์เซียน โดยการหาผลต่างระหว่างภาพ $g_1(x,y)$ กับ $g_2(x,y)$ และทำการนิยามภาพเครื่องหมายลาปลาเซียน $L(x,y)$ ดังสมการต่อไปนี้

$$L(x,y) = \begin{cases} + & \text{if } g_1(x,y) - g_2(x,y) > 0 \\ - & \text{if } g_1(x,y) - g_2(x,y) < 0 \end{cases} \quad (3.6)$$

ภาพผลลัพธ์ที่ได้จากขั้นตอนนี้จะอยู่ในรูปของภาพเครื่องหมาย +, - ดังภาพตัวอย่างที่ 30 ซึ่งจะมีการเปลี่ยนแปลงเครื่องหมายเกิดขึ้นกับจุดภาพที่เป็นขอบ เช่น จุดภาพบริเวณที่เป็นขอบขาเข้าตัวอักษรจะมีการเปลี่ยนแปลงเครื่องหมายจุดภาพจากเครื่องหมายบวกไปเป็นเครื่องหมายลบ และจุดภาพบริเวณที่เป็นขอบขาออกจากตัวอักษรจะมีการเปลี่ยนแปลงเครื่องหมายจุดภาพจากเครื่องหมายลบไปเป็นเครื่องหมายบวก

ภาพที่ 30



แสดงภาพเครื่องหมายลาปลาเซียน $L(x,y)$ ที่นิยามด้วยหลักการของ DOG

ขั้นตอนที่ 2 การคำนวณหาค่าเรดโซลต์ที่พิกัดจุดภาพ (x,y)

ค่าเรดโซลต์จะมีการปรับค่าใหม่ทุกครั้งที่จุดภาพอยู่ในตำแหน่งที่เป็นขอบ คือ มีการเปลี่ยนแปลงเครื่องหมายจุดภาพของภาพลาปลาเซียน $L(x,y)$ นั้นเอง ซึ่งการคำนวณหาค่าเรดโซลต์สามารถหาได้จากขั้นตอนการดำเนินงานต่อไปนี้

2.1 จุดภาพที่พิกัดเริ่มต้น ค่าระดับสีขาว จะเท่ากับ $g_2(x,y)$ และค่าระดับสีดำจะเท่ากับ ศูนย์

$$White = g_2(x,y) \quad (3.7)$$

$$Black = 0 \quad (3.8)$$

2.2 จุดภาพที่พิกัด (x,y) จะมีการปรับค่าระดับสีดำในกรณีที่มีการเปลี่ยนแปลงเครื่องหมายจุดภาพของภาพ $L(x,y)$ จากเครื่องหมายบวกเป็นลบ โดยค่าระดับสีดำจะเท่ากับ

$$Black = g_2(x,y) \quad (3.9)$$

2.3 จุดภาพที่พิกัด (x,y) จะมีการปรับค่าระดับสีขาวในกรณีที่มีการเปลี่ยนแปลงเครื่องหมายจุดภาพของภาพ $L(x,y)$ จากเครื่องหมายลบเป็นบวก โดยค่าระดับสีขาวจะเท่ากับ

$$White = g_2(x,y) \quad (3.10)$$

2.4 ค่าเรดโซลต์ที่พิกัดจุดภาพ (x,y) จะเท่ากับ

$$Thr(x,y) = k * \frac{(White + Black)}{2} \quad (3.11)$$

โดยที่ k คือ ค่าคงที่ใช้สำหรับการถ่วงน้ำหนัก เพราะถ้าปราศจากการถ่วงน้ำหนักค่าแล้วการทำเรดโซลต์ในแต่ละจุดภาพอาจให้ผลผิดพลาดได้อันเนื่องมาจากสิ่งที่ไม่พึงประสงค์ในส่วนของพื้นหลังที่มีรูปแบบที่แน่นอน

ขั้นตอนที่ 3 การทำเรดโซลต์กับภาพ $g_1(x,y)$ โดยค่า $Thr(x,y)$

$$b(x,y) = \begin{cases} 1 & \text{if } g_1(x,y) < Thr(x,y) \\ 0 & \text{otherwise} \end{cases} \quad (3.12)$$

จุดภาพของภาพไบนารีที่นิยาม 1 เป็นจุดภาพของตัวอักษรหรือสัญลักษณ์กราฟฟิก ส่วนจุดภาพที่นิยามด้วย 0 เป็นจุดภาพของส่วนพื้นหลัง

การแยกภาพตัวอักษรโดยอัลกอริธึมการประเมินผลต่างของภาพ

อัลกอริธึมการประเมินผลต่างของภาพเป็นอัลกอริธึมที่นำแนวความคิดบางส่วนของขั้นตอนการทำงานของอัลกอริธึมฟังก์ชันร่วมมาประยุกต์ใช้สำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารที่มีความซับซ้อนเกิดขึ้นในส่วนของพื้นหลัง อัลกอริธึมการแยกภาพตัวอักษรในรูปแบบนี้ใช้พารามิเตอร์ที่สำคัญ 2 ค่า คือ ค่าผลต่างของระดับความเข้มที่มีความเป็นไปได้ระหว่างตัวอักษรกับพื้นหลังในภาพเอกสาร และ ค่าความกว้างของเส้นตัวอักษร

หลักการที่นำเสนอ

โดยปกติแล้วคอมพิวเตอร์จะไม่สามารถเข้าใจได้ว่าจุดภาพของข้อมูลภาพเอกสารระดับความเข้มนั้นจุดภาพในพิกัดใดเป็นจุดภาพของตัวอักษรและจุดภาพในพิกัดใดเป็นจุดภาพของส่วนพื้นหลัง จนกว่าภาพเอกสารระดับความเข้มนั้นจะถูกนิยามให้อยู่ในรูปของสัญลักษณ์ คือ ภาพไบนารี ดังนั้นในอัลกอริธึมที่จะกล่าวต่อไปนี้จะนำเสนอหลักการสำหรับการนิยามภาพเอกสารระดับความเข้มให้อยู่ในรูปของภาพสัญลักษณ์ ซึ่งหลักการที่นำเสนอนี้จะเริ่มต้นจากการนิยามภาพพื้นหลัง (*background image*) ของภาพเอกสารเริ่มต้น โดยจุดภาพของภาพเอกสารใดสามารถพิสูจน์ได้ว่าเป็นจุดภาพของพื้นหลัง ก็จะคงระดับความเข้มของจุดภาพนั้นในภาพพื้นหลัง แต่ถ้าจุดภาพใดที่ไม่สามารถพิสูจน์ได้ว่าเป็นจุดภาพพื้นหลัง ระดับความเข้มของจุดภาพนั้นในภาพพื้นหลังจะได้รับการอินทไปเลระดับความเข้มของแบคกราวด์รอบข้าง หลังจากนั้นทำการนิยามภาพผลลัพธ์โดยการประเมินผลต่างระหว่างภาพเอกสารระดับความเข้มเริ่มต้นกับภาพพื้นหลัง ซึ่งการทำงานด้วยหลักการดังกล่าวประกอบด้วยขั้นตอนการทำงานทั้งหมด 4 ขั้นตอน สำหรับขั้นตอนแรกทำการนิยามจุดภาพของพื้นหลังจากข้อมูลภาพเอกสารระดับความเข้มเริ่มต้นโดยพิจารณาจากเกณฑ์ที่กำหนดขึ้น และภาพผลลัพธ์ที่ได้เป็นภาพไบนารี ที่จุดภาพสีขาวนิยามถึงจุดภาพของพื้นหลัง และจุดภาพสีดำนิยามถึงจุดภาพที่มีความเป็นไปได้ที่จะเป็นจุดภาพของตัวอักษร ซึ่งจุดภาพสีดำเหล่านี้บางจุดภาพจะมีลักษณะเป็น *Particle Noise* เรียกการทำงานในขั้นตอนแรกนี้ว่า ขั้นตอนการกรองหาจุดภาพพื้นหลัง (*Background Filtering*) ส่วนการทำงานในขั้นตอนที่สองจะนำภาพผลลัพธ์ที่ได้จากขั้นตอนแรกมาทำปรับปรุงโดยใช้ความกว้างของเส้นตัวอักษรเป็นเกณฑ์ในการปรับปรุง โดยพิจารณากับจุดภาพสีดำเพื่อตัดสินใจว่าจุดภาพเหล่านี้จุดภาพใดเป็นจุดภาพที่มีคุณสมบัติของพื้นหลัง ภาพผลลัพธ์ที่ได้จากขั้นตอนนี้จะอยู่ในรูปของภาพไบนารีดังเช่นภาพผลลัพธ์ที่

ได้จากขั้นตอนแรก แต่จำนวนจุดภาพของพื้นหลังจะมีจำนวนมากกว่า ซึ่งภาพผลลัพธ์ในขั้นตอนนี้ จุดภาพสีขาวจะนิยามถึงจุดภาพของส่วนพื้นหลัง ส่วนจุดภาพสีดำจะนิยามถึงจุดภาพที่มีความเป็นไปได้ที่จะเป็นจุดภาพของตัวอักษร ซึ่งจุดภาพสีดำเหล่านี้อาจเป็นจุดภาพของตัวอักษรจริงหรือจุดภาพของสิ่งที่ไม่พึงประสงค์ที่มีขนาดใกล้เคียงกับตัวอักษร ขั้นตอนที่สามนำภาพเอกสารถะดับความเข้มเริ่มต้นและภาพไบนารีที่ได้จากการปรับปรุงในขั้นตอนการทำงานที่สองมาประยุกต์ใช้ร่วมกันสำหรับสร้างภาพพื้นหลัง (*background image*) สำหรับขั้นตอนสุดท้ายจะทำการประเมินผลต่างภาพ (*Subtraction Image*) ระหว่างภาพพื้นหลังกับภาพเอกสารถะดับความเข้มเริ่มต้น สำหรับการสร้างภาพผลลัพธ์ โดยใช้ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรกับของพื้นหลังเป็นเกณฑ์ในการประเมิน ภาพผลลัพธ์ที่ได้ในขั้นตอนนี้จะป็นภาพไบนารีที่จุดภาพสีขาวแสดงถึงส่วนที่เป็นพื้นหลัง ส่วนจุดภาพสีดำแสดงถึงส่วนที่เป็นตัวอักษร ซึ่งรูปแบบการทำงานของอัลกอริธึมดังกล่าวนี้สามารถแสดงในรูปของแผนภาพดังภาพที่ 31

วิธีการนำเสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการทำงานออกเป็น 4 ขั้นตอนดังต่อไปนี้

ขั้นตอนที่ 1 การกรองหาจุดภาพพื้นหลัง

นำภาพเอกสารถะดับความเข้มมาผ่านขั้นตอนการกรองหาจุดภาพพื้นหลัง ซึ่งจะอธิบายขั้นตอนการทำงานในรูปของอัลกอริธึมดังต่อไปนี้

ภาพที่ 31



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการหาผลต่างของภาพ
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Begin

if (อย่างน้อยมี W จุดภาพจากเซตของจุดภาพ $\{g(x+i,y), -W \leq i \leq W\}$
 ที่มีระดับความเข้มมากกว่า $g(x,y)$) then $b_1(x,y) = 1$

else

if (อย่างน้อยมี W จุดภาพจากเซตของจุดภาพ $\{g(x,y+i), -W \leq i \leq W\}$
 ที่มีระดับความเข้มมากกว่า $g(x,y)$) then $b_1(x,y) = 1$

else

if (อย่างน้อยมี W จุดภาพจากเซตของจุดภาพ $\{g(x+i,y+i), -W \leq i \leq W\}$
 ที่มีระดับความเข้มมากกว่า $g(x,y)$) then $b_1(x,y) = 1$

else

if (อย่างน้อยมี W จุดภาพจากเซตของจุดภาพ $\{g(x-i,y+i), -W \leq i \leq W\}$
 ที่มีระดับความเข้มมากกว่า $g(x,y)$) then $b_1(x,y) = 1$

else $b_1(x,y) = 0$

End

โดยที่

$b_1(x,y)$ คือ ภาพไบนารี จุดภาพที่นิยามด้วย 1 แสดงถึงจุดภาพที่มีความเป็นไปได้ที่จะมี
 คุณสมบัติของจุดภาพของตัวอักษร ส่วนจุดภาพที่นิยามด้วย 0 แสดงถึงจุดภาพของส่วนพื้นหลัง

ขั้นตอนที่ 2 ปรับปรุงภาพผลลัพธ์ที่ได้จากขั้นตอนแรก

นำภาพไบนารีที่ได้จากขั้นตอนแรกมาทำการปรับปรุง โดยใช้ค่าความกว้างของเส้นตัว
 อักษรเป็นเกณฑ์ในการพิจารณาคุณสมบัติของจุดภาพ ซึ่งการทำงานในขั้นตอนนี้แสดงดัง
 อัลกอริทึมดังต่อไปนี้

Begin

if ($b_1(x,y) = 1$) *then*

 สืบหา d_R, d_L, d_U, d_D แสดงคังภาพที่ 32

 ค่าความกว้างโดยประมาณ (*Width*) เท่ากับ ค่าที่น้อยที่สุดระหว่างค่า $d_U + d_D$ กับ ค่า $d_R + d_L$

if ($(Width < W)$ หรือ $(Width > W)$) *then*

if ($Width < W$) *then* $b_1(x,y) = 0$

else

 สืบหาค่า $d_{UL}, d_{DR}, d_{DL}, d_{UR}$ แสดงคังภาพที่ 32

 ค่าความกว้างโดยประมาณ (*Width*) เท่ากับ ค่าที่น้อยที่สุดระหว่างค่า $d_{UL} + d_{DR}$ กับ

 ค่า $d_{DL} + d_{UR}$

if ($(Width < W)$ หรือ $(Width > W)$) *then* $b_1(x,y) = 0$

End

ขั้นตอนที่ 3 นิยามภาพพื้นหลัง

ทำการประมาณค่าระดับเข้มที่แสดงคุณสมบัติของพื้นหลังในแต่ละจุดภาพของภาพ $b_1(x,y)$ โดยใช้สมการการทำนายเชิงเส้น (*linear interpolation*) ซึ่งการทำงานดังกล่าวแสดงในรูปของอัลกอริธึมดังต่อไปนี้

Begin

if ($b_1(x,y) = 1$) then

สืบหาค่า $d_U, d_D, d_R, d_L, d_{UL}, d_{DR}, d_{DL}, d_{UR}$

if (not ($d_R > W$ หรือ $d_L > W$) และ ($d_U > W$ หรือ $d_D > W$)) then

if ($d_R > W$ หรือ $d_L > W$) then

$$f_b(x,y) = \frac{g(x,y-d_U)(1/d_U) + g(x,y+d_D)(1/d_D)}{(1/d_U) + (1/d_D)}$$

else

if ($d_U > W$ หรือ $d_D > W$) then

$$f_b(x,y) = \frac{g(x+d_R,y)(1/d_R) + g(x-d_L,y)(1/d_L)}{(1/d_R) + (1/d_L)}$$

else

$$f_b(x,y) = \frac{g(x,y-d_U)(1/d_U) + g(x,y+d_D)(1/d_D) + g(x+d_R,y)(1/d_R) + g(x-d_L,y)(1/d_L)}{(1/d_U) + (1/d_D) + (1/d_R) + (1/d_L)}$$

else

if ($d_{UL} > W$ หรือ $d_{DR} > W$) then

$$f_b(x,y) = \frac{g(x+d_{UR},y-d_{UR})(1/d_{UR}) + g(x-d_{DL},y+d_{DL})(1/d_{DL})}{(1/d_{UR}) + (1/d_{DL})}$$

else

if ($d_{UR} > W$ หรือ $d_{DL} > W$) then

$$f_b(x,y) = \frac{g(x-d_{UL},y-d_{UL})(1/d_{UL}) + g(x+d_{DR},y+d_{DR})(1/d_{DR})}{(1/d_{UL}) + (1/d_{DR})}$$

else

$$f_b(x,y) = \frac{g(x+d_{UR},y-d_{UR})(1/d_{UR}) + g(x-d_{DL},y+d_{DL})(1/d_{DL}) + g(x-d_{UL},y-d_{UL})(1/d_{UL}) + g(x+d_{DR},y+d_{DR})(1/d_{DR})}{(1/d_{UR}) + (1/d_{DL}) + (1/d_{UL}) + (1/d_{DR})}$$

else

$$f_b(x,y) = g(x,y)$$

End

ขั้นตอนที่ 4 การประเมินผลต่างของภาพ

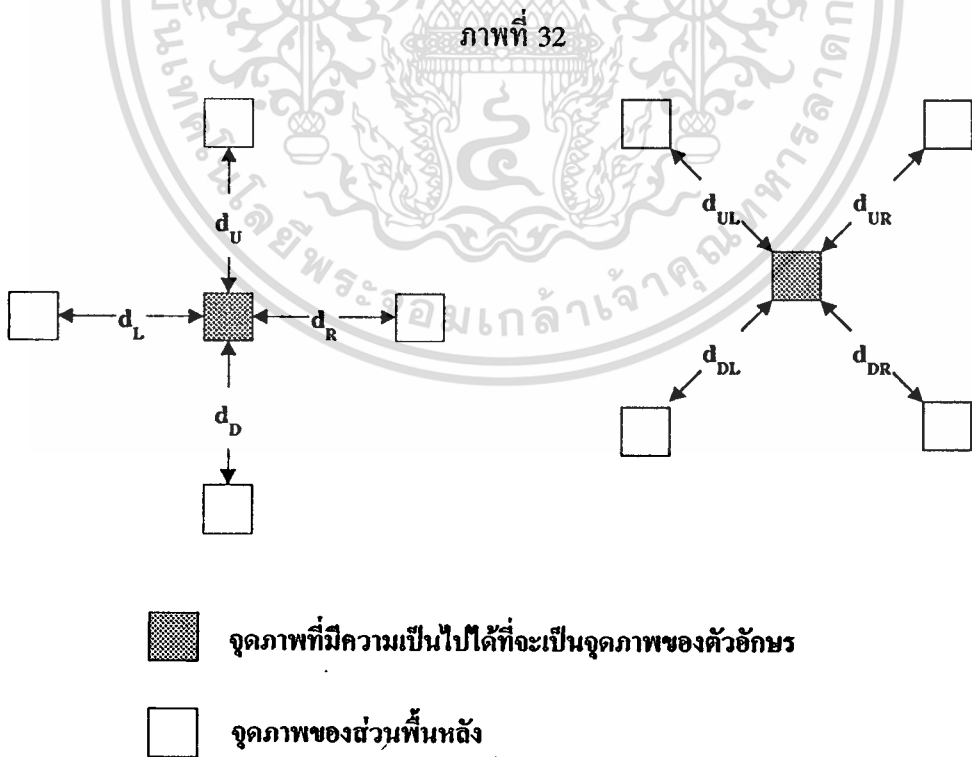
ทำการตัดสินใจสร้างภาพผลลัพธ์ โดยการประเมินผลต่างระหว่างภาพพื้นหลัง $f_b(x,y)$ กับภาพเอกสารเริ่มต้น $g(x,y)$ ด้วยหลักการต่อไปนี้

```

Begin
  if ( $f_b(x,y) - g(x,y) \geq t$ ) then  $b(x,y) = 1$ 
  else  $b(x,y) = 0$ 
End
    
```

โดยที่ t คือ ค่าผลต่างของระดับความเข้มที่มีความเป็นไปได้ระหว่างตัวอักษรกับพื้นหลัง

จุดภาพของภาพไบนารีที่นิยาม 1 เป็นจุดภาพของตัวอักษร จุดภาพที่นิยามด้วย 0 เป็นจุดภาพของส่วนพื้นหลัง



แสดงตำแหน่งของจุดภาพที่ใช้ในขั้นตอนการปรับปรุงภาพไบนารี $b_1(x,y)$ และขั้นตอนเอกสารนี้ การประมาณค่าระดับความเข้มที่มีคุณสมบัติของส่วนพื้นหลังที่พิกัด (x,y) ของภาพ $b_1(x,y)$ ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแยกภาพตัวอักษรโดยการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของ จุดภาพในแนวราสเตอร์สแกน

อัลกอริทึมการแยกภาพตัวอักษรในรูปแบบที่จะกล่าวต่อไปนี้เป็นอัลกอริทึมอีกรูปแบบหนึ่งที่อยู่บนพื้นฐานของการเซกเมนต์ภาพโดยการทำเรดโซลด์แบบปรับค่า (Adaptive Thresholding) เหมาะสำหรับภาพเอกสารที่มีความไม่สม่ำเสมอเกิดขึ้นในส่วนของพื้นหลัง หรือเกิดขึ้นในส่วนของตัวอักษร หรือเกิดขึ้นทั้งสองส่วน โดยค่าเรดโซลด์ที่ใช้ในแต่ละพิกัดจุดภาพจะถูกปรับเปลี่ยนค่าให้มีความเหมาะสมกับแต่ละพิกัดจุดภาพนั้น

หลักการที่นำเสนอ

หลักการที่นำเสนอในอัลกอริทึมรูปแบบนี้ คือ การคำนวณหาค่าเรดโซลด์ใหม่ทุก ๆ พิกัดจุดภาพ โดยปกติแล้วในภาพเอกสารระดับความเข้ม จุดภาพที่มีคุณสมบัติของตัวอักษรจะมีค่าเฉลี่ยระดับความเข้มน้อยกว่าจุดภาพที่มีคุณสมบัติของพื้นหลัง และจุดภาพที่มีคุณสมบัติของตัวอักษรนั้นจะมีค่าระดับความเข้มน้อยกว่าค่าเฉลี่ยระดับความเข้มของจุดภาพที่เฉลี่ยภายในบล็อกที่มีขนาดเหมาะสม

$$\text{Character คือ } g(x,y) < \overline{g(x,y)}$$

ดังนั้นค่าเรดโซลด์ที่เหมาะสมควรมีหลักการที่แปรผันตามหลักการที่กล่าวข้างต้น คือ ค่าเรดโซลด์ที่ใช้ในแต่ละพิกัดจุดภาพจะพิจารณาจากค่าเฉลี่ยระดับความเข้มของจุดภาพนั้นเป็นหลัก แต่ในทางปฏิบัติพบว่าความไม่สม่ำเสมอของระดับความเข้มพื้นหลังอันเนื่องมาจากสัญญาณรบกวนภาพประเภท *White Noise* ทำให้จุดภาพที่มีคุณสมบัติพื้นหลังบางจุดภาพมีระดับความเข้มน้อยกว่าค่าเฉลี่ยระดับความเข้มของจุดภาพนั้น แต่น้อยกว่าไม่มาก ดังนั้นจุดภาพที่จะถูกกำหนดให้เป็นจุดภาพของตัวอักษรจะต้องมีค่าระดับความเข้มน้อยกว่าค่าเฉลี่ยระดับความเข้มในสัดส่วนที่กำหนด มิฉะนั้นจุดภาพนั้นจะถูกกำหนดให้เป็นจุดภาพของพื้นหลัง

$$\text{Character คือ } g(x,y) - \overline{g(x,y)} < t$$

$$\text{หรือ } g(x,y) < k * \overline{g(x,y)}$$

สำหรับการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่ต้องอ้างอิงกับจุดภาพรอบข้าง นั้น ในอัลกอริทึมนี้จะคำนวณโดยการอ้างอิงระดับความเข้มของจุดภาพรอบข้างในแนว 1 มิติ คือ คำนวณหาในแนวราสเตอร์สแกนดังภาพที่ 33 และรูปแบบการทำงานของอัลกอริทึมดังกล่าวนี้ สามารถแสดงในรูปของแผนภาพดังภาพที่ 34

ภาพที่ 33



แสดงแนวทางการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพ ที่อ้างอิงระดับความเข้มของจุดภาพรอบข้างในแนว 1 มิติ

วิธีการที่นำเสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการประมวลผลออกเป็น ขั้นตอนย่อย ๆ ดังต่อไปนี้

ขั้นตอนที่ 1 การคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพ

1.1 กำหนดขนาดของ S (คือ จำนวนจุดภาพที่ใช้สำหรับการหาค่าเฉลี่ย) ซึ่งค่า S ที่มีความเหมาะสมจะมีค่าโดยประมาณเท่ากับ $2*W$ โดยที่ W คือ ขนาดความกว้างของเส้นตัวอักษร

1.2 ทำการประมวลผลข้อมูลภาพ โดยการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่พิกัด n โดยให้ P_n คือ ค่าระดับความเข้มของจุดภาพที่พิกัด n ซึ่งอัลกอริทึมรูปแบบนี้จะมองจุดภาพของภาพเอกสารทั้งภาพในลักษณะที่เป็นจุดภาพแถวเดียวกัน (ดังภาพที่ 35) ซึ่งขั้นตอนการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพจะอ้างอิงระดับความเข้มของจุดภาพในพิกัดก่อนหน้านี้และระดับความเข้มของจุดภาพในพิกัดถัดไปดังสมการต่อไปนี้

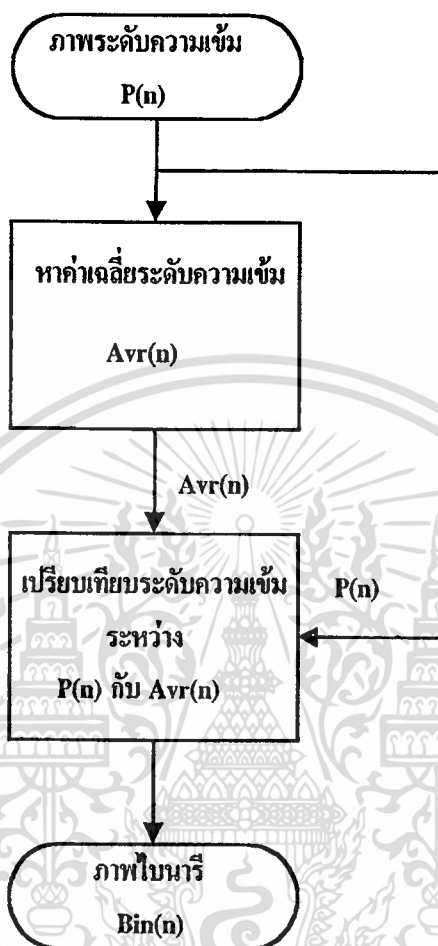
ให้ $f_s(n)$ เป็นค่าผลบวกของระดับความเข้มจุดภาพที่พิกัด n โดย $f_s(n)$ จะมีค่าเท่ากับ

$$f_s(n) = \sum_{i=0}^{s-1} p_{n+\frac{s}{2}-i} \quad (3.13)$$

ดังนั้นค่าเฉลี่ยระดับความเข้มของจุดภาพที่พิกัด n จะเท่ากับ

$$Avr(n) = \frac{f_s(n)}{s} \quad (3.14)$$

ภาพที่ 34



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเซด โซลด์
แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพในแนวราสเตอร์สแกน

ขั้นตอนที่ 2 การทำเซด โซลด์ในแต่ละจุดภาพ

ทำการพิจารณาคูณสมบัติของจุดภาพที่พิกัด m ว่าเป็นจุดภาพของตัวอักษรหรือส่วนของพื้นหลัง โดยการเปรียบเทียบค่าระดับความเข้มของจุดภาพกับค่าเฉลี่ยระดับความเข้ม และนำการถ่วงน้ำหนักค่ามาประยุกต์ใช้กับรูปแบบการเปรียบเทียบ เพราะถ้าปราศจากการถ่วงน้ำหนักค่าแล้วการทำเซด โซลด์ในแต่ละจุดภาพอาจให้ผลผิดพลาดได้โดยอันเนื่องมาจากสิ่งที่ไม่พึงประสงค์ในส่วนของพื้นหลังที่มีรูปแบบที่แน่นอน หรือสัญญาณรบกวนภาพ

$$b(n) = \begin{cases} 1 & \text{if } p_n < k * Avr(n) \\ 0 & \text{otherwise} \end{cases} \tag{3.15}$$

จุดภาพของภาพไบนารีที่นิยาม 1 เป็นจุดภาพของตัวอักษรหรือสัญลักษณ์กราฟฟิก จุดภาพที่นิยามด้วย 0 เป็นจุดภาพของส่วนพื้นหลัง

ภาพที่ 35



แสดงพิกัดจุดภาพของภาพเอกสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแยกภาพตัวอักษรโดยอัลกอริธึมการทำเซดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพ

อัลกอริธึมสำหรับการแยกภาพตัวอักษรที่จะนำเสนอในรูปแบบต่อไปนี้จะใช้หลักการเดียวกับอัลกอริธึมการทำเซดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพในแนวราสเตอร์สแกน แต่สิ่งที่แตกต่างระหว่างอัลกอริธึมทั้งสอง คือ รูปแบบการคำนวณหาค่าเฉลี่ยของระดับความเข้ม ดังนั้นอัลกอริธึมการแยกภาพตัวอักษรในรูปแบบที่จะนำเสนอต่อไปนี้อยู่บนพื้นฐานของการเซกเมนต์ภาพโดยการทำเซดโซลด์แบบปรับค่า (Adaptive Thresholding)

หลักการที่นำเสนอ

จุดภาพที่จะถูกกำหนดให้เป็นจุดภาพของตัวอักษรจะต้องมีค่าระดับความเข้มน้อยกว่าค่าเฉลี่ยระดับความเข้มในสัดส่วนที่กำหนด มิฉะนั้นจุดภาพนั้นจะถูกกำหนดให้เป็นจุดภาพของพื้นหลัง

$$\text{Character คือ } g(x,y) - \overline{g(x,y)} < t$$

$$\text{หรือ } g(x,y) < k * \overline{g(x,y)}$$

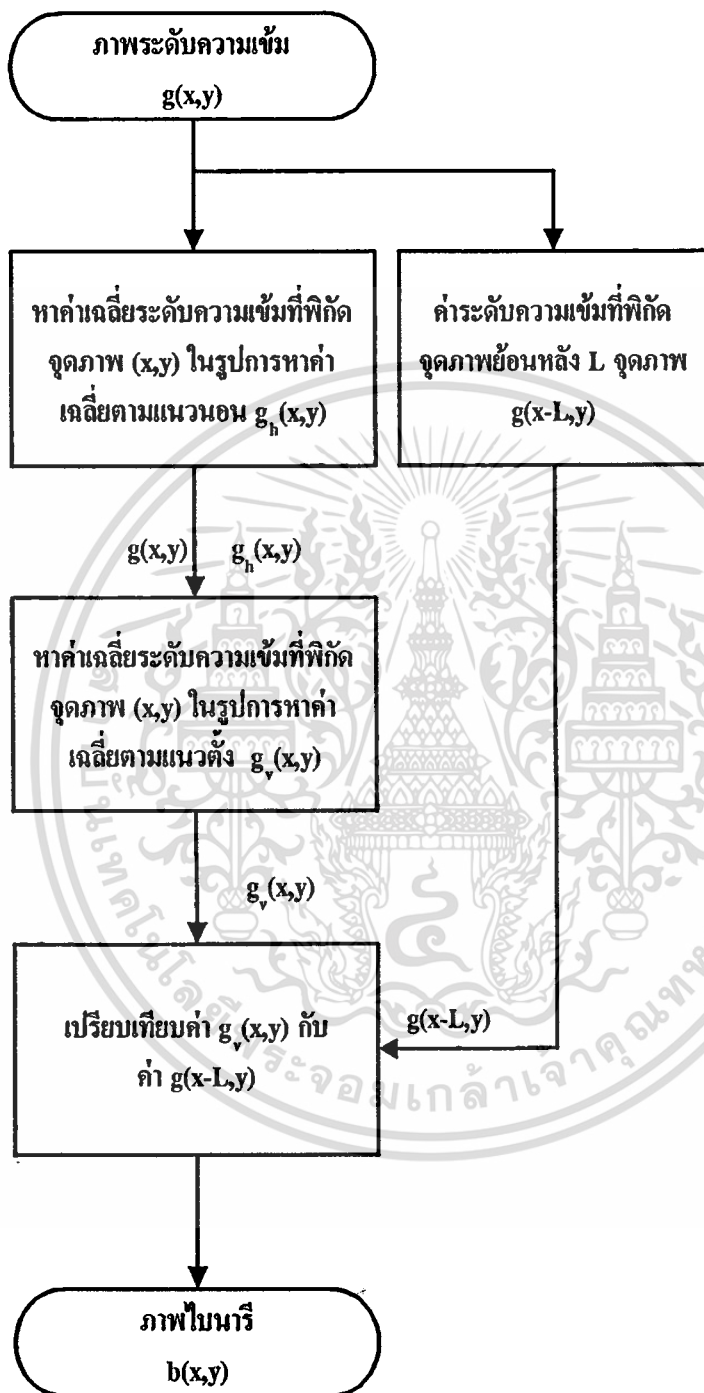
สำหรับการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่ต้องอ้างอิงกับจุดภาพรอบข้างนั้น ในอัลกอริธึมนี้จะคำนวณหาจากข้อมูลที่ผ่านมาแล้วใน 2 มิติ คือ คำนวณหาจากค่าเฉลี่ยระดับความเข้มในแนวนอนและค่าเฉลี่ยระดับความเข้มในแนวตั้งดังภาพที่ 36 และคุณสมบัติที่สำคัญของค่าเฉลี่ยระดับความเข้มที่คำนวณได้ในอัลกอริธึมนี้ คือ ค่าเฉลี่ยจะมีการปรับค่าอย่างรวดเร็วและเหมาะสมเมื่อจุดภาพออกจากบริเวณเส้นตัวอักษรที่มีระดับความเข้มต่ำ เพื่อป้องกันตัวอักษรในตำแหน่งถัดไปที่มีระดับความเข้มสูงกว่าถูกกำจัดทิ้งในภาพผลลัพธ์ สำหรับขั้นตอนการทำงานของอัลกอริธึมดังกล่าวสามารถแสดงในรูปของแผนภาพดังภาพที่ 37 ต่อไปนี้

ภาพที่ 36



แสดงแนวทางการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพ
ที่คำนวณหาจากข้อมูลที่ผ่านมาแล้วใน 2 มิติ

ภาพที่ 37



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเซรโด
แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพ

วิธีการที่นำเสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการทำงานออกเป็นขั้นตอนย่อย ๆ ดังต่อไปนี้

ขั้นตอนที่ 1 การคำนวณหาค่าเฉลี่ยระดับความเข้มแต่ละจุดภาพ

นำภาพเอกสารเริ่มต้นมาทำการประมวลผลโดยการคำนวณหาค่าเฉลี่ยระดับความเข้มในแต่ละพิกัดจุดภาพ ซึ่งการคำนวณหาค่าเฉลี่ยระดับความเข้มนั้นจะอยู่ในรูปของการหาค่าเฉลี่ยระดับความเข้มของจุดภาพในแนวนอนและแนวตั้งดังสมการเชิงเส้นสองสมการต่อไปนี้

$$g_h(x,y) = g_h(x-1,y) + W_h[g(x,y) - g_h(x-1,y)] \quad (3.16)$$

$$g_v(x,y) = g_v(x,y-1) + W_v[g_h(x,y) - g_v(x,y-1)] \quad (3.17)$$

โดยที่

$g_h(x,y)$ คือ ค่าเฉลี่ยระดับความเข้มในแนวนอน

$g_v(x,y)$ คือ ค่าเฉลี่ยระดับความเข้มในแนวตั้ง และใช้ค่า $g_v(x,y)$ เป็นค่าเฉลี่ยระดับความเข้มจุดภาพ

$$\overline{g(x,y)} = g_v(x,y) \quad (3.18)$$

W_h, W_v คือ ฟังก์ชันที่ใช้สำหรับการปรับแนวโน้มของค่าเฉลี่ยระดับความเข้ม ซึ่งค่าผลลัพธ์ที่ได้จากฟังก์ชันทั้งสองนั้นจะมีค่าเป็นศูนย์เสมอเมื่ออาร์กิวเมนต์ของฟังก์ชันมีค่าเป็นศูนย์ และค่าผลลัพธ์ที่ได้จากฟังก์ชันทั้งสองจะมีค่าอยู่ในช่วงระหว่างค่าศูนย์กับค่าอาร์กิวเมนต์ที่เข้าไป อัลกอริทึมนี้ได้กำหนดแนวทางของค่าผลลัพธ์ที่จะได้จากฟังก์ชันทั้งสองให้มีลักษณะดังภาพที่ 38 (ก) และ (ข) ตามลำดับ เมื่อมีค่าอาร์กิวเมนต์ของฟังก์ชันตามที่กำหนด

จุดภาพที่พิกัดเริ่มต้น $(0,0)$ กำหนดให้ค่า $g_h(x,y) = g(x,y)$ และค่า $g_v(x,y) = g(x,y)$

ขั้นตอนที่ 2 การทำเรคโวลต์กับแต่ละจุดภาพ

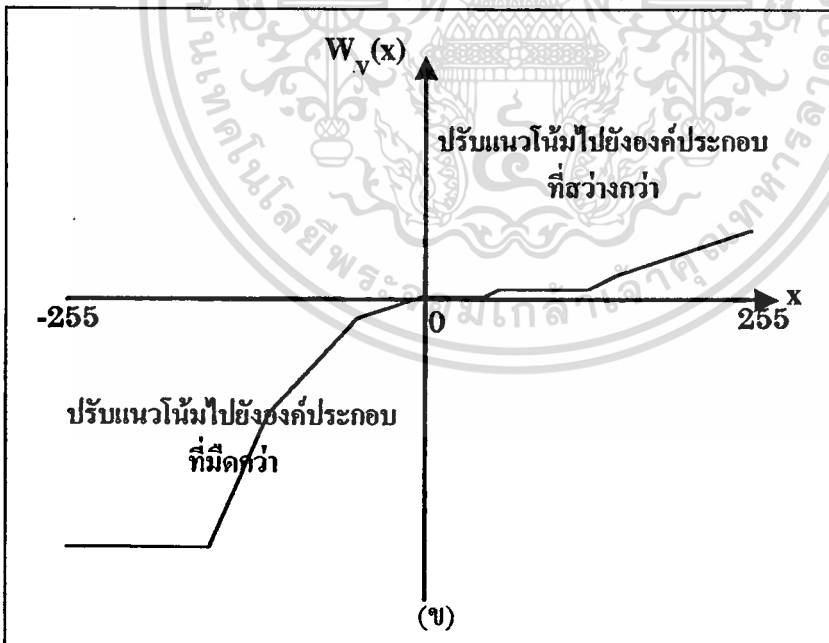
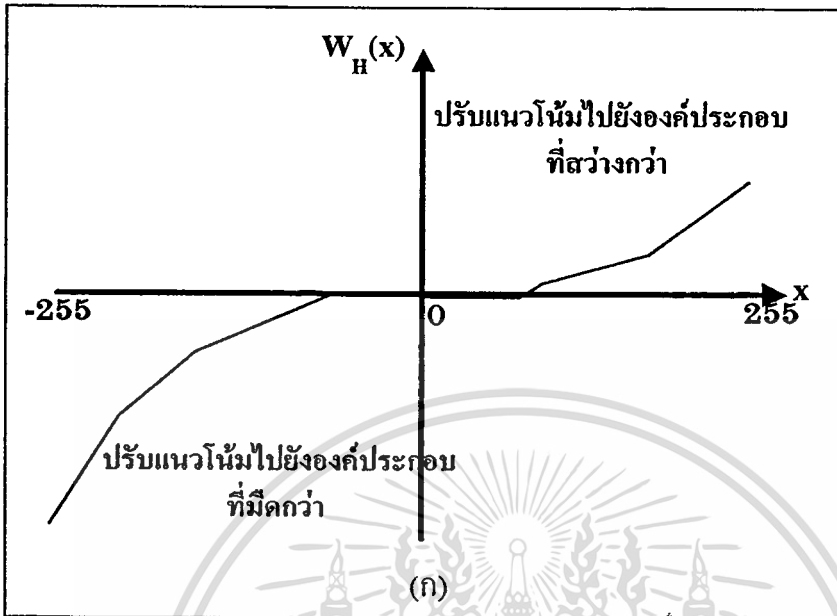
ทำการเปรียบเทียบค่าระดับความเข้มในแต่ละจุดภาพกับค่าเฉลี่ยระดับความเข้ม $g_v(x,y)$ แต่จากสมการเชิงเส้นที่ (3.16) และ (3.17) พบว่าค่าเฉลี่ยที่คำนวณได้เป็นค่าที่คำนวณมาจากทิศทางเดียว คือ เฉลี่ยจากจุดภาพที่ผ่านมาแล้วและเป็นการใช้ข้อมูลของจุดภาพในพิกัดก่อนหน้านี้มาคำนวณหาค่าเฉลี่ยระดับความเข้ม ดังนั้นอัลกอริทึมนี้จึงปรับรูปแบบการเปรียบเทียบโดยใช้ระดับความเข้มของจุดภาพในพิกัดย้อนหลัง L พิกัด $g(x-L,y)$ มาเปรียบเทียบกับค่าเรคโวลต์ $g_v(x,y)$ สำหรับค่า L ที่มีความเหมาะสมจะประมาณเท่ากับค่าความกว้างของเส้นตัวอักษร และใช้การถ่วงน้ำหนักค่ามาประยุกต์ใช้กับรูปแบบการเปรียบเทียบ เพราะถ้าปราศจากการถ่วงน้ำหนักค่าแล้วการทำเรคโวลต์ในแต่ละจุดภาพอาจให้ผลผิดพลาดได้โดยอันเนื่องมาจากสิ่งที่ไม่พึงประสงค์ในส่วนของพื้นหลังที่มีรูปแบบที่แน่นอน

$$b(x,y) = \begin{cases} 1 & \text{if } k * g(x-L,y) > g_v(x,y) \\ 0 & \text{otherwise} \end{cases} \quad (3.18)$$

k คือ ค่าคงที่ที่ใช้ในการถ่วงน้ำหนัก

จุดภาพของภาพไบนารีที่นิยาม 1 เป็นจุดภาพของตัวอักษร จุดภาพที่นิยามด้วย 0 เป็นจุดภาพของพื้นหลัง

ภาพที่ 38



แสดงแนวทางของค่าผลลัพธ์ที่ได้จากฟังก์ชันการปรับค่า (ก) W_h (ข) W_v

การแยกภาพตัวอักษรโดยอัลกอริธึมการทำเซรคโฮลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ

อัลกอริธึมสำหรับการแยกภาพตัวอักษรที่จะนำเสนอในรูปแบบต่อไปนี้จะใช้หลักการเดียวกับอัลกอริธึมที่ 5 และอัลกอริธึมที่ 6 แต่สิ่งที่แตกต่างไปจากสองอัลกอริธึมดังกล่าว คือ รูปแบบการคำนวณหาค่าเฉลี่ยของระดับความเข้ม และการนำค่าความแปรปรวนของระดับความเข้มจุดภาพมาประยุกต์ใช้ ดังนั้นอัลกอริธึมการแยกภาพตัวอักษรในรูปแบบที่จะนำเสนอต่อไปนี้จึงอยู่บนพื้นฐานของการเซกเมนต์ภาพโดยการทำเซรคโฮลด์แบบปรับค่า (Adaptive Thresholding)

หลักการที่นำเสนอ

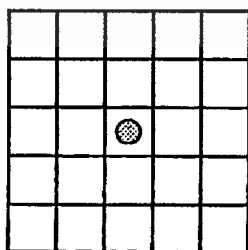
จุดภาพที่จะถูกกำหนดให้เป็นจุดภาพของตัวอักษรจะต้องมีค่าระดับความเข้มน้อยกว่าค่าเฉลี่ยระดับความเข้มในสัดส่วนที่กำหนด มิฉะนั้นจุดภาพนั้นจะถูกกำหนดให้เป็นจุดภาพของพื้นหลัง

$$\text{Character คือ } g(x,y) - \overline{g(x,y)} < t$$

$$\text{หรือ } g(x,y) < k * \overline{g(x,y)}$$

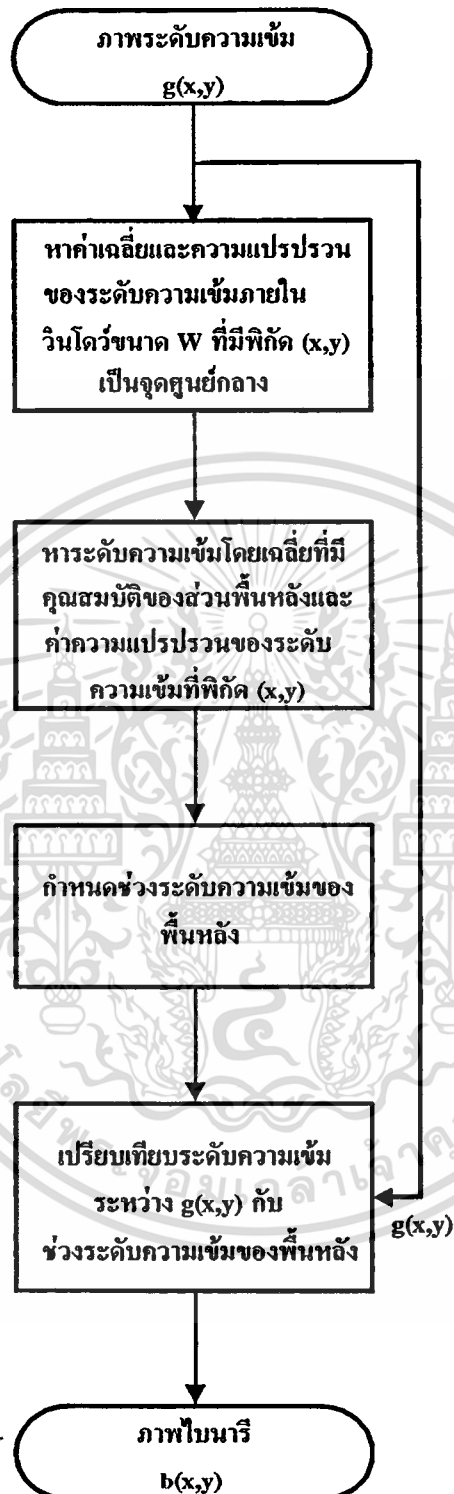
สำหรับการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่ต้องอ้างอิงกับจุดภาพรอบข้างนั้น ในอัลกอริธึมนี้จะคำนวณหาจากข้อมูล 2 มิติในบล็อกรจขจร ดังภาพที่ 39 และอัลกอริธึมนี้ได้้นำค่าความแปรปรวนของจุดภาพมาประยุกต์ใช้ ซึ่งรายละเอียดจะกล่าวถึงในส่วนต่อไป สำหรับขั้นตอนการทำงานของอัลกอริธึมดังกล่าวสามารถแสดงในรูปแบบของแผนภาพดังภาพที่ 40

ภาพที่ 39



แสดงแนวทางการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่คำนวณหาจากข้อมูล 2 มิติในบล็อกรจขจร

ภาพที่ 40



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริธึมการทำเรดโซลต์
แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนระดับความเข้มของจุดภาพ

วิธีการที่นำเสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการทำงาน ออกเป็นขั้นตอนย่อย ๆ ดังต่อไปนี้

ขั้นตอนที่ 1 การคำนวณหาค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มที่พิกัดจุดภาพ (x,y)

1.1 เลือกขนาดของวินโดว์ (บล็อกจตุรัส) ที่มีความเหมาะสมกับข้อมูลภาพเอกสาร ระดับความเข้มเริ่มต้น คือ W^2 โดยที่ $W = 2w + 1$

1.2 คำนวณหาค่าเฉลี่ย (μ_w) และค่าความแปรปรวน (σ_w^2) ของระดับความเข้มบนวินโดว์ ที่จุดศูนย์กลางของวินโดว์อยู่ที่พิกัดจุดภาพ (x,y)

$$\mu_w = \frac{1}{W^2} \sum_{\substack{x-w \leq i \leq x+w \\ y-w \leq j \leq y+w}} g(i,j) \quad (3.19)$$

$$\sigma_w^2 = \frac{1}{(W^2 + 1)} \sum_{\substack{x-w \leq i \leq x+w \\ y-w \leq j \leq y+w}} (g(i,j) - \mu_w)^2 \quad (3.20)$$

1.3 ทำการเฉลี่ยข้อมูลของจุดภาพ (μ_b, σ_b^2) ในพิกัดปัจจุบัน กับข้อมูลของจุดภาพในพิกัดก่อนหน้า

$$\mu_b = \frac{\mu_w + \mu_b}{2} \quad (3.21)$$

$$\sigma_b^2 = \frac{\sigma_w^2 + \sigma_b^2}{2} \quad (3.22)$$

จุดภาพที่พิกัด $(0,0)$ กำหนดให้ $\mu_b = \mu_w$ และ $\sigma_b^2 = \sigma_w^2$

ขั้นตอนที่ 2 การคำนวณหาเรดโซลด์ที่พิกัดจุดภาพ (x,y)

โดยค่าเรดโซลด์ของจุดภาพจะอ้างอิงค่าเฉลี่ยระดับความเข้มของจุดภาพเป็นหลัก และนำค่าความแปรปรวนของจุดภาพมาประยุกต์ใช้ โดยค่าเรดโซลด์ที่พิกัดจุดภาพ (x,y) เท่ากับ

$$Thr(x,y) = \mu_b - k * \sigma_b \quad (3.23)$$

ขั้นตอนที่ 3 การทำเซรคโซลด์ที่พิกัดจุดภาพ (x,y)

$$b(x,y) = \begin{cases} 1 & \text{if } g(x,y) < Thr(x,y) \\ 0 & \text{otherwise} \end{cases} \quad (3.24)$$

จุดภาพของภาพไบนารีที่นิยาม 1 เป็นจุดภาพของตัวอักษรหรือสัญลักษณ์กราฟฟิก จุดภาพที่นิยามด้วย 0 เป็นจุดภาพของพื้นหลัง



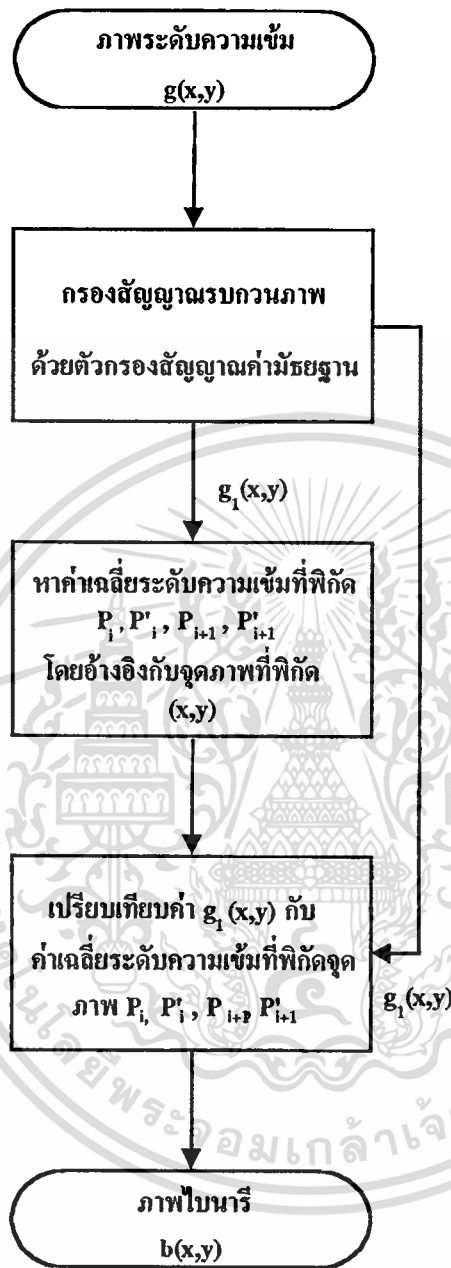
การแยกภาพตัวอักษรโดยอัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง

อัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง เป็นอัลกอริธึมที่นำหลักการของอัลกอริธึมการทำเรด โซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มมาประยุกต์เพื่อใช้สำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้นที่มีลักษณะของความซับซ้อนเกิดขึ้นในส่วนของพื้นหลัง

หลักการที่นำเสนอ

หลักการที่นำเสนอในอัลกอริธึมรูปแบบนี้ คือ พิจารณาคุณสมบัติของจุดภาพ โดยการประเมินผลการเปรียบเทียบระหว่างค่าระดับความเข้มของจุดภาพ $g(x,y)$ กับค่าเฉลี่ยระดับความเข้มของจุดภาพที่พิกัดรอบข้าง คือ ที่พิกัด $P_i, P'_i, P_{i+1}, P'_{i+1}$ (แสดงดังภาพที่ 42) ซึ่งระยะห่างระหว่างจุดภาพที่พิกัด (x,y) ถึงกับจุดภาพที่พิกัดรอบข้าง $P_i, P'_i, P_{i+1}, P'_{i+1}$ จะประมาณเท่ากับขนาดความกว้างของเส้นตัวอักษร ถ้าระดับความเข้มของจุดภาพ $g(x,y)$ น้อยกว่าค่าเฉลี่ยระดับความเข้มทั้ง 4 ค่า อย่างน้อย r ระดับความเข้ม เราจะนิยามให้จุดภาพดังกล่าวมีคุณสมบัติของตัวอักษร มิฉะนั้นแล้วจุดภาพดังกล่าวจะมีคุณสมบัติของพื้นหลัง แต่หลักการที่นำเสนอข้างต้นไม่เหมาะที่จะนำไปใช้ประมวลผลกับภาพเอกสารเริ่มต้นที่มีสิ่งไม่พึงประสงค์ขนาดเล็ก ๆ เกิดบนส่วนพื้นหลังของภาพ และสิ่งที่ไม่พึงประสงค์เหล่านี้มีระดับความเข้มใกล้เคียงหรือระดับเดียวกับกับระดับความเข้มของตัวอักษร ซึ่งข้อจำกัดที่เกิดขึ้นดังกล่าวสามารถแก้ไขได้โดยการนำภาพเอกสารระดับความเข้มเริ่มต้นมาผ่านขั้นตอนการประมวลผลก่อน (Image Preprocessing) ในรูปของการกรองสัญญาณรบกวน ก่อนที่จะนำภาพนี้ไปใช้ประมวลผลตามหลักการที่กล่าวข้างต้นสำหรับรูปแบบการทำงานของอัลกอริธึมดังกล่าวข้างต้นสามารถแสดงในรูปแบบแผนภาพดังภาพที่

ภาพที่ 41



แผนภาพแสดงขั้นตอนการทำงานของอัลกอริทึมการประเมิน
ค่าเฉลี่ยระดับความเข้มของจุดภาพในพิกัดรอบข้าง

วิธีการที่นำเสนอ

การแยกภาพตัวอักษรด้วยหลักการข้างต้น สามารถแบ่งขั้นตอนการประมวลผลออกเป็นขั้นตอนย่อย ๆ ดังต่อไปนี้

ขั้นตอนที่ 1 การกรองสัญญาณรบกวนภาพ

นำภาพเอกสารระดับความเข้มเริ่มต้นมากรองสัญญาณรบกวนด้วยตัวกรองสัญญาณค่ามัธยฐาน ขนาด 3×3 เพื่อใช้กำจัดสัญญาณรบกวนหรือสิ่งที่ไม่พึงประสงค์ในส่วนของพื้นหลังที่มีขนาดเล็ก ๆ แต่มีระดับความเข้มใกล้เคียงกับตัวอักษร

ขั้นตอนที่ 2 กำหนดค่าเฉลี่ยระดับความเข้มที่พิกัดรอบข้าง

กำหนดค่าเฉลี่ยระดับความเข้มที่พิกัด $P_i, P'_i, P_{i+1}, P'_{i+1}$ ที่อ้างอิงกับพิกัดจุดภาพ (x, y) ดังสมการที่ (3.25) ต่อไปนี้

$$avr(P) = \sum_{-W \leq i \leq W} \sum_{-W \leq j \leq W} g(P_x - i, P_y - j) / (2 * W + 1)^2 \quad (3.25)$$

$$P'_i = P_{(i+4) \bmod 8}, i = 0, \dots, 7 \quad (3.26)$$

โดยที่ P_x และ P_y เป็นโคออดิเนตที่พิกัด P และ W เป็นพารามิเตอร์ซึ่งค่าที่เหมาะสมคือ ค่าความกว้างของเส้นตัวอักษร

ขั้นตอนที่ 3 พิจารณาคุณสมบัติของจุดภาพที่พิกัด (x, y)

ทำการเปรียบเทียบระดับความเข้มของจุดภาพที่พิกัด (x, y) กับค่าเฉลี่ยระดับความเข้มเพื่อทำการตัดสินใจว่าจุดภาพนั้นเป็นจุดภาพของตัวอักษรหรือจุดภาพของส่วนพื้นหลัง ดังสมการต่อไปนี้

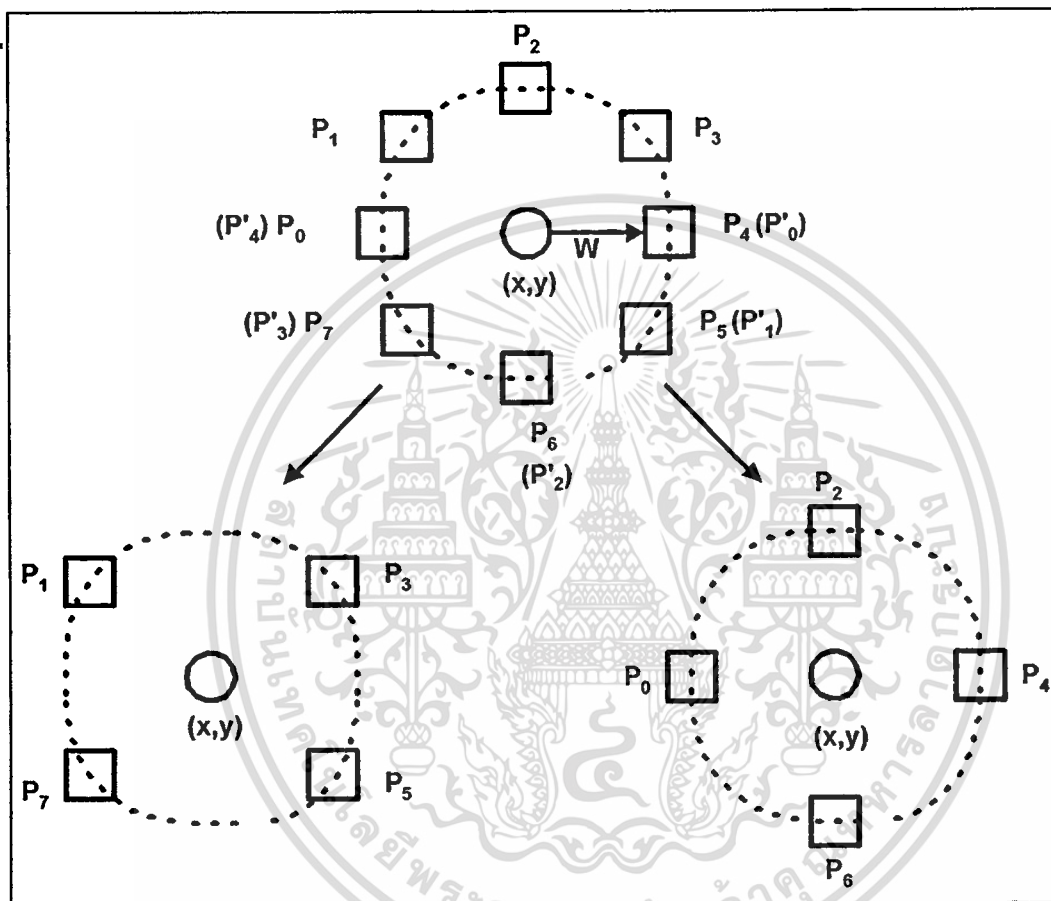
$$L(P) = avr(P) - g(x, y) \geq t \quad (3.27)$$

$$b(x, y) = \begin{cases} 1 & \text{if } \bigvee_{i=0}^3 [L(P_i) \wedge L(P'_i) \wedge L(P_{i+1}) \wedge L(P'_{i+1})] \text{ is true} \\ 0 & \text{otherwise} \end{cases} \quad (3.28)$$

$$\bigvee_{i=0}^3 [L(P_i) \wedge L(P'_i) \wedge L(P_{i+1}) \wedge L(P'_{i+1})] = \underbrace{\bigvee_{0,2} [L(P_i) \wedge L(P'_i)]}_{0.2} \wedge \underbrace{\bigvee_{1,3} [L(P_{i+1}) \wedge L(P'_{i+1})]}_{1.3} \quad (3.29)$$

จุดภาพของภาพไบนารีที่นิยามด้วย 1 เป็นจุดภาพของตัวอักษรหรือสัญลักษณ์กราฟฟิก
จุดภาพที่นิยามด้วย 0 เป็นจุดภาพของส่วนพื้นหลัง

ภาพที่ 42



แสดงพิกัด $P_i, P'_i, P_{i+1}, P'_{i+1}$ ที่ใช้สำหรับหาค่าเฉลี่ยระดับความเข้ม

สรุป

อัลกอริธึมที่ใช้สำหรับการแยกภาพตัวอักษรที่นำเสนอในงานวิจัยบทนี้ แต่ละอัลกอริธึมจะใช้หลักการที่แตกต่างกันออกไป ดังนั้นภาพผลลัพธ์ที่ได้ในแต่ละอัลกอริธึมอาจมีลักษณะที่ใกล้เคียงกัน หรือ แตกต่างกันออกไปขึ้นอยู่กับหลักการที่ใช้ในรูปแบบของอัลกอริธึมนั้น ซึ่งบางอัลกอริธึมใช้ค่าความกว้างของเส้นตัวอักษรเป็นค่าพารามิเตอร์ที่สำคัญในขั้นตอนการแยกภาพตัวอักษร หรือบางอัลกอริธึมใช้ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรกับพื้นหลังเป็นค่าพารามิเตอร์ที่สำคัญในขั้นตอนการแยกภาพตัวอักษร แต่อย่างไรก็ตามรูปแบบของอัลกอริธึมที่นำเสนอเหล่านี้ก็จะอยู่บนพื้นฐานหลักของการทำเซกเมนต์ภาพ โดยการทำให้ขอบเขตของจุดภาพในสองลักษณะ คือ การทำเซกเมนต์แบบครอบคลุม และการทำเซกเมนต์แบบปรับค่า



บทที่ 4

ผลการทดลอง

ในวิทยานิพนธ์นี้ ได้ทำการเสนอขั้นตอนการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้นที่มีความหลากหลายของส่วนพื้นหลัง โดยใช้อัลกอริธึมการแยกภาพตัวอักษรในรูปแบบต่าง ๆ แต่รูปแบบก็จะมีหลักการที่แตกต่างกันออกไป ดังได้อธิบายในบทที่ผ่านมา และในบทนี้จะนำเสนอตัวอย่างของการทดสอบประสิทธิภาพของอัลกอริธึมการแยกภาพตัวอักษร โดยนำอัลกอริธึมการแยกภาพตัวอักษรทั้ง 8 อัลกอริธึม มาประมวลผลกับข้อมูลภาพเอกสารตัวอย่าง 2 ภาพ คือ ภาพ Document 1 และภาพ Document 2 (แสดงดังภาพที่ 43 (ก) และ (ข)) โดยภาพ Document 1 เป็นภาพเอกสารที่จำลองขึ้นมาในลักษณะที่มีรอยขีดข่วนและทะเลาะในส่วนพื้นหลังของภาพ ส่วนภาพ Document 2 เป็นภาพเอกสารจริงที่สามารถพบเห็นได้โดยทั่วไป ในกรณีที่มีความหลากหลายของสีพื้นหลังของภาพเอกสาร ภาพทั้งสองถูกสแกนด้วยความละเอียดขนาด 300 dpi ในลักษณะของภาพถ่ายขาวดำโดยมีรายละเอียดของแต่ละจุดภาพขนาด 8 บิต หรือ 256 ระดับเทา

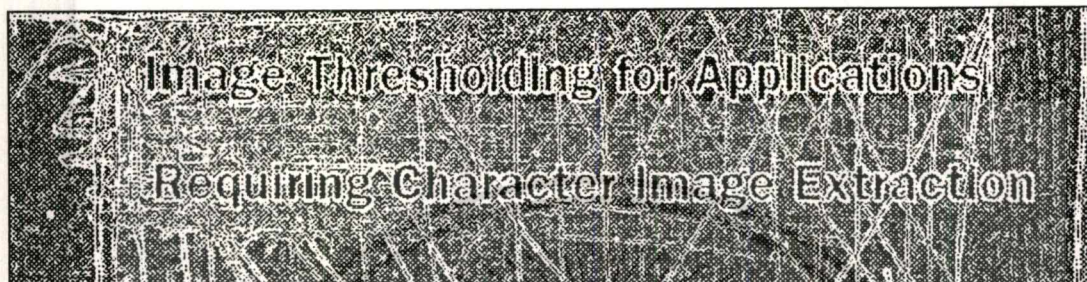
ผลการแยกภาพตัวอักษรโดยอัลกอริธึมฟังก์ชันร่วม

ผลของการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้น โดยใช้อัลกอริธึมฟังก์ชันร่วมจะให้ภาพผลลัพธ์แสดงดังภาพที่ 47 (ก) และ (ข) ตามลำดับ โดยภาพที่ 44 (ก) และ (ข) แสดงภาพโมดูลกราดิเอนท์ที่ได้จากการนำตัวดำเนินการของโซเบลมาทำโอเปอเรชันกับภาพเอกสารระดับความเข้มเริ่มต้น คือ ภาพ Document 1 และ ภาพ Document 2 ตามลำดับ ภาพที่ 45 (ก) และ (ข) แสดงฮิสโตแกรมระดับความเข้มของภาพโมดูลกราดิเอนท์ พร้อมทั้งค่าเรคโวลต์ที่คำนวณได้ โดยนำวิธีการทำเรคโวลต์ตามคุณลักษณะของวัตถุ [2] มาประยุกต์ใช้ในขั้นตอนการคำนวณหาค่าเรคโวลต์ของภาพโมดูลกราดิเอนท์ โดยวัตถุในที่นี้ก็คือ ขอบของตัวอักษร ค่าเรคโวลต์ที่คำนวณได้จากภาพโมดูลกราดิเอนท์ของภาพ Document 1 เท่ากับ 216 โดยค่าเรคโวลต์ค่าแรก r_1 ที่คำนวณได้เท่ากับ 152 และค่าเรคโวลต์ค่าที่สอง r_2 ที่ได้จากการปรับความเหมาะสมของค่าเท่ากับ 216 ซึ่งเป็นค่าเรคโวลต์ค่าสุดท้ายที่ได้จากการปรับแล้วมีค่าไม่เกินเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าระดับความเข้มที่แสดงคุณลักษณะของวัตถุ (ขอบของตัวอักษร) ซึ่งมีค่าเท่ากับ 255 ดังนั้นจึงใช้ค่า t_2 เป็นค่าเรคโพลด์ของภาพโมดูลกราเดียนท์ของภาพ Document 1 ส่วนค่าเรคโพลด์ที่คำนวณได้จากภาพโมดูลกราเดียนท์ของภาพ Document 2 เท่ากับ 206 โดยค่าเรคโพลด์ค่าแรก t_1 ที่คำนวณได้เท่ากับ 124 และค่าเรคโพลด์ค่าที่สอง t_2 ที่ได้จากการปรับความเหมาะสมของค่าเท่ากับ 206 ซึ่งเป็นค่าเรคโพลด์ค่าสุดท้ายที่ได้จากการปรับแล้วมีค่าไม่เกินค่าระดับความเข้มที่แสดงคุณลักษณะของวัตถุ (ขอบของตัวอักษร) ซึ่งมีค่าเท่ากับ 255 ดังนั้นจึงใช้ค่า t_2 เป็นค่าเรคโพลด์ของภาพโมดูลกราเดียนท์ของภาพ Document 2 และการประมวลผลในขั้นตอนการคำนวณหาค่าเรคโพลด์ของภาพโมดูลกราเดียนท์ได้กำหนดค่าของพารามิเตอร์ t (ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างขอบของตัวอักษร) เท่ากับ 30 สำหรับภาพที่ 46 (ก) และ (ข) แสดงภาพ $S(x,y)$ ซึ่งเป็นภาพที่มี 3 สถานะ และเป็นภาพที่ใช้เป็นข้อมูลเข้าสำหรับการประมวลผลในขั้นตอนสุดท้ายในการสร้างภาพผลลัพธ์



ภาพที่ 44



(ก)

เครื่องไมโครคอมพิวเตอร์ประเภทระบบ clock-tripled มีเครื่องเจ้ายี่ห้อ Ascentia 800N และ AST ซึ่งใช้ 486DX4, 76 เมกะเฮิรตซ์ ทำให้มีประสิทธิภาพเหนือกว่า เครื่องที่ใช้โปรเซสเซอร์ความเร็ว 60 เมกะ- เฮิรตซ์อื่นๆ จากการทดสอบแบบมาตรฐาน คะแนนที่ได้มีขยับเป็นของ Pentium TurboCT ของ Toshiba ซึ่งเป็นคะแนนของสถาบันของ แบบทดสอบที่ทราบจนถึง 8 ชั่วโมง 24 นาที เราจึงไปทราบ PCBench 3.0, WinBench 4.0, Winstone 34 Version 1.0 และ ZDigit 3.0 จากห้องปฏิบัติการ ของ PC Magazine เพื่อเปรียบเทียบ ประสิทธิภาพของเครื่องไมโครคอมพิวเตอร์

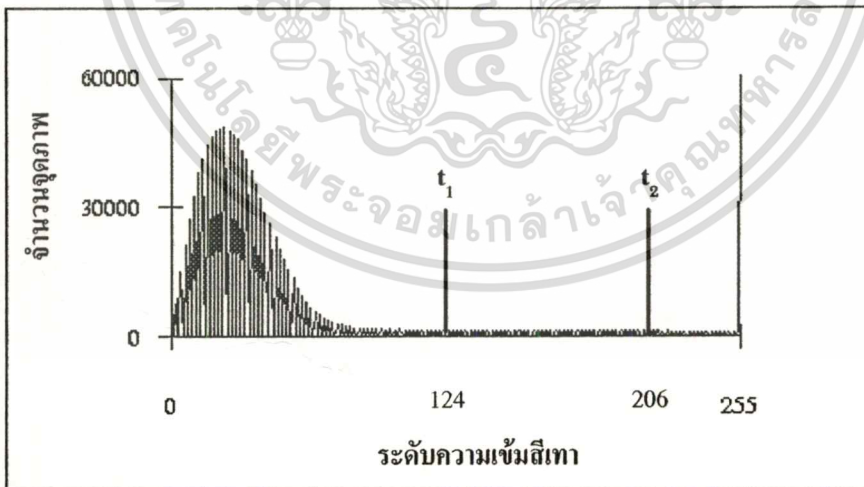
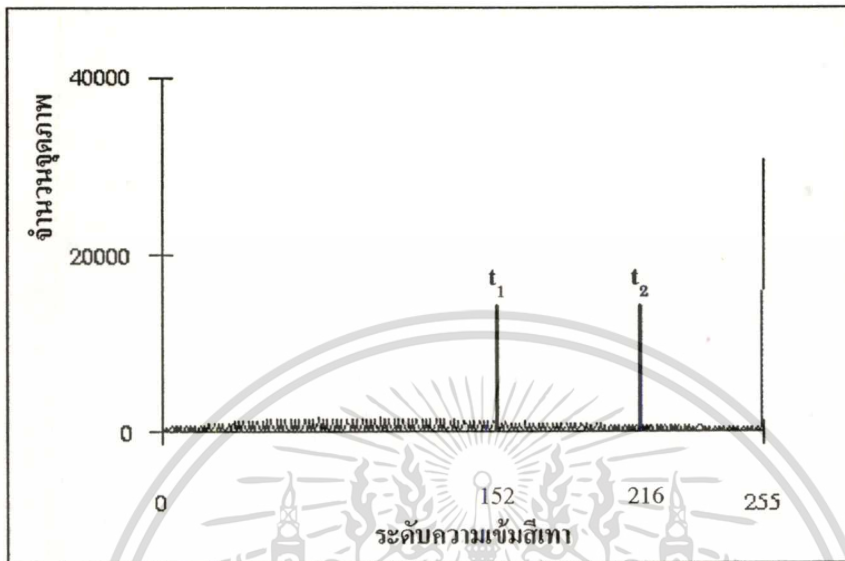
ชื่อเครื่อง (ยี่ห้อ)	คะแนน WinMark (คะแนน)	คะแนน WinBench (คะแนน)	คะแนน ZDigit (คะแนน)	คะแนน Winstone (คะแนน)	คะแนน ZDigit (คะแนน)
AST Ascentia 800N (DX4/76)	40.1	30.9	6.3	3,670	774
IBM ThinkPad 910C (DX4 50, 33)	29.8	18.3	4.3	3,426	434
NEC Verso 8/30 (DX2/33)	30	23.9	4.3	3,504	500
Toshiba T2400CT (DX2/33)	30.3	25.9	3.6	2,448	257
Toshiba T3600 CT (DX2/33)	31.8	25.7	3.5	2,448	224

(ข)

แสดงภาพโมดูลกราฟิคยี่ห้อ (ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 45



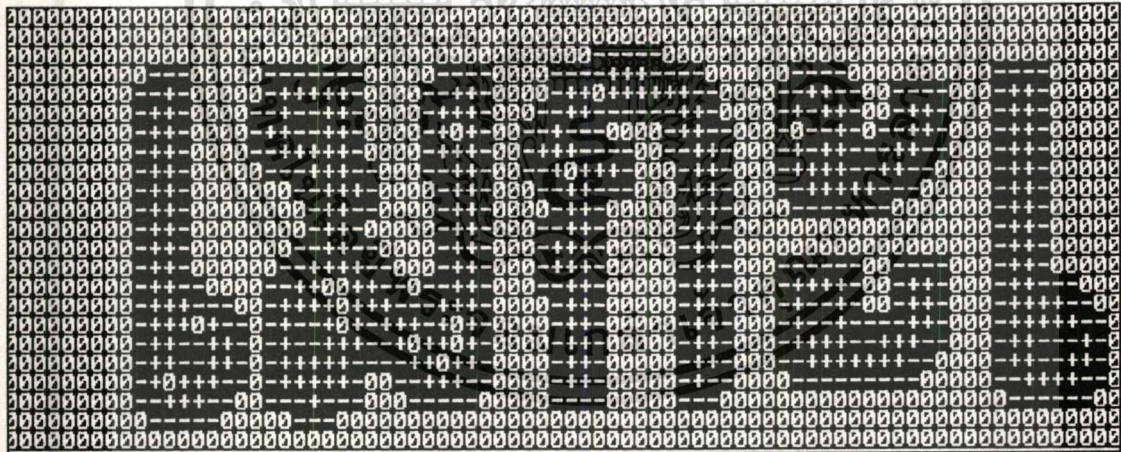
แสดงฮิสโตแกรมระดับความเข้มของภาพ โมดุลกราฟเดียนท์พร้อมทั้งค่าเรด โซลด์

(ก) ภาพ Document 1 (ข) ภาพ Document2

ภาพที่ 46



(ก)



(ข)

แสดงภาพเครื่องภาพที่มี 3 สถานะ S(x,y)

(ก) ภาพ Document 1 (ข) ภาพ Document2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 47

Image Thresholding for Applications' Requiring Character Image Extraction

(ก)

เครื่องโน้ตบุ๊คที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascenia 900N ของ AST ซึ่งใช้ 486DX4, 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะเฮิร์ตซ์อื่น ๆ จากการทดสอบเบนช์มาร์คคะแนนที่ตีพิมพ์เป็นของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอายุงานของแบตเตอรี่ซึ่งยาวนานถึง 8 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0, WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊คเหล่านี้ ๓

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	WinMark (score)	Copy
AST Ascenia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

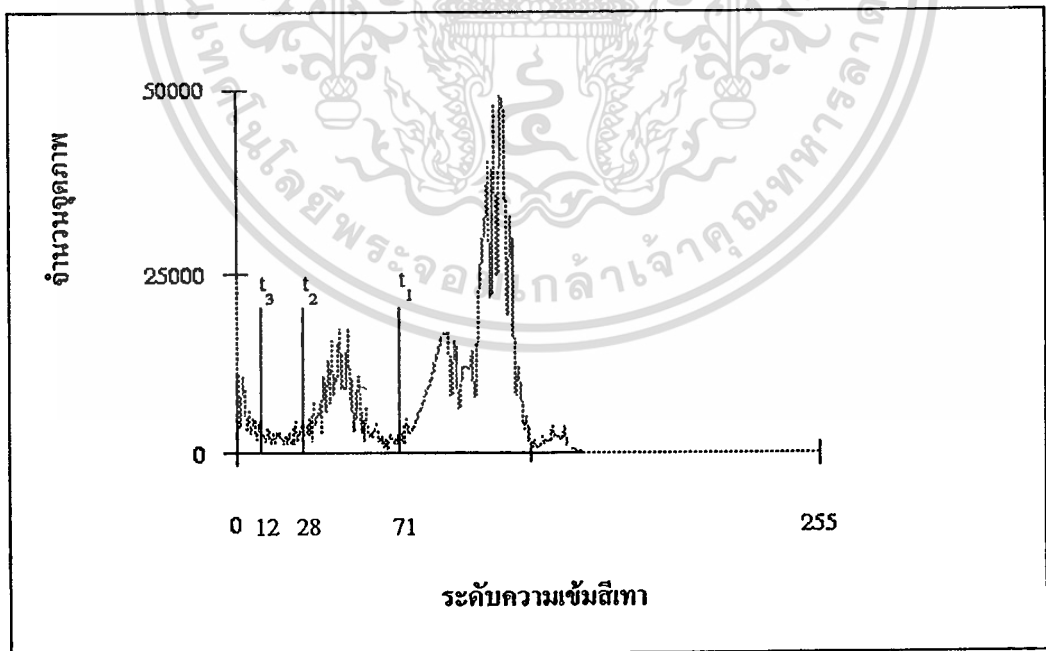
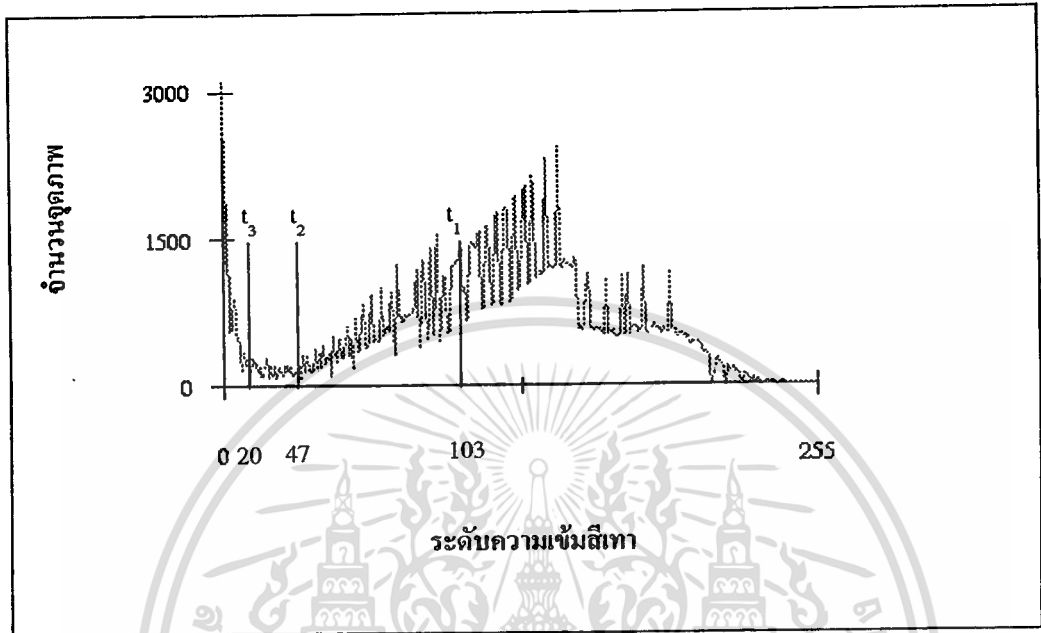
แสดงภาพผลลัพธ์โดยอัลกอริทึมฟังก์ชันร่วม

(ก) ภาพ Document 1 (ข) ภาพ Document 2

ผลของการแยกภาพตัวอักษรโดยอัลกอริธึมการทำเรดโชลด์ตามคุณลักษณะของวัตถุ

ผลของการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้น โดยใช้อัลกอริธึมการทำเรดโชลด์ตามคุณลักษณะของวัตถุ จะให้ภาพผลลัพธ์แสดงดังภาพที่ 49 (ก) และ (ข) ตามลำดับ โดยภาพที่ 48 (ก) และ (ข) แสดงฮิสโตแกรมระดับเข้มของภาพเอกสาร พร้อมทั้งค่าเรดโชลด์ที่คำนวณได้ ค่าเรดโชลด์ที่มีความเหมาะสมสำหรับภาพ *Document 1* เท่ากับ 20 โดยค่าเรดโชลด์ค่าแรก t_1 ที่คำนวณได้มีค่าเท่ากับ 103 และค่าเรดโชลด์ค่าที่สอง t_2 ที่ได้จากการปรับความเหมาะสมของค่ามีค่าเท่ากับ 47 และค่าเรดโชลด์ค่าที่สาม t_3 ที่ได้จากการปรับมีค่าเท่ากับ 20 ซึ่งเป็นค่าเรดโชลด์ค่าสุดท้ายที่ได้จากการปรับแล้วมีค่าไม่ต่ำกว่าค่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร ซึ่งมีค่าเท่ากับ 0 ดังนั้นจึงใช้ค่า t_3 เป็นค่าเรดโชลด์ของภาพ *Document 1* ส่วนค่าเรดโชลด์ที่มีความเหมาะสมสำหรับภาพ *Document 2* เท่ากับ 12 โดยค่าเรดโชลด์ค่าแรก t_1 ที่คำนวณได้มีค่าเท่ากับ 71 และค่าเรดโชลด์ค่าที่สอง t_2 ที่ได้จากการปรับความเหมาะสมของค่ามีค่าเท่ากับ 28 และค่าเรดโชลด์ค่าที่สาม t_3 ที่ได้จากการปรับมีค่าเท่ากับ 12 ซึ่งเป็นค่าเรดโชลด์ค่าสุดท้ายที่ได้จากการปรับแล้วมีค่าไม่ต่ำกว่าค่าระดับความเข้มที่แสดงคุณลักษณะของตัวอักษร ซึ่งมีค่าเท่ากับ 2 ดังนั้นจึงใช้ค่า t_3 เป็นค่าเรดโชลด์ของภาพ *Document 2* สำหรับการประมวลผลภาพเอกสารข้างต้นกำหนดให้พารามิเตอร์ r (ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรในหน้าเอกสาร) เท่ากับ 10

ภาพที่ 48



แสดงฮิสโตแกรมระดับความเข้มของภาพเอกสารพร้อมทั้งค่าเซดโซลด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในสำนักงานเท่านั้น มิใช่เผยแพร่ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 49

Image Thresholding for Applications Requiring Character Image Extraction

(ก)

เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascentia 900N ของ AST ซึ่งใช้ 486DX4. 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะ-

เฮิร์ตซ์อื่นๆ จากการทดสอบเบนช์มาร์ค คะแนนที่ดีเยี่ยมเป็นของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอายุงานของแบตเตอรี่ซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0.

WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ ■

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Life (hours: minutes)
AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:33
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

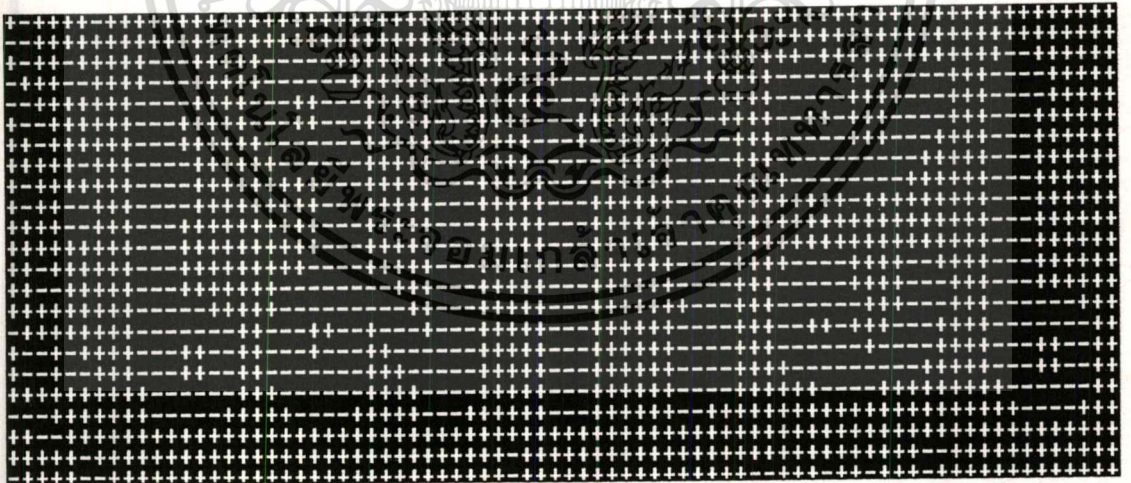
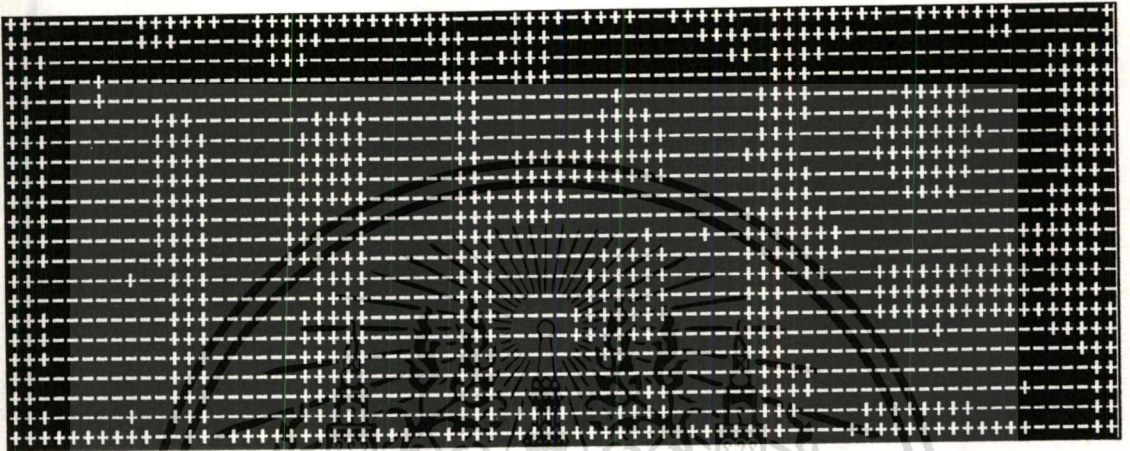
แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรคโคจลด์ตามคุณลักษณะของวัตถุ

(ก) ภาพ Document 1 (ข) ภาพ Document 2

ผลของการแยกภาพตัวอักษรโดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามเครื่อง หมายจุดภาพของภาพลาปลาเซียน

ผลการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้น โดยใช้อัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียนจะให้ภาพผลลัพธ์แสดงดังภาพที่ 53 (ก) และ (ข) ตามลำดับ โดยภาพที่ 50 (ก) และ (ข) แสดงภาพเครื่องหมายลาปลาเซียนของภาพเอกสาร *Document 1* และภาพเอกสาร *Document 2* ตามลำดับ โดยใช้หลักการของดิฟเฟอเรนเชียลของเกาส์เซียน (Difference Of Gaussian) ในการนิยามภาพเครื่องหมาย ภาพที่ 51 เป็นกราฟแสดงระดับความเข้มของจุดภาพ โดยภาพที่ 51 (ก) เป็นกราฟแสดงค่าระดับความเข้มของจุดภาพในแถวที่ 47 ของภาพ *Document 1* ตั้งแต่คอลัมน์ที่ 97 ถึงคอลัมน์ 297 พร้อมทั้งแสดงค่าเรดโซลด์ในแต่ละจุดภาพนั้น ส่วนภาพที่ 51 (ข) เป็นกราฟแสดงค่าระดับความเข้มของจุดภาพในแถวที่ 509 ของภาพ *Document 2* ตั้งแต่คอลัมน์ที่ 41 ถึงคอลัมน์ 370 พร้อมทั้งแสดงค่าเรดโซลด์ในแต่ละจุดภาพนั้น ภาพที่ 52 เป็นกราฟแสดงระดับความเข้มของจุดภาพ โดยภาพที่ 52 (ก) เป็นกราฟแสดงระดับความเข้มของจุดภาพในคอลัมน์ที่ 117 ของภาพ *Document 1* ตั้งแต่แถวที่ 32 ถึง 192 พร้อมทั้งแสดงค่าเรดโซลด์ในแต่ละจุดภาพนั้น ส่วนภาพที่ 52 (ข) เป็นกราฟแสดงระดับความเข้มของจุดภาพในคอลัมน์ที่ 114 ของภาพ *Document 2* ตั้งแต่แถวที่ 212 ถึง 612 พร้อมทั้งค่าเรดโซลด์ในแต่ละจุดภาพนั้น และการประมวลผลภาพเอกสารตัวอย่างทั้ง 2 ภาพดังกล่าว อัลกอริธึมได้กำหนดพารามิเตอร์ค่า k ให้มีค่าเท่ากับ 0.8

ภาพที่ 50

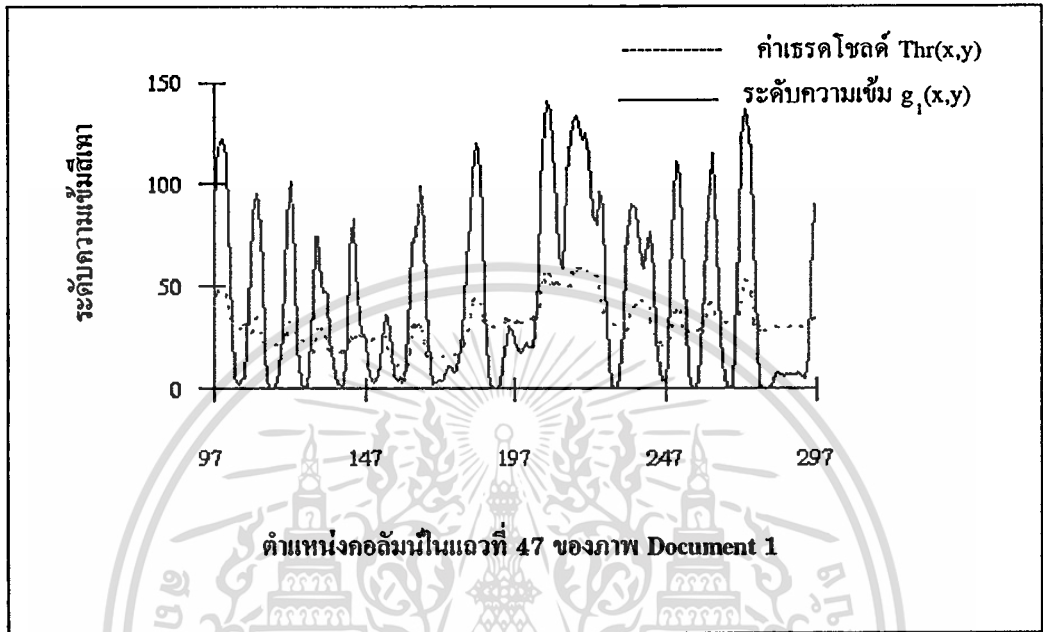


(ข)

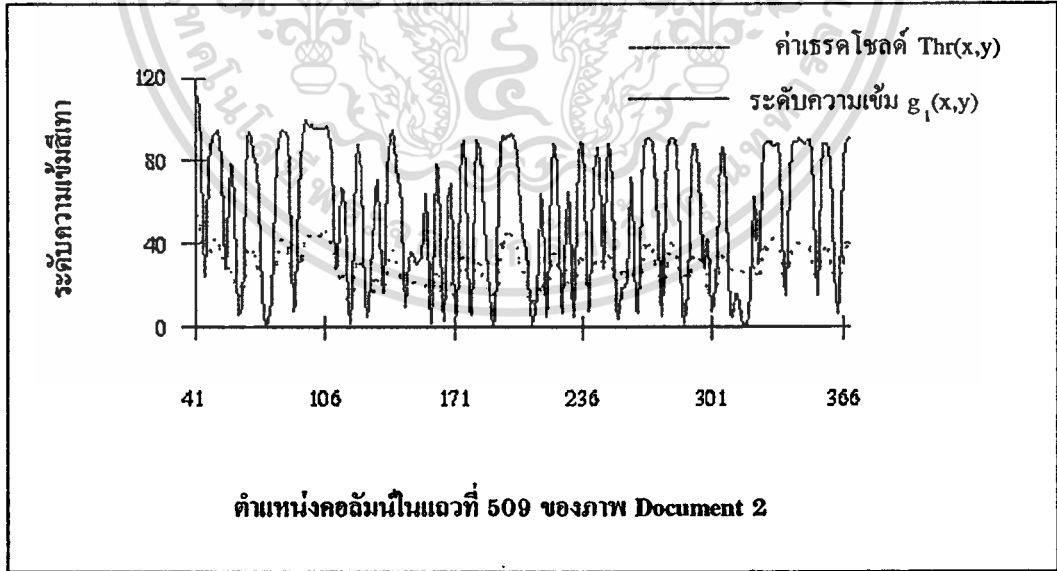
แสดงภาพเครื่องหมายลาปลาเซียนที่นิยามโดยหลักการของ DOG

(ก) ภาพ Document 1 (ข) ภาพ Document 2

ภาพที่ 51



(ก)



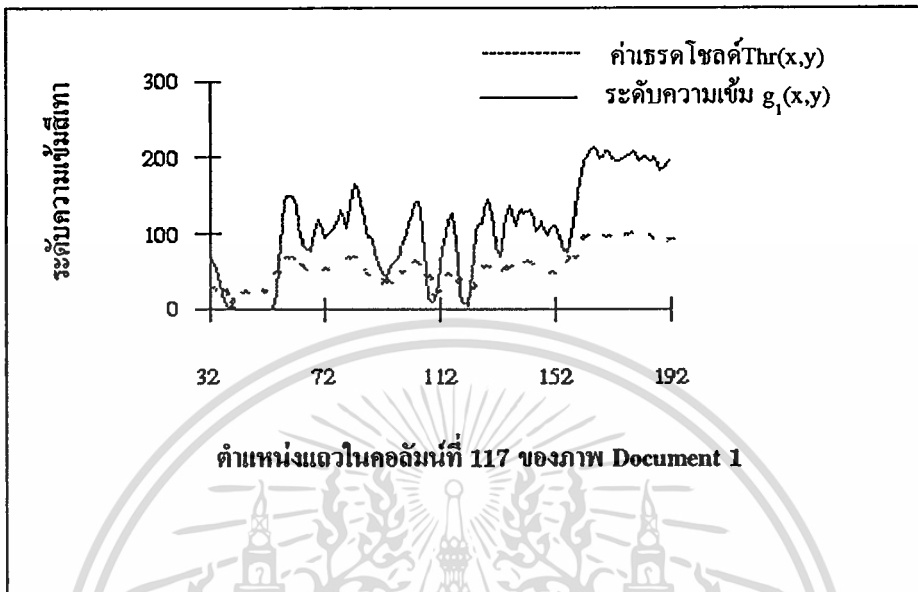
(ข)

กราฟแสดงระดับความชัดของจุดภาพในแนวนอนเปรียบเทียบกับค่าเบรคโฮลด์

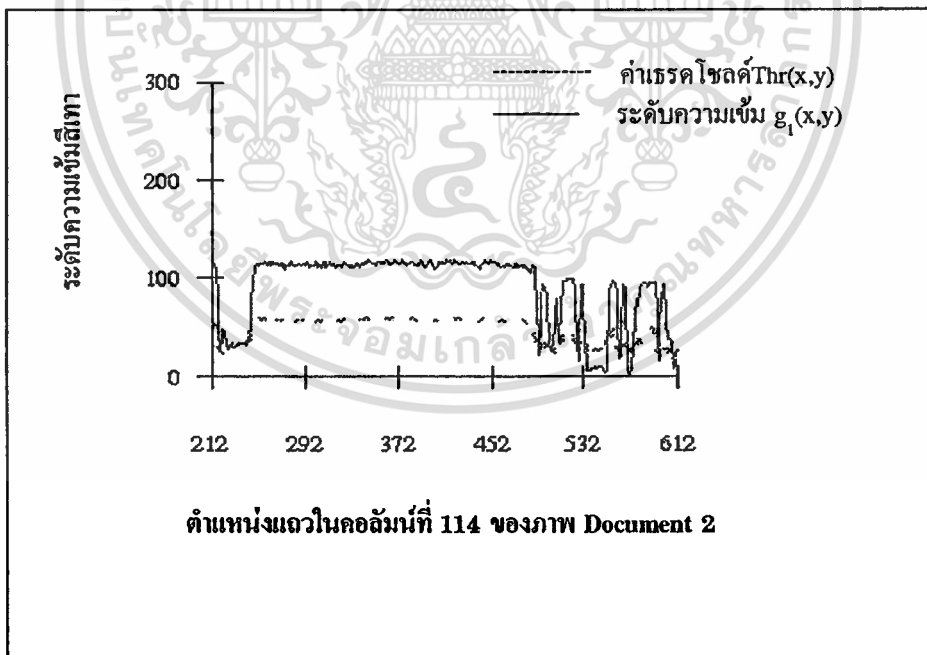
(ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 52



(ก)



(ข)

กราฟแสดงระดับความเข้มของจุดภาพในแนวตั้งเปรียบเทียบกับค่าเรดโซลต์

(ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 53

Image Thresholding for Applications Requiring Character Image Extraction

(ก)

เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascentia 900N ของ AST ซึ่งใช้ 4x6DX4. 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะ-
เฮิร์ตซ์อื่นๆ จากการทดสอบเบนช์มาร์ก คะแนนที่ดีที่สุดเป็นของ Pentium T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอายุงานของ แบตเตอรี่ซึ่งยาวมากถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PC Bench 8.0.

WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการ ของ PC Magazine เพื่อเปรียบเทียบ ประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ c

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Lite (hours: minutes)
AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

แสดงภาพผลลัพธ์ โดยอัลกอริทึมการทำเซลดโซลด์แบบปรับค่าตามเครื่องหมาย

จุดภาพของภาพลาปลาเซียน(ก) ภาพ Document 1 (ข) ภาพ Document 2

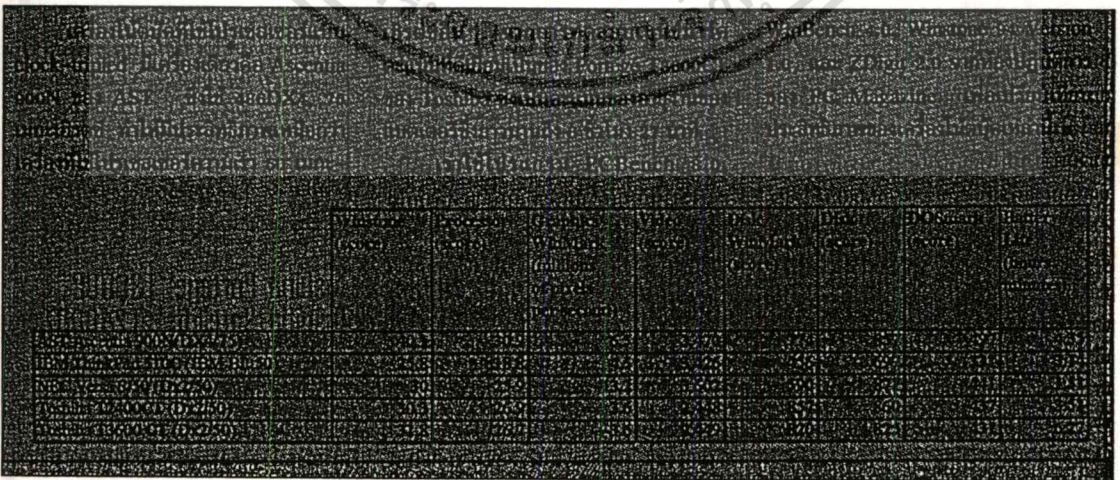
ผลจากการแยกภาพตัวอักษรโดยอัลกอริธึมการประเมินผลต่างของภาพ

ผลการการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้น โดยใช้อัลกอริธึมการประเมินผลต่างของภาพจะให้ภาพผลลัพธ์แสดงดังภาพที่ 57 (ก) และ (ข) ตามลำดับ ซึ่งอัลกอริธึมนี้กำหนดค่าพารามิเตอร์ t เท่ากับ 60 สำหรับการประมวลผลกับภาพ Document 1 และค่าพารามิเตอร์ t เท่ากับ 40 สำหรับการประมวลผลกับภาพ Document 2 ภาพที่ 54 (ก) และ (ข) แสดงภาพไบนารีที่ได้จากขั้นตอนการกรองหาจุดภาพพื้นหลังของภาพ Document 1 และ Document 2 ตามลำดับ โดยจุดภาพสีขาวนิยามถึงจุดภาพของพื้นหลัง ส่วนจุดภาพสีดำนิยามถึงจุดภาพที่มีความเป็นไปได้ที่จะเป็นจุดภาพของตัวอักษร ส่วนภาพที่ 55 (ก) และ (ข) แสดงภาพไบนารีที่ได้รับการปรับปรุงภาพ โดยใช้ความกว้างของเส้นตัวอักษรมาเป็นเกณฑ์ในการปรับปรุง สำหรับภาพที่ 56 (ก) และ (ข) แสดงภาพพื้นหลังของภาพเอกสาร Document 1 และ Document 2 ตามลำดับ

ภาพที่ 54



(ก)



(ข)

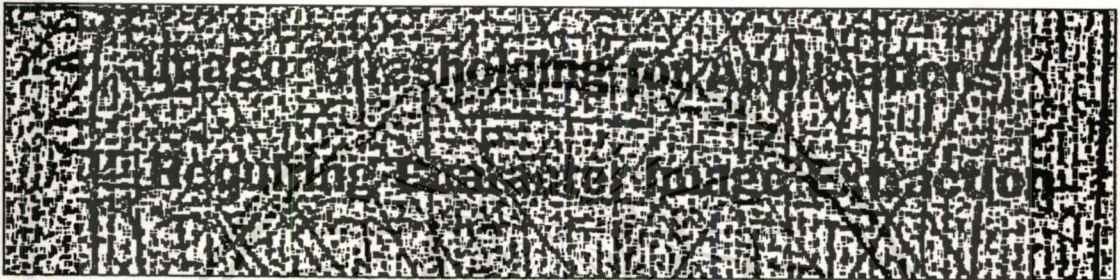
แสดงภาพที่ได้จากขั้นตอนการกรองหาจุดภาพพื้นหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับงานวิจัยและการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

(ก) ภาพ Document 1 (ข) ภาพ Document 2

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 55



(ก)

เครื่องไปรษณีย์อิเล็กทรอนิกส์แบบ clock-matrix มีจอแสดงผล Ascendia 900N หรือ AST รุ่น 486DX+ 75 เมกะไบต์ ทั่วไปในประเภทภาพแทนกราฟิกเครื่องไปรษณีย์เหล่านี้มี ๑๗ เมกะ

เครื่อง ๗ จากภาพทดสอบเหล่านี้คือ คะแนนทดสอบเป็นของ Pentium 1660CT ของ Toshiba ซึ่งฟังก์ชันทดสอบของงานของ แดคเตอร์ที่กำหนดถึง ๑ ชั่วโมง 27 นาที

เราใช้โปรแกรม PC Bench 8.0, WinBench 4.0, WinStone 24 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องไปรษณีย์เหล่านี้

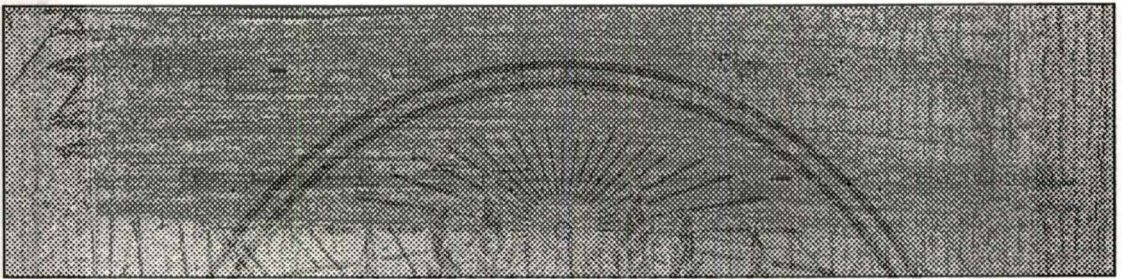
	Whistona (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Lite (hours: minutes)
AST Ascendia 900N(DXA/75)	40.1	30.9	6.5	3,470	774	857	497	2:33
IBM ThinkPad 510G (IBM BI/50)	24.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa 8/50 (Dx2/50)	30	25.9	3.4	3,602	590	871	441	1:31
Toshiba T1400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:24
Toshiba T1600CT (Dx2/50)	31.8	26.7	3.5	2,449	820	825	431	3:27

(ข)

แสดงภาพใบนารีที่ได้รับการปรับปรุง

(ก) ภาพ Document 1 (ข) ภาพ Document 2

ภาพที่ 56



(๗)

แสดงภาพพื้นหลัง (ก) ภาพ Document 1 (๗) ภาพ Document 2

ภาพที่ 57

Image Thresholding for Applications Requiring Character Image Extraction

(ก)

เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascenia 900N ของ AST ซึ่งใช้ 486DX4. 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะ-เฮิร์ตซ์อื่นๆ จากการทดสอบเบนช์มาร์ค คะแนนที่ดีที่สุดเป็นของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอาณาจักรของ แบนด์เวดอร์ซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0.

WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ ■

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Life (hours: minutes)
AST Ascenia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

แสดงภาพผลลัพธ์โดยอัลกอริทึมการประเมินผลต่างของภาพ

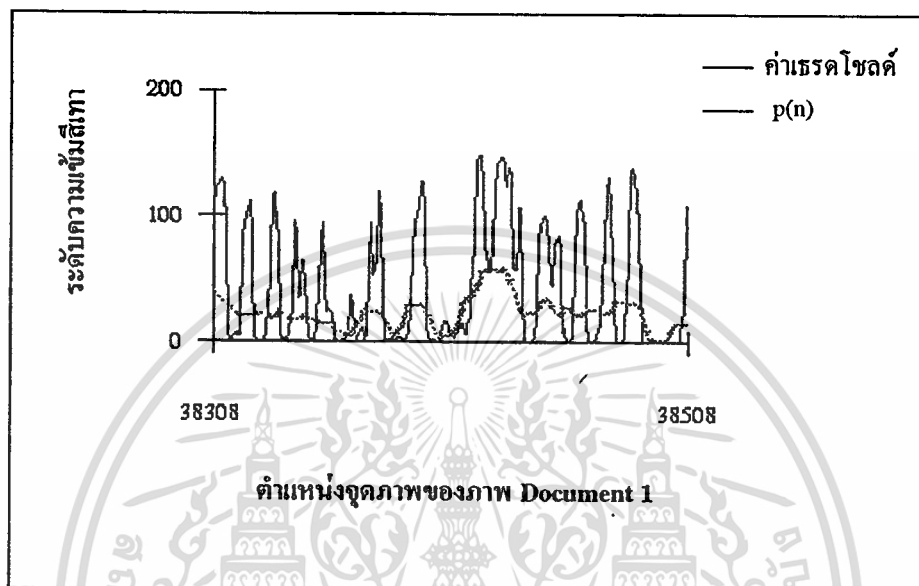
(ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

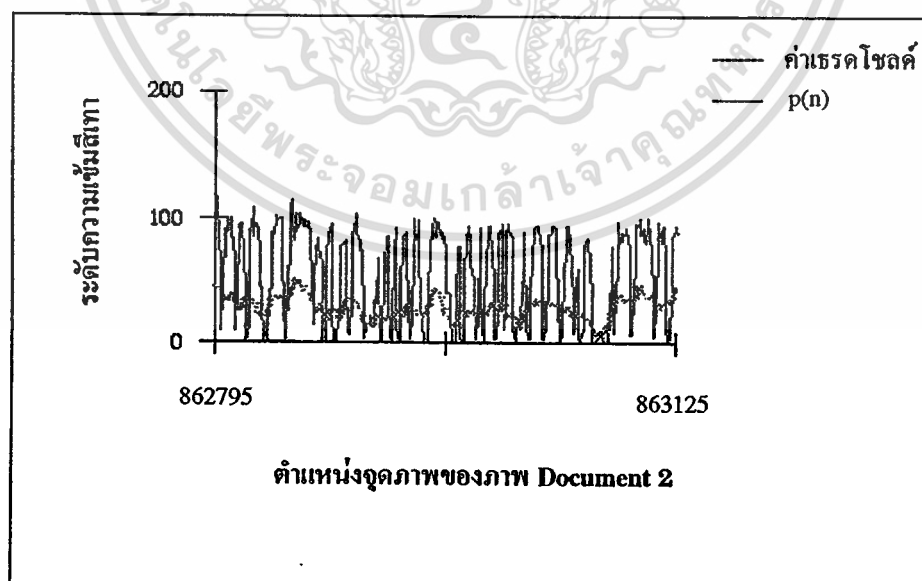
ผลจากการแยกภาพตัวอักษรโดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ย ระดับความเข้มของจุดภาพในแนวราสเตอร์สแกน

ผลการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยใช้อัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพในแนวราสเตอร์จะให้ภาพผลลัพธ์แสดงดังภาพที่ 59 (ก) และ (ข) ตามลำดับ ภาพที่ 58 เป็นกราฟที่แสดงระดับความเข้มของจุดภาพ โดยภาพที่ 58 (ก) เป็นกราฟแสดงค่าระดับความเข้มของจุดภาพในตำแหน่งที่ 38308 ถึง 38508 ของภาพ *Document 1* พร้อมทั้งแสดงค่าเรดโซลด์ในแต่ละจุดภาพนั้น ซึ่งค่าเรดโซลด์เหล่านี้จะเป็นค่าเรดโซลด์ที่ผ่านการถ่วงน้ำหนักค่า โดยสัมประสิทธิ์ที่ใช้สำหรับการถ่วงน้ำหนักจะมีค่าเท่ากับ 0.7 ส่วนภาพที่ 59 (ข) เป็นกราฟแสดงค่าระดับความเข้มของจุดภาพในตำแหน่งที่ 862795 ถึง 863125 ของภาพ *Document 2* พร้อมทั้งแสดงค่าระดับความเข้มที่ใช้เป็นค่าเรดโซลด์ในแต่ละจุดภาพนั้น ซึ่งค่าเรดโซลด์เหล่านี้จะเป็นค่าเรดโซลด์ที่ผ่านการถ่วงน้ำหนักค่า โดยสัมประสิทธิ์ที่ใช้สำหรับการถ่วงน้ำหนักจะมีค่าเท่ากับ 0.8

ภาพที่ 58



(ก)



(ข)

กราฟแสดงระดับความเข้มของภาพเริ่มต้นเปรียบเทียบกับค่าเรดโซลด์ที่เปลี่ยนไปตามข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ (ก) ภาพ Document 1 (ข) ภาพ Document 2 ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 59

Image Thresholding for Applications Requiring Character Image Extraction

(ก)

<p>เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascentia 900N ของ AST ซึ่งใช้ 486DX4, 75 เมกะเฮิรตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะ-</p>	<p>เฟิร์มแวร์อื่น ๆ จากการทดสอบเบนซ์มาร์ค คะแนนที่ตีพิมพ์เป็นของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของทางงานของ แบตเตอรี่ซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0,</p>	<p>WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบ ประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ ..</p>																																																
	<table border="1"> <thead> <tr> <th>Winstone (score)</th> <th>Processor (score)</th> <th>Graphics WinMark (millions of pixels per second)</th> <th>Video (score)</th> <th>Disk WinMark (score)</th> <th>Disk (score)</th> <th>DOSmark (score)</th> <th>Battery Lite (hours: minutes)</th> </tr> </thead> <tbody> <tr> <td>AST Ascentia 900N(DX4/75)</td> <td>40.1</td> <td>30.9</td> <td>6.5</td> <td>3,470</td> <td>774</td> <td>857</td> <td>4:35</td> </tr> <tr> <td>IBM ThinkPad 510C (IBM 131, 50)</td> <td>25.8</td> <td>18.8</td> <td>4.5</td> <td>3,426</td> <td>456</td> <td>697</td> <td>3:14</td> </tr> <tr> <td>NEC Versa S/50 (Dx2/50)</td> <td>30</td> <td>25.9</td> <td>3.2</td> <td>3,602</td> <td>590</td> <td>871</td> <td>1:51</td> </tr> <tr> <td>Toshiba T2400CT (Dx2/50)</td> <td>30.3</td> <td>26.9</td> <td>3.6</td> <td>2,448</td> <td>857</td> <td>860</td> <td>4:37</td> </tr> <tr> <td>Toshiba T3600 CT (Dx2/50)</td> <td>31.8</td> <td>26.7</td> <td>3.6</td> <td>2,449</td> <td>820</td> <td>823</td> <td>6:27</td> </tr> </tbody> </table>	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Lite (hours: minutes)	AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	4:35	IBM ThinkPad 510C (IBM 131, 50)	25.8	18.8	4.5	3,426	456	697	3:14	NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	1:51	Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	4:37	Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	823	6:27	
Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Lite (hours: minutes)																																											
AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	4:35																																											
IBM ThinkPad 510C (IBM 131, 50)	25.8	18.8	4.5	3,426	456	697	3:14																																											
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	1:51																																											
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	4:37																																											
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	823	6:27																																											

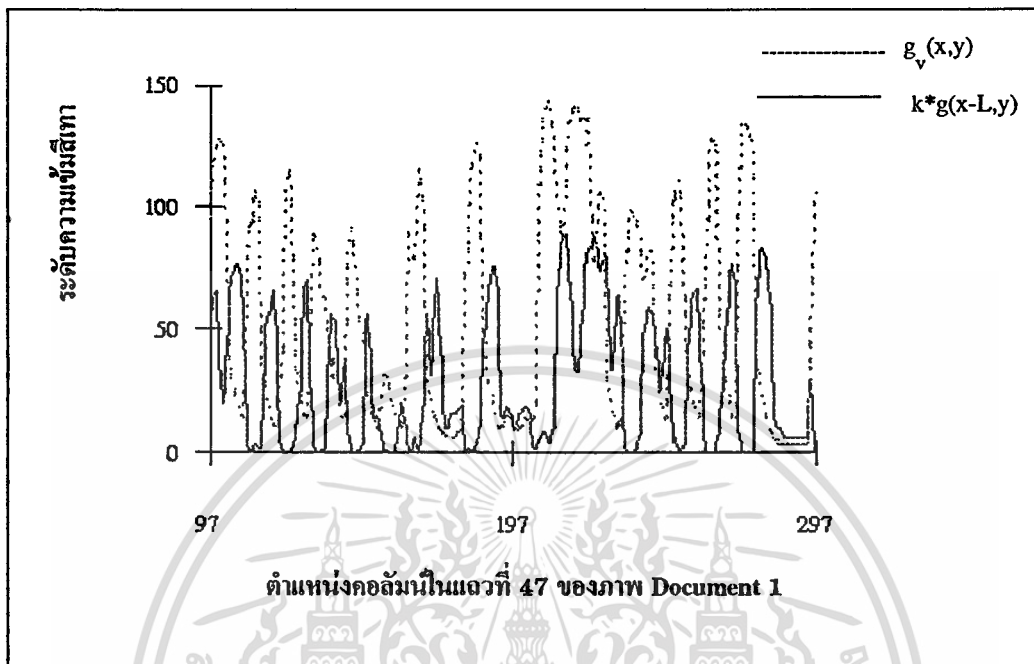
(ข)

แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรคโวลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้ม
ของจุดภาพในแนวราสเตอร์สแกน (ก) ภาพ Document 1 (ข) ภาพ Document 2

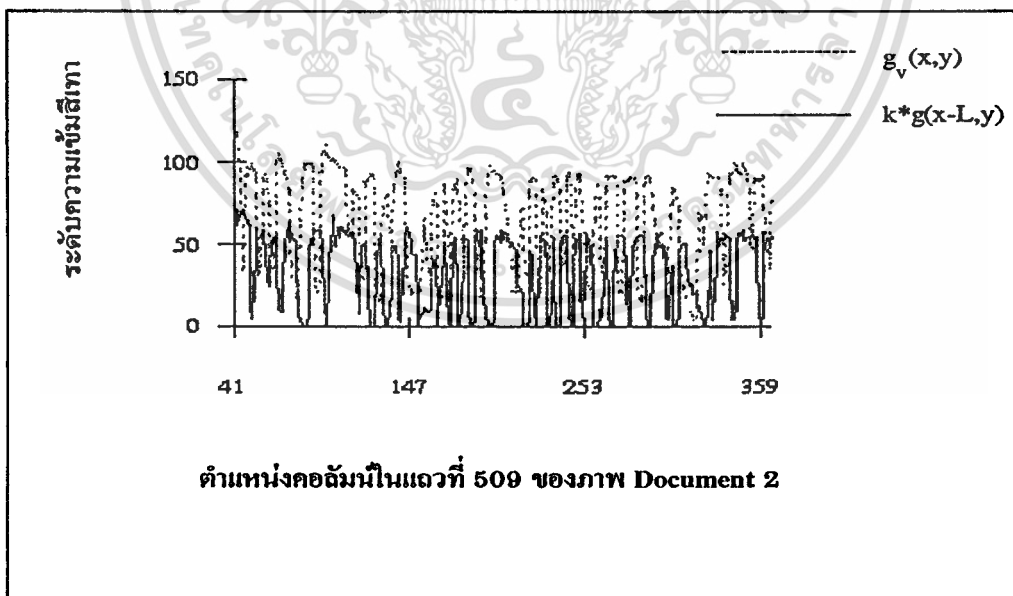
ผลจากการแยกภาพตัวอักษรโดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ย ระดับความเข้มของจุดภาพ

ผลการแยกภาพตัวอักษรออกจากภาพเอกสารเริ่มต้น โดยใช้อัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพจะให้ภาพผลลัพธ์แสดงดังภาพที่ 61 (ก) และ (ข) ตามลำดับ ภาพที่ 60 เป็นกราฟที่แสดงค่าเรดโซลด์ของจุดภาพ โดยภาพที่ 60 (ก) เป็นกราฟแสดงค่าเรดโซลด์ของจุดภาพในแถวที่ 47 ของภาพ *Document1* ตั้งแต่คอลัมน์ที่ 97 ถึงคอลัมน์ 297 พร้อมทั้งแสดงค่าระดับความเข้มของจุดภาพที่พิกัดย้อนหลัง L จุดภาพ คือ $g(x-L, y)$ ซึ่งค่า $g(x-L, y)$ นี้จะถูกนำมาใช้ในการทำเรดโซลด์กับค่าเรดโซลด์ของจุดภาพ และค่า $g(x-L, y)$ เหล่านี้จะเป็นค่าผ่านการถ่วงน้ำหนักค่า โดยสัมประสิทธิ์ที่ใช้สำหรับการถ่วงน้ำหนักค่าจะเท่ากับ 0.85 ส่วนภาพที่ 60 (ข) เป็นกราฟแสดงค่าเรดโซลด์ของจุดภาพในแถวที่ 509 ของภาพ *Document2* ตั้งแต่คอลัมน์ที่ 41 ถึงคอลัมน์ 370 พร้อมทั้งแสดงค่าระดับความเข้มของจุดภาพที่พิกัดย้อนหลัง L จุดภาพ ซึ่งค่าระดับความเข้มนี้จะถูกนำมาใช้ในขั้นตอนการทำเรดโซลด์

ภาพที่ 60



(ก)



(ข)

กราฟแสดงค่าเรคโวลต์ของจุดภาพพร้อมทั้งค่าระดับความเข้ม
ของจุดภาพในพิคัดชั่นหลัง (ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่ 61

Image Thresholding for Applications Requiring Character Image Extraction

เครื่องโน้ตบุ๊กที่ไร้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascenia 900N ของ AST ซึ่งใช้ i486DX4. 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ไร้โปรเซสเซอร์ความถี่ 50 เมกะ-เฮิร์ตซ์อื่นๆ จากการทดสอบเบนช์มาร์ก

คะแนนที่ใกล้เคียงกับของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของรายงานของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ * เราใช้โปรแกรม PCBench 3.0.

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSeMark (score)	Battery Life (hours:minutes)
AST Ascenia 900N (DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,476	456	697	331	3:14
NEC Versa S/20 (DX2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T7400CT (DX2/50)	30.3	26.9	3.6	2,448	437	860	437	3:31
Toshiba T3600 CT (DX2/50)	31.8	26.7	3.6	2,469	820	825	431	6:27

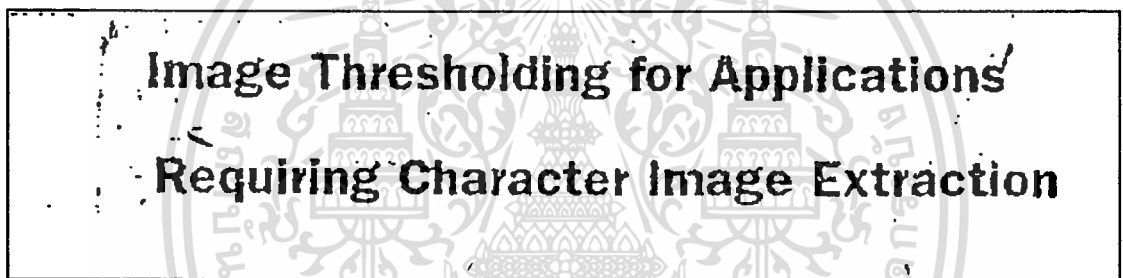
(ข)

แสดงภาพผลลัพธ์โดยอัลกอริทึมการทำเซกชันด้วยวิธีปรับค่าตาม
ค่าเฉลี่ยระดับความเข้มจุดภาพ (ก) ภาพ Document 1 (ข) ภาพ Document 2

ผลจากการแยกภาพตัวอักษรโดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ย และค่าความแปรปรวนของระดับความเข้มจุดภาพ

ผลการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยใช้อัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ จะให้ภาพผลลัพธ์แสดงดังภาพที่ 62 (ก) และ (ข) ตามลำดับ ซึ่งอัลกอริธึมนี้ได้กำหนดขนาดของวินโดวที่ใช้สำหรับการหาค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพเท่ากับ $(2w + 1)^2$ โดยที่ w มีค่าเท่ากับ 20 และสัมประสิทธิ์ r มีค่าเท่ากับ 1.68

ภาพที่ 62



(ก)

เครื่องโน้ตบุ๊คที่ใช้ไปรษณีย์ระบบ clock-tripled มีชื่อผลิตภัณฑ์ Ascentia 900N ของ AST ซีพียู 486DX4, 75 เมกะเฮิร์ต ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้ไปรษณีย์ธรรมดาทั่ว 50 เมกะ-

ให้เรดซ์ลับ ๆ จากการทดสอบแบบซิมาร์ก คณะเทคโนโลยีบัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี ซึ่งเน้นคณะเทคโนโลยีการคอมพิวเตอร์และวิศวกรรมคอมพิวเตอร์ ซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PC Bench 8.0.

WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊คเหล่านี้

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Life (hours: minutes)
AST Ascentia 900N(DX4-75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S-50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

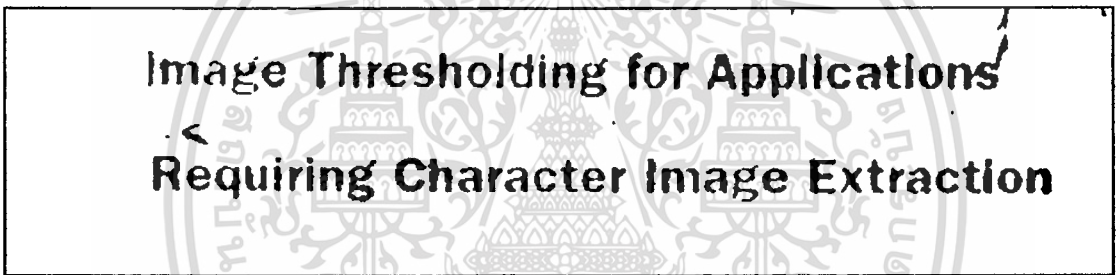
แสดงภาพผลลัพธ์โดยอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยและ

ค่าความแปรปรวนของระดับความเข้มจุดภาพ (ก) ภาพ Document 1 (ข) ภาพ Document 2 เอกสารนี้... ไม่ว่าการมีได้ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลจากการแยกภาพตัวอักษรโดยอัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง

ผลการแยกภาพตัวอักษรออกจากภาพเอกสาร โดยใช้อัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง จะให้ภาพผลลัพธ์แสดงดังภาพที่ 63 (ก) และ (ข) ตามลำดับ ซึ่งอัลกอริทึมนี้ได้กำหนดพารามิเตอร์ r ให้มีค่าเท่ากับ 55 สำหรับการประมวลผลกับภาพเอกสาร Document 1 และกำหนดให้พารามิเตอร์ r มีค่าเท่ากับ 40 สำหรับการประมวลผลกับภาพเอกสาร Document 2

ภาพที่ 63



(ก)

เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascentia 900N ของ AST ซึ่งใช้ 486DX4, 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะ-

เฮิร์ตซ์อื่นๆ จากการทดสอบเบนช์มาร์คคะแนนที่ตีพิมพ์เป็นของ Portégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอุทยานของแบบเดอริซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0,

WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ ■

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	Video (score)	Disk WinMark (score)	Disk (score)	DOSMark (score)	Battery Life (hours: minutes)
AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa 6/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	837	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	825	431	6:27

(ข)

แสดงภาพผลลัพธ์โดยอัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพ ณ พิกัด

รอบข้าง (ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อผู้จัดทำเห็นว่าไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริธึมการแยกภาพตัวอักษร

จากการทดสอบประสิทธิภาพของอัลกอริธึมการแยกภาพตัวอักษร โดยการนำอัลกอริธึมการแยกภาพตัวอักษรทั้ง 8 อัลกอริธึม มาประมวลผลกับภาพเอกสารตัวอย่างที่เตรียมไว้ ภาพผลลัพธ์ที่ได้ในแต่ละอัลกอริธึมอาจมีลักษณะที่ใกล้เคียงกันหรือแตกต่างกันไป ดังที่ได้แสดงไว้ในคอนต้นของบทนี้ ดังนั้นในส่วนนี้จะทำการสรุปและวิเคราะห์ผลการทดลองที่เกิดขึ้นในแต่ละอัลกอริธึม พร้อมทั้งวิเคราะห์รูปแบบของอัลกอริธึมที่นั้น

อัลกอริธึมฟังก์ชันร่วม

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริธึมฟังก์ชันร่วมสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์ พบว่าภาพผลลัพธ์ที่ได้จากอัลกอริธึมรูปแบบนี้จัดอยู่ในเกณฑ์ที่มีความเหมาะสม แต่สำหรับภาพเอกสารเริ่มต้นที่มีความความซับซ้อนเกิดขึ้นสูงในส่วนของพื้นหลัง ภาพผลลัพธ์ที่ได้ อาจจะมีสิ่งไม่พึงประสงค์เกิดขึ้นบนพื้นหลังของภาพ ซึ่งสิ่งสำคัญที่จะมีผลต่อการเกิดสิ่งไม่พึงประสงค์ในภาพผลลัพธ์ คือ ค่าเรคโพลด์ของภาพโมดูลกราเดียนท์ ถ้าค่าเรคโพลด์ที่คำนวณได้มีค่าสูงก็จะทำให้มีสิ่งไม่พึงประสงค์เกิดขึ้นบนภาพผลลัพธ์น้อย แต่สิ่งนี้อาจจะเป็นสาเหตุทำให้เกิดความไม่สมบูรณ์ของภาพตัวอักษรในภาพผลลัพธ์ เพราะถ้าค่าเรคโพลด์ที่สูงเกินไปอาจจะทำให้สูญเสียความเป็นขอบของตัวอักษรในขั้นตอนการนิยามขอบของตัวอักษร และถ้าค่าเรคโพลด์มีค่าต่ำเกินไปอาจเป็นสาเหตุทำให้เกิดสิ่งไม่พึงประสงค์ในอัตราที่สูงขึ้น ดังนั้นถ้าใช้ค่าเรคโพลด์ที่มีความเหมาะสมแล้ว สิ่งไม่พึงประสงค์ที่เกิดขึ้นในภาพผลลัพธ์จะเป็นสิ่งที่ไม่พึงประสงค์ที่มีระดับความเข้มที่ใกล้เคียงหรือระดับเดียวกับระดับความเข้มของตัวอักษร และที่สำคัญมีขนาดใกล้เคียงกับขนาดของตัวอักษรด้วย เพราะในอัลกอริธึมนี้ได้นำค่าความกว้างของเส้นตัวอักษรมาใช้เป็นเกณฑ์พิจารณาคุณสมบัติของจุดภาพสำหรับการสร้างภาพผลลัพธ์ในขั้นตอนสุดท้าย

2. ข้อจำกัดที่เกิดขึ้นกับอัลกอริธึมการแยกภาพตัวอักษรในรูปแบบนี้ คือ รูปแบบการทำงานของอัลกอริธึมนี้ไม่เหมาะสมสำหรับการประมวลผลกับภาพเอกสารที่มีความหลากหลาย

ของขนาดตัวอักษร เพราะอัลกอริธึมนี้จะใช้ความกว้างของเส้นตัวอักษรเป็นเกณฑ์ในการพิจารณาสร้างภาพผลลัพธ์

3. พารามิเตอร์ที่ใช้ในอัลกอริธึมนี้มี 2 ค่า คือ r และ W โดยที่ r คือ ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างขอบของตัวอักษรในหน้าเอกสาร ส่วน W คือ ขนาดความกว้างของเส้นตัวอักษร

อัลกอริธึมการเรคโพลด์ตามคุณลักษณะของวัตถุ

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริธึมการทำเรคโพลด์ตามคุณลักษณะของวัตถุสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์พบว่า ภาพผลลัพธ์ที่ได้จากอัลกอริธึมรูปแบบนี้ถือว่าเป็นภาพผลลัพธ์จัดที่อยู่ในเกณฑ์ที่ดี มีความถูกต้องสูง สิ่งไม่พึงประสงค์ที่จะเกิดในภาพผลลัพธ์ได้นั้นจะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงมากหรือระดับเดียวกันกับระดับความเข้มของตัวอักษร และจากการวิเคราะห์รูปแบบการทำงานของอัลกอริธึม *OAT* พบว่าการทำงานด้วยหลักการดังกล่าวเปรียบเสมือนกับการสร้างเซตของค่าเรคโพลด์ขึ้นมาหนึ่งเซตจากค่าที่สามารถเป็นไปได้ 256 ค่า สมาชิกภายในเซตจะประกอบไปด้วย ค่าเรคโพลด์ที่ได้จากการปรับในแต่ละครั้ง หลังจากนั้นก็ทำการเลือกสมาชิกภายในเซตที่มีคุณสมบัติเหมาะสมที่สุด ก็คือ ค่าเรคโพลด์ค่าสุดท้ายที่ได้จากการปรับ ซึ่งการทำงานด้วยหลักการดังกล่าวนี้ถือว่า อัลกอริธึม *OAT* เป็นรูปแบบของอัลกอริธึมที่มีความยืดหยุ่นสูงพอที่จะใช้ประมวลผลกับภาพเอกสารเริ่มต้นที่มีความซับซ้อนในส่วนของพื้นหลังสูงและให้ภาพผลลัพธ์ที่มีถูกต้องเหมาะสม

2. จากขั้นตอนการคำนวณหาระดับความเข้มที่มีคุณลักษณะของตัวอักษร (A_i) พบว่าค่า r ยิ่งสูงยิ่งแสดงว่าภาพมีซับซ้อนในส่วนของพื้นหลังสูง และองค์ประกอบของภาพที่เป็นตัวอักษรจะมีจำนวนน้อยเมื่อเทียบกับองค์ประกอบอื่น ๆ ภายในภาพเอกสารนั้น

3. พารามิเตอร์ที่ใช้ในอัลกอริธึม คือ r โดยที่ r คือ ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรในหน้าเอกสาร

4. อัลกอริธึม *OAT* นี้จัดเป็นอัลกอริธึมในรูปแบบของแอฟพลิเคชันความเร็วสูง

อัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียน

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริทึมการทำเรดโซลด์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียนสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์ พบว่า ภาพผลลัพธ์ที่ได้จากอัลกอริทึมรูปแบบนี้ถือว่าเป็นภาพผลลัพธ์จัดที่อยู่ในเกณฑ์ที่ดีมีความเหมาะสม ลักษณะของภาพที่ได้จะมีความราบเรียบไม่มีความไวต่อสัญญาณรบกวนภาพ ซึ่งลักษณะเช่นนี้มีสาเหตุอันเนื่องมาจากการใช้หลักการของ *DOG* ในการนิยามภาพเครื่องหมายลาปลาเซียน สำหรับสิ่งที่ไม่พึงประสงค์ที่เกิดขึ้นในภาพผลลัพธ์จะมีสาเหตุอันเนื่องมาจากสิ่งที่ไม่พึงประสงค์นั้นมีระดับความเข้มใกล้เคียงหรือระดับเดียวกับกับระดับความเข้มของตัวอักษร และสิ่งที่ไม่พึงประสงค์ที่เกิดขึ้นในภาพผลลัพธ์จะมีขนาดค่อนข้างหลากหลาย เพราะการแยกภาพตัวอักษรในรูปแบบนี้มีได้นำขนาดความกว้างของเส้นตัวอักษรมาใช้เป็นเกณฑ์พิจารณาจุดภาพในขั้นตอนการสร้างภาพผลลัพธ์

2. พารามิเตอร์ที่ใช้ในอัลกอริทึมมี 1 ค่า คือ สัมประสิทธิ์ที่ใช้สำหรับการถ่วงน้ำหนักในสมการการคำนวณค่าเรดโซลด์ (สมการที่ 3.11) ซึ่งสัมประสิทธิ์ที่ใช้ต้องมีค่าที่เหมาะสม ถ้ามีค่าน้อยเกินไปอาจทำให้ภาพตัวอักษรในภาพผลลัพธ์มีลักษณะของการกัดเซาะเกิดขึ้น (*Erosion Line Character*) แต่ถ้ามากเกินไปอาจทำให้ภาพผลลัพธ์มีความผิดพลาดเกิดขึ้นได้ แต่อย่างไรก็ตามการกำหนดค่าสัมประสิทธิ์ก็ต้องคำนึงถึง *Configuration* ของภาพเอกสารที่นำมาประมวลผลด้วย ดังเช่น กรณีภาพเอกสารนั้นมีตัวอักษรที่มีระดับความเข้มสูง (ความแตกต่างของระดับความเข้มระหว่างอักษรกับพื้นหลังน้อย) ค่าสัมประสิทธิ์ที่ใช้ควรมีค่าสูงกว่าค่าสัมประสิทธิ์ที่ใช้สำหรับภาพเอกสารที่มีตัวอักษรสีเข้มแตกต่างกับพื้นหลังอย่างชัดเจน

3. ข้อเสียที่เกิดขึ้นกับอัลกอริทึมการแยกภาพตัวอักษรในรูปแบบนี้ คือ เวลาที่ใช้ในการประมวลผล โดยเฉพาะขั้นตอนการนิยามภาพเครื่องหมายลาปลาเซียนถือว่าเป็นขั้นตอนที่ใช้เวลาในการประมวลผลนานที่สุดในขั้นตอนการแยกภาพตัวอักษรในอัลกอริทึมรูปแบบนี้

อัลกอริทึมการประเมินผลต่างของภาพ

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริทึมการประเมินผลต่างของภาพ สามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์ พบว่าภาพผลลัพธ์ที่ได้จากอัลกอริทึมรูปแบบนี้ จะมีความเหมาะสม สิ่งไม่พึงประสงค์ที่สามารถเกิดขึ้นบนพื้นหลังของภาพ จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มและขนาดใกล้เคียงกับตัวอักษร ที่เป็นเช่นนี้เพราะอัลกอริทึมในรูปแบบนี้ได้ใช้ค่าความกว้างของตัวอักษรและค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรกับพื้นหลังมาเป็นเกณฑ์ในการพิจารณาจุดภาพ และจากการวิเคราะห์ขั้นตอนการทำงานของอัลกอริทึมรูปแบบนี้สามารถสรุปได้ว่า วัตถุประสงค์ของแต่ละขั้นตอนการทำงานย่อย คือ การหาจุดภาพที่มีคุณสมบัติของพื้นหลังโดยพิจารณาจากเกณฑ์ที่กำหนด และนิยามจุดภาพเหล่านั้นให้เป็นจุดภาพของพื้นหลัง ดังนั้นภาพผลลัพธ์ที่ได้จากขั้นตอนการทำงานหลัง ๆ จะมีจำนวนของจุดภาพที่มีคุณสมบัติของพื้นหลังมากกว่าภาพผลลัพธ์ที่ได้จากขั้นตอนการทำงานแรก ๆ

2. พารามิเตอร์ที่ใช้ในอัลกอริทึมรูปแบบนี้มี 2 ค่า คือ W กับ r ซึ่ง W คือ ขนาดความกว้างของเส้นตัวอักษร ซึ่งในอัลกอริทึมนี้ได้ใช้ค่า W เป็นเกณฑ์การพิจารณาจุดภาพในขั้นตอนที่ 2 หากจุดภาพที่มีความเป็นไปได้ที่จะเป็นจุดภาพตัวอักษรซึ่ง ส่วน r คือ ค่าผลต่างของระดับความเข้มที่เป็นไปได้ระหว่างตัวอักษรกับพื้นหลัง ค่านี้จะถูกนำไปใช้เป็นเกณฑ์ในการพิจารณาจุดภาพสำหรับการสร้างการภาพผลลัพธ์ในขั้นตอนสุดท้าย สำหรับภาพเอกสารที่ตัวอักษรมีระดับความเข้มสูง (ความแตกต่างของระดับความเข้มระหว่างอักษรกับพื้นหลังน้อย) ค่า r ที่ใช้ควรมีค่าน้อยกว่าค่า r ที่ใช้สำหรับภาพเอกสารที่มีตัวอักษรสีเข้มแตกต่างกับพื้นหลังอย่างชัดเจน

3. ข้อจำกัดที่เกิดขึ้นกับอัลกอริทึมการแยกภาพตัวอักษรในรูปแบบนี้ คือ รูปแบบการทำงานของอัลกอริทึมนี้ไม่เหมาะสมสำหรับการประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษรและภาพเอกสารที่มีความหลากหลายของระดับความเข้มตัวอักษร

อัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพตามแนว ราสเตอร์สแกน

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริธึมการทำเรดโซลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพตามแนวราสเตอร์สแกนสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์พบว่า ภาพผลลัพธ์ที่ได้จัดอยู่เกณฑ์ที่มีความเหมาะสม และจากวิเคราะห์รูปแบบการคำนวณหาค่าเฉลี่ยระดับความเข้มของจุดภาพที่ใช้ในอัลกอริธึมรูปแบบนี้พบว่า จำนวนของจุดภาพที่ใช้ในการหาค่าเฉลี่ยจะมีผลต่อลักษณะการเกิดของภาพผลลัพธ์ ซึ่งถ้าจำนวนจุดภาพที่ใช้ในการค่าเฉลี่ยน้อยเกินไป จะทำให้เกิดความไม่สมบูรณ์ในองค์ประกอบของเส้นตัวอักษร คือ เส้นตัวอักษรจะมีลักษณะเหมือนถูกกัดเซาะ (*Erosion Line Character*) แต่ถ้าจำนวนจุดภาพที่ใช้ในการค่าเฉลี่ยมากเกินไป จะมีผลต่อการเกิดสิ่งไม่พึงประสงค์ในพื้นที่หลังในอัตราที่สูงขึ้น และจากการทดสอบอัลกอริธึมในรูปแบบนี้กับภาพเอกสารหลากหลายพบว่า จำนวนจุดภาพที่มีความเหมาะสมที่จะใช้สำหรับการหาค่าเฉลี่ยระดับความเข้มจะมีค่าโดยประมาณเท่ากับ $2W$ โดยที่ W คือ ค่าความกว้างของเส้นตัวอักษร

2. พารามิเตอร์ที่ใช้ในอัลกอริธึมรูปแบบนี้มี 2 ค่า คือ S กับ k ซึ่งค่า S คือ จำนวนจุดภาพที่ใช้สำหรับการหาค่าเฉลี่ยระดับความเข้ม ส่วน k คือ สัมประสิทธิ์ที่ใช้ในการถ่วงน้ำหนักค่าเรดโซลด์ ซึ่งการกำหนดค่าพารามิเตอร์ทั้งสองควรมีความสัมพันธ์ ดังเช่น กรณีที่กำหนดค่า S มีค่ามากเกินไป ค่า k ควรมีค่าต่ำลง แต่ในกรณีที่มีการกำหนดค่า S ที่เหมาะสม การกำหนดค่า k ที่ดีควรคำนึงถึง *Configuration* ของภาพเอกสารที่นำมาประมวลผลด้วย ดังเช่น กรณีภาพเอกสารนั้นมีตัวอักษรที่มีระดับความเข้มสูง (ความแตกต่างของระดับความเข้มระหว่างอักษรกับพื้นหลังน้อย) ค่า k ที่ใช้ควรมีค่าสูงกว่าค่า k ที่ใช้สำหรับภาพเอกสารที่มีตัวอักษรสีเข้มแตกต่างกับพื้นหลังอย่างชัดเจน

3. อัลกอริธึมแยกภาพตัวอักษรในรูปแบบนี้ สามารถจัดอยู่ในประเภทแอปพลิเคชันความเร็วสูง คือ เวลาที่ใช้ในการประมวลผลภาพน้อย

อัลกอริธึมการทำเรคโพลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพ

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริธึมการทำเรคโพลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มของจุดภาพสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์ พบว่า ภาพผลลัพธ์ที่ได้มีความเหมาะสม และจากการวิเคราะห์หลักการของอัลกอริธึมในรูปแบบนี้พบว่า ภาพตัวอักษรในภาพผลลัพธ์อาจจะไม่สมบูรณ์ในองค์ประกอบของเส้นเกิดขึ้น คือ เส้นของตัวอักษรจะมีลักษณะของการถูกกัดเซาะเส้น (*Erosion Line Character*) ซึ่งลักษณะที่เกิดขึ้นดังกล่าวนี้ส่วนหนึ่งจะมีสาเหตุอันเนื่องมาจากความไม่เหมาะสมของขนาดที่ใช้ในกำหนดพิกัดของจุดภาพย้อนหลัง (คือ ค่า L) เพื่อที่จะนำค่าระดับความเข้มของจุดภาพในพิกัดย้อนหลัง $g(x-L, y)$ ไปเปรียบเทียบกับค่าเรคโพลด์หรือค่าเฉลี่ยระดับความเข้ม $g_v(x, y)$ และนอกจากนี้ช่วงห่างระหว่างตัวอักษรก็จะมีผลต่อความสมบูรณ์ของเส้นตัวอักษรในภาพผลลัพธ์ด้วย เช่น กรณีที่ช่วงห่างระหว่างตัวอักษรมีค่าน้อยมาก และน้อยกว่าขนาดความกว้างของเส้นตัวอักษร สิ่งนี้จะเป็นสาเหตุของผิดพลาดที่เกิดขึ้นในขั้นตอนการนำระดับความเข้มของจุดภาพย้อนหลังไปเปรียบเทียบกับค่าเรคโพลด์ ซึ่งจะทำให้ภาพผลลัพธ์ที่ได้มีความไม่เหมาะสม

2. พารามิเตอร์ที่ใช้ในอัลกอริธึมรูปแบบนี้จะอยู่ในรูปของสัมประสิทธิ์ที่ใช้ในฟังก์ชันการปรับค่า W_h และ W_v ตามลำดับ และการกำหนดค่าเริ่มต้นให้กับพารามิเตอร์เหล่านี้จำเป็นต้องรู้วัตถุประสงค์ของฟังก์ชันเสียก่อนและสามารถคาดการณ์ถึงลักษณะของค่าผลลัพธ์ที่จะได้จากฟังก์ชัน และสัมประสิทธิ์ที่ใช้ในสมการของฟังก์ชัน W_h และ W_v นั้นอาจจะมีการเปลี่ยนแปลงค่าตามเกณฑ์ที่กำหนดขึ้นเพื่อให้ค่าผลลัพธ์ที่ได้อยู่ในลักษณะที่ต้องการ ดังเช่นอัลกอริธึมในรูปแบบนี้สัมประสิทธิ์ของฟังก์ชันจะมีการเปลี่ยนแปลงค่าตามเกณฑ์ที่กำหนดขึ้น เพื่อให้ค่าผลลัพธ์ของฟังก์ชัน W_h และ W_v มีลักษณะดังเช่นกราฟในภาพที่ 38 (ก) และ (ข) ตามลำดับ และการกำหนดค่าของสัมประสิทธิ์ที่มีเหมาะสมจริงนั้นอาจทำให้การถ่วงน้ำหนักค่าเรคโพลด์ไม่มีความจำเป็นต่อขั้นตอนการแยกภาพตัวอักษรในอัลกอริธึมนี้เลยก็ได้

3. อัลกอริธึมแยกภาพตัวอักษรในรูปแบบนี้ สามารถจัดอยู่ในประเภทแอปพลิเคชันความเร็วสูง คือ เวลาที่ใช้ในการประมวลผลภาพน้อย

อัลกอริทึมการทำเซรโดไซด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริทึมการทำเซรโดไซด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์พบว่า ภาพผลลัพธ์ที่ได้จากอัลกอริทึมรูปแบบนี้จัดอยู่ในเกณฑ์ที่เหมาะสม และจากการทดสอบโดยการนำอัลกอริทึมในรูปแบบนี้ไปประมวลผลกับชุดของภาพเอกสารหลากหลายที่เตรียมไว้ พบว่าขนาดของวินโดว์ที่เหมาะสมสำหรับการหาค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มที่พิกัดจุดภาพ (x,y) โดยที่ (x,y) เป็นจุดศูนย์กลางของวินโดว์นั้นจะมีค่าเท่ากับ W^2 โดยที่ $W=(2w+1)$ ซึ่งค่าของพารามิเตอร์ w ที่ความเหมาะสมจะมีค่าโดยประมาณเท่ากับ 20 และพารามิเตอร์ k ซึ่งเป็นสัมประสิทธิ์นั้น ความเป็นไปได้ของค่าจะอยู่ในช่วง $[0,\infty]$ แต่ค่าของ k ที่มีนัยสำคัญจะอยู่ในช่วงค่า $[0,3]$ ซึ่งถ้ากำหนดค่าของ k ให้มีค่ามากกว่าช่วงที่กำหนดภาพของตัวอักษรในภาพผลลัพธ์จะอยู่ในลักษณะที่ไม่สมบูรณ์ คือ เส้นของตัวอักษรจะมีลักษณะเหมือนถูกกัดเซาะ (*Erosion Line Character*) การกำหนดค่า k ที่ดีควรคำนึงถึง *Configuration* ของภาพเอกสารที่นำมาประมวลผลด้วย ดังเช่น กรณีภาพเอกสารนั้นมีตัวอักษรที่มีระดับความเข้มสูง (ความแตกต่างของระดับความเข้มระหว่างอักษรกับพื้นหลังน้อย) ค่า k ที่ใช้ควรมีค่าต่ำกว่าค่า k ที่ใช้สำหรับภาพเอกสารที่มีตัวอักษรสีเข้มแตกต่างกับพื้นหลังอย่างชัดเจน และจากการทดสอบกับภาพเอกสารหลากหลายนั้นพบว่าค่าของ k ที่มีความเหมาะสมจะมีค่าประมาณ 1.68

2. พารามิเตอร์ที่ใช้ในอัลกอริทึมรูปแบบนี้มี 2 ค่า คือ k กับ w การกำหนดค่าเริ่มต้นให้กับพารามิเตอร์เหล่านี้สามารถทำได้โดยง่าย เพียงเข้าใจหลักการของสมการเสียก่อนและใช้ผลของการทดลองซ้ำ ๆ มาเป็นแนวทางในการปรับค่าของพารามิเตอร์เหล่านี้ให้มีความเหมาะสมยิ่งขึ้น เพื่อให้ภาพผลลัพธ์ที่ได้มีความถูกต้องมากที่สุด

3. ข้อจำกัดที่เกิดขึ้นในอัลกอริทึมแยกภาพตัวอักษรรูปแบบนี้ คือ ความเร็วที่ใช้ในการประมวลผลต่ำ เพราะขั้นตอนการประมวลผลในแต่ละจุดภาพจะต้องมีการอ้างอิงถึงจุดภาพในพิกัดรอบข้าง

อัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง

การวิเคราะห์ผลการทดลองและรูปแบบของอัลกอริทึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้างสามารถสรุปได้ดังต่อไปนี้

1. จากการประเมินภาพผลลัพธ์พบว่า ภาพผลลัพธ์ที่ได้จากอัลกอริทึมรูปแบบนี้จัดอยู่ในเกณฑ์ที่เหมาะสม แต่ภาพผลลัพธ์จะมีการสูญเสียรายละเอียดบริเวณขอบของตัวอักษรไปบ้าง ซึ่งลักษณะเช่นนี้จะมีสาเหตุมาจากขั้นตอนการกรองสัญญาณรบกวนภาพ ซึ่งทำให้มีการสูญเสียรายละเอียดของขอบภาพตัวอักษรไปบ้าง สำหรับสิ่งไม่พึงประสงค์ที่เกิดขึ้นบนพื้นหลังของภาพผลลัพธ์นี้จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับระดับความเข้มของอักษร และจากการวิเคราะห์รูปแบบการทำงานพบว่าอัลกอริทึมในรูปแบบนี้จะมีบางขั้นตอนการทำงานที่คล้ายคลึงกับอัลกอริทึมแบบฟังก์ชันร่วม เช่น ขั้นตอนของการเปรียบเทียบ $L(P)$ จะคล้ายคลึงกับขั้นตอน $\nabla g(x,y) > Thr$ ในอัลกอริทึมฟังก์ชันร่วม และ ขั้นตอน $L(P) \wedge L(P)$ จะคล้ายคลึงกับขั้นตอนการตัดสินค่า/ขาวกับจุดภาพที่อยู่ภายในขอบเขตของขอบ เป็นต้น

2. พารามิเตอร์ที่ใช้ในอัลกอริทึมรูปแบบนี้จะมี 2 ค่า คือ t กับ W ซึ่ง t คือ ค่าผลต่างของระดับความเข้มที่เป็นไปได้ได้ระหว่างตัวอักษรกับพื้นหลังในหน้าเอกสาร ส่วนค่า W คือ ขนาดความกว้างของเส้นตัวอักษร

3. ข้อจำกัดที่เกิดขึ้นกับอัลกอริทึมการแยกภาพตัวอักษรในรูปแบบนี้ คือ รูปแบบการทำงานของอัลกอริทึมนี้ไม่เหมาะสมสำหรับการประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร เพราะอัลกอริทึมนี้จะใช้ความกว้างของเส้นตัวอักษรเป็นค่าตัวแปรที่สำคัญสำหรับการประมวลผลในขั้นตอนการแยกภาพตัวอักษร

การประเมินรูปแบบของอัลกอริธึมการแยกภาพตัวอักษร

เกณฑ์ที่ใช้สำหรับการประเมินรูปแบบของอัลกอริธึมการแยกภาพตัวอักษรมีอยู่หลากหลายเกณฑ์ บางเกณฑ์นั้นอาจขึ้นอยู่กับการวิเคราะห์หลักการหรือทฤษฎีที่ใช้ในรูปแบบของอัลกอริธึมนั้น หรือบางเกณฑ์ขึ้นอยู่กับผลของการทดลองที่ได้ในแต่ละอัลกอริธึม ซึ่งเป็นไปได้ยากที่อัลกอริธึมรูปแบบหนึ่งจะมีคุณสมบัติที่เหมาะสมครบถ้วนตามเกณฑ์ที่ตั้งไว้ ในงานวิจัยนี้ได้กำหนดเกณฑ์หลัก ๆ ที่ใช้ในการประเมินอัลกอริธึมการแยกภาพตัวอักษรทั้งหมด 6 ข้อดังนี้

1 การประเมินจากภาพผลลัพธ์ ทำการประเมินความเหมาะสมของภาพผลลัพธ์ที่ได้ในแต่ละอัลกอริธึม ซึ่งหลักที่ใช้ในการตัดสินความเหมาะสมของภาพมีดังนี้ คือ การตรวจสอบความเหมาะสมของโครงสร้างของเส้นตัวอักษร ตรวจสอบความสมบูรณ์ของภาพตัวอักษร ตรวจสอบลักษณะของสิ่งที่ไม่พึงประสงค์ที่เกิดขึ้นกับภาพ เป็นต้น นอกจากนี้ยังสามารถทำการประเมินจากเปอร์เซ็นต์การจดจำตัวอักษรเมื่อนำภาพผลลัพธ์ที่ได้ไปใช้ในการประมวลผลภาพเอกสารในรูปแบบของการจดจำตัวอักษร

2. หน่วยความจำ การประเมินด้านหน่วยความจำของอัลกอริธึม นั้น จะวัดจากปริมาณของหน่วยความจำที่ใช้ในขั้นตอนการแยกภาพตัวอักษรเท่านั้น โดยไม่รวมถึงปริมาณของหน่วยความจำที่ใช้ในขั้นตอนการเตรียมข้อมูลภาพที่ได้มาจากขั้นตอนการสแกนภาพแต่อย่างใด โดยพิจารณาจากปริมาณของหน่วยความจำที่ใช้มากที่สุด ในขั้นตอนใดขั้นตอนหนึ่งของขั้นตอนการแยกภาพตัวอักษรในอัลกอริธึมนั้น และไม่คำนึงถึงประเภทของข้อมูลบนหน่วยความจำ

3. ความเร็ว การประเมินด้านความเร็วของอัลกอริธึมในงานวิจัยนี้ วัดจากเวลาที่ใช้ในการประมวลผลภาพเอกสาร นอกจากนี้ยังพบว่าอุปกรณ์ทางด้านฮาร์ดแวร์จะมีผลต่อความเร็วในขั้นตอนการทำงานอีกด้วย และที่สำคัญเทคนิคที่ใช้การเขียนโปรแกรมแอปพลิเคชันแยกภาพตัวอักษรก็มีผลต่อความเร็วในขั้นตอนการทำงานเหมือนกัน

4. ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร อัลกอริธึมการแยกภาพตัวอักษรบางอัลกอริธึมจะใช้ค่าความกว้างของเส้นตัวอักษรจะเป็นพารามิเตอร์ที่สำคัญสำหรับขั้นตอนการสร้างภาพผลลัพธ์ ซึ่งสิ่งนี้ถือว่าเป็นข้อจำกัดข้อหนึ่งของอัลกอริธึมที่จำเป็นต้องรู้ถึงข้อมูลที่เป็นจริงของภาพเอกสารก่อน เพื่อที่จะนำข้อมูลนั้นมาเป็นพารามิเตอร์ที่สำคัญเพื่อใช้ในขั้นตอนการ

ประมวลผลภาพ และข้อจำกัดที่เกิดขึ้นอีกข้อหนึ่งของอัลกอริทึมในรูปแบบนี้ คือ ไม่เหมาะที่จะนำไปประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร

5. จำนวนของพารามิเตอร์ อัลกอริทึมที่ใช้สำหรับแยกภาพตัวอักษรแต่ละอัลกอริทึมจะประกอบด้วยพารามิเตอร์ในรูปแบบต่าง ๆ เช่น อยู่ในรูปของสัมประสิทธิ์ของสมการเชิงเส้น เป็นต้น ส่วนความกว้างของเส้นตัวอักษรก็ถือว่าเป็นพารามิเตอร์ค่าหนึ่งในอัลกอริทึม จำนวนของพารามิเตอร์ในอัลกอริทึมสามารถแสดงถึงประสิทธิภาพของรูปแบบอัลกอริทึมได้อีกด้วย ซึ่งอัลกอริทึมที่ดีควรมีจำนวนพารามิเตอร์น้อย คือ การให้ข้อมูลของภาพเอกสารกับอัลกอริทึมน้อยที่สุด แต่อัลกอริทึมสามารถให้ภาพผลลัพธ์ที่มีความถูกต้องมากที่สุด

6. การกำหนดค่าพารามิเตอร์ ขั้นตอนการกำหนดค่าให้กับพารามิเตอร์สามารถนำไปใช้เป็นเครื่องมือสำหรับการประเมินความยุ่งยากของรูปแบบอัลกอริทึมได้ ซึ่งการกำหนดค่าให้กับพารามิเตอร์ในบางอัลกอริทึมจะทำได้ด้วยความยากลำบาก ดังเช่น การกำหนดค่าสัมประสิทธิ์ในสมการเชิงเส้นของฟังก์ชัน W_h และ W_v จำเป็นต้องทราบวัตถุประสงค์ของฟังก์ชันเสียก่อนและสามารถคาดการณ์ถึงลักษณะของค่าผลลัพธ์ที่ได้เป็นอย่างดี เพื่อให้การกำหนดค่าของพารามิเตอร์เป็นไปอย่างมีประสิทธิภาพ

สำหรับงานวิจัยนี้จะทำการประเมินรูปแบบอัลกอริทึมการแยกภาพตัวอักษรที่นำเสนอทั้ง 8 อัลกอริทึม พร้อมทั้งประเมินรูปแบบอัลกอริทึมการแยกภาพตัวอักษร Local Contrast ของ Giuliano, Paitra และ Stringa [23] ควบคู่กันไป โดยใช้เกณฑ์ในการประเมินทั้งหมด 6 ข้อ ดังที่กล่าวข้างต้น ซึ่งผลของการประเมินแสดงดังตารางที่ 1

สำหรับหลักการและผลการทดลองของอัลกอริทึมการแยกภาพตัวอักษร Local Contrast ของ Giuliano, Paitra และ Stringa [23] จะอ้างอิงจากภาคผนวก ก.

ตารางที่ 1

อัลกอริธึม	การประเมินภาพผลลัพธ์	หน่วยความจำ	ความเร็ว (วินาที)		ข้อจำกัดที่ขึ้นกับความกว้างของตัวอักษร	จำนวนพารามิเตอร์	การกำหนดค่าพารามิเตอร์
			ภาพ 1	ภาพ 2			
1	เหมาะสม	$(2*W+3)M$	4.5	41.2	ขึ้นกับความกว้างของตัวอักษร	2	ทำง่าย
2	เหมาะสม	0	1.4	4.2	ไม่ขึ้น	1	ทำง่าย
3	เหมาะสม	2MN	13.7	91.0	ไม่ขึ้น	1	ทำง่าย
4	เหมาะสม	2MN	5.8	50.0	ขึ้นกับความกว้างของตัวอักษร	2	ทำง่าย
5	เหมาะสม	2M	1.2	9.2	ความกว้างของเส้นตัวอักษรมีผลต่อลักษณะของภาพผลลัพธ์	2	ทำง่าย
6	เหมาะสม	2M	1.1	7.5	ความกว้างของตัวอักษรมีผลต่อลักษณะของภาพผลลัพธ์	3	ทำยาก
7	เหมาะสม	$(2w+1)M$	116	307	ไม่ขึ้น	2	ทำง่าย
8	เหมาะสม	$(4W-1)M$	94	123.7	ขึ้นกับความกว้างของตัวอักษร	2	ทำง่าย
9	เหมาะสม	9M	129	370	ความกว้างของตัวอักษรมีผลต่อลักษณะของภาพผลลัพธ์	4	ทำยาก

ตารางแสดงผลของการประเมินรูปแบบของอัลกอริธึมโดยใช้เกณฑ์ที่กำหนด

หมายเหตุ

ลำดับของอัลกอริธึมจะเรียงตามการนำเสนอในบทที่ 3 ส่วนอัลกอริธึมลำดับที่ 9 เป็นอัลกอริธึมของ Giuliano, Paitra และ Stringa [23]

สำหรับการประเมินด้านความเร็วจะวัดจากเวลาที่ใช้ในการประมวลผลภาพ ที่ทำการประมวลผลบนเครื่อง PC DX4 100MHz

สำหรับหน่วยที่ใช้บอกปริมาณการใช้งานหน่วยความจำนั้น ค่า M จะมีขนาดเท่ากับความกว้างของภาพเอกสาร (คือ หนึ่งแถวข้อมูลภาพ) ค่า N จะมีขนาดเท่ากับความยาวของภาพเอกสาร (คือ จำนวนแถวของข้อมูลภาพทั้งหมด)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ในการเรียนเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สรุปผลการประเมินรูปแบบอัลกอริธึมการแยกภาพตัวอักษร

จากการประเมินรูปแบบอัลกอริธึมการแยกภาพตัวอักษรทั้ง 8 อัลกอริธึม ที่นำเสนอในงานวิจัยชิ้นนี้ กับอีก 1 อัลกอริธึมการแยกภาพตัวอักษร ซึ่งเป็นผลงานวิจัย Giuliano, Paitra และ Stringa [23] โดยใช้กฎเกณฑ์การประเมินทั้งหมด 6 กฎเกณฑ์ (ดังที่กล่าวในหัวข้อที่ผ่านมา) ซึ่งผลของการประเมินแสดงดังตารางที่ 1 และจากผลการประเมินสามารถบอกลถึงข้อดีและข้อจำกัดที่เกิดขึ้นในแต่ละรูปแบบของอัลกอริธึมได้ ซึ่งสามารถสรุปได้ดังต่อไปนี้

อัลกอริธึมฟังก์ชันร่วม

ข้อดี

1. อัลกอริธึมฟังก์ชันร่วมจะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มและขนาดใกล้เคียงกับตัวอักษรเท่านั้น (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริธึมนี้ไม่เหมาะที่จะนำไปประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)

อัลกอริธึมการทำแตรคโครด์ตามคุณลักษณะของวัตถุ

ข้อดี

1. อัลกอริธึม OAT จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรมากที่สุดเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

2. อัลกอริธึม OAT เป็นอัลกอริธึมที่จัดได้ว่าเป็นแอลพลีเคชันความเร็วสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)

3. อัลกอริธึม OAT เป็นอัลกอริธึมที่ไม่มีปัญหาเรื่องการใช้พื้นที่หน่วยความจำสำหรับการประมวลผลภาพ (พิจารณาจากเกณฑ์ข้อ 2 การประเมินด้านหน่วยความจำ)

อัลกอริทึมการทำเรคโวลต์แบบปรับค่าตามเครื่องหมายจุดภาพของภาพลาปลาเซียน

ข้อดี

1. อัลกอริทึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริทึมนี้จะใช้เวลาในการประมวลผลก่อนข้างจะสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)
2. อัลกอริทึมนี้จะใช้พื้นที่หน่วยความจำสำหรับการประมวลผลภาพในปริมาณที่สูง (พิจารณาจากเกณฑ์ข้อ 2 การประเมินด้านหน่วยความจำ)

อัลกอริทึมการประเมินผลต่างภาพ

ข้อดี

1. อัลกอริทึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มและขนาดใกล้เคียงกับตัวอักษรเท่านั้น (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริทึมนี้ไม่เหมาะที่จะนำไปประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริทึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)
2. อัลกอริทึมนี้จะใช้พื้นที่หน่วยความจำสำหรับการประมวลผลภาพในปริมาณที่สูง (พิจารณาจากเกณฑ์ข้อ 2 การประเมินด้านหน่วยความจำ)

อัลกอริธึมการทำเรดโพลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มจุดภาพในแนว ราสเทอร์สแกน

ข้อดี

1. อัลกอริธึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)
2. อัลกอริธึมนี้จัดได้ว่าเป็นแอลพลิเคชันความเร็วสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)

ข้อจำกัด

1. อัลกอริธึมนี้ไม่เหมาะสมที่จะนำไปประมวลผลกับภาพเอกสารที่ความกว้างของเส้นตัวอักษรมีค่ามากกว่าช่วงห่างระหว่างตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)

อัลกอริธึมการทำเรดโพลด์แบบปรับค่าตามค่าเฉลี่ยระดับความเข้มจุดภาพ

ข้อดี

1. อัลกอริธึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)
2. อัลกอริธึมนี้จัดได้ว่าเป็นแอลพลิเคชันความเร็วสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)

ข้อจำกัด

1. อัลกอริธึมนี้ไม่เหมาะสมที่จะนำไปประมวลผลกับภาพเอกสารที่ความกว้างของเส้นตัวอักษรมีค่ามากกว่าช่วงห่างระหว่างตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)
2. การกำหนดค่าเริ่มให้กับพารามิเตอร์ในอัลกอริธึมนี้ทำได้ยาก (พิจารณาจากเกณฑ์ข้อ 6 การกำหนดค่าเริ่มต้นสำหรับพารามิเตอร์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อัลกอริธึมการทำเรคโพลด์แบบปรับค่าตามค่าเฉลี่ยและค่าความแปรปรวนของระดับความเข้มจุดภาพ

ข้อดี

1. อัลกอริธึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริธึมนี้จะใช้เวลาในการประมวลผลภาพสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)
2. อัลกอริธึมนี้จะใช้พื้นที่หน่วยความจำสำหรับการประมวลผลภาพในปริมาณที่สูง (พิจารณาจากเกณฑ์ข้อ 2 การประเมินด้านหน่วยความจำ)

อัลกอริธึมการประเมินค่าเฉลี่ยระดับความเข้มของจุดภาพรอบข้าง

ข้อดี

1. อัลกอริธึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริธึมนี้ไม่เหมาะที่จะนำไปประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)
2. อัลกอริธึมนี้จะใช้เวลาในการประมวลผลภาพก่อนข้างจะสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)

อัลกอริธึม Local Contrast [23]

ข้อดี

1. อัลกอริธึมนี้จะให้ภาพผลลัพธ์ที่มีความเหมาะสม สิ่งไม่พึงประสงค์ที่จะเกิดบนภาพผลลัพธ์ได้นั้น จะต้องเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มใกล้เคียงกับตัวอักษรเท่านั้น แต่ขนาดอาจมีความหลากหลาย (พิจารณาจากเกณฑ์ข้อ 1 การประเมินภาพผลลัพธ์)

ข้อจำกัด

1. อัลกอริธึมนี้ไม่เหมาะสมที่จะนำไปประมวลผลกับภาพเอกสารที่ความกว้างของเส้นตัวอักษรมีค่ามากกว่าช่วงห่างระหว่างตัวอักษร (พิจารณาจากเกณฑ์ข้อ 4 ข้อจำกัดของอัลกอริธึมที่ขึ้นกับความกว้างของเส้นตัวอักษร)
2. อัลกอริธึมนี้จะใช้เวลาในการประมวลผลค่อนข้างจะสูง (พิจารณาจากเกณฑ์ข้อ 3 การประเมินด้านความเร็ว)
3. การกำหนดค่าเริ่มให้กับพารามิเตอร์ในอัลกอริธึมนี้ทำได้ยาก (พิจารณาจากเกณฑ์ข้อ 6 การกำหนดค่าเริ่มต้นสำหรับพารามิเตอร์)

บทที่ 5

สรุปและข้อเสนอแนะ

สรุปผลงานวิจัย

วิทยานิพนธ์ฉบับนี้ได้ทำการวิจัยเกี่ยวกับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิคออกจากภาพเอกสารระดับความเข้มสีเทา (*grayscale*) เพื่อเก็บเป็นภาพไบนารีที่มีเพียงสองระดับ คือระดับความเข้มของตัวอักษรและสัญลักษณ์กราฟฟิค กับระดับความเข้มของพื้นหลัง การทำวิจัยในครั้งนี้ได้นำเสนออัลกอริธึมทั้งหมด 8 อัลกอริธึม เพื่อใช้สำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิค ภาพผลลัพธ์ที่ได้ในแต่ละอัลกอริธึมอาจมีลักษณะที่ใกล้เคียงหรือแตกต่างกันไป ขึ้นอยู่กับหลักการที่ใช้ในแต่ละอัลกอริธึม ดังนั้นในบทนี้ของวิทยานิพนธ์จะทำการสรุปเนื้อหาหลัก ๆ ของงานวิจัยดังต่อไปนี้

1. อัลกอริธึมที่ใช้สำหรับแยกภาพตัวอักษรที่นำเสนอในงานวิจัยชิ้นนี้ทั้งหมดจะอยู่บนพื้นฐานของการเซกเมนต์ภาพโดยการทำเรดโซลต์กับจุดภาพ โดยอัลกอริธึมฟังก์ชันร่วม และอัลกอริธึมการทำเรดโซลต์ตามคุณลักษณะของวัตถุอยู่บนพื้นฐานของการทำเรดโซลต์แบบครอบคลุม (*global thresholding*) ส่วนอัลกอริธึมที่เหลืออีก 6 อัลกอริธึมอยู่บนพื้นฐานของการทำเรดโซลต์แบบปรับค่า (*adaptive thresholding*) โดยปกติทั่วไปแล้วเทคนิคการทำเรดโซลต์แบบครอบคลุมไม่เหมาะสำหรับขั้นตอนการแยกภาพตัวอักษรออกจากภาพเอกสารที่มีความซับซ้อนของส่วนพื้นหลัง ดังนั้นอัลกอริธึมฟังก์ชันร่วมและอัลกอริธึมการทำเรดโซลต์ตามคุณลักษณะของวัตถุจึงนำเทคนิคการทำซ้ำ (*iterative thresholding*) มาประยุกต์เพื่อใช้สำหรับการปรับค่าโกลบอลเรดโซลต์ให้มีความเหมาะสมกับภาพมากที่สุด และทำให้การแยกภาพตัวอักษรออกจากภาพเอกสารระดับความเข้มเป็นไปอย่างมีประสิทธิภาพมากที่สุด

2. อัลกอริธึมการแยกภาพตัวอักษรในรูปแบบที่ใช้ค่าความกว้างของเส้นตัวอักษรเป็นเกณฑ์ในการพิจารณาคูณสมบัติของจุดภาพในขั้นตอนการแยกภาพตัวอักษร ไม่เหมาะสำหรับการเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมวลผลกับภาพเอกสารที่มีความหลากหลายของขนาดตัวอักษร ส่วนรูปแบบอัลกอริทึมที่มีการใช้ค่าผลต่างของระดับความเข้มระหว่างตัวอักษรกับพื้นหลังเป็นเกณฑ์ในการพิจารณาคุณสมบัติของจุดภาพในขั้นตอนการแยกภาพตัวอักษร ไม่เหมาะสำหรับการประมวลผลกับภาพเอกสารที่มีความหลากหลายของระดับความเข้มตัวอักษร สำหรับรูปแบบอัลกอริทึมที่มีการใช้พารามิเตอร์ทั้งสองค่าข้างต้นร่วมกันในการพิจารณาคุณสมบัติจุดภาพจะให้ภาพผลลัพธ์ที่มีความถูกต้องในระดับที่สูง สิ่งไม่พึงประสงค์ที่จะเกิดขึ้นบนภาพผลลัพธ์ได้นั้นจะเป็นสิ่งไม่พึงประสงค์ที่มีระดับความเข้มและขนาดใกล้เคียงกับตัวอักษร

3. ข้อจำกัดที่เกิดขึ้นกับอัลกอริทึมที่อยู่บนพื้นฐานของการทำเรดโซลด์แบบปรับค่าที่การประมวลผลในแต่ละจุดภาพต้องอ้างอิงจุดภาพรอบข้าง คือ ความเร็วที่ใช้ในการประมวลผลต่ำ

การวิจัยในครั้งนี้มุ่งเน้นไปที่การนำเสนอหลักการที่สามารถใช้ในการแยกภาพตัวอักษรตัวอักษรและสัญลักษณ์กราฟฟิคออกจากภาพเอกสาร เพื่อเก็บเป็นภาพไบนารี ซึ่งในแต่ละหลักการที่นำเสนอได้ทำการทดสอบกับภาพเอกสารหลากหลาย โดยเขียนโปรแกรมการทำงานด้วยหลักการดังกล่าวเป็นภาษา *Borland C3.1 for Window* บนเครื่อง *PC* และทำการประมวลผลกับภาพเอกสารที่อยู่ในรูปของแฟ้มข้อมูลรูปภาพ ที่มี *Format* เป็น *TIFF* และมีรายละเอียดของจุดภาพเป็นระดับความเข้มสีเทา 256 ระดับ

ข้อเสนอแนะ

การนำเสนอหลักการที่ใช้สำหรับการแยกภาพตัวอักษรและสัญลักษณ์กราฟฟิคออกจากภาพเอกสารที่มีความซับซ้อนของพื้นหลังในงานวิจัยนี้ อาจเป็นแนวทางสำหรับผู้สนใจที่จะทำวิจัยเกี่ยวกับการแยกตัวอักษรออกจากภาพเอกสาร โดยการพัฒนารูปแบบให้มีประสิทธิภาพสูงขึ้น ให้ภาพผลลัพธ์ที่มีความถูกต้องสูง ข้อจำกัดของหลักการที่ใช้น้อยลง และขั้นตอนการทำงานเป็นแบบเร็วไหม้จริง ๆ รวมถึงการพัฒนารูปแบบการแยกภาพตัวอักษรออกจากสัญลักษณ์กราฟฟิคด้วย

เอกสารอ้างอิง

- [1] ยุทธพงษ์ รั้งสรรค์เสรี, สิทธิศักดิ์ รอดทุกข์, “การแยกภาพตัวอักษรโดยอัลกอริทึมฟังก์ชันร่วม,” *การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18*, 2538, หน้า 939-942
- [2] ยุทธพงษ์ รั้งสรรค์เสรี, สิทธิศักดิ์ รอดทุกข์, “การทำเรดโซลต์ภาพตามคุณลักษณะของวัตถุ,” *การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18*, 2538, หน้า 961-964
- [3] ยุทธพงษ์ รั้งสรรค์เสรี, “การออกแบบเคอร์เนลไบโนเมียลด้วยเงื่อนไขสอดคล้องกับคุณสมบัติของเกาส์เซียนสองมิติ,” *การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 17*, 2537
- [4] D. Marr, E. Hildreth, “Theory of edge detection,” *Proceeding of Royal Society of London*, vol 207, pp 187-212, 1980
- [5] J.M. White, G.D. Roher, “Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction,” *IBM J. Res. Dev*, vol 27(4), pp. 400-411, July 1983
- [6] N. Otsu, “A Threshold Selection Method From Gray-Level Histograms,” *IEEE Trans Systems Man Cybernet*, SMC-9, pp 62-69, 1979
- [7] P.K. Sahoo, S. Sotani, A.K.C Wong, “A Survey of Thresholding Technique,” *Computer Vision Graphics and Image Processing*, vol 41, pp. 230-260, 1988
- [8] J.S. Weszka, A. Rosenfeld, “Histogram modification for threshold selection,” *IEEE Trans Systems Man Cybernet*, pp 38-52, Jan 1979
- [9] J.S. Weszka, A. Rosenfeld, “Thresholding evaluation techniques,” *IEEE Trans Systems Man Cybernet*, vol 8, pp 630-632, Aug 1978.
- [10] Hon-Son Don, “A Noise Attribute Thresholding Method for Document Image Binarization,” *ACCV'95 SECOND ASIAN CONFERENCE ON COMPUTER VISION*
- [11] K.S. Fu, J.K.Mui, “A survey of threshold techniques,” *Pattern Recognition*, vol 13, pp. 13-16, 1981
- [12] T.W. Ridler, S. Calvard, “Picture Thresholding Using an Iterative Selection Method,” *IEEE Trans Systems Man Cybernet*, vol 8, pp 622-629, Aug 1978.

- [13] D.G. Elliman, I.T. Lancaster, "A Review of segmentation and contextual analysis techniques for text recognition", *Pattern Recognition*, vol 23, pp 337-346, 1990
- [14] B. chanda, B.B. Chaudhri, "A Modified scheme for segmenting the Noisy images," *IEEE Trans Systems Man Cybernet*, vol 18, No 3, pp 337-346, 1990.
- [15] W.K. Pratt, *Digital Image Processing*, Second Edition, USA Jon Wiley & Sons Inc, 1991
- [16] R.C. Gonzalez, P. Wintz, *Digital Image Processing*, Secon Edition, Addison-Wesley, Reading, MA, 1987
- [17] R.M. Haralick, L.G. Shapiro, *Computer and Robot Vision*, vol 1, Addison Wesley, Reading, 1992
- [18] R.M. Haralick, L.G. Shapiro, *Computer and Robot Vision*, vol 2, Addison Wesley, Reading, 1992
- [19] A. Rosenfeld, A.C. Kak, *Digital Picture Processing*, Second Edition, Vol 1, Academic Press, 1982
- [20] R. Jain, R. Kasturi, B.G. Schunck, *Machine Vision*, Mcgraw-Hill International Editions, 1995.
- [21] ยอดเยี่ยม ปราโมทย์ภิกขุ, การจำแนกภาพโดยวิธีเซ็กเมนต์, วทานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรมหาบัณฑิต, สาขาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2532
- [22] สาธิต อินทจักร์, การหาขอบภาพโดยใช้แบบจำลองฟัซซี่และนิเวรอลเน็ตเวิร์ค, วทานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรมหาบัณฑิต, สาขาวิศวกรรมไฟฟ้า สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2538
- [23] E. Giulino, O. Paitra, and L. Stringa, "Electronic Character Reading System," U.S. Patent 4,047,15 (Sept. 6,1977)



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

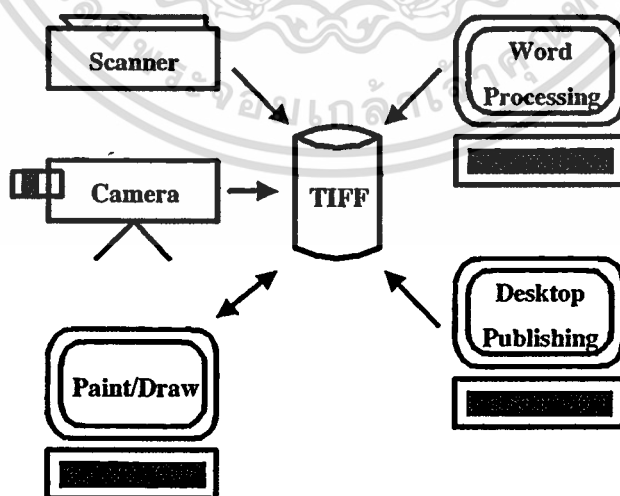
ภาคผนวก ก

เพิ่มข้อมูลกราฟิกรูปแบบ TIFF Version 5.0

การรวมของเพิ่มข้อมูลกราฟิกรูปแบบ TIFF

เพิ่มข้อมูล Tag Image File Format หรือเรียกสั้นๆ ว่า TIFF ถูกพัฒนาขึ้นโดยบริษัท Aldus Corporation เพื่อใช้จัดเก็บข้อมูลภาพแบบ Bit-map ซึ่งพบว่าโปรแกรมด้าน Desktop-publishing เกือบทั้งหมด รวมทั้งโปรแกรมสำหรับสแกนเนอร์ต่างสนับสนุน TIFF ทั้งนี้ ซึ่งมีความสามารถในการจัดเก็บรายละเอียดทั้งหมดของภาพไว้ได้ ทำให้ TIFF เป็นที่นิยมและถูกพัฒนาต่อขึ้นมาเป็นซูเปอร์เซตของรูปแบบเพิ่มข้อมูลอื่นๆ ที่ใช้เก็บภาพ

ภาพที่ 64



ฟังก์ชันของ scanned image ในงาน Desktop Publishing

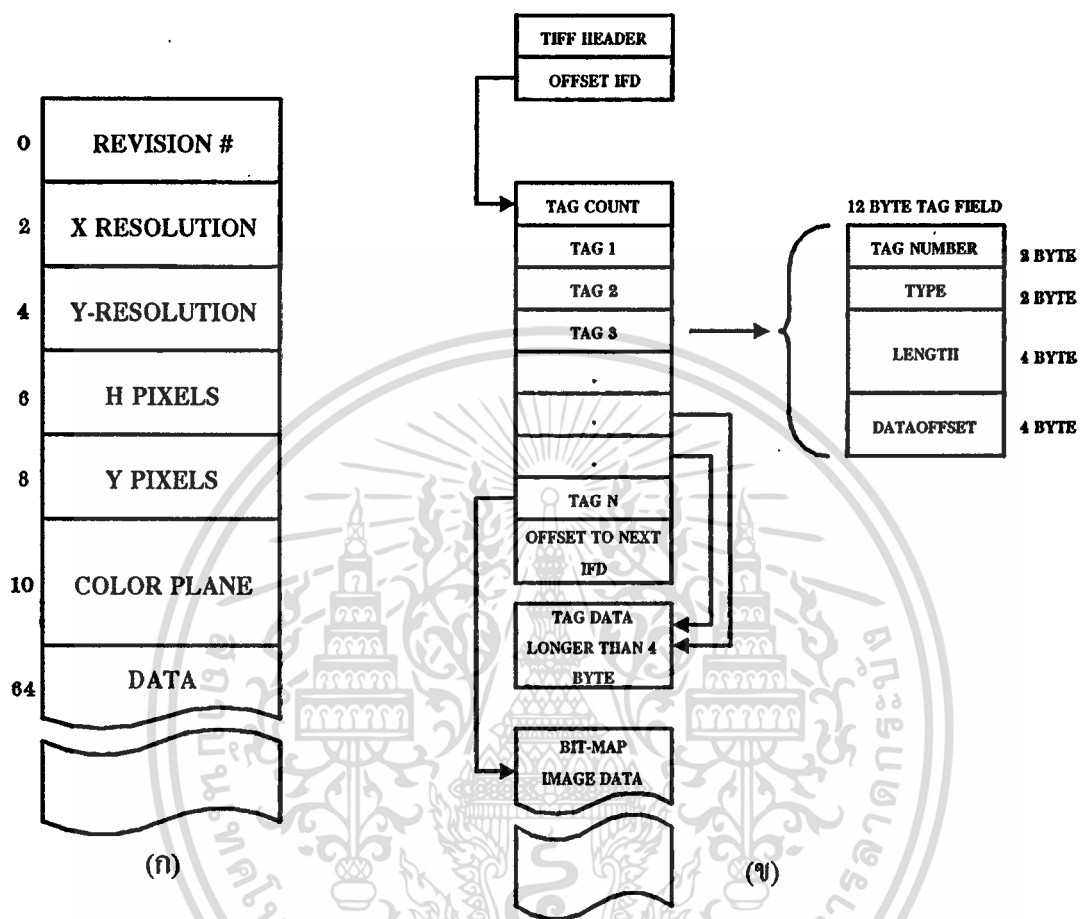
ข้อแตกต่างอย่างหนึ่งที่ TIFF ต่างจากแฟ้มข้อมูลรูปแบบอื่นๆ คือ ข้อมูลส่วนใหญ่ที่เก็บจะไม่ถูกกำหนดตำแหน่งแน่นอนตายตัว โดยมีเพียงข้อมูลเพียง 8 ไบต์แรกเท่านั้นที่เป็นตำแหน่งพิเศษและเก็บไว้ที่ตำแหน่งนี้เสมอ ส่วนอื่นๆ ที่เหลือจะถูกอ้างอิงกับส่วนเริ่มต้นของแฟ้มข้อมูลจึงเป็นการง่ายต่อการเพิ่มเติมข้อมูล

เดิมทีแล้วแฟ้มข้อมูลรูปภาพโดยทั่วๆ ไป จะใช้รูปแบบแฟ้มข้อมูลที่แน่นอน (Fixed-format organization) ซึ่งสามารถทราบถึงตำแหน่งโครงสร้างและตำแหน่งของข้อมูลได้ก่อนการเปิดอ่านแฟ้มข้อมูล ยกตัวอย่างเช่น X-resolution จะเก็บอยู่ตำแหน่งไบต์ 2 และ 3 และส่วนของข้อมูลจะเริ่มต้นที่ตำแหน่งไบต์ที่ 64 เป็นต้น

สำหรับแฟ้มข้อมูล TIFF จะใช้เทคนิคที่แตกต่างกัน โดยแบ่งเป็น Tags เก็บรายละเอียดของข้อมูล ในแต่ละ Tags จะมีความยาว 12 ไบต์ และอธิบายลักษณะในแต่ละส่วนของภาพ ตัวอย่างเช่น ในแฟ้มข้อมูลจะประกอบด้วย Tag สำหรับ X และ Y Resolution และ Tag สำหรับเก็บตำแหน่งของข้อมูล (StripOffsets)

หากพิจารณาถึงโครงสร้างของแฟ้มข้อมูล TIFF พบว่ามีการจัดเก็บรายละเอียดหลายอย่างที่ยังไม่มีความจำเป็น แต่ในทางกลับกันพบว่ากลายเป็นข้อได้เปรียบกว่าแฟ้มข้อมูลแบบตายตัว โดยเมื่อต้องการเพิ่มเติม Tag ใหม่ก็ไม่จำเป็นต้องเขียนโปรแกรมในส่วนจัดเก็บใหม่ ตัวอย่างเช่น ถ้าเพิ่ม Tag ที่ใช้เก็บวันเวลาที่บันทึกแฟ้มข้อมูล (DateTime) ในขณะที่โปรแกรมยังไม่ Support ข้อมูลของ Tag DateTime โอกาสที่จะเกิดข้อผิดพลาดก็จะไม่เกิดขึ้นในกรณีที่มี Tag ที่ยังไม่มีการใช้งาน (Non-support Tag) อยู่ในแฟ้มข้อมูล ต่างกับแฟ้มข้อมูลแบบตายตัวเมื่อเพิ่มเติมข้อมูลเข้าไปจะต้องเขียนโปรแกรมจัดการใหม่ และเมื่อโปรแกรมที่ใช้งานเป็นโปรแกรมรุ่นเก่า ซึ่งเมื่อต้องการข้อมูลบางอย่างตรงตำแหน่งที่กำหนด กลับพบว่าข้อมูลในตำแหน่งนั้นเปลี่ยนไป หรือหาข้อมูลไม่พบเลยก็ได้ ผลลัพธ์ที่ได้ก็คือ การทำงานที่ผิดพลาด

ภาพที่ 65



(ก) ตัวอย่าง Fixed-file format

(ข) ตัวอย่าง โครงสร้างทั่วไปของ TIFF

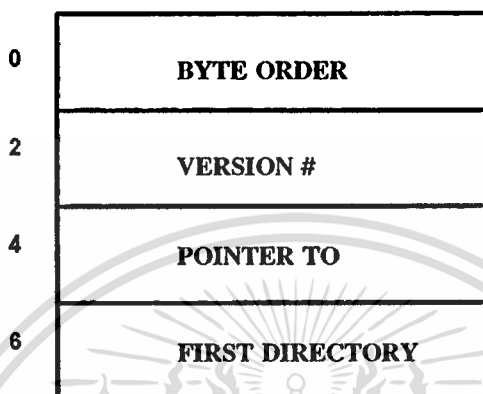
โครงสร้างทั่วไปของ TIFF

แฟ้มข้อมูลภาพที่อยู่ในรูปแบบของ TIFF หนึ่งๆ จะประกอบด้วยส่วนประกอบหลัก 3 ส่วน ดังต่อไปนี้

1. Header File

ในส่วนหัวนี้将有ความยาว 8 ไบท์ ประกอบด้วยข้อมูลที่แสดงดังภาพต่อไปนี้

ภาพที่ 66



TIFF Header

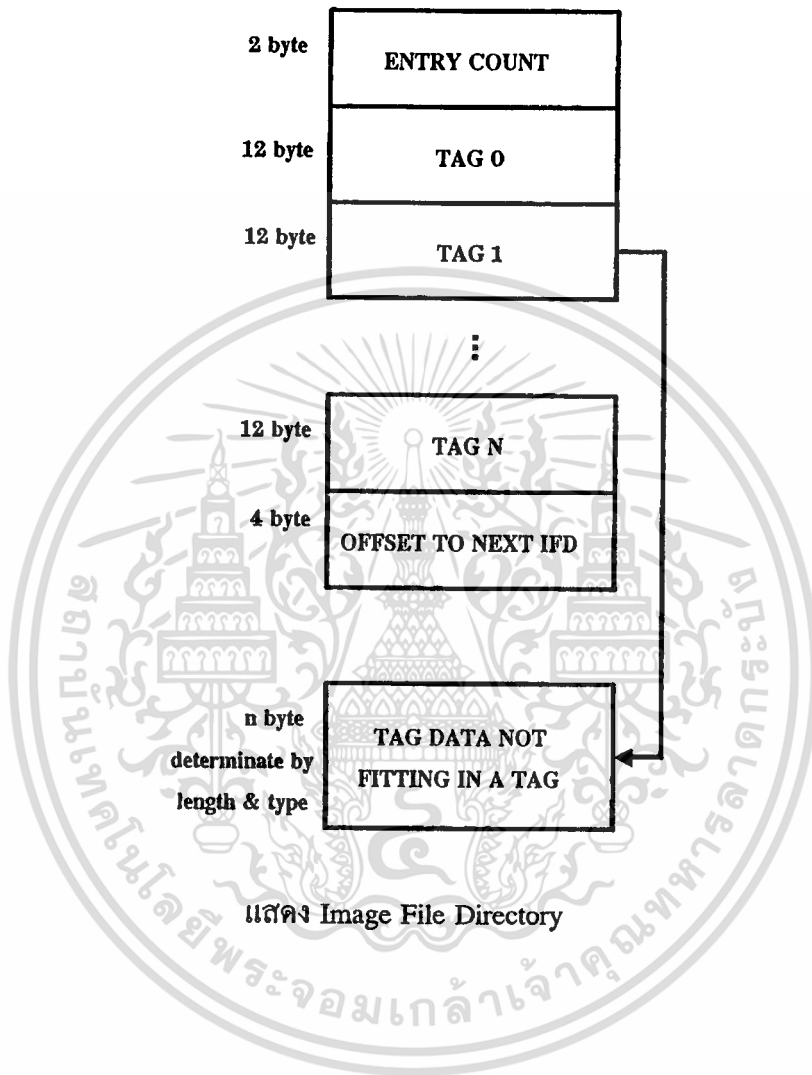
2. The Image File Directory (IFD)

ในส่วนนี้จะเป็นส่วนของการแนะนำลักษณะต่างๆ ในแฟ้มข้อมูล ซึ่ง IFD จะประกอบด้วย 3 ส่วน คือ Entry count, Tags ต่างๆ ซึ่งแต่ละอันมีความยาว 12 ไบต์เรียงต่อกัน และส่วนที่ชี้ไปยังตำแหน่ง IFD ชุดต่อไป ซึ่งส่วนนี้ยาว 4 ไบต์ ซึ่งทั้ง 3 ส่วนจะแสดงดังภาพที่ 64

3. The Tags Field

แฟ้มข้อมูลภาพหนึ่งๆ สามารถที่จะมี Tag ที่แตกต่างกันได้ถึง 50 Tags โดยแต่ละ Tag จะมีความยาว 12 ไบต์ และประกอบด้วย 4 ส่วน คือ Tag number, Type of tag value, Number of values in tag และ Tag value of offset

ภาพที่ 67



แสดง Image File Directory

ภาพที่ 68

Tag #	Type	Length	Data/Offset
2 Bytes	2 Bytes	4 Bytes	4 Bytes

แสดง 12 bytes of a TIFF tag

ตัวอย่าง Tag ที่สำคัญและคำอธิบาย

1. The ImageWidth Tag

ImageWidth Tag = 256(100H)

Tag = SHORT or LONG

Number of distinct values = 1

Value range : All positive integers in Type

ImageWidth บอกถึงความกว้างของรูปภาพที่มีหน่วยเป็น จุดภาพ (Pixels) ความกว้างจะยึดตามแกน X หรือตามแนวนอน ความกว้างของภาพอาจเรียกอีกอย่างว่าเป็นจำนวนคอลัมน์ในรูปภาพ ซึ่ง Tag นี้เป็น Tag ที่สำคัญ เพราะจะไม่สามารถอ่านรูปภาพได้เลขหากไม่ทราบความกว้างของรูปเสียก่อน

2. The ImageLength Tag

ImageLength Tag = 257(101H)

Type = SHORT or LONG

Number of distinct value = 1

Value range : All positive integer in Type

ImageLength บอกถึงความยาวของรูปภาพที่มีหน่วยเป็น จุดภาพ (Pixels) ความยาวจะยึดตามแกน Y หรือตามแนวตั้ง ความยาวของภาพอาจเรียกอีกอย่างว่าเป็นจำนวนแถวในรูปภาพ ซึ่ง Tag นี้เป็น Tag ที่สำคัญ เพราะจะไม่สามารถอ่านรูปภาพได้เลขหากไม่ทราบความยาวของรูปเสียก่อน

3. The StripOffsets Tag

StripOffset Tag = 273(111H)

Type = SHORT or LONG

Number of distinct values

= one per strip for single plane bitmaps

= number of planes x strip/plane for multi-plane bitmaps

Value range : All positive integers in Type

StripOffset เป็น Tag ที่สำคัญและจำเป็นต้องทราบค่า เพราะ Tag เพราะ Tag นี้จะบอกว่าส่วนของรูปภาพเริ่มต้นที่ตำแหน่งใด

ในการเปิดอ่านแฟ้มข้อมูลรูปภาพในรูปแบบของ TIF นี้จำเป็นที่จะต้องทราบค่าที่สำคัญอย่างน้อย 3 ค่า คือ ความกว้าง ความยาว และตำแหน่งของข้อมูลภาพที่เก็บในแฟ้มข้อมูลชนิดนี้ ซึ่งก็คือ ค่าต่างๆ ที่อยู่ใน Tag ทั้ง 3 ดังที่กล่าวมาแล้วข้างต้นนั่นเอง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

การคำนวณหาค่าเรดโชลต์โดยใช้อัลกอริธึมของ Otsu

หลักการทํางาน

อัลกอริธึมของ Otsu [6] เป็นอัลกอริธึมที่ใช้ในการหาค่าเรดโชลต์ของภาพ โดยคำนวณหาจากฮิสโตแกรมระดับเข้มของภาพ ค่าเรดโชลต์ที่ได้จะเป็นแบ่งระดับความเข้มของภาพออกเป็น 2 กลุ่มอย่างเหมาะสม ซึ่งการปฏิบัติตามอัลกอริธึมของ Otsu สามารถกระทำได้ดังหลักการต่อไปนี้

ทำการอนุมานว่าได้ทำการแบ่งระดับความเข้มออกเป็น 2 กลุ่มคือ C_0 และ C_1 โดยค่าเรดโชลต์ค่าหนึ่งทีระดับเข้ม k ซึ่ง C_0 แสดงถึงจุดภาพที่อยู่ในช่วงระดับความเข้ม $[0, k-1]$ และ C_1 แสดงถึงจุดภาพที่อยู่ในช่วงระดับความเข้ม $[k, L-1]$ และให้ σ_w^2 (with-in class variance) คือ ความแปรปรวนที่เกิดขึ้นภายในกลุ่ม, σ_B^2 (between-class variance) คือ ความแปรปรวนที่เกิดขึ้นระหว่างกลุ่ม, σ_T^2 (total variance of levels) คือ ความแปรปรวนของระดับความเข้มทั้งหมด ส่วนการประเมินความเหมาะสมของค่าเรดโชลต์ (ทีระดับความเข้ม k) จะใช้บรรทัดฐานในการประเมินดังต่อไปนี้

$$\lambda = \frac{\sigma_B^2}{\sigma_w^2} \quad (1)$$

$$\beta = \frac{\sigma_T^2}{\sigma_w^2} \quad (2)$$

$$\eta = \frac{\sigma_B^2}{\sigma_T^2} \quad (3)$$

โดยที่

$$w_0 = \sum_{i=0}^k p_i = w(k) \quad (4)$$

เอกสารนี้เป็นเอกสารที่ $w_1 = 1 - w(k)$ เป็นการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไป (5) โยชน์ด้านการค้า
ไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\mu_0 = \sum_{i=0}^{k-1} \frac{ip_i}{w_0} \quad (6)$$

$$\mu_1 = \sum_{i=k}^{L-1} \frac{ip_i}{w_1} \quad (7)$$

$$\mu_T = \sum_{i=0}^{L-1} ip_i \quad (8)$$

$$\sigma_0^2 = \frac{\sum_{i=0}^{k-1} (1-\mu_0)^2 p_i}{w_0} \quad (9)$$

$$\sigma_1^2 = \frac{\sum_{i=k}^{L-1} (1-\mu_1)^2 p_i}{w_1} \quad (10)$$

$$\sigma_B^2 = w_0 w_1 (\mu_0 - \mu_1)^2 \quad (11)$$

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu_T)^2 p_i \quad (12)$$

$$\sigma_W^2 = w_0 \sigma_0^2 + w_1 \sigma_1^2 \quad (13)$$

จากสมการที่ (1), (2), (3) พบว่าค่า σ_W^2 และ σ_B^2 เป็นฟังก์ชันที่ขึ้นกับค่า k แต่ค่า σ_T^2 จะไม่ขึ้นกับค่า k เพราะฉะนั้นค่า η จึงเป็นเครื่องมือในการประเมินผลที่ง่ายที่สุดที่ขึ้นกับค่า k ดังนั้นจึงนำค่า η มาเป็นเครื่องมือสำหรับการประเมินความเหมาะสมของค่าเรดโชลด์ที่ระดับความเข้ม k ซึ่งค่าเรดโชลด์ที่เหมาะสมจะขึ้นอยู่กับค่า η หรือ σ_B^2 ที่มากที่สุด โดยทำการตรวจสอบจากสมการต่อไปนี้

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_T^2} \quad (14)$$

$$\sigma_B^2(k) = \frac{(\mu_T w(k) - \mu(k))^2}{w(k)(1-w(k))} \quad (15)$$

ดังนั้นค่าเรดโชลด์ที่เหมาะสม คือ ค่าระดับเข้ม (k^*) ที่

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ของคณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก

การแยกภาพตัวอักษรโดยอัลกอริธึม Local Contrast Technique

อัลกอริธึม Local Contrast Technique เป็นอัลกอริธึมที่ Guiliano, Paitra และ Stringa [23] ทำการพัฒนาขึ้นมาเพื่อใช้สำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารที่มีความหลากหลาย ในส่วนของพื้นหลัง ซึ่งการทำงานของอัลกอริธึมดังกล่าวจะอยู่บนพื้นฐานของการทำเรดโซลต์ แบบปรับค่า ซึ่งการพิจารณาคุณสมบัติของจุดภาพในแต่ละจุดภาพจะต้องมีการอ้างอิงกับจุดภาพ รอบข้างในพิกัดที่แสดงดังภาพที่ 69 สำหรับหลักการการทำงานของอัลกอริธึม Local Contrast Technique นี้สามารถอธิบายในรูปของอัลกอริธึมดังต่อไปนี้

Begin

if $(g(x,y) < T_1)$ *then* $b(x,y) = 1$

else

$a_1 =$ Average of 9 pixels in area A_1

$A_{2t} = \{(x,y) \mid (x,y) \text{ in } A_2 \text{ and } g(x,y) > T_2\}$

$a_2 =$ Average of pixels in area A_{2t}

if $((T_3 * a_2) + T_5) > (T_4 * a_1)$ *then* $b(x,y) = 1$

else $b(x,y) = 0$

End

โดยที่

T_1 เป็นพารามิเตอร์ที่กำหนดขึ้นมาเพื่อใช้สำหรับการคัดแยกจุดภาพที่มีระดับความเข้มน้อยกว่า (จุดภาพของตัวอักษร)

T_2 เป็นพารามิเตอร์ที่กำหนดขึ้นมาเพื่อใช้สำหรับการสืบหาจำนวนจุดภาพทั้งหมดใน A_2 ที่มีระดับความเข้มมากกว่า T_2

T_3, T_4, T_5 พารามิเตอร์ทั้งสามนี้เปรียบเหมือนสัมประสิทธิ์ที่ใช้ในการถ่วงน้ำหนักค่า a_1 กับ a_2



ภาพที่ 70

Image Thresholding for Applications Requiring Character Image Extraction

(ก)

เครื่องโน้ตบุ๊กที่ใช้โปรเซสเซอร์แบบ clock-tripled มีเครื่องเดียวคือ Ascentia 900N ของ AST ซึ่งใช้ 486DX4, 75 เมกะเฮิร์ตซ์ ทำให้มีประสิทธิภาพเหนือกว่าเครื่องที่ใช้โปรเซสเซอร์ความเร็ว 50 เมกะเฮิร์ตซ์อื่น ๆ จากการทดสอบเบนช์มาร์คคะแนนที่ดีเยี่ยมเป็นของ Pontégé T3600CT ของ Toshiba ซึ่งเป็นคะแนนของอายุงานของแบตเตอรี่ซึ่งยาวนานถึง 6 ชั่วโมง 27 นาที เราใช้โปรแกรม PCBench 8.0, WinBench 4.0, Winstone 94 Version 1.0 และ ZDigit 2.0 จากห้องปฏิบัติการของ PC Magazine เพื่อเปรียบเทียบประสิทธิภาพของเครื่องโน้ตบุ๊กเหล่านี้ ■

	Winstone (score)	Processor (score)	Graphics WinMark (millions of pixels per second)	VMeo (score)	Disk WinMark (score)	Disk (score)	DOSmark (score)	Battery Life (hours: minutes)
AST Ascentia 900N(DX4/75)	40.1	30.9	6.5	3,470	774	857	497	4:35
IBM ThinkPad 510C (IBM BL 50)	25.8	18.8	4.5	3,426	456	697	331	3:14
NEC Versa S/50 (Dx2/50)	30	25.9	3.2	3,602	590	871	441	1:51
Toshiba T2400CT (Dx2/50)	30.3	26.9	3.6	2,448	857	860	437	3:51
Toshiba T3600 CT (Dx2/50)	31.8	26.7	3.6	2,449	820	823	431	6:27

(ข)

แสดงภาพผลลัพธ์โดยอัลกอริทึม Local Contrast

(ที่ $T_1 = 40, T_2 = 40, T_3 = 1, T_4 = 1, T_5 = -3$)

(ก) ภาพ Document 1 (ข) ภาพ Document 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การแยกภาพของตัวอักษรโดยอัลกอริธึมแบบฟังก์ชันร่วม

Character Image Extraction using Integrated Function Algorithm

ดร.ยุทธพงษ์ วังสรรค์เสรี • สิทธิศักดิ์ รอดทุกข์*

บทคัดย่อ

การแยกภาพของตัวอักษรจากเอกสารใดๆ โดยวิธีการเรขาคณิตสามารถยอมรับได้ในกรณีที่ตัวอักษรพิมพ์ด้วยสีเข้ม แต่ต่างชัดเจนจากพื้นหลังของกระดาษ สำหรับกรณีของเอกสารที่มีคุณภาพการพิมพ์ต่ำ วิธีนี้อาจให้ผลลัพธ์ที่ผิดพลาดได้ บทความวิจัยนี้จะเสนอวิธีการแยกภาพตัวอักษรที่มีคุณภาพสูงสำหรับเอกสารทุกประเภท โดยการประยุกต์ใช้โอเปอเรเตอร์พื้นฐาน เช่น กราเดียนท์ ลาปลาเซียน และอื่นๆ ร่วมกัน ผลที่ได้เรียกว่าอัลกอริธึมแบบฟังก์ชันร่วม ที่สามารถแยกข้อความตัวอักษรออกจากพื้นหลังของกระดาษคุณภาพต่างๆ ได้เป็นอย่างดี

ABSTRACT

The character image extraction by mean of thresholding produces acceptable results for documents exhibiting high contrast between characters and background. This method may fail in case of low-quality documents. This paper will present an algorithm that integrates a set of basic operators such as gradient, laplacian, etc. The resulting algorithm called Integrated Function Algorithm is capable of extracting idealized binary images of characters from complex background of documents.

1. บทนำ

การแยกภาพของตัวอักษรจากเอกสารกระดาษ เพื่อเก็บเป็นภาพไบนารีสำหรับการประมวลผลโดยคอมพิวเตอร์ เช่น สำหรับการรู้จำตัวอักษร (Optical Character Recognition) โดยทั่วไปสามารถกระทำได้โดยการสแกนภาพด้วยเครื่องสแกนเนอร์ซึ่งถูกเลือกให้ทำงานในโหมดภาพไบนารี หลักการทำงานในโหมดนี้หรือซอฟต์แวร์ที่ใช้กันอยู่ทั่วไป อาศัยการเปรียบเทียบระดับของ

แต่ละพิกเซลกับเรขาคณิตซึ่งเป็นค่าคงที่ สำหรับตัดสินใจว่าพิกเซลนั้นจะเป็นสีขาว (พื้นหลัง) หรือสีดำ (ตัวอักษร)

ภาพไบนารีที่ได้จากวิธีดังกล่าวจะมีคุณภาพดีในกรณีที่เอกสารเริ่มต้นมีคุณภาพการพิมพ์ (หรือเขียน) ที่ดีเท่านั้น เช่น เอกสารที่พิมพ์จากเครื่องพิมพ์เลเซอร์บนกระดาษขาวเป็นต้น ส่วนกรณีของเอกสารที่มีคุณภาพการพิมพ์ต่ำ เช่น พิมพ์จากเครื่องพิมพ์ดอตเมตริกด้วยริบบอนที่ซีดจาง มีการใช้กระดาษสีหรือลวดลาย หรือมีการใช้สีระบายเพื่อเน้นข้อความบางส่วน เป็นต้น การใช้เรขาคณิตคงที่ค่าเดียวมักจะไม่ให้ผลลัพธ์ที่มีคุณภาพเพียงพอสำหรับประมวลผลต่อไปได้

ในบทความนี้ เราได้ศึกษาวิธีการแก้ปัญหาดังกล่าว โดยการพัฒนาอัลกอริธึมสำหรับการแยกภาพของตัวอักษรจากภาพเริ่มต้น ซึ่งเป็นภาพระดับเทาขนาด 8 บิต ซึ่งจะให้ผลลัพธ์เป็นภาพไบนารีคุณภาพสูง ที่มีค่าของพิกเซลที่อยู่ในส่วนของตัวอักษรเป็นสีดำ และพิกเซลที่ปรากฏบนพื้นหลังของกระดาษเป็นสีขาว เหมาะสำหรับการประมวลผลต่อไปในงานด้านวิเคราะห์เอกสาร (Document Image Analysis)

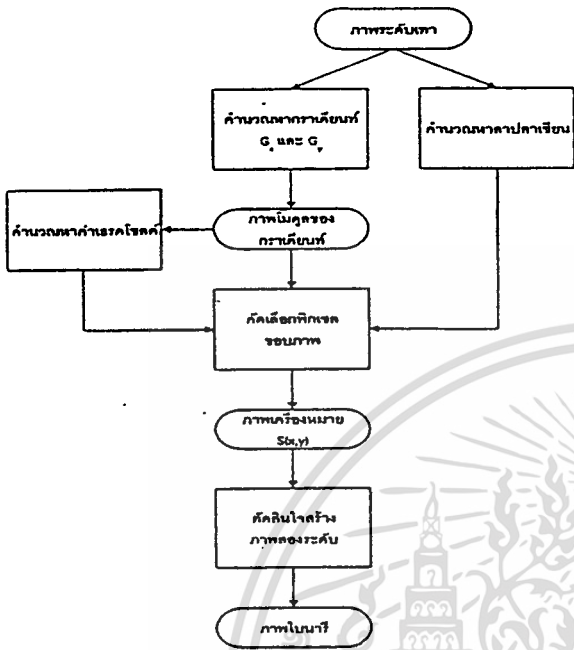
2. อัลกอริธึมแบบฟังก์ชันร่วม

การแยกภาพตัวอักษรโดยอัลกอริธึมแบบฟังก์ชันร่วม (Integrated Function Algorithm) [1] เป็นการรวบรวมอัลกอริธึมที่ง่าย ๆ แต่มีประสิทธิภาพมาใช้ร่วมกันกับภาพที่มีลักษณะความยุ่งยากขึ้นตอนการทำงานของอัลกอริธึมดังกล่าว แสดงดังรูปที่ 1

ทำการสแกนภาพเอกสารโดยใช้สแกนเนอร์มาเก็บไว้ในรูปของแฟ้มข้อมูลรูปภาพ 8 บิต 256 ระดับเทา ที่มีรูปแบบ (format) เป็น TIFF ไฟล์ เพื่อนำแฟ้มข้อมูลรูปภาพมาใช้ในการทำงานในขั้นตอนต่อไป

คำนวณหากราฟเดียนท์ของภาพ เพื่อทำการนิยามพิกเซลที่วางอยู่ในบริเวณที่เป็นขอบ เพราะพิกเซลที่เป็นขอบจะเป็นพิกเซลที่มีอัตราการเปลี่ยนแปลงระดับเทาสูง เมื่อเทียบกับพิกเซลข้างเคียงในทิศทางที่กำหนด ซึ่งการหากราฟเดียนท์ของภาพในทางปฏิบัติส่วนมากจะหากราฟเดียนท์ใน 2 ทิศทาง คือ ทิศทาง x และ

*อาจารย์ คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
*นักศึกษาระดับปริญญาโท วิทยาการคอมพิวเตอร์และเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง



รูปที่ 1 แผนภาพการทำงานของอัลกอริธึมแบบฟังก์ชันร่วมทิศทาง y ซึ่งเป็นการปฏิบัติที่อยู่บนพื้นฐานความรู้ของ First-order Derivative

สำหรับงานวิจัยนี้ใช้ โรเบลโอเปอเรเตอร์ (sobel operator) [3] ในการหาค่าความเค้นของภาพ เพราะต้องการให้ลักษณะการเค้น (magnitude of gradient) ของภาพ อยู่ในรูปแบบที่ Smooth หลังจากหาค่าความเค้นของภาพแล้ว โดยการนำ Sobel Mask ขนาด 3×3 ทั้งที่ใช้สำหรับการหาค่าความเค้นในทิศทาง x และ ทิศทาง y มาทำโอเปอเรชันกับทุกพิกเซลของภาพ โดย Sobel Mask ที่ใช้สำหรับหา G_x และ G_y จะถูกแสดงดังรูปที่ 2

-1	-2	-1
0	0	0
1	2	1

(ก)

-1	0	1
-2	0	2
-1	0	1

(ข)

รูปที่ 2 แสดง Sobel Mask (ก) สำหรับหา G_x (ข) สำหรับหา G_y

ดังนั้น Magnitude of Gradient ของภาพที่ได้จากโรเบลโอเปอเรเตอร์ จะเท่ากับ

$$\nabla F(x,y) = |G_x| + |G_y|$$

หลังจากหาค่าความเค้นของภาพแล้ว จะได้ภาพใหม่ที่แสดงลักษณะความเป็นขอบ ซึ่งเป็นทั้งขอบของตัวอักษรและขอบของสิ่งที่ไม่พึงประสงค์ที่อยู่ส่วนของพื้นหลัง

นำภาพที่ผ่านการหาค่าความเค้นที่มีค่าความหาค่าเรตโซลด์เพื่อนำค่าเรตโซลด์นั้นมาใช้ในการสืบหาเฉพาะส่วนที่เป็นขอบของตัวอักษร ซึ่งขั้นตอนการหาค่าเรตโซลด์ในงานวิจัยนี้จะประยุกต์มาจากวิธี การเรตโซลด์คั้งตามคุณลักษณะของวัตถุ (Object Attribute Threshold) [2] ซึ่งวัตถุในส่วนนี้ก็คือ ขอบของตัวอักษร

ขั้นตอนต่อไปจะทำการนิยามภาพลาปลาเซียนนี้จากภาพเอกสารถดับเทาเริ่มต้น โดยการนิยามทิกเซลของภาพเอ็กสาร์ทักหมดให้อยู่ในรูปของเครื่องหมาย โดยวิธีลาปลาเซียนโอเปอเรเตอร์ [3,4] ซึ่งปฏิบัติการนี้จะมีอยู่บนพื้นฐานความรู้ของ Second-order Derivative คือ ภาพลาปลาเซียนจะมี zero-crossing เกิดขึ้นในบริเวณที่เป็นขอบ

$$\nabla^2 F(x,y) = \frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2}$$

สำหรับขั้นตอนในการคำนวณหา ค่าลาปลาเซียนของภาพ ($\nabla^2 F(x,y)$) จะได้จากกา้นำ Laplacian Mask ขนาด 3×3 นี้มาใช้โอเปอเรชันกับทุกพิกเซลของภาพเอกสาร์ โดย Laplacian Mask จะถูกแสดงดังรูปที่ 3

0	-1	0
-1	4	-1
0	-1	0

รูปที่ 3 แสดง Laplacian Mask

ทิกเซลของภาพลาปลาเซียนจะถูกนิยามด้วยเครื่องหมายบวกเมื่อ $\nabla^2 F(x,y) \geq 0$ และนิยามด้วยเครื่องหมายลบเมื่อ $\nabla^2 F(x,y) < 0$

หลังจากนั้นนำทั้ง 3 ค่า คือ $\nabla F(x,y)$, Th (ค่าเรตโซลด์), $\nabla^2 F(x,y)$ มาประยุกต์ใช้ร่วมกันสำหรับการคัดคืนสร้างภาพใหม่ $S(x,y)$ ที่มี 3 สถานะ ซึ่งภาพ $S(x,y)$ นี้ เฉพาะส่วนที่เป็นขอบของ

ตัวอักษรหรือขอบของสิ่งที่ไม่พึงประสงค์ที่มีระดับเทาใกล้เคียงกับตัวอักษรเท่านั้นที่ถูกนิยามด้วยเครื่องหมาย (+,-) ซึ่งพิกเซลในภาพใหม่ $S(x,y)$ จะถูกนิยามดังต่อไปนี้

$$S(x,y) = \begin{cases} 0 & \text{if } \nabla F(x,y) < Thr \\ + & \text{if } \nabla F(x,y) \geq Thr \text{ and } \nabla^2 F(x,y) \geq 0 \\ - & \text{if } \nabla F(x,y) \geq Thr \text{ and } \nabla^2 F(x,y) < 0 \end{cases}$$

สัญลักษณ์ 0,+,- ทั้ง 3 จะถูกนำมาใช้แทนระดับเทาของภาพ โดยพิกเซลทั้งหมดที่ไม่อยู่ในบริเวณที่เป็นขอบที่ต้องการ (ซึ่งตัดสินโดย $\nabla F(x,y) < Thr$) จะถูกนิยามเป็นศูนย์และพิกเซลทั้งหมดที่เป็นขอบและอยู่ทางด้านมืดจะถูกนิยามโดยเครื่องหมายบวก ส่วนพิกเซลทั้งหมดที่เป็นขอบทางด้านสว่างจะถูกนิยามโดยเครื่องหมายลบ จากเงื่อนไขข้างต้นทำให้ขอบที่เกิดจากสิ่งที่ไม่พึงประสงค์บางส่วนจะถูกจำกัดทิ้งไปในภาพ $S(x,y)$ และนำภาพ $S(x,y)$ ไปใช้ในการสร้างภาพ 2 ระดับ ในขั้นตอนสุดท้าย

จากภาพ $S(x,y)$ จะพบว่า พิกเซลที่อยู่ภายในเส้นตัวอักษรจะถูกนิยามด้วยเครื่องหมาย + หรือไม่ก็ 0 และมีขอบเขตที่จะแสดงตามลำดับดังต่อไปนี้

$$\dots, -, +, \dots, [S(x,y) = 0 \text{ or } +], \dots, +, -, \dots$$

ความกว้างของเส้นเป็นเกณฑ์ที่สำคัญสำหรับการสร้างภาพผลลัพธ์ โดยพิกเซลที่นิยามด้วยเครื่องหมายลบทั้งหมดจะถูกทำให้เป็นสีขาว และพิกเซลที่นิยามด้วยศูนย์หรือเครื่องหมายบวกแต่อยู่นอกขอบเขตของขอบ (+,-) ก็จะถูกกำหนดให้เป็นสีขาวส่วนการตัดสินใจสร้างภาพ 2 ระดับ ดำ/ขาว กับพิกเซลที่อยู่ในขอบเขตของขอบ (+,-) จะทำโดยใช้เมตริกการตัดสินใจ คือ จากพิกเซลที่พิจารณาจะทำการกวาดหาขอบทั้งสองด้านและตรวจสอบระยะทางที่ได้ว่ามีความสัมพันธ์ได้สัดส่วนกับที่ต้องการหรือไม่ (ความกว้างของเส้น) ถ้าไม่ได้สัดส่วนพิกเซลนั้นจะถูกกำหนดให้เป็นสีขาว ถ้าได้สัดส่วนพิกเซลนั้นจะถูกกำหนดให้เป็นสีดำ ซึ่งการทดสอบในแต่ละพิกเซลจะถูกจำกัดไว้เป็น 2 ทิศทางคือทิศทาง x และ y สัมพันธ์กันไป ซึ่งจากการตรวจสอบข้างต้นจะทำให้ส่วนของพื้นหลังที่มีคือ มีระดับเทาใกล้เคียงหรือระดับเดียวกับตัวอักษรจะถูกจำกัดทิ้งไป ยกเว้นส่วนที่เป็นพื้นหลังนั้นจะมีขนาดใกล้เคียงกับขนาดของตัวอักษร ผลลัพธ์ที่ได้เป็นภาพไบนารีซึ่งมีตัวอักษรสีดำบนพื้นหลังสีขาว

ผลผลิตที่ได้เป็นภาพไบนารีซึ่งมีตัวอักษรสีดำบนพื้นหลังสีขาว

3. ผลการทดลอง

ทำการประเมินประสิทธิภาพของอัลกอริทึมฟังก์ชันร่วม โดยทำการทดสอบกับภาพเอกสารที่มีพื้นหลังในรูปแบบต่างๆ ผลปรากฏว่าอัลกอริทึมสามารถทำงานได้ตามต้องการ รูปที่ 4 แสดงให้เห็นถึงการทำงานของอัลกอริทึมในแต่ละขั้นตอน รูปที่ 4(ก) แสดงภาพเอกสารต้นฉบับ 256 ระดับเทา ความละเอียดของภาพเท่ากับ 300 จุดต่อนิ้ว ที่มีความยุ่งยากเกิดขึ้นในส่วนช่องพื้นหลัง รูปที่ 4(ข) แสดง Magnitude of Gradient ของภาพ รูปที่ 4(ค) แสดงภาพ $S(x,y)$ ที่นิยามด้วย 3 สถานะ รูปที่ 4(ง) แสดงภาพผลลัพธ์ที่อยู่ในรูปของภาพไบนารี

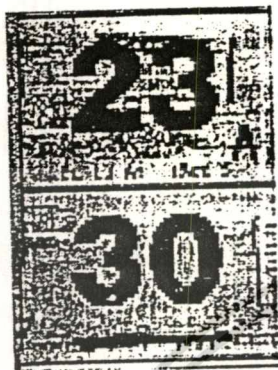
4. สรุปและวิจารณ์ผลการทดลอง

บทความนี้ได้เสนอขั้นตอนสำหรับการแยกภาพตัวอักษรออกจากภาพเอกสารที่มีลักษณะความยุ่งยากโดยใช้อัลกอริทึมแบบฟังก์ชันร่วม ซึ่งอัลกอริทึมนี้สามารถที่จะให้ภาพผลลัพธ์ที่มีความเหมาะสม แต่อัลกอริทึมแบบฟังก์ชันร่วมนี้ก็สามารถที่จะให้ภาพผลลัพธ์ที่ผิดพลาดได้ ในกรณีที่หน้าเอกสารนั้นมีความหลากหลายของขนาดตัวอักษรเกิดขึ้น หรือมีอัตราความแตกต่างของระดับเทาที่เกิดขึ้นระหว่างตัวอักษรกับพื้นหลังน้อยกว่า 20 % ในขณะที่อัตราความแตกต่างที่เกิดขึ้นเฉพาะในส่วนของพื้นหลังเองเกิน 20%

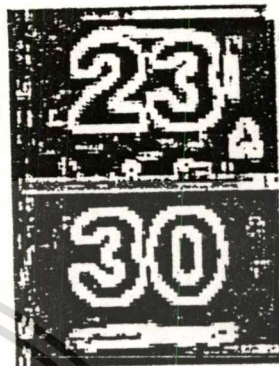
เอกสารอ้างอิง

- [1] J.M. White, G.D. Rohrer, "Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction," *IBM J. Res. Develop.*, vol.27, No 4, pp. 400-409, July 1983
- [2] ยุทธพงษ์ รั้งสรวิเศษ, สิทธิศักดิ์ รอดทุกข์, พงศ์เดช ณรงค์ฤทธิ์, "การเรขาคณิตตามคุณลักษณะของวัตถุ," *การประชุมวิชาการทางวิศวกรรมไฟฟ้าครั้งที่ 18*, 2538
- [3] Jae S. Lim, *Two-Dimensional Signal and Image Processing*, Prentice-Hall, 1990
- [4] Rafael C. Gonzalez, Richard E. Wood, *Digital Image Processing*, Addison-Wesley, 1992

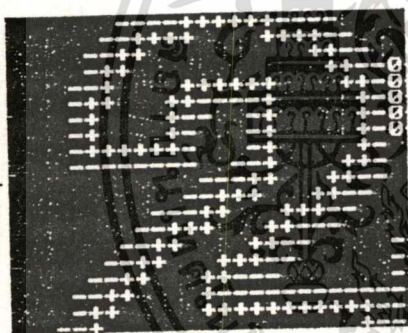
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก)



(ข)



(ค)



(ง)

รูปที่ 4 ผลการทดลองของอัลกอริทึมฟังก์ชันร่วมกับภาพเอกสาร (ก) ภาพเอกสารต้นฉบับ (ข) แสดง Magnitude of Gradient ของภาพ (ค) ภาพที่นิยามด้วย 3 สถานะ $S(x,y)$ (ง) ภาพผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำแธรดโซลด์ตามคุณลักษณะของวัตถุ

Object Attribute Thresholding

ดร. ยุทธพงษ์ รั้งวรรณเสรี* สิทธิศักดิ์ รอดทุกขี** พงศ์เดช ณรงค์ฤทธิ์***

บทคัดย่อ

การสร้างภาพไบนารีจากทำได้ไม่สมบูรณ์ สำหรับภาพเอกสารใดๆ อัลกอริธึมที่ใช้สำหรับการสร้างภาพไบนารีไม่ว่าอยู่ในรูปแบบของ โบบอล หรือ โลกอล ก็สามารถที่จะให้ข้อผิดพลาดได้กับภาพเอกสารที่มีสิ่งไม่พึงประสงค์ หรือ มีความซับซ้อนเกิดขึ้นในส่วนพื้นหลัง บทความนี้จึงนำเสนออัลกอริธึมที่ใช้ในการสร้างภาพไบนารีสำหรับภาพเอกสาร โดยการวิเคราะห์ฮิสโตแกรมระดับเทา และการกระจายระดับเทาของพิกเซลภายในภาพ ผลการทดลองกับภาพเอกสารใดๆ จำนวนมาก แสดงให้เห็นถึงความถูกต้องของการทำแธรดโซลด์ด้วยวิธีดังกล่าว

คำสำคัญ

Document image binarization is not a completely solved problem for unconstrained document images. Binarization algorithms, whether global or local, can easily fail on images with noisy or complex background. This paper will present the binarization of images by a new approach algorithm based on gray scale histogram and distribution of pixel analysis. Experimental results on unconstrained document images show that the binarization is correctly performed.

1. บทนำ

ในปัจจุบันนี้ถือว่าขั้นตอนการทำแธรดโซลด์ (thresholding) เป็นขั้นตอนที่สำคัญสำหรับการทำภาพเอกสารให้อยู่ในรูปแบบภาพไบนารี ซึ่งคุณภาพของภาพไบนารีที่ดีถือว่าเป็นสิ่งสำคัญสำหรับขั้นตอนการวิเคราะห์ภาพตัวอักษรในส่วนอื่นๆ เช่น การรู้

* อาจารย์คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณ

เทพารักษ์นครินทร์

** นักศึกษาระดับปริญญาโท หลักสูตรวิชาเทคโนโลยีสารสนเทศและเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

*** นักศึกษาระดับปริญญาโท คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

จำตัวอักษรไม่ว่าจะเป็นตัวเขียนหรือตัวพิมพ์ ค่าแธรดโซลด์ที่เหมาะสมจะขึ้นอยู่กับความแตกต่างที่เกิดขึ้นระหว่างช่วงระดับเทาของตัวอักษรกับช่วงระดับเทาของส่วนพื้นหลัง และในทางปฏิบัติแล้วเป็นไปได้ยากมากสำหรับการหาค่าแธรดโซลด์ที่เหมาะสมจากกรณีที่ระดับเทาของส่วนพื้นหลังมีความหลากหลาย และมีบางส่วนที่ระดับเทาใกล้เคียงกับช่วงระดับเทาของตัวอักษร หรือจากเอกสารที่มีคุณภาพการพิมพ์ต่ำ ซึ่งค่าแธรดโซลด์ที่สูงหรือต่ำเกินไปก็จะทำให้ภาพผลลัพธ์ขาดความถูกต้อง ในปัจจุบันนี้พบว่าอัลกอริธึมที่ใช้สำหรับการคำนวณหาค่าแธรดโซลด์ของภาพเอกสารนั้น มีอยู่หลายอัลกอริธึม แต่ละอัลกอริธึมก็จะให้คุณภาพของภาพไบนารีที่แตกต่างกัน และในกรณีที่ภาพเอกสารนั้นมีความซับซ้อนเกิดขึ้นสูงในส่วนของพื้นหลัง บางอัลกอริธึมก็จะให้ภาพไบนารีที่มีความผิดพลาดได้ บทความวิจัยนี้จะนำเสนอขั้นตอนการการสร้างภาพไบนารีโดยแธรดโซลด์อัลกอริธึมรูปแบบหนึ่ง ซึ่งเป็นอัลกอริธึมที่พัฒนามาเพื่อแก้ปัญหาที่เกิดจากข้อจำกัดของแธรดโซลด์อัลกอริธึมในรูปแบบเดิม และมีความยืดหยุ่นสูงขึ้นเหมาะสำหรับภาพเอกสารที่มีความยุ่งยากสูงในส่วนของพื้นหลัง

2. การคำนวณหาค่าแธรดโซลด์โดยใช้

อัลกอริธึมของ Otsu

อัลกอริธึมของ Otsu [1,2] เป็นอัลกอริธึมที่ใช้ในการหาค่าแธรดโซลด์ของภาพ โดยคำนวณมาจากฮิสโตแกรมระดับเทา ซึ่งค่าแธรดโซลด์ที่ได้จะเป็นแบ่งระดับเทาของภาพออกเป็น 2 คลาสอย่างเหมาะสม ซึ่งการปฏิบัติตามอัลกอริธึมของ Otsu สามารถทำได้โดยง่าย ดังหลักการต่อไปนี้

ถ้าการอนุมานว่าได้ทำการแบ่งพิกเซลออกเป็น 2 คลาสคือ C_0 และ C_1 โดยค่าแธรดโซลด์ค่าหนึ่งที่ระดับเทา k ซึ่ง C_0 แสดงถึงพิกเซลที่อยู่ในระดับเทา $\{1, 2, \dots, k\}$ และ C_1 แสดงถึงพิกเซลที่อยู่ในระดับเทา $\{k+1, \dots, L\}$ และให้ σ_{ij}^2 (within class variance) คือค่า

แม้ว่าการแบ่งเท่าๆกัน ยกทั้งที่ไม่พบเหตุแต่แปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความแปรปรวนที่เกิดขึ้นภายในคลาส, σ_B^2 (between-class variance) คือ ความแปรปรวนที่เกิดขึ้นระหว่างคลาส, σ_T^2 (total variance of levels) คือ ความแปรปรวนของระดับเทาทั้งหมด ส่วนการประเมินความเหมาะสมของค่าเรตโซลต์ (ที่ระดับเทา k) จะใช้บรรทัดฐานในการประเมินดังต่อไปนี้

$$\lambda = \frac{\sigma_B^2}{\sigma_W^2}$$

$$\beta = \frac{\sigma_T^2}{\sigma_W^2}$$

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

จากสมการข้างต้นพบว่าค่า σ_W^2 และ σ_B^2 เป็นฟังก์ชันของค่าเรตโซลต์ k แต่ค่า σ_T^2 จะไม่ขึ้นกับ k เพราะฉะนั้นค่า η จึงเป็นเครื่องมือในการประเมินผลที่ง่ายที่สุดที่ขึ้นกับค่า k ดังนั้นจึงนำค่า η มาเป็นเครื่องมือสำหรับการประเมินความเหมาะสมของค่าเรตโซลต์ที่ระดับเทา k ซึ่งค่าเรตโซลต์ที่เหมาะสมจะขึ้นอยู่กับค่า η หรือ $\sigma_B^2(k)$ ที่มากที่สุด โดยทำการตรวจสอบจากสมการต่อไปนี้

$$\eta(k) = \frac{\sigma_B^2(k)}{\sigma_T^2}$$

$$\sigma_B^2(k) = \frac{(\mu w(k) - \mu(k))^2}{w(k)(1-w(k))}$$

ดังนั้นค่าเรตโซลต์ที่เหมาะสม คือ ค่าระดับเทา (k^*) ที่ $\sigma_B^2(k^*) = \text{MAX}(\sigma_B^2(k)) : 1 \leq k < L$

โดย

$$w_0 = \sum_{i=1}^k p_i = w(k), w_1 = 1 - w(k)$$

$$\mu_0 = \sum_{i=1}^k \frac{i p_i}{w_0}, \mu_1 = \sum_{i=k+1}^L \frac{i p_i}{w_1}, \mu = \sum_{i=1}^L i p_i$$

$$\sigma_0^2 = \frac{\sum_{i=1}^k (1 - \mu_0)^2 p_i}{w_0}$$

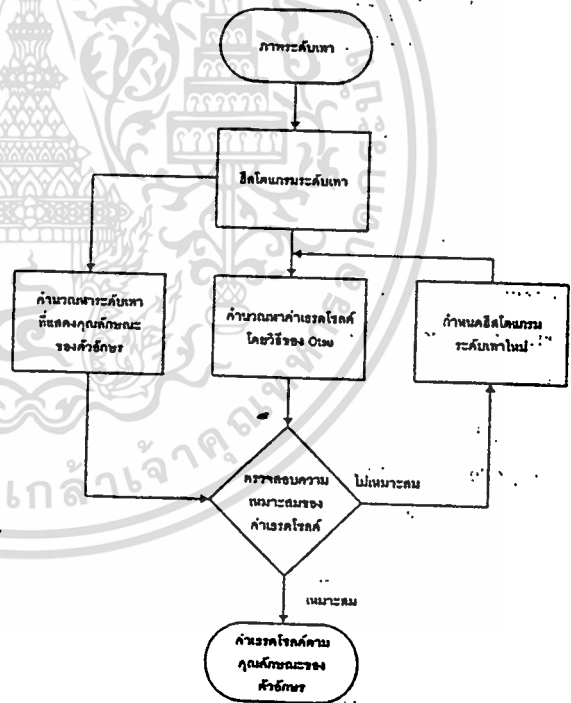
$$\sigma_1^2 = \frac{\sum_{i=k+1}^L (1 - \mu_1)^2 p_i}{w_1}$$

$$\sigma_B^2 = w_0 w_1 (\mu_0 - \mu_1)^2$$

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu)^2 p_i$$

$$\sigma_W^2 = w_0 \sigma_0^2 + w_1 \sigma_1^2$$

3. การทำเรตโซลต์ตามคุณลักษณะของวัตถุ การทำเรตโซลต์ด้วยอัลกอริทึมของ Otsu เหมาะสำหรับภาพเอกสารที่มีจำนวนทีก (peak) ที่เกิดขึ้นบนฮิสโตแกรมระดับเทาแค่ 2 ทีก แต่กรณีของภาพเอกสารที่มีลักษณะของความยุ่งยาก คือ มีความหลากหลายของจำนวนทีกบนฮิสโตแกรมระดับเทา จะพบว่า การทำเรตโซลต์โดยอัลกอริทึมของ Otsu จะให้ภาพไบนารีที่ไม่ถูกต้อง ดังนั้นงานวิจัยชิ้นนี้จึงนำเสนอขั้นตอนการทำงานเรตโซลต์ในรูปแบบหนึ่ง คือ การทำเรตโซลต์ตามคุณลักษณะของวัตถุ (Object Attribute Thresholding, 'OAT') โดยนำอัลกอริทึมของ Otsu มาประยุกต์ใช้ร่วมด้วย สำหรับประโยชน์ที่ได้จาก OAT นี้จะเป็นรากฐานที่สำคัญในการทำให้ภาพเอกสารมีคุณสมบัติที่ถูกต้องมากที่สุดเมื่อทำเป็นภาพไบนารี ซึ่งขั้นตอนการทำงานของ OAT จะถูกแสดงดังรูปที่ 1



รูปที่ 1 แผนภาพแสดงขั้นตอนการทำงานของ OAT

เพราะฉะนั้นความยุ่งยากที่เกิดขึ้นอันเนื่องมาจาก ความหลากหลายของจำนวนทีกบนฮิสโตแกรมระดับเทา จะถูกจำกัดลง โดยการปรับค่าเรตโซลต์ให้มีความเหมาะสม และใช้คุณลักษณะของวัตถุ (ค่าระดับเทาที่มีคุณสมบัติว่าเป็นระดับเทาตัวอักษร) มาพิจารณาความเหมาะสมของค่าเรตโซลต์ ซึ่งค่าเรตโซลต์ที่ได้จากการปรับในแต่ละครั้งจะเข้าใกล้ค่าระดับเทาที่แสดงคุณ

เอกสารนี้เป็นลิขสิทธิ์ของสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง การนำเอกสารนี้ไปเผยแพร่โดยไม่ได้รับอนุญาตถือว่าผิดกฎหมาย การนำเอกสารนี้ไปใช้ในการค้าจากกรปรับในแต่ละครั้งจะเข้าใกล้ค่าระดับเทาที่แสดงคุณ

ลักษณะของตัวอักษรมากยิ่งขึ้น และในงานชิ้นนี้ได้ใช้ อัลกอริทึมของ Otsu สำหรับหาค่าเรตโซลต์ในการปรับแต่ละครั้ง ภาพภาพเอกสารที่ใช้เป็นข้อมูลสำหรับ OAT อัลกอริทึม ได้จากการสแกนหน้าเอกสารใดๆ ด้วยสแกนเนอร์ ที่มีความละเอียดของภาพ 300 จุดต่อนิ้ว แล้วเก็บไว้ในแฟ้มข้อมูลรูปภาพ 8 บิต 256 ระดับเทา ในรูปแบบ (format) TIFF หลังจากนั้นจะนำแฟ้มข้อมูลรูปภาพมาทำฮิสโตแกรมระดับเทาเพื่อใช้สำหรับการทำงานในขั้นตอนต่อไป

วิธี OAT นี้ ได้นำอัลกอริทึมของ Otsu มาประยุกต์ใช้กับฮิสโตแกรมระดับเทาของภาพ โดยการทำซ้ำเพื่อใช้สำหรับการปรับค่าเรตโซลต์ ซึ่งการปฏิบัติสามารถกระทำดังต่อไปนี้

3.1 กำหนดเงื่อนไข โดยใช้คุณลักษณะของวัตถุมาเป็นเกณฑ์ในการกำหนด ซึ่งค่าเรตโซลต์ที่ได้จากการปรับในแต่ละครั้งต้องอยู่ภายในเงื่อนไขที่กำหนด

3.2 กำหนดให้ $j = 1$

3.3 กำหนดฮิสโตแกรมระดับเทาของภาพ (G_j)

3.4 นำอัลกอริทึมของ Otsu มาคำนวณหาค่าเรตโซลต์ t_j จาก G_j

3.5 กำหนดฮิสโตแกรมระดับเทาใหม่โดยให้ G_{j+1} เท่ากับ ส่วนของ G_j ที่อยู่ระหว่าง "black" กับ t_j

3.6 เพิ่มค่า j แล้วทำซ้ำข้อ 3.4 และ 3.5 จนกระทั่งค่าเรตโซลต์ที่ได้จากการปรับไม่อยู่ในเงื่อนไขที่กำหนด ดังนั้นค่าเรตโซลต์ที่ได้จากวิธีของ OAT นี้คือ ค่าเรตโซลต์ค่าสุดท้ายที่ได้จากการปรับและอยู่ในเงื่อนไขที่กำหนด

เงื่อนไขก็คือ ค่าเรตโซลต์ที่คำนวณได้ต้องไม่น้อยกว่าค่าที่กำหนด ซึ่งค่าที่กำหนดขึ้นมานั้นจะคำนวณมาจากขั้นตอนดังต่อไปนี้ คือ การทำพรีเทสต์ (pretest) กับ ภาพก่อน โดยการหาค่าเรตโซลต์ หลังจากนั้นทำการตรวจสอบลักษณะการกระจายของระดับเทา โดยดูจากค่าความน่าจะเป็นของแต่ละระดับเทาในฮิสโตแกรมระหว่างส่วน "black" กับ ค่าเรตโซลต์ที่ได้จากการทำพรีเทสต์ ระดับเทาที่มีค่าความน่าจะเป็นสูงในช่วงนี้ หรือ มีอัตราการเพิ่มของค่าความน่าจะเป็นสูง เมื่อเทียบค่าความน่าจะเป็นที่คำนวณได้ในช่วงนี้ กับค่าความน่าจะเป็นที่หาจากฮิสโตแกรมระดับเทาทั้งหมด จะเป็นระดับเทาที่อนุमानว่าเป็นค่าที่อยู่ในช่วงของระดับเทาที่แสดงคุณลักษณะของตัวอักษร แต่ในบางส่วนของหน้าเอกสารพบว่าตัวอักษรจะมีระดับเทาที่แตกต่างกันเพราะฉะนั้นค่าที่กำหนดให้เป็นเงื่อนไขสำหรับค่าเรตโซลต์ต้องคุมถึงจุดนี้ด้วย ดังนั้นค่าที่กำหนดเป็นเงื่อนไขจะทำการหาค่าระดับเทาที่ครอบคลุม

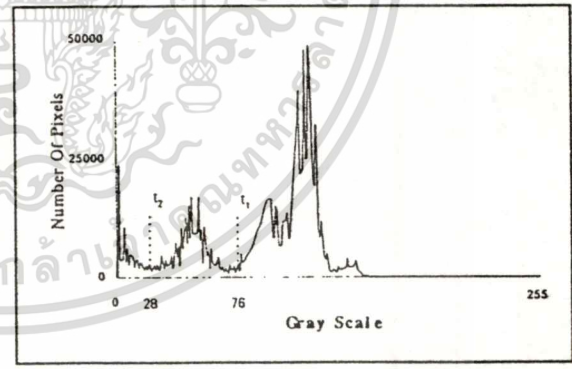
ระดับเทาที่อยู่ในช่วงของค่าที่แสดงคุณลักษณะของตัวอักษรในเปอร์เซนต์ที่กำหนด

4. ผลการทดลอง

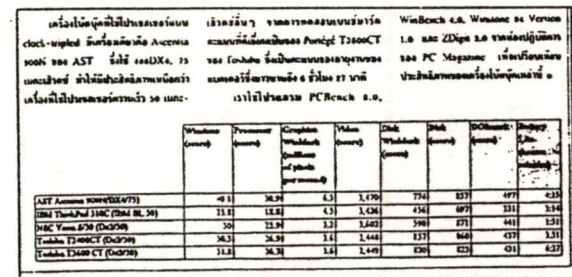
ทำการประเมินประสิทธิภาพของ OAT อัลกอริทึมโดยการทดสอบกับภาพเอกสารที่มีความไม่สม่ำเสมอเกิดขึ้นในรูปแบบต่างๆ กัน ผลปรากฏว่าภาพเอกสารต่างๆ เหล่านี้จะถูกสร้างเป็นภาพไบนารีที่มีความถูกต้องเหมาะสมในระดับหนึ่ง ซึ่งตัดสินจากการมองเห็น ผลการทดลองของ OAT อัลกอริทึม กับภาพเอกสารแสดงดังรูปที่ 2 โดยรูปที่ 2 (ก) แสดงภาพเอกสารเริ่มต้น



(ก)



(ข)



(ค)

รูปที่ 2 แสดงผลการทดลองของ OAT อัลกอริทึม (ก) ภาพเอกสารต้นฉบับ (ข) ฮิสโตแกรมระดับเทาของภาพและค่าเรตโซลต์ t_1 และ t_2 ที่คำนวณได้ (ค) ภาพผลลัพธ์ที่ได้จากการทำการทำเรตโซลต์ด้วย t_1 และต้องอ้างอิงการทำเรตโซลต์ด้วย t_2 ทุกครั้งที่มีการนำไปใช้

การประชุมวิชาการทางวิศวกรรม... ปี ครั้งที่ ๒๕

256 ระดับเทา รายละเอียดของภาพ 300 จุดต่อนิ้ว ที่รูปแบบของพื้นหลังมีความไม่สม่ำเสมอ รูปที่ 2 (ข) แสดงฮิสโตแกรมระดับเทาของภาพ โดยที่ t_1 และ t_2 แสดงถึงค่าเรตซีลด์ที่คำนวณได้ ซึ่งค่าเหล่านี้คำนวณหาจากฮิสโตแกรมระดับเทา ด้วยอัลกอริธึมของ Otsu โดยค่าเรตซีลด์ค่าแรก (t_1) เท่ากับ 76 ส่วนค่าเรตซีลด์ค่าที่สอง (t_2) ที่ได้จากการปรับเท่ากับ 28 ซึ่งเป็นค่าเรตซีลด์ที่เหมาะสมสำหรับภาพเอกสารชิ้นนี้ ตามหลักการของการทำเรตซีลด์ตามคุณลักษณะของวัตถุ ภาพที่ 2 (ค) แสดงภาพผลลัพธ์ที่อยู่ในรูปของภาพไบนารีที่เกิดจากการทำเรตซีลด์กับภาพเอกสารต้นฉบับด้วยค่า t_2

5. สรุปและวิจารณ์ผลการทดลอง

อัลกอริธึมของ OAT แสดงให้เห็นว่า ค่าเรตซีลด์จะปรับค่าตามลักษณะของภาพ จนสามารถที่จะอธิบายลักษณะของสิ่งที่คาดหวัง (ตัวอักษร) ดังนั้นสามารถกล่าวได้ว่า อัลกอริธึมของ OAT

นี้มีความยืดหยุ่นมากพอที่จะใช้จัดการกับภาพเอกสารที่มีความหลากหลายสูง แต่อัลกอริธึมของ OAT นี้ก็สามารถให้ผลที่ผิดพลาดได้ ในกรณีที่ภาพเอกสารนั้นมีชุดของตัวอักษรที่มีลักษณะเบลอ หรือมีอัตราความแตกต่างของระดับเทาน้อยมากเมื่อเทียบกับพื้นหลัง

เอกสารอ้างอิง

- [1] N. Otsu, "A Threshold Selection Method. From Gray-Level Histograms," *IEEE Trans. Systems Man Cybernet*, SMC-9, pp. 62-66, 1979.
- [2] P.K. Sahoo, S. Sotani, A.K.C. Wong, "A Survey of Thresholding Techniques," *Computer Vision, Graphics and Image Processing*, vol. 41, pp. 230-260, 1988.

ประวัติผู้เขียน

ชื่อ	นายสิทธิศักดิ์ รอดทุกข์
เกิดวันที่	1 สิงหาคม 2513
สถานที่เกิด	จังหวัดกรุงเทพมหานคร
วุฒิการศึกษา	ระดับประถมศึกษา โรงเรียนอานวยศิลป์ ระดับมัธยมศึกษาตอนต้น โรงเรียนอานวยศิลป์ ระดับมัธยมศึกษาตอนปลาย โรงเรียนเตรียมอุดมศึกษา ระดับปริญญาตรี วิทยาศาสตร์บัณฑิต สาขาสถิติ มหาวิทยาลัยศิลปากร

