การศึกษามาตรฐานการบีบอัดข้อมูลภาพนิ่ง JPEG

ชนิด SEQUENTIAL BASELINE SYSTEM

STUDY OF JPEG IMAGE COMPRESSION STANDARD

นางสาวพัฒชรินทร์ ฝุ่นทอง รหัส 44362705

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 25 พ.ค. 2553
เลขทะเบียน 150504
เลขเรียกหนังสือ ปร.
มหาวิทยาลัยนเรศวร
2547

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2547

ใบรับรองโครงงานวิศวกรรม

หัวข้อโครงงาน	การศึกษามาตรฐานการบีบอัดข้อมูลภาพนิ่ง JPEG ชนิด SEQUENTIAL BASELINE SYSTEM
ผู้ดำเนินโครงงาน	นางสาวพัฒชรินทร์ ฝุ่นทอง
อาจารย์ที่ปรึกษา	อาจารย์ศิริพร เดชะศิลารักษ์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2547

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้โครงงานฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะกรรมการสอบโครงงานวิศวกรรม

.....ประธานกรรมการ
(อ. ศิริพร เดชะศิลารักษ์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. สุชาติ แย้มเม่น)

.....กรรมการ
(ดร. พนมขวัญ ริยะมงคล)

หัวข้อโครงงาน	การศึกษามาตรฐานการบีบอัดข้อมูลภาพนิ่ง JPEG ชนิด SEQUENTIAL BASELINE SYSTEM
ผู้ดำเนินโครงงาน	นางสาวพัฒชรินทร์ ฝุ่นทอง
อาจารย์ที่ปรึกษา	อาจารย์ศิริพร เดชะศิลารักษ์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2547

บทคัดย่อ

โครงงานนี้เป็นการศึกษามาตรฐานการบีบอัดข้อมูลภาพนิ่ง JPEG (Joint Photographic Expert Group) ชนิด Sequential Baseline System ซึ่งรับข้อมูลภาพนิ่งในรูป Windows Bitmap File(.bmp) ชนิด 24 บิต/พิกเซล ขนาด 256×256 พิกเซล มาทำการแบ่ง MCU ให้มีขนาด 16×16 พิกเซล แล้วนำมาการแปลงระบบสีจาก RGB ไปเป็น YCbCr จากนั้นทำการซับแซมปลิ้ง (Subsampling) ข้อมูลภาพในอัตราส่วน Y:Cb:Cr เท่ากับ 4:1:1 และผ่านกระบวนการแปลงค่าสัมประสิทธิ์ DCT และควอนไทเซชั่น (Quantization) ซึ่งเป็นการปรับลดค่าสัมประสิทธิ์ DCT แล้วนำค่ามาเรียงข้อมูลแบบซิกแซก (Zigzag) โดยการเรียงค่าสัมประสิทธิ์ DCT ในบล็อกข้อมูล 8×8 พิกเซล โดยเรียงจากค่าสัมประสิทธิ์ของความถี่ต่ำไปหาค่าสัมประสิทธิ์ความถี่สูง ในขั้นสุดท้ายคือ การเข้ารหัสฮัฟแมน (Huffman encoding) และเขียนข้อมูลออกมาเป็นไฟล์ JFIF (JPEG File Interchange Format)

โปรแกรมสามารถเปลี่ยนค่าควอนไทเซอร์ที่ใช้ในขั้นตอนควอนไทเซชั่นได้หลายระดับ ซึ่งมีผลต่ออัตราส่วนการบีบอัดข้อมูล(Compression ratio) และคุณภาพของภาพที่ถูกบีบอัด โดยที่อัตราส่วนการบีบอัดข้อมูลจะแปรผันตามค่าควอนไทเซอร์และคุณภาพของภาพที่ถูกบีบอัดจะลดลงอย่างมากเมื่อใช้ควอนไทเซอร์ที่มีค่าสูง

Project Title Study of JPEG Image Compression Standard

Name Miss Patcharin Foonthong ID. 44362705

Project Advisor Miss Siriporn Dachasilaruk

Major Computer Engineering

Department Electrical and Computer Engineering

Academic Year 2004

.....

ABSTRACT

This project is to study JPEG image compression standard using JPEG-type sequential baseline system. Image from Windows Bitmap file (.bmp) type 24 bits/pixel is compressed with subsampling rate ratio Y:Cb:Cr is 4:1:1. For JPEG encoding, the input image pixel samples are grouped into 8×8 blocks . Each block is transformed by the Forward DCT (FDCT) into a set of 64 values known as a DCT coefficient. The 64 DCT coefficients are quantized. This essentially means reducing their accuracy by some degree usually resulting in some of them being reduced to zero DCT coefficients. The coefficients are prepared for entropy encoding by converting them into what is known as a zigzag sequence. This effectively takes a two – dimensional array of samples and convert into a one – dimensional array ordered as nearly a possible in ascending order of frequency components. The zigzag sequence is then encoded by using Huffman encoding. The compressed data are saved in JFIF file (.jpg).

Program can change quantizer value which is used in quantization in many levels that have effect to the compression ratio and the quality of compressed image. Compression ratios will increase with quantizer value. The quality of compressed image will degrade with increasing quantizer value.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จได้ ด้วยความกรุณาของอาจารย์ศิริพร เดชะศิลารักษ์ อาจารย์ที่ปรึกษาโครงงาน ซึ่งได้ให้คำปรึกษา ข้อชี้แนะ และความช่วยเหลือในหลายสิ่งหลายอย่างจนกระทั่งลุล่วงไปด้วยดี ผู้ทำโครงงานขอกราบขอบพระคุณเป็นอย่างสูงมา ณ ที่นี้

ขอขอบพระคุณ ผู้ช่วยศาสตราจารย์ ดร.สุชาติ แย้มเม่น และ ดร.พนมขวัญ ริยะมงคล ที่ได้กรุณาเป็นกรรมการคุมสอบโครงงานนี้

ขอขอบพระคุณ ภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้มี การเรียนการสอนรายวิชานี้ ทำให้เหมือนได้ศึกษาชีวิตการทำงาน ก่อนที่จะก้าวออกไปเผชิญชีวิตการทำงานจริง ในอนาคต

ขอขอบพระคุณ คณะวิศวกรรมศาสตร์ที่ได้ให้ทุนสนับสนุนการทำการโครงงานนี้

ขอขอบคุณและขอบใจ พี่ เพื่อน และน้องภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ทุกคน ที่คอยไถ่ถามด้วยความห่วงใยว่าเมื่อไหร่จะสำเร็จการศึกษา และขอขอบคุณเพื่อนในชั้น ที่ช่วยในคำแนะนำในการเขียนโปรแกรม และช่วยเหลือในบางโอกาส รวมถึงผู้มีพระคุณทุกท่านที่ไม่ได้เอ่ยนามไว้ ณ ที่นี้ด้วย

สุดท้ายนี้ขอกราบขอบพระคุณแม่อัญชรีพร อ่อนปาน บุพการีผู้ให้ทุกอย่างกับผู้ทำโครงงานนี้

นางสาว พัฒชรินทร์ ฝุ่นทอง

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ก

บทคัดย่อภาษาอังกฤษ ..... ข

กิตติกรรมประกาศ..... ค

สารบัญ ..... ง

สารบัญตาราง ..... ฉ

สารบัญรูป..... ซ

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญ..... 1

1.2 วัตถุประสงค์..... 2

1.3 ขอบข่าย..... 2

1.4 ขั้นตอนการดำเนินงาน..... 2

1.5 ผลที่คาดว่าจะได้รับ..... 3

1.6 งบประมาณ..... 3

บทที่ 2 หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 หลักการบีบอัดภาพโดยทั่วไป..... 4

2.1.1 การบีบอัดแบบไม่มีการสูญเสีย (Lossless Compression)..... 4

2.1.2 การบีบอัดข้อมูลแบบสูญเสียข้อมูล (Lossy Compression)..... 5

2.2 มาตรฐานการบีบอัดข้อมูลภาพ JPEG 7

2.3 ไฟล์ภาพชนิดต่าง 7

2.3.1 ไฟล์ภาพ BMP(Bitmap)..... 7

2.3.2 ไฟล์ภาพ GIF (Graphics Interchange Format)..... 8

2.3.3 ไฟล์ภาพ JPEG (Joint Photographic Experts Group)..... 8

2.3.4 ไฟล์ภาพ PNG (Portable Network Graphics) 8

2.3.5 ไฟล์ภาพ TIFF (Tag Image File Format) 9

สารบัญ (ต่อ)

หน้า

2.4 การบีบอัดข้อมูลภาพ JPEG ชนิด Sequential Baseline System..... 9

2.4.1 ไฟล์ข้อมูลภาพต้นแบบ.....9

2.4.2 ระบบสี.....11

2.4.3 การซับแซมปลิ้งสี (Color Subsampling)..... 12

2.4.4 การแปลง DCT (Discrete Cosine Transform) 12

2.3.5 การเข้ารหัสฮัฟแมน(Huffman Encoding).....19

บทที่ 3 การออกแบบและเขียนโปรแกรม

3.1 การเตรียมข้อมูลสำหรับการบีบอัดข้อมูลภาพนิ่ง JPEG 24

3.1.1 การแบ่งเป็น MCU (Minimum Coded Unit)..... 24

3.1.2 การแปลงระบบสี RGB ให้เป็น YCbCr..... 25

3.1.3 การซับแซมปลิ้ง (Subsampling) 25

3.2 การบีบอัดข้อมูล JPEG 26

3.2.1 การแปลงบล็อกข้อมูลด้วยวิธี DCT 26

3.2.2 การควอนไทเซชั่น (Quantization) 27

3.2.3 การเรียงข้อมูลแบบ ซิกแซก (Zigzag) 28

3.2.4 การเข้ารหัสฮัฟแมน(Huffman Encoding)..... 29

3.3 การเก็บข้อมูลในรูปไฟล์ JFIF 39

3.3.1 เฟรมเฮดเดอร์ 41

3.3.2 โครงสร้างสแกนเฮดเดอร์.....42

3.3.3 โครงสร้างตารางค่าควอนไทเซชั่น..... 44

3.3.4 โครงสร้างตารางฮัฟแมน 45

3.3.5 โครงสร้างการกำหนดช่วง Restart Interval Marker.....47

3.3.6 ส่วนของข้อมูล.....47

3.4 สรุปโฟล์วชาร์ทของขั้นตอนบีบอัดข้อมูล JPEG 47

สารบัญ (ต่อ)

หน้า

บทที่ 4 ผลการทดลอง

4.1 การทำงานของโปรแกรม 52

4.2 การบีบอัดข้อมูลภาพที่ค่า Quantize Factor ต่างๆ 55

4.3 ตัวอย่างภาพที่ใช้ในการทดลอง..... 57

4.4 กราฟแสดงความสัมพันธ์ระหว่าง Quantize Factor และ Compression radio 60

บทที่ 5 สรุปผลและข้อเสนอแนะ

5.1 สรุปผล 61

5.2 ข้อเสนอแนะ 61

เอกสารอ้างอิง 62

ประวัติผู้เขียนโครงงาน..... . 63

สารบัญตาราง

ตารางที่		หน้า
1.1	ตารางการดำเนินงาน	2
2.1	Bitmap File Header	10
2.2	Bitmap Information	10
2.3	น้ำหนักสัมประสิทธิ์ DCT 1 มิติ	15
2.4	ค่าพิกเซลเริ่มต้น	18
2.5	สัมประสิทธิ์ความถี่	18
2.6	สัมประสิทธิ์ การควอนไทซ์	18
2.7	สัมประสิทธิ์ความถี่ควอนไทซ์	19
2.8	ค่าพิกเซลใหม่ที่ได้จาก IDCT	19
2.9	ค่าความผิดพลาด	19
3.1	JPEG Coefficient Coding Categories	29
3.2	ตารางฮัฟแมนของลูมิแนนซ์ AC	31
3.3	ตารางฮัฟแมนของ โครมิแนนซ์ AC	34
3.4	สรุปความยาว ลูมิแนนซ์ และโครมิแนนซ์ของสัมประสิทธิ์ AC	37
3.5	ตารางฮัฟแมนของลูมิแนนซ์ DC	37
3.6	ตารางฮัฟแมนของโครมิแนนซ์DC	37
3.7	บล็อกข้อมูลภาพตัวอย่าง ขนาด 8× 8 พิกเซล	38
3.8	บล็อกข้อมูลภาพตัวอย่าง 8 ×8 พิกเซล ลบกับ 128	38
3.9	บล็อกข้อมูลภาพที่ทำการแปลง FDCT แล้ว	38
3. 10	สัมประสิทธิ์การควอนไทซ์	38
3.11	ค่าที่ได้จากการควอนไทซ์แล้ว	39
3.12	มาร์คเกอร์ที่ใช้กับ JPEG	40
3.13	ส่วนประกอบของเฟรมเฮดเดอร์	41
3.14	การกำหนดค่าพารามิเตอร์ของแต่ละองค์ประกอบ	42
3.15	ส่วนประกอบของสแกนเฮดเดอร์	44
3.16	ฟิลด์ตารางการควอนไทเซชั่น	45

สารบัญตาราง (ต่อ)

ตารางที่	หน้า
3.17 ตารางฮัฟแมน	46
3.18 การเริ่มมาร์คเกอร์ใหม่	47
3.19 ส่วนของข้อมูล	47
4.1 ผลการทดสอบการบีบอัดข้อมูลภาพบิตแมพ	55

สารบัญรูป

รูปที่ หน้า

2.1 การบีบอัดข้อมูลภาพแบบไม่มีการสูญเสียข้อมูล 5

2.2 การบีบอัดแบบมีการสูญเสียข้อมูล 6

2.3 การแปลงไปข้างหน้า..... 6

2.4 การแปลงย้อนกลับ 6

2.5 ส่วนประกอบของบิตแมพ 10

2.6 ความสัมพันธ์ระหว่างค่าสัมประสิทธิ์ DCT 1 มิติ..... 13

2.7 ตัวอย่างน้ำหนักของสัมประสิทธิ์ 15

2. 8 รูปแบบความหนาแน่นของตัวอย่างของ DCT 2 มิติ 16

2.9 ภาพต้นแบบและภาพใหม่ที่ผ่าน DCT และการควอนไทซ์ 17

2.10 ระดับ การควอนไทซ์ขององค์ประกอบความถี่ต่างๆ จากมากสุดไปน้อยสุด 18

2. 11 เรียงข้อมูลตามลำดับความถี่จากมากไปน้อย 20

2. 12 นำความถี่น้อยที่สุด 2 ค่ามารวมกัน 20

2. 13 รวมค่าความถี่ที่สุดที่สุดในลำดับต่อมา 20

2.14 โครงสร้างทรี การหารหัสฮับแมนตามลำดับ 21

2.15 รหัสฮัฟแมนที่ได้จากทรี 22

3.1 บล็อกไดอะแกรมของโปรแกรม 24

3.2 ตัวอย่างการซับแซมปลิ้งด้วยอัตราส่วน H: 2 , V: 2 25

3.3 องค์ประกอบภายใน MCU เมื่อใช้อัตราส่วน H : 2 ,V : 2
ในการซับแซมปลิ้งค่า Cb และ Cr 26

3.4 แสดงลำดับการจัดข้อมูลแบบ Zigzag 28

3.5 รูปแบบการบีบอัดข้อมูลภาพแบบ Sequential DCT-based 40

3.6 โครงสร้างของเฟรมเฮดเดอร์ 41

3.7 โครงสร้างของสแกนเฮดเดอร์ 43

สารบัญรูป(ต่อ)

รูปที่		หน้า
3.8	ส่วนประกอบของตารางควอนไทเซชั่น	45
3.9	ส่วนประกอบของมาร์คเกอร์ตารางฮัฟแมน	46
3.10	ขั้นตอนการเข้ารหัสภาพ	48
3.11	การเข้ารหัสเฟรม	48
3.12	การเข้ารหัสในส่วนของสแกน	49
3.13	ขั้นตอนสำหรับเข้ารหัส Restart Interval	50
3.14	ลำดับขององค์ประกอบภาพใน MCU	50
3.15	ขั้นตอนสำหรับเข้ารหัส MCU	50
4.1	หน้าตาของโปรแกรม	52
4.2	เลือกไฟล์ภาพต้นแบบ	53
4. 3	ภาพต้นแบบที่ได้	53
4.4	ภาพเปรียบเทียบระหว่างภาพต้นแบบและภาพที่ถูกบีบอัดแล้ว	54
4.5	รูป Lena ที่ระดับ Q.F. ต่างๆ	57
4.6	รูป Mandrill ที่ระดับ Q.F. ต่างๆ	58
4.7	รูป perper ที่ระดับ Q.F. ต่างๆ	59
4.8	กราฟความสัมพันธ์ระหว่าง ค่า Quantizer Factor และ ค่า Compress Ratio	60

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญ

ในการรับรู้ของมนุษย์นั้นได้การแก่ ตาดูสิ่งที่เห็น เช่น ภาพ เป็นต้น หูฟังเสียงที่ได้ยิน จมูกรับกลิ่น มือสัมผัส ปากพูดคุย มนุษย์ใช้ความสามารถที่ติดตัวมาแต่กำเนิดในการติดต่อสื่อสารกับผู้อื่น ซึ่งการสื่อสารที่ได้ผลดีได้แก่ การใช้ภาพเพราะภาพสามารถอธิบายความหมายได้ดีกว่าคำพูดนับพันคำ ซึ่งบางคนไม่จำเป็นต้องมีคำพูดประกอบภาพก็สามารถที่จะเข้าใจได้ทันที ดังนั้นจึงได้มีการใช้ภาพที่ในสื่อความหมายอย่างแพร่หลาย โดยใช้เป็นภาษาสัญลักษณ์ที่มีความเป็นสากล ทำให้คนต่างชาติต่างภาษาได้สามารถสื่อสารกันได้ เช่น ภาพป้ายสัญญาณจราจร ป้ายห้องน้ำ แผ่นป้ายโฆษณาต่างๆเป็นต้น ในการศึกษาของโครงงานนี้เป็นการศึกษาภาพดิจิตอล ในปัจจุบันได้มีการใช้การอย่างแพร่หลาย ทั้งทางคอมพิวเตอร์ในรูปแบบของเว็บเพ็จ มือถือที่มีกล้องติดอยู่ เป็นต้น เราจะพบว่าภาพที่เราเห็นมีหลายขนาด ถ้ามองลึกไปในคุณสมบัติของภาพ จะเห็นว่าไฟล์ภาพเหล่านั้นมีการจัดเก็บที่ไม่เหมือนกัน บางภาพมีขนาดใหญ่มาก เวลาใช้แต่ละครั้งต้องเสียเวลาโหลดเป็นเวลานาน และเปลืองเนื้อที่ในการเก็บข้อมูลด้วย การมองเห็นปัญหาในด้านนี้ทำให้มีผู้คิดค้นที่จะทำให้ภาพมีขนาดเล็กลง วิธีการนี้เรียกว่าการบีบอัดข้อมูลภาพ (Image Compression)

เนื่องจากความคับคั่งของสื่อสารบนอินเทอร์เน็ต ทำให้การส่งผ่านข้อมูลทำได้ช้าลง แม้ว่าปัจจุบันจะได้มีการขยายแบนวิธ (Bandwidth) ออกไปอย่างมากก็ตาม แต่ก็ยังไม่เพียงพอต่อการใช้งาน การบีบอัดข้อมูลก่อนที่จะส่งข้อมูลไปยังผู้ใช้จึงมีความจำเป็นอย่างหลีกเลี่ยงไม่ได้ ไฟล์กราฟิกที่ใช้บนเว็บเพ็จจึงจำเป็นต้องมีขนาดเล็กที่สุดและด้วยเทคโนโลยีการบีบอัดไฟล์รูปแบบไฟล์กราฟิกบนอินเทอร์เน็ตเกือบทั้งหมดจึงอยู่ในรูปแบบที่ถูกบีบอัด

ในโครงงานได้เลือกทำการศึกษาไฟล์ภาพ JPEG เพราะมีการใช้กันอย่างแพร่หลาย ทั้งทางอินเตอร์เน็ท กล้องถ่ายรูปดิจิตอล กล้องถ่ายรูปดิจิตอลที่ติดบนมือถือ ประการสำคัญคือมีการเผยแพร่หลักการบีบอัด ตัวโปรแกรมที่ใช้ในการบีบอัด ซึ่งเป็นประโยชน์ต่อผู้ที่สนใจจะทำการศึกษาและพัฒนาต่อไป

1.2 วัตถุประสงค์

1. เพื่อศึกษามาตรฐานการบีบอัดภาพนิ่ง JPEG ชนิด Sequential Baseline System
2. เพื่อศึกษาวิธีการจัดเก็บของข้อมูลภาพนิ่ง JPEG ชนิด Sequential Baseline System
3. เพื่อเป็นแนวทางให้กับผู้ที่สนใจเรื่องมาตรฐานการบีบอัดได้พัฒนาต่อในลำดับต่อไป

1.3 ขอบเขต

1. ภาพต้นแบบ(Original Image) ที่ใช้ในการบีบอัดข้อมูลภาพเป็นแบบบิตแมพ (Bitmap) โหมดสี RGB ชนิด 24 บิต/พิกเซล ขนาด 256×256 พิกเซล
2. ศึกษามาตรฐานการบีบอัดข้อมูลภาพนิ่งJPEG ชนิด Sequential Baseline System
3. ใช้ภาษาจาวา(JAVA) ในการพัฒนาโปรแกรมบีบอัดข้อมูลภาพนิ่ง JPEG บนระบบปฏิบัติการของไมโครซอฟท์วินโดว์ (Microsoft Window)
4. ศึกษาคุณภาพของภาพที่ได้จากการบีบอัดข้อมูลภาพและอัตราการบีบอัดข้อมูลภาพ JPEG โดยเปรียบเทียบกับภาพต้นแบบ

1.4 แผนการดำเนินงาน

ตารางที่ 1.1 ตารางการดำเนินงาน

การดำเนินงาน	พ.ย. 46	ธ.ค. 46	ม.ค. 47	ก.พ. 47	มี.ค. 47	เม.ย. 47	พ.ค. 47	มิ.ย. 47	ก.ค. 47	ส.ค. 47	ก.ย. 47	ต.ค. 47
1. ศึกษาค้นคว้าข้อมูล	←	—	→									
2. ศึกษาโปรแกรมภาษาจาวา				←	—	→						
3. เขียนโปรแกรม												
- อ่านไฟล์ภาพ *.bmp					↔							
- การแปลง DCT						↔						
- การควอนไทเซชั่น (quantization)								↔				
- การเขารหัสฮัฟแมน (Huffman coding)									↔			
- เขียนไฟล์ภาพ *.jpg										↔	↔	
4. ทดสอบโปรแกรม											↔	
5. ทำรายงาน												↔

1.5 ผลที่คาดว่าจะได้รับ

1. เข้าใจหลักการบีบอัดข้อมูลภาพโดยทั่วไป
2. เข้าใจวิธีการบีบอัดภาพนิ่ง JPEG ชนิด Sequential Baseline System
3. เข้าใจรูปแบบการจัดเก็บของไฟล์ภาพ JPEG
4. เป็นความรู้พื้นฐานสำหรับผู้ที่มีความสนใจเรื่องการบีบอัดข้อมูลภาพ
5. สามารถนำความรู้ที่ได้พัฒนาต่อในมาตรฐานการบีบอัดข้อมูลแบบวีดีโอหรือ MPEG

1.6 งบประมาณ

1. ค่าวัสดุคอมพิวเตอร์ เป็นจำนวน 200 บาท
2. ค่าจ้างถ่ายเอกสาร เป็นจำนวน 600 บาท
3. ค่าพิมพ์เอกสาร เป็นจำนวน 200 บาท

รวมเป็นเงิน 1000 บาท

(ถัวเฉลี่ยทุกรายการ)

บทที่ 2
หลักการและทฤษฎีที่เกี่ยวข้อง

2.1 หลักการบีบอัดข้อมูลภาพทั่วไป

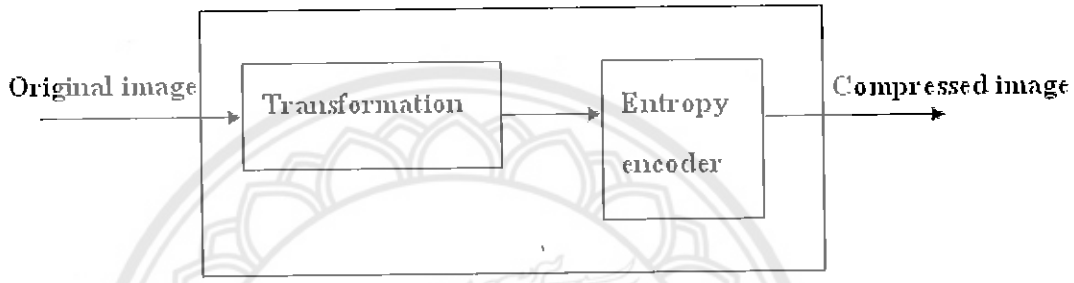
การบีบอัดภาพ คือการลดจำนวนของข้อมูลภาพ โดยนำข้อมูลที่มีความความสำคัญน้อยออก ในการบีบอัดข้อมูลภาพได้มีผู้สนใจมากกว่า 35 ปี จนพัฒนามาเป็นมาตรฐานการบีบอัดข้อมูลภาพในศตวรรษที่ 1940 ในปัจจุบันการบีบอัดข้อมูลภาพรู้จักกันในนาม "Enabling technology" [1] การบีบอัดข้อมูลภาพคือเทคโนโลยีธรรมชาติสำหรับการเพิ่มความคมชัดของภาพและมาตรฐานการแพร่ภาพโทรทัศน์ ในอนาคตการบีบอัดข้อมูลภาพจะเข้ามามีบทบาทสำคัญและสามารถนำมาใช้ให้เกิดประโยชน์คือ

1. การประชุมทางวิดีทัศน์ (Videoconference) คือการประชุมหรือสัมมนาผ่านทางเครือข่ายคอมพิวเตอร์ ผู้ร่วมประชุมไม่จำเป็นต้องอยู่ที่เดียวกันแต่สามารถพูดคุยกันได้ มองเห็นกันโดยผ่านทางจอภาพของคอมพิวเตอร์ ส่วนหนึ่งของจอภาพอาจแบ่งไว้เป็นที่สำหรับเขียนข้อความหรือภาพ หรือแสดงข้อมูลจากโปรแกรมที่เตรียมไว้ ลักษณะการทำงานเป็นระบบเครือข่าย (Network)
2. การควบคุมทางไกล (Remote sensing) คือการส่งสัญญาณมาจากดาวเทียมสำหรับการพยากรณ์ อากาศหรือสิ่งอื่นๆที่เป็นประโยชน์ต่อโลก
3. เอกสารและภาพทางการแพทย์
4. การรับส่งแฟกซ์ (FAX) ในปัจจุบันอาจจะต่อสารแฟกซ์ผ่านโมเด็มเข้าสู่ตัวเครื่องคอมพิวเตอร์ได้ ในกรณีนี้จะเป็นการรับหรือส่งข้อมูลจากคอมพิวเตอร์เลยไม่ต้องใช้กระดาษ คล้ายจดหมายอิเล็กทรอนิกส์
5. การควบคุมระยะไกลของการขับเคลื่อนเครื่องบินทางการทหาร
6. การควบคุมจรวดในอวกาศ
7. การควบคุมเครื่องจักรในที่อันตราย

ภาพจะสามารถทำไปใช้ประโยชน์ได้มากน้อยเพียงใดขึ้นอยู่กับคุณภาพของภาพ การบีบอัดข้อมูลภาพมีจุดมุ่งหมายเพื่อลดขนาดเนื้อที่ที่ใช้เก็บข้อมูลและลดเวลาในการโหลดข้อมูล โดยการนำข้อมูลมาผ่านกระบวนการต่างๆเพื่อให้ได้ข้อมูลภาพที่ถูกบีบอัดแล้วมีจำนวนบิตที่ใช้ในการจัดเก็บน้อยลง การบีบอัดข้อมูลภาพแบ่งออกเป็น 2 ประเภท คือ

1. การบีบอัดแบบไม่มีการสูญเสีย (Lossless Compression) เป็นการบีบอัดที่ข้อมูลเดิมไม่มีการสูญเสียเลย ภาพที่ถูกบีบอัดด้วยวิธีนี้ จะทำการบีบอัดและขยายภาพคืนแบบพิกเซลต่อพิกเซลเหมือนภาพต้นแบบ ไฟล์ภาพที่ได้จะมีความละเอียดสูง แต่จะมีขนาดใหญ่ทำให้เปลืองหน่วยความจำ การบีบอัดแบบไม่มีการสูญเสียข้อมูลที่นิยมใช้ในกล้องดิจิตอล ได้แก่ ไฟล์นามสกุล BMP, PNG, TIF

วิธีการบีบอัดแบบไม่มีการสูญเสียคือ นำข้อมูลภาพต้นแบบ (Original image) มาเข้ากระบวนการเข้ารหัส โดยการผ่านการแปลงข้อมูล (Transformation) และการเข้ารหัสเอนโทรปี (Entropy encoder) จะได้ภาพที่ถูกบีบอัด (Compressed image)

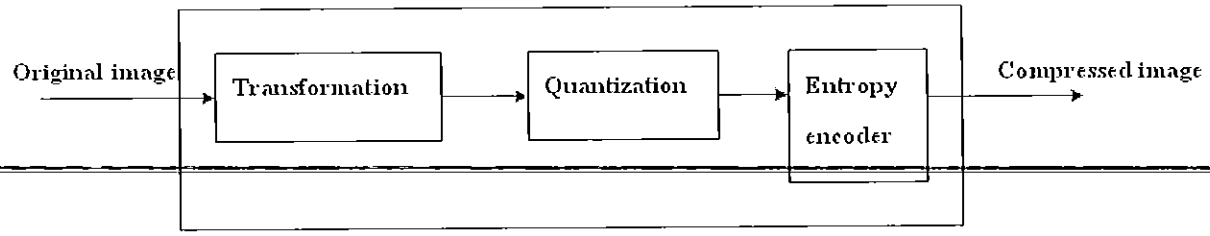


รูปที่ 2.1 การบีบอัดข้อมูลภาพแบบไม่มีการสูญเสียข้อมูล

2. การบีบอัดข้อมูลแบบสูญเสียข้อมูล (Lossy Compression) เป็นการบีบอัดข้อมูลภาพในลักษณะที่มีการตัดทอนข้อมูลภาพบางส่วนออกไป เพื่อให้ไฟล์ภาพมีขนาดเล็กลง แต่จะทำให้สูญเสียข้อมูลบางอย่างไปและไม่สามารถเรียกกลับคืนได้ เช่น สีผิดเพี้ยน เป็นต้น การบีบอัดแบบสูญเสียข้อมูลที่นิยมใช้ในกล้องดิจิตอล ได้แก่ ไฟล์นามสกุล JPG

การบีบอัดข้อมูลแบบสูญเสียข้อมูลนั้นจะตรวจดูกลุ่มของพิกเซล เพื่อจะตัดสินใจว่าการเปลี่ยนแปลงระหว่างพิกเซลนั้นเป็นอย่างไรแล้วทำการบันทึกค่าของการเปลี่ยนแปลงระหว่างพิกเซลแทนที่จะเป็นค่าของแต่ละพิกเซล การบีบอัดแบบนี้แม้ว่าจะสูญเสียข้อมูลบางอย่างไป แต่จะเป็นการใช้ประโยชน์ของสายตามนุษย์ที่ไวต่อการเปลี่ยนแปลงของความสว่างหรือความเข้มของแสง แต่จะไม่ไวต่อการเปลี่ยนแปลงของสีหรือโทนสีมากนัก ภาพที่บีบอัดจึงมีการเก็บข้อมูลเกี่ยวกับการเปลี่ยนแปลงด้านความสว่างระหว่างพิกเซลมากกว่าการเปลี่ยนแปลงของสีระหว่างพิกเซล

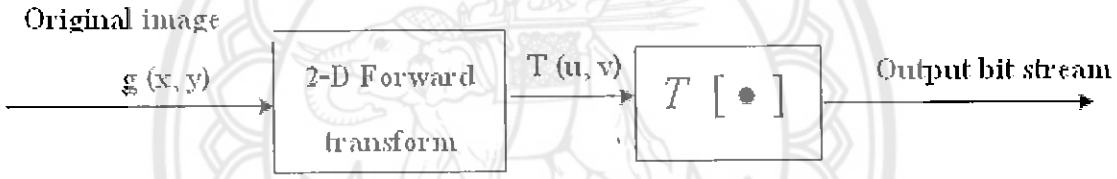
การบีบอัดข้อมูลแบบสูญเสียข้อมูล มีขั้นตอนคือ นำข้อมูลภาพต้นแบบ (Original image) มาเข้ากระบวนการบีบอัดข้อมูลภาพ โดยผ่านการแปลงข้อมูล (Transformation) และผ่านการควอนไท-เซชั่น (Quantization) และเข้ารหัสเอนโทรปี (Entropy encoder) จะได้ภาพที่ถูกบีบอัด (Compressed image)



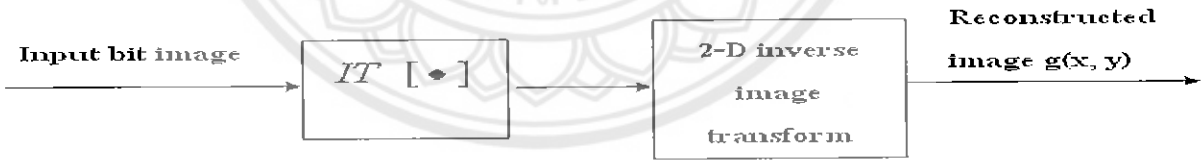
รูปที่ 2.2 การบีบอัดแบบมีการสูญเสียข้อมูล

การกระบวนบีบอัดข้อมูลภาพทั่วไป สามารถแยกออกเป็นส่วนๆได้ดังนี้

1.การแปลงข้อมูล (Transformation) คือการกระทำข้อมูลด้วยวิธีการทางคณิตศาสตร์ใดๆ แล้วทำให้ข้อมูลมีการเปลี่ยนแปลงไป เพื่อให้สามารถนำไปใช้ในขั้นตอนต่อไป มีทั้งการแปลงแบบ แปลงไปข้างหน้า(Forward transform) และการแปลงแบบย้อนกลับ (Inverse transform) สมการการแปลงได้แก่ การแปลง DFT (Discrete Fourier Transform) การแปลง DCT (Discrete Cosine Transform)



รูปที่ 2.3 การแปลงไปข้างหน้า



รูปที่ 2.4 การแปลงย้อนกลับ

2.การควอนไทเซชั่น (Quantization) คือการลดความแม่นยำของข้อมูล เพื่อทำให้ข้อมูลน้อยลง ง่ายต่อการเข้ารหัสและถอดรหัสในการควอนไทเซชั่นนี้ถ้าจะทำให้ข้อมูลหายไปไม่สามารถเรียกกลับคืนได้

3.การทำจำนวนของบิตให้น้อยที่สุด (Minimization of number of bits) หรือการเข้ารหัส ซึ่งการเข้ารหัสมีหลายวิธีการได้แก่ การเข้ารหัสฮัฟแมน (Huffman coding), การเข้ารหัสอะริธเมติก (Arithmetic coding) การเข้ารหัสเลมเปล (LZW coding หรือ Lempel-Ziv-Welch) เป็นต้น

2.2 มาตรฐานการบีบอัดข้อมูลภาพ JPEG

ข้อมูลภาพ JPEG (Joint Photographic Experts Group) ได้พัฒนาขึ้นโดยคณะกรรมการของผู้เชี่ยวชาญด้านคอมพิวเตอร์กราฟิกที่สนับสนุนร่วมกันโดยองค์การระหว่างประเทศว่าด้วยมาตรฐาน (ISO) และคณะกรรมการที่ปรึกษาระหว่างประเทศเกี่ยวกับโทรเลขและโทรศัพท์ (CCITT) ในการพัฒนามาตรฐาน JPEG เป็นแนวคิดอันดีสำหรับภาพของธรรมชาติที่มีความซับซ้อน ซึ่งรวมถึงภาพถ่าย ภาพศิลปะ และภาพวาด (รูปแบบนี้ไม่เหมาะสำหรับภาพวาดลายเส้น ข้อความ หรือภาพการ์ตูนง่ายๆ) JPEG ใช้การบีบอัดภาพแบบมีการสูญเสียข้อมูลเข้าช่วย โดยอาศัยหลักการมองเห็นของมนุษย์ที่ว่าการเปลี่ยนแปลงของสีเพียงเล็กน้อยจะสังเกตเห็นได้ไม่ชัดเจนเท่ากับการเปลี่ยนแปลงของแสง นอกจากนี้การบีบอัดข้อมูล JPEG จะไม่สามารถสังเกตเห็นขอบบล็อกหรือสัญญาณรบกวนได้ ถ้าไม่ใช้อัตราส่วนของการบีบอัดที่สูง

การบีบอัดข้อมูลภาพ JPEG แบ่งออกเป็น 3 ระบบ ได้แก่

1.Baseline Coding System เป็นระบบการแปลงโดยใช้ DCT ยอมสูญเสียข้อมูลบางส่วนไป มักถูกเรียกอีกชื่อว่า Sequential Baseline System

2.Extended Coding System สำหรับภาพที่ต้องการการบีบอัดดีกว่า มีความแม่นยำสูงกว่า Baseline Coding System ทำให้ได้ภาพที่มีรายละเอียดสูงกว่า Baseline Coding System

3..Independent Coding System การบีบอัดที่สามารถย้อนกลับไปเป็นภาพต้นแบบได้โดยไม่มีการสูญเสียข้อมูล

การบีบอัดข้อมูลภาพ JPEG ทั้ง 3 ระบบที่กล่าวมาแล้วนั้น จะเห็นได้ว่า การระบบ Sequential Baseline System เป็นระบบที่ซับซ้อนน้อยที่สุด มีการคำนวณที่ง่ายกว่าระบบอื่นๆ และสะดวกต่อการจัดเก็บข้อมูลเพราะไม่ต้องการความละเอียดของข้อมูลมากนัก ทำให้ได้ภาพที่ถูกบีบอัดมามีขนาดไฟล์ภาพที่เล็กกว่าระบบอื่นๆ

2.3 ไฟล์ภาพชนิดต่างๆ

ไฟล์ภาพที่นิยมใช้ในปัจจุบันมีหลายชนิดได้แก่

2.3.1 ไฟล์ภาพ BMP (Bitmap) เป็นรูปแบบไฟล์มาตรฐานที่ใช้กันอยู่ทั่วไปบนระบบปฏิบัติการวินโดว์และดอส ไฟล์รูปแบบนี้รองรับหลายชนิดการทำงาน ได้แก่ 1,4,8,24 บิต/พิกเซล ซึ่งเป็นแบ่งเป็นภาพขาวดำและภาพสี ระบบสีที่ใช้เป็นระบบสี RGB มีการไล่ระดับสีได้ 256 ระดับทำให้รวมสีได้ 16 ล้านสี สามารถนำไปใช้ในงานภาพเคลื่อนไหวได้ บิตแมพไม่ต้องการเก็บข้อมูลของรหัสสีสำหรับแต่ละพิกเซลในทุกแถว ต้องการเฉพาะข้อมูลที่ชี้สีใหม่ที่ใช้แสดงของแถว ดังนั้นภาพที่ใช้สีมากต้องการบิตแมพขนาดเล็กเนื่องจากบิตแมพใช้วิธีการแบบราสเตอร์กราฟิก (Raster graphic) ในการกำหนดภาพ ทำให้ภาพไม่สามารถขยายสัดส่วนเพราะทำให้สูญเสียความคมชัด ในขณะที่ภาพแบบ

เวกเตอร์กราฟิก (Vector graphic) ได้รับออกแบบให้ขยายสัดส่วนได้ทันที ทำให้มีความนิยมในการสร้างภาพแบบเวกเตอร์กราฟิกก่อนแล้วจึงแปลงเป็นภาพแบบราสเตอร์หรือเป็นบิตแมพ

2.3.2 ไฟล์ภาพแบบ GIF (Graphic Interchange Format) เป็นรูปแบบที่นิยมกันมากที่สุดในปัจจุบัน GIF ไฟล์มีสองเวอร์ชันคือ 87a และ 89a โดยที่เวอร์ชัน 87a เป็นรูปกราฟิกเพียงอย่างเดียว ขณะที่เวอร์ชัน 89a สามารถสนับสนุนการทำภาพเคลื่อนไหวได้ โดยการนำภาพหลายๆ ภาพมาเรียงต่อกันและบรรจุอยู่ในไฟล์เดียวกัน ข้อดีของ GIF คือสามารถนำไปสร้างเป็นภาพเคลื่อนไหวได้ นอกจากนี้ยังสามารถทำให้พื้นหลังโปร่งใส (Transparent) เพื่อให้ภาพนั้นกลมกลืนกับภาพพื้นหลัง (Background) ลักษณะสำคัญอีกประการหนึ่งของไฟล์รูปแบบ GIF คือ การสอดประสาน (Interface) เป็นการแสดงภาพในขณะที่กำลังรอให้ปรากฏเป็นชั้นๆ โดยที่แต่ละชั้นจะค่อยๆ เพิ่มความชัดของภาพขึ้นมาจนปรากฏทั้งหมด ซึ่งเมื่อกราฟิกถูกโหลดจะแสดงภาพจากหยาบไปหาละเอียด ในขณะที่กราฟิกถูกโหลดมาทีละน้อยจนเสร็จสมบูรณ์จึงทำให้ผู้ชมที่มีแบนวิธต่ำๆ สามารถเห็นภาพโดยรวมได้ก่อนที่ภาพจะถูกโหลดจนสมบูรณ์ ข้อเสีย ของไฟล์GIF คือเป็นลิขสิทธิ์ของบริษัท Compuserv ซึ่งเป็นบริษัทผู้ผลิตซอร์ฟแวร์จะใช้ได้จะต้องขออนุญาตและจ่ายค่าลิขสิทธิ์เพื่อใช้ไฟล์ GIF จึงทำให้มีผู้พัฒนาไฟล์รูปแบบ PNG ขึ้นใช้แทน

2.3.3 **ไฟล์ภาพ JPEG (Joint Photographic Experts Group)** เป็นรูปแบบที่พัฒนาขึ้นเพื่อมุ่งเน้นการบีบอัดไฟล์กราฟิกให้มีขนาดเล็กที่สุด ดังนั้นเมื่อเราสร้างไฟล์ประเภทนี้ต้องเลือกว่าต้องการให้ไฟล์มีคุณภาพเป็นเท่าไหร่ เพราะเมื่อกราฟิกมีคุณภาพที่ดีขึ้นก็จะมีขนาดของไฟล์ที่ใหญ่ขึ้นตามไปด้วย เราจึงต้องเลือกระหว่างคุณภาพของกราฟิกกับขนาดของไฟล์เสมอ ข้อดีของ JPEG นอกจากความสามารถในการบีบอัดไฟล์แล้ว JPEG ยังใช้ในเว็บที่ต้องการเน้นภาพในลักษณะที่ให้ความคมชัดสูง นอกจากนี้ ยังสามารถแสดงคุณสมบัติพิเศษที่เรียกว่า Progressive คือภาพจะค่อยๆ ปรากฏคล้ายกับภาพสอดประสานในรูปแบบไฟล์ GIF

2.3.4 ไฟล์ภาพ PNG (Portable Network Graphics) เป็นรูปแบบไฟล์กราฟิกแบบบีบอัดที่พัฒนาขึ้นเพื่อมาแทน GIF โดยคณะกรรมการอินเทอร์เน็ต จึงเน้นให้ทุกคนสามารถใช้ได้ฟรี คุณสมบัติทั่วไปของ PNG จะคล้ายกับ GIF คือ บีบอัดภาพได้โดยไม่สูญเสียคุณภาพ ดังนั้น PNG จึงไม่ใช่คู่แข่งของ JPEG

นอกจากนี้ PNG ยังมีข้อดีกว่า GIF ในอีกหลายๆ ประการ ดังนี้

1) ไฟล์แบบ PNG สามารถบีบอัดได้มากกว่า GIF ประมาณ 10-30%

2) สามารถทำสีโปร่งใสได้เหมือน GIF แถมยังปรับค่าความโปร่งใสได้อีกด้วย

3) รูปแบบ Interlaced PNG ทำงานได้เร็วกว่า Interlaced GIF

4) ไฟล์แบบ PNG มี Gamma Correction ทำให้ปรับแต่งความสว่างของภาพได้

5) PNG สามารถบันทึกไฟล์แบบ True Color หรือ 16 ล้านสีได้ ในขณะที่ GIF สนับสนุนเพียง

256 สีเท่านั้นแต่ PNG ไม่สนับสนุนภาพเคลื่อนไหว เพราะไม่สามารถเก็บภาพหลายภาพไว้ด้วยกันได้ แต่สามารถขยายขีดความสามารถ (Extensible) ได้

2.3.5 ไฟล์ภาพ TIFF (Tag Image File Format) มีนามสกุล .TIF สามารถใช้กับระบบปฏิบัติการหลายชนิด สามารถจัดเก็บเก็บภาพสี ขาวดำ และเกรย์สเกลหรือภาพหลายระดับเทา (Grayscale) ภาพส่วนใหญ่จะเป็นภาพที่เกิดจากการสแกนรูปภาพต่าง ๆ และจะจัดเก็บเป็นไฟล์นามสกุล .TIF เป็นไฟล์ที่นิยมใช้ในการโอนถ่ายข้อมูลระหว่างโปรแกรมหรือระหว่างเครือข่ายคอมพิวเตอร์ระบบต่างๆเนื่องจาก ไฟล์ภาพ TIFF นี้เป็นไฟล์ที่ยืดหยุ่นมาก สามารถนำไปใช้ได้ในโปรแกรมหลายๆชนิด ใช้การบีบอัดแบบไม่เสียรายละเอียดของภาพ

2.4 การบีบอัดข้อมูลภาพ JPEG ชนิด Sequential Baseline System

ในการบีบอัดข้อมูลภาพชนิด Sequential Baseline System มีขั้นตอนที่สำคัญได้แก่

2.4.1 ไฟล์ข้อมูลภาพต้นแบบ ดังได้กล่าวมาแล้วเกี่ยวกับ ไฟล์ภาพรูปแบบต่างๆ ในการบีบอัดข้อมูลภาพชนิด Sequential Baseline System นี้จะเลือกใช้ไฟล์ภาพประเภท BMP (.bmp) มีด้วยกัน 3 ชนิด คือ

1. 1-bit color หน่วยที่เล็กที่สุด ต่อหนึ่งหน่วยพิกเซล ซึ่งไฟล์ภาพสามารถประมวลผลหรือจัดเก็บได้ ในแต่ละบิตสีจะเป็นสีขาว หรือสีดำ อย่างใดอย่างหนึ่ง
2. 8 -bit color/grayscale หากภาพมี 8 บิต/พิกเซล ในแต่ละพิกเซล จะสามารถแสดงระดับสีได้ 256 สี หรือการแสดงระดับความอ่อนแก่ของสีเทาในภาพขาวดำนั่นเอง
3. 24 -bit color ในภาพสีแบบ RGB ที่แสดงสีได้ 8 บิต/พิกเซล ต่อหนึ่ง สี หรือรวมทั้งหมด 24 บิต ต้องนำตัวเลขของพิกเซลทั้งหมดมาคูณกัน จะได้ 256R×256G×256B เท่ากับ 16.7 ล้านสี

รายละเอียดการจัดเก็บข้อมูลของไฟล์บิตแมพมีดังต่อไปนี้

เริ่มต้น : ไฟล์ภาพ BMP ถูกคิดค้นโดยบริษัทไมโครซอฟท์

รูปแบบและการใช้ : มีการจัดเก็บเป็นแบบราสเตอร์ มีโครงสร้างของสีที่เป็นอิสระต่อกันใช้ได้โดยอุปกรณ์เดียวถูกนำมาใช้กับไมโครซอฟท์วินโดว์ 3.0 หรือสูงกว่าและโปรแกรมประยุกต์ที่ดำเนินการบนวินโดว์ 3.0 หรือ สูงกว่า เป็นลิขสิทธิ์ของบริษัทไมโครซอฟท์ภายใต้เครื่องหมายการค้าไมโครซอฟท์, วินโดว์, วินโดว์/286 และวินโดว์/386 แต่สามารถใช้ได้ฟรีโดยไม่เสียค่าใช้จ่าย

ชนิดของไฟล์ข้อมูล : เป็นไฟล์ภาพ แบบราสเตอร์สองมิติ

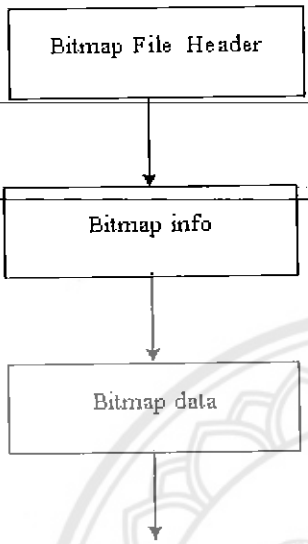
การแสดงสี : เป็นแบบโมโนโครม อินเด็กซ์สี และระบบสีRGB

กลุ่มข้อมูล : เป็นลำดับ ค่าอาร์เรย์สองมิติ

การบีบอัด : มีทั้งไม่มีการบีบอัดและมีการบีบอัดโดยใช้ Run-Length

ขนาดของภาพ : กว้าง× ยาว

ไฟล์ภาพ BMP จะแบ่งออกเป็นส่วนๆดังนี้



Signature: สองบิตแรกต้องเป็นรหัส ASCII "BM" หรือในแบบเลขฐานสิบหก "42 4D"

รูปที่ 2.5 ส่วนประกอบของบิตแมพ

ตารางที่ 2.1 Bitmap File Header

ไบท์ที่	ข้อมูล	รายละเอียด
1-2	ประเภทของไฟล์	ต้องแสดง ASCII Code เป็น "BM"
3-6	ขนาดของไฟล์	เป็นเลขจำนวนเต็ม
7-10	ไม่ถูกใช้	มีค่าเป็น 0
11-14	ออฟเซ็ทของข้อมูลบิตแมพ	ออฟเซ็ทของไฟล์เฮดเดอร์

ตารางที่ 2.2 Bitmap Information

ไบท์ที่	ข้อมูล	รายละเอียด
1-4	เลขของไบท์ในเฮดเดอร์	แสดง 40 ไบท์
5-8	ความกว้างของภาพบิตแมพ	ในหน่วยพิกเซล
9-12	ความสูงของภาพบิตแมพ	ในหน่วยพิกเซล
13-14	เลขของเพลนสี	ถูก เซทให้เป็น 1
15-16	เลขของบิตต่อพิกเซล	ต้องเป็น 1 ,4 ,8 หรือ 24
17-20	ชนิดของการบีบอัด	0 ไม่มีการบีบอัด 1 หรือ 2 เป็นการบีบอัดแบบ Run Length
21-24	ขนาดของภาพ	เป็นไบท์
25-28	ระยะคมชัดในแนวนอน	เป็น พิกเซล/เมตร
29-32	ระยะคมชัดในแนวตั้ง	เป็น พิกเซล/เมตร
37-40	อินเด็กซ์ของเลขสีที่สำคัญ	เป็น 0 เมื่อทุกสีสำคัญหมด
41	ค่าสีน้ำเงิน	เป็นไปตามค่าสี RGB
42	ค่าสีเขียว	เป็นไปตามค่าสี RGB
43	ค่าสีแดง	เป็นไปตามค่าสี RGB
44	ไม่ถูกใช้	เป็น 0

บิตแมพถูกจัดเก็บในแนวแถวและคอลัมน์ จุดเริ่มต้นของบิตแมพอยู่ที่มุมซ้าย บิตของแต่ละพิกเซลถูกรวมกันเป็นไบท์และแต่ละ เส้นถูกเก็บเป็นศูนย์ ถ้าสำคัญจะเก็บเป็นเลข 32บิต

2.4.2 ระบบสี

ปกติเมื่อพูดถึงสี มักจะนึกถึงแม่สี 3 สี แต่อย่างไรก็ตาม การใช้สีกับงานกราฟิกในคอมพิวเตอร์มีรายละเอียดหลายประการซึ่งเป็นลักษณะเฉพาะดังนั้นจึงควรทราบระบบสีของคอมพิวเตอร์ก่อน ระบบสีของคอมพิวเตอร์จะเกี่ยวข้องกับการแสดงผลแสงที่แสดงบนจอคอมพิวเตอร์ โดยมีลักษณะการแสดงผล คือ ถ้าไม่มีแสดงผลสีใดเลย บนจอภาพจะแสดงเป็น "สีดำ" หากสีทุกสีแสดงผลพร้อมกัน จะเห็นสีบนจอภาพเป็น "สีขาว" ส่วนสีอื่นๆ เกิดจากการแสดงสีหลายๆ สี แต่มีค่าแตกต่างกัน การแสดงผลลักษณะนี้ เรียกว่า การแสดงสี

1. ระบบสี RGB (Red, Green, Blue) เป็นระบบสีของแสง ซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึม จะเกิดแถบสีที่เรียกว่า สีรุ้ง (Spectrum) ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ แดง เหลือง เขียว น้ำเงิน คราม ม่วง ซึ่งเป็นพลังงานอยู่ในรูปของรังสี ที่มีช่วงคลื่นที่สายตา สามารถมองเห็นได้ แสงสีม่วงมีความถี่คลื่นสูงที่สุด คลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วง เรียกว่า อุลตราไวโอ

เลต (Ultra Violet) และคลื่นแสงสีแดง มีความถี่คลื่นต่ำที่สุดคลื่นแสง ที่ต่ำกว่าแสงสีแดงเรียกว่า อินฟราเรด (InfraRed) คลื่นแสงที่มีความถี่สูงกว่าสีม่วงและต่ำ กว่าสีแดงนั้น สายตาของมนุษย์ไม่สามารถรับได้ และเมื่อศึกษาดูแล้วแสงสีทั้งหมดเกิดจาก แสงสี 3 สี คือ สีแดง (Red) สีน้ำเงิน (Blue) และสีเขียว (Green)ทั้งสามสีถือเป็นแม่สีของแสง เมื่อนำมาฉายรวมกันจะทำให้เกิดสีใหม่ อีก 3 สี คือ สีแดงมาเจนต้า (Magenta) สีฟ้าไซ-แอน(Cyan) และสีเหลือง ซึ่งถ้าฉายแสงสีทั้งหมดรวมกันจะได้แสงสีขาว จากคุณสมบัติของแสงนี้เราได้นำมาใช้ประโยชน์ทั่วไป ในการฉายภาพยนตร์ การบันทึกภาพวิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์และการจัดแสงสีในการแสดง เป็นต้น

2.ระบบสี YUV ระบบสีแบบ YUV ใช้สำหรับโทรทัศน์แบบ PAL และ SECAM ซึ่งยังมีใช้อยู่ในหลาย ๆ ประเทศโดย Y คือค่าความสว่างของภาพ ส่วนสัญญาณ U และ V เป็นสัญญาณที่เก็บค่าสีของภาพ ต่อมาได้มีระบบ YIQ มาใช้แทน เนื่องจากพบว่าสัญญาณ I และ Q สามารถลดแบนวิธ (Bandwidth) ได้มากกว่าสัญญาณ U และ V ในขณะที่ได้ภาพที่มีคุณภาพเท่ากัน

2.4.3 การซับแซมปลิ้งสี (Color Subsampling)

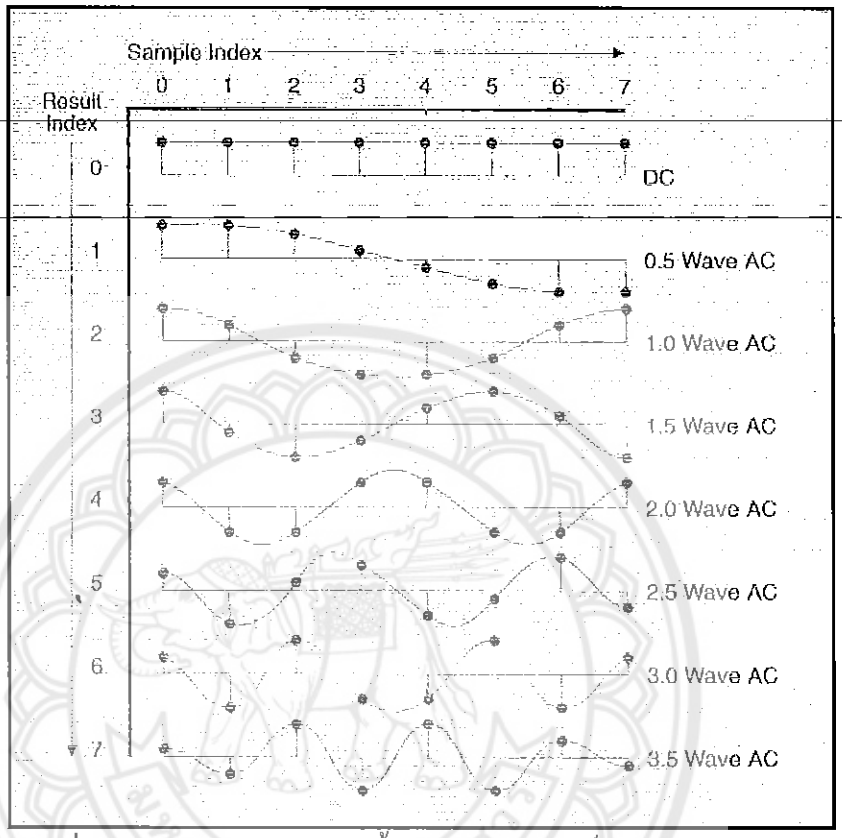
การซับแซมปลิ้งเป็นเทคนิคการลดจำนวนข้อมูล สำหรับชนิดภาพจะใช้คุณสมบัติของระบบการมองเห็นของมนุษย์มาใช้ประโยชน์เพื่อลดจำนวนข้อมูล นั่นคือ ตาของมนุษย์จะมีความไวต่อการเปลี่ยนแปลงของแสงมากกว่าความเปลี่ยนแปลงของสี ซึ่งถูกนำมาใช้ในมาตรฐานการออกอากาศสำหรับโทรทัศน์ (NTSC, PAL และ SECAM) CCIR-601 ได้กำหนดวิธีการซับแซมปลิ้งสีสำหรับภาพดิจิตอล คือ จะมีตัวเลข 3 ตัวแล้วคั่นกลางด้วย (:) ตัวเลขตัวแรก แทน Y (Luminance) ตัวที่สองแทน Cr (Red – Y) และตัวที่สามแทนด้วย Cb (Blue –Y) อัตราส่วนการซับแซมปลิ้งที่ใช้กันมากคือ 4 :1 : 1 และ 4 : 2 : 2 ซึ่ง 4 : 1 : 1 หมายถึง สำหรับทุก 4 พิกเซลของตัวอย่างลูมิแนนซ์จะมีโครมิแนนซ์อย่างละ 1 พิกเซล และ 4 : 2 : 2 หมายถึงสำหรับทุก 4 พิกเซลของตัวอย่างลูมิแนนซ์จะมีโครมิแนนซ์อย่างละ 2 พิกเซล

2.4.4 การแปลง DCT (Discrete Cosine Transform)

การแปลง DCT 1 มิติ

DCT เป็นการคำนวณทางคณิตศาสตร์ที่คำนวณองค์ประกอบความถี่ของสัญญาณตัวอย่างที่อัตราการแซมปลิ้ง DCT จะเป็นประโยชน์กับข้อมูลที่เป็นเลขที่มีการจำกัดเขตข้อมูล DCT 1 มิติจะทำการแปลงอาร์เรย์ของขนาดของสัญญาณที่ระยะทางใดๆไปเป็นอีกอาร์เรย์ของขนาดขององค์ประกอบความถี่จากสัญญาณเริ่มต้น อาร์เรย์ผลลัพธ์ที่ได้จะมีขนาดเท่ากับอาร์เรย์เริ่มต้น อาร์เรย์ผลลัพธ์ตัวแรกจะถูกเรียกว่า สัมประสิทธิ์ DC (DC Coefficient) อาร์เรย์ผลลัพธ์ที่เหลือ เรียกว่า สัมประสิทธิ์ AC (AC Coefficient) ความถี่ประกอบด้วยชุดของตัวอย่างที่แต่ละความถี่ถูกคำนวณโดยนำค่าน้ำหนักเฉลี่ยของ

ชุดทั้งหมด ค่าน้ำหนักแต่ละอินพุตคำนวณได้จากการคูณอาร์เรย์อินเด็กซ์ปัจจุบันด้วยไพ (PI) และอินเด็กซ์ของตัวอย่างอินพุต จะได้อนุกรมของค่าสัมประสิทธิ์ เป็นโคซายน์เวฟ ที่ซึ่งความถี่เป็นสัดส่วนกับอาร์เรย์อินเด็กซ์ ค่าน้ำหนักสูงสุดจะมีค่า เท่ากับ 1 เท่านั้น



รูปที่ 2.6 ความสัมพันธ์ของค่าน้ำหนักสัมประสิทธิ์ DCT 1 มิติ

จากรูปที่ 2.6 ความสัมพันธ์ของค่าน้ำหนักสัมประสิทธิ์ DCT 1 มิติ มี 8 อาร์เรย์ตัวอย่างมีความถี่สูงสุดที่ 3.5 ไซเคิล ในตารางที่ 2.3 จะแสดงค่าสัมประสิทธิ์ที่ใช้สำหรับการคำนวณน้ำหนักเฉลี่ยของแต่ละองค์ประกอบความถี่ ซึ่งคำนวณจากสมการดังต่อไปนี้

$$Coeff(k,m) = C(k)\cos\left[\frac{(2m+1)k\pi}{2N}\right] \qquad (2.1)$$

เมื่อ $C(k) = \frac{1}{\sqrt{2}}$ เมื่อ $k = 0$

$C(k) = 1$ เมื่อ $k \neq 0$

k คือ อินเด็กซ์อาร์เรย์ผลลัพธ์

m คือ อินเด็กซ์อาร์เรย์ตัวอย่าง

N คือ ขนาดอาร์เรย์ตัวอย่าง

จากสมการสามารถกล่าวได้ว่า เมื่ออินเด็กซ์อาร์เรย์ผลลัพธ์เป็น 0 ตัวเลข ในเศษส่วนจะเป็น 0 ค่าโคซายน์ของ 0 คือ 1 ดังนั้นขนาดแฟกเตอร์จะเหมือนกันทั้งหมดในกรณีนี้ขยายสมการหาค่าน้ำหนักเฉลี่ย ดังสมการ 2.2 ซึ่งมีขนาดอาร์เรย์ตัวอย่างเท่ากับ N

$$X(k) = \sqrt{\frac{2}{N}} C(k) \sum_{m=0}^{N-1} x(m) \cos\left[\frac{(2m+1)k\pi}{2N}\right] \qquad (2.2)$$

เมื่อ $C(k) = \frac{1}{\sqrt{2}}$ เมื่อ $k = 0$

$C(k) = 1$ เมื่อ $k \neq 0$

X () คือ อาร์เรย์ผลลัพธ์

x () คือ อาร์เรย์ตัวอย่าง

N คือ ขนาดอาร์เรย์

k คือ อินเด็กซ์อาร์เรย์ผลลัพธ์

m คือ อินเด็กซ์อาร์เรย์ตัวอย่าง

DCT ได้ถูกคำนวณไว้เป็นเมตริกสัมประสิทธิ์ ดังแสดงในตารางที่ 2.3 สามารถคำนวณครั้งเดียวแล้วได้ค่าทั้งหมดดังรูปที่2.7 ชุดตัวอย่างที่จะแปลงให้อยู่ในรูปของสัมประสิทธิ์ความถี่ สามารถเลือกควอนไทซ์เพื่อลดขนาดของชุดข้อมูลและรักษาข้อมูลส่วนสำคัญ ถ้าสัมประสิทธิ์ความถี่ถูกควอนไทซ์ ข้อมูลเริ่มต้นจะไม่สามารถเก็บได้สมบูรณ์ จะอยู่ในรูปของผลรวมของสัมประสิทธิ์ความถี่และน้ำหนักสัมประสิทธิ์ สำหรับสมการการแปลงกลับของชุดตัวอย่างคือ IDCT สำหรับ DCT 1 มิติ ดังแสดงในสมการ 2.3

$$x(m) = \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} X(k) C(k) \cos\left[\frac{(2m+1)k\pi}{2N}\right] \qquad (2.3)$$

เมื่อ $C(k) = \frac{1}{\sqrt{2}}$ เมื่อ $k = 0$

$C(k) = 1$ เมื่อ $k \neq 0$

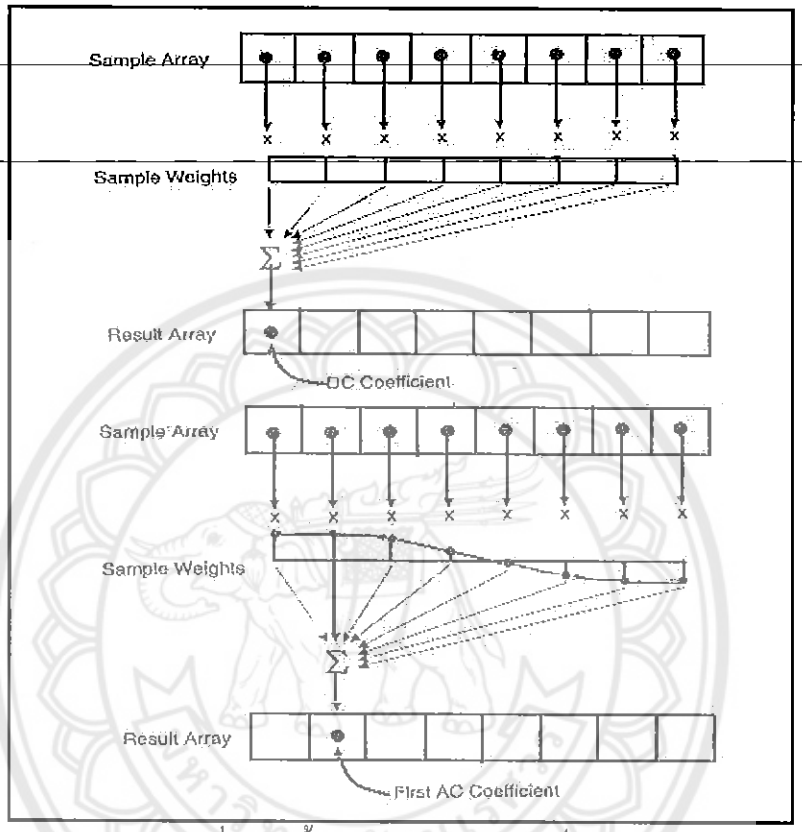
X () คือ อาร์เรย์ผลลัพธ์

x () คือ อาร์เรย์ตัวอย่าง

N คือ ขนาดอาร์เรย์

k คือ อินเด็กซ์อาร์เรย์ผลลัพธ์

m คือ อินเด็กซ์อาร์เรย์ตัวอย่าง



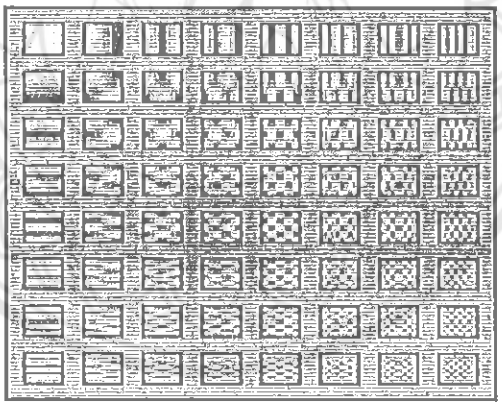
รูปที่ 2.7 น้ำหนักของสัมประสิทธิ์ DCT

ตารางที่ 2.3 น้ำหนักสัมประสิทธิ์ DCT 1 มิติ

	0	1	2	3	4	5	6	7
0	+0.707	+0.707	+0.707	+0.707	+0.707	+0.707	+0.707	+0.707
1	+0.981	+0.831	+0.556	+0.195	-0.195	-0.556	-0.831	-0.981
2	+0.924	+0.383	-0.383	-0.924	-0.924	-0.383	+0.383	+0.924
3	+0.831	-0.195	-0.981	-0.556	+0.556	+0.981	+0.195	-0.831
4	+0.707	-0.707	-0.707	+0.707	+0.707	-0.707	-0.707	+0.707
5	+0.556	-0.981	+0.195	+0.831	-0.831	-0.195	+0.981	-0.556
6	+0.383	-0.924	+0.924	-0.383	-0.383	+0.924	-0.924	+0.383
7	+0.195	-0.556	+0.831	-0.981	+0.981	-0.831	+0.556	-0.195

การแปลง DCT 2 มิติ

เมื่อเรารู้หลักการของ DCT 1 มิติ แล้วเราก็จะสามารถพิจารณา DCT 2 มิติได้ ชุดข้อมูลจะเป็นอาร์เรย์ของพิกเซล เมื่อพิจารณาอาร์เรย์ 2 มิติของพิกเซลขนาด 8x8 จะสามารถแบ่งเป็น 8 แถว หรือ 8 หลักซึ่งแต่ละแถวหรือหลักมี 8 พิกเซล เมื่อแปลง DCT 1 มิติของแต่ละแถวจะได้ผลลัพธ์จะเป็น 8 แถวของสัมประสิทธิ์DCT ถ้าทั้ง 8 แถวของสัมประสิทธิ์ความถี่ถูกเรียงเป็น 8 คอลัมน์ คอลัมน์แรกจะเก็บค่าสัมประสิทธิ์ DC คอลัมน์ที่สองจะเก็บค่าแรกของสัมประสิทธิ์ AC จากแต่ละแถว ซึ่งยังไม่สามารถหาค่าสัมประสิทธิ์ DCT ของหลักได้ ดังนั้น DCT 1 มิติ สามารถถูกนำมาใช้อีกครั้งเป็นคอลัมน์ แต่ละส่วนของผลลัพธ์อาร์เรย์ 2 มิติขององค์ประกอบความถี่จะถูกแสดงเป็นองค์ประกอบความถี่ 2 มิติ ส่วนในมุมบนขวาสุดคือ สัมประสิทธิ์ DC สำหรับแต่ละสัมประสิทธิ์ความถี่ เมื่อใช้ IDCT 2 มิติ จะได้รูปแบบที่แตกต่างของพื้นที่ที่สว่างและพื้นที่มืด พร้อมกับสัดส่วนความเข้มแสงถูกนิยามโดยขนาดของสัมประสิทธิ์ ดังรูปที่ 2.8 แสดงรูปแบบที่ได้ในแต่ละองค์ประกอบสัมประสิทธิ์ความถี่ ใน ผลลัพธ์ DCT ขนาด 8x8 พิกเซล บล็อกบนขวาเป็นสีเทา แสดงค่าเฉลี่ยสำหรับบล็อกขนาด 8x8 พิกเซล บล็อกล่างสุดด้านขวาดูคล้ายกระดานหมากรุก เพราะแสดงค่าองค์ประกอบความถี่สูงสุดในแนวตั้งและแนวนอน บล็อกบนสุดด้านขวาและล่างสุดด้านซ้ายแสดงแถบแนวตั้งและแนวนอนตามลำดับ เพราะแสดง ความถี่สูงสุดในแนวตั้งและ ความถี่ที่เป็น 0 ในแนวนอน สำหรับ สมการDCT 2 มิติแสดงในสมการ 2.4



รูปที่ 2. 8 รูปแบบความหนาแน่นของตัวอย่างของ DCT 2 มิติ

Forward DCT

$$F[u,v] = \tfrac{1}{N} CuCv \sum_{m=0}^{N-1}\sum_{n=0}^{N-1} f[m,n]\cos\left[\frac{(2m+1)u\pi}{2N}\right]\cos\left[\frac{(2n+1)v\pi}{2N}\right] \qquad (2.4)$$

Inverse DCT

$$f(m,n) = \sum_{U=0}^{N-1}\sum_{V=0}^{N-1} CuCvF[u,v]\cos\left[\frac{(2m+1)u\pi}{2N}\right]\cos\left[\frac{(2n+1)v\pi}{2N}\right] \qquad (2.5)$$

เมื่อ Cu และ Cv = 1 เมื่อ u,v = 0,0 (องค์ประกอบ DC)

Cu และ Cv = 2 เมื่อ u,v ≠ 0,0

$F[u,v]$ คือ สัมประสิทธิ์ DCT

$f(m,n)$ คือ ค่าระดับสีใน พิกเซล

จากสมการเป็นการแสดงผลรวมในรูปโคซายน์ สำหรับแนวตั้งและแนวนอนสำหรับแต่ละส่วนใน อาร์เรย์ผลลัพธ์ สมการเป็นการรวมกันของสมการ DCT 2 มิติในแนวตั้งและแนวนอน

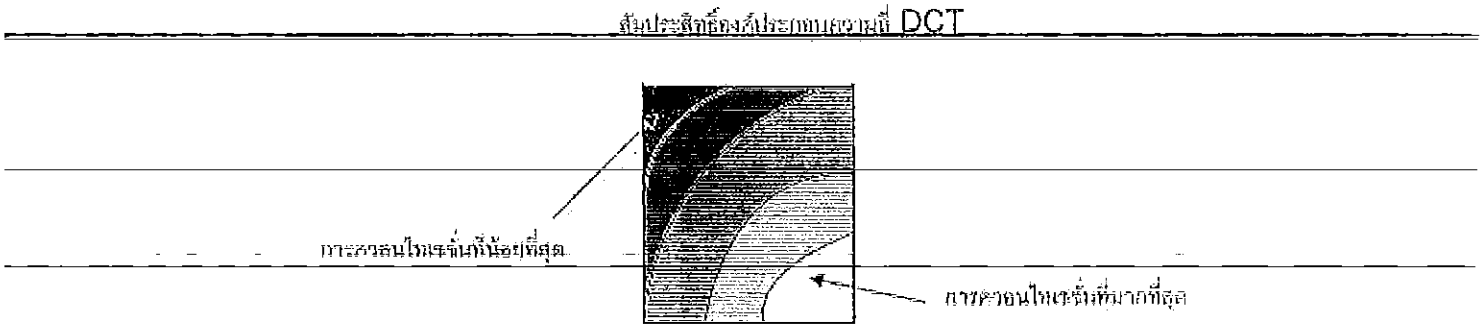
ข้อดีของ DCT คือสามารถเลือกการควอนไซท์ได้หลายองค์ประกอบความถี่ เพื่อใช้คำนวณลดค่าความสำคัญของข้อมูล ข้อเสียคือหลังจากสร้างภาพขึ้นใหม่ จะมองเห็นขอบบล็อกของพิกเซล ดังรูปที่ 2.9 แสดงภาพต้นแบบและภาพที่สร้างขึ้นใหม่ โดยใช้ DCT ขนาด 8x8 ค่า ในการแปลงและการสร้างภาพใหม่

รูปที่ 2.9 ภาพต้นแบบและภาพใหม่ที่ผ่าน DCT และการควอนไทซ์

ในทางปฏิบัติ ระดับของการควอนไซท์สามารถใช้ได้หลายองค์ประกอบความถี่ สำหรับภาพสีจะทำการควอนไทซ์องค์ประกอบสีมากกว่าองค์ประกอบแสง ในการคำนวณ DCT สำหรับบล็อกภาพจะใช้ตัวเลขจำนวนเต็มมากกว่าเลขทศนิยมเพราะว่าเลขทศนิยมใช้เวลาในการคำนวณมาก ดังรูปที่2.10 แสดงระดับการควอนไทซ์ขององค์ประกอบความถี่ต่างๆจากความถี่มากสุดไปความถี่ต่ำสุด

ตัวอย่างเช่นเมื่อพิจารณาบล็อกขนาด 8×8 พิกเซลของ 8 บิตลูมิแนนซ์ ที่ได้มาจากไฟล์ภาพแสดงค่าดังตารางที่ 2.4 หลังจากผ่านสมการ DCT 2 มิติ จะได้ผลลัพธ์ของค่าสัมประสิทธิ์ DCT ที่แสดงให้ตารางที่ 2.5 ซึ่งส่วนใหญ่จะมีค่าสัมประสิทธิ์ DCT ที่ต่ำ นำค่าค่าสัมประสิทธิ์ DCT หารด้วยค่าสัมประสิทธิ์ควอนไทซ์ที่ตรงกันที่แสดงในตารางที่ 2.6 หลังจากการควอนไทซ์จะได้อาร์เรย์สัมประสิทธิ์ DCT ที่ถูกควอนไทซ์แล้วดังตารางที่ 2.7 ทำให้ข้อมูลส่วนใหญ่มีค่าเป็นศูนย์เพื่อนำไปใช้ในการเข้ารหัส Run Length เมื่อบล็อกพิกเซลทำการสร้างข้อมูลกลับโดยนำสัมประสิทธิ์ DCT ที่ถูกควอนไทซ์แล้วมาคูณกับ

สัมประสิทธิ์ควอนไทซ์ และนำมาแปลง IDCT จะได้ผลลัพธ์แสดงในตารางที่ 2.8 และหาค่าผิดพลาดของการบีบอัดได้ดังตารางที่ 2.9



รูปที่ 2.10 ความสัมพันธ์ของระดับการควอนไทซ์กับขนาดของค่าสัมประสิทธิ์ DCT

ตารางที่ 2.4 ค่าพิกเซลเริ่มต้น

97	97	97	102	102	102	102	102
88	88	97	97	97	97	102	97
88	88	97	88	88	97	88	88
88	88	88	88	88	88	88	88
80	92	88	88	88	88	88	88
80	88	83	88	82	88	88	88
80	83	80	86	88	86	86	86
80	80	83	83	80	80	83	88

ตารางที่ 2.5 ค่าสัมประสิทธิ์ความถี่ DCT

+2853	-47	-20	-16	-14	0	0	-10
+166	-4	-14	+10	-7	+11	+14	+12
+43	-23	+1	0	+21	+12	0	+6
+45	-13	-3	+2	+2	-7	-29	-16
+8	+12	+12	-6	+12	-7	-8	+1
+2	+11	+1	+21	+12	-4	+1	-4
-5	+15	+10	-3	-11	-1	+18	-13
+4	+7	-11	+3	-9	-14	+7	0

ตารางที่ 2.6 ค่าสัมประสิทธิ์การควอนไทซ์

8	8	8	8	8	8	8	8
8	8	8	8	8	8	8	8
8	8	8	8	256	256	256	256
8	8	8	256	256	256	256	256
8	8	256	256	256	256	256	256
8	8	256	256	256	256	256	256
8	8	256	256	256	256	256	256
8	8	256	256	256	256	256	256

ตารางที่ 2.7 ค่าสัมประสิทธิ์DCT ที่ถูกควอนไทซ์แล้ว

+2853	-5	-2	-2	-1	0	0	-1
+20	0	-1	+1	0	+1	+1	+1
+5	-2	0	0	0	0	0	0
+5	-1	0	0	0	0	0	0
+1	+1	0	0	0	0	0	0
0	+1	0	0	0	0	0	0
0	+1	0	0	0	0	0	0
0	0	0	0	0	0	0	0

ตารางที่ 2.8 ค่าพิกเซลใหม่ที่ได้จาก IDCT

95	96	101	100	100	103	102	100
88	89	95	95	96	99	99	98
88	89	92	91	90	91	90	88
86	88	89	89	87	88	87	87
84	87	87	88	87	88	88	88
80	85	84	86	85	86	87	88
79	84	82	85	83	85	86	88
78	83	80	83	81	82	83	85

ตารางที่ 2.9 ค่าความผิดพลาดของข้อมูล

+2	+1	-4	+2	+2	-1	0	+2
0	-1	+2	+2	+1	-2	+3	-1
0	-1	+5	-3	-2	+6	-2	0
+2	0	-1	-1	+1	0	+1	+1
-4	+5	+1	0	+1	0	0	0
0	+3	-1	+2	-3	+2	+1	0
+1	-1	-2	+1	+5	+1	0	-2
+2	-3	+3	0	-1	-2	0	+3

2.4.5 การเข้ารหัสฮัฟแมน (Huffman encoding)

การเข้ารหัสฮัฟแมน เป็นการนำเอนโทรปีมาคิดเชิงสถิติ โดยจะใช้ความยาวของรหัสแทนข้อมูลภาพใช้รหัสที่สั้นที่สุดแทนข้อมูลที่มีโอกาสเกิดขึ้นมากที่สุดและใช้รหัสที่ยาวที่สุดแทนข้อมูลที่เกิดขึ้นน้อยที่สุด ผลลัพธ์ที่ได้คือ ข้อมูลส่วนใหญ่จะมีรหัสสั้น ดังนั้นเป็นการลดจำนวนของข้อมูล รหัสฮัฟแมนมีลักษณะพิเศษกว่ารหัสที่ใช้เทคนิคความยาวของรหัสอื่นๆ คือ

1. ไม่มีรหัสที่เหมือนกัน
2. รหัสมีจุดเริ่มต้นและจุดจบด้วยตัวมันเอง
3. มีการเรียงจำนวนของสัญลักษณ์จากโอกาสที่เกิดขึ้นจากมากไปน้อย ความยาวของค่ารหัสจะเกี่ยวข้องกับสัญลักษณ์ จะน้อยกว่าหรือเท่ากับความยาวของรหัสที่อยู่ใกล้เคียงกัน

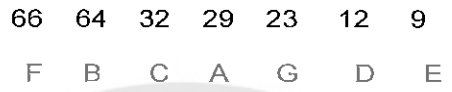
ในการสร้างตารางรหัสฮัฟแมน มีลำดับขั้นตอนดังนี้

1. นำข้อมูลมาเรียงค่าความถี่จากมากไปหาน้อย
2. เลือกความถี่ที่น้อยที่สุด 2 ค่ามีบวกกัน เปรียบเทียบค่าความถี่ที่ได้แล้วนำมาเรียงค่าความถี่จากมากไปหาน้อยใหม่

ตัวอย่างการสร้างรหัสฮัฟแมน

ตัวอย่างข้อมูล ซึ่งมีความถี่ดังนี้ คือ A : 29 , B : 64 ,C : 32 ,D :12, E : 9,F : 66,G : 23

ขั้นแรก เรียงตัวอักษรจากความถี่สูงสุดไปความถี่ต่ำสุด ได้ดังนี้

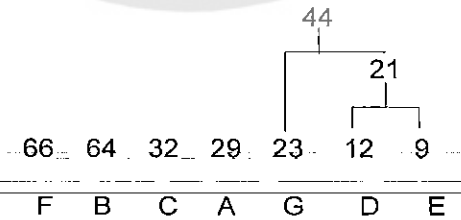


รูปที่ 2.11 เรียงข้อมูลตามลำดับความถี่จากมากไปน้อย

ขั้นที่สอง เลือกความถี่ที่น้อยที่สุดสองตัว บวกกัน ดังตัวอย่าง D และ E รวมกันได้ 21

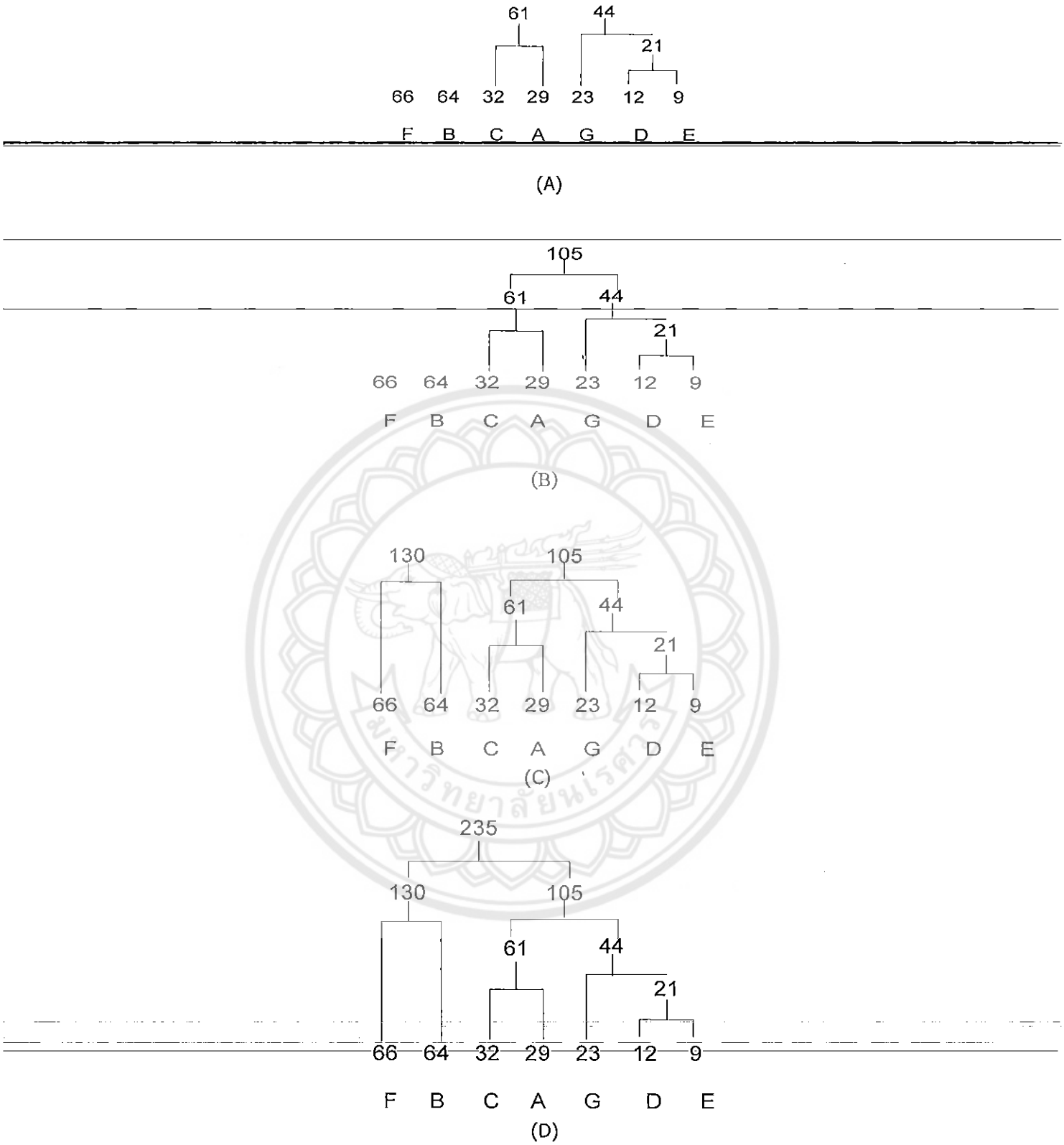
รูปที่ 2.12 นำความถี่น้อยที่สุด 2 ค่ามารวมกัน

เปรียบเทียบค่าความถี่ที่ได้จะเห็นว่า 21 มีค่าน้อยที่สุด นำค่า 21 (D+E) มีรวมกับ 23 (G) จะได้ 44



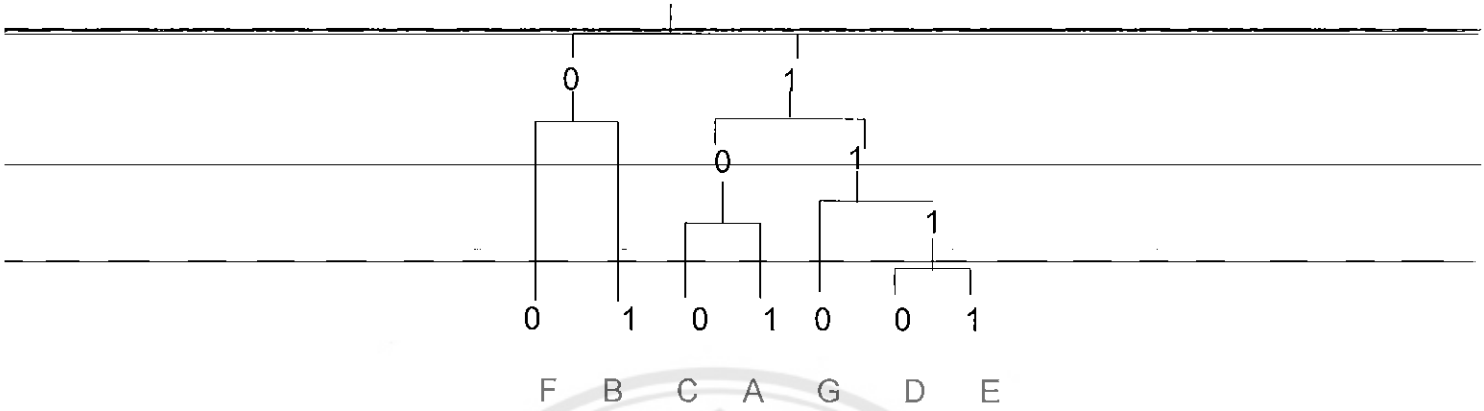
รูปที่ 2. 13 รวมค่าความถี่ที่ต่ำที่สุดในลำดับต่อมา

เปรียบเทียบค่าความที่ได้ จะได้ว่า 44 (D+E+G) มีค่ามากกว่า 29 และ 32 เรียงลำดับความที่ใหม่จะได้ว่า 66 64 44 32 29 แล้วนำค่าความที่น้อยที่สุดมารวมกันจะได้ 61 (C+A) และทำให้ทำนองเดียวกันไปจนกว่าจะหมดจำนวนข้อมูล จะได้ดังรูปที่ 2.14



รูปที่ 2.14 โครงสร้างทรี การหารหัสฮัฟแมนตามลำดับ

เมื่อหมดจำนวนข้อมูลแล้ว จะได้ โครงสร้างทรี ดังรูปที่ 2. 14 (D) แทนตัวเลขในแต่ละกิ่งของทรีด้วยเลข 0 และ 1 ที่ไม่ซ้ำกัน โดยเริ่มจากยอดเป็น 0 ซึ่งจะได้ดังรูปที่ 2.15



รูปที่ 2.15 รหัสฮัฟแมนที่ได้จาก ทรี

จะได้รหัสฮัฟแมนดังนี้

F: 00

B: 01

C: 100

A: 101

G: 110

D: 1110

E: 1111

จะสังเกตได้ว่ารหัสฮัฟแมนที่ได้ไม่มีข้อมูลตัวไหนมีรหัสเหมือนกันและข้อมูลที่มีความถี่มากสุดมีรหัสฮัฟแมนที่มีความยาวสั้นที่สุดและมีค่าน้อยที่สุด ซึ่งใช้แทนข้อมูลที่เกิดขึ้นมากที่สุด

บทที่ 3
การออกแบบและเขียนโปรแกรม

ในการเขียนโปรแกรมภาษาต่างๆจะต้องเครื่องมือที่ใช้กับการเขียนโปรแกรมนั้นๆ ซึ่งในโครงงานนี้ได้เลือกภาษาจาวา เพราะมีข้อดีที่สำคัญคือ การที่เขียนโปรแกรมครั้งเดียว แต่สามารถรันได้บนเครื่องคอมพิวเตอร์หลายรูปแบบโดยไม่ต้องมาคอมไพล์หรือเขียนโปรแกรมใหม่ ทำให้ช่วยประหยัดเวลาการพัฒนาลงไปได้มาก ช่วยในการจัดการกับหน่วยความจำได้ดี รวมทั้งมีเครื่องมือที่ช่วยในการเขียนโปรแกรมและภาษาจาวาสนับสนุนแนวความคิดของการเขียนโปรแกรมเชิงวัตถุ ก่อนการเขียนโปรแกรมจึงมีการเตรียมความพร้อมในกับเครื่องคอมพิวเตอร์ดังนี้

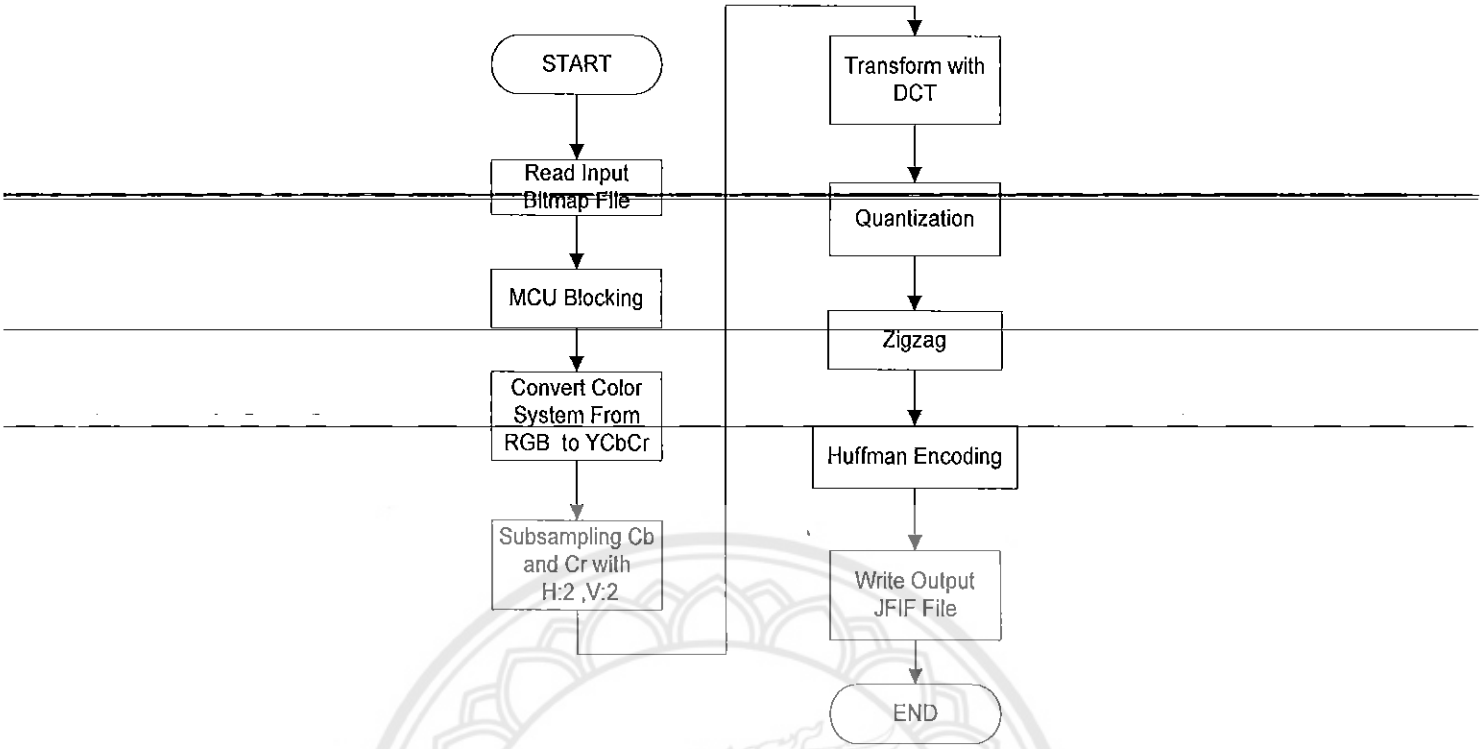
ความต้องการของระบบ

1. ระบบปฏิบัติการ : Windows 95/98 หรือ Professional/2000 Server/2000 รวมทั้งระบบปฏิบัติการลีนุกซ์ หรืออื่นๆ
2. ซีพียู: ควรเป็น Pentium ความเร็วอย่างน้อย 166 MHz
3. แรม: ควรใช้หน่วยความจำอย่างน้อย 128 MB เพราะโปรแกรมที่ออกแบบค่อนข้างใหญ่
4. ฮาร์ดดิสก์ : ควรเหลือเนื้อที่ว่างในฮาร์ดดิสก์อย่างน้อย 150 MB หรือมากกว่า

เครื่องมือที่ใช้

1. การดาวน์โหลดไฟล์เวอร์ชันล่าสุดได้จาก http://java.sun.com/j2sc/1.5.0/download.jsp
2. สำหรับการติดตั้งสามารถศึกษาได้จากหนังสือ คู่มือการเขียนโปรแกรมภาษาจาวา
3. โครงงานนี้ได้ใช้ Eclipse เป็นเครื่องมือช่วยในการเขียนโปรแกรม ช่วยทำให้การแก้ไขโปรแกรมได้รวดเร็วขึ้น

การออกแบบโปรแกรมการบีบอัดข้อมูลภาพ JPEG ชนิด Sequential Baseline System เป็นไปตามบล็อคไดอะแกรมดังรูปที่ 3.1 และขั้นตอนการบีบอัดข้อมูลภาพ JPEG ในแต่ละส่วนมีรายละเอียดดังต่อไปนี้



รูปที่ 3.1 บล็อกไดอะแกรมของโปรแกรม

3.1 การเตรียมข้อมูลภาพสำหรับการบีบอัดข้อมูลภาพนิ่งแบบ JPEG

ภาพต้นแบบที่ใช้จะถูกเก็บไว้ในรูปแบบไฟล์บิตแมพ ในการบีบอัดข้อมูลภาพแบบ JPEG ชนิด Sequential Baseline System นี้ ต้องมีการจัดการให้เหมาะสมก่อนคือ

3.1.1 การแบ่งเป็น MCU (Minimum Coded Unit) ในการบีบอัดข้อมูลแบบ JPEG นั้นจะแบ่งการบีบอัดภาพเป็นหน่วยย่อยๆ เรียกว่า MCU โดยมีขนาดหน่วยละ 16×16 พิกเซล ขึ้นอยู่กับอัตราส่วนการซับแซมปลิ้งที่ใช้ดังนั้นภาพที่มีขนาดไม่สามารถแบ่งให้เป็นหน่วย MCU ได้ลงตัวพอดี จำเป็นต้องมีการเพิ่มขนาดของภาพเพื่อให้สามารถแบ่งเป็นหน่วย MCU ได้ครอบคลุมภาพทั้งหมดโดยไม่มีส่วนใดส่วนหนึ่งของภาพขาดหายไป การเพิ่มขนาดของภาพนี้อาจต้องเพิ่มจำนวนจุดพิกเซลทางด้านซ้ายของแต่ละแถวหรือเพิ่มจำนวนแถวของจุดพิกเซลทางด้านล่างของภาพหรือต้องเพิ่มทั้งคู่ สีของจุดพิกเซลในส่วนที่เพิ่มขึ้นนั้นควรมีค่าเท่ากับสีของจุดพิกเซลในบริเวณขอบด้านที่จุดพิกเซลเพิ่มนั้นติดต่อด้วย เนื่องจากถ้าสีของจุดพิกเซลที่เพิ่มขึ้นมีค่าแตกต่างกันมากกับสีของจุดพิกเซลของภาพบริเวณนั้นจะทำให้ข้อมูลของ MCU ในภาพบริเวณนั้นมีการกระจายตัวอยู่ในระดับความถี่ที่สูง ซึ่งในการบีบอัดข้อมูลภาพจะมีการลดความสำคัญของข้อมูลในระดับความถี่สูง ในขั้นตอนการควอนไทเซชั่นซึ่งจะมีผลทำให้ข้อมูลภาพบริเวณนั้นเมื่อถูกขยายคืน (Decompress) มีความผิดพลาดมาก ถึงแม้ว่าในการขยายภาพจุดพิกเซลที่เพิ่มขึ้นจะถูกตัดทิ้งโดยอุปกรณ์หรือโปรแกรมที่ทำหน้าที่ขยายภาพอัตโนมัติก็ตาม

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

3.1.2 การแปลงระบบสี RGB ให้เป็น YCbCr สาเหตุที่เลือกใช้ระบบสี YCbCr เพราะในระบบสี YCbCr นั้นได้มีการแยกออกเป็นค่าลูมิแนนซ์หรือแสง (Luminance) และค่าโครมิแนนซ์หรือสี (Chrominance) อย่างชัดเจน ในระบบการมองเห็นของมนุษย์นั้น ตามนุษย์จะมีความไวต่อการเปลี่ยนแปลงของแสงมากกว่าความเปลี่ยนแปลงของสี ดังนั้นระบบสี YCbCr จึงมีประโยชน์ต่อการลดจำนวนของสีต่อจำนวนจุดพิกเซลของสีโดยการลดจำนวนสีโครมิแนนซ์ด้วยวิธีการซับแซมปลิ้ง ซึ่งจะไม่ทำให้คุณภาพของภาพเปลี่ยนไปมากนัก ซึ่งมีสมการดังนี้

$$\begin{aligned} Y &= 0.299R + 0.587G + 0.114B \\ Cb &= -0.1687R - 0.3313G + 0.5B \\ Cr &= 0.5R - 0.1487G - 0.0813B \end{aligned} \quad (3.1)$$

โดยที่ $0 \le Y \le 255, -128 \le Cb \le 128, -128 \le Cr \le 128$

3.1.3 การซับแซมปลิ้ง (Subsampling) ข้อมูลในอัตรา Y: Cb: Cr เท่ากับ 4:1:1 การซับแซมปลิ้งเป็นการลดจำนวนข้อมูลด้วยการแทนกลุ่มของข้อมูลด้วยค่าเฉลี่ยของกลุ่มข้อมูลนั้น การซับแซมปลิ้งสามารถทำได้ 2 ทิศทาง คือ แนวตั้ง (Vertical:V) และแนวนอน (Horizontal :H) เช่น การซับแซมปลิ้งด้วยอัตราส่วน H:2 V:2 คือการจับกลุ่มข้อมูลในแนวนอน 2 ตัวและในแนวตั้ง 2 ตัวจะได้กลุ่มข้อมูลกลุ่มละ 4 ตัว โดยแต่ละกลุ่มจะถูกแทนที่ด้วยข้อมูล 1 ตัว ซึ่งหามาจากค่าเฉลี่ยของข้อมูลในกลุ่มที่ถูกแทนที่นั่นเองดังแสดงในรูปที่ 3.2 วิธีนี้ทำให้จำนวนข้อมูลจะลดลงเหลือเพียงหนึ่งในสี่

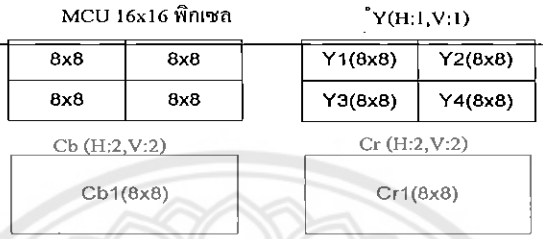
แทนข้อมูล 1 ตัว

1	2	3	4	5	6	7	8
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32

(1+2+9+10)/4	(3+4+11+12)/4	(5+6+13+14)/4	(7+8+15+16)/4
(17+18+25+26)/4	(19+20+27+28)/4	(21+22+29+30)/4	(23-24+31+32)/4

รูปที่ 3.2 ตัวอย่างการซับแซมปลิ้งด้วยอัตราส่วน H: 2 และ V: 2

ด้วยวิธีดังกล่าว JPEG จะใช้การซับแซมปลิ้งค่า Cb และ Cr ด้วยอัตราส่วน H:1, V:1 หรือ H:2 ,V:1 หรือ H:2 , V:2 โดยคงจำนวนค่า Y ไว้ดังเดิม (H:1,V:1) ในโครงงานนี้ได้เลือกใช้อัตราส่วนของการซับแซมปลิ้งค่า Y : Cb : Cr เท่ากับ 4 : 1 : 1 คือใช้อัตราส่วน H : 2 , V : 2 ในการซับแซมปลิ้งค่า Cb และ Cr ซึ่งมีผลทำให้หน่วย MCU หลังจากการ ซับแซมปลิ้งจะประกอบด้วยบล็อกข้อมูลของค่า Y จำนวน 4 บล็อก Cb จำนวน 1 บล็อก และ Cr จำนวน 1 บล็อก (Y : Cb : Cr = 4 : 1 :1) ขนาดบล็อกละ 8×8 ค่า ดังแสดงในรูปที่ 3.3 และมีการเรียงลำดับของบล็อกภายใน MCU คือ Y1, Y2, Y3,Y4, Cb1, Cr1



รูปที่ 3.3 องค์ประกอบภายใน MCU เมื่อใช้การซับแซมปลิ้งค่า Cb และ Cr อัตราส่วน H: 2, V : 2

3.2 การบีบอัดข้อมูลภาพแบบ JPEG

ในการบีบอัดข้อมูลแบบ JPEG จะทำการบีบอัดข้อมูล MCU ทีละหน่วยโดยมีลำดับการบีบอัดเรียงจากด้านซ้ายไปด้านขวาและจากด้านบนลงไปด้านล่างของภาพ การบีบอัดข้อมูล MCU แต่ละหน่วยจะทำทีละบล็อก (8×8 พิกเซล) ขององค์ประกอบภายใน MCU หน่วยนั้นโดยเรียงลำดับตามลำดับการเรียงตัวของบล็อกองค์ประกอบภายใน MCU ที่แสดงไปแล้ว ซึ่งการบีบอัดข้อมูลในแต่ละบล็อกมีขั้นตอนดังนี้

3.2.1 การแปลงบล็อกข้อมูลด้วยวิธี DCT

การบีบอัดข้อมูลในระบบนี้จะทำในโดเมนของความถี่โดยใช้ DCT แปลงสัญญาณภาพในสปาเชียลโดเมน (Spatial domain) ไปเป็นค่าสัมประสิทธิ์การแปลงในโดเมนของความถี่ โดยมีสมการการแปลงค่าเป็น

$$C(u,v) = \frac{1}{\sqrt{2N}}\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)Cos[\frac{(2x+1)u\pi}{2N}]Cos[\frac{(2y+1)v\pi}{2N}] \qquad (3.2)$$

เมื่อ $C(u,v)$ คือ ค่าสัมประสิทธิ์ DCT

$f(x,y)$ คือ ค่าของระดับสีในพิกเซล

$\alpha(i) = \frac{1}{\sqrt{2}}$ เมื่อ $i = 0$ และ $\alpha(i) = 0$ เมื่อ $i > 0$

ในการแปลงค่าสัมประสิทธิ์ DCT ถ้าจำนวนข้อมูลยิ่งมาก (N ยิ่งมาก) จะทำให้ใช้เวลาและกำลังในการคำนวณมาก ดังนั้น JPEG จึงกำหนดการแปลงภาพโดนแบ่งการแปลงเป็นบล็อกย่อยๆ ของภาพขนาดบล็อกละ 8×8 พิกเซล (N=8) ก่อนที่จะทำการแปลง บล็อกข้อมูลจะต้องมีการลดระดับ DC ของบล็อกข้อมูลก่อน ด้วยการลบข้อมูลทุกตัวภายในบล็อกด้วย 128 (ซึ่งมาจากค่าของ 2^{n-1} เมื่อ n คือจำนวนบิตที่ใช้เก็บข้อมูลสีแต่ละสีในจุดพิกเซล ซึ่งในที่นี้จะเท่ากับ 8) จากนั้นทำการแปลงบล็อกข้อมูลด้วย DCT ตามสมการผลที่ได้คือบล็อกข้อมูลสัมประสิทธิ์ DCT ขนาด 8×8 ค่า โดยข้อมูลที่ตำแหน่งบนซ้ายสุดของบล็อกข้อมูลคือค่าสัมประสิทธิ์ DC แสดงระดับ DC ของบล็อกข้อมูล ส่วนข้อมูลในตำแหน่งที่เหลือคือค่าสัมประสิทธิ์ AC แสดงระดับ AC ในความถี่ต่างๆ โดยจะมีแนวโน้มที่ลดลงเมื่อค่าสัมประสิทธิ์ข้อมูลของความถี่ที่สูงขึ้น เนื่องมาจากภาพส่วนใหญ่จะมีการกระจายตัวของข้อมูลสีอยู่ในช่วงความถี่ต่างๆ

3.2.2 การควอนไทเซชั่น (Quantization)

การควอนไทเซชั่นเป็นการลดค่าของข้อมูลภายในบล็อกข้อมูลสัมประสิทธิ์ DCT เพื่อจะลดจำนวนบิตที่ใช้ในการเก็บข้อมูลเหล่านี้ในขั้นตอนการเข้ารหัสต่อไป และจากการที่ภาพส่วนใหญ่มีการกระจายตัวของข้อมูลสีอยู่ในช่วงความถี่ต่ำๆ JPEG จึงให้ความสำคัญต่อข้อมูลในช่วงความถี่ต่ำมากกว่าข้อมูลในช่วงความถี่สูง ดังนั้นการควอนไทเซชั่นคือความพยายามลดความสำคัญของข้อมูลในช่วงความถี่สูงโดยพยายามทำให้เป็นค่าศูนย์ให้มากที่สุดนั่นเอง โดยใช้บล็อกข้อมูลของควอนไทเซอร์ (Quantizer) 8×8 ค่า หารด้วยบล็อกข้อมูลสัมประสิทธิ์ DCT 8×8 ค่าดังสมการต่อไปนี้

$$\hat{T}(u,v) = round\ [\frac{T(u,v)}{Q(u,v)}] \qquad (3.3)$$

เมื่อ $T(u,v)$ คือ ค่าสัมประสิทธิ์ DCT

$\hat{T}(u,v)$ คือค่าสัมประสิทธิ์ DCT เมื่อถูกควอนไทซ์

$Q(u,v)$ คือค่าควอนไทเซอร์

$round$ คือการหาจำนวนเต็มที่มีค่าใกล้เคียงที่สุด

โดยที่ $0 \le u \le 7, 0 \le v \le 7$

นอกจากนี้ JPEG ยังได้แยกเป็นค่าควอนไทเซอร์ที่ใช้สำหรับข้อมูลลูมิแนนซ์และโครมิแนนซ์ โดยให้ค่าควอนไทเซอร์ของโครมิแนนซ์มีค่ามากกว่าลูมิแนนซ์ เนื่องจาก JPEG ให้ความสำคัญต่อข้อมูลลูมิแนนซ์มากกว่าโครมิแนนซ์ บล็อกข้อมูลควอนไทเซอร์ที่ใช้ในโปรแกรมสำหรับลูมิแนนซ์และโครมิแนนซ์เป็นดังนี้

$$Q(u,v)Lumi_nance = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 12 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 81 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$Q(u,v)Chromi_nance = \begin{bmatrix} 17 & 18 & 24 & 47 & 99 & 99 & 99 & 99 \\ 18 & 21 & 26 & 66 & 99 & 99 & 99 & 99 \\ 24 & 26 & 56 & 99 & 99 & 99 & 99 & 99 \\ 47 & 66 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \\ 99 & 99 & 99 & 99 & 99 & 99 & 99 & 99 \end{bmatrix}$$

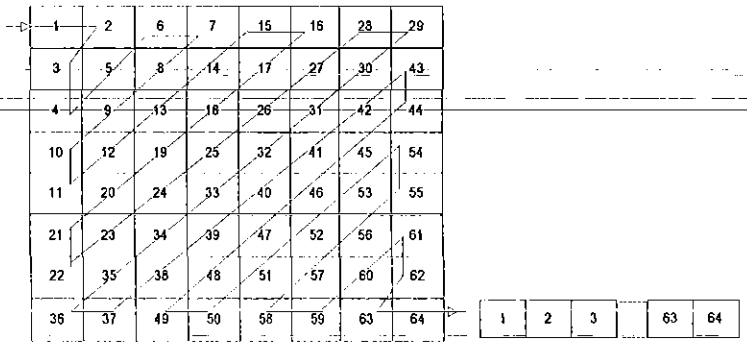
และได้ออกแบบให้โปรแกรมสามารถเปลี่ยนค่า Q (u,v) ภายในบล็อกของลูมิแนนซ์และโครมิแนนซ์ได้หลายระดับ ด้วยการนำค่าของ Quantization factor มาเพิ่มให้กับควอนไทเซอร์ทั้ง 64 ตัวภายในบล็อกทั้งสองโดยรับค่ามาจากโปรแกรมซึ่งกำหนดให้เพิ่มได้ครั้งละ 10 จะได้สมการควอนไทเซชั่นที่ใช้ในโปรแกรมดังสมการที่ 3.4

$$\hat{T}(u,v) = round\ [\frac{T(u,v)}{Q(u,v) + Q.F.}] \qquad (3.4)$$

เมื่อ Q.F. คือค่า Quantization_Factor มีค่าตั้งแต่ 0 ถึง 100

3.2.3 การเรียงข้อมูลแบบซิกแซก (Zigzag)

การเรียงข้อมูลแบบซิกแซกเป็นความพยายามที่จะนำค่าสัมประสิทธิ์ DCT ที่ถูกควอนไทซ์ ที่เป็นค่าศูนย์มาเรียงให้ติดกัน เพื่อประโยชน์ในการบีบอัดข้อมูลในขั้นตอนเข้ารหัส โดยมีลำดับการเรียงข้อมูลภายในบล็อกดังแสดงต่อไปนี้



รูปที่ 3.4 ลำดับการจัดข้อมูลแบบซิกแซก

3.2.4 การเข้ารหัสฮัฟแมน (Huffman Encoding)

การเข้ารหัสฮัฟแมนเป็นวิธีการเข้ารหัสข้อมูลด้วยการสร้างรหัสขนาดต่างๆ ขึ้นมาจากสถิติของจำนวนข้อมูลที่เกิดขึ้นเพื่อนำรหัสที่สร้างขึ้นนั้นไปใช้แทนตัวข้อมูลเดิม โดยพยายามสร้างรหัสที่มีขนาดสั้นสำหรับใช้แทนตัวข้อมูลที่เกิดขึ้นจำนวนมากๆ และรหัสที่มีขนาดยาวขึ้นเพื่อแทนตัวข้อมูลที่เกิดขึ้นน้อยลงลดหลั่นกันไป ตามลำดับ สำหรับวิธีการเข้ารหัสฮัฟแมนที่ใช้ใน JPEG นั้น ได้มีการอาศัยเทคนิคการเข้ารหัสวิธีอื่นมาช่วยในการบีบอัดข้อมูลด้วยคือการเข้ารหัส Run Length

ในการเข้ารหัสแถวข้อมูลที่ได้จากขั้นตอนซิกแซกจะแยกเป็น 2 ส่วน คือ การเข้ารหัสค่า DC และค่า AC ตามลำดับ

1) การเข้ารหัสค่า DC ของแถวข้อมูล ในการเข้ารหัสค่า DC ของแถวข้อมูลนั้น (ข้อมูลตัวแรกในแถว) จะถูกเข้ารหัสเฉพาะค่าความแตกต่างระหว่างค่า DC ของแถวข้อมูลปัจจุบันกับค่า DC ของแถวข้อมูลก่อนหน้านี้ ที่เป็นสีชนิดเดียวกันที่ถูกเข้ารหัส เช่น เมื่อทำการเข้ารหัสค่า DC ของแถวข้อมูลสี Cr ซึ่งมีค่าเท่ากับ 45 จะต้องหาค่าผลต่างกับค่า DC ของแถวข้อมูลสี Cr ที่ถูกเข้ารหัสผ่านไปแล้วแถวล่าสุด (สมมติเท่ากับ 20) ดังนั้นค่าผลต่างที่นำมาเข้ารหัสคือ 25 (45-20) วิธีการเข้ารหัสข้อมูลดังกล่าวจะมีรหัสเอาท์พุต 2 ตัว สำหรับใช้แทนค่าข้อมูลโดยรหัสตัวแรกหาจากตารางฮัฟแมน DC และอีกตัวหนึ่งหาจากค่าข้อมูลที่นำมาเข้ารหัส

ตารางที่ 3.1 ตารางค่า Categories

Categories	Magnitude Range
0	0
1	-1,1
2	-3,-2,2,3
3	-7…-4,4…7
4	-15…-8,8…15
5	-31…-16,16…31
6	-63…-32,32…63
7	-127…-64,64…127
8	-255…-128,128…255
9	-511…-256,256…511
A	-1023…-512,512…1023
B	-2047…-1024,1024…2047

ในตารางฮัฟแมนของ DC จะเก็บรหัสความยาวต่างๆ ที่ใช้แทนข้อมูลตามค่า Category ของข้อมูลนั้น ซึ่งการหาค่า Category ของข้อมูลสามารถหาได้จากตารางที่ 3.1 ตารางค่า Categories เมื่อได้ค่า Category แล้วก็สามารถหารหัส จากตารางฮัฟแมนได้ด้วยการเทียบหาตามค่า Category จะเห็นว่า รหัสที่ได้จากตารางฮัฟแมนนั้นใช้เป็นตัวบอกช่วงของข้อมูลที่ถูกเข้ารหัสเท่านั้น ซึ่งการระบุค่าของข้อมูลนั้นจำเป็นต้องใช้รหัสตัวที่สองเป็นตัวระบุคือหาจากค่าของข้อมูลโดยตรง โดยจะมีจำนวนบิตเท่ากับค่า Category ที่หามาได้ (เป็นเทคนิคของ Variable-Length Encoding) โดยรหัสนี้จะเอามาจาก LSB (Least

Significant Bit) จำนวนบิตเท่ากับ Category ของข้อมูลที่นำมาเข้ารหัสนั่นเองและจะเอามาจากค่า 2's Complement ถ้าข้อมูลมีค่าเป็นลบ

2) การเข้ารหัสค่าAC ของแถวข้อมูล การเข้ารหัสค่า AC (ข้อมูลตัวที่ 2 ถึงตัวที่ 64 ในแถวข้อมูล) จะต่างจากการเข้ารหัสค่า DC คือทำการเข้ารหัสค่าของข้อมูลโดยตรง (จะไม่เข้ารหัสเฉพาะค่าผลต่างเหมือนใน DC) และการเข้ารหัสค่า AC นั้นจะทำการเข้ารหัสเฉพาะค่า AC ภายในแถวที่มีค่าไม่เป็นศูนย์เท่านั้น ส่วนค่า AC ที่เป็นศูนย์นั้น JPEG จะอาศัยเทคนิคการเข้ารหัสแบบ Run-Length มาช่วยในการเข้ารหัสดังนี้ คือการเข้ารหัสค่า AC ที่ไม่เป็นศูนย์แต่ละตัวจะมีการนับจำนวนข้อมูล AC ที่เป็นศูนย์ซึ่งอยู่ติดกันด้านหน้าของข้อมูลที่จะเข้ารหัสนั้นโดยถือเป็นค่า Run ของข้อมูลในการเข้ารหัส เช่น ค่า AC ภายในแถวมีค่าเป็น 5 8 0 0 9 7 0 0 0 0 4 ... จะทำการเข้ารหัสเฉพาะข้อมูลที่เป็นศูนย์และมีค่า Run ของข้อมูลแต่ละตัวดังนี้คือ 5 (Run = 0) 8 (Run = 0) 9(Run = 2) 7(Run = 0) 4(Run = 4) ... โดยมีวิธีการเข้ารหัสดังกล่าวคล้ายใน DC คือจะหารหัสเอาท์พุต 2 ตัวสำหรับใช้แทนข้อมูลแต่ละตัว โดยรหัสตัวแรกหาได้จากตารางฮัฟแมนของ AC และรหัสตัวที่สองหาจากค่าของข้อมูลที่นำมาเข้ารหัสโดยตรง

เนื่องจากข้อมูลที่นำมาเข้ารหัสมีค่า Run เข้ามาเกี่ยวข้อง ดังนั้นตารางฮัฟแมนของ AC จะต่างจากตารางฮัฟแมนของ DC คือจะมีการเปลี่ยนแปลงจาก Category ใน DC เป็น Run/Category นั่นคือการหารหัสจากตารางฮัฟแมนของ AC จะต้องทราบค่า Run และ Category ของข้อมูลที่นำมาเข้ารหัสซึ่งการหาค่า Category ของข้อมูล AC จะเทียบหาจากตารางค่า Categories ตามตารางที่ 3.1 เช่นเดียวกับใน DC เมื่อทราบค่า Run และ Category ของข้อมูลแล้วก็จะสามารถหารหัสจากตารางฮัฟแมนของ AC ได้โดยเทียบตามค่า Run/Category

ภายในตารางฮัฟแมนของ AC จะมีรหัสพิเศษ 2 ตัวคือรหัส เมื่อค่า Run/Category เท่ากับ 0/0 ใช้ในกรณีเมื่อทำการเข้ารหัสข้อมูลภายในแถวจนกระทั่งเหลือแต่ข้อมูลที่เป็นศูนย์เพียงอย่างเดียวก็ใช้ รหัสนี้เป็นตัวบอกในขั้นตอนการถอดรหัสว่าข้อมูลที่เหลือทั้งหมดภายในแถวมีค่าเป็นศูนย์ และอีก รหัสตัวหนึ่งเมื่อ Run/Category เท่ากับ F/0 ใช้เมื่อมีข้อมูลที่มีค่าศูนย์อยู่ติดกัน 16ตัวภายในแถว โดยยังมีข้อมูลที่ไม่เป็นศูนย์ที่ยังไม่ถูกเข้ารหัสเหลืออยู่ในแถวอีก จะใช้รหัสตัวนี้แทนข้อมูลที่มีค่าเป็นศูนย์16 ตัว

จะเห็นได้ว่ารหัสที่ได้จากตารางฮัฟแมนของ AC นั้นใช้ในการบอกจำนวนศูนย์ (คือค่า Run) ที่อยู่ด้านหน้าของข้อมูลนั้นและช่วงของค่าข้อมูล (คือค่า Category) ที่นำมาเข้ารหัส ดังนั้นจะต้องมีรหัสตัวที่สองสำหรับใช้ในการระบุค่าของข้อมูลซึ่งการหารหัสตัวที่สองนี้วิธีเดียวกันกับใน DC คือเอามาจาก LSB จำนวน Category บิตของข้อมูลที่นำมาเข้ารหัส และเอามาจากค่า 2's Complement ถ้าข้อมูลมีค่าเป็นลบ

JPEG ได้กำหนดตารางฮัฟแมน ของ AC และ DC ดังนี้

ตารางที่ 3.2 ตารางฮัฟแมนของลูมิแนนซ์ AC

Run/Cat	ความยาวรหัส	รหัส
0x00	4	1010 (EOB)
0x01	2	00
0x02	2	01
0x03	3	100
0x04	4	1011
0x05	5	11010
0x06	7	1111000
0x07	8	11111000
0x08	10	1111110110
0x09	16	1111111110000010
0x0A	16	1111111110000011
0x11	4	1100
0x12	5	11011
0x13	7	1111001
0x14	9	111110110
0x15	11	11111110110
0x16	16	1111111110000100
0x17	16	1111111110000101
0x18	16	1111111110000110
0x19	16	1111111110000111
0x1A	16	1111111110001000
0x21	5	11100
0x22	8	11111001
0x23	10	1111110111
0x24	12	111111110100
0x25	16	1111111110001001
0x26	16	1111111110001010
0x27	16	1111111110001011
0x28	16	1111111110001100
0x29	16	1111111110001101
0x2A	16	1111111110001110
0x31	6	111010
0x32	9	111110111
0x33	12	111111110101
0x34	16	1111111110001111
0x35	16	1111111110010000
0x36	16	1111111110010001
0x37	16	1111111110010010
0x38	16	1111111110010011
0x39	16	1111111110010100
0x3A	16	1111111110010101
0x41	6	111011
0x42	10	1111111000
0x43	16	1111111110010110
0x44	16	1111111110010111
0x45	16	1111111110011000
0x46	16	1111111110011001
0x47	16	1111111110011010
0x48	16	1111111110011011
0x49	16	1111111110011100
0x4A	16	1111111110011101
0x51	7	1111010
0x52	11	11111110111
0x53	16	1111111110011110

ตารางที่ 3.2 (ต่อ)

Run/Cat	ความยาวรหัส	รหัส
0x54	16	1111111110011111
0x55	16	1111111110100000
0x56	16	1111111110100001
0x57	16	1111111110100010
0x58	16	1111111110100011
0x59	16	1111111110100100
0x5A	16	1111111110100101
0x61	7	1111011
0x62	12	111111110110
0x63	16	1111111110100110
0x64	16	1111111110100111
0x65	16	1111111110101000
0x66	16	1111111110101001
0x67	16	1111111110101010
0x68	16	1111111110101011
0x69	16	1111111110101100
0x6A	16	1111111110101101
0x71	8	11111010
0x72	12	111111110111
0x73	16	1111111110101110
0x74	16	1111111110101111
0x75	16	1111111110110000
0x76	16	1111111110110001
0x77	16	1111111110110010
0x78	16	1111111110110011
0x79	16	1111111110110100
0x7A	16	1111111110110101
0x81	9	111111000
0x82	15	111111111000000
0x83	16	1111111110110110
0x84	16	1111111110110111
0x85	16	1111111110111000
0x86	16	1111111110111001
0x87	16	1111111110111010
0x88	16	1111111110111011
0x89	16	1111111110111100
0x8A	16	1111111110111101
0x91	9	111111001
0x92	16	1111111110111110
0x93	16	1111111110111111
0x94	16	1111111111000000
0x95	16	1111111111000001
0x96	16	1111111111000010
0x97	16	1111111111000011
0x98	16	1111111111000100
0x99	16	1111111111000101
0x9A	16	1111111111000110
0xA1	9	111111010
0xA2	16	1111111111000111
0xA3	16	1111111111001000
0xA4	16	1111111111001001
0xA5	16	1111111111001010
0xA6	16	1111111111001011
0xA7	16	1111111111001100

ตารางที่ 3.2 (ต่อ)

Run/Cat	ความยาวรหัส	รหัส
0xA8	16	1111111111001101
0xA9	16	1111111111001110
0xAA	16	1111111111001111
0xB1	10	1111111001
0xB2	16	1111111111010000
0xB3	16	1111111111010001
0xB4	16	1111111111010010
0xB5	16	1111111111010011
0xB6	16	1111111111010100
0xB7	16	1111111111010101
0xB8	16	1111111111010110
0xB9	16	1111111111010111
0xBA	16	1111111111011000
0xC1	10	1111111010
0xC2	16	1111111111011001
0xC3	16	1111111111011010
0xC4	16	1111111111011011
0xC5	16	1111111111011100
0xC6	16	1111111111011101
0xC7	16	1111111111011110
0xC8	16	1111111111011111
0xC9	16	1111111111100000
0xCA	16	1111111111100001
0xD1	11	11111111000
0xD2	16	1111111111100010
0xD3	16	1111111111100011
0xD4	16	1111111111100100
0xD5	16	1111111111100101
0xD6	16	1111111111100110
0xD7	16	1111111111100111
0xD8	16	1111111111101000
0xD9	16	1111111111101001
0xDA	16	1111111111101010
0xE1	16	1111111111101011
0xE2	16	1111111111101100
0xE3	16	1111111111101101
0xE4	16	1111111111101110
0xE5	16	1111111111101111
0xE6	16	1111111111110000
0xE7	16	1111111111110001
0xE8	16	1111111111110010
0xE9	16	1111111111110011
0xEA	16	1111111111110100
0xF0	11	11111111001 (ZRL)
0xF1	16	1111111111110101
0xF2	16	1111111111110110
0xF3	16	1111111111110111
0xF4	16	1111111111111000
0xF5	16	1111111111111001
0xF6	16	1111111111111010
0xF7	16	1111111111111011
0xF8	16	1111111111111100
0xF9	16	1111111111111101
0xFA	16	1111111111111110

ตารางที่ 3.3 ตารางฮัฟแมนของโครมิแนนซ์ AC

Run/Cat	ความยาว	รหัส
0x00	2	00 (EOB)
0x01	2	01
0x02	3	100
0x03	4	1010
0x04	5	11000
0x05	5	11001
0x06	6	111000
0x07	7	1111000
0x08	9	111110100
0x09	10	1111110110
0x0A	12	111111110100
0x11	4	1011
0x12	6	111001
0x13	8	11110110
0x14	9	111110101
0x15	11	11111110110
0x16	12	111111110101
0x17	16	1111111110001000
0x18	16	1111111110001001
0x19	16	1111111110001010
0x1A	16	1111111110001011
0x21	5	11010
0x22	8	11110111
0x23	10	1111110110
0x24	12	111111110110
0x25	15	111111111000010
0x26	16	1111111110001100
0x27	16	1111111110001101
0x28	16	1111111110001110
0x29	16	1111111110001111
0x2A	16	1111111110010000
0x31	5	11011
0x32	8	11111000
0x33	10	1111111000
0x34	12	111111110111
0x35	16	1111111111001000 1
0x36	16	1111111111001001 0
0x37	16	1111111111001001 1
0x38	16	1111111111001010 0
0x39	16	1111111111001010 1
0x3A	16	1111111111001011 0
0x41	6	111010
0x42	9	111110110
0x43	16	1111111110010111
0x44	16	1111111110011000
0x45	16	1111111110011001
0x46	16	1111111110011010
0x47	16	1111111110011011
0x48	16	1111111110011100
0x49	16	1111111110011101
0x4A	16	1111111110011110
0x51	6	111011
0x52	10	1111111001
0x53	16	1111111100111111

ตารางที่ 3.3 (ต่อ)

Run/Cat	ความยาวรหัส	รหัส
0x54	16	1111111101000000
0x55	16	1111111101000001
0x56	16	1111111101000010
0x57	16	1111111101000011
0x58	16	1111111101000100
0x59	16	1111111101000101
0x5A	16	1111111101000110
0x61	7	1111001
0x62	11	11111110111
0x63	16	1111111101000111
0x64	16	1111111101001000
0x65	16	1111111101001001
0x66	16	1111111101001010
0x67	16	1111111101001011
0x68	16	1111111101001100
0x69	16	1111111101001101
0x6A	16	1111111101001110
0x71	7	1111010
0x72	11	11111111000
0x73	16	1111111110101111
0x74	16	1111111110110000
0x75	16	1111111110110001
0x76	16	1111111110110010
0x77	16	1111111110110011
0x78	16	1111111110110100
0x79	16	1111111110110101
0x7A	16	1111111110110110
0x81	8	11111001
0x82	16	1111111110110111
0x83	16	1111111110111000
0x84	16	1111111110111001
0x85	16	1111111110111010
0x86	16	1111111110111011
0x87	16	1111111110111100
0x88	16	1111111110111101
0x89	16	1111111110111110
0x8A	16	1111111110111111
0x91	9	111110111
0x92	16	1111111111000000
0x93	16	1111111111000001
0x94	16	1111111111000010
0x95	16	1111111111000011
0x96	16	1111111111000100
0x97	16	1111111111000101
0x98	16	1111111111000110
0x99	16	1111111111000111
0x9A	16	1111111111001000
0xA1	9	111111000
0xA2	16	1111111111001001
0xA3	16	1111111111001010
0xA4	16	1111111111001011
0xA5	16	1111111111001100
0xA6	16	1111111111001101
0xA7	16	1111111111001110

ตารางที่ 3.3 (ต่อ)

Run/Cat	ความยาวรหัส	รหัส
0xA8	16	1111111111001111
0xA9	16	1111111111010000
0xAA	16	1111111111010001
0xB1	9	111111001
0xB2	16	1111111111010010
0xB3	16	1111111111010011
0xB4	16	1111111111010100
0xB5	16	1111111111010101
0xB6	16	1111111111010110
0xB7	16	1111111111010111
0xB8	16	1111111111011000
0xB9	16	1111111111011001
0xBA	16	1111111111011010
0xC1	9	111111010
0xC2	16	1111111111011011
0xC3	16	1111111111011100
0xC4	16	1111111111011101
0xC5	16	1111111111011110
0xC6	16	1111111111011111
0xC7	16	1111111111100000
0xC8	16	1111111111100001
0xC9	16	1111111111100010
0xCA	16	1111111111100011
0xD1	11	11111111001
0xD2	16	1111111111100100
0xD3	16	1111111111100101
0xD4	16	1111111111100110
0xD5	16	1111111111100111
0xD6	16	1111111111101000
0xD7	16	1111111111101001
0xD8	16	1111111111101010
0xD9	16	1111111111101011
0xDA	16	1111111111101100
0xE1	14	11111111101100
0xE2	16	1111111111101101
0xE3	16	1111111111101110
0xE4	16	1111111111101111
0xE5	16	1111111111110000
0xE6	16	1111111111110001
0xE7	16	1111111111110010
0xE8	16	1111111111110011
0xE9	16	1111111111110100
0xEA	16	1111111111110101
0xF0	10	1111111010 (ZRL)
0xF1	15	111111111000011
0xF2	16	1111111111110110
0xF3	16	1111111111110111
0xF4	16	1111111111111000
0xF5	16	1111111111111001
0xF6	16	1111111111111010
0xF7	16	1111111111111011
0xF8	16	1111111111111100
0xF9	16	1111111111111101
0xFA	16	1111111111111110

ตารางที่ 3.4 สรุปความยาว ลูมิแนนซ์ และโครมิแนนซ์ของค่าสัมประสิทธิ์ AC

ความยาว	ลูมิแนนซ์	โครมิแนนซ์
1	0	0
2	2	2
3	1	1
4	3	2
5	3	4
6	2	4
7	4	3
8	3	4
9	5	7
10	5	5
11	4	4
12	4	4
13	0	0
14	0	1
15	1	2
16	125	119

ตารางที่ 3.5 ตารางฮัฟแมนของลูมิแนนซ์ DC

Category	ความยาว	รหัส
0	2	00
1	3	010
2	3	011
3	3	100
4	3	101
5	3	110
6	4	1110
7	5	11110
8	6	111110
9	7	1111110
A	8	11111110
B	9	111111110

ตารางที่ 3.6 ตารางฮัฟแมนของโครมิแนนซ์DC

Category	ความยาว	รหัส
0	2	00
1	2	01
2	2	10
3	3	110
4	4	1110
5	5	11110
6	6	111110
7	7	1111110
8	8	11111110
9	9	111111110
A	10	1111111110
B	11	11111111110

ตัวอย่างการเข้ารหัสข้อมูล DC และ AC เริ่มจากข้อมูลเป็นบล็อกขนาด 8x8 พิกเซล ดังแสดงในตารางที่ 3.7 ซึ่งก่อนที่ทำมาแปลง FDCT ต้องลบแต่ละพิกเซลด้วย 2^7 ก่อนจะผลลัพธ์ในตาราง ที่ 3.8 ต่อจากนั้นทำการแปลง FDCT จะได้สัมประสิทธิ์ DCT ดังตารางที่ 3.9 แล้วนำสัมประสิทธิ์ DCT ที่ได้มาควอนไทซ์จะได้ดังตารางที่ 3.11

ตารางที่ 3.7 บล็อกข้อมูลภาพตัวอย่าง ขนาด 8× 8 พิกเซล

52	55	61	66	70	61	64	73
63	59	66	90	109	85	69	72
-62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

ตาราง ที่ 3.8 บล็อกข้อมูลภาพตัวอย่าง 8 ×8 พิกเซล ลบกับ 128

-76	-73	-67	-62	-58	-67	-64	-55
-65	-69	-62	-38	-19	-43	-59	-56
-66	-69	-60	-15	16	-24	-62	-55
-65	-70	-57	-6	26	-22	-58	-59
-61	-67	-60	-24	-2	-40	-60	-58
-49	-63	-68	-58	-51	-65	-70	-53
-43	-57	-64	-69	-73	-67	-63	-45
-41	-49	-59	-60	-63	-52	-50	-34

ตารางที่ 3.9 บล็อกข้อมูลภาพที่ทำการแปลง FDCT แล้ว

-415	-29	-62	-25	-55	-20	-1	3
7	-21	-62	9	11	-7	-6	6
-46	8	77	-25	-30	10	7	-5
-50	13	35	-15	-9	6	0	3
11	-8	-13	-2	-1	1	-4	1
-10	1	3	-3	-1	0	2	-1
-4	-1	2	-4	2	-3	1	-2
-1	-1	-1	-2	-1	-1	0	-1

ตารางที่ 3. 10 ค่าสัมประสิทธิ์การควอนไทซ์

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

ตารางที่ 3.11 ค่าที่ได้จากการควอนไทซ์แล้ว

-26	-3	-6	2	2	0	0	0
1	-2	-4	0	0	0	0	0
-3	1	5	-1	-1	0	0	0
-4	1	2	-1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

เมื่อนำสัมประสิทธิ์ DCT ที่ถูกควอนไทซ์แล้วมาเรียงลำดับซิกแซกจะได้ ข้อมูลดังนี้

[-26 - 3 1 -3 -2 -6 2 -4 1 -4 1 1 5 0 2 0 0 -1 2 0 0 0 0 0 -1 -1 EOB]

จากนั้น นำลำดับซิกแซกที่ได้นี้ ไปเข้ารหัสฮัฟแมน ซึ่งแยกเป็นสองส่วนคือเข้ารหัส DC และ AC

ในการเข้ารหัส DC เป็นการเข้ารหัสผลต่างของ DC จำเป็นที่จะต้องรู้ค่าของ DC ของบล็อกก่อนหน้านี้ จากตัวอย่างสมมติให้ ค่า DC ของบล็อกก่อนหน้านี้เป็น -17 จะได้ ค่าผลต่างเป็น -26 – (-17) = -9 Category ของ -9 คือ 4 เมื่อเทียบค่าในตารางฮัฟแมนของลูมิแนนซ์ DC จะได้รหัสส่วนแรกคือ 101 และรหัสส่วนที่สองจะได้จาก - 9 มีค่าเป็นลบ ต้องทำเป็น 2's Complement แล้วลบด้วย 1 จะได้ 0111 -1 =0110 ดังนี้จะได้รหัสของ DC คือ 1010110

ส่วนการเข้ารหัส AC จะได้ Run/Category ของแต่ละ AC ตามลำดับ ดังนี้

-3 (0/2) 1 (0/1) -3 (0/2) -2 (0/2) -6 (0/3) 2 (0/2) -4 (0/3) 1 (0/1) -4 (0/3) 1 (0/1) 1 (0/1)

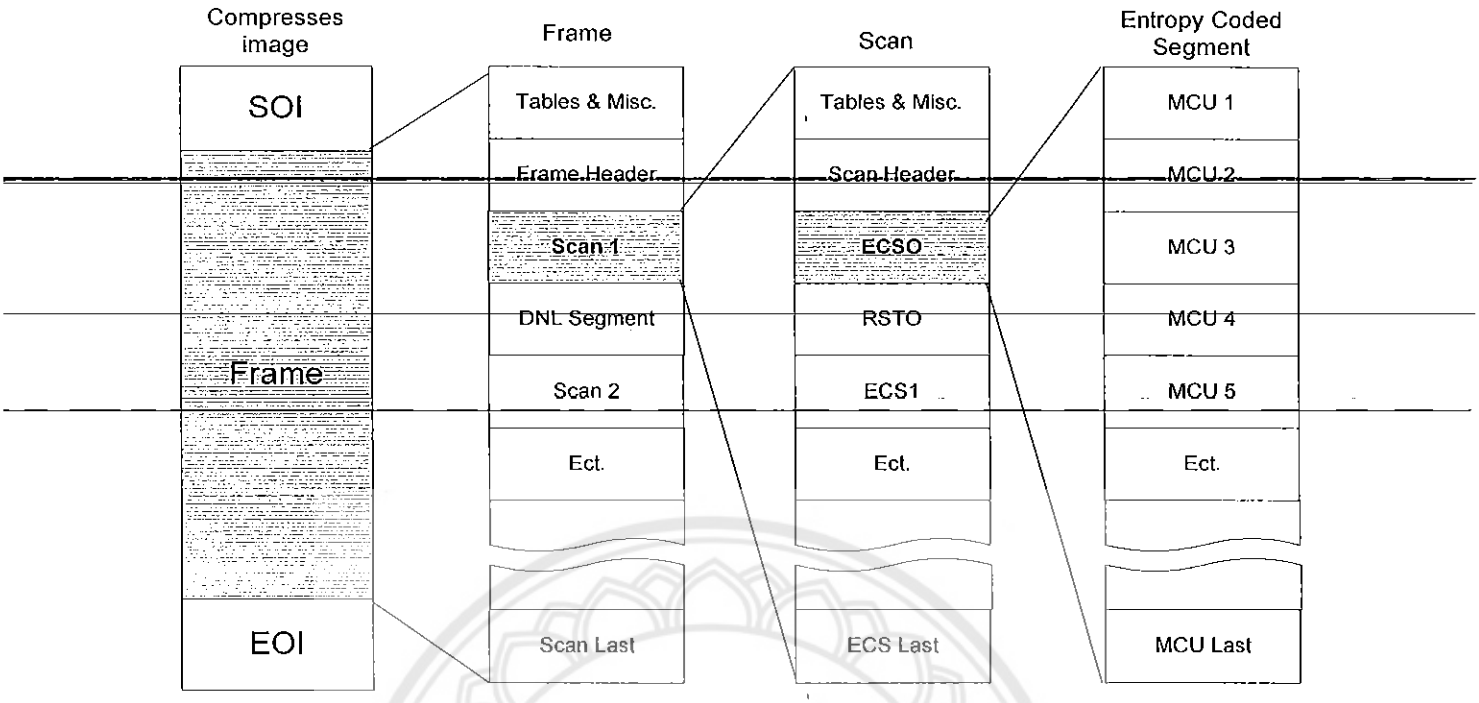
5 (0/3) 2 (1/2) -1 (2/1) 2 (0/2) -1 (5/1) -1 (0/1) EOB (0/0)

จะได้รหัสคือ รหัสส่วนแรก จะได้จาก Run/Category รหัสส่วนที่สองจะนำข้อมูลมาเข้ารหัสเลยถ้าเป็น ลบให้ ทำ 2's Complement แล้วลบด้วย 1 ก่อน จะได้รหัส ดังนี้ 0100 001 0100 0101 100001 0110 100011 001 100011 001 001 100101 1100110 110110 0110 11110100 000 1010

3.3 การเก็บข้อมูลในรูปไฟล์ JFIF

ข้อมูลที่ได้จากการบีบอัดข้อมูลแบบ JPEG นั้นจะจัดเก็บในรูปไฟล์ JFIF โดยมีการเก็บข้อมูลที่จำเป็นในการขยายข้อมูลภาพรวมเข้าไปในไฟล์ด้วยคือตารางฮัฟแมนของ AC และ DC บล็อกข้อมูลของควอนไทเซอร์ จำนวนสี ขนาดของภาพ สัดส่วนการซับแซมปลิ้ง โดยจะเก็บไว้ในส่วนเฮดเดอร์ ซึ่งแบ่งออกเป็นส่วนย่อยๆ แต่ละส่วนมีรูปแบบการเก็บข้อมูลต่างๆกัน ดังนี้

จากตารางที่ 3.12 แสดงมาร์คเกอร์และพารามิเตอร์ที่ใช้กับ JPEG ซึ่งเป็นส่วนสำคัญในมาตรฐานการบีบอัด JPEG



รูปที่ 3.5 รูปแบบการบีบอัดข้อมูลภาพแบบ Sequential DCT-based

ตารางที่ 3.12 มาร์คเกอร์ที่ใช้กับ JPEG

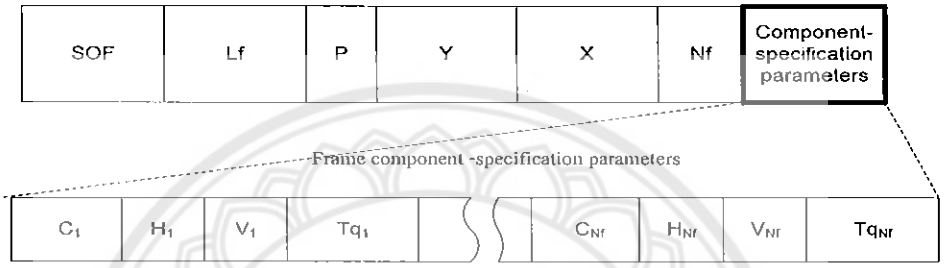
0xFFC0	(SOF) มาร์คเกอร์เฟรมเริ่มต้น
0xFFC4	(DHT) กำหนดตารางฮัฟแมน
0xFFD0 ถึง 0xFFD7	(RST) รีสตาร์ทช่องระหว่างการสิ้นสุด
0xFFD8	(SOI) จุดเริ่มต้นของภาพ
0xFFD9	(EOI) จุดสิ้นสุดของภาพ
0xFFDA	(SOS) จุดเริ่มสแกน
0xFFDB	(DQT) กำหนดตารางการควอนไทเซชั่น
0xFFDC	(DNL) กำหนดจำนวนเส้น
0xFFDD	(DRI) กำหนด Restart Interval
0xFFE0 ถึง 0xFFEF	(APP) ส่วนของมาร์คเกอร์
0xFFFE	(COM) จุดเริ่มคอมเมนต์

ในการบีบอัดข้อมูลจะเริ่มด้วยมาร์คเกอร์จุดเริ่มต้นของภาพ (Start Of Image: SOI) และจบด้วยมาร์คเกอร์จุดสิ้นสุดของภาพ (Ends of Image: EOI) ข้อมูลภาพได้แบ่งออกเป็นส่วนๆ แต่ละเฟรมประกอบด้วยส่วนของสแกนเรียงเป็นลำดับ เฟรมเฮดเดอร์สามารถกำหนดตารางที่ใช้ในการแปลง DCT

และการเข้ารหัสฮัฟแมน รูปแบบของตารางจะแสดงในหัวข้อถัดไป ส่วนของสแกน (Scan) ประกอบด้วยรหัสเอนโทรปี แต่ละส่วนของสแกนประกอบด้วยหน่วยของรหัสที่เล็กที่สุดหรือ MCU

3.3.1 เฟรมเฮดเดอร์

เฟรมเฮดเดอร์ประกอบด้วยพารามิเตอร์ของข้อมูลหลายค่าและมีค่าที่แน่นอน ส่วนประกอบของเฟรมเฮดเดอร์มีทั้งหมด 6 ส่วนดังรูปที่ 3.6 และตารางที่ 3.13 รายละเอียดในแต่ละส่วน เป็นดังนี้



รูปที่ 3.6 โครงสร้างของ เฟรมเฮดเดอร์

ตารางที่ 3.13 ส่วนประกอบของเฟรมเฮดเดอร์

พารามิเตอร์	ขนาด(บิต)	ค่า	รายละเอียด
SOF	16	0xFFC0	จุดเริ่มต้นของเฟรม
Lf	16	3Nf + 8	ความยาวเฟรมเฮดเดอร์
P	8	8	ค่าความแม่นยำ
Y	16	0-65535	จำนวนเส้น
X	16	1-65535	ข้อมูลต่อเส้น
Nf	8	1-255	จำนวนขององค์ประกอบของภาพ

1) จุดเริ่มต้นของเฟรม (SOF) เป็นพารามิเตอร์จุดเริ่มต้นของเฟรม มาร์คเกอร์จุดเริ่มต้นของเฟรมสามารถให้ค่าหนึ่งค่าหรือหลายค่า เพื่อแสดงชนิดของรหัสที่ถูกใช้ในเฟรมนี้โดยเฉพาะ สำหรับเป้าหมายของส่วนนี้คือใช้เฉพาะเข้ารหัสแบบ Baseline DCT Sequential เท่านั้น ค่ามาร์คเกอร์ SOF คือ 0xFFC0

2) ความยาวของเฟรม (Lf) บอกความยาวของเฟรมเฮดเดอร์

3) ความแม่นยำข้อมูล (P) แสดงจำนวนของบิตที่ใช้ในแต่ละค่าสำหรับองค์ประกอบที่สร้างเฟรม สำหรับ Baseline Sequential DCT เท่ากับ 8 เสมอ

4) จำนวนของเส้น (Y) แสดงจำนวนของแถวพิกเซลที่สร้างองค์ประกอบพร้อมด้วยจำนวนของค่าในแนวตั้งเมื่อมีการซับแซมปลิ้ง ถ้าไม่มีการซับแซมปลิ้งส่วนนี้จะแสดงจำนวนของเส้นในแต่ละองค์ประกอบส่วนนี้มีค่าได้ ตั้งแต่ 0 ถึง 65535

5) จำนวนของค่าต่อเส้น (X) กำหนดจำนวนของพิกเซลตามแนวนอนในภาพ ถ้ามีการซับแซมปลิ้งจะกำหนดจำนวนของพิกเซลแนวนอนในองค์ประกอบพร้อมด้วยขนาดตามนอนที่ใหญ่ที่สุด ส่วนนี้สามารถมีค่าได้ตั้งแต่ 1 ถึง 65535

6) จำนวนขององค์ประกอบในเฟรมของภาพ (Nf) บอกจำนวนขององค์ประกอบของภาพในเฟรม โดยปกติจะมี 3 จำนวน ค่าของ Nf จะเท่ากับจำนวนของเซตของพารามิเตอร์ที่แสดงในเฟรมเฮดเดอร์ซึ่งมีส่วนประกอบดังนี้

ตารางที่ 3.14 การกำหนดค่าพารามิเตอร์ของแต่ละองค์ประกอบ

พารามิเตอร์	ขนาด (บิต)	ค่า	รายละเอียด
C_i	8	0-255	ค่าองค์ประกอบ
H_i	4	1-4	แซมปลิ้งแฟกเตอร์ตามแนวนอน
V_i	4	1-4	แซมปลิ้งแฟกเตอร์ตามแนวตั้ง
Tq_i	8	0-3	การกำหนดตารางควอนไทเซชั่น

6.1) ค่าองค์ประกอบ (Ci) ใช้แสดงตัวเลขในแต่ละองค์ประกอบของภาพและลำดับขององค์ประกอบที่ใช้ในเฟรม ตัวเลขเหล่านี้ถูกใช้ในสแกนเฮดเดอร์เพื่อแสดงองค์ประกอบที่ใช้ในสแกน โดยปกติจะแสดงเป็น 0, 1, 2,... ตามลำดับ

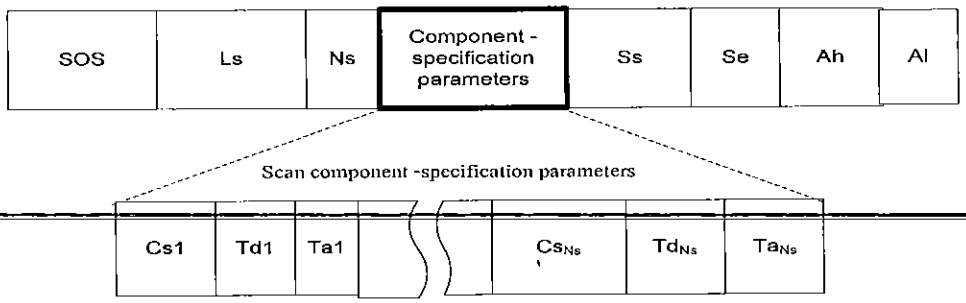
6.2) แซมปลิ้งแฟกเตอร์ตามแนวนอน (Hi) เป็นการกำหนดบล็อกขนาด 8×8 ขององค์ประกอบที่ถูกใช้ในแต่ละ MCU ตามแนวนอน

6.3) แซมปลิ้งแฟกเตอร์ตามแนวตั้ง (Vi) กำหนดบล็อกขนาด8×8ขององค์ประกอบที่ถูกใช้ในแต่ละ MCU ตามแนวตั้ง

6.4) การกำหนดตารางควอนไทเซชั่น (Tqi) กำหนดตารางการควอนไทเซชั่นเพื่อใช้ในการดีควอนไทเซชั่นสัมประสิทธิ์ DCT ถ้าอยู่ในกระบวนถอดรหัส ตารางนี้จะถูกนำมาใช้ ตารางที่เลือกจะไม่สามารถเปลี่ยนจนกระทั่งองค์ประกอบทั้งหมดจะถอดรหัสเรียบร้อย

3.3.2 โครงสร้างสแกนเฮดเดอร์

ในส่วนประกอบของเฟรมจะมีเฟรมเฮดเดอร์แล้วตามด้วยสแกน ซึ่งจะมีสแกนเฮดเดอร์และรหัสเอนโทรปี เป็นส่วนประกอบ ซึ่งมีรายละเอียดดังนี้



รูปที่ 3.7 โครงสร้างของสแกนเฮดเดอร์

1) จุดเริ่มต้นของสแกนเฮดเดอร์ (SOS) เป็นมาร์คเกอร์เริ่มต้นของสแกนเฮดเดอร์

2) ความยาวสแกนเฮดเดอร์ (Ls) กำหนดความยาวของสแกนเฮดเดอร์รวมความยาวฟิลด์แต่ไม่รวมมาร์คเกอร์จุดเริ่มต้นของสแกน

3) จำนวนขององค์ประกอบของภาพ (Ns) กำหนดจำนวนเซตของพารามิเตอร์ที่แสดงองค์ประกอบสแกนรวมไปถึงในสแกนเฮดเดอร์ ในฟิลด์นี้พารามิเตอร์สามารถเปลี่ยนแปลงค่าได้โดยตรง แต่ละเซตประกอบด้วย 3 ส่วนคือ

3.1) ตัวเลือกองค์ประกอบของสแกน (Csj) ฟิลด์นี้ในสแกนเฮดเดอร์อ้างถึงองค์ประกอบของภาพจากเฟรมเฮดเดอร์และต้องเข้ากันอย่างน้อยหนึ่งตัว ดังนั้นลำดับขององค์ประกอบในสแกนเฮดเดอร์ต้องเข้ากับลำดับขององค์ประกอบในเฟรมเฮดเดอร์ ถ้าสแกนมีองค์ประกอบมากกว่าหนึ่งจะมีการซับแซมปลิ้ง ผลรวมของแซมปลิ้งแฟกเตอร์ทั้งแนวนอนและแนวตั้งขององค์ประกอบทั้งหมดจะมีค่าไม่เกิน 10 ตัวอย่างเช่น ถ้า สามองค์ประกอบถูกใช้และสองในสามถูกซับแซมปลิ้งโดยแฟกเตอร์ 2 ทั้งแนวตั้งและแนวนอน ผลรวมคือ 1×1+2×2+2×2 เท่ากับ 9 ถ้าซับแซมปลิ้งแฟกเตอร์มากกว่าสอง องค์ประกอบเหล่านั้นจะถูกเข้ารหัสแยกกัน

3.2) ตารางรหัสเอนโทรปี DC (Tdj) เลือกหนึ่งในสี่ของตารางรหัสเอนโทรปีของ DC ที่จะถอดรหัสสัมประสิทธิ์ DC สำหรับองค์ประกอบที่อยู่ในฟิลด์ Csj

3.3) ตารางรหัสเอนโทรปี AC (Taj) ในฟิลด์นี้เลือกหนึ่งในสี่ของตารางรหัสเอนโทรปี AC ที่จะถอดรหัสสัมประสิทธิ์ AC สำหรับองค์ประกอบที่อยู่ในฟิลด์ Csj

4) การเลือกจุดเริ่มต้นของสเปกตรัม (Ss) เป็นพารามิเตอร์ที่บอกสัมประสิทธิ์ DCT ตัวแรกของแต่ละบล็อกที่จะเข้ารหัส จะมีค่าเป็น 0 สำหรับ Bascline Sequential Mode

5) การเลือกจุดสุดท้ายของสเปกตรัม (Se) ฟิลด์นี้กำหนดสัมประสิทธิ์ DCT ตัวสุดท้ายในแต่ละบล็อกที่จะเข้ารหัส สำหรับ Baseline Sequential กำหนดค่า 63 เป็นค่าสุดท้ายของบล็อก

6) Successive Approximation Bit Position High (Ah) ลำดับความต่อเนื่องที่เข้าใกล้บิตสูง ฟิลด์นี้เป็น 0 สำหรับการประมวลผลแบบ Baseline Sequential DCT

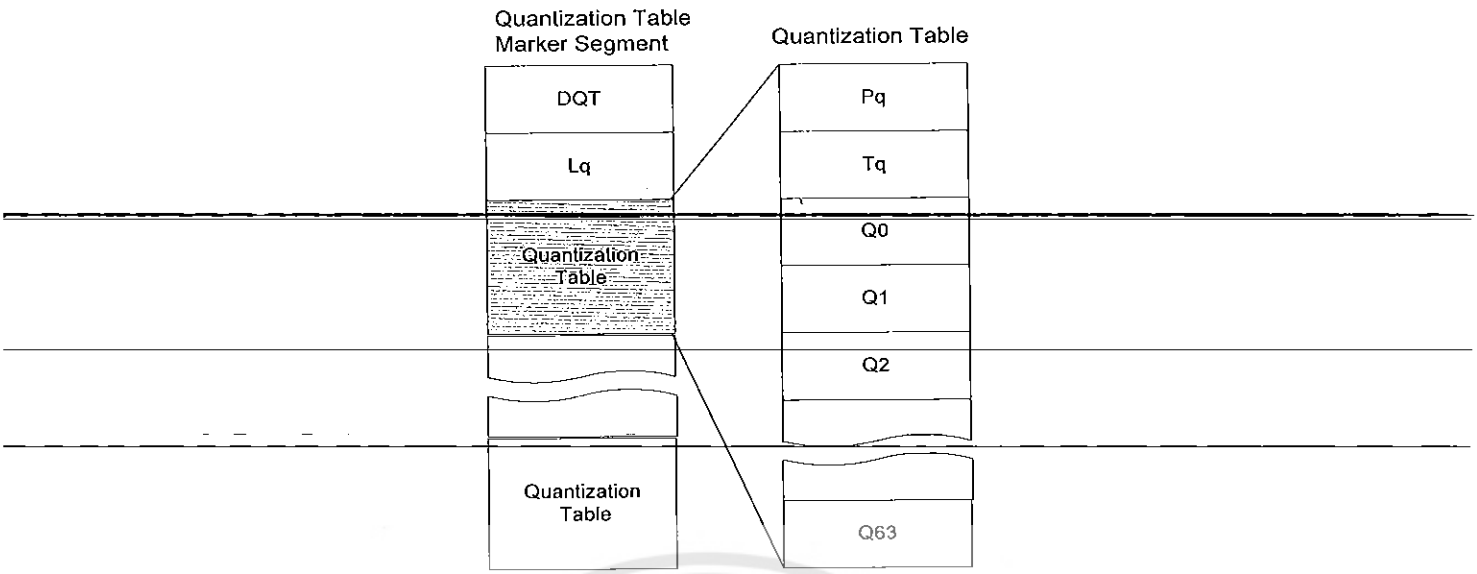
7) Successive Approximation Bit Position Low (Al) ลำดับความต่อเนื่องที่เข้าใกล้บิตสูง ฟิลด์นี้เป็น 0 สำหรับการประมวลผลแบบ Baseline Sequential DCT

ตารางที่ 3.15 ส่วนประกอบของสแกนเฮดเดอร์

พารามิเตอร์	ขนาด (บิต)	ค่า	รายละเอียด
SOS	16	0xFFDA	จุดเริ่มต้นของสแกน
Ls	16	2Ns+6	ความยาวสแกนเฮดเดอร์
Ns	8	1-4	จำนวนองค์ประกอบของภาพ
Csj	8	0-255	ตัวเลือกสแกนองค์ประกอบ
Tdj	4	0-1	ตาราง DC
Taj	4	0-1	ตาราง AC
Ss	8	0	จุดเริ่มต้นของสเปกตรัม
Se	8	63	จุดสุดท้ายของสเปกตรัม
Ah	4	0	ความต่อเนื่องที่เข้าใกล้บิตสูง
Al	4	0	ความต่อเนื่องที่เข้าใกล้บิตต่ำ

3.3.3 โครงสร้างตารางค่าควอนไทเซชั่น

ตารางควอนไซเซชั่นถูกกำหนดในมาร์คเกอร์เริ่มต้นพร้อมกับกำหนดมาร์คเกอร์ของตารางควอนไทเซชั่น (defined quantization table : DQT) เท่ากับ 0xFFD8 แล้วตามด้วยมาร์คเกอร์ (Lq) กำหนดความยาวของตาราง จากนั้นคือกลุ่มของฟิลด์ที่กำหนดในตารางควอนไทเซชั่น ดังรูป 3.8 ในฟิลด์แรกของตารางควอนไทเซชั่น (Pq) กำหนดความแม่นยำของสัมประสิทธิ์การควอนไทเซชั่น ถ้าฟิลด์นี้มีค่าเป็น 0 จะใช้ สัมประสิทธิ์การควอนไทเซชั่น 8 บิต ถ้ามีค่าเป็น 1 จะใช้สัมประสิทธิ์การควอนไทเซชั่น 16 บิต สำหรับ Baseline Sequential DCT Mode ใช้สัมประสิทธิ์การควอนไทเซชั่น 8 บิต จากนั้นเป็นฟิลด์กำหนดตารางค่าควอนไทเซชั่น (Tq) ที่จะนำมาใช้ ส่วนที่เหลือของตารางจะประกอบด้วยสัมประสิทธิ์การควอนไทเซชั่น 64 ตัว ตารางที่ 3.16 สรุปฟิลด์ภายในตารางค่าการควอนไทเซชั่นและค่าที่ใช้ในแต่ละฟิลด์



รูปที่ 3.8 ส่วนประกอบของตารางควอนไทเซชั่น

ตารางที่ 3.16 ฟิลด์ตารางการควอนไทเซชั่น

พารามิเตอร์	ขนาด(บิต)	ค่า	รายละเอียด
DQT	16	0xFFD8	มาร์คเกอร์กำหนดตารางการควอนไทเซชั่น
Lq	16	65n+2	ความยาวตารางการควอนไทเซชั่น
Pq	4	0	ค่าความแม่นยำตารางการควอนไทเซชั่น
Tq	4	0-3	กำหนดตารางการควอนไทเซชั่น
Qk	8	1-255	ส่วนประกอบในตารางการควอนไทเซชั่น

เมื่อ n คือจำนวนของตารางการควอนไทเซชั่นที่กำหนด

3.3.4 โครงสร้างตารางฮัฟแมน

ตารางฮัฟแมนจะเริ่มด้วยฟิลด์แรกคือมาร์คเกอร์ของตารางฮัฟแมน DHT (Define Huffman Table: DHT) 16 บิต 0xFFC4 ตามด้วยฟิลด์ข่องความยาวของตารางฮัฟแมน (Lh) ที่กำหนดความยาวทั้งหมดของส่วนนี้ จากนั้นตามด้วยตารางค่าฮัฟแมนทั้ง 4 ตาราง แต่ละตารางจะสร้างหลายๆ ฟิลด์ ดังรูป 3.9 ซึ่งมีส่วนประกอบดังนี้

1) ตารางคลาส (Tc) ถ้าเป็น 0 เลือกตาราง DC ถ้าเป็น 1 เลือกตาราง AC

2) กำหนดตารางฮัฟแมน (Th) เลือกค่าหนึ่งใน 4 ของตารางฮัฟแมนที่จะเข้ารหัส

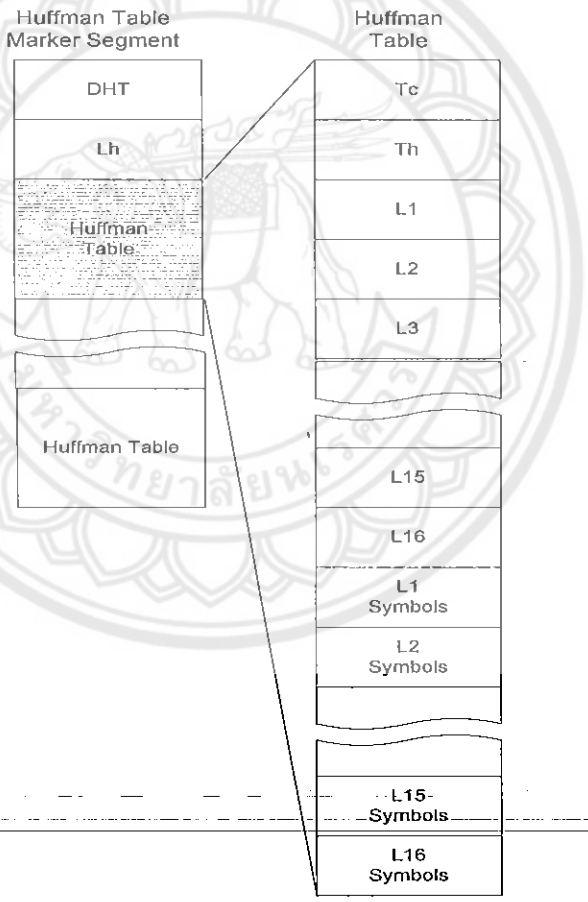
3) ความยาวของรหัสฮัฟแมน (Li) ในตารางฮัฟแมนจะมี 16 ฟิลด์ตามค่า JPEG โดยฟิลด์แรกจะมี 1 บิต ฟิลด์ที่สอง 2 บิต ตามลำดับ

4) ค่าของรหัสฮัฟแมน (Vij) จากตารางที่กำหนด ซึ่งแต่ละค่าถูกกำหนดโดยการเข้ารหัสฮัฟแมน ตารางที่3.17 สรุปการสร้างตารางฮัฟแมนและค่าแต่ละฟิลด์

ตารางที่ 3.17 โครงสร้างตารางฮัฟแมน

พารามิเตอร์	ขนาด(บิต)	ค่า	รายละเอียด
DHT	16	0xFFC4	มาร์คเกอร์กำหนดตารางฮัฟแมน
Lh	16	*	กำหนดความยาวตารางฮัฟแมน
Tc	4	0,1	ตารางคลาส 0 = DC 1 =AC
Th	4	0,1	กำหนดตารางฮัฟแมน
Li	8	0-255	ความยาวรหัสฮัฟแมน
Vij	8	0-255	ค่ารหัสฮัฟแมน

หมายเหตุ * ค่านี้ขึ้นอยู่กับจำนวนของตารางที่กำหนดและจำนวนของรหัสที่กำหนดในแต่ละความยาว



รูปที่ 3.9 ส่วนประกอบของมาร์คเกอร์ตารางฮัฟแมน

3.3.5 โครงสร้างการกำหนด Restart Interval

การทำให้รหัสเอนโทรปีสั้น เพื่อความสะดวกต่อการหาข้อผิดพลาดและเพิ่มความถูกต้อง มาร์คเกอร์ที่สร้างขึ้นมีด้วยกัน 3 ฟิลด์ ฟิลด์แรกเป็นจุดเริ่มมาร์คเกอร์ (Define Restart Interval Marker:DRI) มีค่าเป็น 0xFFDD ฟิลด์ที่สองเป็นความยาวของมาร์คเกอร์ (Lr) มีค่าเป็น 4 เสมอและฟิลด์ที่สาม (Ri) มีค่า 16 บิต ซึ่งกำหนดโดยจำนวนของ MCU ใน Restart Interval การกำหนด Restart Interval เพื่อไม่ให้สับสนจะอยู่ในส่วนของการเข้ารหัสบิตสตรีม Restart Interval สามารถกำหนดได้โดยพิจารณาจำนวนของ MCU ที่ถูกเข้ารหัสก่อนมาร์คเกอร์ถัดมาที่จะเข้าไปในบิตสตรีม ดังแสดงในตาราง

ตารางที่ 3.18 การเริ่มมาร์คเกอร์ใหม่

พารามิเตอร์	ขนาด(บิต)	ค่า	รายละเอียด
DRI	16	0xFFDD	กำหนดจุดเริ่มมาร์คเกอร์ใหม่
Lr	16	4	ความยาวของ DRI
Ri	16	0-65535	จำนวนของ MCU ใน Restart Interval

3.3.6 ส่วนของข้อมูล

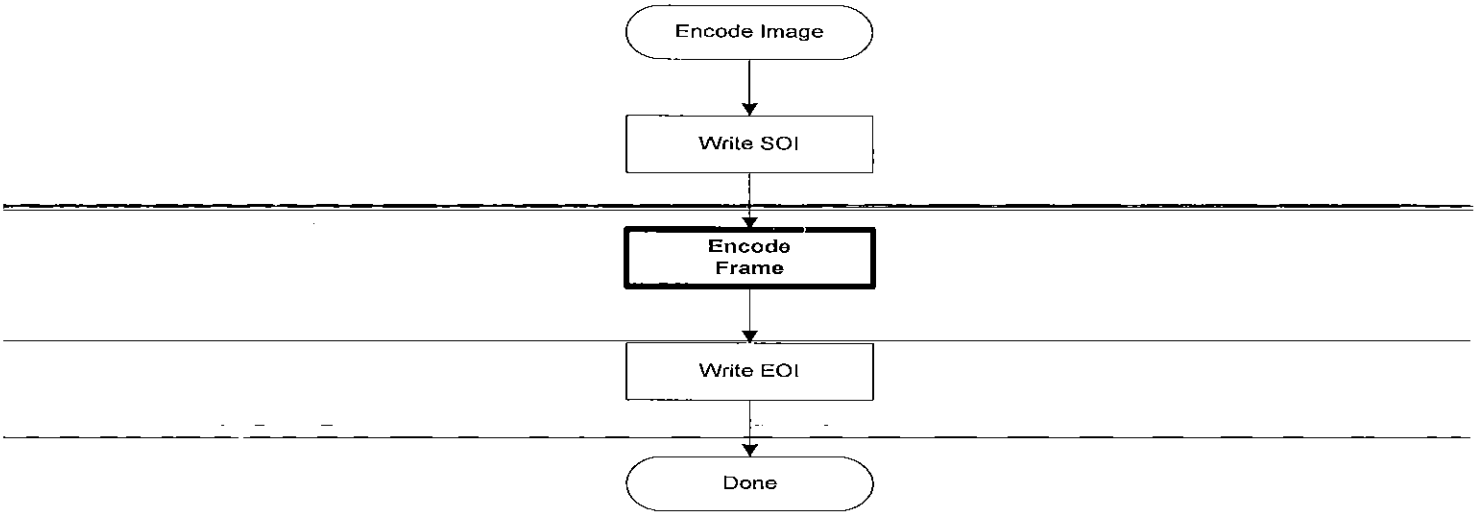
มาร์คเกอร์ข้อมูล (Application Data Marker :APP,0xFFE0,0xFFEF) ใน JPEG ข้อมูลที่อยู่ในบิตสตรีมของการเข้ารหัสภาพมีด้วยกัน 3 ฟิลด์ ในฟิลด์แรกเป็นมาร์คเกอร์แสดงค่าข้อมูล ฟิลด์ที่สองเป็นความยาวของมาร์คเกอร์ในฟิลด์แรก (Lp) ฟิลด์ที่สามเป็นข้อมูล (Ap) ดังตารางที่ 3.19

ตารางที่ 3.19 ส่วนของข้อมูล

พารามิเตอร์	ขนาด(บิต)	ค่า	รายละเอียด
APP	16	0xFFE0 ถึง 0xFFEF	ข้อมูลมาร์คเกอร์
Lp	16	2-65535	ความยาวส่วน APP
Ap	8	0-255	ข้อมูล

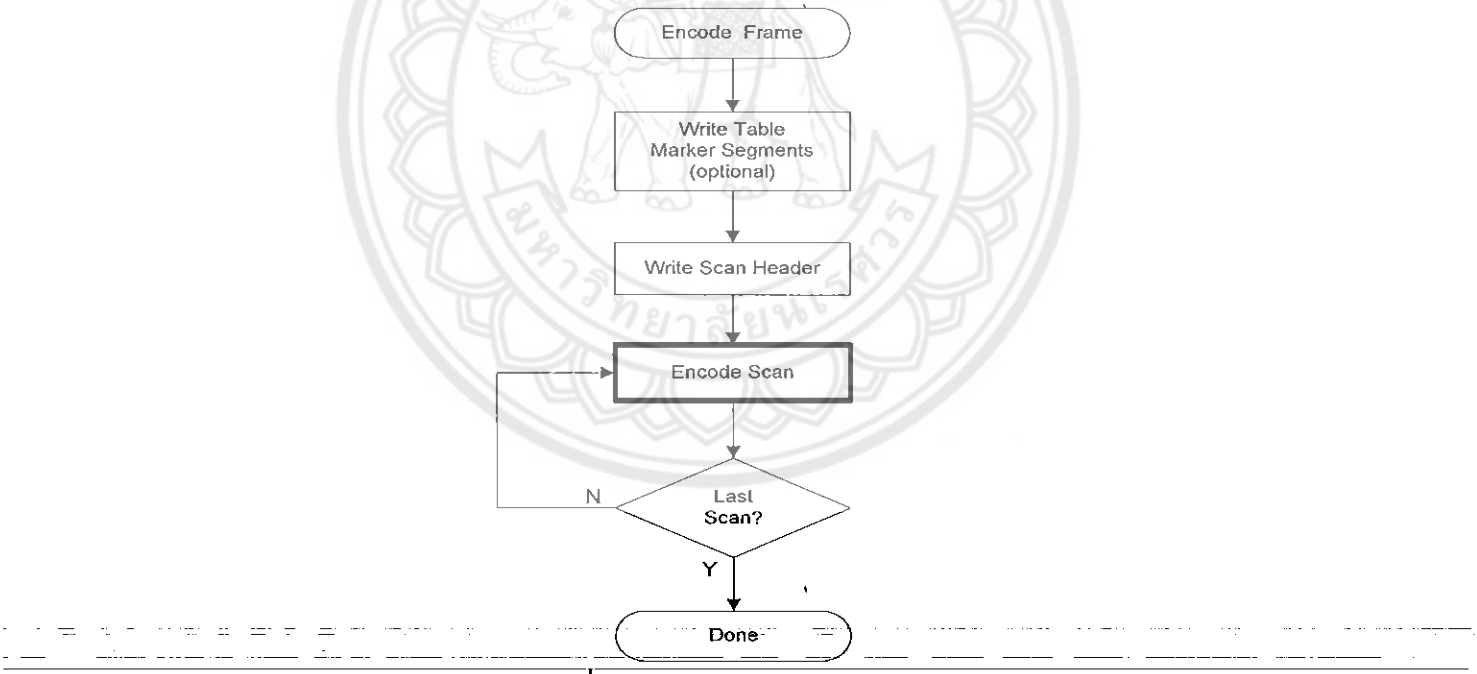
3.4 สรุปโฟล์วชาร์ทของขั้นตอนบีบอัดข้อมูล JPEG

การอธิบายขั้นตอนสำหรับการเข้ารหัสเฟรมโดยใช้โฟล์วชาร์ทในการอธิบาย ดังรูปที่ 3.8 เป็นขั้นตอนการเขียนไฟล์ภาพโดยรวม ซึ่งมีการเขียนมาร์คเกอร์จุดเริ่มต้นของภาพและเข้ารหัสเฟรม แล้วตามด้วยมาร์คเกอร์จุดสิ้นสุดภาพ



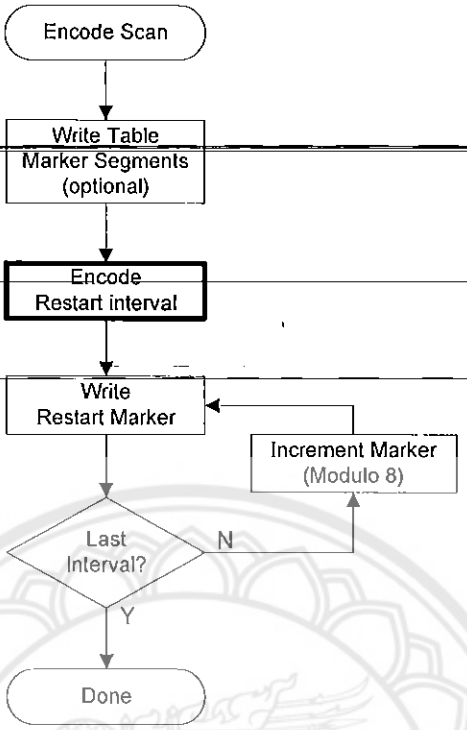
รูปที่ 3.10 ขั้นตอนการเข้ารหัสภาพ

ในส่วนของการเข้ารหัสเฟรม เขียนมาร์กเกอร์ตามโครงสร้างของเฟรมเฮดเดอร์แล้วตามด้วยการเขียนสแกนเฮดเดอร์ จากนั้นเข้ารหัสสแกน ดังแสดงในรูป 3.11



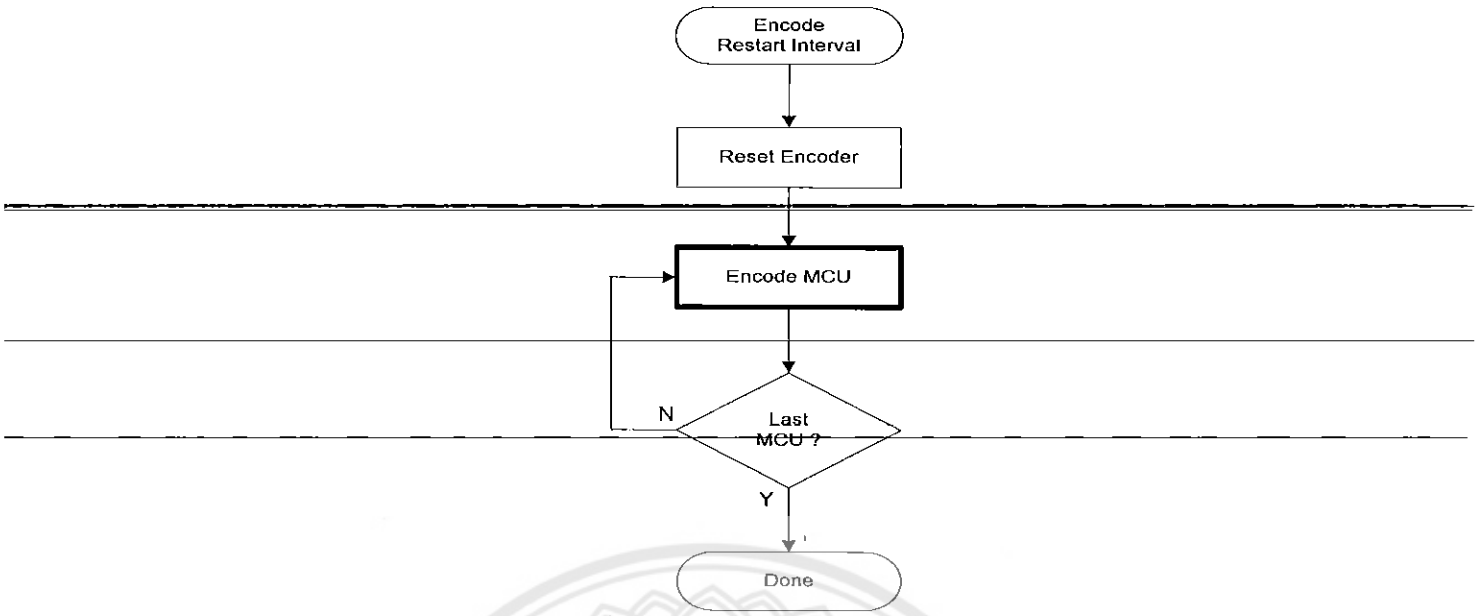
รูปที่ 3.11 การเข้ารหัสเฟรม

ในการเข้ารหัสสแกน ทำการเขียนโครงสร้างของสแกน แล้วเข้ารหัส Restart Interval Restart Interval สร้างด้วย MCU หนึ่งหรือมากกว่านั้น จุด Restart สามารถกำหนดในตารางส่วนที่เพิ่มขึ้น ที่เกิดขึ้นก่อนเฟรมเฮดเดอร์หรือก่อนสแกนเฮดเดอร์ ถ้าสแกนไม่ได้แยกเป็นหลายส่วน สแกนจะมีจุด Restart อันเดียว ในกรณีอื่นๆ สแกนโดยปกติจะตามด้วยมาร์คเกอร์ Restart Interval เดียว ดังรูป

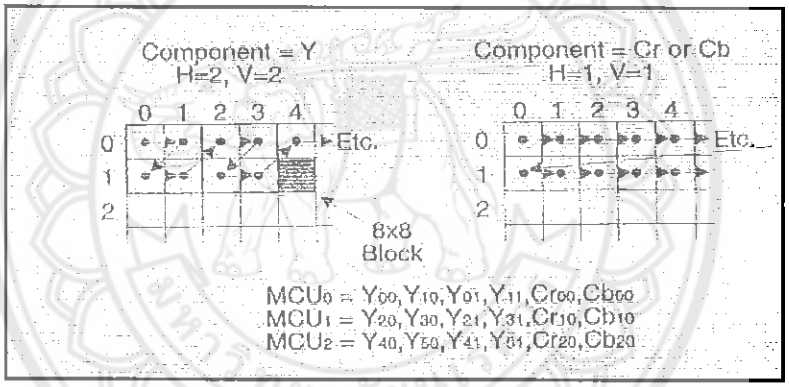


รูปที่ 3.12 การเข้ารหัสในส่วนของสแกน

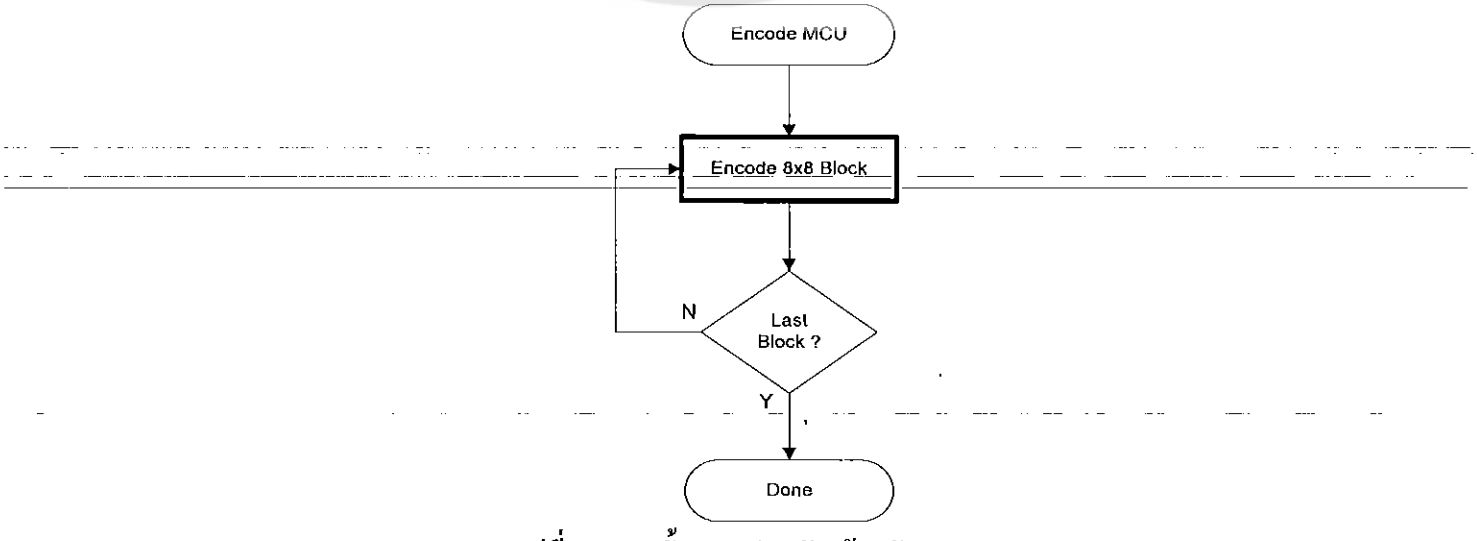
การเข้ารหัส Restart Interval ประกอบด้วยของการเข้ารหัสจำนวน MCU ที่กำหนดใน Restart Interval ดังรูปที่ 3.13 สิ่งที่สำคัญคือการรีเซ็ต นั่นคือในระหว่างขั้นตอนการถอดรหัสทุกครั้งจะจบด้วย Restart Interval เพื่อไม่ให้ความผิดพลาดที่อาจเกิดขึ้นในการเข้ารหัสที่ผ่านมาแล้ว เกิดขึ้นกับการเข้ารหัสส่วนต่อมา โดยทั่วไปการเข้ารหัส MCU จะเข้ารหัสจากซ้ายไปขวาและจากบนลงล่าง ในแต่ละบล็อกของ MCUจะมีองค์ประกอบแตกต่างกัน ถ้ามีมากกว่าหนึ่งองค์ประกอบ จะดำเนินการตามลำดับที่กำหนดในเฟรมเฮดเดอร์ ตัวอย่างเช่น ถ้าภาพที่สร้างมี 3 องค์ประกอบ คือหนึ่งลูมิแนนซ์และสองโครมิแนนซ์ องค์ประกอบโครมิแนนซ์จะถูกซับแซมปลิ้ง ดังรูปที่ 3.14 แสดง ลำดับการเข้ารหัสของ MCU และรูปที่ 3.15 แสดงขั้นการเข้ารหัส MCU



รูปที่ 3.13 ขั้นตอนสำหรับเข้ารหัส Restart Interval



รูปที่ 3.14 ลำดับขององค์ประกอบภาพใน MCU



รูปที่ 3.15 ขั้นตอนสำหรับเข้ารหัส MCU

ขั้นตอนการเข้ารหัสของMCUมีดังนี้

1. นำ MCU มาผ่านการแปลง FDCT ทีละบล็อกขนาด 8×8 พิกเซล

2. หารด้วยตารางควอนไทเซชั่นที่กำหนดไว้

3. หาผลต่างของสัมประสิทธิ์ DC ตัวปัจจุบันกับสัมประสิทธิ์ DC ของบล็อกที่ผ่านมา เพื่อใช้ในการเข้ารหัสสัมประสิทธิ์ DC

4. เข้ารหัสผลต่างโดยใช้ตารางฮัฟแมนของ DC จากนั้นเข้ารหัสสัมประสิทธิ์ AC โดยใช้ตารางฮัฟแมนของ AC ที่อยู่ในส่วนของเฮดเดอร์

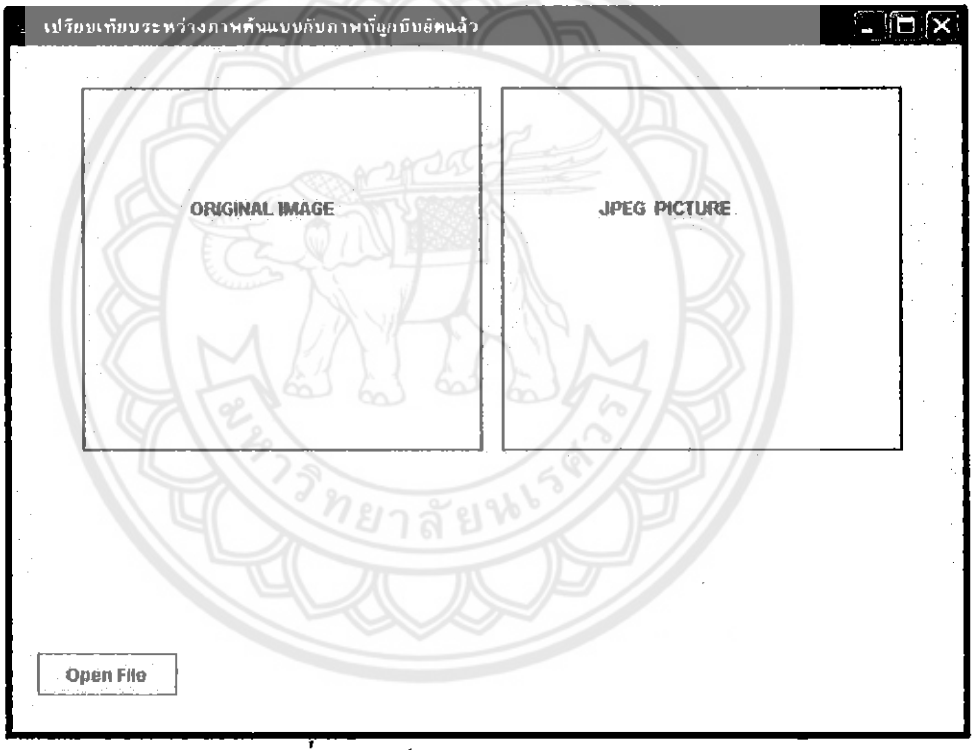
บทที่ 4

ผลการทดลอง

4.1 การทำงานของโปรแกรม

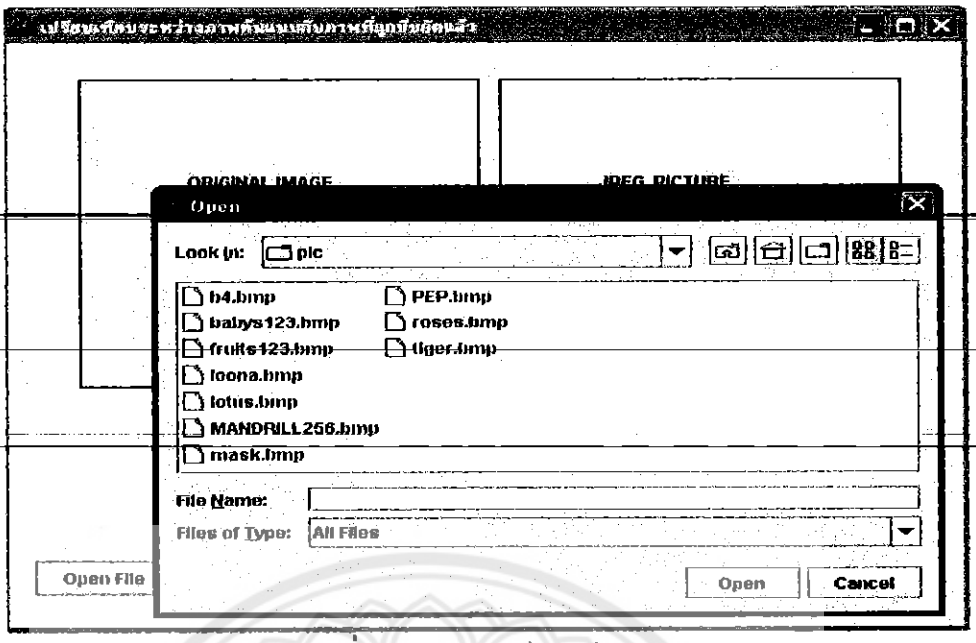
ในส่วนของหน้าจอแสดงผลของโปรแกรมได้แบ่งออก ดังนี้ คือ

ส่วนที่ 1 หน้าแรกของโปรแกรมจะแสดงตำแหน่งของภาพต้นแบบ ภาพที่ถูกบีบอัดแล้วและ มีปุ่ม "Open File" สำหรับเปิดไฟล์ภาพ



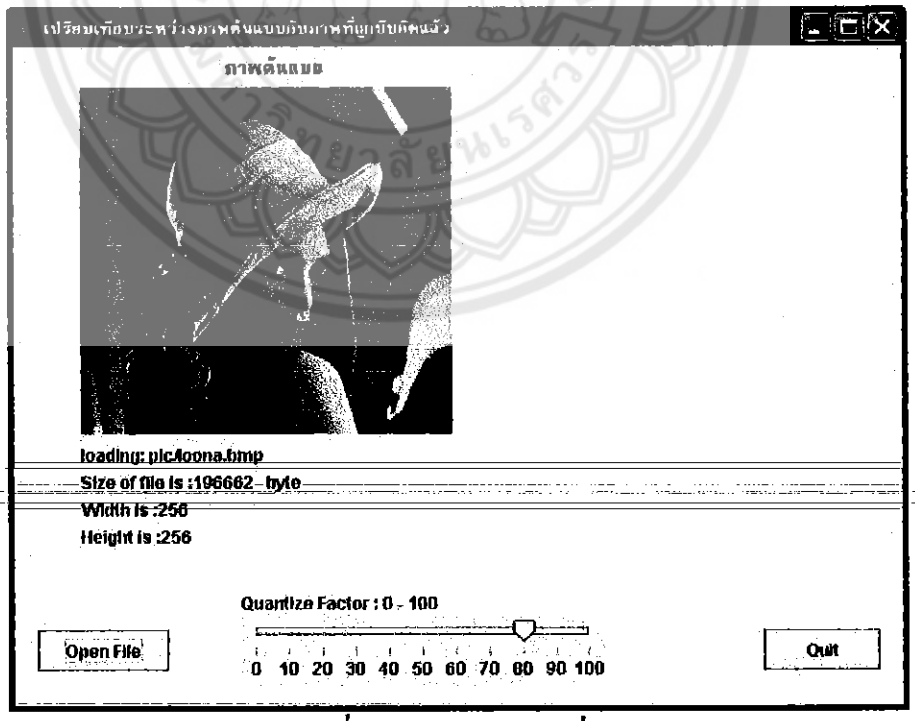
รูปที่ 4.1 หน้าจอแสดงผลของโปรแกรม

ส่วนที่ 2 เมื่อมีการใช้เมาส์กดปุ่ม "Open File" จะมองเห็นไฟล์ภาพที่อยู่ใน โฟล์ดเดอร์(Folder) ที่ชื่อว่า Pic ซึ่งได้ กำหนดไว้เป็นค่าเริ่มต้น สามารถเลือกภาพต้นแบบโดยใช้เมาส์คลิ๊ก หรืออาจพิมพ์ชื่อที่เห็นอยู่ในโฟล์ดเดอร์ก็ได้



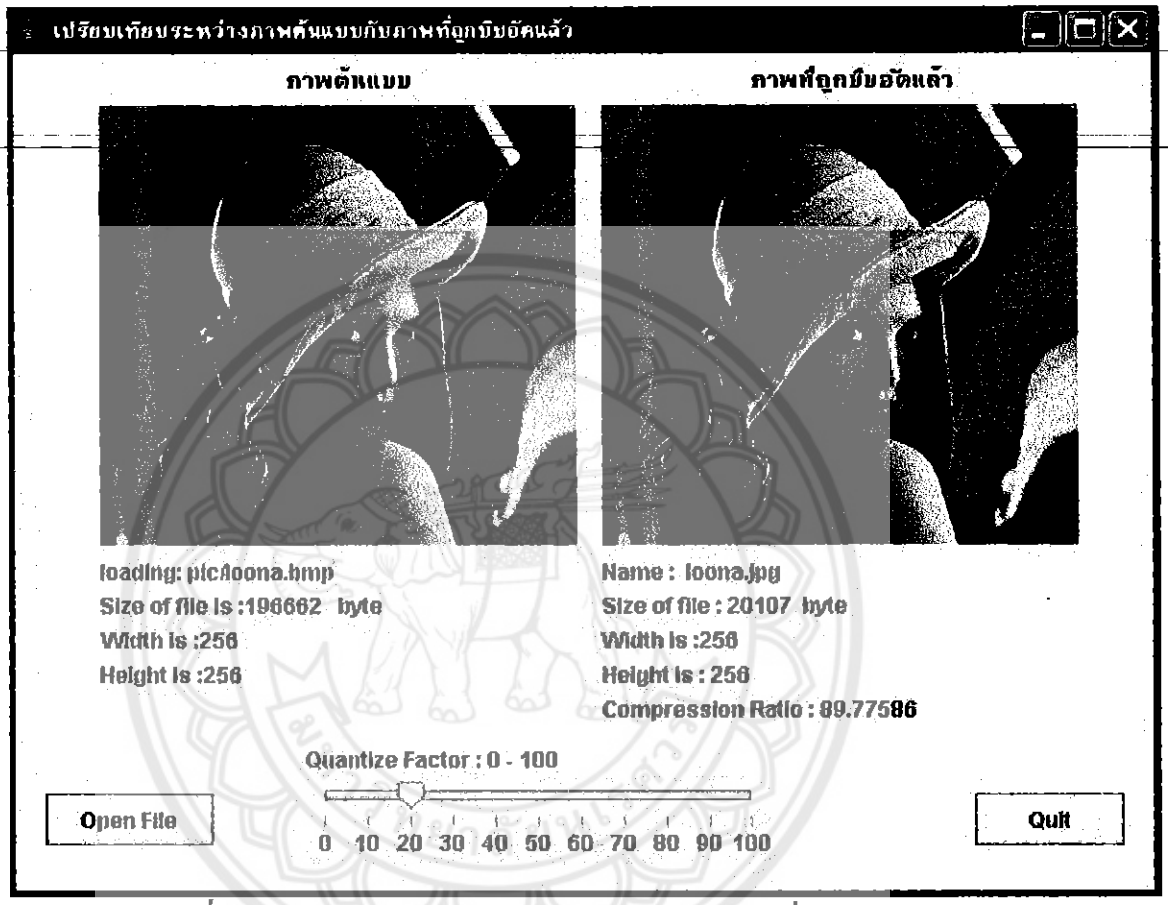
รูปที่ 4.2 เลือกไฟล์ภาพต้นแบบ

ส่วนที่ 3 เมื่อเลือกภาพต้นแบบได้แล้ว โปรแกรมจะแสดงภาพต้นแบบ ชื่อภาพ **ขนาดไฟล์ภาพ** ความกว้างและ ความสูงของภาพ และมีสไลด์เดอร์บาร์ (Slider Bar) สำหรับเลื่อน **เพิ่ม หรือ ลดค่า** Quantize Factor ซึ่งกำหนดค่าให้ ตั้งแต่ 0 - 100



รูปที่ 4.3 ภาพต้นแบบที่ได้

ส่วนที่ 4 เมื่อเลื่อนสไลด์เดอร์บาร์แล้วจะได้ภาพที่ถูกบีบอัดแล้วแสดงขึ้นมาคู่กับภาพต้นแบบ และอัตราส่วนการบีบอัดข้อมูลภาพ ซึ่งทำให้สามารถเปรียบเทียบความแตกต่างของภาพทั้งสองได้ ถ้าต้องการเปลี่ยนภาพใหม่ สามารถกดปุ่ม "Open File" เพื่อแสดงภาพต้นแบบใหม่ได้และเมื่อต้องการสิ้นสุดการทำงานของโปรแกรม ให้กดปุ่ม Quit เพื่อจบการทำงาน



รูปที่ 4.4 ภาพเปรียบเทียบระหว่างภาพต้นแบบและภาพที่ถูกบีบอัดแล้ว

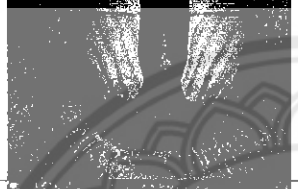
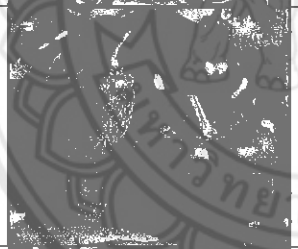
4.2 การบีบอัดข้อมูลภาพที่ ค่า Quantize Factor ต่างๆ

ในการทดสอบได้นำภาพต้นแบบที่เป็นภาพสี RGB ซึ่งเป็นไฟล์ภาพBMP ชนิด 24 บิต/พิกเซล ขนาด 256×256 พิกเซล จำนวน 10 ภาพ มาทดสอบในโปรแกรมที่ได้ออกแบบไว้ ที่ระดับค่า Quantize Factor ต่าง ๆ เพื่อศึกษาอัตราส่วนการบีบอัดข้อมูลและคุณภาพของภาพที่ได้ ซึ่งได้ผลการทดลองดังตารางที่ 4.1

อัตราส่วนการบีบอัดข้อมูลภาพ = ขนาดภาพต้นแบบ/ ขนาดของภาพที่ถูกบีบอัดแล้ว

ตารางที่ 4.1 ผลการทดสอบการบีบอัดข้อมูลภาพบิตแมพ

รูปภาพ	ตัวอย่างรูปภาพ	Quantize Factor (Q.F.)	ขนาดของภาพที่ถูกบีบอัดแล้ว(ไบท์)	อัตราส่วนการบีบอัดข้อมูล
Bacelona.bmp		0	103,131	1.90
ขนาด		50	14,342	13.71
196,662		100	2,892	68.00
ไบท์				
baby123.bmp		0	120,307	1.63
ขนาด		50	14,124	13.92
196,662		100	2,959	66.46
ไบท์				
fruits123.bmp		0	114,488	1.71
ขนาด		50	12,439	15.81
196,662		100	2,785	70.61
ไบท์				
Lena.bmp		0	124,154	1.58
ขนาด		50	11,381	17.28
196,662		100	2,883	68.21
ไบท์				

รูปภาพ	ตัวอย่างรูปภาพ	Quantize Factor (Q.F.)	ขนาดของภาพที่ถูกบีบอัดแล้ว (ไบท์)	อัตราส่วนการบีบอัดข้อมูล
lotus.bmp		0	129,926	1.51
ขนาด		50	16,727	11.75
196,662 ไบท์		100	3,243	60.64
mandrill.bmp		0	151,189	1.30
ขนาด		50	16,204	12.13
196,662 ไบท์		100	2,762	71.20
mask.bmp		0	162,515	1.21
ขนาด		50	22,601	8.70
196,662 ไบท์		100	4,074	48.27
pepper.bmp		0	132,794	1.48
ขนาด		50	13,083	15.03
196,662 ไบท์		100	2,995	65.66
roses.bmp		0	128,746	1.52
ขนาด		50	12,355	15.91
196,662 ไบท์		100	2,820	69.73
tiger.bmp		0	90,797	2.18
ขนาด		50	14,183	13.86
196,662 ไบท์		100	3,049	64.56

4.3 ตัวอย่างภาพที่ใช้ในการทดลอง

เมื่อนำภาพ Lena, Mandrill และ pepper มาทดลองที่ระดับค่า Q.F. เท่ากับ 0 ,50 และ 100 จะได้ภาพที่ถูกบีบอัดแล้วดังรูปที่ 4.5 ,4.6 และ 4.7 ตามลำดับ

(a) ภาพต้นแบบ Lena

(b) เมื่อถูกบีบอัดโดยใช้ Q.F. =0

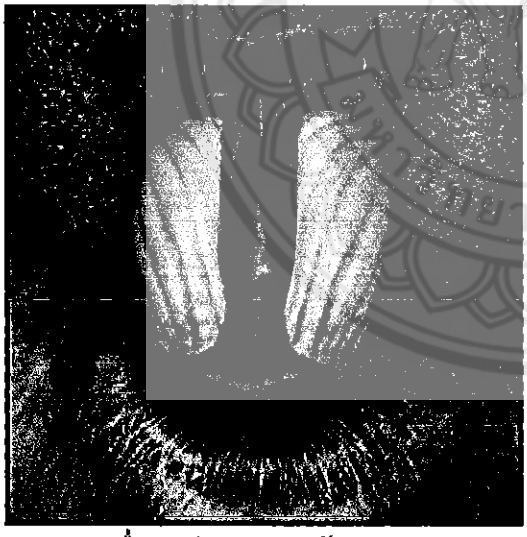
(c) เมื่อถูกบีบอัดโดยใช้ Q.F. =50

(d) เมื่อถูกบีบอัดโดยใช้ Q.F.=100

รูปที่ 4.5 การบีบอัดข้อมูลภาพ Lena ที่ระดับ Q.F. ต่างๆ

(a) ภาพต้นแบบ Mandrill

(b) เมื่อถูกบีบอัดโดยใช้ Q.F. =0

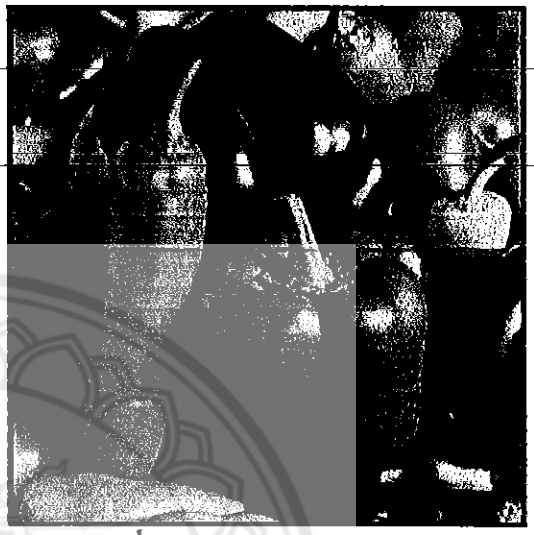


(c) เมื่อถูกบีบอัดโดยใช้ Q.F. =50

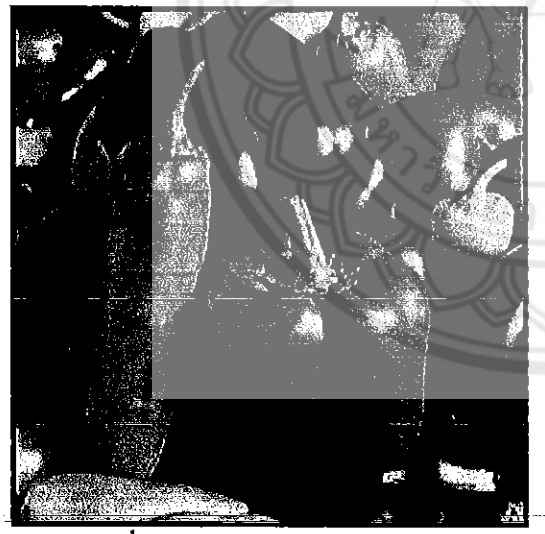
(d) เมื่อถูกบีบอัดโดยใช้ Q.F. =100

รูปที่ 4.6 รูป Mandrill ที่ระดับ Q.F. ต่างๆ

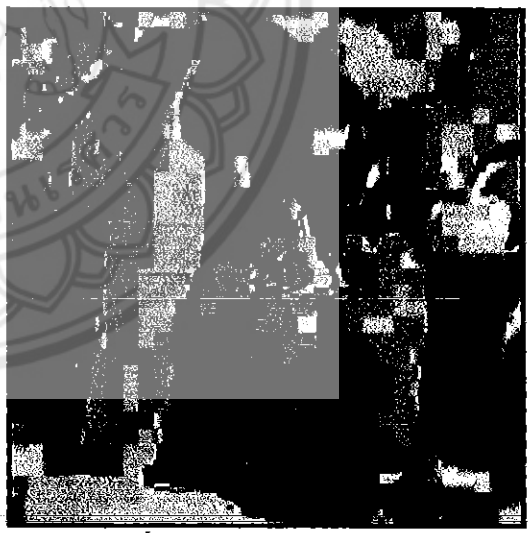
(a) ภาพต้นแบบ pepper



(b) เมื่อถูกบีบอัดโดยใช้ Q.F. =0



(c) เมื่อถูกบีบอัดโดยใช้ Q.F. =50

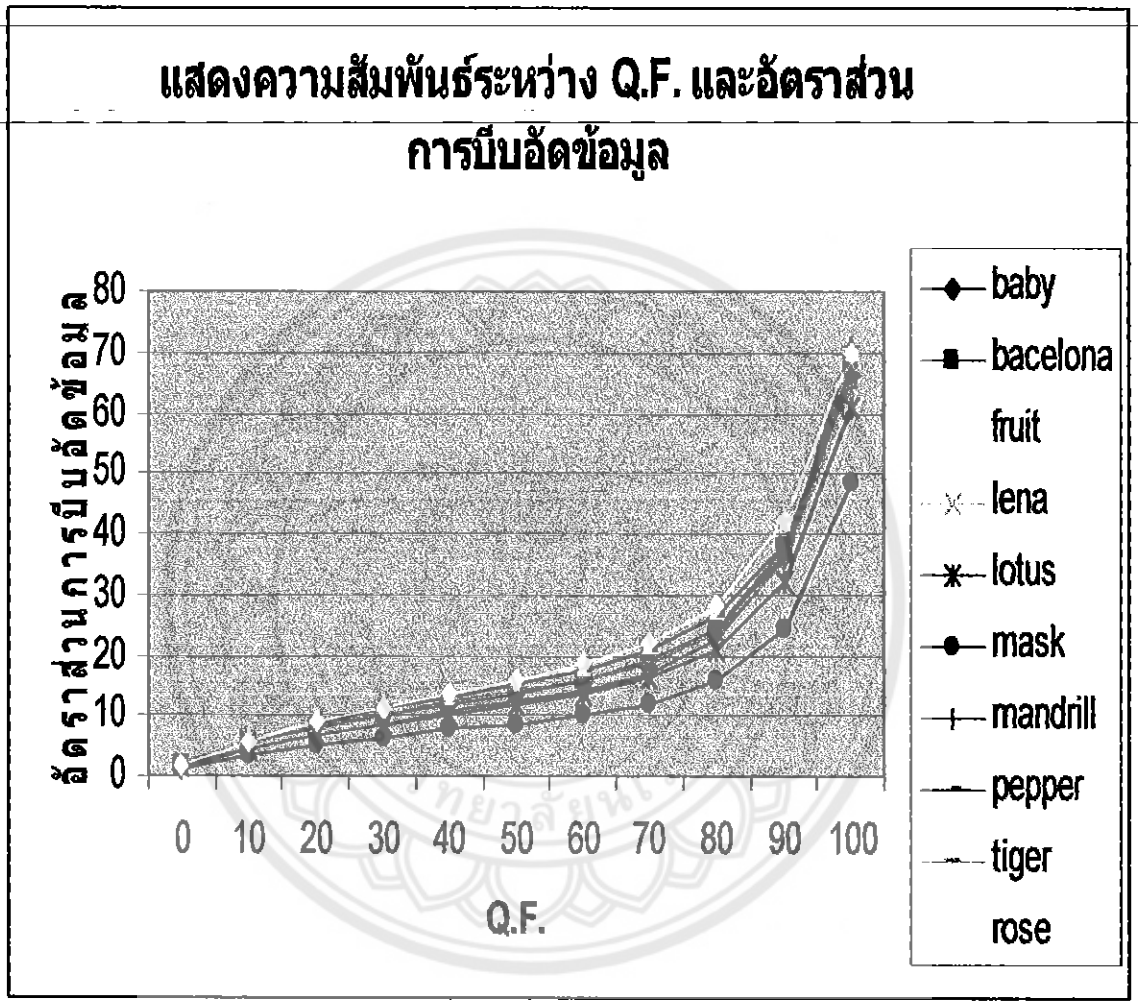


(d) เมื่อถูกบีบอัดโดยใช้ Q.F. =100

รูปที่ 4.7 รูป PEPPER ที่ระดับ Q.F. ต่างๆ

4.4 กราฟแสดงความสัมพันธ์ระหว่างค่า Q.F.และอัตราส่วนการบีบอัดข้อมูล

จากการทดลองการบีบอัดข้อมูลภาพที่ได้เมื่อนำข้อมูลมาเขียนแสดงความสัมพันธ์ระหว่างค่า Q.F.และ อัตราส่วนการบีบอัดข้อมูล ในรูปแบบกราฟเส้น จะได้กราฟแสดงความสัมพันธ์ดังรูปที่ 4.8



รูปที่ 4.8 กราฟความสัมพันธ์ระหว่าง ค่า Q.F.และ อัตราส่วนการบีบอัดข้อมูล

จากกราฟที่ได้ สามารถสรุปได้ว่า เมื่อค่า Q.F. มีค่าเพิ่มขึ้น อัตราส่วนการบีบอัดข้อมูลภาพจะเพิ่มขึ้นด้วย แต่คุณภาพของภาพที่ได้จะลดลง เมื่อมี Q.F. ต่ำจะได้ภาพเหมือนภาพต้นแบบ และเมื่อเปลี่ยน Q.F. สูงจะได้ภาพที่ถูกบีบอัดแล้วมีลักษณะเป็นบล็อก ดังตัวอย่างของภาพในหัวข้อ 4.3

บทที่ 5
สรุปผลการทดลองและข้อเสนอแนะ

5.1 สรุปผลการทดลอง

จากการศึกษามาตรฐานการบีบอัดข้อมูลภาพ JPEG ชนิด Sequential Baseline System ได้ออกแบบและพัฒนาโปรแกรมด้วยภาษาจาวา ภาพต้นแบบที่ใช้เป็นภาพสี RGB ชนิด 24 บิต/พิกเซล ขนาด 256×256 พิกเซล โดยโปรแกรมสามารถเปลี่ยนค่า Q.F. ได้หลายระดับตั้งแต่ 0-100 ซึ่งกำหนดให้เพิ่มค่าได้ครั้งละ 10 ได้ผลการทดสอบดังนี้

ภาพ Lena.bmp ที่ค่า Q.F. = 0 จะได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 1.58
ที่ค่า Q.F. = 50 จะได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 17.28
และที่ค่า Q.F. = 100 ได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 68.21

ภาพ pepper.bmp ที่ค่า Q.F. = 0 ได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 1.48
ที่ค่า Q.F. = 50 ได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 15.03
และที่ค่า Q.F. = 100 ได้อัตราส่วนการบีบอัดข้อมูล เท่ากับ 65.66

รวมทั้งภาพอื่นๆก็ได้ผลทดลองในแนวเดียวกันนี้ ทำให้สามารถสรุปได้ว่า เมื่อใช้ Q. F. ต่ำจะได้อัตราส่วนการบีบอัดข้อมูลต่ำ และเมื่อใช้ Q.F. สูง จะได้อัตราส่วนการบีบอัดข้อมูลที่สูงเช่นกันและคุณภาพของภาพที่ถูกบีบอัดแล้วจะลดลง คือเมื่อใช้ Q.F. ต่ำจะได้ภาพที่เหมือนกับต้นแบบ เมื่อเปลี่ยนมาใช้ Q.F. สูงภาพที่ได้มองเห็นเป็นบล็อก ซึ่งไม่สามารถนำมาใช้ประโยชน์ได้

5.2 ข้อเสนอแนะ

1. ในส่วนของตัวโปรแกรมที่ได้ออกแบบนี้ ยังมีข้อจำกัดในเรื่องชนิดของภาพซึ่งใช้กับภาพสี RGB ขนาด 256×256 พิกเซล ถ้ามีผู้สนใจสามารถนำไปพัฒนาต่อได้
2. ในด้านคุณสมบัติของโปรแกรม ยังขาดในส่วนของการบันทึกภาพในแต่ละช่วง Q.F.
3. สำหรับผู้ที่สนใจสามารถเป็นพื้นฐานสำหรับผู้ที่ศึกษาการบีบอัดข้อมูลภาพนิ่ง JPEG2000 ซึ่งเป็นมาตรฐานการบีบอัดภาพนิ่ง JPEG อีกชนิดหนึ่ง
4. โปรแกรมนี้สามารถเป็นพื้นฐานสำหรับการศึกษา การบีบอัดข้อมูลวิดีโอ(MPEG)

เอกสารอ้างอิง

[1] Rafael C. Gonzalez and Richard E. Woods. **Digital Image Processing.** Addison Wesley Publishing Company, 2000

[2] เทอดศักดิ์ ธนกิจประภาและดร.ไกรสิน ส่งวัฒนา. **โปรแกรมบีบอัดภาพนิ่งด้วยวิธี JPEG ชนิด Sequential Baseline System.**การประชุมวิชาการทางวิศวกรรมไฟฟ้า ครั้งที่ 18, คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีมหานคร,2538

[3] Phillip E. Mattison. **Pratical Digital Video with Programming Example in C.** John Wiley & Son, Inc., 1994

[4] C.Wayne Brown and Barry J.Shepherd. **Graphics File Formats.** Manning PublicationsCo,1995

[5] International Standard . **Information Technology –Digital Compression and Coding of Continuous –tone Image : Requirements and guidelines** . Reference ISO/IEC 10918 ,1994.

ประวัติผู้เขียนโครงงาน

ชื่อ นางสาวพัฒชรินทร์ ฝุ่นทอง

ภูมิลำเนา 95 หมู่ 1 ต.ศรีคีรีมาศ อ.คีรีมาศ จ.สุโขทัย

ประวัติการศึกษา

- จบระดับมัธยมศึกษาจากโรงเรียนคีรีมาศพิทยาคม
- ปัจจุบันกำลังศึกษาในระดับปริญญาตรีชั้นปีที่ 4 สาขาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

E-mail: put_patty@hotmail.com