

การกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ

นางสาวธนรัตน์ เหลืองอ่อน

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ครุศาสตร์อุตสาหกรรมมหาบัณฑิต

สาขาวิชาเทคโนโลยีคอมพิวเตอร์ ภาควิชาคอมพิวเตอร์ศึกษา

บัณฑิตวิทยาลัย มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ปีการศึกษา 2550

ลิขสิทธิ์ของมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

ชื่อ : นางสาวธนรัตน์ เหลืองอ่อน
ชื่อวิทยานิพนธ์ : การกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ
สาขาวิชา : เทคโนโลยีคอมพิวเตอร์
มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก : ผู้ช่วยศาสตราจารย์ ดร.กันต์พงษ์ วรรณปัญญา
อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม : ผู้ช่วยศาสตราจารย์ ดร.สุพจน์ นิตย์สุวรรณ
ปีการศึกษา : 2550

บทคัดย่อ

การกู้คืนภาพเป็นกระบวนการที่สำคัญในการลดส่วนที่เสียหายจากการถูกตัวอักษรซ้อนทับของภาพ ในงานวิจัยนี้ได้เสนอกระบวนการกู้คืนส่วนที่เสียหายของภาพโดยแบ่งออกเป็น 2 กระบวนการหลักคือ การหาส่วนที่เป็นตัวอักษร (Text Detection) และการปกปิดพื้นผิว (Texture Concealment) ในกระบวนการหาส่วนที่เป็นตัวอักษรประกอบไปด้วย การแบ่งภาพเป็นภาพย่อย การแบ่งกลุ่มข้อมูล (Clustering) การทำภาพเป็นขาว – ดำ (Binarization) และการลดสิ่งรบกวน (Noise Reduction) ซึ่งเป็นการกำหนดตำแหน่งของตัวอักษรโดยจะเรียกในส่วนนี้ว่า Text Mask ในการกู้คืนส่วนที่เสียหายของภาพจะใช้กระบวนการของการปกปิดพื้นผิวเข้ามาช่วยในการกู้คืนส่วนที่เสียหายของภาพโดยมี Text Mask เป็นตัวกำหนดตำแหน่ง การสังเคราะห์พื้นผิวเป็นการสร้างพื้นผิวขึ้นในส่วนที่ภาพเกิดความเสียหาย หลังจากนั้นใช้การปกปิดความผิดพลาดในการขจัดส่วนที่เป็นร่องรอยของบล็อก (Blocking Artifacts) ที่เกิดจากการสังเคราะห์พื้นผิว โดยผลลัพธ์ที่ได้จากการทดลองด้วยวิธีการนี้ทำให้ได้ภาพในส่วนที่เสียหายกลับคืนมาอย่างชัดเจนและมีคุณภาพสูง

(วิทยานิพนธ์มีจำนวนทั้งสิ้น 67 หน้า)

คำสำคัญ : การแบ่งกลุ่ม ฟัซซี่ การสังเคราะห์พื้นผิว การปกปิดความผิดพลาด

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

Name : Miss Tanarat Luang-on
Thesis Title : Recovery of Image Partially Occluded by Text
Major Field : Computer Technology
King Mongkut's University of Technology North Bangkok
Thesis Advisor : Assistant Professor Dr.Kuntpong Woraratpanya
Co-Advisor : Assistant Professor Dr.Supot Nitsuwat
Academic Year : 2007

Abstract

An image recovery is an important procedure to minimize the partial image damage caused by text occlusions. This thesis proposes an optimized recovery of partial damage images. In this method, there are two principal procedures in cooperation: text detection and texture concealment. The text detection procedure – composed of block-image partition, intensity clustering, fuzzy binarization, and noise reduction – identifies text locations called “text masks.” In order to recover the partial damage images, the texture concealment is applied to the damage areas identified by text masks. In this phase, the texture synthesis is initially used to damage images. Then the proposed error concealment is employed to suppress blocking artifacts caused by the texture synthesis phase. The experimental results illustrate that this method can evidently improve the quality of partial damage images.

(Total 67 pages)

Keywords: Clustering, Fuzzy, Texture Synthesis, Error Concealment

Advisor

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยความช่วยเหลืออย่างดียิ่งของท่านอาจารย์ที่ปรึกษาวิทยานิพนธ์ ผู้ช่วยศาสตราจารย์ ดร.กันต์พงษ์ วรรณปัญญา และอาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม ผู้ช่วยศาสตราจารย์ ดร.สุพจน์ นิตยสุวัฒน์ ตลอดจนกรรมการสอบวิทยานิพนธ์ทุกท่าน ซึ่งได้กรุณาใช้เวลาให้คำปรึกษา เสนอแนะข้อคิดเห็นต่างๆ ผู้วิจัยจึงขอขอบพระคุณในความกรุณาเป็นอย่างยิ่ง

ขอขอบคุณ คุณณัฐวัฒน์ เศรษฐเถียร และคุณประสิทธิ์ ช่วยเกื้อที่คอยให้คำปรึกษาทางด้านการเขียนโปรแกรม คุณรวีกานต์ ศดิยสุข คุณพิชิต จันทราประภาวัฒน์ คุณธงชัย จรรยาสิทธิกุล และบุคคลรอบข้างที่คอยให้กำลังใจและให้คำปรึกษาในด้านต่างๆ เสมอ

สุดท้ายขอกราบขอบพระคุณบิดา มารดา ที่คอยให้กำลังใจ ถวายภัตตาหาร สันนับสนุน และ ส่งเสริมทั้งทางด้านการศึกษาและการดำเนินชีวิตเสมอมา

ธนรัตน์ เหลืองอ่อน

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	ข
บทคัดย่อภาษาอังกฤษ	ค
กิตติกรรมประกาศ	ง
สารบัญ	จ
สารบัญตาราง	ช
สารบัญภาพ	ซ
บทที่ 1 บทนำ	1
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์ของการทำวิจัย	2
1.3 ขอบเขตของงานวิจัย	2
1.4 คำจำกัดความในงานวิจัย	2
1.5 ประโยชน์ที่คาดว่าจะได้รับ	3
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	4
2.1 ภาพเชิงจิตตอล	4
2.2 แบบจำลองสี่	8
2.3 ฟิชชีลลอจิก	10
2.4 การแบ่งกลุ่มข้อมูล	15
2.5 การตัดแบ่งภาพ	20
2.6 การสังเคราะห์พื้นผิว	23
2.7 การปกปิดความผิดพลาด	25
2.8 การวัดประสิทธิภาพ	27
2.9 งานวิจัยที่เกี่ยวข้อง	27
บทที่ 3 ขั้นตอนการวิจัย	29
3.1 ขั้นตอนการเตรียมภาพ	30
3.2 ขั้นตอนการแบ่งกลุ่ม	30
3.3 ขั้นตอนการทำภาพเป็นขาว - ดำ	35
3.4 ขั้นตอนการลดจำนวนสิ่งรบกวน	37
3.5 ขั้นตอนการปกปิดพื้นผิว	40

สารบัญ (ต่อ)

	หน้า
3.6 การเตรียมข้อมูลการทดลอง	42
3.7 การวัดประสิทธิภาพ	44
3.8 สรุป	44
บทที่ 4 ผลการวิจัยและวิเคราะห์ผลการวิจัย	45
4.1 ผลการวิจัย	45
4.2 วิเคราะห์ผลการวิจัย	61
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	63
5.1 สรุปผลการวิจัย	63
5.2 ข้อเสนอแนะ	63
เอกสารอ้างอิง	65
ประวัติผู้วิจัย	67

สารบัญตาราง

ตารางที่		หน้า
4-1	ตัวอย่างผลการทดลองการแบ่งกลุ่ม	46
4-2	ตัวอย่างการทดลองการทำภาพขาว - ดำ	48
4-3	การเปรียบเทียบการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย ฟิซซีซีมีน การทำภาพขาว-ดำโดยใช้ฟิซซี และการลดสิ่งรบกวนด้วย การหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 32×32 พิกเซล	51
4-4	การเปรียบเทียบการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย ฟิซซีซีมีน การทำภาพขาว-ดำโดยใช้ฟิซซี และการลดสิ่งรบกวนด้วย การหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 64×64 พิกเซล	52
4-5	การเปรียบเทียบการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย ฟิซซีซีมีน การทำภาพขาว-ดำโดยใช้ฟิซซี และการลดสิ่งรบกวนด้วย การหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 128×128 พิกเซล	53

สารบัญภาพ

ภาพที่		หน้า
2-1	ภาพในความหมายของฟังก์ชัน 2 มิติ $g(x,y)$ ของค่าความเข้มแสง	4
2-2	ตัวอย่างภาพสี RGB	5
2-3	ตัวอย่างภาพโทนสีเทา	6
2-4	ตัวอย่างภาพขาว-ดำ	6
2-5	ลักษณะกราฟฮิสโตแกรมที่ได้จากภาพทางด้านซ้าย	7
2-6	ระบบเพื่อนบ้านของจุดภาพ p ใดๆ	8
2-7	ภาพแสดงระบบสี RGB	9
2-8	องค์ประกอบทั้ง 3 ของระบบจำลองสี YUV	10
2-9	ตัวอย่างเซตย่อยสองเซต	11
2-10	ฟังก์ชันความเป็นสมาชิกในเซตคนที่อยู่ในบ้าน	12
2-11	กราฟของฟังก์ชันรูปตัวเอส	13
2-12	กราฟของฟังก์ชันรูปตัวแซด	14
2-13	กราฟของฟังก์ชันรูปโค้งกลับอย่างตัวซีหรือตัวเอส	14
2-14	ขั้นตอนการแบ่งกลุ่มแบบเคมีน	17
2-15	ขั้นตอนการแบ่งกลุ่มแบบฟัซซี่มิน	18
2-16	ภาพแสดงค่าเทรชโฮล	21
2-17	การสังเคราะห์หนึ่งจุดภาพ	23
2-18	การปะติดปะต่อพื้นผิว	24
2-19	การทำรอยต่อให้มีความกลมกลืน	25
2-20	การปกปิดความผิดพลาด	26
3-1	กระบวนการกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ	29
3-2	ผลการเปรียบเทียบการแบ่งกลุ่มข้อมูลด้วยเคมีน	33
3-3	การทำภาพเป็นขาว - ดำโดยใช้อะแดปทีฟเทรชโฮล	36
3-4	การทำภาพขาว - ดำโดยใช้ฟัซซี่	37
3-5	ค่าที่ได้จากการใช้ฟัซซี่	37
3-6	การทำงานของ CRLA	38
3-7	การเริ่มหาขอบของวัตถุ	38

สารบัญภาพ (ต่อ)

ภาพที่	หน้า
3-8 การหาขอบของวัตถุ	39
3-9 การวัดขนาดของวัตถุ	39
3-10 การหาส่วนที่จะนำมาเป็นภาพตัวอย่างในการสังเคราะห์	40
3-11 การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปร	41
3-12 การปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้	42
3-13 ข้อมูลภาพที่ใช้ในงานวิจัย	43
4-1 ภาพตัวอย่างผลการทดลองการลดจำนวนสิ่งรบกวน	50
4-2 กราฟเปรียบเทียบค่า PSNR ในการกู้คืนรายละเอียดภาพ	54
4-3 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย พีชชีชีมิน การทำภาพขาว-ดำโดยใช้พีชชี และการลดจำนวน สิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (im03.bmp)	56
4-4 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย พีชชีชีมิน การทำภาพขาว-ดำโดยใช้พีชชี และการลดจำนวน สิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (im10.bmp)	57
4-5 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย พีชชีชีมิน การทำภาพขาว-ดำโดยใช้พีชชี และการลดจำนวน สิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (im12.bmp)	58
4-6 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย พีชชีชีมิน การทำภาพขาว-ดำโดยใช้พีชชี และการลดจำนวน สิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (im15.bmp)	59
4-7 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วย พีชชีชีมิน การทำภาพขาว-ดำโดยใช้พีชชี และการลดจำนวน สิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (im18.bmp)	60
4-8 กระบวนการกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ	62

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ปัจจุบันเทคโนโลยีในการผลิตสื่อได้มีวิวัฒนาการอย่างต่อเนื่อง และสื่อส่วนใหญ่มักนำภาพมาประกอบ เพื่อให้สื่อเป็นที่น่าสนใจ มีความชัดเจนในการนำเสนอมากกว่าการใช้ข้อความอธิบายและประหยัดเนื้อที่ ซึ่งภาพบางภาพนั้นไม่สามารถหาภาพต้นฉบับที่เป็นฟิล์มของภาพ หรือไม่สามารถกลับไปถ่ายใหม่ได้ เนื่องจากเป็นภาพทางประวัติศาสตร์ ภาพประทับใจ หรืออาจเป็นภาพสถานที่ที่ไม่มีอยู่ในปัจจุบันแล้ว แต่ปรากฏว่ามีข้อความอยู่บนรูปภาพนั้นๆ จึงเกิดการซ้อนทับของสีสันในส่วนที่เป็นพื้นหลัง (Background) กับส่วนของตัวอักษรที่นำเสนอในสื่อ ซึ่งทำให้รายละเอียดบางส่วนของภาพที่นำมาใช้ขาดหายไปจากการถูกตัวอักษรหรือข้อความซ้อนทับ

ดังนั้นเพื่อเป็นการกู้คืนรายละเอียด (Recover) ของภาพกลับมา การศึกษาการแยกตัวอักษร (Text Segmentation) เป็นส่วนสำคัญอย่างหนึ่งที่จะเป็นการกู้รายละเอียดของภาพกลับมา ซึ่งการแยกตัวอักษรมีการพัฒนาอย่างต่อเนื่องโดยเริ่มแรกเป็นการทำภาพให้เป็นภาพขาว-ดำ (Monochromatic Document Image) [1] ก่อนที่จะทำการแยกตัวอักษร ต่อมาในปี ค.ศ. 1998 Savakis [2] เสนอวิธีการแยกตัวอักษรโดยใช้ค่าอะแดปทีฟเทรชโฮลดิ้ง (Adaptive Thresholding)

วิธีการที่กล่าวมาข้างต้นนั้น คือ การแยกตัวอักษรที่ไม่ได้ซ้อนทับอยู่บนภาพ ซึ่งต่อมา Wu , Chiu และ Chen [3] ได้พัฒนาวิธีการแยกตัวอักษรที่ซ้อนทับอยู่บนภาพ โดยใช้ค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานในการแบ่งกลุ่มภาพ และใช้วิธีการหาค่าเทรชโฮล (Thresholding) ในการแยกตัวอักษรออกจากส่วนที่เป็นพื้นหลังโดยภาพที่ได้จากการแยกตัวอักษรออกไปแล้วนั้นจะมีรายละเอียดบางส่วนของภาพไม่ชัดเจนเกิดขึ้น ทำให้ภาพนั้นขาดความสมบูรณ์ไปจากต้นฉบับ เพื่อเป็นการกู้คืนรายละเอียดของภาพกลับมา Savchenko, Kojekine และ Unno [4] ได้เสนอวิธีการตกแต่งภาพโดยใช้เทคนิคระยะห่างระหว่างพื้นที่ (Space-Mapping) ในการแก้ไขส่วนที่เสียหาย ซึ่งวิธีการนี้จะใช้ได้ผลกับพื้นที่เสียหายที่มีขนาดเล็ก และมีลักษณะเป็นเส้นตรง หรือเกือบเป็นเส้นตรง

วิธีการของ Efros และ Leung [5] เป็นการการสังเคราะห์พื้นผิวโดยใช้วิธีการสุ่มตัวอย่างจากนอนพารามเมตริก (Non-parametric Sampling) ซึ่งจะทำการสร้างภาพใหม่รอบๆ จุดเริ่มต้น (Initial Seed) โดยใช้การสุ่มบริเวณ (Markov Random Field) ในการคาดการณ์ล่วงหน้าและกำหนด

เงื่อนไขในการสังเคราะห์เพื่อนบ้าน (Neighbor Synthesis) จากภาพตัวอย่างและหาเพื่อนบ้าน (Neighborhoods) ที่เหมือนกัน ซึ่งการสังเคราะห์พื้นผิวด้วยวิธีนี้จะใช้เวลานานในการสังเคราะห์

วิธีการของ Efros และ Freeman [6] เสนอวิธีการปะติดปะต่อภาพ (Image Quilting) โดยเริ่มจากการสุ่มหาส่วนที่เหมือนกับบล็อกที่นำเข้ามา จากนั้นหาส่วนที่มีการซ้อนทับกันในส่วนหนึ่งของบล็อกนั้น และทำการหาค่าความผิดพลาดที่น้อยที่สุด (Minimum Error Boundary Cut) จากส่วนที่ซ้อนทับกัน เพื่อให้ได้ภาพที่มีความกลมกลืนกันมากที่สุด แต่ภาพที่ได้ออกมานั้นก็ยังมี ความไม่กลมกลืนให้เห็นบ้างในส่วนที่เป็นรอยต่อ

สำหรับงานวิจัยนี้ ได้ทำการศึกษาการหาส่วนที่เป็นตัวอักษร โดยมีขั้นตอนย่อย 4 ขั้นตอนคือการแบ่งภาพเป็นภาพย่อย การแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว-ดำ และการลดสิ่งรบกวน (Noise Reduction) โดยทำการเติมเต็มส่วนที่เสียหายโดยใช้การสังเคราะห์พื้นผิว (Texture Synthesis) ในการสังเคราะห์รายละเอียดของภาพ และทำให้ภาพสมบูรณ์ยิ่งขึ้นด้วยการใช้การปกปิดความผิดพลาด (Error Concealment) ในการทำภาพให้มีความกลมกลืนมากยิ่งขึ้น

1.2 วัตถุประสงค์ของการวิจัย

เพื่อพัฒนาขั้นตอนวิธีการกู้คืนรายละเอียดของภาพ ที่ขาดหายไปจากการแยกตัวอักษรหรือข้อความออกไปโดยการสังเคราะห์พื้นผิวและการปกปิดความผิดพลาด

1.3 ขอบเขตของงานวิจัย

1.3.1 ภาพที่นำมาทำการทดสอบเป็นภาพดิจิทัลแบบ RGB และมีขนาดความละเอียดข้อมูลภาพ 256×256 พิกเซล จำนวน 20 ภาพ และต้องเปลี่ยนเป็นแบบจำลองสี YUV ก่อนที่จะเข้าสู่กระบวนการแยกตัวอักษรออกจากภาพ

1.3.2 ใช้วิธีการหาส่วนที่เป็นตัวอักษร โดยมีขั้นตอนย่อย 3 ขั้นตอนคือ ขั้นตอนการแบ่งกลุ่มข้อมูลโดยทำการศึกษาการแบ่งกลุ่ม 3 วิธีคือ การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน การแบ่งกลุ่มด้วยเคมีน (K-means Clustering) และการแบ่งกลุ่มข้อมูลด้วยฟัซซี่ซีมีน (Fuzzy C-means Clustering) ซึ่งตัวอักษรในภาพจะต้องเป็นสีดำเท่านั้น ขั้นตอนการเปลี่ยนภาพส่วนหน้าให้เป็นภาพขาว - ดำโดยทำการศึกษาการทำภาพเป็นขาว - ดำโดยใช้อะแดปทีฟไทรเชอไรเซชัน (Adaptive Binarization) และการทำภาพเป็นขาว - ดำโดยใช้ฟัซซี่ (Fuzzy Binarization) และขั้นตอนการลดจำนวนสิ่งรบกวนโดยทำการศึกษาการลดสิ่งรบกวนด้วยการควบคุมความยาวในการวิ่ง (Constrained Run-Length Algorithm หรือ CRLA) และการลดสิ่งรบกวนด้วยการหาขอบของวัตถุและขนาดของวัตถุ (Boundary Detection) ในส่วนที่ไม่ใช่ตัวอักษร

1.3.3 ใช้วิธีการสังเคราะห์พื้นผิวแบบปะติดปะต่อ (Image Quilting) โดยการแบ่งภาพเป็นบล็อกขนาด 10×10 พิกเซลในส่วน of พื้นผิวที่นำมาเรียงต่อกัน และทำการปกปิดความผิดพลาด (Error Concealment) หลังจากการสังเคราะห์พื้นผิว

1.4 คำจำกัดความในงานวิจัย

1.4.1 Foreground หมายถึง ส่วนหน้าหรือตัวอักษรที่ถูกแยกออกมาจากภาพพื้นหลัง

1.4.2 Background หมายถึง รูปภาพที่ถูกแยกตัวอักษรหรือข้อความ

1.4.3 Block size หมายถึง ขนาดภาพย่อยที่ใช้ในการแบ่งภาพ

1.4.4 Text Mask หมายถึง ส่วนที่เป็นตัวอักษรที่ได้จากการหาส่วนที่เป็นตัวอักษรเพื่อใช้ในการระบุตำแหน่งที่ต้องทำการกู้คืนภาพ

1.4.5 Texture Synthesis หมายถึง การสังเคราะห์พื้นผิวเพื่อใช้เป็นส่วนหนึ่งในการกู้คืนรายละเอียดที่สูญเสียไปโดยการสังเคราะห์พื้นผิวแบบปะติดปะต่อของ Efros และ Freeman

1.4.6 Texture Concealment หมายถึง ขั้นตอนการกู้คืนภาพซึ่งประกอบไปด้วยการสังเคราะห์พื้นผิวและการปกปิดความผิดพลาด ซึ่งในงานวิจัยนี้จะเรียกโดยย่อว่า “การปกปิดพื้นผิว”

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 สามารถกู้คืนภาพกลับคืนมาและมีความกลมกลืนของภาพ

1.5.2 ได้ขั้นตอนวิธีที่มีประสิทธิภาพที่สามารถกู้คืนภาพที่สูญเสียรายละเอียดไปกลับคืนมา

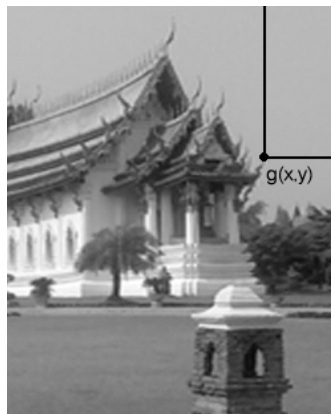
บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

เนื้อหาในบทนี้กล่าวถึงความรู้พื้นฐานและทฤษฎีที่เกี่ยวข้องกับวิชานี้ ได้แก่ ภาพเชิงดิจิทัล แบบจำลองสี การแยกตัวอักษรออกจากพื้นหลังโดยการหาค่าเทรชโฮลดิ้ง ทฤษฎีการหาบริเวณ การกู้คืนรายละเอียดโดยการสังเคราะห์พื้นผิว และการปกปิดความผิดพลาด

2.1 ภาพเชิงดิจิทัล (Digital Image)

ภาพดิจิทัล (Digital Image) คือ ฟังก์ชัน 2 มิติโดยที่ $g(x,y)$ ของค่าความเข้มของแสงโดยที่ x และ y คือ ค่าแสดงตำแหน่งในระบบพิกัดฉาก และค่าฟังก์ชัน g ณ ตำแหน่ง (x,y) ใดๆ จะเป็นสัดส่วนกับความสว่างของแสง (Intensity) ณ ตำแหน่งนั้น และเมื่อ x และ y และค่าฟังก์ชัน g เป็นค่าจำกัด (Finite Value) จึงเรียกรูปภาพนี้ว่า “ภาพดิจิทัล” (Digital Image) แสดงในภาพที่ 2-1



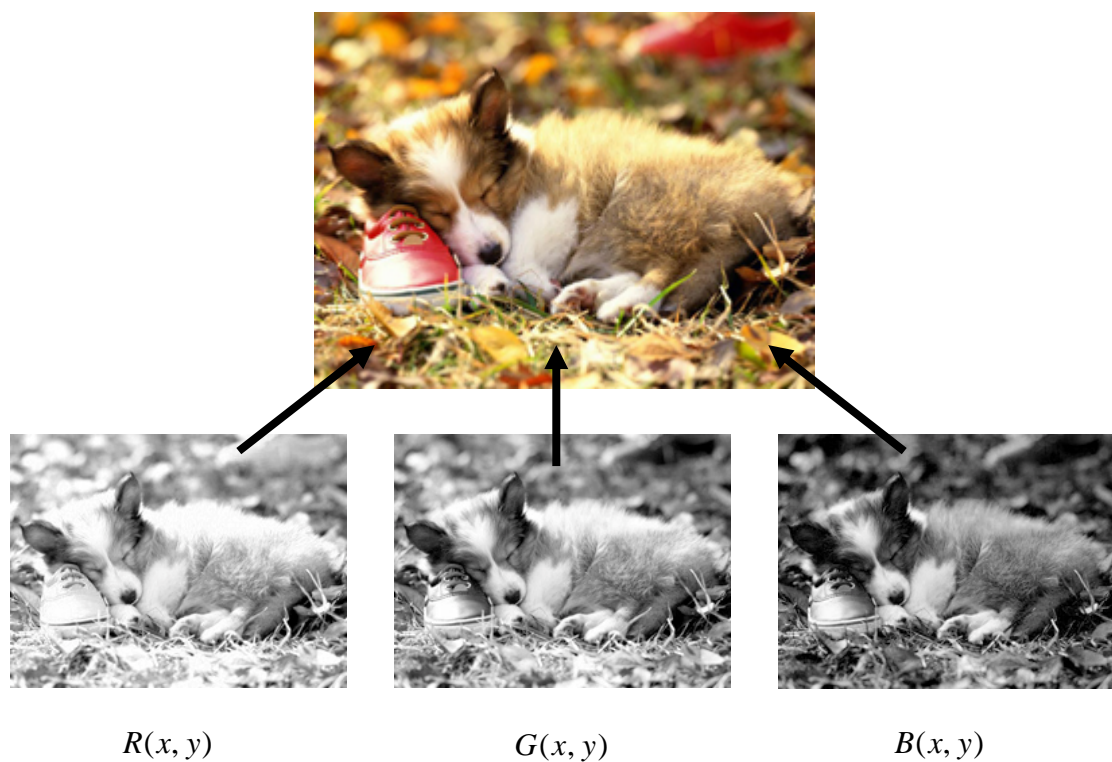
ภาพที่ 2-1 ภาพในความหมายของฟังก์ชัน 2 มิติ $g(x,y)$ ของค่าความเข้มแสง

ภาพดิจิทัลสามารถสร้างได้ด้วยวิธีการต่างๆ เช่น การกราดภาพ (Scan) การถ่ายภาพด้วยกล้องดิจิทัล และการสร้างภาพขึ้นใหม่ด้วยคอมพิวเตอร์ หรือภาพที่เกิดจากกระบวนการทางแสง (Optical Process) ซึ่งเกิดจากพลังงานคลื่นแม่เหล็กไฟฟ้า (Electromagnetic Spectrum) หลายๆ

ช่วงความถี่ เช่นแสงธรรมดา รังสีอินฟราเรด (Infrared) รังสีเอกซ์เรย์ (X-ray) เป็นต้น โดยทั่วไปสามารถแบ่งประเภทของภาพดิจิทัลได้ ดังนี้

2.1.1 ภาพสี (Color Image)

โดยทั่วไปภาพดิจิทัลจะเก็บค่าของความเข้มเป็นปริมาณเวกเตอร์ซึ่ง ในแต่ละจุดภาพจะถูกแทนด้วยค่ามากกว่าหนึ่งค่า เช่น ในระบบแบบจำลองสี RGB ที่ประกอบด้วย 3 องค์ประกอบ คือ ความเข้มแสงสีแดง R สีเขียว G และสีน้ำเงิน B มาผสมกันในอัตราส่วนที่ต่างกันเพื่อประกอบกันเป็นภาพหนึ่งภาพ ตัวอย่างของภาพสี RGB แสดงในภาพที่ 2-2



ภาพที่ 2-2 ตัวอย่างภาพสี RGB

2.1.2 ภาพโทนสีเทา (Gray Scale Image)

เป็นภาพที่มีค่าความเข้มเป็นปริมาณสเกลาร์ บางครั้งเรียกว่า “ภาพสีเดียว” ภาพโทนสีเทาเกิดจากการกำหนดค่าให้กับความเข้มแสงของแต่ละจุดภาพโดย สีขาวจะถูกแทนด้วย “255” และสีดำจะถูกแทนด้วยค่า “0” ในขณะที่ระดับของโทนสีจะเข้มขึ้นเรื่อยๆ จากสีขาวจนถึงสีดำ ดังแสดงในภาพที่ 2-3



ภาพที่ 2-3 ตัวอย่างภาพโทนสีเทา

2.1.3 ภาพขาวดำ (Binary Image)

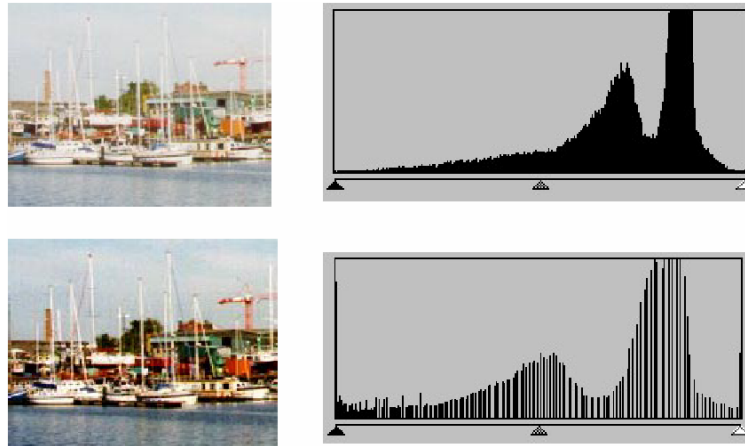
เป็นภาพดิจิทัลที่แต่ละจุดภาพสามารถมีค่าที่เป็นไปได้เพียงแค่ 2 ค่า คือ ค่าสีขาวแทนค่าด้วย “1” และค่าสีดำแทนค่าด้วย “0” บางครั้งเรียกว่า “ภาพสองระดับ” (Bi-level) ภาพขาวดำ (Black-and-white) หรือภาพโมโนโครม (Monochrome) แสดงในภาพที่ 2-4



ภาพที่ 2-4 ตัวอย่างภาพขาว-ดำ

2.1.4 ฮิสโตแกรมของภาพ (Gray-Level Histogram)

เป็นฟังก์ชันที่แสดงว่าในแต่ละระดับความเข้มมีจำนวนจุดภาพ (Pixel) ในภาพที่มีค่าความเข้มนี้เท่าไร เมื่อนำมาเขียนเป็นกราฟ แกนนอนคือระดับความเข้ม (Gray Level) และแกนย่น คือ ความถี่ของค่าความเข้มนั้นๆ (Number of Pixels) ฮิสโตแกรมเป็นการนำเสนอภาพในลักษณะแบบรวมๆ (Global Characteristic) แสดงในภาพที่ 2-5



ภาพที่ 2-5 ลักษณะกราฟฮิสโตแกรมที่ได้จากภาพทางด้านซ้าย

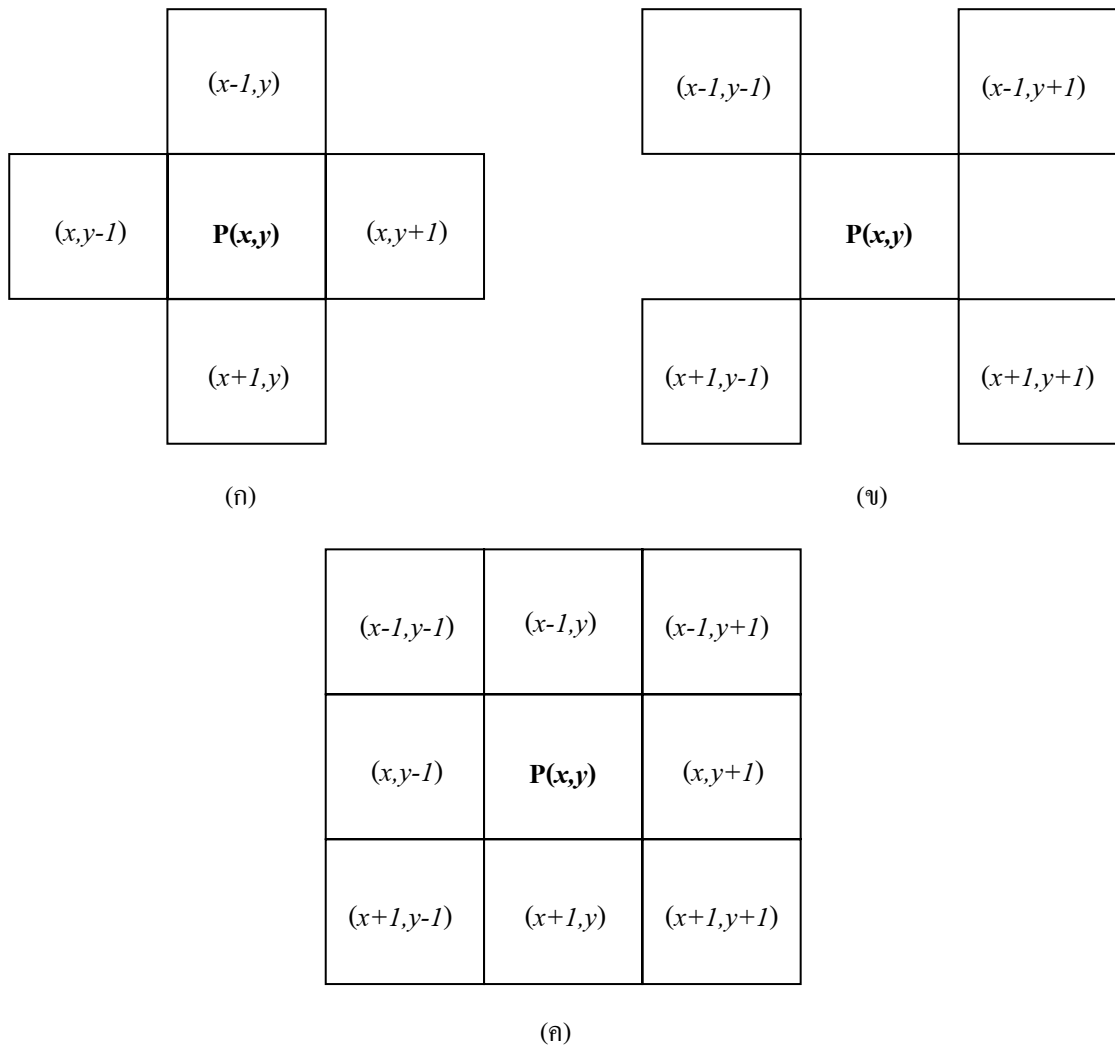
2.1.5 ความสัมพันธ์ระหว่างจุดภาพ (Relation Between Pixels)

เป็นที่ทราบกันว่ามีความน่าจะเป็นสูงมากที่จุด 2 จุดภาพที่อยู่ติดกันจะเป็นบริเวณที่เป็นวัตถุเดียวกัน จุดภาพที่อยู่ติดกันเรียกว่าระบบเพื่อนบ้าน (Neighbor System of a Pixel) เมื่อพิจารณาจุดภาพใดๆ จำเป็นอย่างยิ่งที่จะต้องพิจารณาอิทธิพลของเพื่อนบ้านรอบข้างด้วย เพราะโดยธรรมชาติของภาพแล้วจุดภาพที่อยู่ใกล้เคียงกันมีความเป็นไปได้สูงมากที่จะถูกครอบคลุมโดยวัตถุเดียวกัน ระบบเพื่อนบ้านที่สำคัญได้แก่

2.1.5.1 ระบบเพื่อนบ้านแบบสี่ (4-Neighbor System- $N_4(p)$) พิจารณาจุดภาพ p ซึ่งอยู่ที่ตำแหน่ง (x,y) ใดๆ เพื่อนบ้านของจุดภาพนี้ คือ จุดภาพรอบข้างที่อยู่ในแนวราบ (ซ้าย-ขวา) และในแนวดิ่ง แสดงดังภาพที่ 2-6 (ก) ซึ่งประกอบด้วยจุดภาพ ณ ตำแหน่งต่างๆ ดังนี้ $(x-1,y)$, $(x+1,y)$, $(x,y-1)$ และ $(x,y+1)$

2.1.5.2 ระบบเพื่อนบ้านแบบเส้นทแยงมุม (Diagonal Neighbor- $N_D(p)$) พิจารณาจุดภาพ p ที่อยู่ตำแหน่ง (x,y) ใดๆ เพื่อนบ้านของจุดภาพนี้ คือ จุดภาพรอบข้างที่อยู่ในแนวทแยงมุมทั้งสอง แสดงดังภาพที่ 2-6 (ข) ประกอบด้วยจุดภาพ ณ ตำแหน่งต่างๆ ดังนี้ $(x-1,y-1)$, $(x+1,y-1)$, $(x-1,y+1)$ และ $(x+1,y+1)$

2.1.5.3 ระบบเพื่อนบ้านแบบแปด (8-Neighbor System- $N_8(p)$) พิจารณาจุดภาพ p ซึ่งอยู่ที่ตำแหน่ง (x,y) ใดๆ เพื่อนบ้านของจุดภาพนี้ คือ จุดภาพรอบข้างตามลักษณะของระบบเพื่อนบ้านแบบสี่ และระบบเพื่อนบ้านแบบเส้นทแยงมุมแสดงดังภาพที่ 2-6 (ค) ประกอบด้วยจุดภาพ ณ ตำแหน่งต่างๆ ดังนี้ $(x-1,y-1)$, $(x+1,y-1)$, $(x-1,y+1)$, $(x-1,y)$, $(x+1,y)$, $(x,y-1)$, $(x,y+1)$ และ $(x+1,y+1)$



ภาพที่ 2-6 ระบบเพื่อนบ้านของจุดภาพ p ใดๆ (ก) เป็นระบบเพื่อนบ้านแบบสี่ (ข) เป็นระบบเพื่อนบ้านแบบเส้นทแยงมุม และ (ค) เป็นระบบเพื่อนบ้านแบบแปด

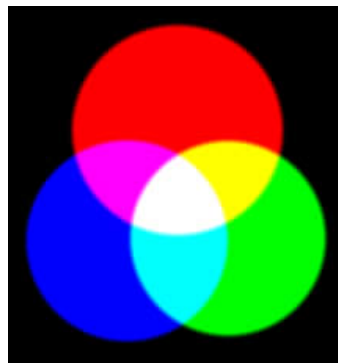
2.2 แบบจำลองสี (Color Model)

แบบจำลองสีที่ใช้อยู่ในปัจจุบันมีอยู่หลายระบบด้วยกัน ทั้งนี้จะขึ้นอยู่กับนำไปใช้ แต่โดยทั่วไปแล้วทุกมาตรฐานจะมีแนวคิดเดียวกันคือ การแทนจุดสีด้วยจุดที่อยู่ภายในพื้นที่ 3 มิติ โดยจะมีแกนอ้างอิงสำหรับจุดสีนั้นในพื้นที่ ซึ่งแต่ละแกนจะมีความเป็นอิสระต่อกัน

2.2.1 ระบบสี RGB

ระบบสี RGB เป็นระบบสีของแสงซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึมจะเกิดแถบสีที่เรียกว่า สเปกตรัม (Spectrum) ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ แดง แสด เหลือง เขียว น้ำเงิน คราม ม่วง ซึ่งเป็นพลังงานอยู่ในรูปของรังสีที่มีช่วงคลื่นที่สายตาสามารถมองเห็นได้ แสง

สีม่วงมีความถี่คลื่นสูงที่สุด คลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วง เรียกว่า “อุลตราไวโอเล็ต” (Ultra Violet) และคลื่นแสงสีแดงมีความถี่คลื่นต่ำที่สุด คลื่นแสงที่ต่ำกว่าแสงสีแดงเรียกว่า “อินฟราเรด” (Infrared) คลื่นแสงที่มีความถี่สูงกว่าสีม่วงและต่ำกว่าสีแดงนั้น สายตามนุษย์ไม่สามารถรับได้ และเมื่อศึกษาดูแล้วแสงสีทั้งหมดเกิดจากแสงสี 3 สี คือ สีแดง (Red) สีน้ำเงิน (Blue) และสีเขียว (Green) ทั้งสามสีถือเป็นแม่สีของแสง เมื่อนำมาฉายรวมกันจะทำให้เกิดสีใหม่ อีก 3 สี คือ สีแดงมาเจนคร่า สีฟ้าไซมอน และสีเหลือง และถ้านำแสงสีทั้งหมดรวมกันจะได้แสงสีขาว แสดงในภาพที่ 2-7 จากคุณสมบัติของแสงนี้เราได้นำมาใช้ประโยชน์ทั่วไปในการฉายภาพยนตร์ การบันทึกภาพวิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์ และการจัดแสงสีในการแสดง เป็นต้น

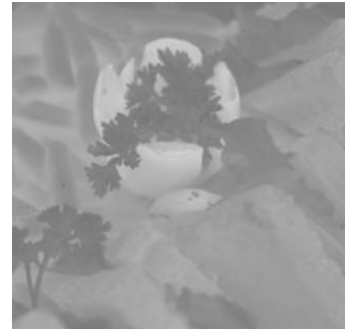
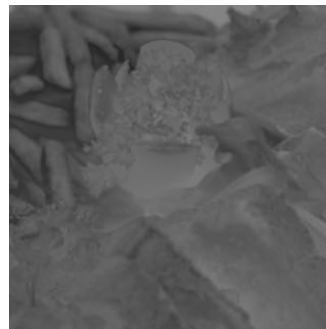


ภาพที่ 2-7 ภาพแสดงระบบสี RGB

ระบบสี RGB สามารถนำไปใช้ในการระบุสีได้มากมายขึ้นอยู่กับความสามารถของระบบที่ใช้ ใช้อยู่ ส่วนมากในการนำไปใช้ 24 บิต ในแต่ละช่องสีจะประกอบด้วย 8 บิต มีค่าระดับสีตั้งแต่ 0 ถึง 255 ระดับสี ฉะนั้นพื้นฐานของ RGB Color Model จะมีทั้งหมด $256 \times 256 \times 256 = 16.7$ ล้านกว่าสี เมื่อค่า RGB ใน 24 บิตต่อ 1 พิกเซล ถูกระบุด้วย 0 – 255 จะได้สีดังตัวอย่างต่อไปนี้ (0, 0, 0) คือ สีดำ (255, 255, 255) คือ สีขาว (255, 0, 0) คือ สีแดง เป็นต้น

2.2.2 ระบบสี YUV

ระบบสีแบบ YUV ใช้สำหรับโทรทัศน์แบบ PAL และ SECAM ซึ่งยังมีใช้อยู่ในหลาย ๆ ประเทศ โดย Y คือค่าความสว่างของภาพ ส่วนสัญญาณ U และ V เป็นสัญญาณที่เก็บค่าสีของภาพ ต่อมาได้มีระบบ YIQ มาใช้แทนเนื่องจากพบว่าสัญญาณ I และ Q สามารถลด Bandwidth ได้มากกว่าสัญญาณ U และ V ในขณะที่ได้ภาพที่มีคุณภาพเท่ากัน ตัวอย่างของภาพสี YUV แสดงในภาพที่ 2-8



Y

U

V

ภาพที่ 2-8 องค์ประกอบทั้ง 3 ของระบบจำลองสี YUV

2.3 ฟัซซี่ลอจิก (Fuzzy Logic)

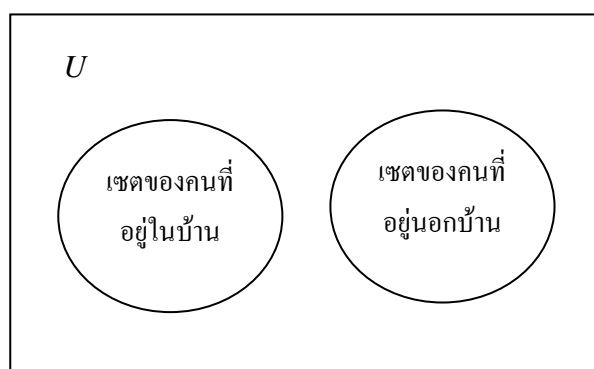
ฟัซซี่ลอจิก เป็นแนวคิดเกี่ยวกับการวิเคราะห์เชิงตรรกะที่แตกต่างไปจากตรรกะแบบเดิมๆ ที่คุ้นเคยกัน ที่มักจะมีเพียงถูกหรือผิด ใช่หรือไม่ใช่ โดยในแวดวงดิจิทัลก็คือ “0” กับ “1” นั่นเอง ซึ่ง ถือว่าเป็นตรรกะที่มีความชัดเจน ไม่มีความคลุมเครือใดๆ แต่ก็มีเหตุการณ์หลายอย่างในความเป็นจริงที่สร้างความยุ่งยากแก่ผู้วิเคราะห์ว่าตรรกะควรเป็นอย่างไร เช่น สีเทาอ่อนปานกลาง ให้บอกว่าคือสีดำใช่หรือไม่ ผู้วิเคราะห์ที่ต่างเพศ ต่างวัย ต่างประสบการณ์ก็จะให้ตรรกะหรือคำตอบที่ไม่เหมือนกัน จะให้ตอบว่าขาวหรือดำก็สร้างความขัดแย้งในใจขึ้น เพราะคำตอบเป็นไปได้เพียง 2 คำตอบเท่านั้น ซึ่งตรรกะแบบเดิมๆ สร้างกรอบความคิดไว้เช่นนั้น จึงนำมาสู่การนำเสนอตรรกะรูปแบบใหม่ เพื่อสะท้อนความเป็นจริงมากที่สุด

ในปี ค.ศ.1965 L. A. Zadeh [8] ได้เสนอทฤษฎีฟัซซี่ สำหรับใช้วิเคราะห์ตรรกะของเหตุการณ์ที่อาจมีความขัดแย้งหรือคลุมเครือ เช่น สีเทาอ่อนปานกลาง ให้บอกว่าคือสีดำใช่หรือไม่ หากตอบแบบฟัซซี่อาจได้คำตอบว่าเป็นสีดำ 50% สีขาว 50% ซึ่งแต่ละคนอาจตอบไม่เหมือนกัน ก็ได้ จะเห็นได้ว่าคำตอบที่ได้จะต่างไปจากแนวคิดเดิมอย่างชัดเจน ดังนั้นการบอกค่าตรรกะแบบฟัซซี่จึงเป็นการสะท้อนความจริงได้มากกว่าตรรกะแบบเดิมนั่นเอง ฟัซซี่ลอจิกถูกออกแบบมา

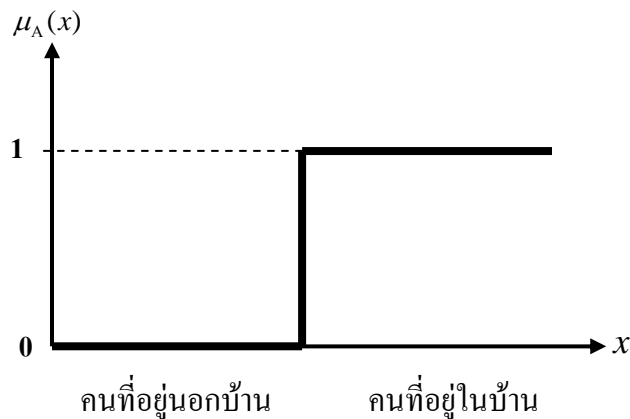
เพื่อใช้เป็นตัวแทนความไม่แน่นอน (Uncertainty) ความคลุมเครือ (Fuzzy) ความไม่ชัดเจน (Exact) สิ่งเหล่านี้เป็นตัวอย่างของความไม่แน่นอนซึ่งเป็นลักษณะทางธรรมชาติที่เกิดขึ้นทั่วไป เซตของเหตุการณ์ที่ไม่แน่นอนเช่นนี้เรียกว่า “ฟัซซีเซต” (Fuzzy Set) จากแนวความคิดของ Zadeh เกี่ยวกับความไม่แน่นอนได้มีการขยายแนวคิดเพื่อนำไปประยุกต์ใช้ในด้านต่าง ๆ เช่น เครื่องปรับอากาศ เครื่องซักผ้า หม้อหุงข้าว เป็นต้น

2.3.1 พื้นฐานแนวคิดแบบฟัซซี [9] ตรรกะแบบฟัซซีเป็นเครื่องมือที่ช่วย ในการตัดสินใจภายใต้ความไม่แน่นอนของข้อมูล โดยยอมให้เกิดความยืดหยุ่น ใช้หลักการที่คล้ายการเลียนแบบวิธีคิดที่ซับซ้อนของมนุษย์ เช่น คนสูงกับคนเตี้ย ถ้าสมศรี สมชาย สมควร และสมศักดิ์ สูง ห้าฟุต ห้า ฟุตครึ่ง หกฟุต และหกฟุตครึ่ง แล้ว ถ้าเรามองโดยใช้แนวคิดแบบตรรกวิทยาเราอาจบอกว่า สมศรีกับสมชายตัวเตี้ย และสมควรกับสมศักดิ์ตัวสูง แต่ถ้ามีสมปองเพิ่มเข้ามาอีกคนโดยที่เขา มีความสูง ห้าฟุตเจ็ดนิ้ว ก็จะทำให้เกิดความคลุมเครือที่ว่า สมปองควรจะจัดอยู่ในกลุ่มพวกที่สูงหรือ เตี้ยดี ลองมองปัญหาโดยใช้แนวคิดตรรกะแห่งความคลุมเครือบ้าง จะเห็นว่าทุกคนเป็นสมาชิก ของกลุ่มคนที่มีความสูงทั้งหมดโดยแต่ละคนมีส่วนความเป็นสมาชิกที่แตกต่างกันไป

ทฤษฎีเซตแบบปกติ หรือเซตทวินัย (Crisp Set) เป็นเซตที่มีค่าความเป็นสมาชิกเพียงสองค่า คือ “0” ไม่เป็นสมาชิก และ “1” เป็นสมาชิก เช่น เซตของคนที่อยู่ในบ้าน สามารถบอกได้ชัดว่า เป็นกลุ่มของคนที่อยู่ในบ้านหรือกลุ่มของคนที่ยอยู่นอกบ้าน ดังภาพที่ 2-9 แสดงเซตย่อยสองเซต คือ เซตของคนที่อยู่ในบ้าน และเซตของคนที่ยอยู่นอกบ้าน จะเห็นได้ว่าน้ำหนึ่งชนิดสามารถเป็นสมาชิกได้เพียงเซตเดียวเท่านั้น ภาพที่ 2-10 จะเห็นว่าคนที่อยู่ในบ้านจะมีค่าความเป็นสมาชิกเป็น 1 ส่วนคนที่ยอยู่นอกบ้านจะมีค่าความเป็นสมาชิกเป็น 0



ภาพที่ 2-9 ตัวอย่างเซตย่อยสองเซต



ภาพที่ 2-10 ฟังก์ชันความเป็นสมาชิกในเซตคนที่อยู่ในบ้าน

2.3.2 ฟัชชีเซต (Fuzzy Set) ทฤษฎีฟัชชีเซต จะครอบคลุมทฤษฎีเซตแบบปกติ โดยฟัชชีเซตยอมให้มีค่าความเป็นสมาชิกของเซตระหว่าง “0” และ “1” เช่น ตัวอย่างเกี่ยวกับความสูง นิยามคำว่า คนสูงในเซตทวินัยอาจกำหนดเป็นคนที่มีความสูงตั้งแต่ 160 ถึง 180 เซนติเมตร โดยนิยามแบบฟัชชีเซตอาจกำหนดเป็นคนที่มีความสูงประมาณ 165 เซนติเมตร ซึ่งเป็นการให้นิยามที่ไม่แสดงถึงขอบเขตที่แน่นอน นิยามของฟัชชีเซต

$$A = \{ (x, \mu_A(x)) \mid x \in X \} \quad (2-1)$$

เมื่อ A หมายถึง ฟัชชีเซต x หมายถึง สมาชิกของเซต (Set Membership) และ $\mu_A(x)$ หมายถึง ฟังก์ชันความเป็นสมาชิก (Membership Function)

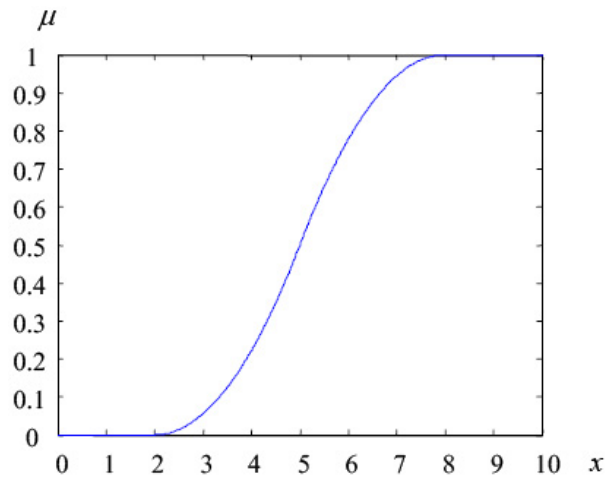
2.3.3 ฟังก์ชันความเป็นสมาชิก เป็นฟังก์ชันที่มีการกำหนดระดับความเป็นสมาชิกของตัวแปรที่ต้องการใช้งาน โดยเริ่มจากการแทนที่กับตัวแทนที่มีความไม่ชัดเจน ไม่แน่นอน และคลุมเครือ ดังนั้นเป็นส่วนที่สำคัญต่อคุณสมบัติหรือการดำเนินการของฟัชชี เพราะรูปร่างของฟังก์ชันความเป็นสมาชิกมีความสำคัญต่อกระบวนการคิดและแก้ไขปัญหา โดยฟังก์ชันความเป็นสมาชิกจะไม่สมมาตรกันหรือสมมาตรกันทุกประการก็ได้

ชนิดของฟังก์ชันความเป็นสมาชิกที่ใช้งานทั่วไปมีหลายชนิด ในที่นี้จะกล่าวถึงเพียง 3 ชนิด ดังนี้

2.3.3.1 ฟังก์ชันรูปตัวเอส (S-shaped Curve Membership Function) ฟังก์ชันรูปตัวเอส มีพารามิเตอร์ทั้งหมด 2 ค่า คือ $\{a, b\}$

$$s(x:a,b) = \begin{cases} 0 & x < a \\ 2\left(\frac{x-b}{b-a}\right)^2 & a \leq x \leq \frac{a+b}{2} \\ 1-2\left(\frac{x-b}{b-a}\right)^2 & \frac{a+b}{2} \leq x \leq b \\ 1 & x \geq b \end{cases} \quad (2-2)$$

ตัวอย่างเมื่อกำหนดให้ค่าความเป็นสมาชิกมีค่าเป็น $a = 2$ และ $b = 8$ จะได้กราฟดังแสดงในภาพที่ 2-11

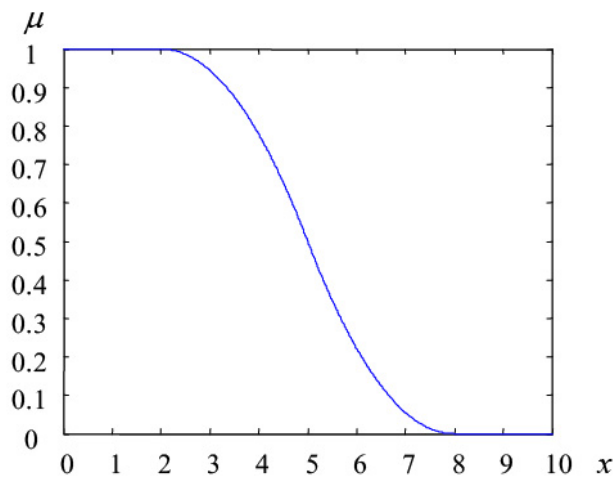


ภาพที่ 2-11 กราฟของฟังก์ชันรูปตัวเอส

2.3.3.2 ฟังก์ชันรูปตัวแซด (Z-shaped Curve Membership Function) ฟังก์ชันรูปตัวแซดมีพารามิเตอร์ทั้งหมด 2 ค่า คือ $\{a,b\}$

$$z(x:a,b) = \begin{cases} 1 & x < a \\ 1-2\left(\frac{x-b}{b-a}\right)^2 & a \leq x \leq \frac{a+b}{2} \\ 2\left(\frac{x-b}{b-a}\right)^2 & \frac{a+b}{2} \leq x \leq b \\ 0 & x \geq b \end{cases} \quad (2-3)$$

ตัวอย่างเมื่อกำหนดให้ค่าความเป็นสมาชิกมีค่าเป็น $a = 2$ และ $b = 8$ จะได้กราฟดังแสดงในภาพที่ 2-12

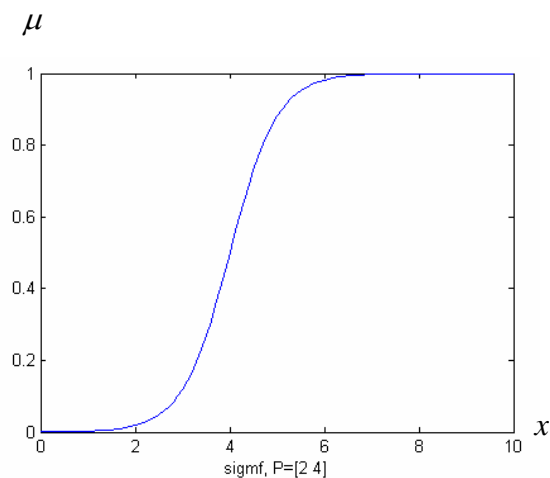


ภาพที่ 2-12 กราฟของฟังก์ชันรูปตัวแซด

2.3.3.3 ฟังก์ชันรูปโค้งกลับอย่างตัวซีหรือเอส (Sigmoid Curve Membership Function)
โดยธรรมชาติแล้วฟังก์ชันรูปโค้งกลับอย่างตัวซีหรือเอสจะเปิดขวาหรือซ้าย ฟังก์ชันรูปโค้งกลับ
อย่างตัวซีหรือเอสมีพารามิเตอร์ทั้งหมด 2 ค่า คือ $\{a, c\}$

$$f(x, a, c) = \frac{1}{1 + e^{-a(x-c)}} \quad (2-4)$$

ตัวอย่างเมื่อกำหนดให้ค่าความเป็นสมาชิกมีค่าเป็น $a = 2$ และ $c = 4$ จะได้กราฟดังแสดงใน
ภาพที่ 2-13



ภาพที่ 2-13 กราฟของฟังก์ชันรูปโค้งกลับอย่างตัวซีหรือตัวเอส

การเลือกฟังก์ชันของความเป็นสมาชิก จะต้องเลือกตามความเหมาะสมความครอบคลุมของข้อมูลที่จะรับเข้ามา โดยสามารถที่ทับซ้อนกันเพื่อให้การดำเนินงานราบเรียบ ซึ่งมีความเป็นสมาชิกหลายค่าได้ และฟังก์ชันความเป็นสมาชิกเปลี่ยนแปลงแก้ไขให้เหมาะกับงานที่กำลังปฏิบัติงานหรือตามความต้องการ

ในงานวิจัยนี้ได้นำฟังก์ชันลอจิกเข้ามาช่วยในการแปลงภาพเป็นขาว-ดำ ในการแบ่งข้อมูลออกเป็นสองส่วนซึ่งก็คือ “0” กับ “1”

2.4 การแบ่งกลุ่มข้อมูล (Clustering)

การจำแนกข้อมูลออกเป็นกลุ่ม เมื่อเราไม่ทราบรายละเอียดในแง่ของจำนวนกลุ่ม หรือลักษณะเฉพาะของกลุ่ม แน่ชัด เราจำเป็นต้องแบ่งข้อมูลโดยใช้ความเหมือนหรือต่างกันของข้อมูลเป็นเกณฑ์ในการแบ่งกลุ่ม การทำงานเช่นนี้เราเรียกว่า การแบ่งกลุ่มข้อมูล (Clustering) ข้อมูลแต่ละกลุ่มที่ถูกแบ่งเราเรียกว่าคลัสเตอร์ (Cluster) จุดมุ่งหมายของเทคนิคเหล่านี้ต้องการจะแบ่งกลุ่มข้อมูลออกเป็นกลุ่มย่อยๆ โดยที่ข้อมูลภายในกลุ่มเดียวกันจะต้องมีความเหมือนกันมาก และข้อมูลต่างกลุ่มกันจะต้องมีความต่างกันมาก โดยหลักการแล้วสิ่งที่จำเป็นสำหรับเทคนิคเหล่านี้จำเป็นจะต้องมีฟังก์ชันความเหมือน (Similarity Function) ที่ชัดเจน ซึ่งจะมีผลกับกลุ่มข้อมูลที่แบ่งออกมาอย่างมาก ส่วนใหญ่แล้วฟังก์ชันเหล่านี้ถูกใช้ในรูปของฟังก์ชันระยะห่าง (Distance Function) ซึ่งเป็นการให้ค่าความต่างกันของข้อมูลมากกว่า โดยจะแตกต่างกันไปตามชนิดของข้อมูล เช่น ข้อมูลที่เป็นตัวเลข (Number) หรือ ข้อมูลที่เป็นประเภท (Category) ฟังก์ชันระยะห่างที่นิยมกับข้อมูลที่เป็นตัวเลข เช่น ฟังก์ชันระยะห่างแมนฮัตตัน (Manhattan Distance) ฟังก์ชันระยะห่างยูคลีเดียน (Euclidean Distance) และฟังก์ชันความเหมือนแจ็กการ์ด (Jaccard Similarity Function) โดยแสดงในสมการที่ 2-5 ถึง 2-7 ตามลำดับ

ฟังก์ชันระยะห่างแมนฮัตตัน

$$D_1 = |\vec{X}_1 - \vec{X}_2| = \sum_{i=1}^d |\vec{X}_1^{(i)} - \vec{X}_2^{(i)}| \quad (2-5)$$

ฟังก์ชันระยะห่างยูคลีเดียน

$$D_2 = \sqrt{(\vec{X}_1 - \vec{X}_2)^2} \quad (2-6)$$

ฟังก์ชันความเหมือนแจ็กการ์ด

$$D_3 = \frac{X_1 \cap X_2}{X_1 \cup X_2} \quad (2-7)$$

โดยที่ D คือ ค่าระยะห่าง

X_1 คือ ข้อมูลที่ 1

X_2 คือ ข้อมูลที่ 2

d คือ จำนวนมิติของข้อมูล

2.4.1 การแบ่งกลุ่มด้วยเคมีน (K-means Clustering)

เคมีน [14] เป็นที่รู้จักกันดีในการแก้ปัญหการแบ่งกลุ่ม ซึ่งเป็นวิธีการที่ง่ายในการแบ่งกลุ่มข้อมูลที่อยู่จำนวนกลุ่มที่ต้องการแบ่ง และเหมาะกับข้อมูลที่มีขนาดใหญ่ แนวคิดหลักของวิธีการนี้คือการกำหนดค่ากลาง (Centroid หรือ Center) ของแต่ละกลุ่ม k การกำหนดค่ากลางเริ่มต้นของแต่ละกลุ่มจะมีความสำคัญเป็นอย่างมากเนื่องจากค่ากลางจะเป็นตัวกำหนดว่าข้อมูลควรจะอยู่ในกลุ่มใด ถ้าการกำหนดค่ากลางออกมาต่างกันผลลัพธ์ที่ได้จากการแบ่งกลุ่มข้อมูลก็จะต่างกันด้วยเมื่อได้ค่ากลางของกลุ่มแล้วก็จะทำการกำหนดข้อมูลลงไปในกลุ่มโดยดูจากค่ากลางที่ใกล้กับข้อมูล ตัวอย่างภาพการแบ่งกลุ่มด้วยเคมีน แสดงในภาพที่ 2-14 ซึ่งสรุปขั้นตอนการแบ่งกลุ่มแบบเคมีนได้ดังนี้

กำหนดจำนวนกลุ่มที่ต้องการแบ่งข้อมูล และกำหนดค่ากลางเริ่มต้นของกลุ่มด้วยการสุ่ม จากนั้นกำหนดข้อมูลลงไปในแต่ละกลุ่มซึ่งข้อมูลจะมีค่าใกล้กับค่ากลางกลุ่ม ทำการคำนวณค่ากลางของกลุ่มใหม่ ทำซ้ำจนกว่าค่ากลางของกลุ่มจะไม่มีเปลี่ยนแปลง ซึ่งการแบ่งข้อมูลในกลุ่มสามารถคำนวณได้จากระยะทางที่น้อยที่สุด (Minimized) หรือจากการคำนวณฟังก์ชันเป้าหมาย (Objective Function) ดังสมการที่ 2-8

$$J = \sum_{j=1}^k \sum_{i=1}^n \|x_i^{(j)} - c_j\|^2 \quad (2-8)$$

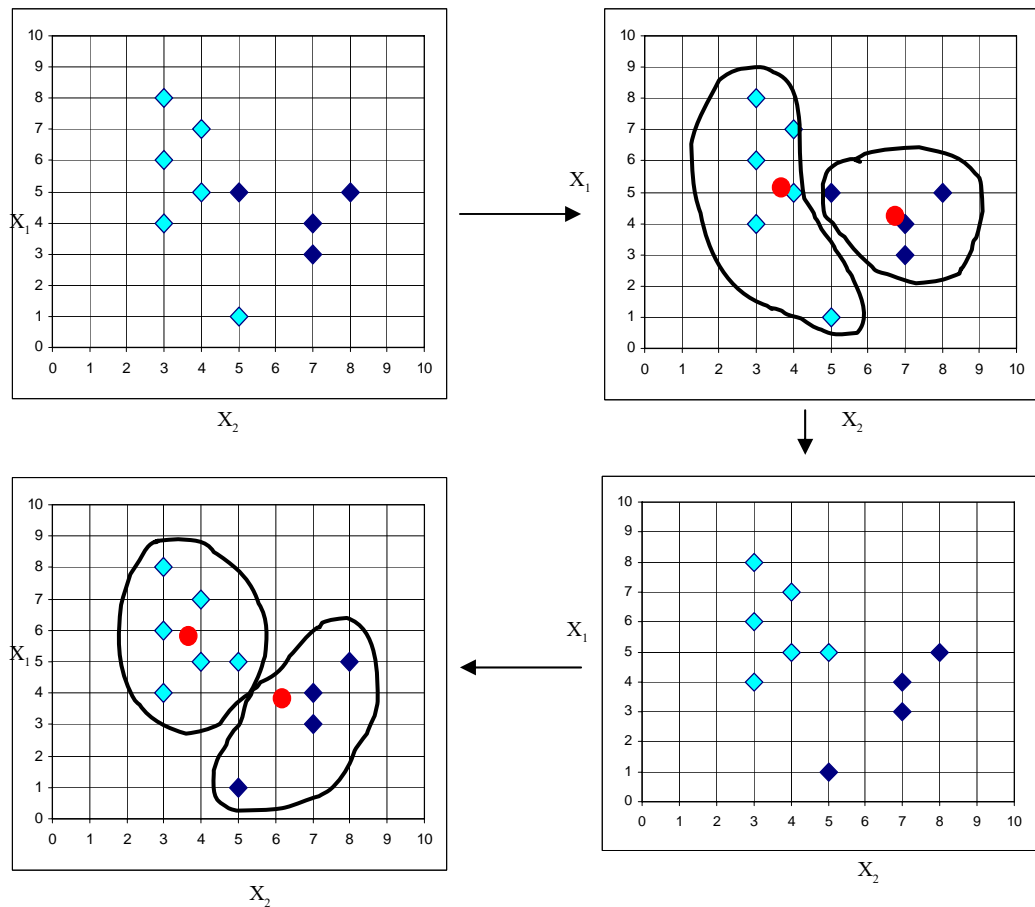
โดยที่ k คือ จำนวนกลุ่มของข้อมูล

n คือ จำนวนข้อมูล

$x_i^{(j)}$ คือ ข้อมูลที่ตำแหน่งที่ i ในกลุ่มที่ j

c_j คือ ค่ากลางของกลุ่มที่ j

J คือ ระยะทางที่น้อยที่สุดจากข้อมูลถึงค่ากลางของแต่ละกลุ่ม



ภาพที่ 2-14 ขั้นตอนการแบ่งกลุ่มแบบเคมีน

2.4.2 การแบ่งกลุ่มด้วยฟัซซี่ซีมีน (Fuzzy C-means Clustering)

ฟัซซี่ซีมีนหรือ FCM [15] เป็นวิธีการแบ่งกลุ่มข้อมูลโดยที่ข้อมูลจะมีค่าความเป็นสมาชิกของแต่ละกลุ่ม การทำงานของ FCM จะเริ่มจากการกำหนดค่ากลางของแต่ละกลุ่มด้วยการสุ่ม ซึ่งตั้งเป้าหมายไว้ที่ค่ากลาง (Mean) ของแต่ละกลุ่ม แต่ส่วนมากแล้วมักจะไม่ใช่ FCM ได้ทำการกำหนดค่าความเป็นสมาชิกของแต่ละกลุ่มเอาไว้ด้วย โดยจะทำการคำนวณหาค่ากลางของแต่ละกลุ่มใหม่ และค่าความเป็นสมาชิกของข้อมูลด้วยการทำซ้ำ (Iterative) ไปเรื่อยๆ จนกว่าค่ากลางไม่มีการเปลี่ยนตำแหน่งหรือถึงค่าการทำซ้ำที่กำหนดไว้ ค่าการทำซ้ำจะขึ้นอยู่กับระยะทางที่น้อยที่สุดจากข้อมูลถึงค่ากลางของกลุ่มโดยมีค่าความเป็นสมาชิกเข้ามามีส่วนร่วมในการตัดสินใจด้วย ซึ่งสามารถคำนวณได้จากสมการที่ 2-9

$$J_m = \sum_{i=1}^N \sum_{j=1}^C u_{ij}^m \|x_i - c_j\|^2 \quad (2-9)$$

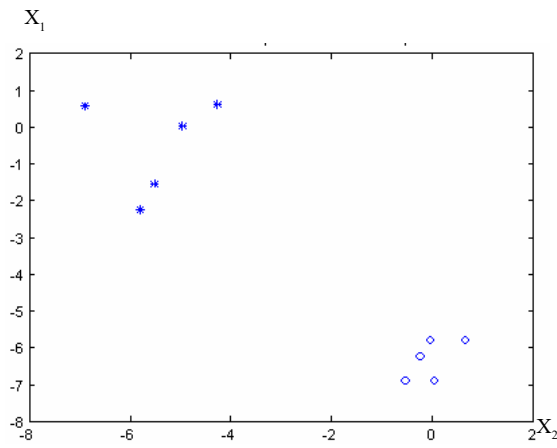
โดยที่ m คือ พหุนามพารามิเตอร์ที่มีค่ามากกว่า 1

u_{ij}^m คือ ค่าความเป็นสมาชิกของ x_i ในกลุ่มที่ j

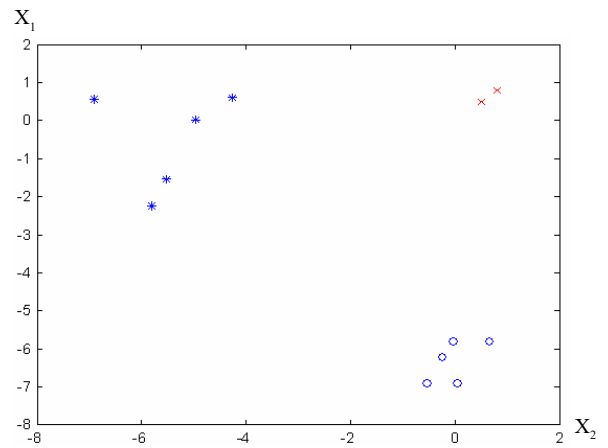
N คือ จำนวนข้อมูล

C คือ จำนวนกลุ่มของข้อมูล

c_j คือ ค่ากลางของกลุ่มที่ j

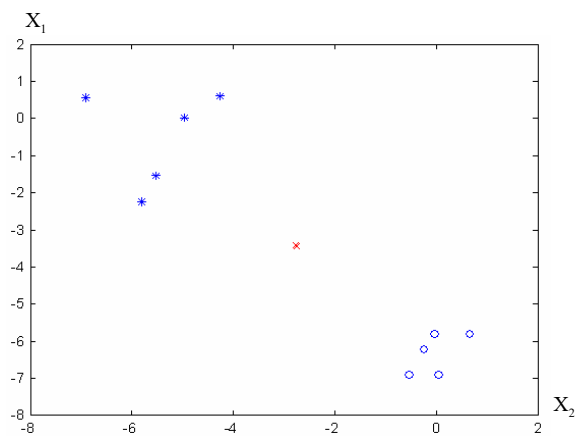


(ก)

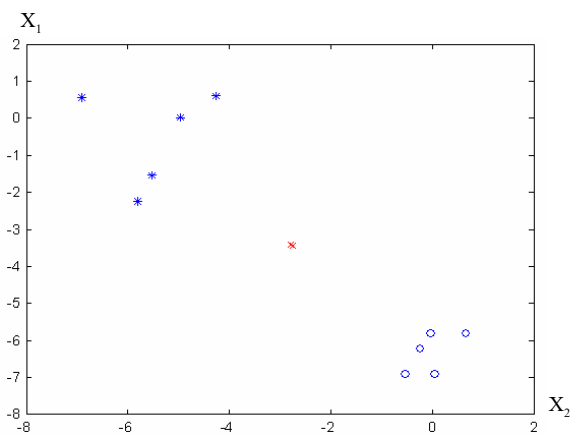


(ข)

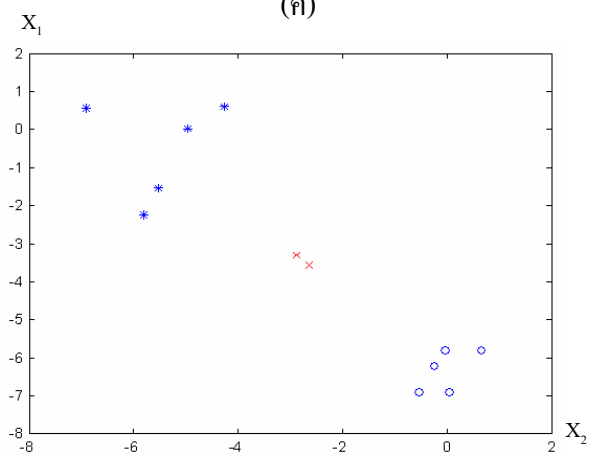
ภาพที่ 2-15 ขั้นตอนการแบ่งกลุ่มแบบฟัซซีซีมีน (ก) กลุ่มข้อมูล (ข) กำหนดค่ากลางเริ่มต้นของแต่ละกลุ่มและหาค่าความเป็นสมาชิกของข้อมูล (ค) คำนวณค่ากลางของกลุ่มและค่าความเป็นสมาชิกใหม่ (ง) คำนวณค่ากลางของกลุ่มและค่าความเป็นสมาชิกใหม่รอบที่ 2 (จ) คำนวณค่ากลางของกลุ่มและค่าความเป็นสมาชิกใหม่รอบที่ 5 (ฉ) คำนวณค่ากลางของกลุ่มและค่าความเป็นสมาชิกใหม่รอบที่ 10 (ช) คำนวณค่ากลางของกลุ่มและค่าความเป็นสมาชิกใหม่รอบที่ 13 ซึ่งค่ากลางของกลุ่มไม่มีการเปลี่ยนตำแหน่งอีก



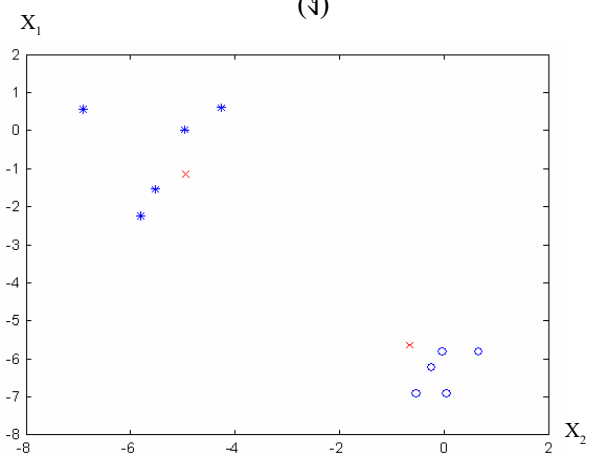
(ก)



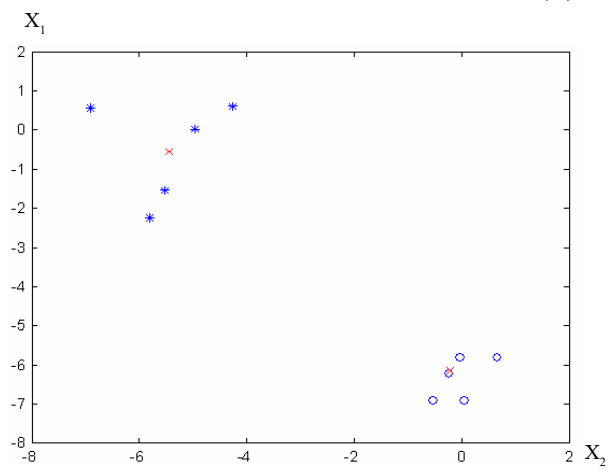
(ง)



(จ)



(ฉ)



(ช)

ภาพที่ 2-15 (ต่อ)

2.5 การตัดแบ่งภาพ (Image Segmentation)

การตัดแบ่งภาพเป็นการทำเพื่อหาอาณาเขตของบริเวณต่างๆ ในภาพซึ่งแต่ละบริเวณอาจสื่อความหมายที่แตกต่างกันออกไป เช่น บริเวณที่เป็นพื้นหลัง บริเวณที่มีค่าความเข้มที่เท่ากันซึ่งมีความเป็นไปได้สูงที่จะเป็นบริเวณที่สมนัยกับวัตถุเชิงกายภาพวัตถุเดียวกัน เป็นต้น โดยทั่วไปการตัดแบ่งภาพแบ่งออกเป็น 2 กลุ่ม คือ การตัดแบ่งตัดโดยอาศัยความสม่ำเสมอของลักษณะของบริเวณแต่ละบริเวณ (Similarity-Based Segmentation) การตัดแบ่งโดยใช้ความไม่ต่อเนื่องของลักษณะของแต่ละบริเวณ (Discontinuity-Based Segmentation) โดยพิจารณาตำแหน่งของจุดเส้นขอบ (Edge) ซึ่งเกิดจากการเชื่อมกันของจุดภาพหลายๆ จุดภาพเข้าด้วยกัน เป็นต้น

2.5.1 ทฤษฎีเทรชโฮลดิ้ง (Thresholding Theory)

สำหรับภาพบางชนิดจะมีลักษณะวัตถุที่เราสนใจ ซึ่งมีความเข้มที่คงที่เมื่อเทียบกับพื้นหลัง ตัวอย่างได้แก่ ภาพของตัวอักษร (Text) เป็นต้น ซึ่งภาพเหล่านี้จะมีความเข้มของวัตถุที่เราสามารถแยกออกพื้นหลังได้อย่างชัดเจน (มีความเข้มขึ้นสองระดับได้แก่ความเข้มของวัตถุและความเข้มของพื้นหลัง)

การทำการแบ่งภาพเป็นส่วนย่อย (Segmentation) สามารถทำได้โดยการกำหนดค่าเทรชโฮลด์ (Threshold) ซึ่งเป็นค่าความเข้มให้มีค่าที่สามารถแยกความแตกต่างของวัตถุและพื้นหลังได้ตัวอย่าง อย่างเช่น ภาพของตัวอักษรที่มีความเข้มของตัวอักษรเป็น “0” (สีดำ) และมีความเข้มของพื้นหลังเป็น “255” (สีขาว) การหาค่าเทรชโฮลดิ้งสามารถหาได้ดังนี้

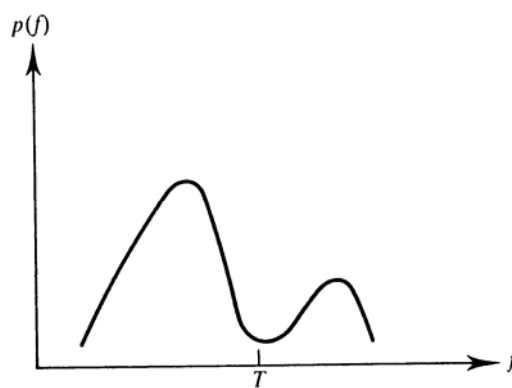
2.5.1.1 โกลบอลเทรชโฮลดิ้ง (Global Thresholding)

เป็นการให้ค่าเทรชโฮลดิ้ง เป็นค่าคงที่ค่าหนึ่ง โดยใช้ฮิสโตแกรมของภาพ ซึ่งค่าเทรชโฮลดิ้งควรที่จะเลือกค่าฮิสโตแกรมที่อยู่จุดต่ำสุดที่อยู่ระหว่างจุดสูงสุด (Peaks)

$$g(x, y) = \begin{cases} 1 & \text{if } (x, y) > T \\ 0 & \text{otherwise} \end{cases} \quad (2-10)$$

เมื่อ $g(x, y)$ คือ ข้อมูลภาพ ณ ตำแหน่งที่ x, y

T คือ ค่า Threshold



ภาพที่ 2-16 ค่าเทรชโฮล (T) คือค่าฮิสโตแกรมที่อยู่ต่ำสุดที่อยู่ระหว่างจุดสูงสุด

2.5.1.2 อะแดพทีฟเทรชโฮลดิ้ง (Adaptive Thresholding)

เป็นการหาค่าเทรชโฮลดิ้งซึ่งมีได้หลายค่า โดยจะทำการหาค่าเทรชโฮลจากค่าความเข้มสีในบริเวณกรอบสี่เหลี่ยมของภาพ ซึ่งกรอบนี้ต้องมีขนาดที่เหมาะสม แต่ถ้าภาพไม่มีลักษณะเป็นขอบสูงชันพอ ค่าเทรชโฮลจะมีผลกระทบต่อตำแหน่งขอบเขตและขนาดของวัตถุจริง ทำให้การวัดขนาดและการคำนวณหาพื้นที่ที่คลาดเคลื่อนได้

2.5.2 การหาบริเวณ (Boundary Detection)

เป็นการแบ่งภาพออกเป็นกลุ่มๆ โดยการค้นหาขอบของวัตถุของแต่ละกลุ่ม การหาขอบสามารถทำได้โดยใช้การหาขอบของวัตถุ (Edge Detection) ซึ่งถ้าเราทราบเส้นรอบวัตถุ ก็จะสามารถคำนวณหาพื้นที่ (ขนาด) ของวัตถุนั้นได้

2.5.2.1 เทมเพลตสำหรับหาขอบภาพ

การหาขอบภาพในแนวนอนอย่างง่าย ทำโดยการหาผลต่างระหว่างจุดหนึ่งกับจุดที่อยู่ข้างล่าง (หรือข้างบน) ของจุดนั้น ดังนี้

$$Y_d(x, y) = I(x, y) - I(x, y + 1) \quad (2-11)$$

โดยที่ Y_d คือ ค่าความแตกต่างในแนวแกนตั้ง

$I(x, y)$ คือ ค่าความเข้มแสงของจุดภาพ

การหาขอบภาพในแนวตั้งก็สามารถหาได้เช่นเดียวกันคือ

$$Y_d(x, y) = I(x, y) - I(x, y - 1) \quad (2-12)$$

โดยที่ Y_d คือ ค่าความแตกต่างในแนวแกนอน

ถ้าต้องการรวมผลต่างของค่าความแตกต่างในแนวแกนอน และแกนตั้งเข้าด้วยกัน เพื่อที่จะได้มีตัววัดความแรงของขอบภาพ (Gradient Magnitude) เพียงตัวเดียว เนื่องจากค่าความแตกต่างอาจมีค่าเป็นบวกหรือลบ ดังนั้น การบวกค่าความแตกต่างของทั้งสองแกนอาจทำให้ขอบภาพเกิดการหักล้างกันเอง ในทางปฏิบัติ จะต้องนำค่าสัมบูรณ์ (Absolute Value) หรือค่ากำลังสอง (Squared Value) ของค่าความแตกต่างของทั้งสองแกนมาบวกกันแทน

นอกจากหาความแรงของขอบภาพแล้ว การหาทิศทางของขอบภาพ (Gradient Direction) ก็ มีประโยชน์เช่นกัน การหาทิศทางของขอบภาพสามารถทำได้โดยใช้สมการต่อไปนี้

$$GD(x, y) = \tan^{-1} \left\{ \frac{Y_d(x, y)}{X_d(x, y)} \right\} \quad (2-13)$$

โดยที่ $GD(x, y)$ คือ ทิศทางของขอบภาพที่ตำแหน่ง (x, y)

2.5.2.2 การตามขอบวัตถุ (Edge Following)

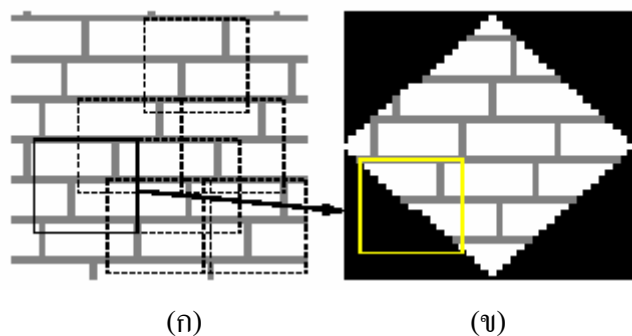
เมื่อเราทราบจุดใดจุดหนึ่งบนขอบภาพเราจะสามารถหาจุดข้างเคียงที่เป็นขอบวัตถุ และสามารถวนไปตามขอบวัตถุจนกลับมายังจุดเริ่มต้นได้ วิธีการตามขอบวัตถุอย่างง่ายทำได้ดังนี้ ให้สมมติให้จุด (x, y) เป็นจุดใดจุดหนึ่งบนขอบวัตถุ จากนั้นตั้งค่าแฟล็กให้จุด (x, y) ว่าเคยผ่านมาแล้ว คำนวณหาความแรงของขอบของจุดทั้งแปดที่อยู่รอบจุด (x, y) โดยเลือกจุด 3 จุดที่มีความแรงสูงสุด แล้วใส่ไว้ในอะเรย์ (Array) แบบ 3 คอลัมน์ โดยเรียงตามความแรงของขอบ เลือกจุดที่มีความแรงสูงสุด แล้วพิจารณาว่าตำแหน่งของจุดอยู่ในทิศทางใด เมื่อเทียบกับจุด (x, y) จากนั้นกำหนดให้ทิศทางคือ d จากนั้นคำนวณหาความแรงของขอบใหม่แต่พิจารณาเฉพาะ จุดที่อยู่ใน 3 ทิศทาง คือ ทิศทาง d , ทิศทาง $(d+1) \bmod 8$ และทิศทาง $(d-1) \bmod 8$ หากไม่พบจุดใดเลยที่มีค่าความแรงของขอบสูงพอในทิศทางที่เคลื่อน ที่ไป ให้ลบจุดนั้นออกจากอะเรย์ และเลือกจุดที่มีความแรงของขอบน้อยกว่าในอันดับถัดไป แล้วทำการหาความแรงของขอบใหม่อีกครั้ง หากทั้ง 3 จุดในแถวถูกลบออกหมด ให้ถอยกลับไปใช้จุดที่เหลือในแถวถัดไป ให้หยุดเมื่อวนกลับมายังจุดเดิม หรือเมื่อมีการใช้เวลาตามขอบภาพนานเกินไป

2.6 การสังเคราะห์พื้นผิว (Synthesis Texture)

พื้นผิว เป็นลักษณะเฉพาะที่ใช้อธิบายความหยาบ ความละเอียด หรือความซับซ้อนของวัตถุภายในภาพ ซึ่งแต่ละภาพอาจจะประกอบด้วยวัตถุที่มีลักษณะพื้นผิวแตกต่างกันออกไป พื้นผิวสามารถแบ่งได้เป็น 2 กลุ่มใหญ่ [5] คือ Deterministic Texture และ Stochastic Texture ซึ่ง Deterministic Texture คือการจำแนกโดยเซตของจุดกำเนิด (Set of Primitives) กับวิธีการวางที่มีกฎเกณฑ์ (Placement Rule) เช่นพื้นกระเบื้อง ในขณะที่ Stochastic Texture คือพื้นผิวที่ค่อนข้างยากต่อการจำแนก เช่น แกรนิต เปลือกไม้ และทราย เป็นต้น

2.6.1 การสังเคราะห์หนึ่งจุดภาพ (Synthesis One Pixel)

เป็นการเริ่มการสังเคราะห์จากจุดภาพใดๆ ในภาพ I โดยให้ $I_{smp} \subset I_{real}$ ซึ่ง I_{real} เป็นภาพพื้นผิวแบบไม่จำกัด (Infinite Texture) ให้ $p \in I$ ซึ่งเป็นจุดภาพ และให้ $\omega(p) \subset I$ ซึ่งเป็นภาพสี่เหลี่ยมจัตุรัสชิ้นเล็กๆ ของขนาด w โดยมี p เป็นจุดศูนย์กลาง การสังเคราะห์ค่า p จะเริ่มจากการประมาณการสุ่มของการกระจาย (Probability Distribution) $P(p | \omega(p))$ และตัวอย่างที่ได้จากการสุ่ม

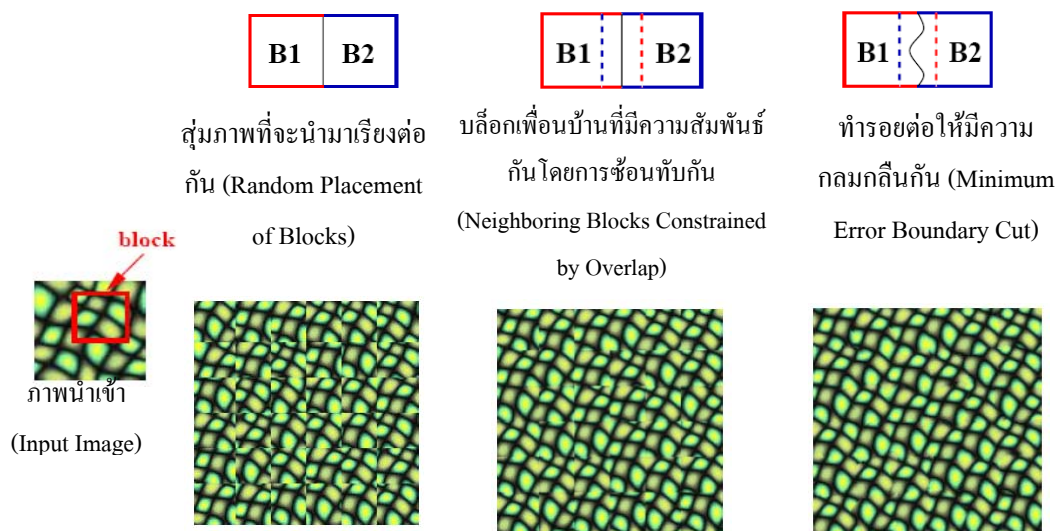


ภาพที่ 2-17 (ก) ตัวอย่างภาพพื้นผิว โดยภาพที่อยู่บนสี่เหลี่ยมเส้นทึบเป็นตัวอย่างที่มีเพื่อนบ้านคล้ายกับส่วนที่ต้องการสังเคราะห์ และ (ข) ภาพที่นำมาทำการสังเคราะห์พื้นผิว

จากภาพที่ 2-17 ภาพ (ข) คือภาพที่เราต้องการนำมาสังเคราะห์พื้นผิวโดยที่มีภาพตัวอย่างคือภาพ (ก) เริ่มการสังเคราะห์โดยการกำหนดจุดเริ่มต้นที่ต้องการสังเคราะห์ ในภาพ (ข) คือจุดที่อยู่ตรงกลางในกรอบสี่เหลี่ยมสีเหลือง และจะทำการเปรียบเทียบกับภาพตัวอย่างในภาพ (ก) ซึ่งมันจะพบว่าภาพที่อยู่รูปสี่เหลี่ยมทึบนั่นตรงกับจุดที่เราต้องการทำการสังเคราะห์

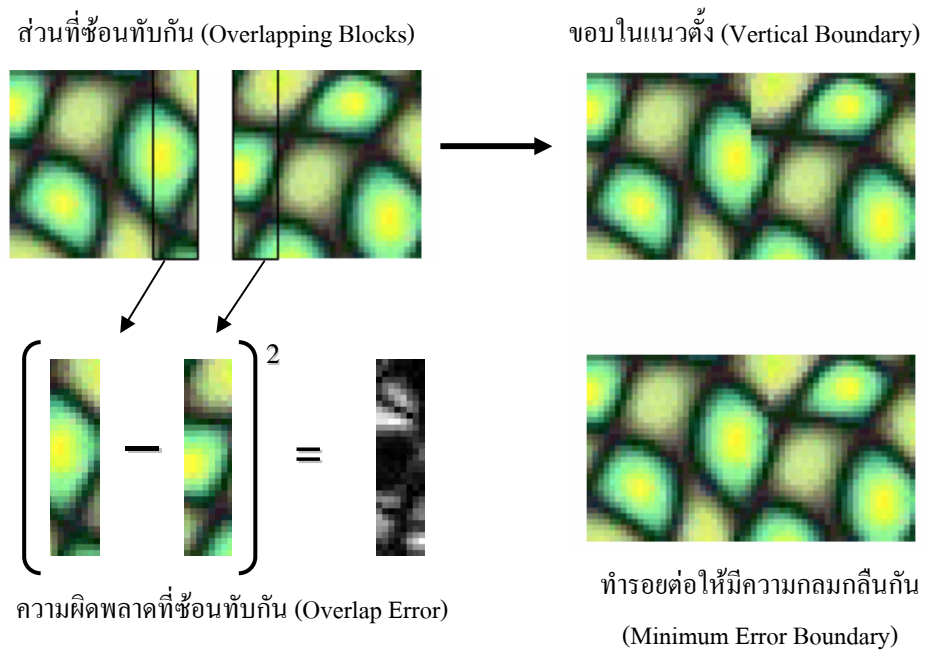
2.6.2 การสังเคราะห์แบบปะติดปะต่อ (Image Quilting)

การสังเคราะห์ [6] จากการกำหนดบล็อก B_i ที่จะใช้ในการสังเคราะห์เป็นสี่เหลี่ยมจัตุรัสจากเซต S_B ของบล็อกที่ซ้อนทับกันในภาพนำเข้า ในการสังเคราะห์ภาพใหม่จะเริ่มจากการสุ่มบล็อกจาก S_B ดังแสดงในภาพ 2-18(ก) ซึ่งผลที่ได้ภาพจะยังคงมีขอบของบล็อกให้เห็นอย่างชัดเจนและภาพที่ได้ยังเห็นได้ชัดถึงความไม่ต่อเนื่องกันของภาพ ดังนั้นจึงต้องทำการหาส่วนที่บล็อกซ้อนทับกันของบล็อกในภาพใหม่ เพื่อทำการหาบริเวณที่เพื่อนบ้านมีส่วนที่ซ้อนทับกัน ดังแสดงในภาพ 2-18(ข) ภาพที่ได้มีโครงสร้างของภาพที่ดีขึ้นแต่ก็ยังคงมีขอบระหว่างบล็อกให้เห็นอยู่ดี จึงต้องทำการเกลี่ยขอบตามขวางอีกครั้งหนึ่งเพื่อให้ขอบของบล็อกที่ปรากฏกลมกลืนกับภาพ



ภาพที่ 2-18 การปะติดปะต่อพื้นผิว (Quilting Texture) (ก) บล็อกที่ได้จากการสุ่มแล้วนำมาเรียงต่อกัน (ข) นำบล็อกเพื่อนบ้านมาซ้อนทับกันซึ่งส่วนที่ซ้อนทับกันจะต้องมีความสัมพันธ์กัน (ค) ลดขอบระหว่างบล็อกโดยการคำนวณค่าต่ำสุดของค่าความผิดพลาดของพื้นผิวที่ซ้อนทับกัน

การทำรอยต่อให้มีความกลมกลืน (Minimum Error Boundary Cut) เป็นการทำระหว่างบล็อก 2 บล็อกที่ซ้อนทับกันโดยที่ 2 พื้นผิวนี้อาจจะต้องมีความตรงกันมากที่สุด ซึ่งการคำนวณค่าต่ำสุดของค่าความผิดพลาดของพื้นผิวที่ซ้อนทับกันต้องดูส่วนของบล็อกทั้ง B_1 และ B_2 ว่าขอบซ้อนทับกันในแนวตั้ง ดังภาพ 2-15(ค) กับบริเวณที่ซ้อนทับ B_1^{ov} และ B_2^{ov} ตามลำดับ ดังนั้นค่าความผิดพลาดของพื้นผิวคือ $e = (B_1^{ov} - B_2^{ov})$ และสามารถคำนวณค่าความผิดพลาดต่ำสุดของ E ทั้งหมดได้จากสมการ 2-14 และแสดงในภาพที่ 2-19



ภาพที่ 2-19 การทำรอยต่อให้มีความกลมกลืน

$$E_{i,j} = e_{i,j} + \min(E_{i-1,j-1}, E_{i-1,j}, E_{i-1,j+1}) \quad (2-14)$$

จากสมการข้างต้นสามารถนำมาประยุกต์ใช้กับส่วนที่มีการซ้อนทับเป็นแนวนอนได้

2.7 การปกปิดความผิดพลาด (Error Concealment)

การปกปิดความผิดพลาดเป็นการทำเพื่อให้ได้ภาพที่มีความผิดพลาดลดลง โดยใช้การคำนวณค่าพิกเซลของบล็อกข้างเคียงทั้ง 4 ซึ่งมีวิธีการดังนี้คือ

ระบุตำแหน่งของพิกเซลที่อยู่ในบล็อกที่ต้องการปกปิดโดยกำหนดเป็นตำแหน่ง (i, j) และนำค่าทั้งหมดสี่ค่าจากพิกเซลสี่ตำแหน่ง โดยกำหนดค่าคงที่ที่อยู่ระหว่างศูนย์ถึงหนึ่งขึ้นสองค่ากำหนดให้เป็น μ_1 และ μ_2 ใช้เป็นตัวคูณในการคำนวณ และ n คือขนาดของความกว้างบล็อกจะได้ดังนี้

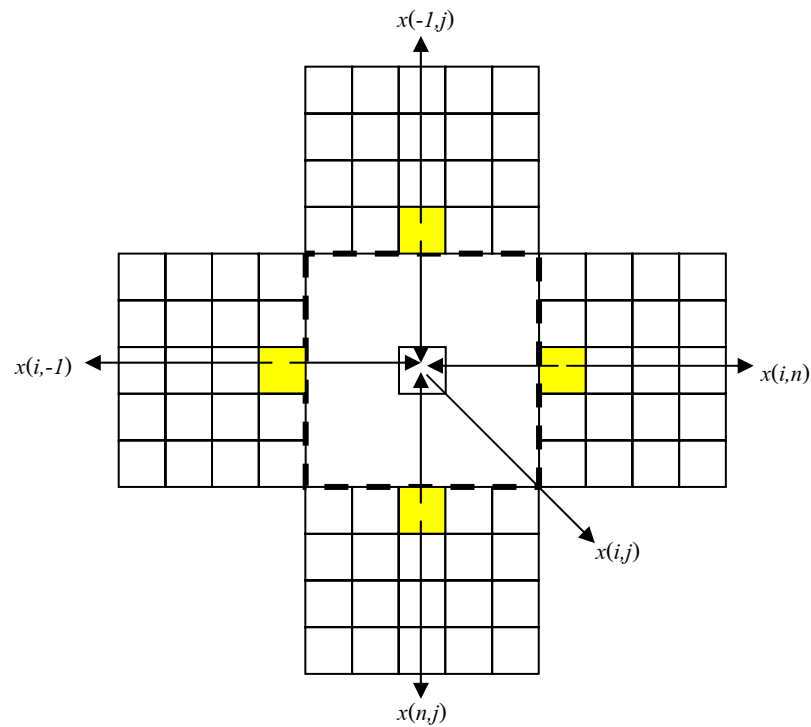
ค่าที่ได้จากค่าของพิกเซลในตำแหน่งที่ $(i+1, j)$ คูณด้วยค่า (μ_2)

ค่าที่ได้จากค่าของพิกเซลตำแหน่ง $(i+n, j)$ คูณด้วยค่า $(1-\mu_2)$

ค่าที่ได้จากค่าของพิกเซลตำแหน่ง $(i, j-1)$ คูณด้วยค่า (μ_1)

ค่าที่ได้จากค่าของพิกเซลตำแหน่ง $(i, j+n)$ คูณด้วยค่า $(1-\mu_1)$

เมื่อทำการคำนวณค่าทั้งสี่ค่าแล้วให้นำมาบวกกันจะได้ค่าของพิกเซลตำแหน่ง (i, j) แล้วทำการหาค่าของพิกเซลทุกตำแหน่งในบล็อกที่ต้องการปกปิด ดังภาพที่ 2-20 และสมการที่ 2-15



ภาพที่ 2-20 การปกปิดความผิดพลาด

$$X_{i,j} = ((\mu_1 X_{i,j-1}) + ((1 - \mu_1) X_{i,j+n}) + (\mu_2 X_{i-1,j}) + ((1 - \mu_2) X_{i+n,j})) \quad (2-15)$$

เมื่อ	$X_{i,j}$	คือ ค่าของพิกเซลที่ตำแหน่ง i, j
	$X_{i,j-1}$	คือ ค่าของพิกเซลที่ตำแหน่ง $i, j-1$
	$X_{i,j+n}$	คือ ค่าของพิกเซลที่ตำแหน่ง $i, j+n$
	$X_{i-1,j}$	คือ ค่าของพิกเซลที่ตำแหน่ง $i-1, j$
	$X_{i+n,j}$	คือ ค่าของพิกเซลที่ตำแหน่ง $i+n, j$
μ_1		คือ ค่าคงที่ที่กำหนดขึ้นสำหรับปรับค่า $X_{i,j-1}$ และ $X_{i,j+n}$
μ_2		คือ ค่าคงที่ที่กำหนดขึ้นสำหรับปรับค่า $X_{i-1,j}$ และ $X_{i+n,j}$

2.8 การวัดประสิทธิภาพ

2.8.1 การวัดค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Square Error หรือ MSE) โดยวิธีการวัดประสิทธิภาพวิธีนี้ จะใช้ค่าความผิดพลาดกำลังเฉลี่ยมาเป็นตัวเปรียบเทียบประสิทธิภาพของการกู้รายละเอียดภาพ หากค่าความผิดพลาดกำลังสองเฉลี่ยมีค่ามากหมายความว่า ภาพที่ได้จากการกู้คืนรายละเอียดมีความผิดเพี้ยนมาก ในทางกลับกันหากค่าความผิดพลาดกำลังสองเฉลี่ยมีค่าน้อยหมายความว่า ภาพที่ได้จากการกู้คืนรายละเอียดภาพมีประสิทธิภาพดี ค่าความผิดพลาดกำลังเฉลี่ยสามารถหาได้จากสมการที่ 2-16

$$MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N [x(i, j) - \hat{x}(i, j)]^2 \quad (2-16)$$

โดยที่ M และ N คือ จำนวนจุดภาพในแนวนอนและแนวตั้งของภาพ

$x(i, j)$ และ $\hat{x}(i, j)$ คือ จุดภาพที่ตำแหน่ง (i, j) ของภาพต้นฉบับและภาพที่ได้จากการกู้คืนรายละเอียดภาพตามลำดับ

2.8.2 การหาค่าสัดส่วนสัญญาณต่อสัญญาณรบกวนสูงสุด (Peak Signal To Noise Ratio หรือ PSNR) ค่าสัดส่วนสัญญาณต่อสัญญาณรบกวนสูงสุดมีหน่วยเป็นเดซิเบล (dB) ซึ่งค่าสัดส่วนสัญญาณต่อสัญญาณรบกวนสูงสุดสามารถหาได้จากสมการที่ 2-17

$$PSNR = 10 \log_{10} \frac{P^2}{MSE} = 10 \log_{10} \frac{(2^n - 1)^2}{MSE} \quad (2-17)$$

โดยที่ P คือ ค่าสูงสุดของขอบเขตของข้อมูล

MSE คือ ค่าความผิดพลาดกำลังสองเฉลี่ยหาได้จากสมการที่ 2-16

n คือ จำนวนบิตที่ใช้แทนจุดภาพ

2.9 งานวิจัยที่เกี่ยวข้อง

Wu, Chui และ Chen [3] ทำการแยกตัวหนังสือกับภาพพื้นหลังออกจากกัน โดยใช้การแบ่งกลุ่มข้อมูลด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน เพื่อให้ได้ส่วนที่มีตัวอักษรอยู่ จากนั้นใช้วิธีอะแดปทีฟเทรซโฮลดิ้ง ในการแยกส่วนที่เป็นตัวหนังสือออกมาจากพื้นหลัง

Efros และ Leung [5] เสนอวิธีการการสังเคราะห์พื้นผิว (Texture Synthesis) โดยการสร้างภาพใหม่รอบๆ จุดเริ่มต้น (Initial Seed) โดยมีพื้นฐานมาจาก Markov Random Field ในการ

คาดการณ์ล่วงหน้าและกำหนดเงื่อนไขการสังเคราะห์เพื่อนบ้าน (Neighbors Synthesis) ทั้งหมดจากภาพตัวอย่างและหาเพื่อนบ้านที่คล้ายกัน งานวิจัยชิ้นนี้ให้ผลลัพธ์ที่ดีสำหรับพื้นผิวแบบกว้าง (Wide Range) ซึ่งตัวแปรที่มีผลต่อการทำคือ ขนาดของหน้าต่าง (Window) ซึ่งมันเกิดจากการสุ่มตัวอย่างที่คล้ายกัน

Efros และ Freeman [6] เสนอการสังเคราะห์พื้นผิวโดยใช้บล็อกที่กำหนดขึ้นจากภาพนำเข้าในการสุ่มเลือกพื้นผิวและนำมาวางเรียงต่อกันซึ่งขนาดของบล็อกที่กำหนดนี้จะต้องมีขนาดใหญ่พอที่จะจับโครงสร้างหลักของภาพได้ จากนั้นทำการเปรียบเทียบขอบของบล็อกที่อยู่ใกล้กันว่ามีส่วนใดของบริเวณเพื่อนบ้านที่มีการซ้อนกันบ้างเพื่อเป็นการทำให้ภาพมีความกลมกลืนกัน แต่ภาพก็ยังคงมีส่วนที่เป็นขอบของบล็อกอยู่ดี จึงต้องทำการหาค่าความผิดพลาดต่ำสุดของพื้นผิวเพื่อทำให้ภาพที่ได้มีความกลมกลืนกันมากที่สุด แต่เมื่อนำวิธีการนี้มาใช้กับภาพที่มีรายละเอียดภาพหลายอย่างอยู่ด้วยกัน ภาพที่ได้ออกมานั้นก็ยังไม่กลมกลืนให้เห็นบ้างในส่วนที่เป็นรอยต่อ

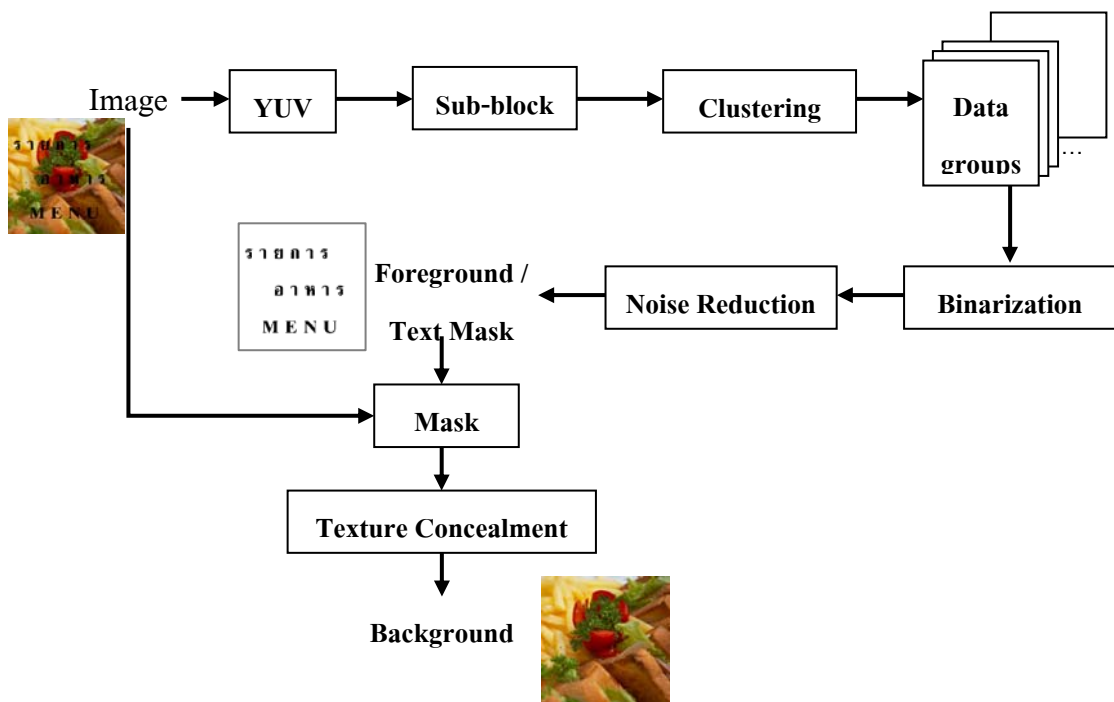
ในงานวิจัยนี้ ได้ทำการศึกษาการหาส่วนที่เป็นตัวอักษร โดยมีขั้นตอนย่อย 4 ขั้นตอนคือการแบ่งภาพเป็นภาพย่อย การแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว-ดำ และทำการลดสิ่งรบกวน โดยทำการเติมเต็มส่วนที่เสียหายโดยใช้การสังเคราะห์พื้นผิว ในการสังเคราะห์รายละเอียดของภาพและทำให้ภาพสมบูรณ์ยิ่งขึ้นด้วยการใช้การปกปิดความผิดพลาดในการทำภาพให้มีความกลมกลืนมากยิ่งขึ้น

บทที่ 3

ขั้นตอนการวิจัย

ในบทนี้กล่าวถึงโครงสร้างของกระบวนการกู้คืนภาพจากการถูกตัวอักษรซ้อนทับและขั้นตอนการวิจัย

จากปัญหาการถูกตัวอักษรซ้อนทับบนภาพสามารถแบ่งกระบวนการกู้คืนภาพออกเป็นสองกระบวนการหลักคือ การหาส่วนที่เป็นตัวอักษร (Text Detection) และการปกปิดพื้นผิว (Texture Concealment) ในส่วนของการหาส่วนที่เป็นตัวอักษรผู้วิจัยได้แบ่งขั้นตอนการทำให้เป็น 4 ขั้นตอนคือ การแบ่งภาพเป็นภาพย่อย การแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว-ดำ และการลดสิ่งรบกวน เพื่อให้ได้ส่วนที่เป็นส่วนหน้าหรือ Text Mask ออกมาให้ชัดเจนมากที่สุด เพื่อใช้ในการกำหนดตำแหน่งที่ต้องทำการกู้คืนส่วนที่สูญเสียไปของภาพ ในส่วนของการปกปิดพื้นผิวนั้นทำเพื่อการกู้คืนภาพในส่วนที่เกิดความเสียหายโดยมีขั้นตอนย่อย 2 ขั้นตอนคือ การสังเคราะห์พื้นผิวแบบปะติดปะต่อและการปกปิดความผิดพลาด ซึ่งขั้นตอนในการวิจัยแสดงดังภาพที่ 3-1



ภาพที่ 3-1 กระบวนการกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ

3.1 ขั้นตอนการเตรียมภาพ

ขั้นตอนการเตรียมภาพประกอบไปด้วย 2 ขั้นตอนดังต่อไปนี้

3.1.1 การแปลงภาพนำเข้าจากระบบสี RGB ไปเป็นภาพระบบสี YUV เพื่อที่จะนำเพน Y ซึ่งเป็นภาพโทนสีเทา (Grayscale Image) มาใช้เพื่อให้ได้ส่วนหน้าออกมา

3.1.2 แบ่งภาพเป็นภาพย่อย (Sub-block) $X_n(i,j)$ โดยแบ่งออกเป็น 3 ขนาดด้วยกันคือ 32×32 , 64×64 และ 128×128 พิกเซล

3.2 ขั้นตอนการแบ่งกลุ่ม

ขั้นตอนการแบ่งกลุ่มเป็นขั้นตอนการจำแนกกลุ่มของข้อมูลที่มีความคล้ายกันให้มาอยู่ในกลุ่มเดียวกัน ซึ่งในขั้นตอนนี้ต้องการส่วนที่เป็นตัวอักษรหรือส่วนหน้า

ในงานวิจัยนี้ได้ทำการทดลองแบ่งกลุ่ม 3 วิธีคือ วิธีที่ 1 แบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน วิธีที่ 2 แบ่งกลุ่มด้วยเคมีน และวิธีที่ 3 แบ่งกลุ่มด้วยฟิชเชอร์มิน เนื่องจากกลุ่มข้อมูลมีขนาดค่อนข้างใหญ่ ซึ่งวิธีการเหล่านี้เหมาะกับกลุ่มข้อมูลประเภทนี้

3.2.1 การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน

ขั้นตอนที่ 1 หาค่าเฉลี่ย ของภาพโทนสีเทา m และ ค่าเบี่ยงเบนมาตรฐาน ของแต่ละส่วน σ ซึ่งคำนวณได้จากสมการดังนี้

$$m_n = \frac{\sum_{i,j} x_n(i,j)}{K \times L} \quad (3-1)$$

$$\sigma_n = \sqrt{\frac{\sum_{i,j} [X_n(i,j) - m_n]^2}{K \times L}} \quad (3-2)$$

โดยที่ K คือ ความกว้างของภาพย่อย

L คือ ความยาวของภาพย่อย

ขั้นตอนที่ 2 แยก $x_n(i,j)$ ออกเป็น 2 กลุ่มโดยใช้ค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน ดังนี้

$$C'_{n1} \text{ และ } C'_{n2} \text{ โดย } C'_{n1} = m_n + 0.5 \times \sigma_n \quad \text{และ} \\ C'_{n2} = m_n - 0.5 \times \sigma_n \quad (3-3)$$

โดยที่ C'_{n1} คือ จุดกลางของกลุ่มข้อมูลที่ 1

C'_{n2} คือ จุดกลางของกลุ่มข้อมูลที่ 2

ขั้นตอนที่ 3 คำนวณระยะทางตามแบบแมนฮัตตันจากแต่ละจุดภาพ $x_n(i, j)$ จนถึง C'_{n1} และ C'_{n2} โดยใช้

$$D'_{ij,1} = |x_n(i, j) - C'_{n1}| \quad \text{และ} \\ D'_{ij,2} = |x_n(i, j) - C'_{n2}| \quad (3-4)$$

โดยที่ $D'_{ij,1}$ คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ 1

$D'_{ij,2}$ คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ 2

จากนั้นแบ่ง $x_n(i, j)$ ออกเป็น 2 กลุ่ม $\eta_k (k = 1, 2)$

$$\eta_1 : \{x_n(i, j) | D'_{ij,1} \leq D'_{ij,2}\} \quad \text{และ} \\ \eta_2 : \{x_n(i, j) | D'_{ij,1} < D'_{ij,2}\} \quad (3-5)$$

ขั้นตอนที่ 4 คำนวณค่าเฉลี่ย m_{nk} และค่าเบี่ยงเบนมาตรฐาน σ_{nk} ของทั้ง 2 กลุ่ม $\eta_k (k = 1, 2)$ โดยใช้สมการที่ 3-1 และ 3-2 ตามลำดับ ซึ่งจะทำให้สามารถแยกกลุ่มที่ 3 ออกมาได้จากเงื่อนไขดังต่อไปนี้

ถ้า $\sigma_{n1} > \sigma_{n2}$ แล้ว $C_{n3} = m_n - 0.5 \times \sigma_n$ และได้ค่า C_{n1} และ C_{n2} ค่าใหม่คือ

$$C_{n1} = m_{n1} + 0.5 \times \sigma_{n1} \quad \text{และ} \\ C_{n2} = m_{n1} - 0.5 \times \sigma_{n1} \quad (3-6)$$

นอกจากนั้น ถ้า $\sigma_{n1} < \sigma_{n2}$ แล้ว $C_{n3} = m_n + 0.5 \times \sigma_n$ และได้ค่า C_{n1} และ C_{n2} ค่าใหม่ คือ

$$C_{n1} = m_{n2} + 0.5 \times \sigma_{n2} \quad \text{และ}$$

$$C_{n2} = m_{n2} - 0.5 \times \sigma_{n2} \quad (3-7)$$

โดยที่ C_{n1} คือ จุดกลางของกลุ่มข้อมูลที่ 1
 C_{n2} คือ จุดกลางของกลุ่มข้อมูลที่ 2
 C_{n3} คือ จุดกลางของกลุ่มข้อมูลที่ 3

ขั้นตอนที่ 5 หลังจากผ่านขั้นตอนที่ 3.1.2.4 แล้วได้ออกมา 3 กลุ่ม $C_{nk} (k = 1, 2, 3)$ และสามารถแบ่ง $x_n(i, j)$ ออกมาเป็น 3 กลุ่มเช่นกัน $\psi_k (k = 1, 2, 3)$

$$\begin{aligned} \psi_1 : \{x_n(i, j) | D_{ij,1} < D_{ij,2} \text{ และ } D_{ij,1} < D_{ij,3}\} \\ \psi_2 : \{x_n(i, j) | D_{ij,2} < D_{ij,1} \text{ และ } D_{ij,2} < D_{ij,3}\} \\ \psi_3 : \{x_n(i, j) | D_{ij,3} < D_{ij,1} \text{ และ } D_{ij,3} < D_{ij,2}\} \end{aligned} \quad (3-8)$$

$$\begin{aligned} \text{โดยที่} \quad D_{ij,1} &= |x_n(i, j) - C_{n1}| \\ D_{ij,2} &= |x_n(i, j) - C_{n2}| \\ D_{ij,3} &= |x_n(i, j) - C_{n3}| \end{aligned} \quad (3-9)$$

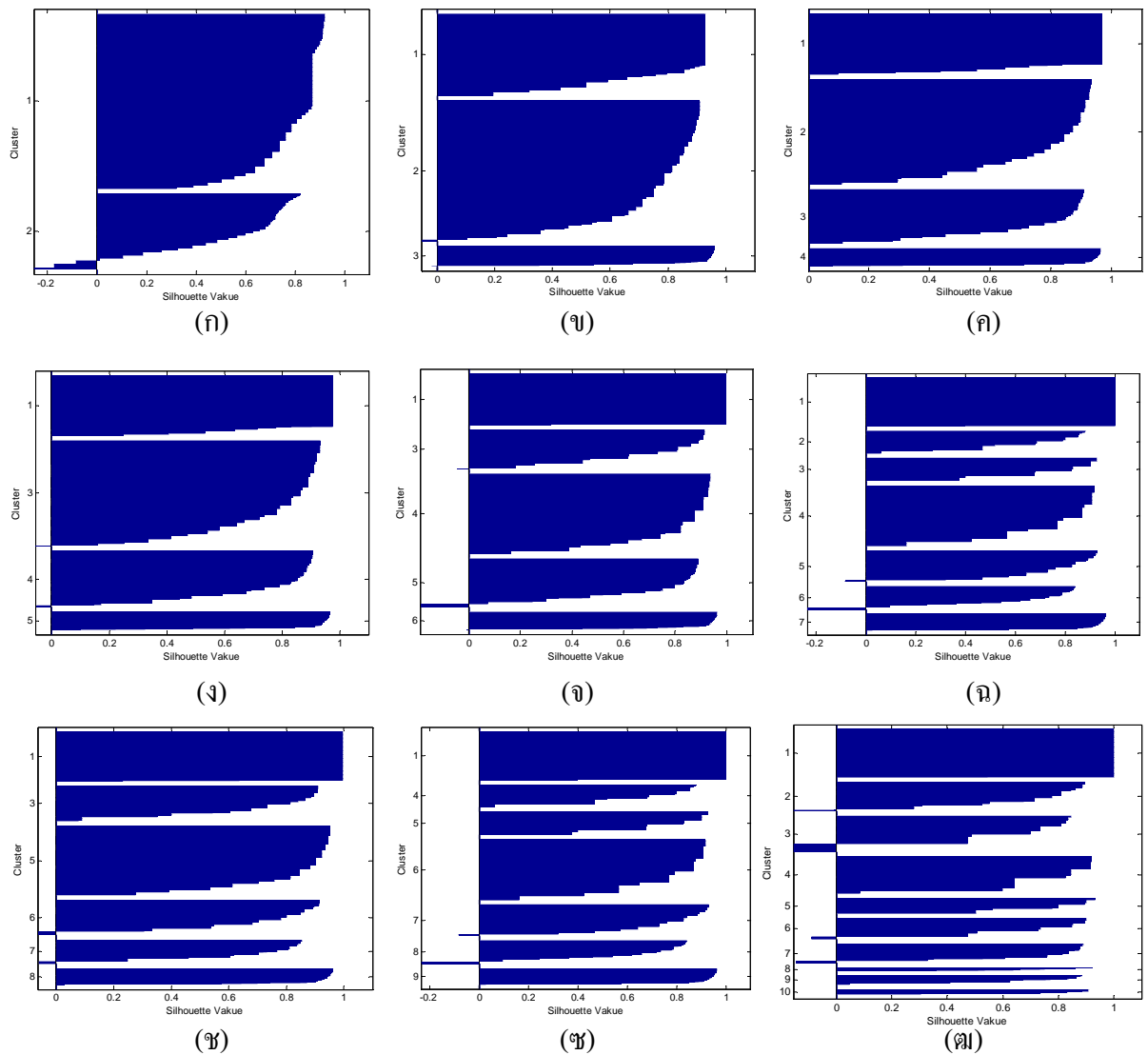
เมื่อ $D_{ij,1}$ คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ 1
 $D_{ij,2}$ คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ 2
 $D_{ij,3}$ คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ 3

ทำซ้ำตั้งแต่ขั้นตอนที่ 1 ถึงขั้นตอนที่ 5 ให้ครบทุกภาพย่อย

ในวิธีการนี้เมื่อแบ่งกลุ่มแล้วจะได้ภาพออกมา 3 เฟรม คือ มีด กลาง และสว่าง ซึ่งตัวอักษรจะอยู่ในส่วนของเฟรมมีด

3.2.2 แบ่งกลุ่มด้วยเคมิน

ขั้นตอนที่ 1 แบ่งกลุ่มโดยกำหนดการแบ่งกลุ่มข้อมูลทั้งหมด 10 กลุ่ม เนื่องจากการแบ่งกลุ่ม 10 กลุ่มให้ผลในการแบ่งกลุ่มตัวอักษรได้ชัดเจนที่สุด ซึ่งผลที่ได้จากการแบ่งกลุ่มที่ต่างกันแสดงในภาพที่ 3-2



ภาพที่ 3-2 ผลการเปรียบเทียบการแบ่งกลุ่มข้อมูลด้วยเคมีน (ก) แบ่งกลุ่มข้อมูล 2 กลุ่ม
 (ข) แบ่งกลุ่มข้อมูล 3 กลุ่ม (ค) แบ่งกลุ่มข้อมูล 4 กลุ่ม (ง) แบ่งกลุ่มข้อมูล 5 กลุ่ม
 (จ) แบ่งกลุ่มข้อมูล 6 กลุ่ม (ฉ) แบ่งกลุ่มข้อมูล 7 กลุ่ม (ช) แบ่งกลุ่มข้อมูล 8 กลุ่ม
 (ซ) แบ่งกลุ่มข้อมูล 9 กลุ่ม (ต) แบ่งกลุ่มข้อมูล 10 กลุ่ม

ขั้นตอนที่ 2 กำหนดค่ากลางเริ่มต้นของแต่ละกลุ่มด้วยการสุ่มค่าเริ่มต้นจำนวน 10 ค่า
 ขั้นตอนที่ 3 นำข้อมูลแต่ละตัวมาทำการหาระยะห่างจากจุดกลางของกลุ่มโดยใช้การคำนวณ
 ระยะทางแบบสแควยูคลิดี้น ตามสมการที่ 3-10

$$D = \|x_i^{(n)} - C_n\|^2 \quad (3-10)$$

โดยที่ D คือ ระยะทางจากจุดข้อมูลถึงจุดกลางของกลุ่มที่ n

C_n คือ จุดกลางของแต่ละกลุ่ม

n คือ จำนวนกลุ่มข้อมูล

ขั้นตอนที่ 4 คำนวณค่ากลางของแต่ละกลุ่มใหม่ตามสมการที่ 3-11

$$C_{new} = \frac{X + C_{old}(X)}{2} \quad (3-11)$$

โดยที่ C คือ ค่ากลางของแต่ละกลุ่มที่อยู่ใกล้ X มากที่สุด

X คือ เวกเตอร์ข้อมูล

ทำซ้ำขั้นตอนที่ 3 และขั้นตอนที่ 4 จนกว่าค่ากลางของแต่ละกลุ่มไม่มีการเปลี่ยนแปลง

ขั้นตอนที่ 5 คำนวณหาระยะห่างที่น้อยที่สุดระหว่างข้อมูลกับค่ากลางของแต่ละกลุ่มจากสมการที่ 3-12

$$J = \sum_{n=1}^K \sum_{i,j=1}^N \|x_i^{(n)} - C_n\|^2 \quad (3-12)$$

โดยที่ J คือ ระยะห่างที่น้อยที่สุดระหว่างข้อมูลกับค่ากลางของกลุ่ม

N คือ จำนวนข้อมูล

K คือ จำนวนกลุ่มข้อมูล

3.2.3 แบ่งกลุ่มด้วยฟuzzyซิมิน

ขั้นตอนที่ 1 แบ่งกลุ่มข้อมูลทั้งหมดเป็น 10 กลุ่ม

ขั้นตอนที่ 2 กำหนดค่ากลางของแต่ละกลุ่มโดยการสุ่มค่าเริ่มต้น

ขั้นตอนที่ 3 คำนวณค่าความเป็นสมาชิกของข้อมูลจากค่ากลางที่ได้ในขั้นตอนข้างบน ดังสมการที่ 3-13

$$u_{in} = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_i - C_n\|}{\|x_i - C_k\|} \right)^{\frac{2}{m-1}}} \quad (3-13)$$

โดยที่ u_{in} คือ ค่าความเป็นสมาชิกของ x_i ในกลุ่มที่ n

k คือ จำนวนของการทำซ้ำ

m คือ ฟังก์ชันพารามิเตอร์ที่มีค่ามากกว่า 1

ขั้นตอนที่ 4 คำนวณค่ากลางของแต่ละกลุ่มใหม่ดังสมการที่ 3-14

$$C_n = \frac{\sum_{i=1}^N u_{in}^m \cdot x_i}{\sum_{i=1}^N u_{in}^m} \quad (3-14)$$

โดยที่ N คือ จำนวนข้อมูล

ทำซ้ำขั้นตอนที่ 3 และขั้นตอนที่ 4 จนกว่าค่ากลางของแต่ละกลุ่มไม่มีการเปลี่ยนแปลงหรือครบตามจำนวนการทำซ้ำที่กำหนดไว้ ซึ่งในงานวิจัยนี้ได้กำหนดค่าการทำซ้ำสูงสุดไว้ที่ 100

ขั้นตอนที่ 5 คำนวณค่าความเป็นสมาชิกและฟังก์ชันเป้าหมาย (Objective Function) จากสมการที่ 3-15

$$J_m = \sum_{i=1}^N \sum_{n=1}^c u_{in}^m \|x_i - C_n\|^2 \quad (3-15)$$

โดยที่ J_m คือ ฟังก์ชันเป้าหมาย

c คือ จำนวนกลุ่มข้อมูล

ซึ่งผลที่ได้จากการทดลองจะแสดงในตารางที่ 4-1 ในบทที่ 4

3.3 ขั้นตอนการทำภาพเป็นขาว - ดำ (Binarization)

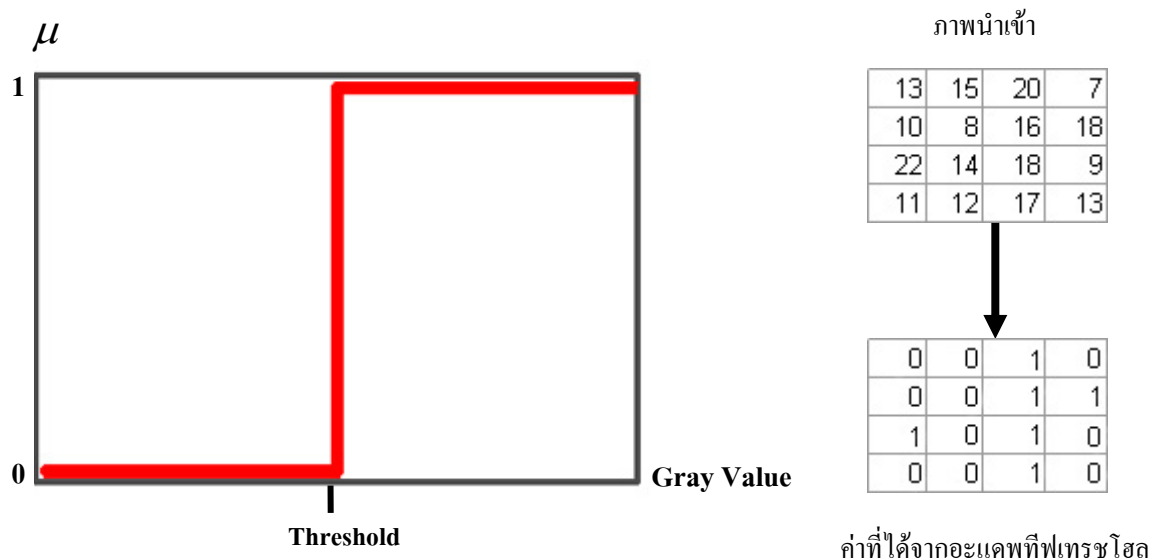
ในขั้นตอนของการทำภาพเป็นขาว - ดำนั้นเป็นการทำเพื่อแยกตัวอักษรออกจากพื้นหลังหลังจากที่ได้ผ่านขั้นตอนของการแบ่งกลุ่มแล้ว โดยได้ใช้ค่าเทรชโฮลเป็นค่ากลางในการตัดสินใจ โดยค่าเทรชโฮลที่ใช้นี้จะคำนวณจากแต่ละภาพย่อย ภาพที่ได้หลังจากผ่านขั้นตอนนี้แล้วจะได้ภาพ

ที่เป็นขาว – ดำหรือภาพไบนารี ซึ่งในงานวิจัยนี้ได้ทำการทดลองการทำภาพเป็นขาว – ดำ 2 วิธี คือ วิธีที่ 1 การทำภาพเป็นขาว – ดำใช้อะแดปทีฟเทรชโฮล (Adaptive Binarization) และวิธีที่ 2 การทำภาพเป็นขาว – ดำโดยใช้ฟัซซี่ (Fuzzy Binarization)

3.3.1 การทำภาพขาว – ดำโดยใช้อะแดปทีฟเทรชโฮล มีขั้นตอนดังนี้

ขั้นตอนที่ 1 นำภาพย่อยมาหาค่าเทรชโฮลของแต่ละภาพย่อย

ขั้นตอนที่ 2 นำค่าเทรชโฮลที่ได้ในขั้นตอนข้างบนมาทำการแบ่งค่าเป็น 2 กลุ่ม โดยที่ข้อมูลที่มีค่าน้อยกว่าหรือเท่ากับค่าเทรชโฮลจะถูกกำหนดให้มีค่าเป็น “0” (สีดำ) และข้อมูลที่มีค่ามากกว่าค่าเทรชโฮลจะถูกกำหนดให้มีค่าเป็น “1” (สีขาว) ดังแสดงในภาพที่ 3-3

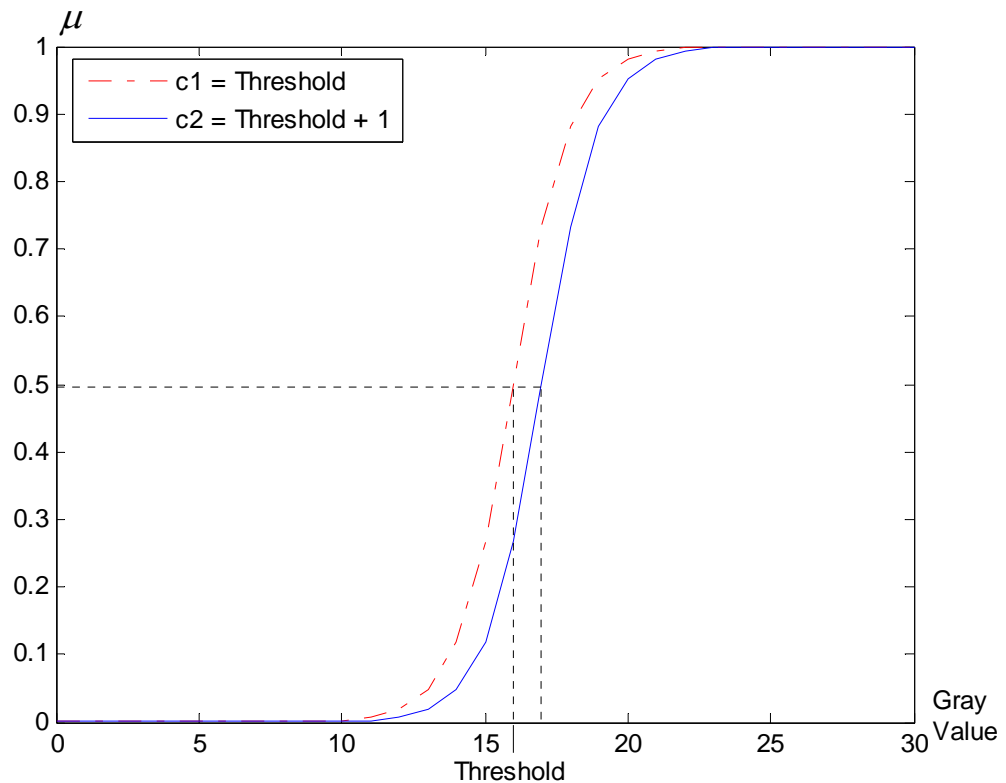


ภาพที่ 3-3 การทำภาพเป็นขาว – ดำโดยใช้อะแดปทีฟเทรชโฮล

3.3.2 การทำภาพเป็นขาว – ดำโดยใช้ฟัซซี่

ขั้นตอนที่ 1 หาค่าเทรชโฮลของแต่ละภาพย่อย

ขั้นตอนที่ 2 นำค่าเทรชโฮลที่ได้ในขั้นตอนข้างบนมาทำการแบ่งเป็น 2 กลุ่ม โดยใช้ฟังก์ชันรูปโค้งกลับดังสมการที่ 2-4 ในบทที่ 2 ค่าพารามิเตอร์ที่ทำการกำหนดคือค่า a และค่า c ในงานวิจัยนี้ได้กำหนดค่าทั้งสองดังนี้ $a = 1$ และ $c = threshold + 1$ ดังภาพที่ 3-4 ซึ่งค่าที่ได้ออกมานั้นจะไม่ใช่ค่า 0 และ 1 เหมือนกับการใช้อะแดปทีฟเทรชโฮล แต่จะได้ออกมาเป็นค่าทศนิยมระหว่าง 0 ถึง 1 ดังภาพที่ 3-5



ภาพที่ 3-4 การทำภาพขาว – ดำโดยใช้ฟัซซี่

0.018	0.1192	0.9526	0
0.0009	0.0001	0.2689	0.7311
0.9933	0.0474	0.7311	0.0003
0.0025	0.0067	0.5	0.018

→

0	0	1	0
0	0	0	1
1	0	1	0
0	0	1	0

ภาพที่ 3-5 ค่าที่ได้จากการใช้ฟัซซี่โดยมีภาพนำเข้าค่าเดียวกับการใช้อะแพททิฟเทรซโซล

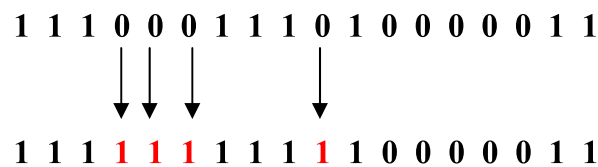
ซึ่งผลที่ได้จากการทดลองจะแสดงในตารางที่ 4-2 ในบทที่ 4

3.4 ขั้นตอนการลดสิ่งรบกวน (Noise Reduction)

เมื่อขั้นตอนการทำภาพเป็นขาว – ดำ แล้วก็จะยังมีส่วนที่เป็นพื้นหลังบ้างบางส่วนติดมาด้วย ดังนั้นเพื่อให้ส่วนหน้ามีพื้นหลังติดมาน้อยที่สุด จึงต้องทำการลดสิ่งรบกวน ซึ่งในงานวิจัยนี้ได้ทำการทดสอบการลดสิ่งรบกวน 2 วิธีคือ วิธีที่ 1 ใช้วิธีการควบคุมความยาวในการวิ่ง (Constrained Run-Length Algorithm : CRLA) [3] และวิธีที่ 2 ใช้วิธีการหาขอบวัตถุและวัดขนาดของวัตถุ

3.4.1 การควบคุมความยาวในการวิ่ง

3.4.1.1 กำหนดค่าความยาวของ 0 ที่อยู่ระหว่าง 1 โดยในงานวิจัยนี้ได้กำหนดให้มีค่าความยาวเท่ากับ 3 ดังภาพที่ 3-6

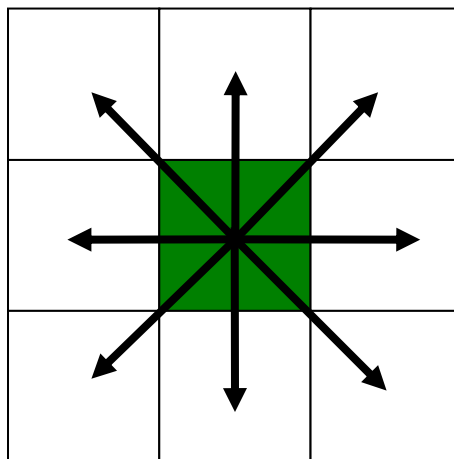


ภาพที่ 3-6 การทำงานของ CRLA

3.4.1.2 ทำตามขั้นตอนข้างบนทั้งแนวดิ่งและแนวนอน และนำค่าที่ได้ทั้งแนวดิ่งและแนวนอนมาทำการ AND กัน

3.4.2 การหาขอบของวัตถุและวัดขนาดของวัตถุ

3.4.2.1 หาจุดเริ่มของสิ่งที่คาดว่าจะป็นวัตถุ เมื่อได้จุดเริ่มแล้วก็จะทำการดูจุดเพื่อนบ้านทั้ง 8 ทิศทางว่ามีจุดใดที่เป็นพวกเดียวกันบ้าง (หรือมีค่าเป็น 0 เหมือนกัน) ดังภาพที่ 3-7 และจะเข้าไปเรื่อยๆ จนกว่าจะวนกลับมาจุดเริ่มต้น ดังภาพที่ 3-8

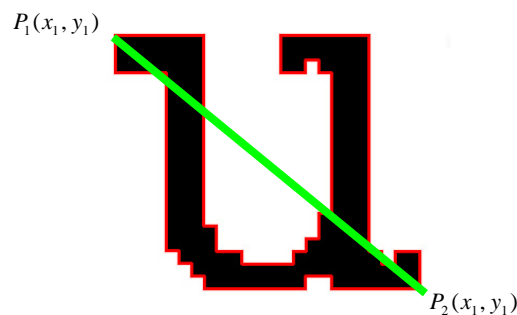


ภาพที่ 3-7 การเริ่มหาขอบของวัตถุ โดยที่จุดสี่เหลี่ยมตรงกลางเป็นจุดเริ่มต้นในการหาขอบ



ภาพที่ 3-8 การหาขอบของวัตถุโดยที่เส้นสีแดงคือขอบของวัตถุที่หาได้

3.4.2.2 เมื่อได้ขอบของวัตถุมาแล้ว ก็จะทำกรหาขนาดของวัตถุที่หาได้โดยการวัดจากจุด $P_1(x_1, y_1)$ ไปถึง $P_2(x_2, y_2)$ ในแนวทแยงดังภาพที่ 3-9 และใช้การวัดระยะทางแบบยูคลิเดียน ดังสมการที่ 3-16 ซึ่งในงานวิจัยนี้ได้กำหนดให้ค่า $D \leq 5$ เป็นสิ่งรบกวนไม่ใช่ตัวอักษร



ภาพที่ 3-9 การวัดขนาดของวัตถุโดยวัดจากจุด $P_1(x_1, y_1)$ ไปถึง $P_2(x_2, y_2)$

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \quad (3-16)$$

โดยที่ D คือระยะทางจากจุด $P_1(x_1, y_1)$ ไปถึง $P_2(x_2, y_2)$

เมื่อจบขั้นตอนของการลดสิ่งรบกวนแล้ว เราจะได้ส่วนหน้าที่มีความชัดเจนของตัวอักษรอยู่ ซึ่งเราจะนำส่วนนี้ไปเป็นตัวแทนในการหาส่วนที่ต้องทำการกู้คืนรายละเอียดภาพในส่วนของพื้นหลัง

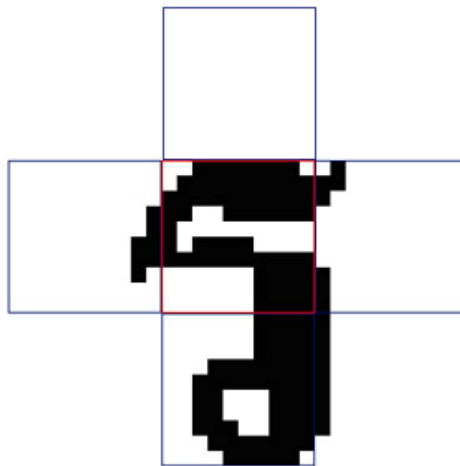
ซึ่งผลที่ได้จากการทดลองจะแสดงในภาพที่ 4-1 ในบทที่ 4

3.5 ขั้นตอนการปกปิดพื้นผิว (Texture Concealment)

เป็นขั้นตอนของการกู้คืนส่วนที่สูญเสียรายละเอียดไปจากการถูกตัวอักษรซ้อนทับ โดยใช้ส่วนหน้าที่ได้หลังจากขั้นตอนการลดจำนวนสิ่งรบกวนมาเป็นตัวทาบในการกำหนดพื้นที่ที่ต้องทำการกู้คืนรายละเอียดของภาพซึ่งขั้นตอนในการกู้คืนรายละเอียดของภาพจะประกอบไปด้วย 2 ขั้นตอนคือ การสังเคราะห์พื้นผิวและการปกปิดความผิดพลาด

3.5.1 การสังเคราะห์พื้นผิว

3.5.1.1 เลือกส่วนที่จะมาทำเป็นภาพตัวอย่างที่จะนำไปสังเคราะห์ โดยทำการแบ่งภาพส่วนหน้าเป็นภาพย่อยขนาด 10×10 พิกเซล และเลือกภาพย่อยใกล้เคียงที่มีส่วนของตัวอักษรอยู่ในภาพย่อยน้อยที่สุด ดังภาพที่ 3-10 โดยคูเพื่อนบ้านสี่ทิศทางคือ ภาพย่อยด้านบน ภาพย่อยด้านซ้าย ภาพย่อยด้านล่าง และภาพย่อยด้านขวา



ภาพที่ 3-10 การหาส่วนที่จะนำมาเป็นภาพตัวอย่างในการสังเคราะห์

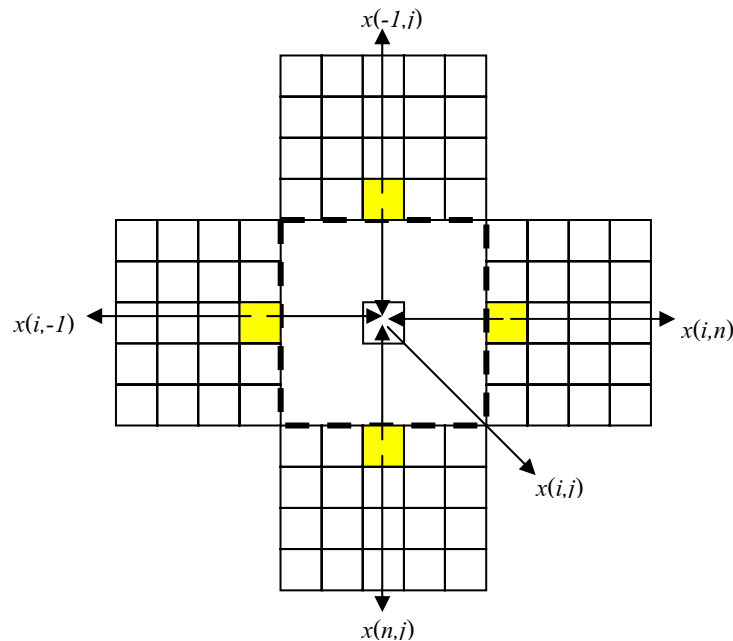
3.5.1.2 นำภาพตัวอย่างที่ได้จากขั้นตอนข้างบนมาระบุตำแหน่งที่ภาพนำเข้าเพื่อที่จะใช้ส่วนนั้นในการสังเคราะห์พื้นผิวในภาพย่อยปัจจุบัน โดยขั้นตอนในการสังเคราะห์พื้นผิวนี้ได้ใช้วิธีการปะติดปะต่อภาพเข้ามาช่วยในการกู้คืนรายละเอียดภาพ

3.5.1.3 เมื่อทำการต่อภาพแล้วมันจะเกิดรอยต่อระหว่างภาพย่อยที่ได้ทำการต่อเอาไว้จึงต้องทำให้รอยต่อนั้นมีความกลมกลืนกันโดยใช้สมการที่ 2-14 ในบทที่ 2 มาทำการปรับความกลมกลืนของรอยต่อทั้งแนวตั้งและแนวนอน

3.5.2 การปกปิดความผิดพลาด

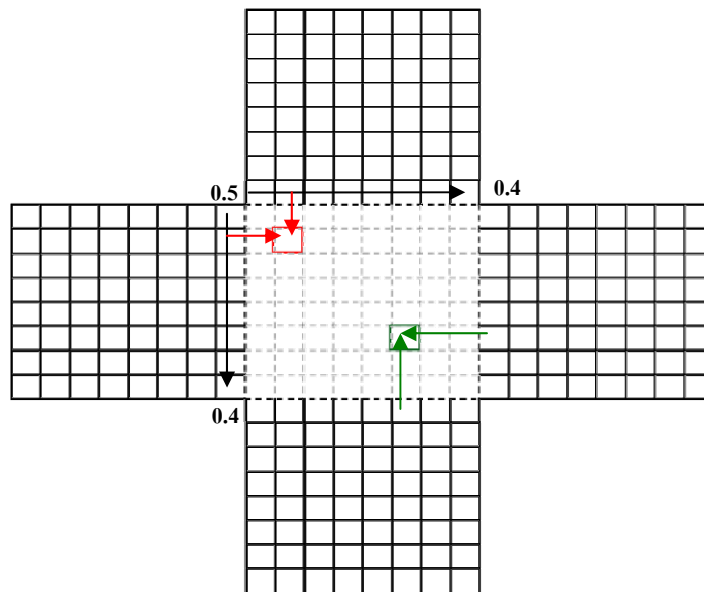
ขั้นตอนนี้เป็นการทำงานให้ภาพที่ได้จากการสังเคราะห์พื้นผิวแล้วมาทำให้ภาพมีความกลมกลืนและสมบูรณ์กันมากขึ้น โดยการทำการปกปิดความผิดพลาดนี้ได้ทำการทดลอง 2 วิธีคือ การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปร (Static Concealment) และการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้ (Adaptive Concealment)

3.5.2.1 การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปร จะเป็นการคำนวณค่าทั้งสี่แล้วนำมาบวกกันจะได้ค่าของพิกเซลตำแหน่ง (i, j) แล้วทำการหาค่าของพิกเซลทุกตำแหน่งในบล็อกที่ต้องการปกปิด ดังภาพที่ 3-11 โดยในงานวิจัยนี้ทำการกำหนดค่า μ_1 และ μ_2 เท่ากับ 0.5 และใช้สมการที่ 2-15 ในบทที่ 2



ภาพที่ 3-11 การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปร

3.5.2.2 การปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้ จะเป็นการคำนวณค่าทั้งสี่แล้วนำมาบวกกันจะได้ค่าของพิกเซลตำแหน่ง (i, j) แต่ค่าทั้งสี่ที่นำมาบวกนั้นจะดูด้วยว่าตำแหน่ง (i, j) อยู่ใกล้กับตำแหน่งใดก็จะกำหนดให้ค่า (i, j) มีน้ำหนักของตำแหน่งที่อยู่ใกล้กว่านั้นมากกว่าค่าที่ตำแหน่งอยู่ไกลกว่า ดังภาพที่ 3-12 และในงานวิจัยนี้ได้กำหนดช่วงของ μ_1 และ μ_2 เท่ากับ $0.4 \sim 0.5$ และใช้สมการที่ 2-15 ในบทที่ 2



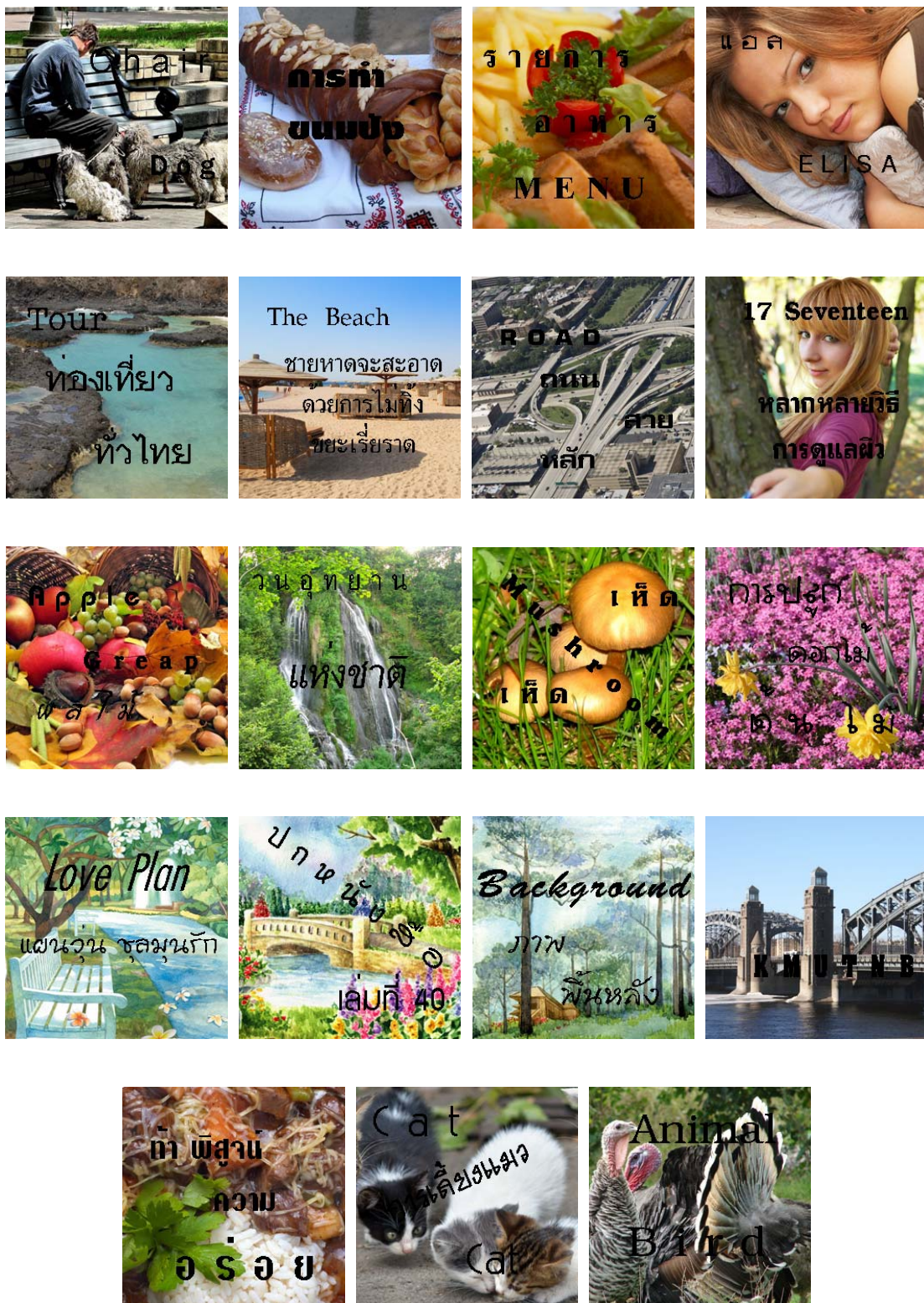
ภาพที่ 3-12 การปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้

ซึ่งผลการทดลองที่ได้จะแสดงในตารางที่ 4-3 ถึงตารางที่ 4-5 และภาพที่ 4-2 ถึงภาพที่ 4-7 ในบทที่ 4

3.6 การเตรียมข้อมูลการทดลอง

ตัวอย่างภาพที่นำมาทำการทดลองนี้เป็นสีในระบบจำลองสี RGB และเป็นไฟล์ภาพที่ไม่ผ่านการบีบอัด เป็นแฟ้มข้อมูลที่มีนามสกุล BMP มีความละเอียดของข้อมูลภาพ 256×256 พิกเซล จำนวน 20 ภาพ

ภาพที่นำมาทดสอบตัวอักษรที่อยู่บนภาพจะต้องเป็นตัวอักษรสีดำเท่านั้น ขนาดของตัวอักษรที่ใช้ในการทดลองมีด้วยกัน 6 ขนาดคือ 14, 16, 18, 20, 24 และ 30 points ซึ่งขนาด 14 points มีรูปแบบของตัวอักษรคือ Arial, DS Sathorn Medium, JS Karabow, JS Chanchai, JS Giat, JS Laongdao และ JS Tina ตัวอักษรขนาด 16 points มีรูปแบบของตัวอักษรคือ JS Angsumalin, JS Jetarin(Italic), JS Chaimongkol และ JS Jukaphan ตัวอักษรขนาด 18 points มีรูปแบบของตัวอักษรคือ DB Erawan, JS Chodok, JS Duangta, JS Thanaporn, JS Toomtam, JS Sirium, JS Oobboon และ JS Neeno ตัวอักษรขนาด 20 points มีรูปแบบของตัวอักษรคือ JS Giat ตัวอักษรขนาด 24 points มีรูปแบบของตัวอักษรคือ JS Jukaphan, JS Macha, JS Macha, JS Sirium, JS Oobboon และ JS Neeno ตัวอักษรขนาด 30 points มีรูปแบบของตัวอักษรคือ JS Macha และ JS Prajuk(Italic) ในโปรแกรม Adobe Photoshop



ภาพที่ 3-13 ข้อมูลภาพที่ใช้ในงานวิจัย

3.7 การวัดประสิทธิภาพ

ในการวัดประสิทธิภาพที่ได้จากการทำส่วนหน้า ใช้วิธีการสังเกตด้วยสายตาว่าวิธีการใดมีการแยกส่วนที่เป็นตัวอักษรออกมาได้ชัดเจน และการวัดประสิทธิภาพของการกู้คืนรายละเอียดของภาพ คือการวัดคุณภาพที่ได้ด้วยการใช้ค่าสัดส่วนสัญญาณต่อสัญญาณรบกวนสูงสุด (Peak Signal to Noise Ratio หรือ PSNR) ดังสมการที่ 2-17 ในบทที่ 2

3.8 สรุป

ในงานวิจัยนี้ประกอบด้วยกระบวนการที่สำคัญ 2 ขั้นตอน ได้แก่ ขั้นตอนการหาส่วนที่เป็นตัวอักษรและขั้นตอนการปกปิดพื้นผิว ซึ่งขั้นตอนการหาส่วนที่เป็นตัวอักษรนั้นประกอบไปด้วยขั้นตอนย่อยอีก 4 ขั้นตอนคือ การแบ่งภาพออกเป็นภาพย่อย การแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว-ดำ และการลดสิ่งรบกวน โดยวิธีการที่เลือกใช้ในงานวิจัยนี้คือ การแบ่งกลุ่มโดยใช้ฟัซซี่ซีมีน การทำภาพเป็นขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนโดยใช้การหาขอบของวัตถุ และการวัดขนาดของวัตถุ

ขั้นตอนของการปกปิดพื้นผิวนั้นใช้การสังเคราะห์พื้นผิว และทำการปกปิดความผิดพลาดแบบแบบตัวแปรปรับเปลี่ยนค่าได้ และทำการเปรียบเทียบกับทำการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปร โดยการปกปิดความผิดพลาดนั้นจะทำหลังจากที่ทำการสังเคราะห์พื้นผิวแล้ว โดยจะแสดงการสรุปวิธีการกู้คืนภาพด้วยภาพที่ 4-8 ในบทที่ 4 และการวัดประสิทธิภาพที่ได้จากการกู้คืนภาพด้วยการวัดคุณภาพที่ได้โดยการใช้ค่าสัดส่วนสัญญาณต่อสัญญาณรบกวนสูงสุด (Peak Signal to Noise Ratio หรือ PSNR)

บทที่ 4

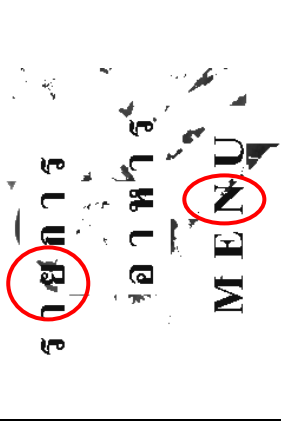



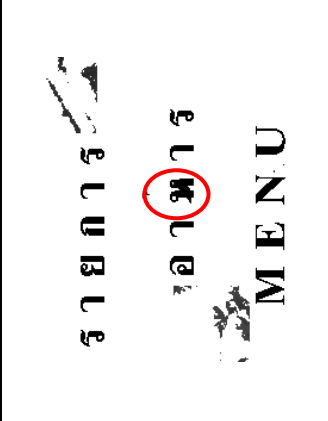
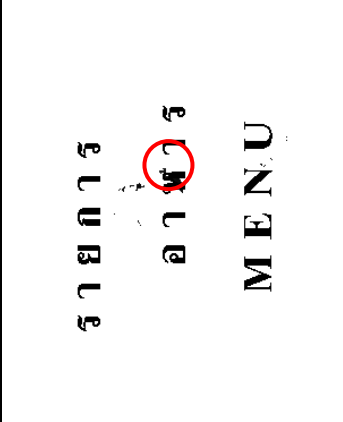
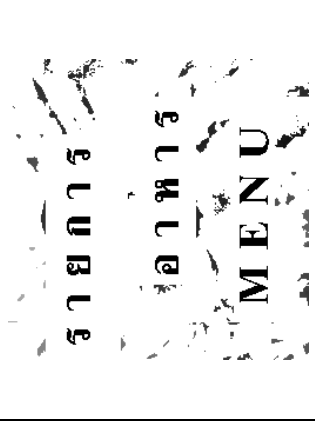
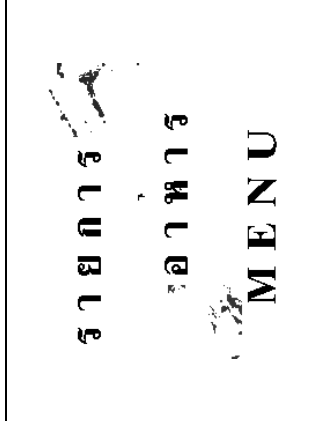
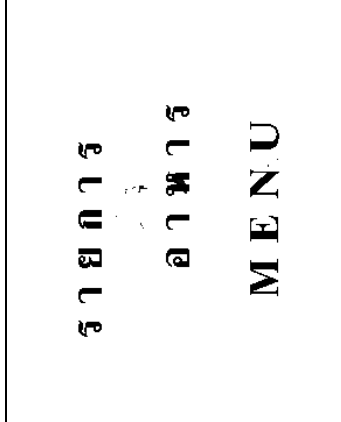
ผลการวิจัยและการวิเคราะห์ผลการวิจัย

งานวิจัยนี้เป็นการกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ โดยแบ่งออกเป็น 2 กระบวนการหลัก คือ การหาส่วนที่เป็นตัวอักษรออกจากภาพ และการปกปิดพื้นผิว ในขั้นตอนของการหาส่วนที่เป็นตัวอักษรได้ทำการเปรียบเทียบวิธีที่ใช้ในการแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว-ดำ และการลดสิ่งรบกวน ส่วนขั้นตอนการปกปิดพื้นผิวได้ทำการเปรียบเทียบผลลัพธ์ที่ได้จากการสังเคราะห์พื้นผิว การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (Static Texture Concealment) และการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว (Adaptive Texture Concealment) ซึ่งผลที่ได้จากการทดลองจะนำมาประเมินผลและวิเคราะห์ผล ดังนี้




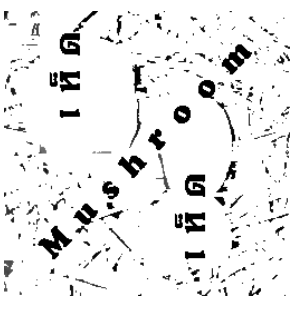





4.1 ผลการวิจัย

4.1.1 ผลการทดลองการแบ่งกลุ่มข้อมูลทั้ง 3 วิธีการ คือ การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน, การแบ่งกลุ่มด้วยเคมีน และการแบ่งกลุ่มด้วยฟuzzyซิมิน โดยแบ่งภาพย่อยทั้งหมด 3 ขนาดคือ 32×32 , 64×64 และ 128×128 พิกเซล ดังตารางที่ 4-1

ตารางที่ 4-1 ตัวอย่างผลการทดลองการแบ่งกลุ่ม

รายการเปรียบเทียบ	ขนาดภาพย่อย		
	32 × 32	64 × 64	128 × 128
1. การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน			
2. การแบ่งกลุ่มด้วยเคมिन			
3. การแบ่งกลุ่มด้วยพีชชีมิน			

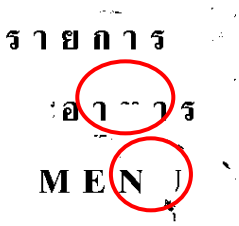

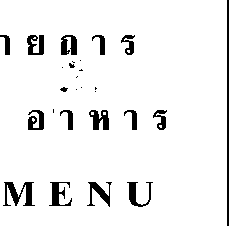


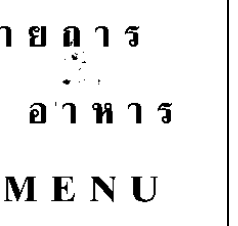
ตารางที่ 4-1 (ต่อ)

รายการเปรียบเทียบ	ขนาดภาพย่อย		
	32 × 32	64 × 64	128 × 128
1. การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน			
2. การแบ่งกลุ่มด้วยเคมิน			
3. การแบ่งกลุ่มด้วยฟuzzyซีมิน			







จากผลการแบ่งกลุ่มโดยใช้ค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐานจะยังคงมีส่วนที่เป็นพื้นหลังติดมากับส่วนที่เป็นตัวอักษรมากกว่าการแบ่งกลุ่มโดยใช้เคมีนและการแบ่งกลุ่มโดยใช้ฟัซซี่ซิมิน ซึ่งการแบ่งกลุ่มโดยใช้เคมีนจะยังคงมีส่วนที่เป็นพื้นหลังติดอยู่กับตัวอักษรบ้างเล็กน้อย แต่เมื่อเทียบกับการแบ่งกลุ่มโดยใช้ฟัซซี่ซิมินจะเห็นว่าตัวอักษรสามารถแยกออกมาได้อย่างชัดเจนไม่มีส่วนที่เป็นพื้นหลังติดมากับตัวอักษรเลย ซึ่งผลการแบ่งกลุ่มโดยใช้ฟัซซี่ซิมินให้ผลดีกว่า ดังแสดงในตารางที่ 4-1

4.1.2 ผลการทดลองการทำภาพเป็นขาว-ดำ 2 วิธีคือ การทำภาพเป็นขาว-ดำ โดยใช้อะแดปทีฟ เทอร์ชโฮล และการทำภาพขาว – ดำโดยใช้ฟัซซี่ โดยแบ่งภาพย่อยทั้งหมด 3 ขนาดคือ 32×32 , 64×64 และ 128×128 พิกเซล ดังตารางที่ 4-2

ตารางที่ 4-2 ตัวอย่างการทดลองการทำภาพขาว-ดำ

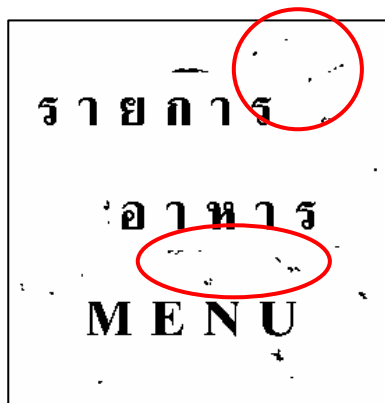
รายการเปรียบเทียบ	ขนาดภาพย่อย		
	32×32	64×64	128×128
1. Adaptive Binarization			
2. Fuzzy Binarization			

ตารางที่ 4-2 (ต่อ)

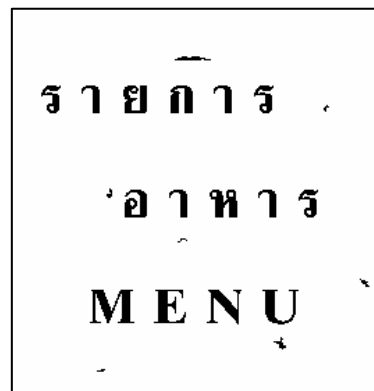
รายการ เปรียบเทียบ	ขนาดภาพย่อย		
	32 × 32	64 × 64	128 × 128
1. Adaptive Binarization			
2. Fuzzy Binarization			

ผลที่ได้จากการทำภาพเป็นขาว – ดำด้วยการใช้ฟัซซี่จะสามารถแยกส่วนที่เป็นตัวอักษรกับพื้นหลังได้ดีกว่าการใช้อะแดปทีฟเทรชโฮลในภาพย่อยที่มีขนาด 32 × 32 พิกเซล เนื่องจากการใช้อะแดปทีฟเทรชโฮลจะทำให้ส่วนที่เป็นตัวอักษรหายไปบางภาพย่อย แต่ถ้านำวิธีการทั้งสองไปใช้กับขนาดภาพย่อยที่มีขนาดใหญ่ขึ้นแทบจะไม่มีผลที่แตกต่างกัน ดังแสดงในตารางที่ 4-2

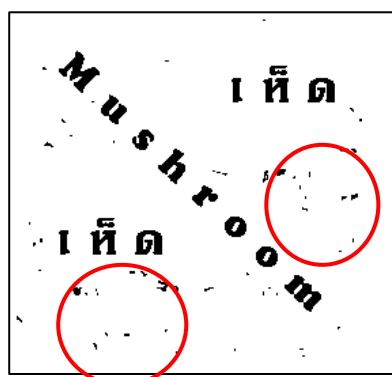
4.1.3 ผลการทดลองการลดสิ่งรบกวน 2 วิธีคือ การลดสิ่งรบกวนด้วยการควบคุมความยาวในการวิ่ง และการลดสิ่งรบกวนด้วยการหาขอบของวัตถุและหาขนาดของวัตถุ ดังภาพที่ 4-1



(ก)



(ข)



(ค)



(ง)

ภาพที่ 4-1 ภาพตัวอย่างผลการลดสิ่งรบกวน (ก) และ (ค) การลดสิ่งรบกวนด้วยการควบคุมความยาวในการวิ่ง (ข) และ (ง) การลดสิ่งรบกวนด้วยการหาขอบของวัตถุและหาขนาดของวัตถุ

ผลที่ได้จากการลดสิ่งรบกวนด้วยการหาขอบของวัตถุและการวัดขนาดของวัตถุนั้นจะลดสิ่งรบกวนหรือส่วนที่ไม่ใช่ตัวอักษรได้มากกว่าการลดสิ่งรบกวนด้วยการควบคุมความยาวในการวิ่ง ดังแสดงในภาพที่ 4-1

4.1.4 ผลการทดลองการกู้คืนรายละเอียดของภาพ แบ่งเป็น 3 วิธีคือ การสังเคราะห์พื้นผิว การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว และการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว

ตารางที่ 4-3 การเปรียบเทียบค่า PSNR ในการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซิมิน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 32×32 พิกเซล

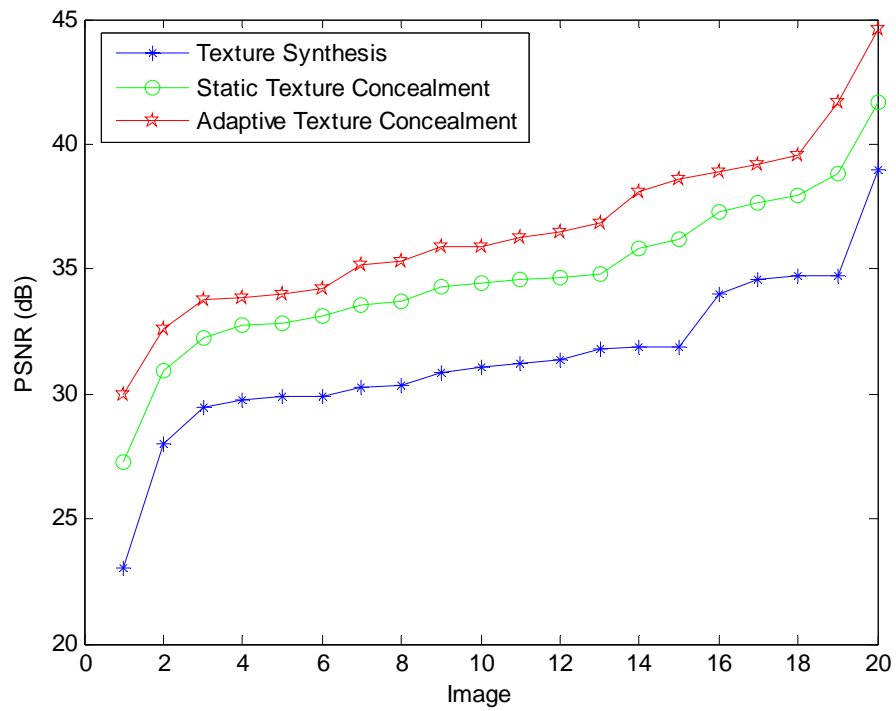
ภาพ	Texture Synthesis	Static Texture Concealment	Adaptive Texture Concealment
im01.bmp	23.05 dB	27.26 dB	29.96 dB
im02.bmp	30.29 dB	32.84 dB	34.20 dB
im03.bmp	34.70 dB	37.68 dB	39.53 dB
im04.bmp	38.98 dB	41.65 dB	44.57 dB
im05.bmp	34.75 dB	37.92 dB	38.87 dB
im06.bmp	31.07 dB	36.22 dB	38.61 dB
im07.bmp	31.39 dB	34.28 dB	35.19 dB
im08.bmp	29.73 dB	32.73 dB	33.88 dB
im09.bmp	31.19 dB	34.45 dB	36.49 dB
im10.bmp	31.91 dB	34.78 dB	36.87 dB
im11.bmp	34.00 dB	37.32 dB	39.16 dB
im12.bmp	30.37 dB	34.60 dB	35.91 dB
im13.bmp	29.49 dB	32.23 dB	33.77 dB
im14.bmp	31.86 dB	35.80 dB	38.09 dB
im15.bmp	29.92 dB	33.59 dB	35.31 dB
im16.bmp	31.78 dB	34.67 dB	36.23 dB
im17.bmp	29.92 dB	33.13 dB	34.01 dB
im18.bmp	30.85 dB	33.69 dB	35.88 dB
im19.bmp	34.61 dB	38.79 dB	41.69 dB
im20.bmp	27.97 dB	30.93 dB	32.58 dB

ตารางที่ 4-4 การเปรียบเทียบค่า PSNR ในการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซิมิน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 64×64 พิกเซล

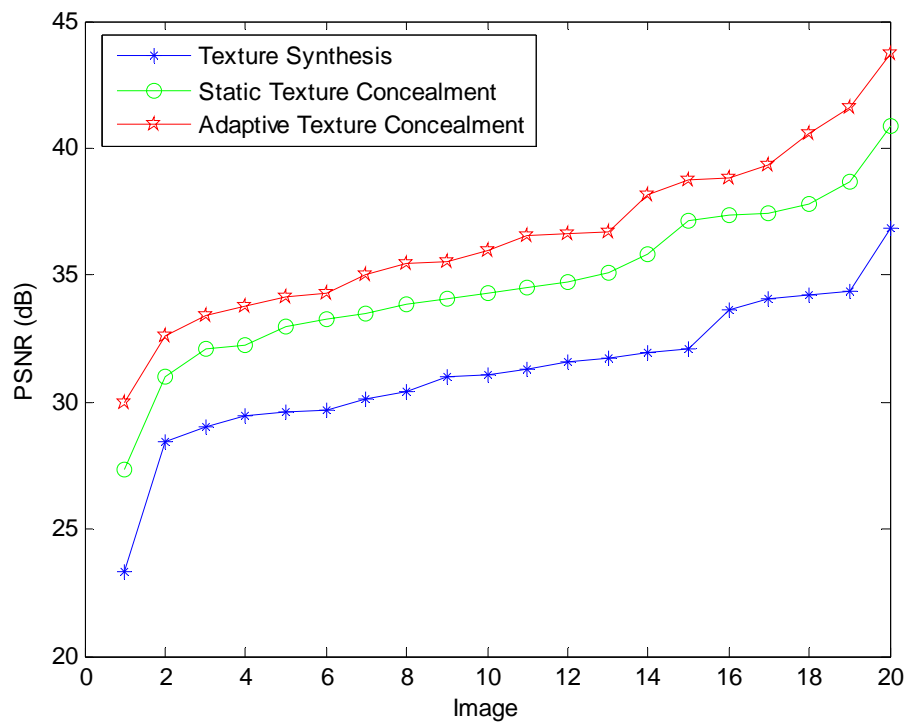
ภาพ	Texture Synthesis	Static Texture Concealment	Adaptive Texture Concealment
im01.bmp	23.34 dB	27.38 dB	29.98 dB
im02.bmp	29.62 dB	32.08 dB	33.38 dB
im03.bmp	34.21 dB	37.15 dB	38.72 dB
im04.bmp	36.88 dB	40.85 dB	43.72 dB
im05.bmp	34.33 dB	37.82 dB	38.79 dB
im06.bmp	31.59 dB	37.34 dB	40.59 dB
im07.bmp	31.09 dB	34.04 dB	35.05 dB
im08.bmp	29.70 dB	32.95 dB	34.30 dB
im09.bmp	31.28 dB	34.54 dB	36.59 dB
im10.bmp	31.93 dB	34.74 dB	36.65 dB
im11.bmp	33.66 dB	37.45 dB	39.35 dB
im12.bmp	30.42 dB	34.26 dB	35.54 dB
im13.bmp	29.05 dB	32.23 dB	33.81 dB
im14.bmp	32.13 dB	35.86 dB	38.16 dB
im15.bmp	29.47 dB	33.51 dB	35.43 dB
im16.bmp	31.74 dB	35.10 dB	36.68 dB
im17.bmp	30.12 dB	33.24 dB	34.15 dB
im18.bmp	30.97 dB	33.83 dB	35.99 dB
im19.bmp	34.05 dB	38.69 dB	41.59 dB
im20.bmp	28.43 dB	31.01 dB	32.60 dB

ตารางที่ 4-5 การเปรียบเทียบค่า PSNR ในการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซิมิน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ ขนาดภาพย่อย 128×128 พิกเซล

ภาพ	Texture Synthesis	Static Texture Concealment	Adaptive Texture Concealment
im01.bmp	22.18 dB	25.67 dB	28.09 dB
im02.bmp	30.64 dB	32.98 dB	34.30 dB
im03.bmp	35.15 dB	37.76 dB	39.65 dB
im04.bmp	37.29 dB	42.10 dB	45.10 dB
im05.bmp	34.19 dB	37.81 dB	38.99 dB
im06.bmp	31.56 dB	37.35 dB	40.59 dB
im07.bmp	31.23 dB	34.21 dB	35.13 dB
im08.bmp	30.12 dB	33.03 dB	34.42 dB
im09.bmp	31.27 dB	34.30 dB	36.35 dB
im10.bmp	32.19 dB	34.98 dB	36.97 dB
im11.bmp	33.79 dB	37.38 dB	39.31 dB
im12.bmp	30.72 dB	34.89 dB	36.16 dB
im13.bmp	29.39 dB	32.29 dB	33.84 dB
im14.bmp	32.24 dB	35.91 dB	38.12 dB
im15.bmp	30.26 dB	33.91 dB	35.75 dB
im16.bmp	32.26 dB	35.11 dB	36.69 dB
im17.bmp	29.66 dB	33.26 dB	34.14 dB
im18.bmp	30.94 dB	33.62 dB	35.74 dB
im19.bmp	33.45 dB	38.38 dB	41.34 dB
im20.bmp	28.35 dB	31.01 dB	32.55 dB

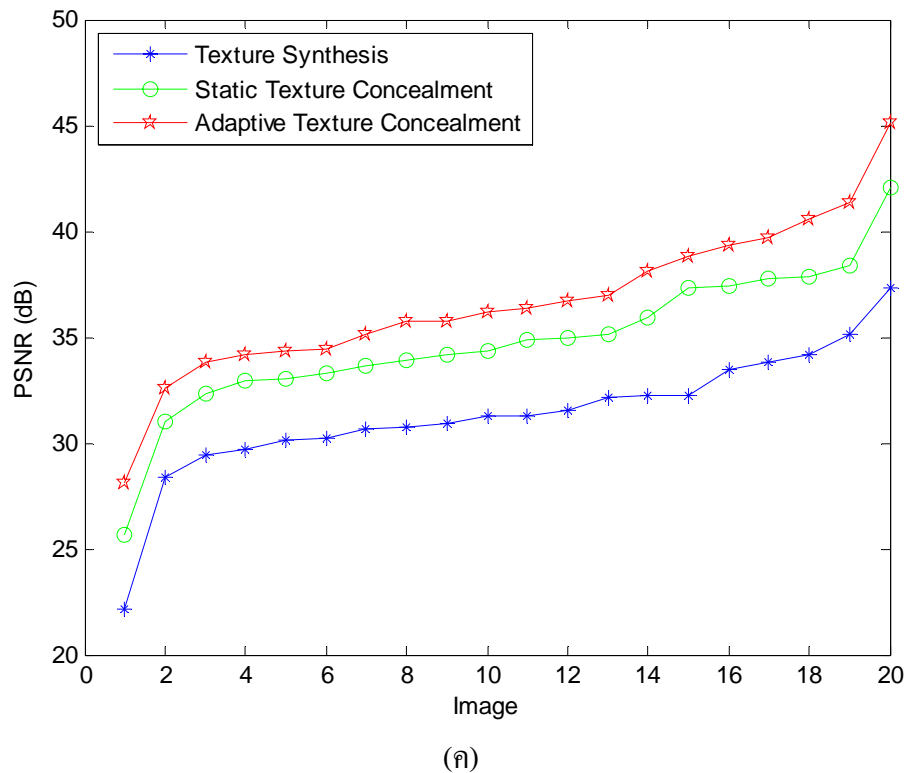


(ก)



(ข)

ภาพที่ 4-2 กราฟเปรียบเทียบค่า PSNR ในการกู้คืนรายละเอียดภาพ (ก) ขนาดภาพย่อย 32×32 พิกเซล (ข) ขนาดภาพย่อย 64×64 พิกเซล (ค) ขนาดภาพย่อย 128×128 พิกเซล



ภาพที่ 4-2 (ต่อ)

ผลที่ได้จากการกู้คืนภาพโดยใช้การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิวให้ผลการกู้คืนภาพดีกว่าการกู้คืนภาพด้วยการสังเคราะห์พื้นผิว แต่การปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิวให้ผลการกู้คืนภาพดีกว่าการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว โดยการเปรียบเทียบจากค่า PSNR ดังแสดงในตารางที่ 4-3 ถึงตารางที่ 4-5 และภาพที่ 4-2

ผลจากการเปรียบเทียบภาพที่ได้จากการกู้คืนภาพจากการสังเคราะห์พื้นผิวนั้นยังคงร่องรอยของส่วนที่เป็นตัวอักษรให้เห็นค่อนข้างชัดเจน เมื่อเทียบกับผลที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิวจะพบว่าร่องรอยที่เป็นตัวอักษรลดลงแต่ก็ยังคงมีให้เห็นอยู่บ้าง แต่เมื่อนำวิธีการทั้งสองวิธีข้างต้นมาเปรียบเทียบกับวิธีการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิวจะเห็นว่าร่องรอยในส่วนที่เป็นตัวอักษรแทบจะมองไม่เห็นหรือไม่ปรากฏให้เห็นเลยในบางภาพ ดังแสดงในภาพที่ 4-3 ถึงภาพที่ 4-7



(ก)



(ข)



(ค)



(ง)

ภาพที่ 4-3 ภาพตัวอย่างที่ได้จากการรู้ถิ่นรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซีมีน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (ก) ภาพนำเข้า im03.bmp (ข) ภาพที่ได้จากการสังเคราะห์พื้นผิว (ค) ภาพที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (ง) ภาพที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว



(ก)



(ข)



(ค)



(ง)

ภาพที่ 4-4 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซีมีน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (ก) ภาพนำเข้า im10.bmp (ข) ภาพที่ได้จากการสังเคราะห์พื้นผิว (ค) ภาพที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (ง) ภาพที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว



(ก)



(ข)



(ค)



(ง)

ภาพที่ 4-5 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซีมีน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (ก) ภาพนำเข้า im12.bmp (ข) ภาพที่ได้จากการสังเคราะห์พื้นผิว (ค) ภาพที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (ง) ภาพที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว



(ก)



(ข)



(ค)



(ง)

ภาพที่ 4-6 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซีมีน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (ก) ภาพนำเข้า im15.bmp (ข) ภาพที่ได้จากการสังเคราะห์พื้นผิว (ค) ภาพที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (ง) ภาพที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว



(ก)



(ข)



(ค)



(ง)

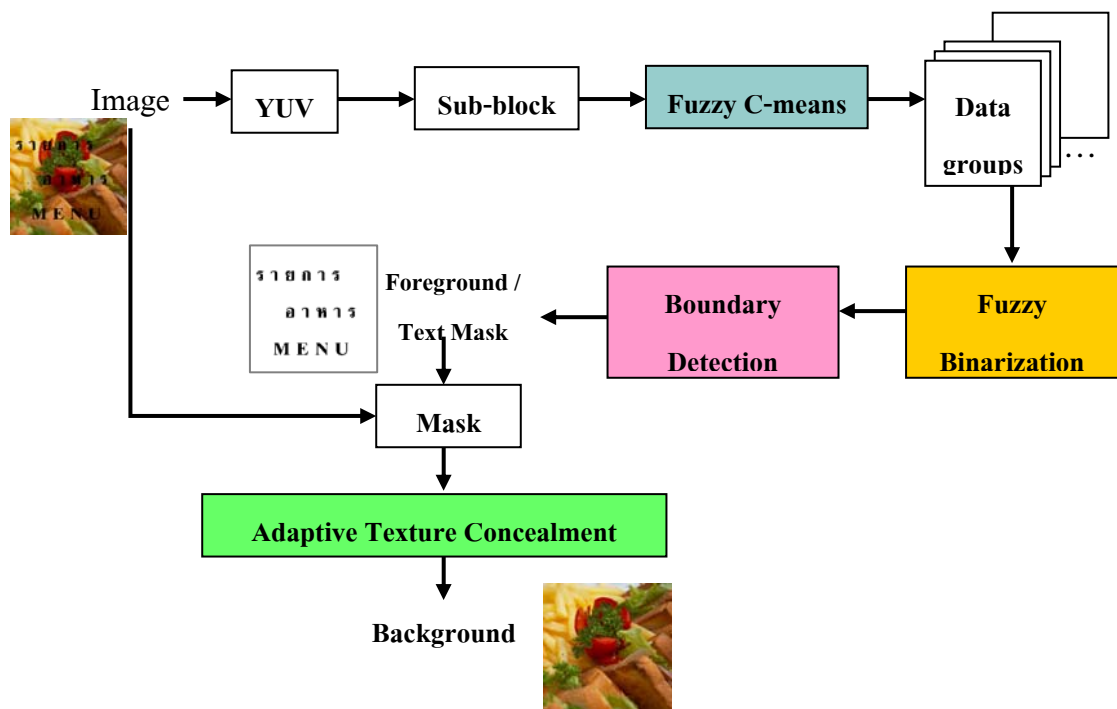
ภาพที่ 4-7 ภาพตัวอย่างที่ได้จากการกู้คืนรายละเอียดภาพโดยการแบ่งกลุ่มด้วยฟัซซี่ซีมีน การทำภาพขาว-ดำโดยใช้ฟัซซี่ และการลดสิ่งรบกวนด้วยการหาขอบวัตถุและขนาดของวัตถุ (ก) ภาพนำเข้า im18.bmp (ข) ภาพที่ได้จากการสังเคราะห์พื้นผิว (ค) ภาพที่ได้จากการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิว (ง) ภาพที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว

4.2 วิเคราะห์ผลการทดลอง

จากผลการทดลองการแบ่งกลุ่มข้อมูลพบว่าขนาดของภาพย่อยมีผลต่อการแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน แต่การแบ่งกลุ่มด้วยเคมีนและพีชชีมีนจะให้ผลการแบ่งกลุ่มได้ดีในทุกขนาดของภาพย่อย ซึ่งพีชชีมีนจะให้ผลที่ดีที่สุดในการแบ่งกลุ่มข้อมูล ในส่วนของการทำงานเป็นขาว – ดำนั้นการใช้พีชชีเข้ามาช่วยในการทำงานเป็นขาว - ดำจะให้ผลลัพธ์ที่ดีกว่าการทำงานโดยใช้อะแดปทีฟเทรชโฮลในภาพย่อยขนาด 32×32 พิกเซล ส่วนขนาดภาพย่อยอื่นทั้งสองวิธีให้ผลลัพธ์ที่ดีพอๆ กัน และในส่วนของกลดสิ่งรบกวนการใช้วิธีการหาขอบของวัตถุและหาขนาดของวัตถุจะทำการกลดสิ่งรบกวนได้ดีกว่าการกลดสิ่งรบกวนด้วยการควบคุมความยาวในการวิ่ง

ผลลัพธ์ในการกู้คืนภาพนั้น การสังเคราะห์พื้นผิวสามารถกู้คืนรายละเอียดภาพและให้ความกลมกลืนของภาพได้ดีในระดับหนึ่ง แต่การนำการปกปิดความผิดพลาดเข้ามาใช้หลังจากการทำงานการสังเคราะห์พื้นผิวแล้วนั้นจะทำให้ภาพมีความกลมกลืนมากขึ้น โดยเฉพาะการการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิวจะให้ผลในการกู้คืนภาพ (ค่า PSNR มากที่สุด) และมีความกลมกลืนภาพดีที่สุด

จากการทดลองนั้นพบว่าถ้าทำการแยกส่วนหน้าหรือ Text Mask ได้ชัดเจน การระบุตำแหน่งในการกู้คืนภาพ และทำการกู้คืนรายละเอียดภาพก็จะได้ผลลัพธ์ดีมากขึ้น ซึ่งสามารถสรุปวิธีการในการกู้คืนภาพได้ดังภาพที่ 4-8 ในขั้นตอนของการแบ่งกลุ่มข้อมูลจะใช้การแบ่งกลุ่มข้อมูลด้วยพีชชีมีน การทำงานเป็นขาว-ดำโดยใช้พีชชี การกลดสิ่งรบกวนใช้วิธีการหาขอบของวัตถุและหาขนาดของวัตถุ และการกู้คืนภาพใช้การปกปิดความผิดพลาดหลังการสังเคราะห์ด้วยพื้นผิว



ภาพที่ 4-8 กระบวนการกู้คืนภาพจากการถูกลบตัวอักษรซ้อนทับ

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

การกู้คืนภาพกลับคืนมาจากการถูกตัวอักษรซ้อนทับเป็นการทดลอง เพื่อพัฒนาขั้นตอนวิธีการกู้คืนรายละเอียดของภาพที่ขาดหายไปจากการแยกตัวอักษรหรือข้อความ โดยผู้วิจัยได้แบ่งขั้นตอนการทดลองออกเป็น 2 ขั้นตอนใหญ่ๆ คือการหาส่วนที่เป็นตัวอักษร และการปกปิดพื้นผิว

ในส่วนของการหาส่วนที่เป็นตัวอักษรแบ่งออกเป็น 4 ขั้นตอนคือ การแบ่งภาพเป็นภาพย่อย การแบ่งกลุ่มข้อมูล การทำภาพเป็นขาว – ดำ และการลดสิ่งรบกวน โดยขั้นตอนการแบ่งกลุ่มข้อมูลผู้วิจัยได้ทำการศึกษา วิเคราะห์และเปรียบเทียบการแบ่งกลุ่มข้อมูล 3 วิธีด้วยกันคือ การแบ่งกลุ่มด้วยค่าเฉลี่ยและค่าเบี่ยงเบนมาตรฐาน การแบ่งกลุ่มด้วยเคมีน และการแบ่งกลุ่มด้วยฟิชชีชีมัน ส่วนขั้นตอนการทำภาพเป็นขาว – ดำ ผู้วิจัยได้ทำการศึกษา วิเคราะห์และเปรียบเทียบ 2 วิธีคือ การทำภาพเป็นขาว - ดำใช้อะแดปทีฟเทรชโฮล และการทำภาพเป็นขาว - ดำใช้ฟิชชีชี และขั้นตอนสุดท้ายการลดสิ่งรบกวนผู้วิจัยได้ทำการศึกษา วิเคราะห์และเปรียบเทียบ 2 วิธีคือ การควบคุมความยาวในการวิ่ง และการหาขอบและขนาดของวัตถุ

จากการทดลองที่ได้จากการหาส่วนที่เป็นตัวอักษรนี้ ผู้วิจัยได้เลือกวิธีการแบ่งกลุ่มข้อมูลด้วยฟิชชีชีมันในการแบ่งกลุ่มตัวอักษรออกจากพื้นหลัง การทำภาพเป็นขาว – ดำโดยใช้ฟิชชีชี และการลดสิ่งรบกวนใช้การหาขอบของวัตถุและวัดขนาดของวัตถุ เพื่อลดส่วนที่ไม่ใช่ตัวอักษรให้ได้มากที่สุด ซึ่งผลที่ได้จากการทำในส่วนนี้เป็นส่วนสำคัญที่จะบ่งบอกถึงตำแหน่งที่ต้องทำการกู้คืนภาพ

ในส่วนของการปกปิดพื้นผิวแบ่งเป็น 2 ขั้นตอนคือ การสังเคราะห์พื้นผิวและการปกปิดความผิดพลาด โดยผู้วิจัยทำการศึกษา วิเคราะห์และเปรียบเทียบการสังเคราะห์พื้นผิว การปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลังการสังเคราะห์พื้นผิวและการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิว

จากการทดลองที่ได้จากการปกปิดพื้นผิว ทำให้ทราบว่า การสังเคราะห์พื้นผิวเพียงอย่างเดียว นั้นให้ผลลัพธ์ที่น่าพอใจในระดับหนึ่งแต่ยังคงเกิดร่องรอยของบล็อกให้เห็นอยู่ ดังนั้นการนำการผิดพลาดเข้ามาช่วยในการจัดส่วนที่เป็นร่องรอยของบล็อกนั้นทำให้ภาพเกิดความกลมกลืนกันมากขึ้น และจากการวิเคราะห์และเปรียบเทียบการปกปิดความผิดพลาดแบบกำหนดค่าตัวแปรหลัง

การสังเคราะห์พื้นผิวและการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิวนั้น ผลลัพธ์ที่ได้จากการปกปิดความผิดพลาดแบบตัวแปรปรับเปลี่ยนค่าได้หลังการสังเคราะห์พื้นผิวนั้นทำให้การกู้คืนภาพมีความกลมกลืนกันและมีคุณภาพที่สูง

จากการทดลองดังกล่าวทำให้ทราบว่าผลจากการกู้คืนภาพนั้น ขั้นตอนการหาส่วนที่เป็นตัวอักษรมีความสำคัญอย่างมาก เพราะการกู้คืนภาพจะได้ภาพออกมาชัดเจนหรือไม่ขึ้นขึ้นอยู่กับขั้นตอนนี้ ดังนั้นการทดลองนี้นับได้ว่าเป็นการพัฒนาขั้นตอนวิธีการกู้คืนรายละเอียดภาพที่ขาดหายไปจากการถูกตัวอักษรซ้อนทับ โดยการสังเคราะห์และปกปิดความผิดพลาดได้อีกวิธีหนึ่ง

5.2 ข้อเสนอแนะ

งานวิจัยนี้สามารถนำไปประยุกต์ใช้กับภาพที่เกิดความเสียหายที่ไม่ได้เกิดจากการถูกตัวอักษรซ้อนทับที่เป็นสีดำเพียงเท่านั้น แต่ยังสามารถใช้ได้กับตัวอักษรหรือส่วนที่เกิดความเสียหายรูปแบบอื่นที่เป็นสีเข้มด้วย เช่น สีน้ำเงินกรมท่า สีเขียวขี้ม้า เป็นต้น

จากการทดลองจะเห็นว่าการแยกตัวอักษรและลบส่วนที่เป็นสิ่งรบกวนนั้น เป็นขั้นตอนหนึ่งที่สำคัญที่จะใช้ในกู้คืนภาพกลับมา และการหาขอบของวัตถุเพื่อทำการลบส่วนที่เป็นสิ่งรบกวนนั้น ได้ทำการกำหนดค่าเพื่อหาขนาดของวัตถุยังทำได้ไม่ดีนัก เพราะว่ายังมีข้อผิดพลาดโดยจะทำการลบส่วนที่เป็นสระของรูปแบบของตัวอักษรบางรูปแบบออกไปบางส่วนด้วย ดังนั้นเพื่อให้การทำงานได้ผลลัพธ์ที่ดีกว่าควรกำหนดระยะทางที่ใช้วัดขนาดโดยอัตโนมัติ เพราะว่าตัวอักษรนั้นมีด้วยกันหลากหลายรูปแบบ ซึ่งแต่ละแบบก็จะมีลักษณะ รูปร่าง และขนาดที่แตกต่างกัน

เอกสารอ้างอิง

1. L.A. Fletcher, and R. Kasturi. "A Robust Algorithm for Text String Separation from mixed text / graphics images." IEEE Trans. Pattern Anal. March. Intell. 10 (November 1988) : 910-918.
2. E. Savakis. Adaptive Document Image Thresholding Using Foreground and Background Clustering. Available online at http://www.ce.rit.edu/~savakis/papers/ICIP98_savakis.pdf
3. B.F. Wu, C.-C. Chiu and Y.-L. Chen "Algorithm for compressing compound document image with large text / background overlap." IEE Proc. -Vis. Image Signal process. 151 (December 2004) : 453- 459.
4. Savehenko, Kojekine and Unno. " A Practical Image Retouching Method." Tokyo: Hosei University (Mimeographed). (2002) : 480.
5. Alexei A. Efros and Thomas K. Leung, "Texture Synthesis by Non-Parametric Sampling." IEEE International Conference on Computer Vision. (September 1999) : 1033-1038.
6. Alexei A. Efros and William T. Freeman. "Image Quilting for Texture Synthesis and Transfer." SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. (2001) : 341-346.
7. Hung-Ming Sun, "Page Segmentation for Manhattan and Non-Manhattan Layout Documents via Selective CRLA." IEEE International Conference on Document Analysis and Recognition. (2005) : 116-120.
8. Klir, G.J. and Yuan, B., "Fuzzy set and Fuzzy logic : theory and application." New Jersey : Prentice Hall, 1995.
9. พยุง มีดีจ. Fuzzy Systems and Neural Networks. กรุงเทพฯ : คณะเทคโนโลยีสารสนเทศ มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2548.
10. Edge Detection. Available online at <http://fivedots.coe.psu.ac.th/~montri/Teaching/240-373/Chapter8.pdf>

11. พุดิพงศ์ พงษ์ตราฐ. การปกปิดความผิดพลาดสำหรับการบีบอัดข้อมูลแบบแฟร็กทัล.
กรุงเทพฯ : คณะเทคโนโลยีสารสนเทศมหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ,
2549.
12. Paul Salame and Edward I. Delp, "Error Concealment in MPEG Video Streams Over ATM
Networks." IEEE Journal on selected areas in communications. 18 (June 2000) : 1129-
1144.
13. Jirapat, Para and Ruttikorn. "Error Concealment for Stereoscopic." Proceedings of ROVISIP
2003 (The International Conference on Robotics, Vision, Information and Signal
Processing) (January 2003) : 39-44.
14. J.B.MacQueen "Some Methods for classification and Analysis of Multivariate
Observations." Berkeley, University of California Press, 1 (1967) : 281-297.
15. J.C.Bezdec, "Pattern Recognition with Fuzzy Objective Function Algorithm." Plenum Press,
New York. (1981) : 256 .

ประวัติผู้วิจัย

ชื่อ : นางสาวธนรัตน์ เหลืองอ่อน
ชื่อวิทยานิพนธ์ : การกู้คืนภาพจากการถูกตัวอักษรซ้อนทับ
สาขาวิชา : เทคโนโลยีคอมพิวเตอร์

ประวัติ

ประวัติส่วนตัว เกิดวันที่ 6 กรกฎาคม 2525 ที่จังหวัดชลบุรี

ประวัติการศึกษา สำเร็จการศึกษาระดับมัธยมศึกษาตอนต้น สาขาวิชาวิทยาศาสตร์บัณฑิต สาขาวิชาระบบสารสนเทศคอมพิวเตอร์ จากมหาวิทยาลัยบูรพา ในปีการศึกษา 2546

ประวัติการทำงาน พ.ศ. 2549 – 2550 พนักงานในตำแหน่งคอมพิวเตอร์ บริษัทเซอร์เรียลสตูดิโอ จำกัด