

การบีบอัดข้อมูลภาพหลายแบนด์โดยใช้การควอนไทซ์แบบเวกเตอร์
ที่คำนวณโดยพีชชีซีมีนส์

MULTISPECTRAL IMAGE COMPRESSION USING
FCM-BASED VECTOR QUANTIZATION



อุทัย แสงทองพราว
UTHAI SANGTHONGPRAOW

เลขหน้.....
เลขทะเบียน 44050
งัน, เดือน, ปี 2 5 ค.ศ. 2545

.b.....
.i.....

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชาวิศวกรรมไฟฟ้า
บัณฑิตวิทยาลัย
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
พ.ศ. 2545

**MULTISPECTRAL IMAGE COMPRESSION USING
FCM-BASED VECTOR QUANTIZATION**



**A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING
SCHOOL OF GRADUATE STUDIES
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG**

2002

ISBN 974-324-021-7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2002

SCHOOL OF GRADUATE STUDIES

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อวิทยานิพนธ์	การบีบอัดข้อมูลภาพหลายแบนด์โดยใช้การควอนไทซ์แบบ
	เวกเตอร์ที่คำนวณโดยพีชชีมีนส์
นักศึกษา	นายอุทัย แสงทองพราว
รหัสประจำตัว	42061105
ปริญญา	วิศวกรรมศาสตรมหาบัณฑิต
สาขาวิชา	วิศวกรรมไฟฟ้า
พ.ศ.	2545
อาจารย์ผู้ควบคุมวิทยานิพนธ์	รศ.ดร.บุทรพงษ์ รังสรรค์เสรี

บทคัดย่อ

การบีบอัดข้อมูลเป็นกระบวนการในการลดจำนวนบิตในการเก็บข้อมูล การใช้เวกเตอร์ควอนไทซ์ชั้นก็เป็นวิธีหนึ่งที่ถูกนำมาใช้ในกระบวนการนี้ ด้วยวิธีการควอนไทซ์แบบเวกเตอร์นี้กลุ่มข้อมูลจะถูกเข้ารหัสด้วยเวกเตอร์อ้างอิงกลุ่มหนึ่ง การใช้วิธีเวกเตอร์ควอนไทซ์ชั้นนี้เป็นวิธีการบีบอัดข้อมูลที่สามารถนำไปประยุกต์ใช้ได้กับข้อมูลหลายๆประเภท เช่น ข้อมูลภาพ เสียง หรือ เอกสาร รวมไปถึงภาพถ่ายดาวเทียม ซึ่งวิธีที่จะกล่าวนี้เป็นวิธีที่ให้ผลลัพธ์ของการบีบอัดข้อมูลที่ดีกว่าวิธีอื่นๆซึ่งใช้การเข้ารหัสแบบสเกลลาร์ วิทยานิพนธ์นี้จะกล่าวถึงวิธีการบีบอัดข้อมูลภาพหลายแบนด์ด้วยวิธีควอนไทซ์แบบเวกเตอร์โดยใช้การจำแนกข้อมูลแบบพีชชีมีนส์มาช่วย ด้วยวิธีดังกล่าวนี้ในขั้นตอนแรกเราจะใช้กระบวนการพีชชีมีนส์ในการกำหนดค่าเริ่มต้นของค่าเวกเตอร์อ้างอิงที่ใช้ในการแทนกลุ่มข้อมูล โดยในขั้นตอนนี้เราจะทำการปรับกระบวนการพีชชีมีนส์ให้มีความเร็วมากขึ้นด้วยเพื่อเป็นการลดเวลาในการคำนวณเวกเตอร์อ้างอิง จากนั้นจะนำกลุ่มข้อมูลเวกเตอร์อ้างอิงเหล่านี้ในการคำนวณค่าการควอนไทซ์แบบเวกเตอร์ที่จะนำไปใช้กับภาพหลายแบนด์และจะนำผลการทดลองมาแสดงให้เห็นถึงประสิทธิภาพของวิธีที่เสนอนี้

Thesis Title	Multispectral Image Compression Using FCM-Based Vector Quantization
Student	Mr. Uthai Sangthongpraow
Student ID.	42061105
Degree	Master of Engineering
Programme	Electrical Engineering
Year	2002
Thesis Advisor	Assoc.Prof. Yutthapong Rangsanseri

ABSTRACT

Image compression is the process of reducing the number of bits required to represent an image. Vector Quantization method (VQ) is a method to deal with this operation. With this method a set of data points is encoded by a reduced set of reference vectors (the codebook). VQ is useful in compressing data that arises in wide range applications and it can achieve better compression performance than any conventional coding technique which is based on the encoding of scalar quantities. This paper will present a multi-spectral image compression method using Vector Quantization technique based on Fuzzy c-Means (FCM). In this propose method we first use FCM algorithm to generate a good initial codebook. In this process we use fast version FCM (FFCM) to reduce the time spend to generate the initial codebook, Then use this initial codebook to design the VQ codebook for multi-spectral image compression. Experimental results are shown to illustrate the performance of the proposed compression method.

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ประสบความสำเร็จลุล่วงได้เป็นอย่างดี ด้วยคำแนะนำและคำปรึกษาจากท่าน รศ.ดร. ยุทธพงษ์ รั้งสรรเสรี ซึ่งเป็นอาจารย์ควบคุมวิทยานิพนธ์ ผู้วิจัยรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอกราบขอบพระคุณเป็นอย่างสูง

ขอบคุณเพื่อนๆ นักศึกษาทุกคนที่ช่วยเหลือให้คำแนะนำต่างๆ ให้การทำวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมีจากวิทยานิพนธ์ฉบับนี้ ผู้วิจัยขอมอบแด่ผู้มีพระคุณทุกท่าน

อุทัย แสงทองพราว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	VII
สารบัญรูป.....	VIII
บทที่ 1 บทนำ.....	1
1.1 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.2 การศึกษาด้านอิมเมจโปรเซสซิ่ง.....	1
1.3 ความรู้เบื้องต้นของการวิเคราะห์ภาพสี.....	2
1.3.1 ข้อมูลภาพดิจิทัล.....	2
1.3.2 ระบบโคออดิเนทของสี.....	4
บทที่ 2 การบีบอัดข้อมูลภาพหลายแบนด์.....	8
2.1 การบีบอัดข้อมูลภาพ.....	8
2.2 การจัดระดับสัญญาณหรือการควอนไทเซชัน.....	9
2.2.1 การควอนไทซ์สัญญาณแบบสเกลลาร์.....	9
2.2.2 การควอนไทซ์สัญญาณแบบเวกเตอร์.....	14
2.3 การประยุกต์ใช้งานเวกเตอร์ควอนไทเซชันในการบีบอัดข้อมูลภาพ.....	16
2.3.1 การฟอร์มเวกเตอร์จากข้อมูลภาพอินพุท.....	16
2.3.2 การออกแบบโค้ดบุค.....	17
2.3.3 การประมาณค่าเวกเตอร์อินพุท.....	19
2.3.4 การเข้ารหัสข้อมูลภาพ.....	19
2.4 การนำข้อมูลจากการวิเคราะห์คลัสเตอร์แบบฟัชซีซีมีนส์มาใช้ในการบีบอัดข้อมูลภาพ ด้วยวิธีเวกเตอร์ควอนไทเซชัน.....	20
2.5 การบีบอัดข้อมูลภาพหลายแบนด์ด้วยโค้ดบุคมาตรฐาน.....	21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อ IV ตรีศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.6 การตรวจสอบประสิทธิภาพของการบีบอัดข้อมูลภาพ	21
2.6.1 ค่า bits per pixel.....	21
2.6.2 ค่า PSNR.....	22
บทที่ 3 การวิเคราะห์คลัสเตอร์ (Cluster Analysis)	23
3.1 ทฤษฎีและความเป็นมาของการวิเคราะห์คลัสเตอร์.....	23
3.2 การคลัสเตอร์เซตของข้อมูล	23
3.3 การวิเคราะห์คลัสเตอร์ข้อมูลภาพด้วยวิธีฟัชชีมีนีส	25
3.4 การเพิ่มความเร็วของกระบวนการฟัชชีมีนีส	27
บทที่ 4 การทดลองและผลการทดลอง.....	32
4.1 ผลการทดลองบีบอัดข้อมูลภาพหลายแบนด์โดยวิธีการเวกเตอร์ควอนไทเซชัน ที่ใช้หลักการ FCM คลัสเตอร์ริง	32
4.1.1 ผลการทดลองการบีบอัดข้อมูลด้วยวิธีเวกเตอร์ควอนไทเซชันโดยใช้วิธีการ ฟอร์มเวกเตอร์แบบอินเตอร์แบนด์	32
4.1.2 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันโดยใช้วิธีการ ฟอร์มเวกเตอร์แบบอินทราแบนด์	34
4.2 ผลการทดลองการเพิ่มความเร็วของกระบวนการฟัชชีมีนีส	38
4.3 ผลการทดลองบีบอัดข้อมูลภาพถ่ายดาวเทียม ops ด้วยโค้ดบุคมาตรฐาน	39
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ	44
5.1 สรุปผลการวิจัย	44
5.2 ข้อเสนอแนะ.....	45
บรรณานุกรม.....	46

สารบัญ(ต่อ)

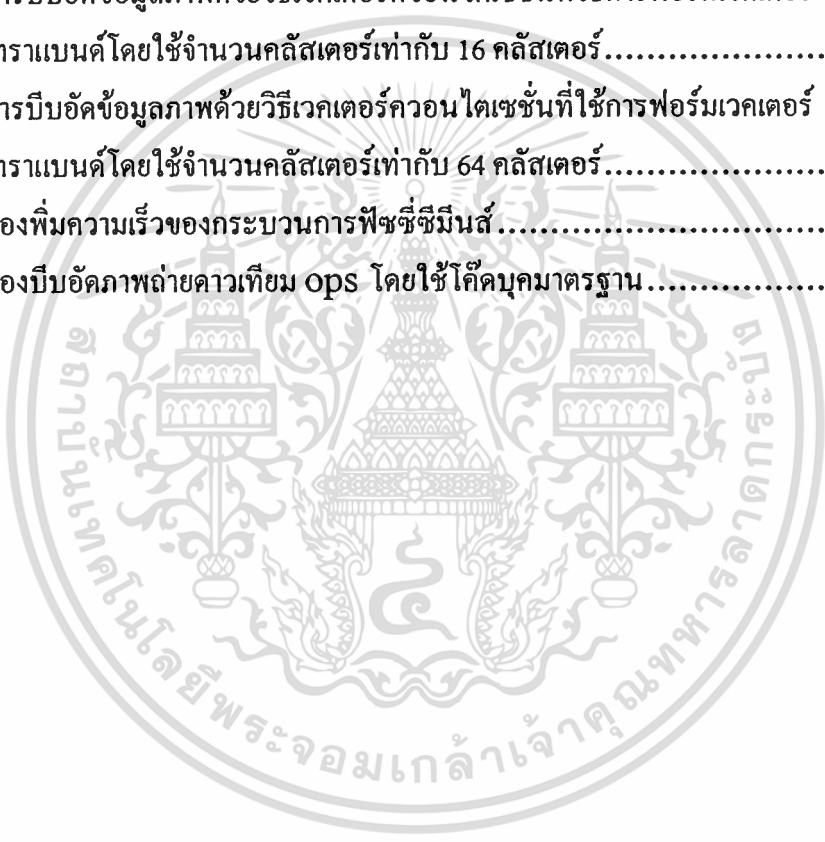
หน้า

ภาคผนวก.....	47
Multispectral Image Compression using FCM-Based Vector Quatization..	48
Intra/Inter-Band Coding in FCM-VQ of Multispectral Images	53
Modified Fuzzy C-Mean for Clustering of Multi-spectral Image	58
การหาขอบเขตของวัตถุในภาพสี โดยอาศัยการจำแนกกลุ่มข้อมูล.....	62
ประวัติผู้เขียน	67



สารบัญตาราง

ตารางที่	หน้า
4.1 คุณสมบัติการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์	32
4.2 คุณสมบัติการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์	34
4.3 คุณสมบัติการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์	34
4.4 คุณสมบัติการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์	34
4.5 ผลการทดลองเพิ่มความเร็วยของกระบวนการพีซีซีมีนส์	39
4.6 ผลการทดลองบีบอัดภาพถ่ายดาวเทียม ops โดยใช้โค้ดบุคมาตรฐาน	40



สารบัญรูป

รูปที่	หน้า
1.1 แสงที่สะท้อนออกจากวัตถุ	2
1.2 โมเดลของระบบกำเนิดข้อมูลภาพดิจิทัล	2
1.3 ฟังก์ชันขั้นบันไดที่ใช้ในการควอนไทซ์สัญญาณภาพดิจิทัล	4
1.4 ระบบโคออดิเนตสี RGB	6
2.1 การควอนไทซ์แบบสเตลลาร์	10
2.2 ควอนไทเซอร์เชิงเส้นขั้นบันได.....	10
2.3 ควอนไทเซอร์ขั้นบันไดแบบไม่เป็นเชิงเส้น	11
2.4 ควอนไทเซอร์โคเด็ก	12
2.5 การแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัลด้วย PCM.....	13
2.6 ความเพี้ยนที่เกิดจากการะบวนการควอนไทเซชัน	14
2.7 บล็อกไดอะแกรมของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชัน	16
2.8 การฟอร์มเวกเตอร์แบบอินทราแบนด์	18
2.9 การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์	18
2.10 การคำนวณค่าเวกเตอร์อ้างอิงจากการวิเคราะห์คลัสเตอร์	19
2.11 การเข้ารหัสและถอดรหัส	20
3.1 โฟล์วไดอะแกรมของการวิเคราะห์คลัสเตอร์.....	24
3.2 การวิเคราะห์คลัสเตอร์ข้อมูลใน 2 มิติ	25
3.3 โฟล์วไดอะแกรมของ FCM.....	28
3.4 โฟล์วไดอะแกรมของการปรับปรุงกระบวนการ FCM	31
4.1 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบ อินเตอร์แบนด์.....	33
4.2 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบ อินทราแบนด์จำนวนอิลิเมนต์ของเวกเตอร์เท่ากับ 4	35
4.3 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบ อินทราแบนด์จำนวนอิลิเมนต์ของเวกเตอร์เท่ากับ 3	36

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.4 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบ อินทราแบนด์จำนวนอิลิเมนต์ของเวกเตอร์เท่ากับ 2	37
4.5 ภาพต้นแบบที่นำมาใช้ในการทดลองเพิ่มความเร็วของกระบวนการ FCM.....	38
4.6 ภาพที่ได้จากการบีบอัดข้อมูลภาพถ่ายดาวเทียม ops 10 ภาพโดยใช้โค้ดบุคมาตรฐาน	41



บทที่ 1

บทนำ

1.1 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา

เนื่องจากในปัจจุบันนั้นการสื่อสารข้อมูลในรูปแบบต่างๆมีบทบาทเป็นอย่างมากในชีวิตปัจจุบัน ข้อมูลภาพเป็นสื่อประเภทหนึ่งที่มนุษย์เรานิยมนำมาใช้ในการสื่อสาร ด้วยเหตุนี้จึงได้มีการศึกษาวิจัยกระบวนการต่างๆที่จะนำมาใช้กับข้อมูลภาพเพื่อให้การสื่อสารข้อมูลเป็นไปได้อย่างรวดเร็วและมีประสิทธิภาพสูงสุด งานวิจัยที่นำมาเสนอภายในวิทยานิพนธ์ฉบับนี้จะเป็นการนำเสนอวิธีการในการบีบอัดข้อมูลภาพให้มีขนาดเล็กเพื่อที่จะเป็นการลดปริมาณหน่วยความจำที่นำมาใช้ในการเก็บข้อมูลภาพนั้นๆ นอกจากนี้ภายในวิทยานิพนธ์เล่มนี้ยังจะนำเสนอวิธีการในการเพิ่มความเร็วจำแนกกลุ่มข้อมูลภาพด้วยวิธีพีซีซีเอ็มด้วย

1.2 การศึกษาด้านอิมเมจโปรเซสซิ่ง

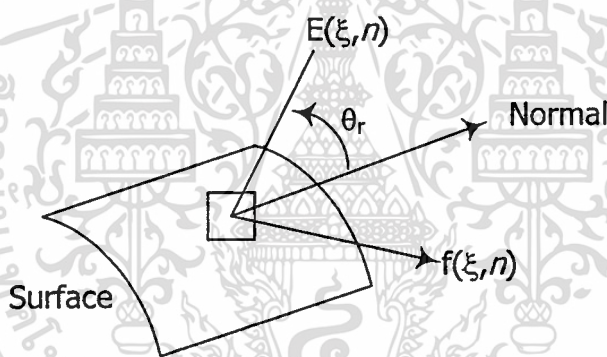
ข้อมูลภาพดิจิทัล (Digital Image) เป็นภาพสองมิติซึ่งมนุษย์เรานำมาใช้แทนสิ่งต่างๆที่เรามองเห็น ปัจจุบันภาพดิจิทัลนั้นถูกนำมาศึกษาและประยุกต์ใช้ในหลากหลายรูปแบบ รวมไปถึงทางด้านศิลปะ, การศึกษาด้านการมองเห็นของมนุษย์, ดาราศาสตร์และทางด้านวิศวกรรม อิมเมจโปรเซสซิ่งเป็นศาสตร์หนึ่งที่มีความเกี่ยวข้องกับกระบวนการต่างๆที่นำมาใช้กับข้อมูลภาพเพื่อจุดประสงค์ใดจุดประสงค์หนึ่ง เช่น การแปลงข้อมูลภาพให้อยู่ในรูปแบบต่างๆซึ่งทำให้เราสามารถส่งข้อมูลภาพเหล่านี้ผ่านไปยังบิตกลางหรือหน่วยความจำของคอมพิวเตอร์ได้โดยง่าย หรืออาจเป็นการลดสัญญาณรบกวน (noise) ภายในข้อมูลภาพหรือ เป็นการจับเอาเฉพาะข้อมูลที่น่าสนใจภายในภาพ กระบวนการทางอิมเมจโปรเซสซิ่งทั่วไปได้แก่ การฟิลเตอร์ลิ่ง, การแซมปลิ่ง, การสเกลลิ่ง, การไค้คดิ่งหรือการเข้ารหัส, การจำแนกข้อมูลภาพ (segmentation), การบีบอัดข้อมูลภาพ (Image Compression) และกระบวนการอื่นๆอีกมากมาย ซึ่งผู้อ่านสามารถศึกษาได้จากหนังสือของ (Pearson. 1991 ; Pitas.1993 ; Clarke. 1995) สำหรับวิทยานิพนธ์ฉบับนี้จะกล่าวเน้นในเรื่องการจำแนกข้อมูลภาพและการบีบอัดข้อมูลภาพ หลายแบบ

ภายในบทนี้จะกล่าวถึงความรู้เบื้องต้นสำหรับการศึกษาด้านอิมเมจโปรเซสซิ่งซึ่งจะนำมาใช้เป็นแนวทางในการศึกษาวิทยานิพนธ์เล่มนี้เพื่อให้ผู้อ่านมองเห็นภาพและเข้าใจได้ง่ายขึ้น

1.3 ความรู้เบื้องต้นของการวิเคราะห์ภาพสี (Fundamentals of color image processing)

1.3.1 ข้อมูลภาพดิจิทัล

อย่างที่ทราบกันแล้วว่ามนุษย์เราสามารถมองเห็นสิ่งต่างๆ ได้จากการที่แสงที่สะท้อนออกมาจากวัตถุมาตกกระทบเข้าที่ดวงตาของเรา "ภาพ" หรือ "อิมเมจ" ก็คือการแทนการมองเห็นของมนุษย์โดยอาศัยความเข้มของแสงที่แผ่กระจายมาจากวัตถุหรือต้นกำเนิดแสง ดังนั้นองค์ประกอบต่างๆ ที่ถูกนำมาใช้ในกระบวนการสร้างข้อมูลภาพก็คือ วัตถุ, ต้นกำเนิดแสงและระบบผลิตข้อมูลภาพ (image formation system) รูปที่ 1.1 แสดงแสงที่สะท้อนออกมาจากวัตถุ $f(\xi, \eta)$ ซึ่งเป็นอินพุทของระบบสร้างข้อมูลภาพดิจิทัล ระบบสร้างข้อมูลภาพดิจิทัลนั้นมักจะประกอบด้วย ออปติคอลเลนส์, ออปติคอลเซ็นเซอร์ และอิมเมจดิจิทัลไคเซอร์ โดยโมเดลของระบบสร้างข้อมูลภาพดิจิทัลแสดงได้ดังรูปที่ 1.2



รูปที่ 1.1 แสงที่สะท้อนออกมาจากวัตถุ



รูปที่ 1.2 โมเดลของระบบกำเนิดข้อมูลภาพดิจิทัล

ในบล็อกของ Optical System H นั้นสามารถถูกมองได้ว่าเป็นระบบชีพสัญญาณแบบเชิงเส้นซึ่งมีการตอบสนองอิมพัลส์แบบสองมิติ $h(x,y)$ ระบบย่อย H นี้มักจะเป็นระบบกรองความถี่ต่ำ ซึ่งจะกำจัดองค์ประกอบความถี่สูงของข้อมูลภาพอินพุท $f(\xi, \eta)$ ดังนั้นข้อมูลภาพเอาต์พุทที่ได้ $b(x,y)$ จึงมักจะมีลักษณะเบลอเหมือนภาพที่ไม่ได้ทำการปรับโฟกัส เนื่องจากสัญญาณ $f(\xi, \eta)$ และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$b(x,y)$ เป็นการแทนความเข้มของแสงที่เราสามารถมองเห็น ค่าของสัญญาณทั้งสองนี้จึงไม่สามารถมีค่าติดลบได้

$$f(\xi, n) \geq 0 \quad (1.1)$$

$$b(x, y) \geq 0 \quad (1.2)$$

โมเดลทางคณิตศาสตร์ของอุปกรณ์เซ็นเซอร์ซึ่งจะขึ้นอยู่กับโฟโตอิเล็กทริกเซ็นเซอร์ ซึ่งอุปกรณ์เซ็นเซอร์นั้นมีมากมายหลากหลายแบบ ชุดอุปกรณ์เซ็นเซอร์พื้นฐานที่มักใช้กันอยู่ทั่วไปก็คือ อุปกรณ์ CIDs (charge injector devices) และอุปกรณ์ CCDs (charge coupled devices) แต่ไม่ว่าจะในกรณีใดก็ตามความสัมพันธ์ระหว่างสัญญาณภาพอินพุต $b(x,y)$ และกระแสไฟฟ้าเอาต์พุต $i(x,y)$ จะเป็นแบบไม่เป็นเชิงเส้น

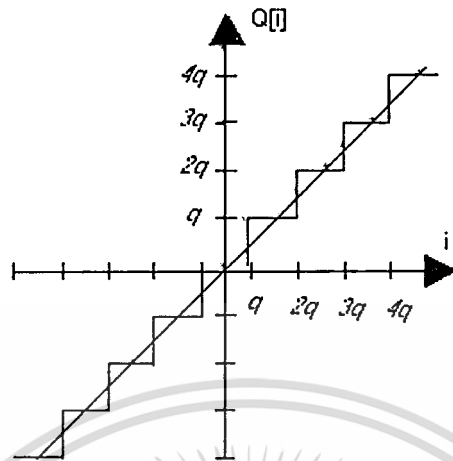
สัญญาณเอาต์พุตของอุปกรณ์เซ็นเซอร์ $i(x,y)$ ยังคงเป็นสัญญาณอนาล็อกในสองมิติ สัญญาณนี้จะถูกนำไปทำการดิจิไตซ์ก่อนที่มันจะถูกนำมาใช้ในกระบวนการต่างๆ โดยคอมพิวเตอร์ การแซมปลิงหรือการดิจิไตซ์จะถูกทำโดยอุปกรณ์แปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิตอล (A/D Converter) ซึ่งจะทำให้การแปลงภาพอนาล็อก $i(x,y)$ ให้เป็นภาพดิจิตอล $i(n_1, n_2)$, $n_1=1,2,3,\dots,N$, $n_2=1,2,3,\dots,M$:

$$i(n_1, n_2) = i(n_1, T_1, n_2, T_2) \quad (1.3)$$

การแซมปลิงตามสมการที่ (1.3) สามารถแสดงได้โดยใช้กริดขั้นบันไดซึ่งมีระดับสัญญาณแซมปลิง เป็น T_1, T_2 เมื่อข้อมูลภาพมีขนาด $N \times M$ พิกเซล โดยขนาดภาพที่นิยมนำมาใช้ในการทดลองคือ 256x256 พิกเซลหรือ 512x512 พิกเซล ในกรณีของภาพสี การแซมปลิงจะถูกทำที่ทุกๆ แชนแนลของสี (แดง, เขียว, น้ำเงิน) อย่างเป็นทางการนั้นจึงเป็นการสร้างภาพดิจิตอลที่มีขนาดเท่าๆกัน ขึ้นมาพร้อมๆกัน 3 ภาพ

นอกจากนี้ตัวแปลงสัญญาณอนาล็อกเป็นดิจิตอลจะเป็นตัวที่ทำการควอนไตซ์แต่ละแซมเปิ้ลภายในภาพอีกด้วย ถ้า q เป็นระยะห่างของการควอนไตซ์ ภาพที่ถูกควอนไตซ์จะสามารถมีความเข้มของแสงเป็น kq , $k=0, 1, 2, \dots$ ดังแสดงได้โดยรูปที่ 1.3 ถ้าแต่ละพิกเซลของภาพสามารถแทนด้วยสัญญาณดิจิตอลขนาด b บิต ระยะห่างของระดับสัญญาณควอนไตซ์จะเขียนได้โดยสมการ

$$q = \frac{1}{2^b} \quad (1.4)$$



รูปที่ 1.3 ฟังก์ชันขั้นบันไดที่ใช้ในการควอนไทซ์สัญญาณภาพดิจิทัล

ในแอปพลิเคชันต่างๆ ภาพขาวดำ (grayscale image) จะถูกควอนไทซ์ออกเป็น 256 ระดับและจำเป็นต้องใช้ข้อมูลขนาด 1 ไบต์ (8 บิต) ในการแทนค่าของสัญญาณแต่ละพิกเซล แต่สำหรับภาพแบบไบนารี ภาพจะถูกควอนไทซ์ออกเป็น 2 ระดับสัญญาณเท่านั้นคือ 0 กับ 1 ดังนั้นจึงสามารถแทนค่าแต่ละพิกเซลได้ด้วยข้อมูลดิจิทัลขนาดเพียง 1 บิต อย่างไรก็ตามในเรื่องของการควอนไทซ์ข้อมูลภาพดิจิทัลจะถูกกล่าวถึงโดยละเอียดในบทที่ 2

1.3.2 ระบบโคออดิเนตของสี

สายตาของมนุษย์นั้นจะแลเห็นและเข้าใจภาพสีได้ดีกว่าภาพขาวดำ ภาพสีหรืออนุกรมของภาพสี (วีดีโอ) การสร้างภาพสีนั้นใช้หลักการผสมสีของแม่สี 3 สี C_1, C_2, C_3 ในอัตราส่วนที่เหมาะสม

$$C = aC_1 + bC_2 + cC_3 \quad (1.4)$$

โดยทฤษฎีนี้ใช้หลักความจริงที่ว่าดวงตามนุษย์มีอวัยวะรูปทรงกรวยซึ่งเราเรียกว่า "โคน (Cones)" ที่แตกต่างกันอยู่ 3 ชนิดในเรติน่า ซึ่งอวัยวะทั้งสามนี้จะตอบสนองที่สี เหลือง-เขียว, เขียว และน้ำเงิน จากสเปกตรัมของแสงที่สามารถมองเห็นได้ จากสมการที่ (1.1) ทุกพิกเซลภายในภาพสีสามารถนำมาแทนได้ด้วยเวกเตอร์ $[a,b,c]$ ในสามโดเมนขั้นสเปซ (C_1, C_2, C_3) ดังนั้นภายใต้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในวิทยานิพนธ์ฉบับนี้จะขอละค่าสัมประสิทธิ์ a, b, c เอาไว้และใช้สัญลักษณ์ C_1, C_2, C_3 แทนค่าสีในแต่ละโดเมนชั้นสเปซ ระดับสีที่ได้จากการผสมสีทั้งสามสามารถนิยามได้โดยอัตราส่วนดังสมการที่ (1.2)

$$c_i = \frac{C_i}{C_1 + C_2 + C_3} \quad (1.2)$$

จะเห็นได้ชัดว่าสองโคออดิเนทของสีเท่านั้นที่จะมีความเป็นอิสระต่อกัน เช่น c_1, c_2 เพราะ

$$c_1 + c_2 + c_3 = 1 \quad (1.3)$$

ระบบโคออดิเนทของสีในแบบต่างๆ ได้ถูกนำเสนอขึ้นมาใช้มากมายในอดีต ต่อมา CIE (Commission Internationale de l'Eclairage) ได้เสนอระบบโคออดิเนทพื้นฐาน RGB ขึ้นมาใช้ซึ่งอ้างอิงจากสีพื้นฐาน R_{CIE} (สีแดง 700 nm), G_{CIE} (สีเขียว 546.1 nm) และ B_{CIE} (สีน้ำเงิน 435.8 nm) โดยสำหรับสีขาวที่ใช้อ้างอิงนั้นจะมีค่า $R_{CIE} = G_{CIE} = B_{CIE} = 1$. แต่อย่างไรก็ตามระบบบสเป็คตรัมสีพื้นฐาน RGB ของ CIE ไม่สามารถสร้างหรือแสดงสีทั้งหมดได้ ดังนั้น CIE จึงได้เสนอระบบพื้นฐาน XYZ ซึ่งเป็นโคออดิเนชัน X, Y, Z ที่ถูกสมมุติขึ้นและมีความสัมพันธ์แบบเชิงเส้นต่อระบบโคออดิเนท RGB ดังนี้

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.490 & 0.310 & 0.200 \\ 0.177 & 0.813 & 0.011 \\ 0.000 & 0.010 & 0.990 \end{bmatrix} \cdot \begin{bmatrix} R_{CIE} \\ G_{CIE} \\ B_{CIE} \end{bmatrix} \quad (1.4)$$

สำหรับสีขาวจะแทนค่า $X = Y = Z = 1$. ทำให้โคออดิเนทของสีมีค่าเป็น

$$x = \frac{X}{X+Y+Z} \quad (1.5)$$

$$y = \frac{Y}{X+Y+Z} \quad (1.6)$$

จากระบบโคออดิเนทของสีดังกล่าวมาข้างต้น ได้มีการนำเสนอโมเดลของสีในแบบต่างๆ ขึ้นเพื่อความสะดวกในการแสดงภาพ โดยอุปกรณ์ที่เป็นฮาร์ดแวร์ เช่นระบบสี RGB ได้ถูกนำเสนอ

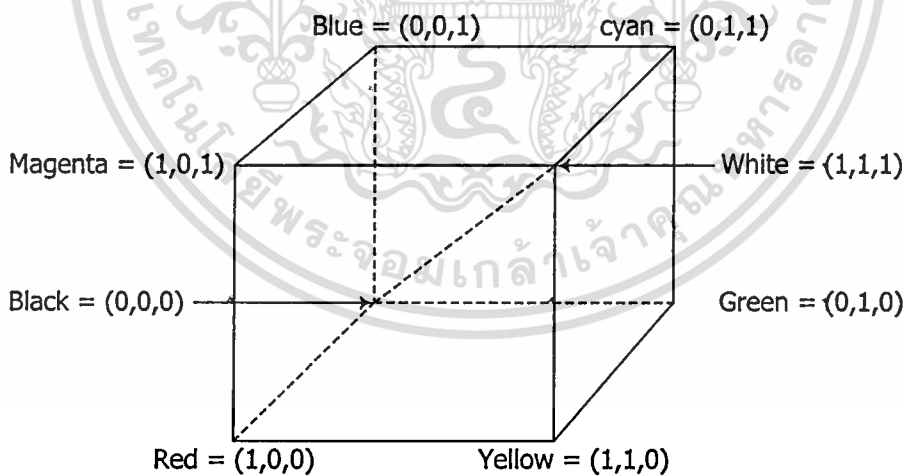
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขึ้นมาโดย National Television Systems Committee (NTSC) เพื่อนำมาใช้ในการแสดงภาพบนจอภาพ CRT, โมเดลของสีแบบ CMY (Cyan Magenta Yellow) ได้ถูกนำมาใช้ในการพิมพ์ภาพสี และสุดท้ายระบบการส่งภาพ NTSC (YIQ) ก็ได้ถูกนำมาใช้ในการส่งข้อมูลภาพซึ่ง ระบบ NTSC RGB เป็นโมเดลของสีใน 3 ระนาบดังแสดงในรูปที่ 1.1 ซึ่งมีความแตกต่างจากระบบสเป็คตรัมพื้นฐานของสี CIE RGB แต่แม้กระนั้น โมเดลทั้งสองนี้ก็ยังคงมีความสัมพันธ์แบบเชิงเส้นต่อกัน ดังนี้

$$\begin{bmatrix} R_{CIE} \\ G_{CIE} \\ B_{CIE} \end{bmatrix} = \begin{bmatrix} 1.167 & -0.146 & -0.151 \\ 0.114 & 0.753 & 0.159 \\ -0.001 & 0.059 & 1.128 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1.7)$$

ดังนั้น ระบบ XYZ จึงมีความสัมพันธ์กับระบบ NTSC RGB แบบเป็นเชิงเส้นด้วยอันสืบเนื่องมาจากสมการที่ 1.4 และ 1.7

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.607 & 0.174 & 0.201 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.117 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1.8)$$



รูปที่ 1.4 ระบบโคออดิเนตสี RGB

จากรูปที่ 1.4 จะเห็นได้ว่าสีเขียวน้ำเงิน(Cyan), สีม่วงแดง(Magenta) และสีเหลือง เป็นสีที่มองเห็นประกอบมาจากสี แดง, เขียว และน้ำเงิน ตามลำดับ ในระบบโมเดล CMY สีต่างๆภายในแต่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ลักษณะของภาพจะเกิดขึ้นจากการหักลบสีออกจากแสงสีขาว จึงเป็นการง่ายที่จะสร้างโคออดิเนท CMY จากระบบ RGB

$$C = 1 - R \quad (1.9)$$

$$M = 1 - G \quad (1.10)$$

$$Y = 1 - B \quad (1.11)$$

ระบบสีแบบ CMY นี้ นิยมนำมาใช้ในการพิมพ์ภาพสี แต่เนื่องจากความไม่สมบูรณ์ของหมึกสีทำให้ระบบนี้ไม่สามารถสร้างสีดำได้อย่างถูกต้อง จึงต้องมีการเพิ่มหมึกสีดำเข้าไปในระบบการพิมพ์ภาพสี ทำให้ระบบนี้ถูกเรียกว่าแทนว่าระบบ CMYK

$$K = \min(C, M, Y) \quad (1.12)$$

$$C = C - K \quad (1.13)$$

$$M = M - K \quad (1.14)$$

$$Y = Y - K \quad (1.15)$$

องค์ประกอบ K ถูกนำมาใช้แทนสีดำ ทำให้องค์ประกอบสีทั้งหมดเป็น 4 สี

โมเดลสีแบบ YIQ ถูกนำมาใช้กับการกระจายเสียงสัญญาณ NTSC TV จุดเด่นของระบบนี้ก็คือสามารถใช้กับ TV ขาวดำได้ การแปลงระบบสี RGB ให้เป็น YIQ สามารถทำได้โดยสมการที่ (1.16)

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1.16)$$

บทที่ 2

การบีบอัดภาพหลายแบนด์

2.1 การบีบอัดข้อมูลภาพ (Image Compression)

การบีบอัดและการเข้ารหัสข้อมูลภาพ (Image Compression) นั้นเป็นกระบวนการในการลดขนาดเนื้อที่หน่วยความจำที่ใช้ในการเก็บข้อมูลภาพ ซึ่งเป็นที่รู้กันดีว่า ไฟล์ภาพที่ได้มาโดยทั่วไปนั้น (raw image) มักจะมีขนาดใหญ่และจำเป็นต้องใช้พื้นที่หน่วยความจำเป็นจำนวนมากเพื่อที่จะจัดเก็บข้อมูลภาพเหล่านี้ ยกตัวอย่างเช่น ภาพสีขนาด 1024×1024 จำเป็นต้องใช้พื้นที่ของหน่วยความจำถึง 3 เมกกะไบต์ในการจัดเก็บ ปริมาณหน่วยความจำจำนวนมากที่ต้องสูญเสียไปนี้จึงเป็นปัญหาหนึ่งที่นักวิจัยพยายามค้นหาวิธีการในการลดขนาดของไฟล์ภาพดิจิทัลที่ถูกนำมาใช้ในแอปพลิเคชันต่างๆ เช่น เอกสารออฟฟิศ, ภาพถ่ายทางการแพทย์, ภาพถ่ายวิดีโอ และภาพถ่ายดาวเทียม กระบวนการการบีบอัดข้อมูลภาพที่สามารถเข้ารหัสและถอดรหัสภาพได้อย่างรวดเร็วก็เป็นจุดหนึ่งที่น่าสนใจเป็นอย่างมาก เนื่องจากการสื่อสารข้อมูลภาพดิจิทัลก็เป็นแอปพลิเคชันหนึ่งที่ต้องใช้เทคนิคของการเข้ารหัสและถอดรหัสเข้ามาช่วยเพื่อประหยัดเวลาที่ใช้ในการสื่อสารข้อมูลภาพ นอกจากนี้เทคนิคการบีบอัดข้อมูลภาพยังได้ถูกนำมาใช้ในการส่งแพ็คเกจ, การประชุมทางวิดีโอ (Video Conference), การสื่อสารข้อมูลภาพทางการแพทย์ ดังนั้นกระบวนการเข้ารหัสและถอดรหัสที่รวดเร็วจึงยังมีความจำเป็นเป็นอย่างมากในแอปพลิเคชันที่มีคุณสมบัติเป็นแบบเรียลไทม์ (real time application) และเนื่องมาจากช่องการสื่อสารในเกือบทุกรูปแบบมักมีเรื่องของสัญญาณรบกวนเข้ามาเกี่ยวข้อง เทคนิคการบีบอัดหรือเข้ารหัสข้อมูลภาพที่มีความต้านทานต่อสัญญาณรบกวนจึงเป็นอีกเรื่องหนึ่งที่สำคัญด้วยเช่นกัน

การบีบอัดข้อมูลภาพดิจิทัลนั้นสามารถแบ่งออกได้เป็น 2 แบบใหญ่ๆ ได้แก่การบีบอัดข้อมูลภาพแบบปราศจากการสูญเสียของข้อมูลภายในภาพ (lossless compression) และแบบที่มีการสูญเสียของข้อมูลภาพไปบางส่วน (lossy compression) โดยในแบบแรกนั้นจะถูกนำมาใช้ในกรณีที่ข้อมูลภาพที่มีอยู่นั้นยากต่อการได้มาและเป็นข้อมูลที่สำคัญและอาจจะถูกทำลายโดยกระบวนการบีบอัดข้อมูลภาพได้เช่นข้อมูลภาพถ่ายทางการแพทย์ ส่วนการบีบอัดข้อมูลภาพที่มีการสูญเสียของข้อมูลภาพไปบางส่วนนั้นจะถูกนำมาใช้ในกรณีที่ภาพที่ได้มานั้นสามารถที่จะสร้างขึ้นใหม่ได้โดยง่ายหรือในกรณีที่ผู้รับข้อมูลสามารถยอมรับต่อปริมาณข้อมูลภาพที่สูญเสียไปได้ยกตัวอย่างเช่นดิจิทัลทีวีหรือการประชุมทางวิดีโอซึ่งสิ่งที่รับข้อมูลภาพวิดีโอนี้ก็คือดวงตาของมนุษย์ซึ่งสามารถยอมรับการสูญเสียของข้อมูลภาพไปบางส่วนได้

การบีบอัดข้อมูลภาพนั้นสามารถทำได้หลายวิธี สำหรับวิทยานิพนธ์นี้จะกล่าวถึงการบีบอัดข้อมูลภาพ โดยใช้วิธีเวกเตอร์ควอนไทเซชัน

2.2 การจัดระดับสัญญาณหรือการควอนไทเซชัน (Quantization)

ปัจจุบันการสื่อสารข้อมูลมีบทบาทในชีวิตประจำวันของมนุษย์เป็นอย่างมาก การเพิ่มความเร็วของระบบการสื่อสารข้อมูลจึงเป็นเรื่องหนึ่งที่มีความสำคัญมาก เทคนิคหนึ่งที่ถูกนำมาใช้ในการเพิ่มความเร็วในการสื่อสารข้อมูลไม่ว่าจะเป็นข้อมูลเสียงหรือภาพก็คือการลดขนาดของข้อมูลต้นแบบให้มีขนาดเล็กลง การควอนไทเซชันเป็นเทคนิคหนึ่งซึ่งถูกนำมาใช้ในการลดขนาดข้อมูลต้นแบบ โดยการควอนไทเซชันนั้นจะใช้หลักการของการจัดระดับสัญญาณหรือข้อมูลอินพุทใหม่เนื่องจากจำนวนระดับของข้อมูลอินพุทหรือข้อมูลต้นแบบที่เราต้องการส่งนั้นประกอบด้วยระดับสัญญาณต่างๆมากมาย ทำให้เราต้องใช้จำนวนบิตของสัญญาณดิจิทัลอย่างมากในการส่งสัญญาณข้อมูลแต่ละแซมเปิล การจัดระดับสัญญาณข้อมูลอินพุทใหม่หรือการควอนไทเซชันจึงถูกนำมาใช้ในการลดจำนวนบิตที่ใช้แทนข้อมูลแต่ละแซมเปิล โดยตัวควอนไทเซอร์จะทำการจัดแบ่งระดับสัญญาณของข้อมูลอินพุทใหม่ให้มีระดับสัญญาณจำกัดอยู่ที่ 2^n ระดับ ทำให้เราสามารถแทนข้อมูลในแต่ละแซมเปิลของสัญญาณอินพุทด้วยสัญญาณดิจิทัลขนาด n บิต

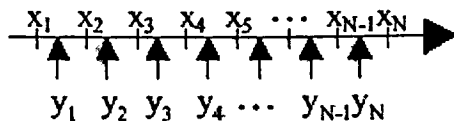
การควอนไทซ์สัญญาณหรือข้อมูลต้นแบบนั้นสามารถแบ่งออกได้เป็นสองแบบตามลักษณะของสัญญาณอินพุทคือ การควอนไทซ์สัญญาณสเกลลาร์ และการควอนไทซ์สัญญาณเวกเตอร์

2.2.1 การควอนไทซ์สัญญาณสเกลลาร์ (Scalar Quantization)

การควอนไทซ์สัญญาณสเกลลาร์ถูกนำมาใช้ในการจัดระดับสัญญาณอินพุทแบบสเกลลาร์ โดยการควอนไทซ์สัญญาณสเกลลาร์เป็นการแมปค่าสัญญาณอินพุท x ให้เป็นสัญญาณ y ซึ่งอยู่ในเซตของสัญญาณที่แน่นอน

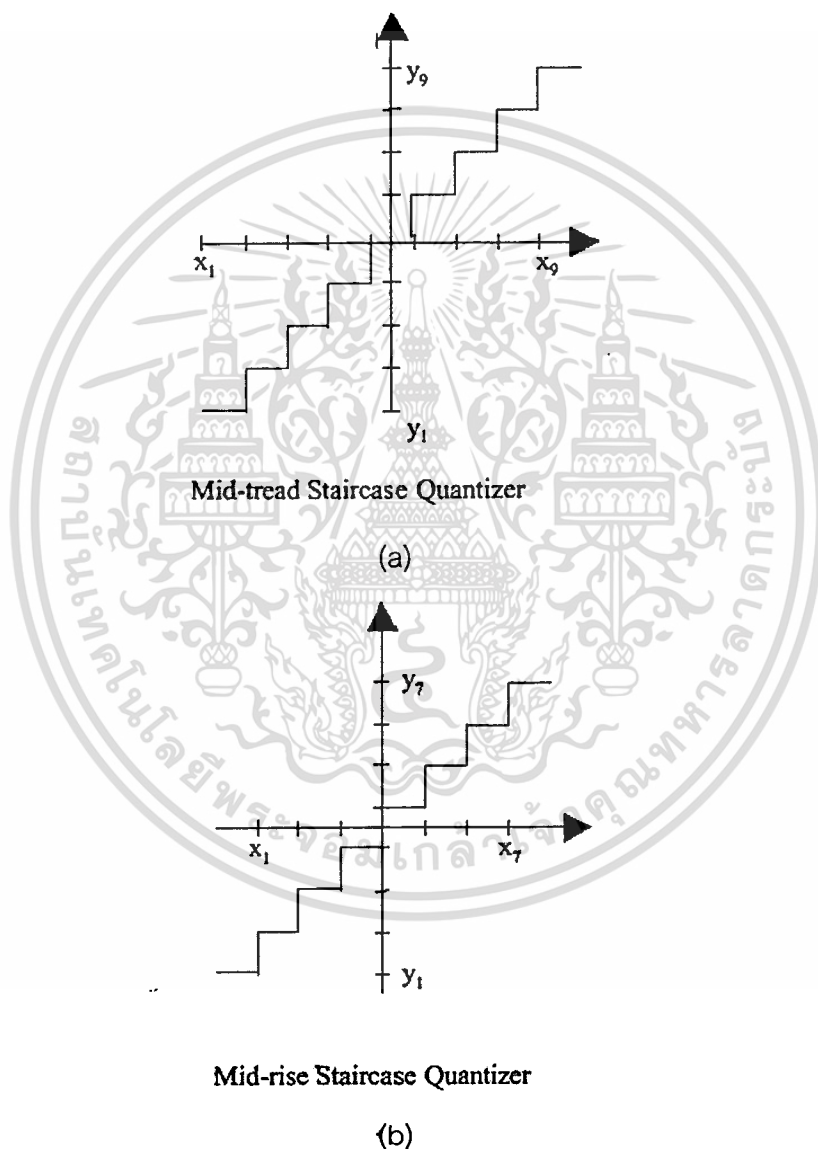
$$Q : x \rightarrow y$$

การควอนไทซ์แบบสเกลลาร์แสดงได้โดยการแบ่งจำนวนบนแกนจำนวนจริงออกเป็นเซลล์ย่อยๆ ดังรูปที่ 2.1 กำแพงที่กั้นระหว่างเซลล์จะเป็นตัวอ้างอิงในการตัดสินใจว่าระดับสัญญาณอินพุทควรถูกจัดอยู่ในเซลล์ใด เช่นหากระดับสัญญาณอินพุท x ตกอยู่ในเซลล์ที่ i เอาท์พุทที่ถูกเลือกใช้แทนสัญญาณ x ก็คือสัญญาณ y_i



รูปที่ 2.1 การควอนไทซ์ข้อมูล

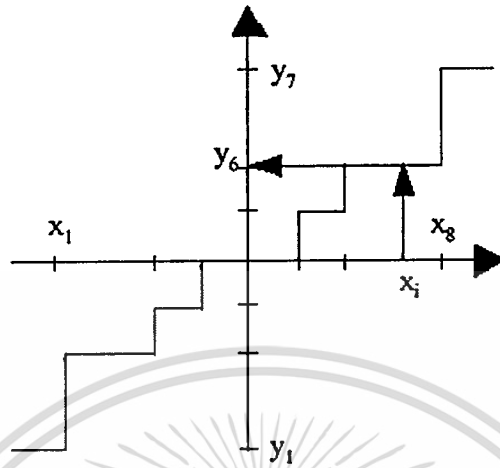
กราฟขั้นบันไดเชิงเส้นสองมิติสามารถนำมาใช้อธิบายการทำงานของตัวควอนไทเซอร์ดังรูปที่ 2.2



รูปที่ 2.2 ควอนไทเซอร์เชิงเส้นขั้นบันได

จากรูปที่ 2.2 แกน x แทนระดับของสัญญาณอินพุตและแกน y แทนระดับสัญญาณเอาต์พุต
 เส้นแบ่งบนแกน x เป็นเส้นอ้างอิงที่ใช้ในการจัดระดับสัญญาณอินพุต ในขณะที่เส้นแบ่งบนแกน y
 เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นระดับของสัญญาณเอทพุท โดยในรูปที่ 2.2 ทั้ง (a) และ (b) นั้นเป็นคอนโทรลเลอร์แบบเชิงเส้น ทั้งคู่ รูปที่ 3 แสดงตัวคอนโทรลเลอร์ขั้นบันไดแบบไม่เป็นเชิงเส้น

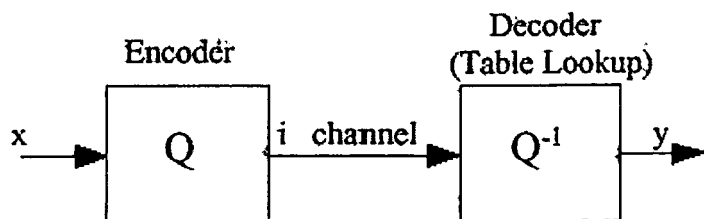


รูปที่ 2.3 คอนโทรลเลอร์ขั้นบันไดแบบไม่เป็นเชิงเส้น

สมมติตามคอนโทรลเลอร์ที่แสดงในรูปที่ 2.3 ถ้ามีสัญญาณอินพุท x ซึ่งมีระดับสัญญาณตกอยู่ระหว่าง x_7 และ x_8 ดังนั้นเอาต์พุทจะถูกแมปเป็นระดับสัญญาณ y_6 ดังแสดงในรูปที่ 3 ตัวดัชนี 6 จะถูกส่งไปแทนสัญญาณเอทพุท

ในอุดมคติ สัญญาณเอทพุทในแต่ละเซลล์ (หรือที่เราเรียกว่า "โค้ดเวิร์ด") จะเป็นค่าที่ดีที่สุดที่เซลล์นั้นๆ ซึ่งค่าที่ดีที่สุดในแต่ละเซลล์ก็คือค่ากึ่งกลางหรือเซ็นเตอร์ของแต่ละเซลล์นั่นเอง ค่ากึ่งกลางของแต่ละเซลล์นี้เราสามารถนิยามได้ว่าเป็นค่าเฉลี่ยของค่าอินพุททุกๆตัวซึ่งตกอยู่ในเซลล์นั้นๆ ดังนั้นการที่จะคำนวณค่าเฉลี่ยในแต่ละเซลล์ได้จำเป็นต้องกำหนดขอบเขตที่แน่นอนของแต่ละเซลล์ก่อน ถ้าเรารู้ขอบเขตของแต่ละเซลล์ก็จะสามารถคำนวณค่าเซ็นเตอร์ที่แน่นอนได้

เนื่องจากระดับสัญญาณเอทพุทมีจำนวนแน่นอน ดังนั้นจึงไม่จำเป็นต้องทำการส่งสัญญาณเอทพุทในระดับสัญญาณเอทพุท แต่เราสามารถทำการส่งค่าดัชนีที่ถูกกำหนดขึ้นแทนแต่ละระดับเอทพุทไปแทนได้ ที่ตัวรับสัญญาณจะรับค่าดัชนีเหล่านี้แล้วไปทำการแมปกลับเป็นระดับสัญญาณเอทพุทที่เหมาะสม โดยดูจากตารางที่เก็บการแมปค่าดัชนีไปเป็นระดับสัญญาณเอทพุท ระบบการเข้ารหัสและถอดรหัส (codec) สามารถแสดงได้ดังรูปที่ 2.4



รูปที่ 2.4 ควอนไทเซอร์โคเด็ก

แต่เนื่องจากสัญญาณอินพุทของควอนไทเซอร์โดยปกติมีได้อยู่ในช่วงที่จำกัด เซลล์ที่ไม่มีขอบเขตสูงสุดหรือขอบเขตต่ำสุดถูกเรียกว่า “โอเวอร์โหลดเซลล์ (overload cells)” โอเวอร์โหลดเซลล์คือเซลล์ที่ครอบคลุมค่าที่ต่ำที่สุดและค่าที่สูงที่สุดของสัญญาณอินพุท เป็นเซลล์ที่มีค่าเข้าใกล้ลบอินฟินิตี้และบวกอินฟินิตี้ ค่าสัญญาณรบกวนที่เกิดที่มาจากโอเวอร์โหลดเซลล์นี้ถูกเรียกว่าสัญญาณรบกวนโอเวอร์โหลด (overload noise) และในทางทฤษฎีค่าของสัญญาณรบกวนโอเวอร์โหลดสามารถมีค่าถึงอินฟินิตี้ได้ แต่ในทางปฏิบัติสัญญาณที่ถูกควอนไทซ์จะอยู่ในช่วงที่จำกัด แต่ก็มีความเป็นไปได้ที่จะทำให้สัญญาณรบกวนโอเวอร์โหลดมีค่าสูงมาก สำหรับเซลล์ที่มีขอบเขตแน่นอนถูกเรียกว่า “กรานูลาร์เซลล์ (granular cells)” สัญญาณรบกวน n ที่เกิดขึ้นภายในกรานูลาร์เซลล์ถูกเรียกว่าสัญญาณรบกวนกรานูลาร์ (granular noise) และค่าสัญญาณรบกวนกรานูลาร์ถูกจำกัดโดยขอบเขตของแต่ละเซลล์

$$n = x - y_i$$

เมื่อ x คือระดับสัญญาณอินพุทซึ่งตกอยู่ในเซลล์ที่ l และ y_i คือระดับสัญญาณเอาต์พุท สำหรับเซลล์ที่มีการกระจายของแอมพลิจูดสม่ำเสมอภายในเซลล์ ค่าสัญญาณรบกวน n_m จะมีค่าสูงสุดเท่ากับ

$$n_m = l/2$$

เมื่อ l คือความกว้างของเซลล์

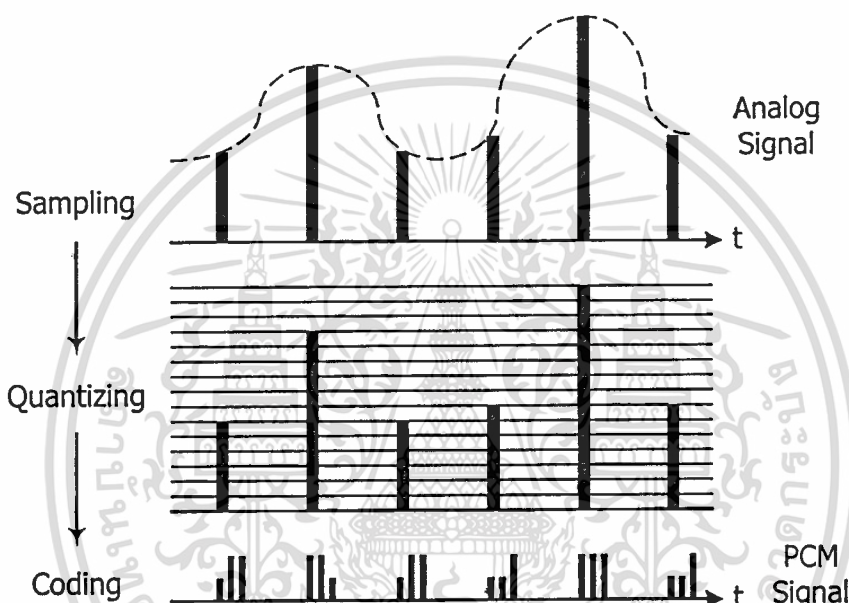
การควอนไทซ์แบบสเกลลาร์มักจะถูกนำมาใช้ในการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล โดยในการแปลงสัญญาณอนาล็อกเป็นดิจิทัลนั้น การแอมพลิจูดของสัญญาณอินพุทจะอยู่ในอัตราคงที่และนำเอาแอมพลิจูดของสัญญาณที่ได้จากการแอมพลีนี้ไปทำการควอนไทซ์

เพื่อให้ผู้อ่านมีความเข้าใจและมองเห็นภาพการควอนไทซ์แบบสเกลลาร์ได้ชัดเจนมากยิ่งขึ้น ผู้เขียนจึงจะขอยกตัวอย่างการประยุกต์ใช้การควอนไทซ์สัญญาณแบบสเกลลาร์ในการแปลงเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่ออนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณอนาล็อกเป็นสัญญาณดิจิทัลด้วยวิธี PCM ซึ่งเป็นมาตรฐานของโทรศัพท์ดิจิทัลที่ใช้กันอยู่ในอเมริกาเหนือมาเพื่อประกอบการทำความเข้าใจ

พัลส์โค้ดโมดูเลชัน

การทำพัลส์โค้ด โมดูเลชันนั้นถูกนำมาใช้ในการเปลี่ยนสัญญาณเสียงซึ่งเป็นสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล PCM โดยสัญญาณเสียงอนาล็อกจะถูกเปลี่ยนไปเป็นสัญญาณดิจิทัล PCM โดยผ่านกระบวนการ แซมปลิง, ควอนไทซิง, และการเข้ารหัส ดังแสดงในรูปที่ 2.5



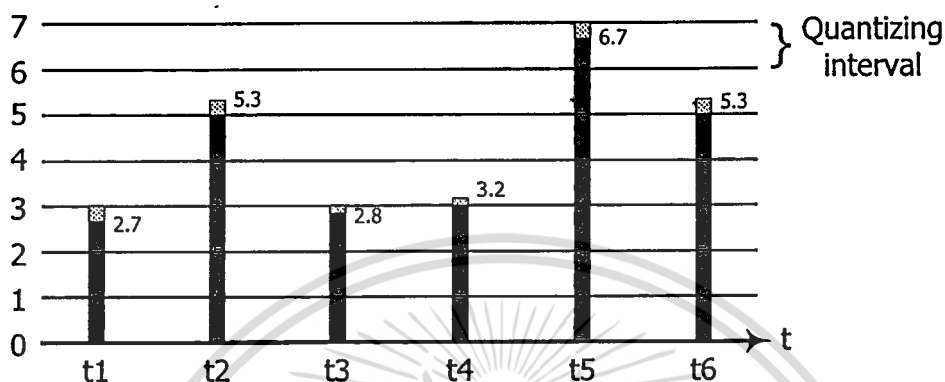
รูปที่ 2.5 การแปลงสัญญาณอนาล็อกเป็นดิจิทัลด้วยวิธี PCM

สำหรับในภาคการแซมปลิงนั้นก็คือการหยิบยกเอาแอมพลิจูดของสัญญาณอนาล็อกมาตามช่วงเวลาที่เหมาะสมเป็นช่วงๆ โดยอาศัยทฤษฎีการแซมปลิง สัญญาณที่ได้จากการแซมปลิงเหล่านี้ถูกเรียกว่าสัญญาณ PAM (Pulse Amplitude Modulation) ซึ่งเป็นสัญญาณที่ถูกจัดเรียงในสเกลของเวลา

หลังจากนั้นสัญญาณ PAM จะถูกจัดระดับโดยการควอนไทซ์ ในกระบวนการควอนไทซ์นี้สเกลของแอมพลิจูดจะถูกแบ่งเป็นช่วงๆเท่าๆกัน ค่าควอนไทซ์แต่ละค่าเรียกว่า ระดับควอนไทซ์ และระยะระหว่างแต่ละระดับเรียกว่า "Quantizing Interval" ขนาดของแต่ละแซมเปิ้ลจะถูกจัดให้อยู่ในระดับการควอนไทซ์ที่อยู่ใกล้ที่สุด

สัญญาณ PAM ที่ได้รับการควอนไทซ์เพียงแต่เป็นการประมาณค่าให้ใกล้เคียงกับสัญญาณอนาล็อกดั้งเดิม ดังนั้นจึงมีข้อผิดพลาดที่เกิดขึ้นบ้างระหว่างค่าระดับแอมพลิจูดของการคำนวณค่า อย่างไรก็ตามมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สัญญาณทั้งสอง ข้อผิดพลาดนี้เรียกว่าเสียงรบกวนที่เกิดจากการควอนไทซ์ ดังแสดงในรูปที่ 2.6 ขนาดของเสียงรบกวนเหล่านี้กระจายอย่างมีระเบียบภายในช่วงห่างระหว่างระดับการควอนไทซ์ โดยไม่คำนึงถึงแอมพลิจูดของสัญญาณเดิม หรือพหุคูณหนึ่งก็คือ กำลังของเสียงรบกวนควอนไทซ์ ซึ่งเกือบจะคงที่ ไม่ขึ้นอยู่กับระดับกำลังของสัญญาณ



รูปที่ 2.6 ความเพี้ยนที่เกิดขึ้นจากระบวนการควอนไทซ์

ถึงแม้เสียงรบกวนที่เกิดจากการควอนไทซ์ซึ่งไม่สามารถจะป้องกันไม่ให้เกิดขึ้นได้แต่ก็ควรจะให้มันน้อยที่สุด เพื่อให้คุณภาพของสัญญาณเสียงมีคุณภาพดี ถ้าเราดูจากรูปจะเห็นว่าเสียงรบกวนที่เกิดจากการควอนไทซ์มีความสัมพันธ์กับช่วงห่างระหว่างระดับการควอนไทซ์ เนื่องจากแอมพลิจูดของเสียงรบกวนที่เกิดจากการควอนไทซ์จะไม่เกินช่วงห่างระหว่างควอนไทซ์ ดังนั้นถ้าทำช่วงห่างระหว่างระดับการควอนไทซ์ให้แคบเสียงรบกวนก็จะสามารถลดลงได้

ขั้นตอนสุดท้ายในการแปลงสัญญาณอนาล็อกเป็นดิจิทัลด้วยวิธีพัลส์โค้ด โมดูเลชันก็คือ การเข้ารหัสให้อยู่ในรูปไบนารีโค้ด ไบนารีโค้ดประกอบด้วยการรวมกันของยูนิตพัลส์ซึ่งอยู่ในรูปของการมีพัลส์หรือ ไม่มีพัลส์คือ 1 และ 0 ในขั้นตอนนี้สัญญาณที่ผ่านการควอนไทซ์มาจะถูกแทนที่ด้วยไค้ดไบนารีที่เรียกว่า “ไค้ดเวิร์ด” ซึ่งเป็นไค้ดที่ใช้แทนในแต่ละระดับของสัญญาณควอนไทซ์

2.2.2 การควอนไทซ์สัญญาณเวกเตอร์ (Vector Quantization:VQ)

การควอนไทซ์สัญญาณแบบเวกเตอร์นั้นเป็นเทคนิคที่พัฒนามาจากการควอนไทซ์ข้อมูลแบบสเกลาร์ซึ่งเป็นปริมาณที่มีมิติเพียงมิติเดียว การเปลี่ยนแปลงจากข้อมูลใน 1 มิติมาเป็นข้อมูลในหลายมิตินั้นเป็นขั้นตอนหลักขั้นตอนหนึ่งในการพัฒนาไปสู่แนวคิดและเทคนิคใหม่ๆ ซึ่งสามารถนำมาเทียบเคียงกับแอปพลิเคชันที่ใช้การควอนไทซ์แบบสเกลาร์ ในขณะที่การควอนไทซ์ปริมาณสเกลาร์ถูกนำมาใช้ในการแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล การควอนไทซ์ปริมาณแบบเวกเตอร์ก็ถูกนำมาใช้ในกระบวนการทางอิมเมจโปรเซสซิ่ง ซึ่งในเกือบทุกกรณีข้อมูล

เอกลักรเป็นเอกลักรที่ส่งไว้สำหรับใช้เพื่อการศึกษาเท่านั้น เมื่อผู้ใดเห็นไปใช้ประโยชน์ในการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อินพุทจะถูกแทนอยู่ในรูปแบบของสัญญาณดิจิทัลอยู่แล้วและเอาท์พุทที่ต้องการมักจะเป็นภาพที่ถูกบีบอัดให้มีขนาดเล็กลงจากข้อมูลภาพอินพุท ถึงกระนั้น โครงสร้าง โมเดล, การวิเคราะห์และเทคนิคการออกแบบต่างๆของแอปพลิเคชันที่ใช้การควอนไทซ์แบบเวกเตอร์มักจะถูกนำมากล่าวไปพร้อมๆกันกับการควอนไทซ์แบบสเกลลาร์

ปริมาณเวกเตอร์สามารถนำมาใช้ในการอธิบายข้อมูลในรูปแบบต่างๆได้มากมาย เช่น ส่วนของสัญญาณคลื่นเสียงหรือภาพ ซึ่งการฟอร์มเวกเตอร์อย่างง่ายก็ได้แก่การฟอร์มเวกเตอร์ขึ้นมาจากแซมเปิลของข้อมูลภาพหรือสัญญาณเสียง เวกเตอร์ควอนไทซ์ชั้นจึงสามารถถูกมองได้ว่าเป็นส่วนหนึ่งของการรู้จำข้อมูล (Pattern Recognition) เมื่อข้อมูลอินพุทถูกประมาณและแทนด้วยข้อมูลพื้นฐานที่อยู่ในเซตเซตหนึ่งที่ถูกกำหนดขึ้น

เวกเตอร์ควอนไทซ์ชั้นเป็นวิธีการบีบอัดข้อมูลวิธีหนึ่งที่ถูกนำมาใช้กับข้อมูลในรูปแบบต่างๆมากมาย ซึ่งรวมไปถึงข้อมูลภาพ, เสียง, การส่งแพ็คเกจ หรือแม้แต่ข้อมูลทางดาวเทียม ด้วยวิธีเวกเตอร์ควอนไทซ์ชั้นนี้ เวกเตอร์ที่ได้มาจากข้อมูลภาพจะถูกเข้ารหัสโดยเวกเตอร์อ้างอิงกลุ่มหนึ่งหรือที่เราเรียกว่า "โค้ดบุค (codebook)" จากทฤษฎีอัตราความผิดพลาดของสัญญาณจะสามารถแสดงได้ว่าวิธีการบีบอัดข้อมูลด้วยวิธีนี้สามารถให้คุณสมบัติการบีบอัดภาพที่คิดว่าการเข้ารหัสด้วยเทคนิคอื่นๆซึ่งใช้หลักการของการเข้ารหัสโดยปริมาณสเกลลาร์ จุดมุ่งหมายของเวกเตอร์ควอนไทซ์ชั้นนั้นก็คือการคำนวณและออกแบบเซตของเวกเตอร์อ้างอิงหรือโค้ดบุคที่ทำให้ภาพเอาท์พุทที่ได้หลังการถอดรหัสมีอัตราความผิดพลาดต่ำที่สุด โดยวิธีการออกแบบโค้ดบุคนี้ผู้อ่านสามารถหาอ่านได้ที่ (Linde. 1980 ; Hsieh. 1992 ; Hsieh. 1991 ; Wu. 1994) วิธีการออกแบบโค้ดบุคที่นิยมนำมาใช้กันมากที่สุดก็คือกระบวนการ LBG Linde (1980)

นิยามของเวกเตอร์ควอนไทซ์ชั้น

เวกเตอร์ควอนไทเซอร์ Q ใน k มิติซึ่งมีขนาดเป็น N เป็นการแมปไปยังเวกเตอร์ใน k มิติกับเซต C ที่แน่นอนซึ่งบรรจุด้วย N เอาท์พุทหรือรีโพรดักชันพ้อยท์ ซึ่งถูกเรียกว่าโค้ดเวกเตอร์หรือโค้ดเวิร์ด ดังนั้น

$$Q : R^k \rightarrow C$$

เมื่อ $C = \{Y_1, Y_2, \dots, Y_N\}$ และ $Y_i \in R^k$ สำหรับทุกๆ $i \in \{1, 2, \dots, N\}$ เซต C ถูกเรียกว่า "โค้ดบุค" หรือ "โค้ด" ซึ่งบรรจุด้วยเวกเตอร์ N เวกเตอร์แต่ละเวกเตอร์อยู่ใน R^k อัตราการเข้าโค้ดหรือที่เราเรียกว่าอัตราของเวกเตอร์ควอนไทเซอร์ $r = (\log_2 N)/k$ ซึ่งเป็นค่าที่บอกถึงจำนวนบิตที่ใช้อต่อ 1 องค์ประกอบเวกเตอร์เพื่อใช้ในการแทนค่าอินพุทเวกเตอร์ และยังสามารถบอกถึงประสิทธิภาพ

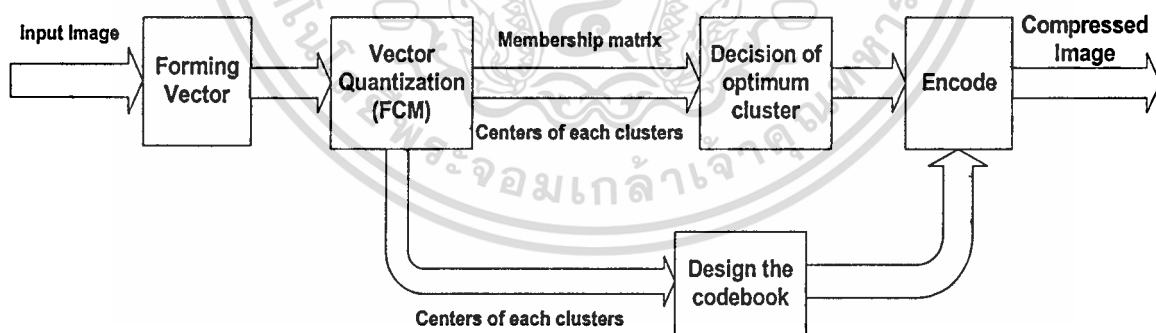
ภาพของการออกแบบโค้ดบुकนั้นด้วย โค้ดบुकนั้นจึงเปรียบเสมือนเป็นตารางที่เก็บค่าของเวกเตอร์ต่างๆเอาไว้ในหน่วยความจำเพื่อใช้ในการแทนค่าเวกเตอร์อินพุท

VQ เป็นรูปแบบใหม่ของการควอนไทซ์ที่พัฒนามาจากการควอนไทซ์ปริมาณสเกลลาร์ VQ เป็นทางเลือกใหม่สำหรับการควอนไทซ์สัญญาณหรือข้อมูลแบบเวกเตอร์ ซึ่งปัจจุบันยังไม่มีเทคโนโลยีการเข้ารหัสใดที่มีประสิทธิภาพดีไปกว่าการเข้ารหัสแบบ VQ ทฤษฎีที่จะกล่าวต่อไปนี้เป็นทฤษฎีที่ถูกนำมาใช้วัดคุณสมบัติและประสิทธิภาพของระบบการเข้ารหัสใดๆซึ่งเป็นระบบที่กระทำบนสัญญาณในรูปแบบสัญญาณเวกเตอร์ ซึ่งกระบวนการเข้ารหัสแบบ VQ นั้นมีคุณสมบัติที่เป็นไปตามทฤษฎีดังกล่าวนี้

ทฤษฎี : สำหรับระบบการเข้ารหัสใดๆซึ่งทำการแมปสัญญาณเวกเตอร์อินพุทเข้ากับเวกเตอร์หนึ่งๆภายใน N ไบนารีเวิร์คเวกเตอร์โดยการประมาณค่าเวกเตอร์นั้น โค้ดบुकขนาด N ที่ถูกสร้างมาจากเวกเตอร์ควอนไทเซอร์ที่คั้นนั้นจะต้องมีเวกเตอร์หนึ่งๆภายในโค้ดบुकที่มีคุณสมบัติใกล้เคียงกับเวกเตอร์อินพุทมากที่สุด

2.3 การประยุกต์ใช้งานเวกเตอร์ควอนไทเซชันในการบีบอัดข้อมูลภาพ

เวกเตอร์ควอนไทเซชันนั้นมักจะถูกนำมาใช้ในการบีบอัดข้อมูลภาพหรือสัญญาณเสียงเพื่อลดขนาดของหน่วยความจำที่ต้องใช้ในการเก็บข้อมูล สำหรับกระบวนการบีบอัดข้อมูลภาพนั้นสามารถแสดงได้ดังบล็อกไดอะแกรมในรูปที่ 2.7



รูปที่ 2.7 : บล็อกไดอะแกรมของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชัน

2.3.1 การฟอร์มเวกเตอร์จากข้อมูลภาพอินพุท (Vector Forming)

จากบล็อกไดอะแกรมในรูปที่ 2.7 ขั้นตอนแรกของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันนั้นข้อมูลภาพอินพุททั้งหมดจะถูกนำมาสร้างเป็นเซตของเวกเตอร์อินพุทซึ่งเราเรียกการทำงานในบล็อกนี้ว่าการฟอร์มเวกเตอร์ (Forming Vector) การฟอร์มเวกเตอร์จากข้อมูลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลายแบนด์นั้นสามารถทำได้ 2 วิธีคือการฟอร์มเวกเตอร์ด้วยวิธี "อินทราแบนด์" และการฟอร์มเวกเตอร์ด้วยวิธี "อินเตอร์แบนด์"

- การฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์ (Intra-band Vector Forming)

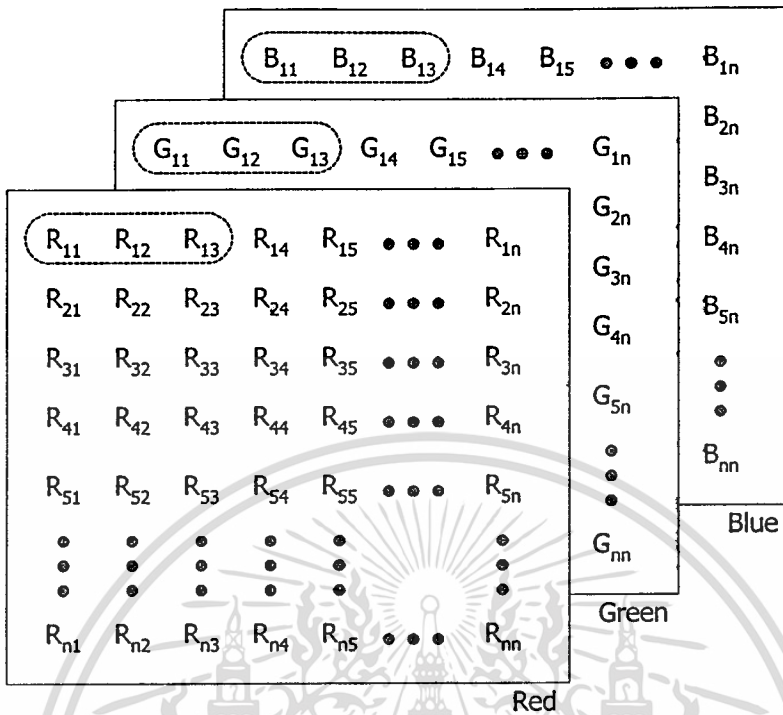
การฟอร์มเวกเตอร์จากข้อมูลภาพอินพุตด้วยวิธีอินทราแบนด์นั้นเป็นการฟอร์มเวกเตอร์โดยการนำข้อมูลภาพในแต่ละแบนด์มาแบ่งออกเป็นบล็อกขนาดเท่าๆกัน เช่น บล็อกขนาด 2×2 , 1×2 หรือ 3×3 จากนั้นบล็อกที่ถูกกำหนดขึ้นมาแต่ละบล็อกนี้จะถูกแทนด้วยเวกเตอร์ 1 เวกเตอร์ จากการฟอร์มเวกเตอร์อินพุตขึ้นมาด้วยวิธีนี้จะเห็นว่าจำนวนมิติของเวกเตอร์จะมีค่าเท่ากับจำนวนอิลลิเมนต์ภายในบล็อกที่ถูกกำหนดขึ้นมา นอกจากนี้ข้อมูลในแต่ละอิลลิเมนต์ของเวกเตอร์ที่ได้จากการฟอร์มเวกเตอร์ด้วยวิธีนี้จะอยู่ภายในแบนด์เดียวกันของข้อมูลภาพ เราจึงเรียกวธีการฟอร์มเวกเตอร์ด้วยวิธีนี้ว่าวิธีการฟอร์มเวกเตอร์อินพุตด้วยวิธี "อินทราแบนด์" การฟอร์มเวกเตอร์ด้วยวิธีนี้แสดงได้ดังรูปที่ 2.8

- การฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์ (Inter-band Vector Forming)

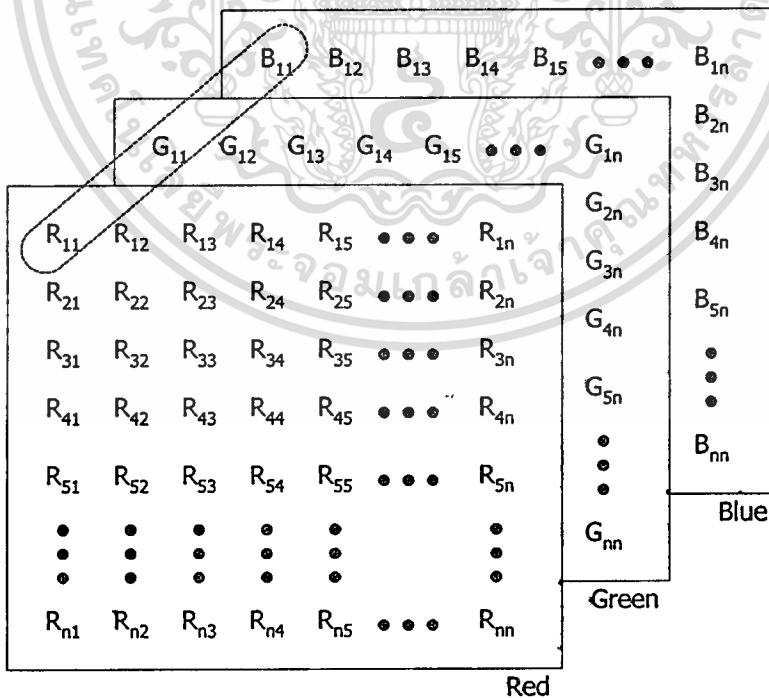
การฟอร์มเวกเตอร์อินพุตจากภาพอินพุตด้วยวิธีนี้จะใช้วิธีการแทนแต่ละพิกเซลของภาพด้วยเวกเตอร์ 1 เวกเตอร์ ดังนั้นจำนวนมิติของเวกเตอร์ที่ได้จะมีมิติเท่ากับจำนวนแบนด์ของภาพอินพุต จะเห็นได้ว่าการฟอร์มเวกเตอร์ด้วยวิธีนี้ข้อมูลในแต่ละมิติของเวกเตอร์จะอยู่ในแต่ละแบนด์ของข้อมูลภาพอินพุต เราจึงเรียกวธีการฟอร์มเวกเตอร์ด้วยวิธีนี้ว่าการฟอร์มเวกเตอร์อินพุตด้วยวิธี "อินเตอร์แบนด์" การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์แสดงได้ดังรูปที่ 2.9

2.3.2 การออกแบบโค้ดบุค (Codebook Designing)

ขั้นตอนหลักอีกที่สำคัญอีกขั้นตอนหนึ่งสำหรับการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันก็คือขั้นตอนการออกแบบโค้ดบุคหรือการคำนวณเซตของเวกเตอร์อ้างอิง ในขั้นตอนการออกแบบโค้ดบุคนี้จะเป็นขั้นตอนที่ทำหน้าที่คำนวณเซตของค่าเวกเตอร์อ้างอิงที่เหมาะสมที่จะนำมาใช้ในการประมาณค่าเวกเตอร์อินพุต โดยค่าเวกเตอร์อ้างอิงที่ถูกคำนวณขึ้นมาเมื่อนำไปใช้ประมาณค่าเวกเตอร์อินพุตแล้วจะต้องให้เวกเตอร์เอาท์พุตที่มีความผิดเพี้ยนไปจากสัญญาณอินพุตต่ำที่สุด สำหรับวิธีที่นิยมใช้ในการออกแบบโค้ดบุคสำหรับการบีบอัดข้อมูลภาพนั้นมักจะใช้วิธีการจำแนกกลุ่มข้อมูลหรือที่เราเรียกว่าการวิเคราะห์คลัสเตอร์นั่นเอง ในการคำนวณค่าเวกเตอร์อ้างอิงจากการวิเคราะห์คลัสเตอร์ข้อมูลภาพอินพุตนี้ก็คือการจำแนกข้อมูลภาพออกเป็นกลุ่มหรือคลัสเตอร์แล้วทำการคำนวณค่าเวกเตอร์เฉลี่ยของกลุ่มนี้ ค่าเวกเตอร์เฉลี่ยที่ได้ก็คือเซ็นเตอร์ของทุกคลัสเตอร์ ดังแสดงได้ในรูปที่ 2.10

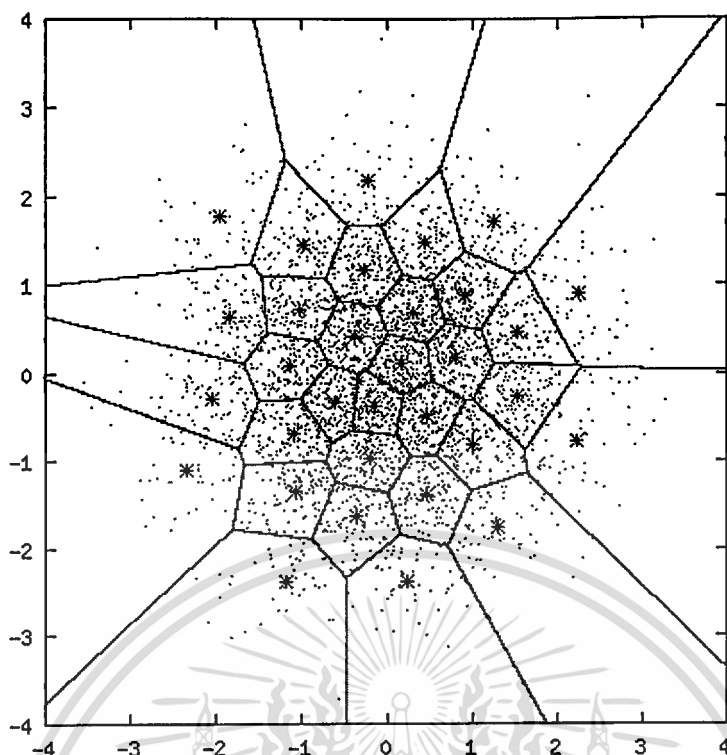


รูปที่ 2.8 การฟอร์มเวกเตอร์แบบอินทราแบนด์



รูปที่ 2.9 การฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



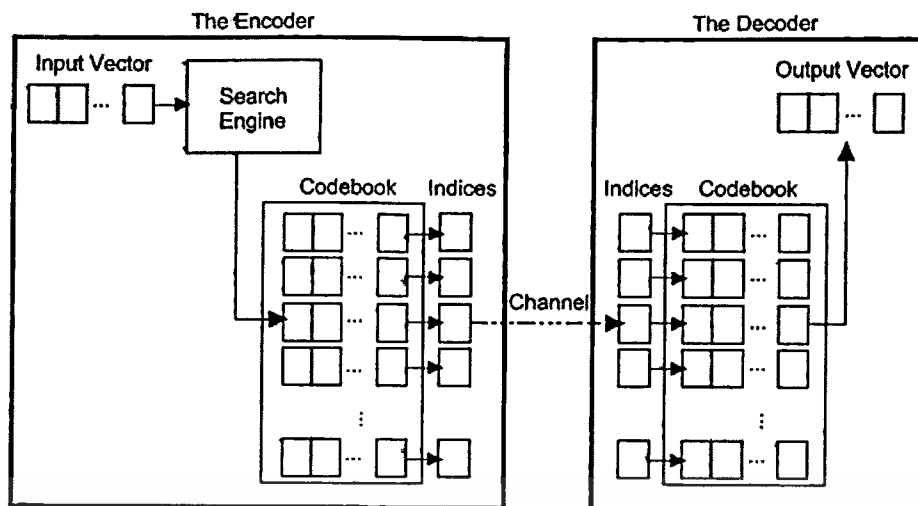
รูปที่ 2.10 แสดงการคำนวณค่าเวกเตอร์อ้างอิงจากการวิเคราะห์คลัสเตอร์

2.3.3 การประมาณค่าเวกเตอร์อินพุท (Decision of optimum cluster)

ในขั้นตอนนี้เวกเตอร์อินพุททุกๆตัวที่ได้จากการฟอร์มเวกเตอร์ในหัวข้อ 2.3.1 จะถูกนำมาประมาณค่าด้วยเวกเตอร์ค่าหนึ่งซึ่งอยู่ภายในเซตของเวกเตอร์อ้างอิงหรือโค้ดบุค โดยเวกเตอร์เอาท์พุทที่ได้จะต้องมีความผิดพลาดอันเกิดจากการควอนไทซ์น้อยที่สุด

2.3.4 การเข้ารหัสข้อมูลภาพ (Image Encoding)

ขั้นตอนสุดท้ายสำหรับการบีบอัดข้อมูลภาพดิจิทัลก็คือการเข้ารหัสข้อมูลภาพ เนื่องจากแต่ละเวกเตอร์ภายในเซตของเวกเตอร์อ้างอิงจะมีรหัสหรือดัชนีประจำแต่ละเวกเตอร์ซึ่งเราเรียกว่า "โค้ดเวิร์ด (Codeword)" ดังนั้นบางครั้งเราจึงเรียกเวกเตอร์ภายในเซตของเวกเตอร์อ้างอิงว่า "โค้ดเวกเตอร์" ดังนั้นในฟังก์ชันบล็อคนี้เวกเตอร์เอาท์พุทที่ได้จากการประมาณค่าเวกเตอร์อินพุทด้วยเวกเตอร์อ้างอิงจะถูกนำมาเข้ารหัสด้วยโค้ดเวิร์ดของแต่ละเวกเตอร์เอาท์พุท รูปที่ 2.11 แสดงระบบการเข้ารหัสและถอดรหัสเวกเตอร์



รูปที่ 2.11 การเข้ารหัสและถอดรหัสเวกเตอร์

ภายในฟังก์ชันบล็อกนี้เวกเตอร์ที่ได้จากภาพอินพุตแต่ละเวกเตอร์จะถูกนำมาประมาณค่าด้วยเวกเตอร์ค่าหนึ่งซึ่งอยู่ในเซตของเวกเตอร์อ้างอิงหรือ “โค้ดบุค” ซึ่งถูกสร้างขึ้นมาโดยกระบวนการเวกเตอร์ควอนไทเซชัน โดยเวกเตอร์เอาต์พุตที่ได้จากการประมาณค่าแล้วนี้จะต้องมีความผิดเพี้ยนของสัญญาณเมื่อเทียบกับสัญญาณอินพุตค่าที่สุด เวกเตอร์แต่ละค่าภายใน โค้ดบุคนี้จะมี “โค้ดเวิร์ด” หรือรหัสประจำตัวของแต่ละเวกเตอร์ เราจึงมักเรียกแต่ละเวกเตอร์อ้างอิงภายใน โค้ดบุคว่า “โค้ดเวกเตอร์” สำหรับการบีบอัดข้อมูลภาพนี้ หลังจากเวกเตอร์ทุกๆเวกเตอร์ที่ได้จากภาพอินพุตถูกประมาณค่าโดยเวกเตอร์ภายใน โค้ดบุคแล้ว เวกเตอร์แต่ละตัวนี้จะถูกเข้ารหัสด้วย โค้ดเวิร์ดประจำตัวของมัน

จากกระบวนการดังกล่าว เวกเตอร์เอาต์พุตทั้งหมดจะถูกแทนด้วยโค้ดประจำแต่ละเวกเตอร์ซึ่งจะเห็นได้ว่าจำนวนบิตที่ใช้ในการแทนค่าเวกเตอร์อ้างอิงเหล่านี้จะมากหรือน้อยนั้นขึ้นอยู่กับขนาดของโค้ดบุค หากโค้ดบุคมีขนาดใหญ่ จะทำให้ภายใน โค้ดบุคบรรจุไปด้วยเวกเตอร์อ้างอิงจำนวนมาก จำนวนบิตที่ใช้ในการแทนเวกเตอร์อ้างอิงหนึ่งเวกเตอร์ก็จะมากตามไปด้วย แต่หากขนาดของโค้ดบุคมีขนาดเล็กจำนวนบิตที่ใช้ในการแทนที่แต่ละเวกเตอร์ก็จะมีค่าน้อย ดังนั้นในการบีบอัดภาพด้วยวิธีนี้นั้นจึงต้องคำนึงถึงจำนวนเวกเตอร์อ้างอิง (reference vector) ที่จะใช้ควบคู่ไปกับอัตราการบีบอัดภาพที่ต้องการด้วย

2.4 การนำข้อมูลจากการคลัสเตอร์แบบพีซีซีมีนส์มาใช้ในการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชัน

สำหรับการนำการวิเคราะห์คลัสเตอร์แบบพีซีซีมีนส์มาประยุกต์ใช้ในการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันนั้น ก็คือการนำข้อมูลที่ได้จากการวิเคราะห์คลัสเตอร์มาคำนวณเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาโค้ดบुक ซึ่งข้อมูลที่ได้จากการวิเคราะห์คลัสเตอร์ที่สำคัญที่นำมาใช้ในการหาโค้ดบुकก็คือค่าเฉลี่ยหรือเซ็นเตอร์ของแต่ละคลัสเตอร์ นอกจากนั้นข้อมูลที่สำคัญอีกตัวหนึ่งสามารถนำมาช่วยในการบีบอัดข้อมูลภาพหลายแบนด์ก็คือ ค่าความเป็นสมาชิกของแต่ละเวกเตอร์ที่มีต่อแต่ละคลัสเตอร์ ซึ่งค่าความเป็นสมาชิกนี้สามารถนำมาใช้ในขั้นตอนของการประมาณค่าเวกเตอร์เพื่อหาว่าแต่ละเวกเตอร์นั้นสมควรจะจัดอยู่ในคลัสเตอร์ใดและควรจะใช้ค่าใดในการประมาณค่า

2.5 การบีบอัดข้อมูลภาพโดยใช้โค้ดบुकมาตรฐาน

เนื่องจากการบีบอัดข้อมูลภาพหลายแบนด์ด้วยวิธีการควอนไทซ์แบบเวกเตอร์ที่คำนวณจากพีซีซีมีมันส์นั้น การคำนวณ โค้ดบुकเป็นกระบวนการที่มีการคำนวณอย่างซับซ้อนและใช้เวลาคำนวณมาก จึงไม่เป็นการสะดวกที่จะทำการคำนวณ โค้ดบुकใหม่ทุกครั้งที่จะทำการบีบอัดข้อมูลภาพอินพุตแต่ละภาพ การสร้าง โค้ดบुकมาตรฐานขึ้น โดยการนำข้อมูลจากภาพอินพุตที่ต่างกันหลายๆภาพมารวมกันแล้วทำการวิเคราะห์คลัสเตอร์เพื่อคำนวณหาเซตของเวกเตอร์อ้างอิงมาตรฐานที่เหมาะสมที่นำไปใช้กับข้อมูลภาพหลายแบนด์ใดใด จึงเป็นวิธีหนึ่งที่สามารถลดระยะเวลาในการบีบอัดข้อมูลภาพหลายแบนด์แต่ละภาพได้ เพราะโค้ดบुकมาตรฐานนี้จะสามารถนำไปใช้กับข้อมูลภาพต่างๆ ได้เลยโดยไม่จำเป็นต้องทำการคำนวณ โค้ดบुकใหม่ทุกครั้งที่จะทำการบีบอัดข้อมูลภาพหลายแบนด์ ดังนั้นด้วยวิธีการนี้เราจะสามารถทำการบีบอัดข้อมูลภาพหลายแบนด์ใดๆได้โดยการนำข้อมูลเวกเตอร์อินพุตที่ได้จากข้อมูลภาพอินพุตมาทำการจัดระดับใหม่โดยเทียบกับเซตของเวกเตอร์อ้างอิงมาตรฐานที่อยู่ใน โค้ดบुकมาตรฐาน จากนั้นเมื่อทำการจัดระดับข้อมูลเวกเตอร์อินพุตแล้วก็สามารถทำการเข้ารหัสโดยการแทนข้อมูลเวกเตอร์อินพุตด้วยรหัสที่เก็บอยู่ใน โค้ดบुकได้เลยทันที

2.6 การตรวจสอบประสิทธิภาพของการบีบอัดข้อมูล

เนื่องจากการบีบอัดข้อมูลภาพหลายแบนด์นั้นมีอยู่หลายวิธี การตรวจสอบประสิทธิภาพของการบีบอัดข้อมูลนั้นจึงเป็นการบ่งบอกว่าการบีบอัดข้อมูลภาพด้วยวิธีการนั้นๆมีประสิทธิภาพที่ดีเพียงใด โดยการตรวจสอบประสิทธิภาพของการบีบอัดข้อมูลภาพนั้นสามารถทำได้โดยการคำนวณค่าพารามิเตอร์มีเตอร์ต่างๆดังนี้

2.6.1 ค่า bpp หรือ bits per pixel

ค่า bpp นี้เป็นพารามิเตอร์ที่บอกถึงจำนวนบิตที่ต้องใช้ในการแทนค่าเวกเตอร์อินพุตหนึ่งเวกเตอร์ ดังนั้นวิธีการบีบอัดข้อมูลภาพใดที่มีค่า bpp น้อยก็แสดงว่าการบีบอัดข้อมูลภาพด้วยวิธีนั้นมีอัตราการบีบอัดภาพได้สูง โดยการคำนวณค่า bpp นั้นสามารถคำนวณได้ดังสมการที่ 2.1

$$bpp(\text{number of bits per pixel}) = \frac{\text{Encoded number of bits}}{\text{Number of pixels}} \quad (2.1)$$

2.6.2 ค่า PSNR (Peak Signal to noise ration)

ค่า PSNR นี้เป็นค่าที่แสดงถึงอัตราส่วนระหว่างสัญญาณอินพุตต่อสัญญาณรบกวนที่เกิดขึ้นอันเนื่องมาจากกระบวนการควอนไทเซชัน ซึ่งการวิธีการบีบอัดข้อมูลภาพที่มีค่า PSNR จะถือว่าเป็นวิธีการบีบอัดข้อมูลภาพที่มีประสิทธิภาพสูงเนื่องจากเกิดสัญญาณรบกวนต่ำ การคำนวณหาค่า PSNR นั้นสามารถคำนวณได้ดังสมการที่ 2.2

$$PSNR(\text{Peak Signal to noise ratio}) = 10 \log_{10} \left[\frac{255^2}{MSE} \right] [dB] \quad (2.2)$$

โดยค่า MSE (Mean Square Error) นั้นเป็นค่าความผิดพลาดของข้อมูลที่เกิดขึ้น ซึ่งสามารถคำนวณได้จากสมการที่ 2.3

$$MSE(\text{Mean square error}) = \frac{1}{3M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=1}^3 \{x_k(j,i) - \hat{x}_k(j,i)\}^2 \quad (2.3)$$

โดยค่า M และ N คือจำนวนพิกเซลตามแนวกว้างและยาวของข้อมูลภาพอินพุต

บทที่ 3

การวิเคราะห์คลัสเตอร์ (Cluster Analysis)

3.1 ความเป็นมาของการวิเคราะห์คลัสเตอร์

การวิเคราะห์คลัสเตอร์ข้อมูลภาพ (Clustering) นั้นเป็นกระบวนการหนึ่งที่สำคัญในการศึกษาด้านอิมเมจโปรเซสซิ่ง โดยจุดมุ่งหมายของการวิเคราะห์คลัสเตอร์ก็คือการจำแนกข้อมูลภาพอินพุตออกเป็นกลุ่มๆ โดยที่สมาชิกที่อยู่ในกลุ่มเดียวกันจะมีคุณลักษณะที่เหมือนหรือคล้ายคลึงกัน ส่วนสมาชิกที่อยู่ต่างกลุ่มกันจะมีคุณลักษณะที่แตกต่างกัน สำหรับภาพสีหรือภาพหลายแบนด์นั้น เราสามารถมองได้ว่าแต่ละพิกเซลภายในภาพเปรียบเสมือนเวกเตอร์เวกเตอร์หนึ่งใน n มิติ (เมื่อ n คือจำนวนแบนด์ของภาพอินพุต) ดังนั้นการวิเคราะห์คลัสเตอร์สำหรับภาพหลายแบนด์หรือภาพสีจึงเป็นการจำแนกเวกเตอร์ที่ได้จากภาพอินพุตเหล่านี้ออกเป็นกลุ่มๆ ข้อมูลที่ได้จากการจำแนกกลุ่มเวกเตอร์เหล่านี้สามารถนำไปใช้ประโยชน์ในการศึกษาด้านอิมเมจโปรเซสซิ่งได้มากมายเช่นสามารถนำไปใช้ช่วยในการหาขอบเขตของวัตถุภายในภาพสีหรือภาพหลายแบนด์ได้ (edge detection) (Bezdek, 1984) แต่สำหรับวิทยานิพนธ์เล่มนี้จะนำการวิเคราะห์คลัสเตอร์มาประยุกต์ใช้ในการออกแบบโค้ดบุคสำหรับการบีบอัดขนาดของข้อมูลภาพ

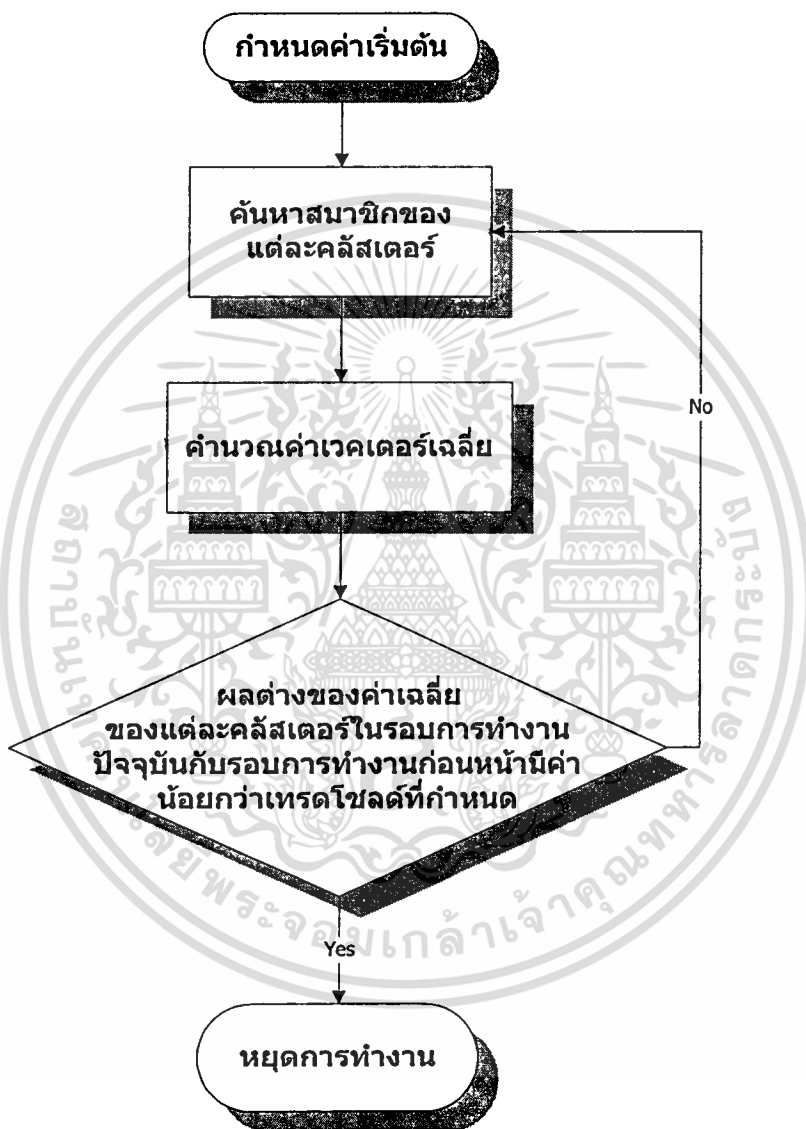
3.2 การคลัสเตอร์เซตของข้อมูล

การวิเคราะห์คลัสเตอร์สำหรับข้อมูลภาพนั้นมีหลายวิธี โดยแต่ละวิธีนั้นจะมีขั้นตอนการทำงานคล้ายคลึงกันดังนี้

1. กำหนดค่าเริ่มต้นในการวิเคราะห์คลัสเตอร์อันได้แก่ จำนวนคลัสเตอร์ที่ต้องการและค่าเริ่มต้นของเซ็นเตอร์ในแต่ละคลัสเตอร์ ซึ่งสำหรับการนำการวิเคราะห์คลัสเตอร์มาใช้ในการออกแบบโค้ดบุคนั้น การเลือกจำนวนคลัสเตอร์จะเป็นการกำหนดขนาดของแต่ละโค้ดเวิร์คและขนาดของโค้ดบุคด้วย
2. ค้นหาสมาชิกของแต่ละคลัสเตอร์โดยการกำหนดให้แต่ละเวกเตอร์ภายในเซตของเวกเตอร์อินพุตเป็นสมาชิกของคลัสเตอร์ที่มีระยะห่างระหว่างเซ็นเตอร์ของคลัสเตอร์กับค่าของเวกเตอร์นั้นๆ น้อยที่สุด
3. ปรับปรุงค่าเซ็นเตอร์ของแต่ละคลัสเตอร์ โดยคำนวณจากค่าเฉลี่ยของเวกเตอร์ที่เป็นสมาชิกอยู่ในคลัสเตอร์นั้นๆ

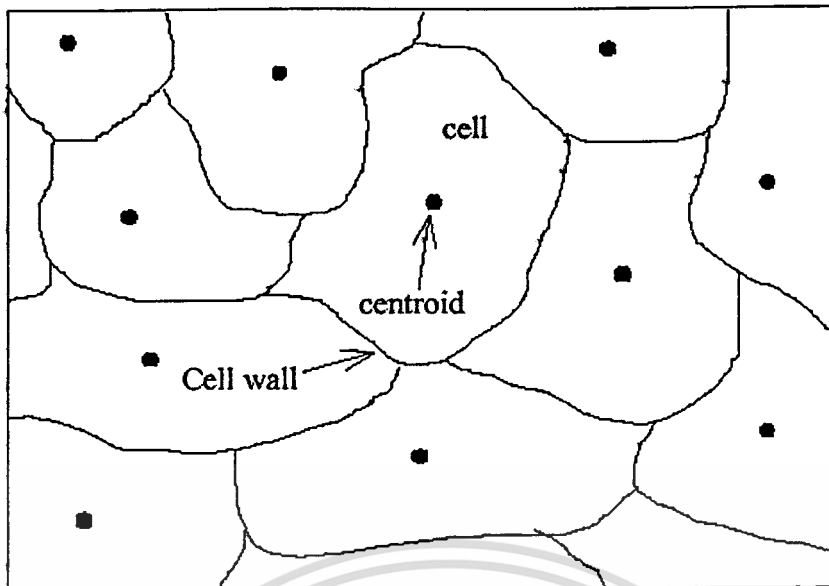
4. ทำซ้ำในขั้นตอนที่ 2 และ 3 จนกระทั่งการเปลี่ยนแปลงค่าเซ็นเตอร์ของทุกคลัสเตอร์ที่คำนวณได้จากผลต่างของค่าเซ็นเตอร์ของแต่ละคลัสเตอร์ในรอบการทำงานปัจจุบันกับรอบการทำงานก่อนหน้ารอบปัจจุบันมีค่าต่ำกว่าค่าเทรชโวลด์ที่กำหนดไว้

รูปที่ 3.1 แสดงโฟลว์ไคอะแกรมของการวิเคราะห์คลัสเตอร์ ส่วนรูปที่ 3.2 ผลการวิเคราะห์คลัสเตอร์ข้อมูลเวกเตอร์สองมิติ



รูปที่ 3.1 Flow Diagram ของการวิเคราะห์คลัสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.2 การวิเคราะห์คลัสเตอร์ข้อมูลเวกเตอร์ใน 2 มิติ

การวิเคราะห์คลัสเตอร์นั้นสามารถทำได้หลายวิธี (Cannon, 1986) สำหรับวิธีที่จะกล่าวถึงภายในวิทยานิพนธ์ฉบับนี้ได้แก่วิธีการวิเคราะห์คลัสเตอร์แบบฟัซซี่ซีมีนส์ (Fuzzy c-means) ซึ่งเป็นวิธีหนึ่งที่นิยมนำมาใช้งานอย่างแพร่หลาย

3.3 การวิเคราะห์คลัสเตอร์ข้อมูลภาพด้วยวิธีฟัซซี่ซีมีนส์ (Fuzzy c-means clustering)

การวิเคราะห์คลัสเตอร์ด้วยวิธีฟัซซี่ซีมีนส์ (Bezdek, 1994) เป็นการวิเคราะห์คลัสเตอร์วิธีหนึ่งที่มีการคำนวณแบบทำซ้ำ (iterative clustering) โดยจุดมุ่งหมายของการวิเคราะห์คลัสเตอร์ด้วยวิธีนี้ก็คือการคำนวณค่าเมมเบอร์ชิพเมตริกซ์และค่าเวกเตอร์เฉลี่ยในแต่ละกลุ่ม โดยค่าเมมเบอร์ชิพเมตริกซ์นี้จะเป็นเมตริกที่มีมิติเป็น $c \times n$ เมื่อ c คือจำนวนกลุ่มของข้อมูลภาพที่เราต้องการจำแนกออก และ n คือจำนวนของเวกเตอร์อินพุตทั้งหมด (สำหรับภาพหลายแบนด์นั้น n จะมีค่าเท่ากับจำนวนพิกเซลทั้งหมดภายในภาพ) โดยค่าที่เก็บอยู่ในเมมเบอร์ชิพเมตริกซ์นี้ก็คือค่าความเป็นสมาชิก (membership value) ของแต่ละพิกเซลหรือแต่ละเวกเตอร์ภายในภาพที่มีต่อคลัสเตอร์แต่ละคลัสเตอร์ ค่าเมมเบอร์ชิพเหล่านี้จะมีค่าไม่ต่ำกว่า 0 และจะมีค่าไม่เกิน 1 โดยหากค่าเมมเบอร์ชิพของพิกเซลหนึ่งๆที่มีต่อคลัสเตอร์หนึ่งๆมีค่าน้อยหรือเข้าใกล้ศูนย์ก็จะแสดงว่าพิกเซลนั้นมีความเป็นสมาชิกของคลัสเตอร์นั้นต่ำ คุณสมบัติของเวกเตอร์พิกเซลนั้นก็จะเป็นสัดส่วนต่างกับค่าเวกเตอร์เฉลี่ยของคลัสเตอร์นั้นๆ แต่หากค่าเมมเบอร์ชิพของพิกเซลนั้นๆที่มีต่อคลัสเตอร์นั้นๆมีค่ามากหรือเข้าใกล้ 1 ก็แสดงว่าพิกเซลนั้นมีความเป็นสมาชิกต่อคลัสเตอร์นั้นมากและค่าเวกเตอร์ของพิกเซลนั้นก็จะค่อนข้างมีค่าใกล้เคียงค่าเวกเตอร์เฉลี่ยที่คำนวณได้จากคลัสเตอร์นั้น สำหรับค่าเวกเตอร์

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ไว้เพื่อการศึกษาค้นคว้าเท่านั้น เมื่อผู้ใดเห็นไปเผยแพร่โดยไม่ได้รับอนุญาต
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดลอกเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เฉลี่ยนั้นคือเวกเตอร์ที่ได้จากการหาค่าเฉลี่ยของเวกเตอร์ทั้งหมดที่เป็นสมาชิกของคลัสเตอร์นั้นๆ
จึงอาจมองได้ว่าค่าเวกเตอร์เฉลี่ยนี้เป็นเซ็นเตอร์ของคลัสเตอร์นั้นๆ

การคำนวณเพื่อจำแนกข้อมูลภาพแบบฟัซซีก็คือการคำนวณให้ค่าสแควร์เออเรอร์ฟังก์ชันมี
ค่าน้อยที่สุด (least-squared errors) ซึ่งสามารถอธิบายได้โดยสมการที่ (3.1)

$$\text{Minimize: } J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ki})^m \|y_k - v_i\|^2 A \quad \dots\dots(3.1)$$

เมื่อ : $Y = \{y_1, y_2, y_3, \dots, y_N\} \subset R^n$ คือเซตของข้อมูลอินพุต,

C คือจำนวนของคลัสเตอร์ที่ใช้ : $2 \leq c < n$,

M คือค่าถ่วงน้ำหนัก : $1 \leq m < \infty$,

$U = \{u_{ki}\}$ คือค่าเมมเบอร์ชิพเมตริกซ์ของ Y

$\|y_k - v_i\|^2 A$ เป็นการนอร์มัลไลซ์บน R^n , และ,

A เป็นค่าเมตริกถ่วงน้ำหนักขนาด $(n \times n)$

ค่าการถ่วงน้ำหนัก m จะมีผลต่อการลดลงของค่าสแควร์เออเรอร์ตามปริมาณที่ขึ้นอยู่กับค่า
ความเป็นสมาชิกในคลัสเตอร์ที่กำลังทำการสังเกต

ขั้นตอนแรกของกระบวนการ FCM ก็คือการสุ่มเมมเบอร์ชิพเมตริกซ์ซึ่งเป็นเมตริกซ์ที่เก็บ
ค่าความเป็นสมาชิกของแต่ละพิกเซลที่มีต่อแต่ละคลัสเตอร์ และเป็นเมตริกซ์ที่ใช้ถ่วงน้ำหนักการ
เป็นสมาชิกของแต่ละพิกเซล จากนั้นจะทำการคำนวณค่าเฉลี่ยหรือเซ็นเตอร์ของแต่ละคลัสเตอร์ ค่า
เซ็นเตอร์ตัวใหม่ที่ได้มานี้จะถูกนำไปปรับปรุงเมมเบอร์ชิพเมตริกซ์ เมื่อเมมเบอร์ชิพเมตริกซ์ถูก
ปรับปรุงแล้วก็จะนำไปเปรียบเทียบกับเมมเบอร์ชิพเมตริกซ์ตัวเก่า(ก่อนที่จะมีการปรับปรุง) ถ้าเมม
เบอร์ชิพเมตริกซ์ทั้งสองมีความแตกต่างกันมากกว่าค่าเทรชโฮลด์ค่าหนึ่งที่กำหนด การทำงานก็จะ
กลับไปวนรอบทำซ้ำจนกว่าค่าเมมเบอร์ชิพเมตริกซ์ที่ได้ค่าใหม่จะแตกต่างจากค่าเก่าน้อยกว่าค่าเท
รชโฮลด์ที่กำหนดจึงจะเสร็จสิ้นการทำงาน

ขั้นตอนการทำงานของ FCM มีดังนี้

1. กำหนดค่า c, A, m, ϵ , และลูปเคาน์เตอร์ $t = 1$
2. สุ่ม $N \times c$ เมมเบอร์ชิพเมตริกซ์ (U)
3. คำนวณเซ็นเตอร์ของแต่ละคลัสเตอร์โดยสมการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$v_i^{(t)} = \sum_{k=1}^N (u_{ki}^{(t)})^m y_k / \sum_{k=1}^N (u_{ki}^{(t)})^m \quad (3.2)$$

4. ปรับปรุงเมมเบอร์ชิพเมตริกซ์โดยสมการ

$$u_{ki}^{(t+1)} = \left(\sum_{j=1}^c \left(\frac{d_{ki}}{d_{kj}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (3.3)$$

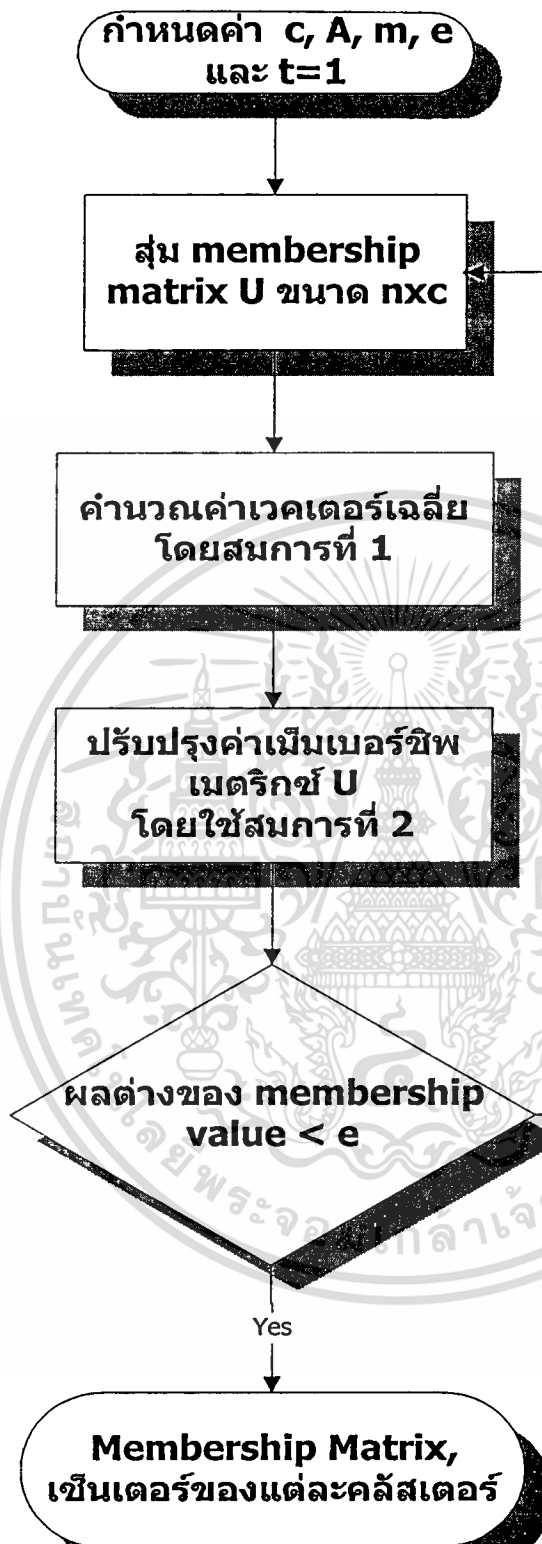
เมื่อ $d_{ki} = \| y_k - v_i^{(t)} \| A$

5. ถ้าค่าสูงสุดของ $\{ |u_{ki}^{(t)} - u_{ki}^{(t-1)}| \} > \varepsilon$ ก็ให้เพิ่มค่า t และกลับไปทำในขั้นตอนที่ 3.

ไฟล์วีดิโอแถมของการวิเคราะห์คลัสเตอร์ด้วยวิธีฟัชซีมีนัสแสดงได้ดังรูปที่ 3.3

3.4 การเพิ่มความเร็วของกระบวนการฟัชซีมีนัส

มีงานวิจัยหลายๆตัวที่นำเสนอการเพิ่มความเร็วในการทำงานของ FCM อัลกอริทึม การวิเคราะห์คลัสเตอร์โดยวิธี FCM แบบประมาณค่า (AFCM: Approximate FCM) (Cannon, 1986) ก็เป็นวิธีหนึ่งในการเพิ่มความเร็วของการวิเคราะห์คลัสเตอร์แบบ FCM โดยจะมีการแทนค่าข้อมูลด้วยตัวเลขจำนวนเต็มและใช้เทคนิคการการค้นหาข้อมูลซึ่งสามารถลดเวลาการทำงานของ FCM ในแต่ละการวนรอบลงได้ นอกจากนี้วิธีการวิเคราะห์คลัสเตอร์ด้วยวิธี FCM แบบประมาณค่านี้แล้วยังมีอีกสองกระบวนการในการเพิ่มความเร็วของ FCM (Kamel, 1994) ในขั้นตอนแรกค่าเฉลี่ยหรือเซ็นเตอร์ของแต่ละคลัสเตอร์จะถูกคำนวณขึ้นมาก่อน แล้วจะนำค่าเซ็นเตอร์ของแต่ละคลัสเตอร์นี้ไปใช้ในการคำนวณหาค่าความเป็นสมาชิก (Membership Value) ของแต่ละพิกเซลที่มีต่อคลัสเตอร์นั้นๆ พร้อมกับคำนวณหาค่าเซ็นเตอร์ใหม่ทุกๆครั้งที่มีการคำนวณในแต่ละพิกเซล ถึงแม้กระบวนการนี้จะต้องเพิ่มการคำนวณมากขึ้นในแต่ละรอบการทำงานแต่ก็จะสามารถทำให้การวิเคราะห์สำเร็จลงได้เร็วขึ้น



รูปที่ 3.3 โฟลว์ไดอะแกรมของ FCM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับในบทความนี้จะขอนำเสนอวิธีการในการปรับปรุงการวิเคราะห์คลัสเตอร์แบบ FCM เพื่อลดเวลาการทำงานในแต่ละรอบการวนรอบของ FCM ในขณะที่ใช้จำนวนการวนรอบการทำงานเท่าเดิม และทำการเปรียบเทียบการวิเคราะห์คลัสเตอร์ FCM แบบปกติและแบบที่ปรับปรุงด้วยวิธีที่กล่าวมา โดยได้ทำการทดลองกับภาพถ่ายดาวเทียมจริงๆ

จากสมการที่ (3.1) เราจะพบว่าเซ็นเตอร์ของทุกๆคลัสเตอร์ V_i คือค่าถ่วงน้ำหนักเฉลี่ยของแซมเปิลในเซตข้อมูล Y ตัวถ่วงน้ำหนักที่ใช้กับแต่ละแซมเปิลก็คือค่าความเป็นสมาชิกใน U^k คลัสเตอร์ ในกระบวนการ FCM โดยทั่วไปนั้นจะคำนวณค่าต่างๆเหล่านี้ได้จากเซตข้อมูลทั้งหมดและเมมเบอร์ชิพเมตริกซ์

ในหัวข้อนี้จะกล่าวถึงวิธีการปรับปรุงกระบวนการวิเคราะห์คลัสเตอร์โดยมีลักษณะการทำงานคือจะทำการคำนวณเซ็นเตอร์ใหม่ของแต่ละคลัสเตอร์และปรับปรุงค่าเมมเบอร์ชิพเมตริกซ์ไปพร้อมๆกันทุกครั้งที่มีการคำนวณในแต่ละพิกเซล ด้วยวิธีนี้จะสามารถลดจำนวนข้อมูลที่ผ่านมาเข้าสู่ระบบ ผลลัพธ์จากการวิเคราะห์คลัสเตอร์ด้วยกระบวนการนี้นั้นไม่ได้ลดจำนวนการวนรอบการทำงานของ FCM แต่จะสามารถลดเวลาที่ใช้ในการคำนวณในแต่ละรอบการทำงานลงได้

ขั้นตอนแรกของการปรับปรุง FCM ด้วยวิธีที่เสนอนี้จะเริ่มจากการสร้างเซตของข้อมูลขึ้นมา 2 เซตคือเมตริกซ์ $P=\{P_i\}$ ขนาด $c \times n$ และเวกเตอร์ q ใน c มิติ ค่าเริ่มต้นนี้หามาได้จากตัวเศษและตัวส่วนจากสมการที่ (3.1) เมื่อทำการสุ่มเมมเบอร์ชิพเมตริกซ์ขึ้นมาแล้ว สมการที่ (3.1) จะสามารถเขียนแทนได้ด้วยสมการที่ (3.2)

$$v_i = \frac{p_i}{q_i} \quad (3.4)$$

เมื่อ P_i คือเวกเตอร์ n มิติ และ q_i คือค่าสเกลลาร์ สัญลักษณ์ตัวห้อยคอก i ถูกนำมาใช้เพื่อหลีกเลี่ยงไม่ให้เกิดความสับสนระหว่างเวกเตอร์กับสเกลลาร์

ทุกๆครั้งที่มีการคำนวณในแต่ละ u_{ki} ในเมมเบอร์ชิพเมตริกซ์ จะมีการเพิ่มค่า P_i หรือตัวเศษของสมการที่ (3.1) ด้วยค่าตามสมการที่ (3.5)

$$\left((u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \right) y_k \quad (3.5)$$

และจะต้องเพิ่มค่าของ q_i หรือตัวส่วนของสมการที่ (3.1) ด้วยค่าดังสมการที่ (3.6)

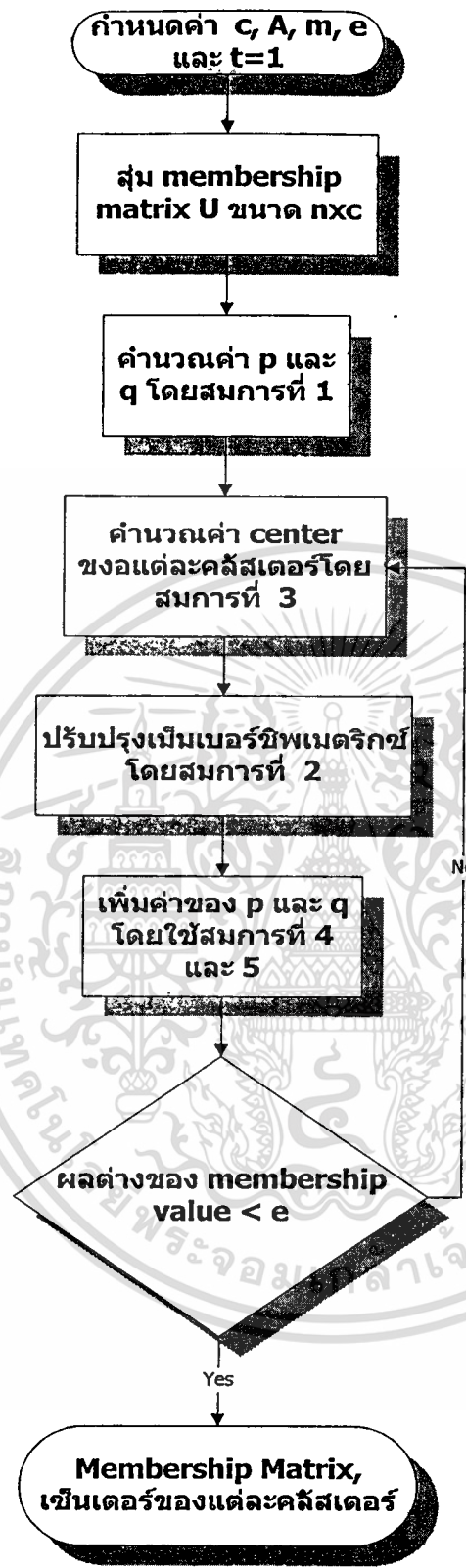
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\left(u_{ki}^{(t+1)}\right)^m - \left(u_{ki}^{(t)}\right)^m \quad (3.6)$$

การเพิ่มขึ้นของค่า P และ q นี้จะสะสมเพิ่มขึ้นเรื่อยๆทุกครั้งที่เมมเบอร์ชิพเมตริกซ์ถูกอัปเดต พจนกลับมารเริ่มต้นคำนวณในรอบถัดไปค่าเซ็นเตอร์ของแต่ละคลัสเตอร์ก็จะถูกคำนวณขึ้นมาใหม่อีกครั้งโดยใช้สมการที่ (3.2) ในการคำนวณ

กระบวนการการปรับปรุง FCM ด้วยวิธีนี้สามารถอธิบายเป็นขั้นตอนได้ดังนี้

1. กำหนดค่า for c, A, m, E, และลูปเคาน์เตอร์ t=1
 2. ทำการสุ่มเมมเบอร์ชิพเมตริกซ์ $U^{(0)}$ ขนาด $N \times c$
 3. คำนวณค่าเริ่มต้นของ P และ q โดยใช้สมการที่ (3.2)
 4. คำนวณคลัสเตอร์เซ็นเตอร์โดยใช้สมการที่ (3.4)
 5. ปรับปรุงเมมเบอร์ชิพเมตริกซ์โดยใช้สมการที่ (3.3) และทุกครั้งที่ทำการอัปเดตเมมเบอร์ชิพเมตริกซ์ก็จะทำการเพิ่มค่าของ P และ q โดยใช้สมการที่ (3.5) และสมการที่ (3.6)
 6. ถ้าค่าสูงสุดของ $\left\{ \left| u_{ki}^{(t)} - u_{ki}^{(t-1)} \right| \right\} > \epsilon$ ก็เพิ่มค่า t และกลับไปทำขั้นตอนที่ 4.
- ไฟล์โค้ดแกรมของการปรับปรุง FCM แสดงดังรูปที่ 3.4



รูปที่ 3.4 โฟลว์ไดอะแกรมของการปรับปรุง FCM

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

ผลการทดลอง

ในวิทยานิพนธ์ฉบับนี้ได้ทำการแบ่งการทดลองออกเป็นสามส่วน ส่วนแรกคือการทดลองบีบอัดข้อมูลรูปภาพด้วยวิธีเวกเตอร์ควอนไทซ์เซชันโดยอาศัยข้อมูลจากการวิเคราะห์คลัสเตอร์ด้วยวิธีพีชชีมีนส์ ส่วนที่สองคือการทดลองเพิ่มความเร็วในการทำงานของกระบวนการวิเคราะห์คลัสเตอร์ด้วยวิธีพีชชีมีนส์ และในส่วนที่สามจะทำการทดลองสร้างโค้ดบุคมาตรฐานขึ้นเพื่อนำไปใช้กับข้อมูลภาพหลายแบนด์หลายๆภาพที่ต่างกัน

4.1 ผลการทดลองบีบอัดข้อมูลภาพหลายแบนด์โดยวิธีเวกเตอร์ควอนไทซ์เซชันที่ใช้หลักการของ FCM คลัสเตอร์ริง

ในการทดลองนี้ผู้เขียนได้ทำการเขียนโปรแกรมบีบอัดข้อมูลภาพหลายแบนด์ด้วยวิธีการเวกเตอร์ควอนไทซ์เซชันโดยอาศัยข้อมูลจากการวิเคราะห์คลัสเตอร์ด้วยวิธีพีชชีมีนส์มาช่วยในการออกแบบโค้ดบุค ซึ่งการทดลองในหัวข้อนี้ได้ถูกแบ่งออกเป็นสองแบบย่อยๆตามรูปแบบการฟอร์มเวกเตอร์จากข้อมูลภาพอินพุท

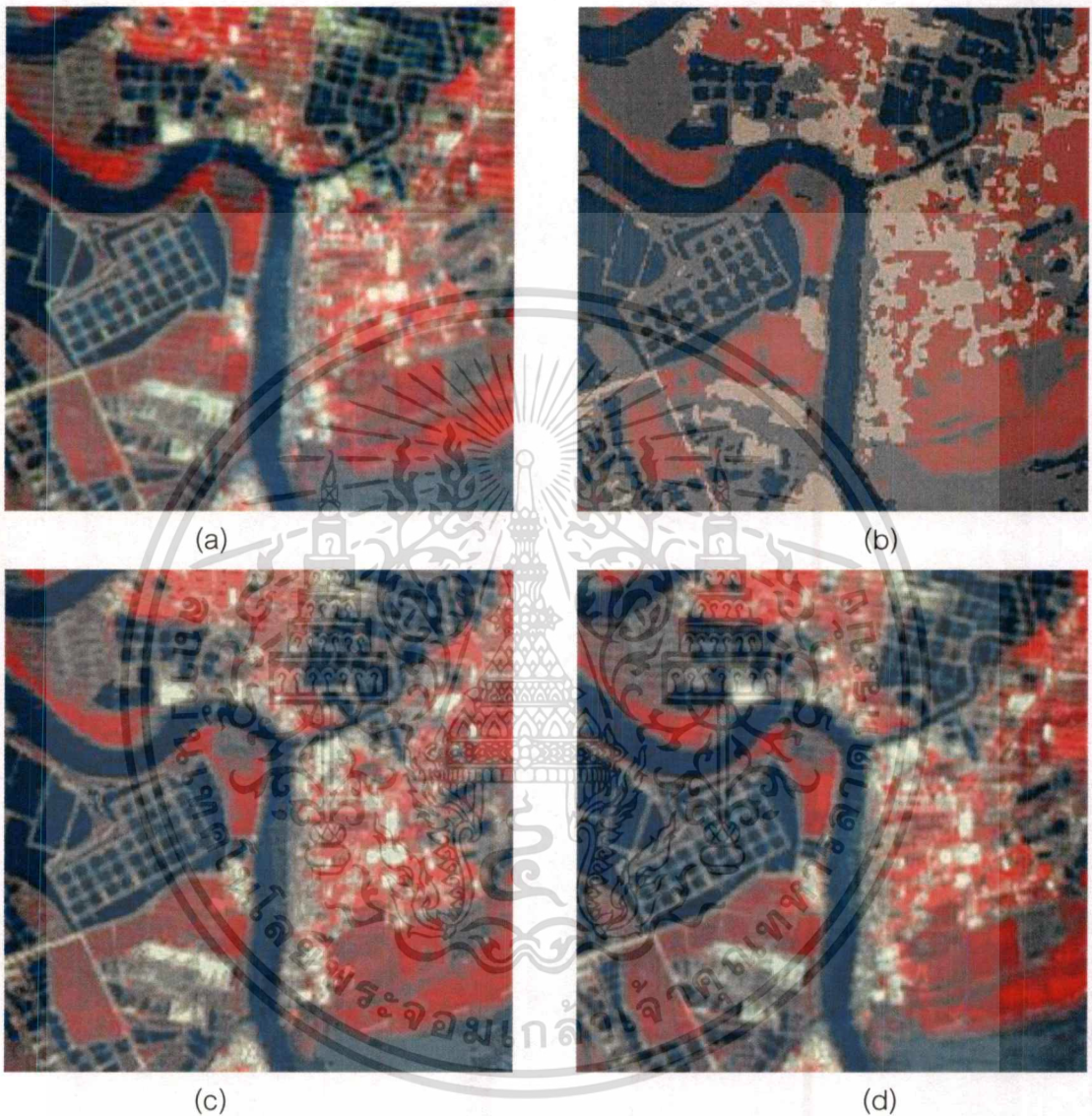
4.1.1 ผลการทดลองการบีบอัดข้อมูลด้วยวิธีเวกเตอร์ควอนไทซ์เซชันโดยใช้วิธีการฟอร์มเวกเตอร์แบบอินเตอร์แบนด์

ผลการทดลองบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทซ์เซชันโดยใช้การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์กับถ่ายควเทียม ops ขนาด $256 \times 256 \times 3$ จำนวนคลัสเตอร์ที่ใช้เท่ากับ 4, 16 และ 64 แสดงได้ดังรูปที่ 4.1

ตารางที่ 4.1 แสดงการคำนวณค่าคุณสมบัติต่างๆของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทซ์เซชันที่ใช้การฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์

Number of clusters	Bpp	PSNR (dB)	MSE
4	2.0016	22.0651	404.1790
16	4.068	27.4127	117.9196
64	6.0293	31.2498	48.7638

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.1 (a) ภาพต้นแบบ (b) ภาพที่ถูกบีบอัดด้วยวิธีการเวกเตอร์ควอนไทเซชัน โดยการใช้ฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์ ใช้จำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์ (c) ภาพที่ถูกบีบอัดด้วยวิธีการเวกเตอร์ควอนไทเซชัน โดยการใช้ฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์ ใช้จำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์ (d) ภาพที่ถูกบีบอัดด้วยวิธีการเวกเตอร์ควอนไทเซชัน โดยการใช้ฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์ ใช้จำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1.2 ผลการทดลองบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันโดยใช้วิธีการฟอร์มเวกเตอร์แบบอินทราแบนด์

ผลการทดลองที่ได้จากการบีบอัดภาพถ่ายดาวเทียม ops ต้นแบบขนาด 256X256X3 โดยใช้วิธีการฟอร์มเวกเตอร์แบบอินทราแบนด์ จำนวนอิลิเมนต์ของเวกเตอร์เท่ากับ 2, 3 และ 4 และจำนวนคลัสเตอร์เท่ากับ 4, 16 และ 64 ตามลำดับแสดงได้ดังรูป 4.2, 4.3 และ 4.4

ตารางที่ 4.2, 4.3 และ 4.4 แสดงการคำนวณค่าคุณสมบัติต่างๆของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์โดยใช้จำนวนเป็น 4, 16 และ 64 ตามลำดับ

ตารางที่ 4.2 แสดงคุณสมบัติของการบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์

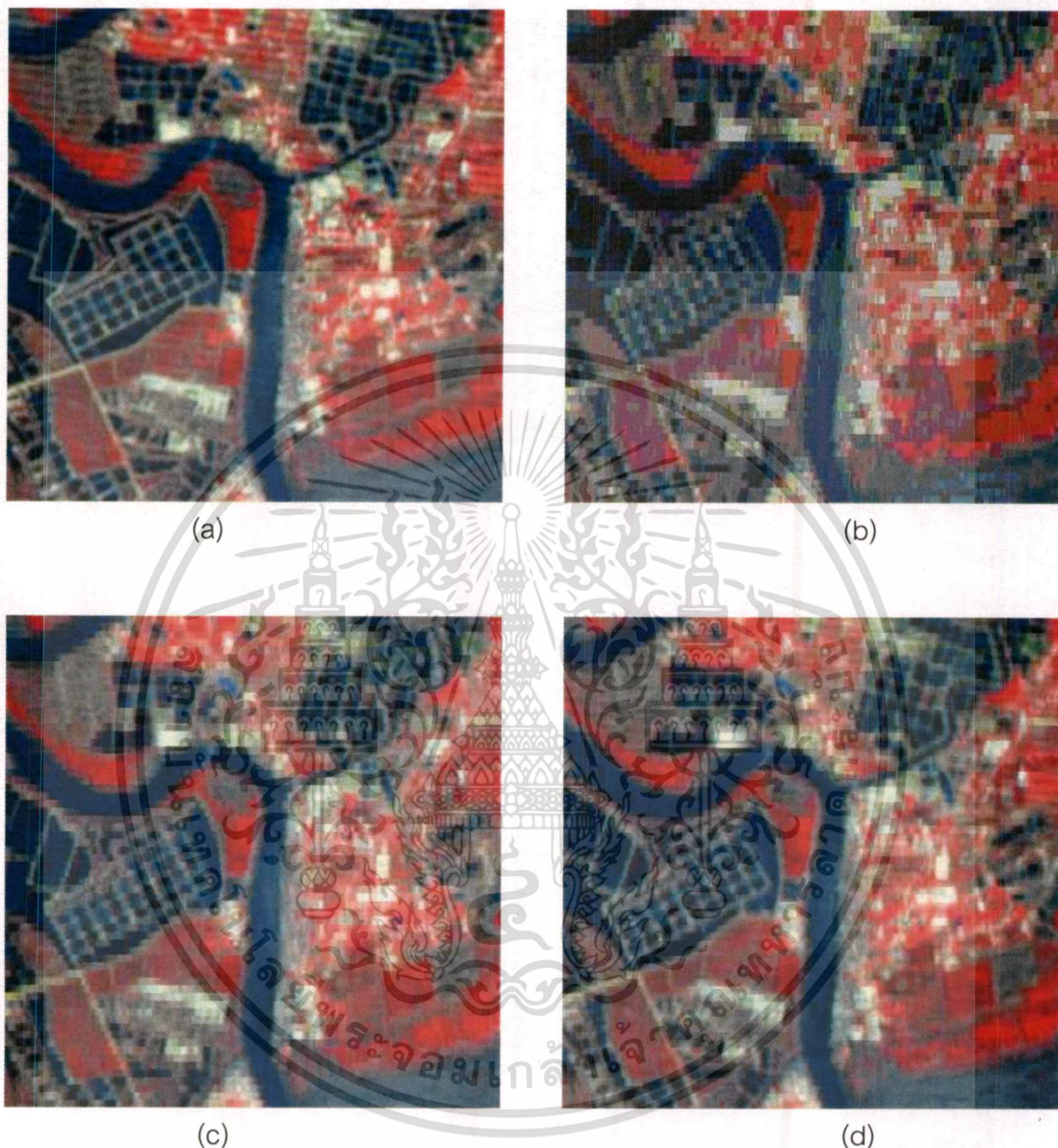
จำนวนมิติของเวกเตอร์ที่ใช้	2	3	4
PSNR(dB)	24.593	23.868	23.303
MSE	225.82	266.859	303.92
bpp	3.001	2.001	1.502

ตารางที่ 4.3 แสดงคุณสมบัติของการบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์

จำนวนมิติของเวกเตอร์ที่ใช้	2	3	4
PSNR(dB)	31.633	28.845	27.465
MSE	44.640	84.838	116.556
bpp	6.004	4.006	3.008

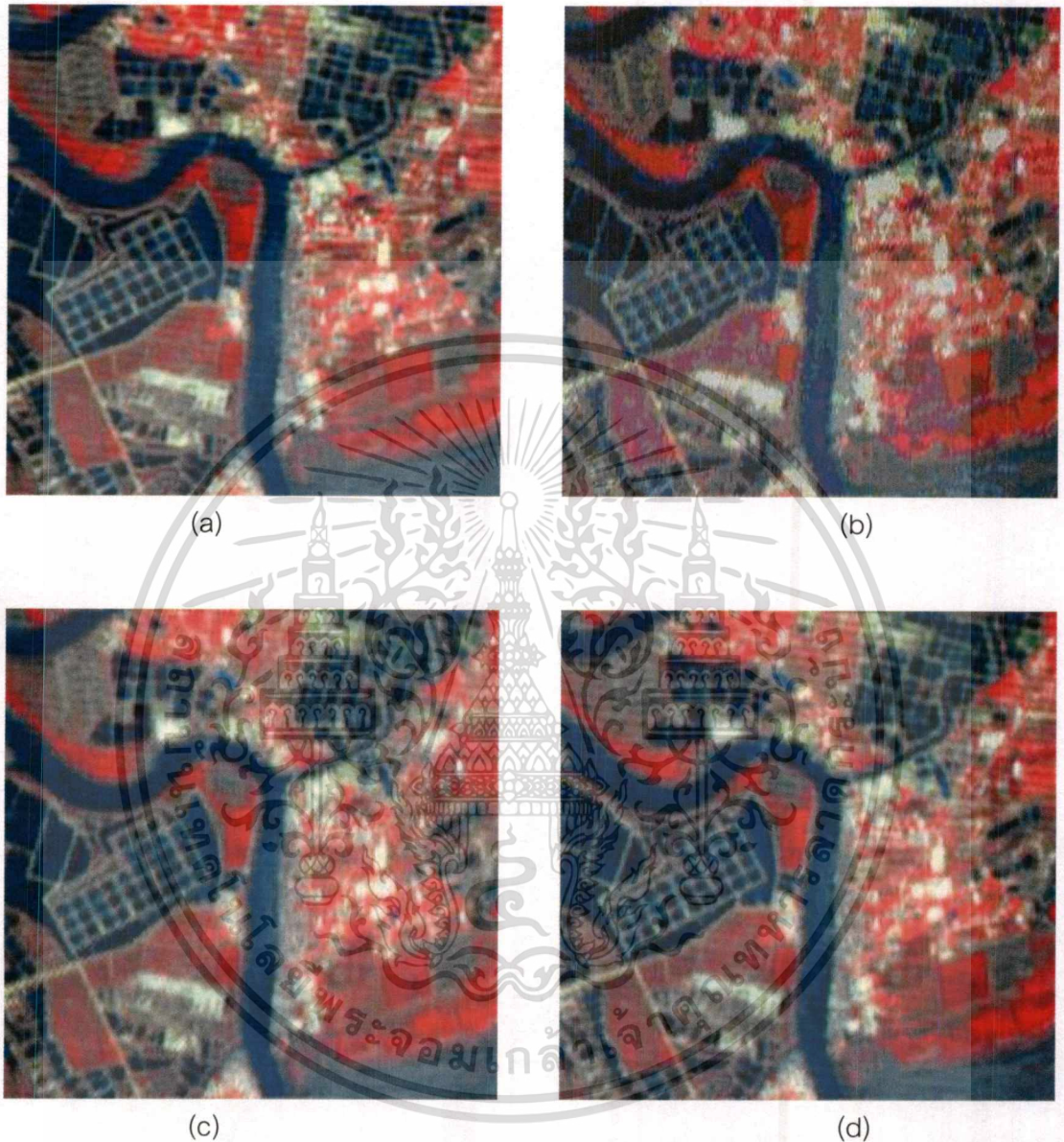
ตารางที่ 4.4 แสดงคุณสมบัติของการบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์โดยใช้จำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์

จำนวนมิติของเวกเตอร์ที่ใช้	2	3	4
PSNR(dB)	35.949	32.736	30.829
MSE	16.528	34.635	53.723
bpp	9.015	6.023	4.531

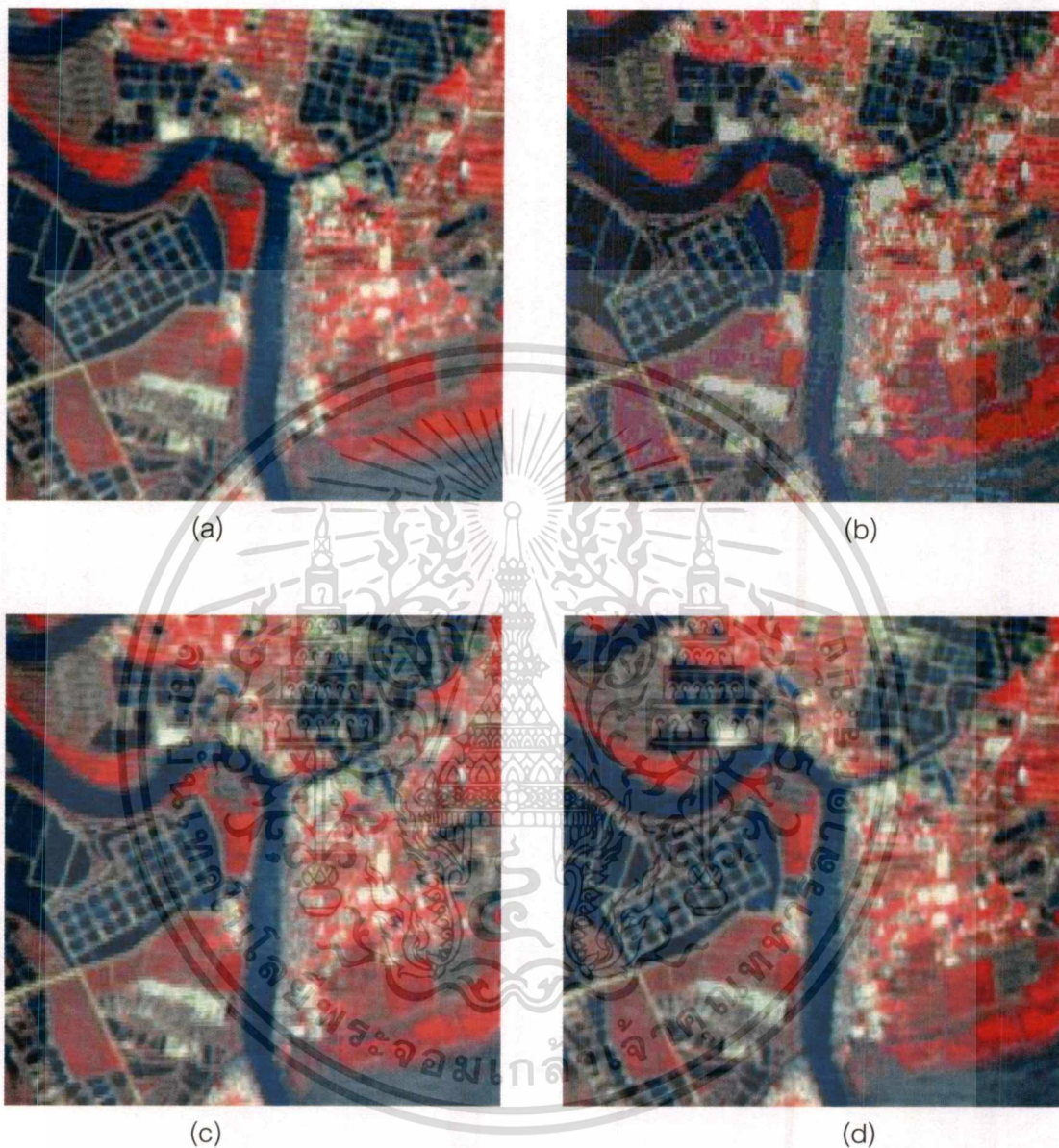


รูปที่ 4.2 (a) ภาพถ่ายดาวเทียม ops ดั้งเดิม (b) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิเลเมนต์เท่ากับ 4 อิเลเมนต์และจำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์ (c) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิเลเมนต์เท่ากับ 4 อิเลเมนต์และจำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์ (d) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิเลเมนต์เท่ากับ 4 อิเลเมนต์และจำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 (a) ภาพถ่ายดาวเทียม ops ดันแบบ (b) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 3 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์ (c) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 3 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์ (d) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 3 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์



รูปที่ 4.4 (a) ภาพถ่ายดาวเทียม ops ต้นแบบ (b) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 2 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 4 คลัสเตอร์ (c) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 2 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 16 คลัสเตอร์ (d) ภาพที่ได้จากการบีบอัดโดยใช้จำนวนเวกเตอร์อิลิเมนต์เท่ากับ 2 อิลิเมนต์และจำนวนคลัสเตอร์เท่ากับ 64 คลัสเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการทดลองค่า PSNR ของการบีบอัดข้อมูลภาพหลายแบนด์โดยใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์มีค่าดีกว่าการฟอร์มเวกเตอร์แบบอินเตอร์แบนด์เมื่อใช้จำนวนอิลิเมนต์ต่อหนึ่งเวกเตอร์เท่าๆกัน แต่ถ้าใช้จำนวนอิลิเมนต์ของแต่ละเวกเตอร์ไม่เท่ากัน ถ้าใช้จำนวนมิติของเวกเตอร์น้อยกว่าจะให้ค่า PSNR ที่ดีกว่า

4.2 ผลการทดลองการเพิ่มความเร็วยของกระบวนการ FCM

ในวิทยานิพนธ์นี้ได้ทำการทดลองนำการวิเคราะห์คลัสเตอร์ด้วยวิธี FCM แบบที่ใช้กันอยู่โดยทั่วไปและวิธีที่ปรับปรุงโดยกระบวนการที่นำเสนอข้างต้นมาทดลองกับภาพถ่ายดาวเทียมโดยใช้โปรแกรม Matlab รันบนเครื่องคอมพิวเตอร์ที่ใช้ CPU Intel Celeron ที่ความเร็ว 400 เม็กกะเฮิร์ต แรม 64 เม็กกะไบท์ และได้ทำการเปรียบเทียบผลการทดลองจากทั้งสองวิธี ตารางที่ 4.5 ได้แสดงการเปรียบเทียบเวลาการทำงานเฉลี่ยในแต่ละรอบการทำงาน โดยทำการแบ่งข้อมูลภาพถ่ายดาวเทียม JERS1/OPS 3 แบนด์ในรูปที่ 4.5 (256x256 พิกเซล) ออกเป็น 3, 4 และ 5 คลัสเตอร์ตามลำดับ ตารางที่ 4.5 แสดงผลการทดลองเปรียบเทียบเวลาเฉลี่ยที่ใช้ในแต่ละรอบการทำงานของ FCM



รูปที่ 4.5 ภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.5 แสดงผลการทดลองเพิ่มความเร็วกระบวนการฟัซซี่ซีมีนส์

Average execution time per iteration (min)					
3 clusters		4 clusters		5 clusters	
Original	Modified	Original	Modified	Original	Modified
FCM	FCM	FCM	FCM	FCM	FCM
31.84	17.73	37.96	22.47	47.825	31.45

จากตารางที่ 4.5 เห็นได้ว่ากระบวนการปรับปรุง FCM ที่ได้นำเสนอไปนั้นสามารถลดเวลาที่ใช้ในการทำงานในแต่ละรอบของกระบวนการ FCM ได้ประมาณ 40% อินพุตพารามิเตอร์ที่ใช้ในทุกๆกรณีคือ $m=2$, $A=I$, $c=3$, และ $\epsilon=0.01$ สำหรับภาพ $512 \times 512 \times 3$ เมื่อแบ่งออกเป็น 3 คลัสเตอร์ด้วยวิธี FCM แบบเก่าต้องใช้เวลา 495.96 นาทีในแต่ละการวนรอบ แต่วิธีการที่ปรับปรุงแล้วใช้เวลา 284.4 นาที ซึ่งจะพบว่าในการนี้กระบวนการปรับปรุง FCM สามารถลดระยะเวลาการทำงานในแต่ละรอบลงได้ประมาณ 43%













4.3 ผลการทดลองการคำนวณโค้ดบุคมาตฐานเพื่อใช้ในการบีบอัดข้อมูลภาพหลายแบนด์

เนื่องจากจากผลการทดลองในหัวข้อที่ 4.2 จะพบว่าเวลาที่ต้องใช้ในขั้นตอนการคำนวณโค้ดบุคสำหรับภาพหลายแบนด์แต่ละภาพนั้นต้องใช้เวลาามาก จึงไม่เป็นการสะดวกที่จะทำการคำนวณค่าโค้ดบุคขึ้นใหม่ทุกครั้ง สำหรับวิทยานิพนธ์เล่มนี้จึงได้ทำการคำนวณโค้ดบุคมาตฐานขึ้นมาเพื่อใช้กับข้อมูลภาพถ่ายดาวเทียม ops โดยการนำข้อมูลจากภาพถ่ายดาวเทียม ops จำนวน 10 ภาพที่แตกต่างกันมารวมกัน โดยแต่ละภาพเป็นภาพขนาด 256×256 ใน 3 แบนด์ แล้วทำการวิเคราะห์โค้ดบุคด้วยวิธีการฟัซซี่ซีมีนส์ เพื่อหาเซตของเวกเตอร์มาตฐานที่เหมาะสมที่จะนำมาใช้กับข้อมูลภาพถ่ายดาวเทียม ops โดยจำนวนโค้ดบุคที่ใช้ในการทดลองใช้จำนวนโค้ดบุคเท่ากับ 64 โค้ดบุค จำนวนอิลิเมนต์ของเวกเตอร์ที่ใช้ในกรณีการฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์เท่ากับ 3 อิลิเมนต์ ซึ่งหมายความว่าแต่ละเวกเตอร์ของภาพอินพุตจะถูกเข้ารหัสด้วยโค้ดเวิร์ดที่มีขนาด 6 บิต ค่า bpp ที่ได้จะมีค่าเท่ากับ 6.0293 โดยในการทดลองนั้นได้นำค่าโค้ดบุคมาตฐานที่คำนวณได้มาทำการบีบอัดภาพต้นแบบทั้ง 10 ภาพและทำการคำนวณค่า PSNR โดยผลการทดลองแสดงได้ดังตารางที่ 4.6

ตารางที่ 4.6 แสดงผลการทดลองบีบอัดภาพถ่ายดาวเทียม ops โดยใช้โค้ดบुकมาตรฐาน

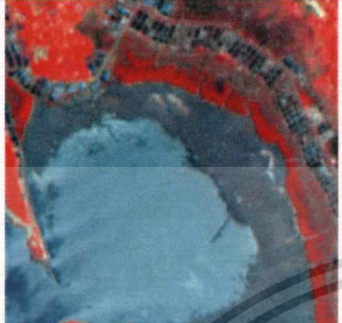
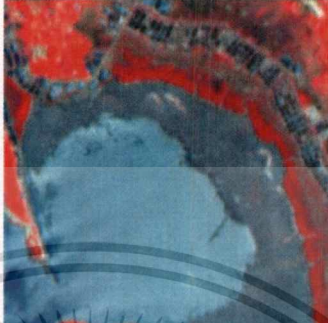
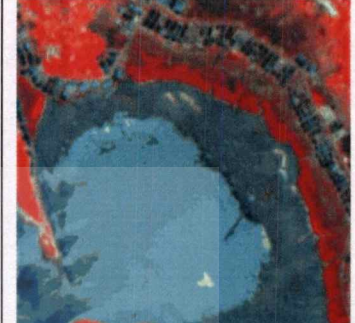









ภาพที่	PSNR	
	ฟอร์มเวกเตอร์ด้วยวิธีอินทราแบนด์	ฟอร์มเวกเตอร์ด้วยวิธีอินเตอร์แบนด์
1	31.0117	30.2542
2	31.2795	29.7997
3	33.0575	31.7525
4	31.9720	31.3827
5	35.1120	30.8360
6	34.4422	31.4047
7	34.4174	32.1505
8	34.7016	32.4231
9	33.8161	31.7143
10	36.0466	32.6897

จากการใช้โค้ดบुकมาตรฐานในการบีบอัดข้อมูลภาพถ่ายดาวเทียมทั้ง 10 ภาพพบว่าสามารถลดระยะเวลาในการบีบอัดข้อมูลภาพได้มาก โดยในการบีบอัดข้อมูลภาพแต่ละภาพนั้นจะใช้เวลาในการบีบอัดเพียง 3-4 นาทีเท่านั้น

รูปที่	ภาพต้นแบบ	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินทราแบนด์	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินเตอร์แบนด์
1			
2			
3			
4			

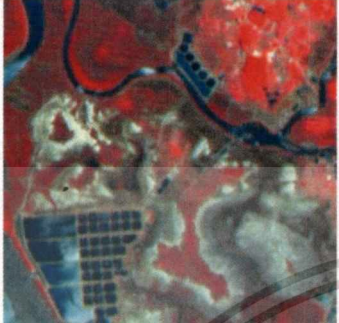

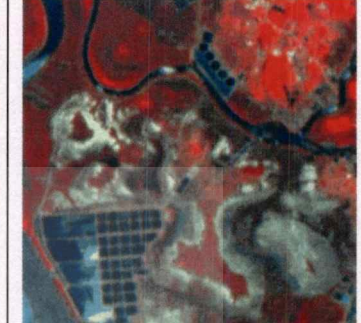


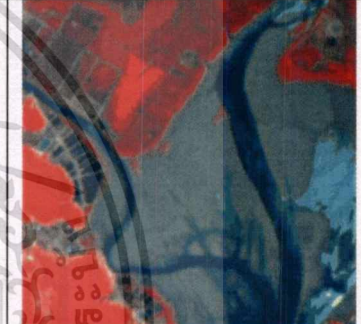
รูปที่ 4.6 ผลการทดลองบีบอัดภาพถ่ายดาวเทียม ops โดยใช้โค้ดอนุกรมมาตรฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่	ภาพต้นแบบ	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินทราแบนด์	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินเตอร์แบนด์
5			
6			
7			
8			

รูปที่ 4.6 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่	ภาพต้นแบบ	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินทราแบนด์	ภาพที่บีบอัดจากการฟอร์ม เวกเตอร์แบบอินเตอร์แบนด์
9			
10			

รูปที่ 4.6 (ต่อ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

สรุปผลการวิจัยและข้อเสนอแนะ

5.1 สรุปผลการวิจัย

จากผลการทดลองในหัวข้อ 4.1 จะเห็นได้ว่าเราสามารถบีบอัดข้อมูลภาพหลายแบนด์หรือภาพสีได้โดยใช้การควอนไทซ์ภาพแบบเวกเตอร์ซึ่งใช้หลักการของการคลัสเตอร์แบบฟัชชีซีมีนส์ โดยการฟอร์มเวกเตอร์ที่จะนำมาใช้ในการควอนไทซ์นั้นมี 2 วิธีคือ การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์และแบบอินทราแบนด์

การบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์นั้นปริมาณการบีบอัดและความผิดเพี้ยนของภาพที่เกิดจากการควอนไทซ์จะขึ้นอยู่กับจำนวนคลัสเตอร์ที่ใช้ โดยหากจำนวนคลัสเตอร์ที่ใช้มีค่าน้อยปริมาณการบีบอัดจะมีความคมชัดภาพเอาท์พุทที่ได้จะมีความเพี้ยนไปจากภาพต้นแบบสูง (ค่า PSNR มีค่าต่ำ) แต่ถ้าจำนวนคลัสเตอร์ที่ใช้มีค่ามากภาพเอาท์พุทที่ได้จะมีความเพี้ยนต่ำ (ค่า PSNR มีค่าสูง) แต่ปริมาณการบีบอัดภาพจะลดลง

สำหรับคุณสมบัติของการบีบอัดข้อมูลภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์นั้นปริมาณการบีบอัดภาพและค่าความเพี้ยนที่เกิดจากการควอนไทซ์จะขึ้นอยู่กับพารามิเตอร์ 2 ตัวคือ จำนวนคลัสเตอร์ที่ใช้และจำนวนอิลิเมนต์ที่ใช้แทนในแต่ละเวกเตอร์ โดยเมื่อเปรียบเทียบที่จำนวนคลัสเตอร์เดียวกันแล้วหากเราใช้จำนวนอิลิเมนต์ในแต่ละเวกเตอร์มาก ค่าความเพี้ยนของภาพเอาท์พุทจะมีค่าสูง แต่ถ้าหากจำนวนอิลิเมนต์ในแต่ละเวกเตอร์มีค่าต่ำ ค่าความเพี้ยนของภาพเอาท์พุทจะมีค่าต่ำลงด้วย

เมื่อเปรียบเทียบคุณสมบัติการบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่ใช้การฟอร์มเวกเตอร์แบบอินเตอร์แบนด์และแบบอินทราแบนด์จะเห็นว่าที่จำนวนคลัสเตอร์เดียวกันและที่จำนวนอิลิเมนต์ที่ใช้แทนแต่ละเวกเตอร์เท่ากัน คุณสมบัติการบีบอัดภาพที่ได้จะมีค่าใกล้เคียงกัน โดยการบีบอัดภาพที่ใช้การฟอร์มเวกเตอร์แบบอินทราแบนด์จะให้ค่า PSNR ที่ดีกว่าเล็กน้อย

จากผลการทดลองในหัวข้อ 4.2 จะเห็นได้ว่าเราสามารถเพิ่มความเร็วในการทำงานแต่ละรอบของกระบวนการฟัชชีซีมีนส์ได้ด้วยวิธีที่นำเสนอ โดยการปรับปรุงกระบวนการคลัสเตอร์แบบฟัชชีซีมีนส์นี้มิได้เป็นการลดจำนวนรอบในการทำงานของ FCM แต่เป็นการลดระยะเวลาในการคำนวณค่าเมมเบอร์ชิพเมตริกและเวกเตอร์เฉลี่ยในแต่ละรอบการวนลูป ซึ่งจากผลการทดลองพบว่าการปรับปรุงกระบวนการฟัชชีซีมีนส์ที่นำเสนอสามารถลดความเร็วในแต่ละรอบการทำงานลงได้ประมาณ 40 % เมื่อเทียบกับวิธีการฟัชชีซีมีนส์ที่ใช้กันอยู่ทั่วไป

เนื่องจากการคำนวณ โค้ดบุคด้วยวิธีคลัสเตอร์แบบพีชชีมีนส์นั้นเป็นกระบวนการที่มีการคำนวณค่าต่างๆอย่างซับซ้อนทำให้ต้องใช้เวลานาน การคำนวณ โค้ดบุคมาตรฐานจึงเป็นวิธีหนึ่งที่สามารถลดระยะเวลาในการบีบอัดภาพหลายแบนด์แต่ละภาพได้เป็นอย่างดี

5.2 ข้อเสนอแนะ

ในการบีบอัดภาพด้วยวิธีเวกเตอร์ควอนไทเซชันที่อาศัยหลักการการจำแนกข้อมูลแบบพีชชีมีนส์นั้นเป็นกระบวนการที่ซับซ้อนและต้องใช้การคำนวณเป็นอย่างมาก ทำให้เวลาที่ต้องใช้ในการวิเคราะห์คลัสเตอร์มีเวลานาน ถึงแม้เราจะประยุกต์ใช้วิธีการปรับปรุงกระบวนการพีชชีมีนส์ที่นำเสนอไปแล้ว แต่กระบวนการดังกล่าวก็ยังคงต้องใช้เวลานานหากจำนวนคลัสเตอร์ที่ใช้มีจำนวนมาก ดังนั้นหากผู้อ่านต้องการทำการวิจัยและพัฒนาวิธีการต่างๆภายในวิทยานิพนธ์ฉบับนี้ต่อไป ผู้อ่านไม่ควรใช้จำนวนคลัสเตอร์มากเกินไปนัก เพราะจะทำให้ระยะเวลาที่ต้องใช้ในกระบวนการวิเคราะห์คลัสเตอร์มีเวลานานมาก และคอมพิวเตอร์ที่ใช้ในการทดลองควรมีความเร็วสูงและมีขนาดของแรมมาก หรืออีกวิธีหนึ่งที่จะสามารถลดระยะเวลาในการวิเคราะห์คลัสเตอร์ได้ก็คือการวิเคราะห์คลัสเตอร์กับกลุ่มข้อมูลที่สุ่มมาจากภาพต้นแบบ (Subsampling) จะเป็นการลดจำนวนเวกเตอร์อินพุตที่นำมาใช้ในการวิเคราะห์และจะสามารถลดเวลาในการวิเคราะห์คลัสเตอร์ลงได้

บรรณานุกรม

- Bezdek, J.C., Ehrlich, R., and Full, W. 1984. FCM : The fuzzy c-means clustering algorithm. Computers and Geosciences, 10, pp. 191-203.
- Cannon, R.L., Dave, J.V. and Bezdek, J.C. 1986. Efficient implementation of the fuzzy c-means clustering algorithms. IEEE Trans. Pattern analysis and Machine Intelligence, 8, pp. 248-255.
- Clark, R.J. 1995. Digital Compression of Still Images and Video. London : Academic Press.
- Hsieh, C.H. 1992, DCT based codebook design for vector quantization of images. IEEE Trans. Circuits and System, 2, pp. 40-1-409.
- Hsieh, C.H., Lu. P.C. and Chung. J.C. 1991. Fast codebook generation algorithm for vector quantization of images. Pattern Recogn. Lett., 12, pp. 605-609.
- Linde, Y., Buzo, A. and Gray, R.M. 1980. An algorithm for vector quantization design. IEEE Trans. Commun., 28, pp. 84-95.
- Pearson, D. 1991. Image Processing. London : McGraw-Hill.
- Pitas, I. 1993. Digital Image Processing Algorithm. New York : Prentice Hall International.
- Wu, X. and Guan, L. 1994. Acceleration of the LBG algorithm. IEEE Trans. Commun., 42, pp.1518-1523.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MULTISPECTRAL IMAGE COMPRESSION USING FCM-BASED VECTOR QUANTIZATION

Uthai SANGTHONGPRAOW and Yuttapong RANGSANSERI

Department of Telecommunications Engineering, Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520

Tel: (66-2)326-9967, Fax: (66-2)326-9086

E-mail: mr_uthai@hotmail.com, kryuttha@kmitl.ac.th

THAILAND

ABSTRACT: Image compression is the process of reducing the number of bits required to represent an image. Vector Quantization method (VQ) is a method to deal with this operation. With this method a set of data points is encoded by a reduced set of reference vectors (the codebook). VQ is useful in compressing data that arises in wide range applications and it can achieve better compression performance than any conventional coding techniques which based on the encoding of scalar quantities. This paper presents a multispectral image compression method using Vector Quantization technique based on Fuzzy c-Means (FCM). In this method we first use FCM algorithm to generate a good initial codebook. A modified version of FCM is used to reduce the computational time. Experimental results are shown to illustrate the performance of the proposed method.

1. INTRODUCTION

Image Compression is a process of reducing the number of bits required to represent an image. There are several methods to compress an image data. One popular method is Vector Quantization (VQ). VQ was used for compressing data in wide range applications, including image processing speech processing facsimile transmission and weather satellites. In this method the training vectors in the image are encoded by a reduced set of reference vectors (the codebook). From rate distortion theory, it can be shown that VQ can achieve better compression performance than any conventional coding technique which based on the encoding of scalar quantities. The objective of VQ is to design the codebook. An optimum VQ system is that one uses a codebook that yields the least average distortion of the reconstructed image. Other methods for designing codebooks have been proposed in. The most popular one is the LBG algorithm. In this paper we use FCM to design the codebook in Vector Quantization process. The major problem of this method is the huge execution time in the FCM process. So, this paper will also present the modification on the FCM algorithm that can reduce the execution time in each iteration.

2. FUZZY C-MEANS CLUSTERING TECHNIQUE

The fuzzy c-means (FCM) algorithm is an iterative clustering method that used to partition a data set. The objective of FCM segmentation is to compute the cluster centers and generate the class membership matrix. This class membership matrix is a cxN matrix, where c is the number of groups and N is the number of samples of image in n -space. Each column of the class membership matrix is the distribution of the

class attribute of its corresponding sample. Each row of the class membership matrix is the membership value of each sample to be a member of that particular cluster.

An optimal fuzzy c-partition is one that minimizes the generalized least-squared errors function. We can explain the fuzzy clustering by the following equation.

$$\text{Minimize: } J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ki})^m \|y_k - v_i\|^2 A$$

Where : $Y = \{y_1, y_2, y_3, \dots, y_N\} \subset \mathbb{R}^n$ is the data set,
 c is the number of clusters in Y : $2 \leq c < n$,
 m is a weighting exponent: $1 \leq m < \infty$,
 $U = \{u_{ki}\}$ is the fuzzy c-partition of Y ,
 $\|y_k - v_i\|_A$ is an induced a -norm on \mathbb{R}^n , and,
 A is a positive-definite ($n \times n$) weight matrix.

The weighting exponent m has the effect of reducing the squared distance error by an amount that depends on the observation's membership in the cluster. As $m \rightarrow 1$, the partitions that minimize J_m become increasingly hard. Conversely, higher values of m tend to soften a samples cluster membership, and the partition becomes increasingly blurred. Generally m must be selected by experimental means.

The first step of FCM algorithm is generate an initial random membership matrix and use this random membership matrix as weight of each samples to belong to each cluster. Then computes the centroid of each cluster. The new cluster centers are used to update the membership matrix. The updated membership matrix is compared with the previous ones. If the difference is greater than some threshold, then another iteration is computed, otherwise the algorithm is stopped. The algorithm is shown below. The notation $x^{(t)}$ signifies the value of variable x at iteration t .

The FCM Algorithm

1. set value for c, A, m, ϵ , and the loop counter $t = 1$
2. create a random $N \times c$ membership matrix U
3. compute each cluster centroid by equation

$$v_i^{(t)} = \sum_{k=1}^N (u_{ki}^{(t)})^m y_k / \sum_{k=1}^N (u_{ki}^{(t)})^m \quad (1)$$

4. update the membership matrix by equation

$$u_{ki}^{(t+1)} = \left(\sum_{j=1}^c \left(\frac{d_{ki}}{d_{kj}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (2)$$

where $d_{ki} = \|y_k - v_i^{(t)}\|_A$

5. If $\max \{u_{ki}^{(t)} - u_{ki}^{(t-1)}\} > \epsilon$ increment t and go to step 3.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. THE MODIFIED FUZZY C-MEANS ALGORITHM

From equation (1) we can find that each cluster center v_i is computed as the weighted average of samples in the data set Y . The weight used for each sample is that sample's membership in the i^{th} cluster. In the original algorithm this is computed by performing a pass over the entire data set and membership matrix.

The following describes an improvement to the algorithm whereby the computation of cluster centers is performed in sequence with the membership matrix updating. This effectively eliminates an entire pass of the data set. The outcome is not a decrease in the number of iterations required for convergence, but decrease in the time of each iteration.

We maintain two extra data structures, a $c \times n$ matrix, $P = \{p_i\}$, and a vector q of length c . We can obtain these two matrix from the first iteration when the random membership matrix is created. If we represent the numerator and denominator of equation (1) by P and q , respectively, then keep them as initial values. So now the equation (1) is replaced by equation (3)

$$v_i = p_i / q_i \quad (3)$$

where p_i is a vector of length n , and q_i is a scalar. The dot- i subscript is used to avoid confusion between vectors and scalars.

Each time an element u_{ki} of the membership matrix is computed there is an increment in the numerator of equation (1), that is an increase in p_i of

$$\left((u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \right) y_k \quad (4)$$

And an increment in the denominator of equation (1), that is, an increase in q_i of

$$(u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \quad (5)$$

These increments are accumulated into P and q respectively as the membership matrix is updated. At the beginning of the next iteration the new cluster centers are again computed by equation (3).

The Modified FCM algorithm

1. set values for c , A , m , ϵ , and loop counter $t=1$
2. create a random $N \times c$ membership matrix $U^{(t)}$
3. initialize data structures P and q using equation (1)
4. compute c cluster centers using equation (3)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. update the membership matrix using equation (2) as each membership is computed, increment the corresponding element of P and q using equation (4) and equation (5)
6. if $\max \{u_{ki}^{(t)} - u_{ki}^{(t-1)}\} > \varepsilon$ increment t and go to step 4.

4. FCM-BASED VECTOR QUANTIZATION

In this paper, FCM process is used to calculate the reference vector or average vector from each cluster. The output of this process is a membership matrix and a set of reference vectors that used to design the vector quantization codebook. The training vector will be encoded with this codebook. The algorithm of this compression method is shown in Figure 1.

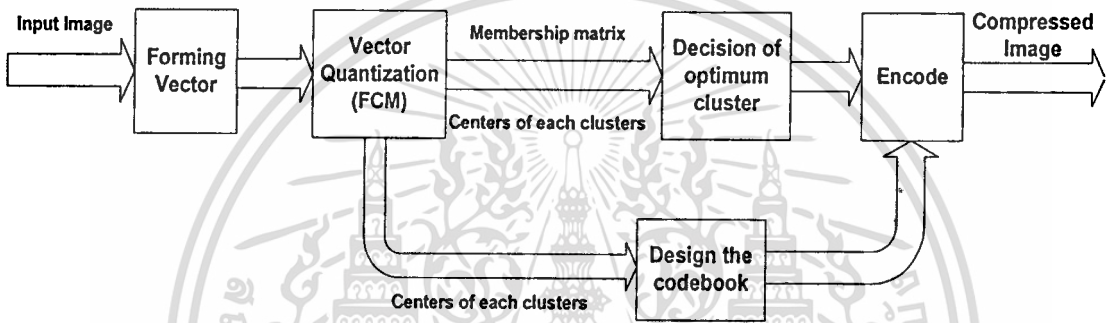


Figure 1: Block diagram of the proposed method.

The evaluation was performed by using the following measures:

$$bpp(\text{number of bits per pixel}) = \frac{\text{Encoded number of bits}}{\text{Number of pixels}} \quad (6)$$

$$PSNR(\text{Peak Signal to noise ratio}) = 10 \log_{10} \left[\frac{255^2}{MSE} \right] [dB] \quad (7)$$

$$MSE(\text{Mean square error}) = \frac{1}{3M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=1}^3 \{x_k(j,i) - \hat{x}_k(j,i)\}^2 \quad (8)$$

5. EXPERIMENTAL RESULTS

The proposed image compression algorithm was implemented using Matlab. Several multispectral images were used in experimental. Table 1 shows the experimental results when applied to a three-band (24 bits) image, size of 256x256 pixels. The input image size is 196 Kbyte. The experiments were carried out with the number of clusters of 4, 16 and 64.

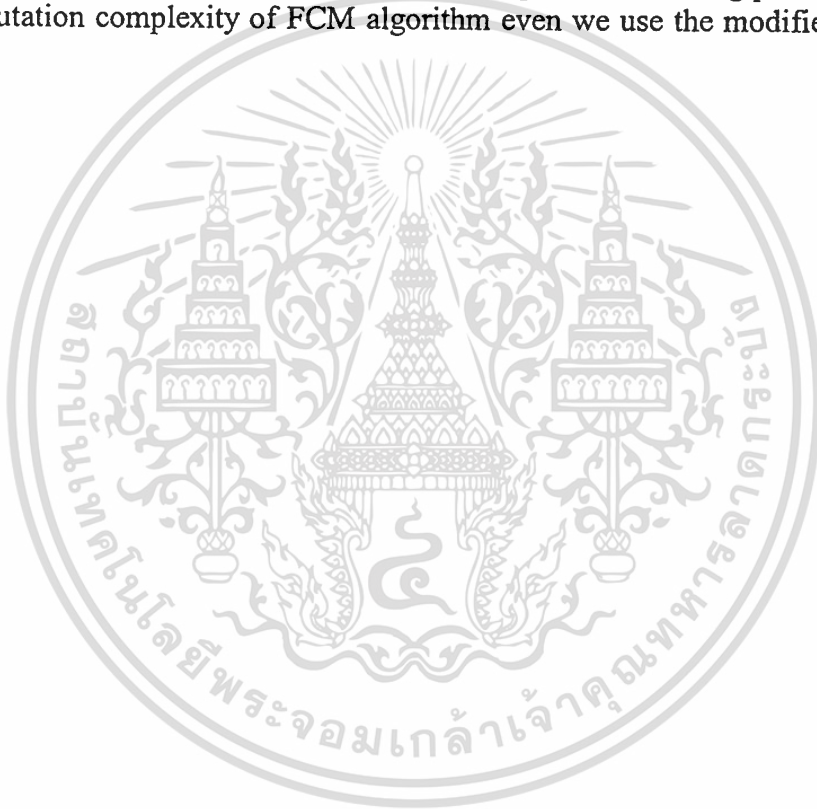
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Table 1: Experimental results.

Number of clusters	bpp	PSNR (dB)	MSE
4	2.0016	22.0651	404.1790
16	4.068	27.4127	117.9196
64	6.0293	31.2498	48.7638

6. CONCLUSION

The experimental result shown that the proposed method can compress the multispectral image data. The compressibility of this method and the image distortion from quantizing process depends on the number of clusters used in FCM process. The major problem of this method is the execution time spent in clustering process caused by the computation complexity of FCM algorithm even we use the modified version of FCM.



Intra/Inter-Band Coding in FCM-VQ of Multispectral Images

Yuttapong Rangsanseri and Uthai Sangthongpraow

Department of Telecommunications Engineering, Faculty of Engineering
King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
Email: kryuttha@kmitl.ac.th and s.uthai@ntt.com

ABSTRACT

Image compression is the process of reducing the number of bits required to represent an image. Vector Quantization method (VQ) is a method to deal with this operation. With this method a set of vector from the input image is encoded by a reduced set of reference vectors (the codebook). VQ is useful in compressing data that arises in wide range applications and it can achieve better compression performance than any conventional coding techniques, which based on the encoding of scalar quantities. Forming vector is the first step of VQ image compression. There are 2 ways to form the vector set from the input image. The first one is "intra-band vector forming". In this method, each vector was formed by dividing the input image into blocks. Each block will be formed as a training vector. The second method in forming vector is "inter-band vector forming". In this method, each pixel from the input image in any space will be formed as a training vector. This paper presents a multispectral image compression method using Vector Quantization technique based on Fuzzy c-Means (FCM). FCM algorithm was used to generate a good initial codebook for image compression. This codebook will be used to encode the input image. A modified version of FCM was used to reduce the computational time. The purpose of the modified FCM method is not reduction of iteration count, but it can reduce the time used in each iteration of FCM process. This image compression method was applied on several multispectral images. "Intra-band" and "Inter-band" vector forming was used in the vector forming process to evaluate the performance of this image compression method. For "intra-band" vector forming, various number of vector elements were used to make a comparison of experimental result. In modified FCM section, The modified version of FCM was applied to multispectral image using various number of clusters. The experimental result shows the average time saving in an iteration count of FCM process.

Keywords : Image Compression, Vector Quantization and FCM

1. INTRODUCTION

Image Compression is a process of reducing the number of bits required to represent an image. There are several methods to compress an image data. One popular method is Vector Quantization (VQ). VQ was used for compressing data in wide range applications, including image processing speech processing, facsimile transmission and weather satellites. In this method the training vectors in the image are encoded by a reduced set of reference vectors (the codebook). From rate distortion theory, it can be shown that VQ can achieve better compression performance than any conventional

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

coding technique which based on the encoding of scalar quantities. The objective of VQ is to design the codebook. An optimum VQ system is that one uses a codebook that yields the least average distortion of the reconstructed image. Other methods for designing codebooks have been proposed in. The most popular one is the LBG algorithm. In this paper we use FCM to design the codebook in Vector Quantization process. The major problem of this method is the huge execution time in the FCM process. So, this paper will also present the modification on the FCM algorithm that can reduce the execution time in each iteration.

2. FUZZY C-MEANS CLUSTERING TECHNIQUE

The fuzzy c-means (FCM) algorithm is an iterative clustering method that used to partition a data set. The objective of FCM segmentation is to compute the cluster centers and generate the class membership matrix. This class membership matrix is a cxN matrix, where c is the number of groups and N is the number of samples of image in n -space. Each column of the class membership matrix is the distribution of the class attribute of its corresponding sample. Each row of the class membership matrix is the membership value of each sample to be a member of that particular cluster.

An optimal fuzzy c-partition is one that minimizes the generalized least-squared errors function. We can explain the fuzzy clustering by the following equation.

$$\text{Minimize: } J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ki})^m \|y_k - v_i\|^2 A$$

Where : $Y = \{y_1, y_2, y_3, \dots, y_N\} \subset \mathbb{R}^n$ is the data set,
 c is the number of clusters in Y : $2 \leq c < n$,
 m is a weighting exponent: $1 \leq m < \infty$,
 $U = \{u_{ki}\}$ is the fuzzy c-partition of Y ,
 $\|y_k - v_i\|_A$ is an induced a -norm on \mathbb{R}^n , and,
 A is a positive-definite $(n \times n)$ weight matrix.

The weighting exponent m has the effect of reducing the squared distance error by an amount that depends on the observation's membership in the cluster. As $m \rightarrow 1$, the partitions that minimize J_m become increasingly hard. Conversely, higher values of m tend to soften a samples cluster membership, and the partition becomes increasingly blurred. Generally m must be selected by experimental means.

The first step of FCM algorithm is generate an initial random membership matrix and use this random membership matrix as weight of each samples to belong to each cluster. Then computes the centroid of each cluster. The new cluster centers are used to update the membership matrix. The updated membership matrix is compared with the previous ones. If the difference is greater than some threshold, then another iteration is computed, otherwise the algorithm is stopped. The algorithm is shown below. The notation $x^{(t)}$ signifies the value of variable x at iteration t .

The FCM Algorithm

2. set value for c, A, m, ϵ , and the loop counter $t = 1$
4. create a random $N \times c$ membership matrix U
5. compute each cluster centroid by equation

$$v_i^{(t)} = \sum_{k=1}^N (u_{ki}^{(t)})^m y_k / \sum_{k=1}^N (u_{ki}^{(t)})^m \quad (1)$$

6. update the membership matrix by equation

$$u_{ki}^{(t+1)} = \left(\sum_{j=1}^c \left(\frac{d_{ki}}{d_{kj}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (2)$$

where $d_{ki} = \|y_k - v_i^{(t)}\| A$

7. If $\max \{u_{ki}^{(t)} - u_{ki}^{(t-1)}\} > \epsilon$ increment t and go to step 3.

3. THE MODIFIED FUZZY C-MEANS ALGORITHM

From equation (1) we can find that each cluster center v_i is computed as the weighted average of samples in the data set Y . The weight used for each sample is that sample's membership in the i^{th} cluster. In the original algorithm this is computed by performing a pass over the entire data set and membership matrix.

The following describes an improvement to the algorithm whereby the computation of cluster centers is performed in sequence with the membership matrix updating. This effectively eliminates an entire pass of the data set. The outcome is not a decrease in the number of iterations required for convergence, but decrease in the time of each iteration.

We maintain two extra data structures, a $c \times n$ matrix, $P = \{p_i\}$, and a vector q of length c . We can obtain these two matrix from the first iteration when the random membership matrix is created. If we represent the numerator and denominator of equation (1) by P and q , respectively, then keep them as initial values. So now the equation (1) is replaced by equation (3)

$$v_i = P_i / q_i \quad (3)$$

where p_i is a vector of length n , and q_i is a scalar. The dot- i subscript is used to avoid confusion between vectors and scalars.

Each time an element u_{ki} of the membership matrix is computed there is an increment in the numerator of equation (1), that is an increase in p_i of

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$\left((u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \right) y_k \quad (4)$$

And an increment in the denominator of equation (1), that is, an increase in q_i of

$$(u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \quad (5)$$

These increments are accumulated into P and q respectively as the membership matrix is updated. At the beginning of the next iteration the new cluster centers are again computed by equation (3).

The Modified FCM algorithm

7. set values for c, A, m, ε , and loop counter $t=1$
8. create a random $N \times c$ membership matrix $U^{(t)}$
9. initialize data structures P and q using equation (1)
10. compute c cluster centers using equation (3)
11. update the membership matrix using equation (2) as each membership is computed, increment the corresponding element of P and q using equation (4) and equation (5)
12. if $\max \{ u_{ki}^{(t)} - u_{ki}^{(t-1)} \} > \varepsilon$ increment t and go to step 4.

4. FCM-BASED VECTOR QUANTIZATION

In this paper, FCM process is used to calculate the reference vector or average vector from each cluster. The output of this process is a membership matrix and a set of reference vectors that used to design the vector quantization codebook. The training vector will be encoded with this codebook. The algorithm of this compression method is shown in Figure 1.

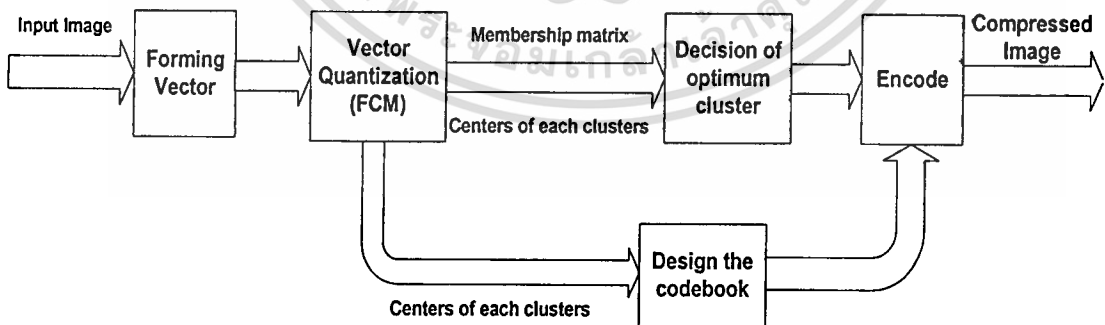


Figure 1: Block diagram of the proposed method.

The evaluation was performed by using the following measures:

$$bpp(\text{number of bits per pixel}) = \frac{\text{Encoded number of bits}}{\text{Number of pixels}} \quad (6)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$PSNR(\text{Peak Signal to noise ratio}) = 10 \log_{10} \left[\frac{255^2}{MSE} \right] [dB] \quad (7)$$

$$MSE(\text{Mean square error}) = \frac{1}{3M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \sum_{k=1}^3 \{x_k(j,i) - \hat{x}_k(j,i)\}^2 \quad (8)$$

5. EXPERIMENTAL RESULTS

The proposed image compression algorithm was implemented using Matlab. Several multispectral images were used in experimental. Table 1 shows the experimental results when applied to a three-band (24 bits) image, size of 256x256 pixels. The input image size is 196 Kbyte. The experiments were carried out with the number of clusters of 4, 16 and 64.

Table 1: Experimental results.

Number of clusters	bpp	PSNR (dB)	MSE
4	2.0016	22.0651	404.1790
16	4.068	27.4127	117.9196
64	6.0293	31.2498	48.7638

6. CONCLUSION

The experimental result shown that the proposed method can compress the multispectral image data. The compressibility of this method and the image distortion from quantizing process depends on the number of clusters used in FCM process. The major problem of this method is the execution time spent in clustering process caused by the computation complexity of FCM algorithm even we use the modified version of FCM.

Modified Fuzzy C-Mean for Clustering of Multi-spectral Image

Dr. Yuttapong Rangsanseri and Mr. Uthai Sangthongpraow
 Department of Telecommunications Engineering, Faculty of Engineering
 King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand
 Tel: (66-2) 326-9967, Fax: (66-2) 326-9086, E-mail: kryuttha@kmitl.ac.th

ABSTRACT

The purpose of cluster analysis is to partition a data set into a number of disjoint groups. The members within a group are more similar to each other than members from different groups. In multi-spectral image or remote sensing image cluster analysis can be used to analyze the natural spectral classes in a data set in order to select representative samples for classification. The Fuzzy C-Means (FCM) clustering is an iterative partitioning method that produces optimal c-partitions. The objective of FCM segmentation is to convert image feature values into class membership values.

Since the standard fuzzy c-means algorithm take a long time to partition a large data set. Because FCM program must read the entire data set into a memory to processing. This paper will propose a new algorithm to reduce the execution time of standard fuzzy clustering by decrement the number of passes per iteration data set and the membership matrix. The result is also shown in this paper.

INTRODUCTION

The fuzzy c-means (FCM) algorithm is an iterative clustering method that used to partition a data set. The objective of FCM segmentation is to compute the cluster centers and generate the class membership matrix. This class membership matrix is a $c \times N$ matrix, where c is the number of groups and N is the number of samples of image in n -space. Each column of the class membership matrix is the distribution of the class attribute of its corresponding sample. Each row of the class membership matrix is the membership value of each sample to be a member of that particular cluster. A formal definition of fuzzy c-partitions is in.

An optimal fuzzy c-partition is one that minimizes the generalized least-squared errors function. We can explain the fuzzy clustering by the following equation.

$$\text{Minimize: } J_m(U, v) = \sum_{k=1}^N \sum_{i=1}^c (u_{ki})^m \|y_k - v_i\|^2 A$$

Where : $Y = \{y_1, y_2, y_3, \dots, y_N\} \subset \mathbb{R}^n$ is the data set,
 c is the number of clusters in Y : $2 \leq c < n$,
 m is a weighting exponent: $1 \leq m < \infty$,
 $U = \{u_{ki}\}$ is the fuzzy c-partition of Y ,
 $\|y_k - v_i\|_A$ is an induced a -norm on \mathbb{R}^n , and,
 A is a positive-definite $(n \times n)$ weight matrix.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The weighting exponent m has the effect of reducing the squared distance error by an amount that depends on the observation's membership in the cluster. As $m \rightarrow 1$, the partitions that minimize J_m become increasingly hard. Conversely, higher values of m tend to soften a samples cluster membership, and the partition becomes increasingly blurred. Generally m must be selected by experimental means.

The first step of FCM algorithm is generate an initial random membership matrix and use this random membership matrix as weight of each samples to belong to each cluster. Then computes the centroid of each cluster. The new cluster centers are used to update the membership matrix. The updated membership matrix is compared with the previous ones. If the difference is greater than some threshold, then another iteration is computed, otherwise the algorithm is stopped.

The algorithm is shown below. The notation $x^{(t)}$ signifies the value of variable x at iteration t .

The FCM Algorithm

3. set value for c, A, m, ϵ , and the loop counter $t = 1$
6. create a random $N \times c$ membership matrix U
7. compute each cluster centroid by equation

$$v_i^{(t)} = \frac{\sum_{k=1}^N (u_{ki}^{(t)})^m y_k}{\sum_{k=1}^N (u_{ki}^{(t)})^m} \quad (1)$$

8. update the membership matrix by equation

$$u_{ki}^{(t+1)} = \left(\sum_{j=1}^c \left(\frac{d_{ki}}{d_{kj}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (2)$$

where $d_{ki} = \|y_k - v_i^{(t)}\| A$

9. If $\max \{u_{ki}^{(t)} - u_{ki}^{(t-1)}\} > \epsilon$ increment t and go to step 3.

THE MODIFIED FUZZY C-MEANS ALGORITHM

Commonly, standard FCM clustering programs read the entire data set into a memory for processing. This is not generally feasible when the input is a large data set such as a multi-spectral image. The PC operating system doesn't have enough virtual memory for portability. There are various approaches to solve this problem such as clustering only small image, clustering image subsets, or using parallel computers. In this paper we will propose an alternative algorithm to solve this problem by memorize only portion of the image at any instant time. This method can use with any images which is stored in a database that allow access only in line by line mode from remote host.

There has been other algorithm on improving the FCM. The objective is to speed up the FCM processing or to decrease the number of iterations require for convergence. The AFCM (approximate FCM) algorithm is an algorithm to speed up FCM by replacing the real valued data structures with integer values and used look up techniques where possible to reduce the execution time of each iteration. In the first algorithm the cluster centers are updated as the new membership grade for each sample are computed to produce more rapid convergence, although each iteration requires more computation. In the second algorithm the membership matrix is updated as each cluster centers is recomputed. More recently, a multi-stage random sampling

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

technique has been applied to the FCM algorithm producing more rapid convergence in the final stage.

Determine the equation (1) we can found that each cluster center v_i is computed as the weighted average of samples in the data set Y . The weight used for each sample is that sample's membership in the i th cluster. In the original algorithm this is computed by performing a pass over the entire data set and membership matrix.

The following describes an improvement to the algorithm whereby the computation of cluster centers is performed in sequence with the membership matrix updating. This effectively eliminates an entire pass of the data set. The outcome is not a decrease in the number of iterations required for convergence, but decrease in the time of each iteration.

Method :

We maintain two extra data structures, a $c \times n$ matrix, $P = \{p_i\}$, and a vector q of length c . We can obtain these two matrix from the first iteration when the random membership matrix is created. If we represent the numerator and denominator of equation (1) by P and q , respectively, then keep them as initial values. So now the equation (1) is replaced by equation (3)

$$v_i = P_i / q_i \quad (3)$$

Where p_i is a vector of length n , and q_i is a scalar. The dot- i subscript is used to avoid confusion between vectors and scalars.

Each time an element u_{ki} of the membership matrix is computed there is an increment in the numerator of equation (1), that is an increase in p_i of

$$\left((u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \right) y_k \quad (4)$$

And an increment in the denominator of equation (1), that is, an increase in q_i of

$$(u_{ki}^{(t+1)})^m - (u_{ki}^{(t)})^m \quad (5)$$

These increments are accumulated into P and q respectively as the membership matrix is updated. At the beginning of the next iteration the new cluster centers are again computed by equation (3).

The Modified FCM algorithm

13. set values for c , A , m , ϵ , and loop counter $t=1$
14. create a random $N \times c$ membership matrix $U^{(t)}$
15. initialize data structures P and q using equation (1)
16. compute c cluster centers using equation (3)
17. update the membership matrix using equation (2) as each membership is computed, increment the corresponding element of P and q using equation (4) and equation (5)
18. if $\max \{ u_{ki}^{(t)} - u_{ki}^{(t-1)} \} > \epsilon$ increment t and go to step 4.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

EXPERIMENTAL RESULT

In this paper we use Matlab to segment various multi-spectral images with the original FCM algorithm and the new modified FCM algorithm. Many multi-spectral images were used to evaluate the improvement. Table 1 show the average time used in each iteration of the original FCM and the modified FCM when apply to a 256x256x3 image into 3, 4 and 5 clusters on an intel celeron 400 MHz with 64 Mbyte of Ram.

Average time used per iteration (min)					
3 cluster		4 cluster		5 cluster	
Original FCM	Modified FCM	Original FCM	Modified FCM	Original FCM	Modified FCM
31.84	17.73	37.96	22.47	47.825	31.45

Table 1 : Comparison of FCM program

From the table above we can see that the new modified FCM can decrease the execution time of each per iteration. The input parameters in each case where: $m=2$, $A=I$, $c=3$, and $\epsilon=0.01$. For a 512x512x3 image when partition into 3 cluster, the original FCM must take 495.96 minutes in each iteration ,but when use new modified FCM algorithm its take 282.4 minutes in each iteration. We found that in this case the modified algorithm can decrease the execution time in each iteration about 43 percent.

CONCLUSION

The advantage of this modified FCM algorithm is its can reduce execution time in each iteration. Especially for the large data set this method can save much time for clustering. Another advantage of this method is we can cluster an entire image on standard computer even a large data set.

การหาขอบเขตของวัตถุในภาพสีโดยอาศัยการจำแนกกลุ่มข้อมูล
COLOR IMAGE EDGE DETECTION USING CLUSTER ANALYSIS
 อุทัย แสงทองพราว, ยุทธพงษ์ รังสรรค์เสรี และ ปัญญา จิตติมัชชีมา

บทคัดย่อ

การหาขอบเขตของวัตถุในภาพสีนั้นเป็นกระบวนการพื้นฐานในการวิเคราะห์ข้อมูลภาพ บทวิจัยนี้จะขอนำเสนอวิธีการหาขอบเขตของวัตถุในภาพสีโดยอาศัยข้อมูลจากการจำแนกกลุ่มข้อมูลมาช่วยในการเพิ่มประสิทธิภาพการทำงาน โดยมีแนวคิดที่จะนำข้อมูลสีของภาพในแต่ละแบนด์มาช่วยในการคำนวณค่ากราเดียนท์ที่ดีที่สุด โดยขั้นตอนแรกจะทำการวิเคราะห์ค่าคลัสเตอร์ภาพสีต้นแบบและเก็บค่าสีหลักๆหรือค่าสีเฉลี่ยของแต่ละคลัสเตอร์เอาไว้เพื่อนำไปใช้ในการคำนวณหาโปรเจกชันเวกเตอร์ที่เหมาะสมที่จะนำไปใช้ในการหาค่ากราเดียนท์สูงสุด ทุกๆพิกเซลในภาพจะถูกจัดให้อยู่คลัสเตอร์ใดคลัสเตอร์หนึ่งที่เหมาะสม และจากวิธีดังกล่าวเส้นขอบเขตวัตถุจะเกิดขึ้นบนบริเวณพิกเซลที่มีการเปลี่ยนแปลงจากคลัสเตอร์หนึ่งไปยังอีกคลัสเตอร์หนึ่ง พร้อมกันนั้นจะทำการหาค่ากราเดียนท์เวกเตอร์ของภาพในแต่ละแบนด์และนำค่ากราเดียนท์ของแต่ละพิกเซลมาฉายไปยังโปรเจกชันเวกเตอร์เหมาะสมจะทำให้เราสามารถคำนวณหาค่ากราเดียนท์ที่ดีที่สุดของภาพได้ วิธีดังกล่าวได้นำมาทดสอบกับภาพสีและผลการทดลองได้แสดงให้เห็นว่าข้อมูลคลัสเตอร์มีประโยชน์สำหรับการหาขอบเขตของวัตถุในภาพสีและภาพหลายแบนด์

บทนำ

การหาขอบเขตของวัตถุเป็นเรื่องที่มีบทบาทอย่างมากในระบบการแปลความหมายข้อมูลในภาพ ด้วยเหตุผลนี้จึงได้มีการนำเสนอกระบวนการในการหาขอบเขตของวัตถุภายในภาพด้วยวิธีต่างๆขึ้นมามากมาย [1] [2] โดยวิธีการหาขอบเขตของวัตถุในภาพสี [3], [5] ที่นิยมใช้กันอยู่ในปัจจุบันนั้นมักจะทำการแปลงภาพสีเป็นภาพขาวดำก่อนแล้วจึงนำภาพขาวดำไปหาค่ากราเดียนท์ซึ่งการหาด้วยวิธีดังกล่าวนี้ยังไม่สามารถนำข้อมูลสีในแต่ละแบนด์มาใช้ได้อย่างมีประสิทธิภาพสูงสุด ในบทความนี้จะกล่าวถึงวิธีการหาขอบเขตของวัตถุภายในภาพสีโดยจะนำวิธีการจำแนกข้อมูลภาพออกเป็นกลุ่มมาช่วยในการเพิ่มประสิทธิภาพการหาขอบเขตวัตถุในภาพสีและเป็นการนำข้อมูลสีในแต่ละแบนด์มาใช้ได้อย่างมีประสิทธิภาพสูงสุดด้วยดังจะกล่าวในหัวข้อต่อไป

การหาขอบเขตของวัตถุภายในภาพสีด้วยวิธีที่ใช้กันอยู่ทั่วไป

ปัจจุบันวิธีที่นิยมใช้ในการหาขอบเขตของวัตถุในภาพสีที่นิยมใช้กันนั้นมีอยู่สองวิธี วิธีหนึ่งจะใช้วิธีแปลงภาพสีเป็นภาพขาวดำก่อนโดยการตัดองค์ประกอบสีของทุกพิกเซลในภาพออกโดยการคำนวณค่าเฉลี่ยของแต่ละสี แล้วนำค่าเฉลี่ยนั้นไปใช้หาค่ากราเดียนท์ของภาพขาวดำ ซึ่งวิธีนี้ไม่ใช่วิธีที่ดีนัก เพราะว่ากรณีใดๆทั้งสิ้น อีกทั้งยังมีให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งมักใช้วิธีการรวมองค์ประกอบของสีในแบบค้ต่างๆ (R,G,B) เข้าด้วยกัน แล้วจึงนำไปหาขอบเขตของวัตถุจากการหากราเดียนท์ของภาพขาวดำนี้ หรืออีกวิธีหนึ่งคือการหากราเดียนท์ของภาพในแต่ละแบบค้ (R,G,B) มาก่อนแล้วจึงนำมารวมกันในแต่ละพิกเซล โดยวิธีที่กล่าวมาทั้งสองวิธีสามารถอธิบายได้ดังสมการ

$$I' = 0.3I_r' + 0.59I_g' + 0.11I_b' \quad (1)$$

$$= (0.3I_r + 0.59I_g + 0.11I_b)' \quad (2)$$

เมื่อ I_r' , I_g' , I_b' คือกราเดียนท์ของแต่ละองค์ประกอบสี และ I_r , I_g , I_b คือความเข้มของแสงในแต่ละองค์ประกอบสีภายในภาพ จากสมการจะเห็นว่าวิธีนี้เปรียบเสมือนการฉายค่ากราเดียนท์เวกเตอร์ไปยังโปรเจกชันเวกเตอร์ $[0.3, 0.59, 0.11]$ เพื่อให้ได้ค่าสเกลลาร์ ค่าที่ได้จากการฉายกราเดียนท์เวกเตอร์นี้บางค่าจะมีค่าน้อยถ้าทิศทางของการฉายของกราเดียนท์เวกเตอร์ไม่ได้ฉายไปในทิศทางเดียวกันกับโปรเจกชันเวกเตอร์ เช่นค่าที่ได้จากการฉายกราเดียนท์เวกเตอร์ระหว่างสีส้ม $((r, g, b) = (1.0, 0.4, 0.0))$ กับสีฟ้า $((r, g, b) = (0.0, 0.72, 1.0))$ จะมีค่าเป็นศูนย์ ถ้าใช้วิธีการตามสมการที่ (1) ดังนั้นถ้าเราสามารถหาโปรเจกชันเวกเตอร์ที่เหมาะสมกับการฉายค่ากราเดียนท์ของแต่ละพิกเซลได้จะทำให้เราสามารถหาขอบเขตของวัตถุในภาพได้อย่างมีประสิทธิภาพสูงสุด

การหาขอบเขตของวัตถุในภาพสีโดยอาศัยข้อมูลจากการวิเคราะห์คลัสเตอร์

การจำแนกข้อมูลออกเป็นกลุ่มนั้นถูกนำมาประยุกต์ใช้ในบทวิจัยนี้เพื่อใช้ในการหาโปรเจกชันเวกเตอร์ที่เหมาะสมที่จะนำไปฉายค่ากราเดียนท์ของภาพเพื่อให้การหาขอบเขตของวัตถุเป็นไปอย่างมีประสิทธิภาพสูงสุด โดยมีขั้นตอนการทำงานดังบล็อกไดอะแกรมในรูปที่ 1. การหากราเดียนท์ในแต่ละแบบค้ด้วยวิธีของโซเบล (Sobel Operator)

ในบทความนี้ได้เลือกใช้การคำนวณค่ากราเดียนท์ของภาพในแต่ละแบบค้ด้วยวิธีของโซเบล [2] โดยค่ากราเดียนท์ในแต่ละแบบค้จะคำนวณได้จากการคอนโวลูชันภาพอินพุตด้วยหน้าตาขนาด 3×3 ในที่นี้จะแทนด้วย S_x และ S_y ซึ่งจะใช้ในการคำนวณค่ากราเดียนท์ในแนวแกน x และแนวแกน y ตามลำดับ โดยหน้าตาต่าง S_x และ S_y นั้นแสดงไว้ดังรูปที่ 2

ขนาดของกราเดียนท์ (M) และเฟส (θ) สามารถคำนวณได้จากสมการที่ (3) และสมการที่

$$M = \sqrt{S_x^2 + S_y^2} \quad (3)$$

$$\theta = \tan^{-1}(G_y / G_x) \quad (4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

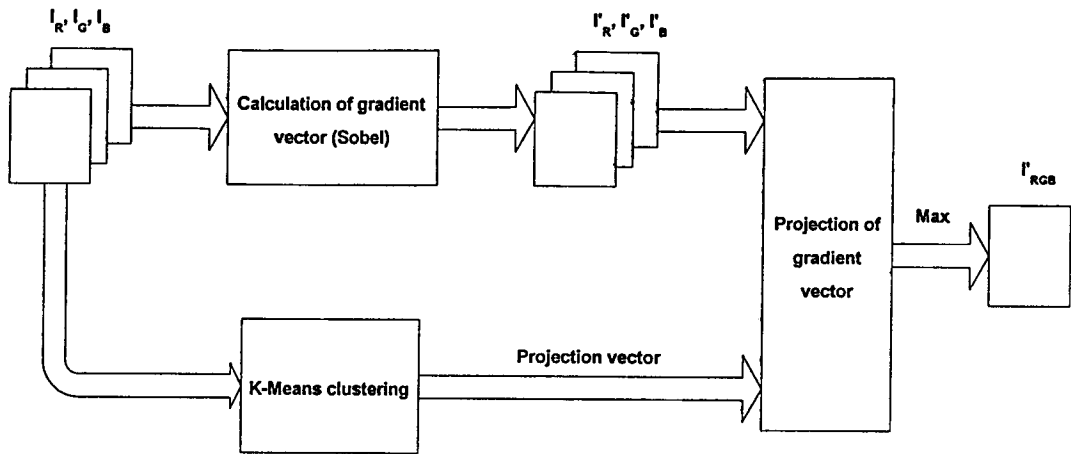


Figure 1: Block Diagram

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad S_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Figure 2: Sobel mask

ค่าการเคลื่อนที่ในแต่ละแบนด์ที่ได้จากวิธีการของโซเบลนี้จะถูกนำไปใช้คำนวณร่วมกับค่าโปรเจกชันเวกเตอร์ที่เหมาะสมซึ่งเป็นข้อมูลที่ได้จากการวิเคราะห์คลัสเตอร์ (Cluster Analysis) ที่จะกล่าวถึงในหัวข้อถัดไปเพื่อหาค่าการเคลื่อนที่สูงสุดของภาพสีต้นแบบ

การจำแนกกลุ่มข้อมูล (Cluster Analysis)

ในขั้นตอนนี้เราจะใช้การวิเคราะห์คลัสเตอร์มาช่วยในการจำแนกกลุ่มข้อมูลออกเป็นกลุ่มย่อยๆ สำหรับภาพสีนั้นแต่ละพิกเซลภายในภาพเปรียบเสมือนเวกเตอร์บนแกน RGB ดังนั้นกลุ่มข้อมูลแต่ละกลุ่มที่ถูกแบ่งนั้นจะประกอบด้วยบริเวณที่มีสีเหมือนกันหรือคล้ายคลึงกัน วิธีการจำแนกกลุ่มข้อมูลนั้นมีหลายวิธี สำหรับบทความนี้ได้เลือกใช้วิธีการจำแนกกลุ่มโดยวิธี K-mean [4] ซึ่งจะทำให้การแบ่งกลุ่มข้อมูลโดยอัตโนมัติและหาค่าเฉลี่ยของเวกเตอร์ภายในกลุ่มแต่ละกลุ่ม โดยมีขั้นตอนการทำงานดังนี้

1. กำหนดเวกเตอร์เฉลี่ยที่จะใช้เป็นค่าเริ่มต้นในการแบ่งกลุ่ม โดยเป็นเวกเตอร์ใดก็ได้ที่มีค่าไม่ซ้ำกันจำนวนเท่ากับจำนวนคลัสเตอร์ที่เราต้องการแบ่งและกำหนดให้เวกเตอร์ของทุกพิกเซลในภาพเป็นสมาชิกอยู่ในคลัสเตอร์ที่มีเวกเตอร์เฉลี่ยใกล้เคียงกับพิกเซลนั้นมากที่สุด
2. คำนวณหาค่าเฉลี่ยใหม่ภายในกลุ่มทุกๆกลุ่ม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ถ้าค่าเฉลี่ยใหม่ที่คำนวณได้มีค่าเท่ากับค่าเฉลี่ยเดิมก็แสดงว่าการจำแนกข้อมูลเสร็จสิ้น แต่ถ้าค่าเฉลี่ยที่คำนวณได้ใหม่ยังไม่เท่ากับค่าเฉลี่ยเดิมก็ให้กลับไปทำข้อ 2 ใหม่

การหาโปรเจกชันเวกเตอร์ที่เหมาะสม

กำหนดให้บริเวณ R_1, R_2 เป็นบริเวณภายในคลัสเตอร์ C_1, C_2 จุดศูนย์กลางของทั้งสองคลัสเตอร์แทนด้วยจุด V_1, V_2 โปรเจกชันเวกเตอร์ที่จะนำมาใช้ในการหาค่ากราเดียนต์เวกเตอร์ของฟังก์ชันระหว่างบริเวณ R_1 และ R_2 ก็ควรจะมีทิศทางเดียวกันคือ $V_1 - V_2$ ฉะนั้น $V_1 - V_2$ จึงเป็นโปรเจกชันเวกเตอร์หนึ่งที่เหมาะสมในการนำไปคำนวณหาค่ากราเดียนต์สูงสุดของฟังก์ชันที่อยู่ระหว่างคลัสเตอร์ R_1 และ R_2 ในภาพสีนั้นถ้าเราแบ่งข้อมูลออกเป็น k กลุ่ม จะมี $k(k-1)/2$ โปรเจกชันเวกเตอร์ที่สามารถหาได้ สำหรับทุกๆ ฟังก์ชันในภาพสีนั้นกราเดียนต์เวกเตอร์ทุกเวกเตอร์จะถูกฉายไปยังโปรเจกชันเวกเตอร์ทั้งหมด $k(k-1)/2$ เวกเตอร์ที่หามาได้และเลือกเอาค่าที่สูงที่สุดมาใช้ในการคำนวณหาขอบเขตของวัตถุต่อไป จะเห็นว่าวิธีดังกล่าวนี้จะสามารถใช้ข้อมูลสีของภาพสีมาใช้ได้อย่างมีประสิทธิภาพ

จะเห็นว่าถ้าในภาพสีถูกแบ่งกลุ่มเป็นจำนวนหลายกลุ่ม จำนวนของโปรเจกชันเวกเตอร์เพิ่มขึ้นมาก วิธีแก้ไขปัญหานี้ก็คือการจำกัดจำนวนคลัสเตอร์ไม่ให้มากเกินไป

ผลการทดลอง

ในบทความนี้ได้ทำการทดลองหาขอบเขตของวัตถุจากภาพสีตามวิธีที่ได้กล่าวมาข้างต้น ข้อมูลภาพสีต้นแบบจะถูกนำมาหากราเดียนต์ในทุกๆ องค์ประกอบสี (R,G,B) โดยใช้วิธีของ Sobel และจำกัดจำนวนคลัสเตอร์ไว้ 4 คลัสเตอร์ ทุก โปรเจกชันเวกเตอร์ถูกนำมา normalize ให้มีขนาดเท่ากับ 0.67 ซึ่งมีขนาดเท่ากับขนาดของโปรเจกชันเวกเตอร์มาตรฐาน [0.3, 0.59, 0.11] และโปรเจกชันเวกเตอร์มาตรฐานนี้ก็ถูกนำมารวมเป็นโปรเจกชันเวกเตอร์หนึ่งในการหากราเดียนต์สูงสุดด้วย ดังนั้นจำนวนโปรเจกชันเวกเตอร์ทั้งหมดที่นำมาใช้ในการคำนวณ จึงมีจำนวนเท่ากับ $(k(k-1)/2)+1$ นอกจากนี้ในบทความนี้ได้ทำการประยุกต์นำการรวมคลัสเตอร์ที่มีค่าเฉลี่ยใกล้เคียงกันเข้าด้วยกันเพื่อลดจำนวนโปรเจกชันเวกเตอร์ โดยถ้าระยะห่างระหว่าง 2 คลัสเตอร์มีค่าน้อยกว่าค่าที่กำหนดจะทำการรวมคลัสเตอร์ทั้งสองเข้าด้วยกัน ค่าของจุดศูนย์กลางใหม่จะแทนด้วยค่าเฉลี่ยของทั้งสองคลัสเตอร์ กระบวนการนี้จะกระทำไปอย่างต่อเนื่องจนกระทั่งระยะห่างระหว่างคลัสเตอร์มีค่ามากกว่าค่าที่กำหนด

ภาพสีต้นแบบที่ใช้ในการทดลองแสดงไว้ดังรูปที่ 3(a) รูปที่ 3(b) แสดงภาพที่ได้จากการวิเคราะห์คลัสเตอร์ภาพสีต้นแบบออกเป็น 4 คลัสเตอร์ ภาพที่ 3(c) แสดงภาพที่ได้จากการหาขอบเขตของวัตถุด้วยวิธีที่นิยมใช้กันอยู่ทั่วและภาพที่ 4(d) แสดงภาพขอบเขตวัตถุที่ได้จากการหาขอบเขตของวัตถุโดยวิธีที่นำเสนอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

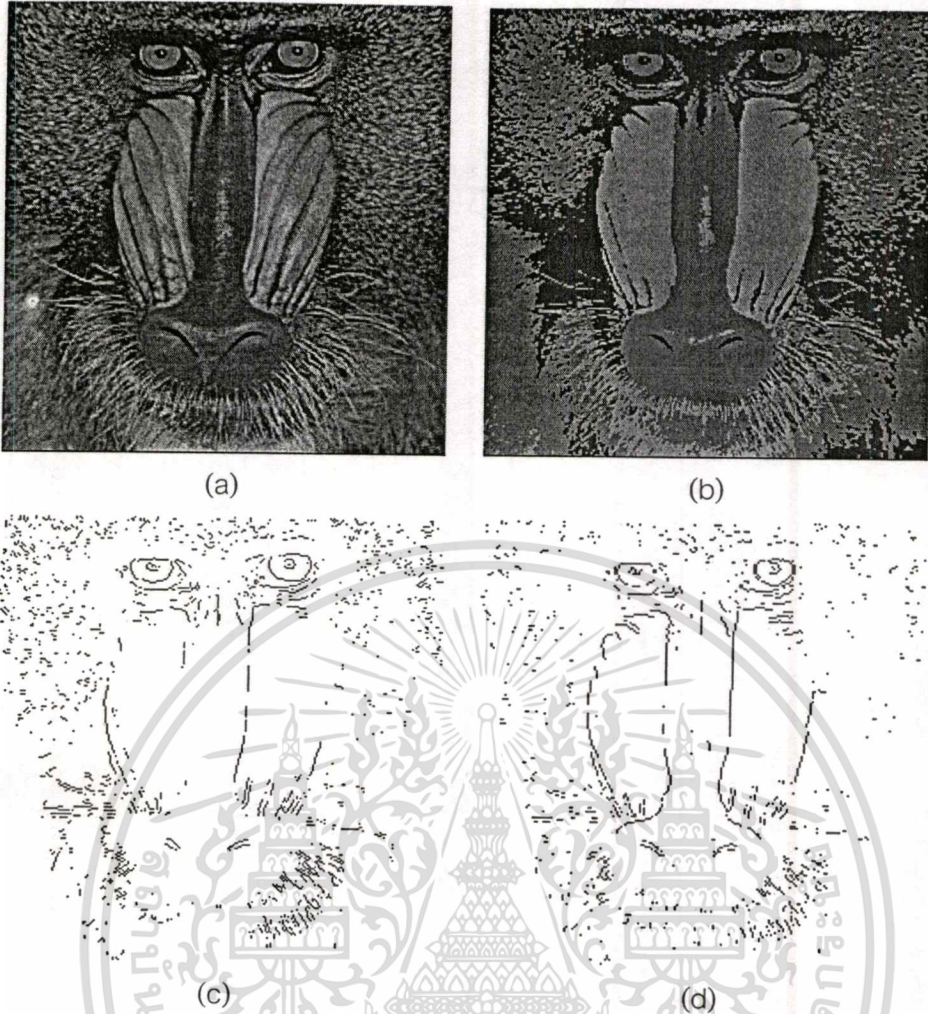


Figure 3 : Experimental Result 3(a) Original color image (b) 4 Clusters image
(c) Edge image from traditional edge detector (d) Edge image from propose method

จากการทดลองจะเห็นได้ชัดว่าการหาขอบเขตด้วยวิธีดังกล่าวจะมีประสิทธิภาพสูงกว่าวิธีเก่า นั่นเป็นเพราะว่ากลุ่มโปรเจกชันเวกเตอร์ที่เหมาะสมถูกเลือกนำมาใช้ในการรวมองค์ประกอบสี

สรุปผลการทดลอง

ประโยชน์ของการหาขอบเขตของวัตถุในภาพสีโดยการจำแนกกลุ่มข้อมูลคือสามารถนำข้อมูลของสีภายในภาพมาเป็นแนวทางในการหาโปรเจกชันเวกเตอร์ต่างๆที่เหมาะสม ทำให้การหาขอบเขตของวัตถุภายในภาพสีมีประสิทธิภาพสูงขึ้น แต่ปัญหาของการหาขอบเขตของวัตถุในภาพสีด้วยวิธีนี้คือความกระบวนกรที่ซับซ้อนในการจำแนกข้อมูลออกเป็นกลุ่ม เราสามารถแก้ไขปัญหานี้โดยการแบ่งภาพต้นแบบออกเป็นส่วนแล้วหาขอบเขตของวัตถุในแต่ละส่วนของภาพต้นแบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

นายอุทัย แสงทองพราว เกิดเมื่อวันที่ 5 มิถุนายน 2520 ที่จังหวัดกรุงเทพมหานคร สำเร็จการศึกษาวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมโทรคมนาคม จากสถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง ปีการศึกษา 2540

ปี 2542 เข้าทำงานในบริษัทซอฟต์แวร์ปาร์ค ในตำแหน่งโปรแกรมเมอร์ ต่อมาในปลายปี 2543 ได้ย้ายมาทำงานให้กับบริษัทเอ็นทีทีคอมมิวนิเคชั่น (ประเทศไทย) ในแผนกซิสเต็มเซอร์วิส จนกระทั่งปัจจุบัน

