สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมแสดงการวิเคราะห์การบีบอัดสัญญาณภาพ

VISUAL MPEG ANALYSIS

โดย

นายนวพล ฉิมกลีบ

ว่าที่ ร้อยตรี ระพี แสงอรุณ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมสารสนเทศ

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2546

เลขหมู่.....
เลขทะเบียน 55722
วัน,เดือน,ปี 25 พ.ค. 2548

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

VISUAL MPEG ANALYSIS

BY

MR. NAWAPOL CHIMKLEEB

ACTING 2, LT RAPEE SANG-ARUN

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR THE DEGREE OF

BACHELOR IN DEPARTMENT OF INFORMATION ENGINERRING

FACULTY OF ENGINEERING

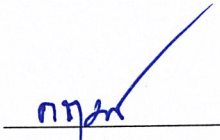
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2003

หัวข้อปริญญานิพนธ์	โปรแกรมแสดงการวิเคราะห์การบีบอัดสัญญาณภาพ
นักศึกษา	นายนวพล ฉิมกลีบ รหัสประจำตัว 44015695
	ว่าที่ ร.ต. ระพี แสงอรุณ รหัสประจำตัว 44015707
ระดับการศึกษา	ปริญญาตรี วิศวกรรมศาสตรบัณฑิต
	สาขาวิศวกรรมสารสนเทศ
ภาควิชา	วิศวกรรมสารสนเทศ
ปีการศึกษา	2546

ปริญญานิพนธ์ฉบับนี้ได้รับความเห็นชอบจากอาจารย์ที่ปรึกษาเป็นที่เรียบร้อยแล้ว

(ผศ. ไพศาล สิทธิโยภาสกุล)
อาจารย์ที่ปรึกษา



(อ. กฤดากร กล่อมการ)
อาจารย์ที่ปรึกษา

หัวข้อปริญญานิพนธ์	โปรแกรมแสดงการวิเคราะห์การบีบอัดสัญญาณภาพ	
ชื่อนักศึกษา	นายนวพล ฉิมกลีบ	รหัสประจำตัว 44015695
	ว่าที่ ร.ต.ระพี แสงอรุณ	รหัสประจำตัว 44015707
อาจารย์ที่ปรึกษา	ผศ. ไพศาล สิทธิโยภาสกุล	
	อ. กฤดากร กล่อมการ	
ระดับการศึกษา	ปริญญาตรีวิศวกรรมศาสตรบัณฑิต	
	สาขาวิศวกรรมสารสนเทศ	
ภาควิชา	วิศวกรรมสารสนเทศ	
ปีการศึกษา	2546	

บทคัดย่อ

ปริญญานิพนธ์นี้เป็นโปรแกรมแสดงการวิเคราะห์การบีบอัดสัญญาณภาพตามมาตรฐานเอ็มเป็ก-1 ซึ่งเริ่มต้นจากการที่บีบอัดข้อมูลทั้ง การบีบอัดข้อมูลแบบไม่มีการสูญเสีย และ การบีบอัดแบบมีการสูญเสีย และอาศัยหลักคณิตศาสตร์ คือ การใช้ดิสครีทโคไซน์ทรานสฟอร์ม ซึ่งเป็นการนำข้อมูลมาเสนอใหม่ในรูปของส่วนประกอบทางความถี่ จากนั้นใช้วิธีการ คอนไตเซชัน เพื่อทำการลดจำนวนบิตที่ใช้ในการเก็บสัมประสิทธ์จากการ ทรานสฟอร์มจะทำให้ค่าสัมประสิทธิ์ที่สำคัญและมีค่าสูงกระจุกอยู่บริเวณมุมบนซ้าย และใช้วิธี ซิกแซ็กสแกนในการตัดส่วนที่เป็นศูนย์ต่อเนื่องกันออกไปจากนั้นจึงนำไปบีบอัดโดยใช้ ฮัฟฟ์แมน โค้ด สุดท้ายทำการวิเคราะห์การเคลื่อนที่จะได้ค่าเวกเตอร์การเคลื่อนที่ ที่เกิดจากภาพนิ่งหลายภาพต่อเนื่องกัน เพื่อนำมาใช้ในการเพิ่มอัตราส่วนการบีบอัดข้อมูลเพิ่มมากขึ้น

Thesis Title	Visual MPEG Analysis	
Student	Mr. Nawapol Chimkleeb	ID 44015695
	Acting 2,LT Rapee Sang-arun	ID 44015707
Advisor	Asst. Prof Pisan Sidthiyopasakul	
	Prof. Kitdakol Komkal	
Graduate Level	Bachelor Degree of Information Engineering	
Department	Information Engineering	
Academic Year	2003	

ABSTRACT

This thesis is about Visual MPEG Analysis of MPEG-1 which begins by the two basic methods of compression. They are lossless compression and lossy compression which used the mathematical method of transformation. It is to use discrete cosine transform (DCT) that the representation of data is a new component of frequencies and after that uses Quantization. This method use to reduce the number coefficients of the bits. The height magnitude of DCT coefficient has density on left top of block. And Zigzag Scan method is used to cut a part of zero which is continued. After that it is compressed by Huffman coding. The last step use movement analysis to take the motion vector and increase compression ratio by it.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลงได้ด้วยความช่วยเหลืออย่างดียิ่งจากหลาย ๆ ฝ่าย ซึ่งผู้จัดทำใคร่ขอขอบคุณทุก ๆ ท่านที่มีส่วนร่วม ช่วยเหลือและแนะนำในทุก ๆ ด้าน

ขอขอบพระคุณ ผศ.ไพศาล สิทธิโยภาสกุล อาจารย์ที่ปรึกษาปริญญานิพนธ์ที่ได้กรุณาเสียสละเวลาให้คำปรึกษาและข้อเสนอที่เป็นประโยชน์ และ อาจารย์ กฤดากร กล่อมการ ที่ช่วยแนะนำและให้สถานที่ในการค้นหาข้อมูลทางอินเตอร์เน็ต จึงทำให้การทำปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงไปด้วยดี ตลอดจน อาจารย์ พนารัตน์ ระวิวรรณ ที่ให้คำปรึกษาและเอกสารที่สำคัญ

ขอขอบคุณ เพื่อน ๆ นักศึกษาทุกคนที่ช่วยเหลือให้คำแนะนำต่าง ๆ ในการจัดทำและตรวจสอบปริญญานิพนธ์จนสำเร็จสมบูรณ์และยังกำลังใจต่อคณะผู้จัดทำตลอดมา

ขอกราบขอบพระคุณบิดา มารดา ผู้ให้โอกาสและคอยให้กำลังใจเสมอมา

นายนวพล ฉิมกลีบ

ว่าที่ร.ต.ระพี แสงอรุณ

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....ก

บทคัดย่อภาษาอังกฤษ.....ข

กิตติกรรมประกาศ.....ค

สารบัญ.....ง

สารบัญรูป.....ช

สารบัญตาราง.....ญ

บทที่ 1 บทนำ.....1

1.1. บทนำ.....1

1.2. วัตถุประสงค์ของโครงงาน.....2

1.3. ของเขตของโครงงาน.....3

1.4. ผลที่คาดว่าจะได้รับ.....3

บทที่ 2 ทฤษฎีและหลักการ.....5

2.1 บทนำ.....5

2.1.1 Codec.....5

2.1.2 Motion-JPEG.....5

2.1.3 M-JPEG.....5

2.1.4 MPEG.....6

2.2 ทฤษฎีการบีบอัดข้อมูล.....6

2.2.1 การบีบอัดข้อมูลแบบไม่มีการสูญเสีย.....7

2.2.2การบีบอัดข้อมูลแบบมีการสูญเสีย.....7

2.3 กระบวนการเข้ารหัสแบบเจเป็ก (JPEG CODEC).....7

2.4 ทฤษฎีสี.....11

2.4.1 รูปจำลองของสี (color model).....11

2.4.2 รูปจำลองของสีแบบอาร์จีบี (RGB color model).....11

2.4.3 แซมเปิ้ลของสีต่อพิกเซล.....12

2.5 การแปลงระบบสี RGB เป็นระบบสี YCbCr.....13

สารบัญ (ต่อ)

หน้า

2.5.1 การแปลงระบบ RGB ในเป็นสี YCbCr โดยใช้ตารางสี.....14

2.5.2 การแปลงระบบ RGB ในเป็นสี YCbCr โดยตรง.....14

2.6 การซับแซมปลิง (Subsampling).....14

2.7 การแปลงแบบโคซายน์.....17

2.8 การควอนไตซ์ (Quantized).....22

2.8.1การเลือกเมทริกซ์ที่จะนำมาทำการควอนไตซ์..... 23

2.9 การควอนไตซ์เป็นลำดับแบบซิกแซก.....24

2.10 การเข้ารหัสแบบเจเป็ก (JPEG Coding) 25

2.10.1 การเข้ารหัสเอนโทรปี (Entropy Encoding)..... 26

2.10.2 การเข้ารหัสแบบรันเลนจ์ (Run Length).....26

2.10.3 การเข้ารหัสฮัฟแมน (Huffman Coding).....27

2.11 หลักการบีบข้อมูลวีดีโอและมาตรฐาน MPEG.....28

2.11.1. รูปแบบของการลำดับภาพนิ่ง.....28

2.11.2. การลดที่ว่างที่ไม่จำเป็นที่เกิดขึ้นชั่วคราว..... 28

2.11.3. การลดที่ว่างที่ไม่จำเป็น.....29

2.11.4. การชดเชยการเคลื่อนไหว.....30

2.12 การทำนายสำหรับการชดเชยการเคลื่อนที่.....31

2.12.1 บรรทัดฐานการเปรียบเทียบ (Matching Criterion).....33

2.13 อัลกอริธึมสำหรับ Motion estimation.....33

2.13.1 วิธีการค้นหาแบบเต็มรูปแบบ.....33

2.13.2 วิธีการค้นหาแบบลอการิทึมสองมิติ..... 34

2.13.3 วิธีการค้นหาแบบลำดับชั้นคู่ขนานหนึ่งมิติ 36

2.13.4 วิธีการประมาณค่าการเคลื่อนที่แบบลำดับชั้น.....37

2.14 มาตรฐานเอ็มเป็ก (MPEG).....40

2.14.1 พื้นฐานเบื้องต้นของ MPEG.....40

2.15 ชนิดของภาพ.....45

2.15.1 Intra pictures (I-pictures)..... 45

สารบัญ (ต่อ)

หน้า

2.15.2 Predicted pictures (P-pictures).....45
2.15.3 Bidirectionally predicted pictures (B-pictures).....46
2.15.4 DC-coded picture หรือ D-picture..... 47
2.16 การเข้ารหัสวีดีโอ..... 49
2.16.1 กระบวนการก่อนกระบวนการจริง (Preprocessing).....49
2.16.2 การประมาณการเคลื่อนที่และการชดเชยการเคลื่อนที่.....49
2.17 Video decoder..... 52
2.18 โครงสร้างของในการเข้ารหัสของบิทสตรีมที่เป็นภาพเคลื่อนไหว.....53
2.18.1 Mocroblock coding..... 55
บทที่ 3 การออกแบบระบบ.....57
3.1 กระบวนการทำภาพ JPEG..... 57
3.2 การแปลงข้อมูลจาก RGB ไปเป็น YCrCb..... 59
3.3 ซับแซมปลิ้ง..... 59
3.4 การถอดรหัสของเฟรมภาพ..... 62
3.5 การขยายข้อมูลแบบ Chorma 2 ทิศทาง 4:1:1..... 62
3.4 Motion Estimate..... 66
3.5 Motion Compensated..... 68
บทที่ 4 ผลการทดลอง.....71
บทที่ 5 สรุป..... 82
บรรณานุกรม..... 83

สารบัญรูป

รูปที่ หน้า

1.1 แสดงโครงสร้างตัวเข้ารหัสและถอดรหัส MPEG.....3

2.1 แสดงการEncoder ของ JPEG..... 6

2.2 แสดงการเข้ารหัสแบบ JPEG.....7

2.3 แสดงคอมโพเนนท์ของภาพสี.....15

2.4 ความละเอียดของคอมโพเนนท์ในการประมวลผลข้อมูลคอมโพเนนท์..... 16

2.5 แสดงการประมวลผลข้อมูลแต่ละคอมโพเนนท์แบบไม่ซ้อนกันทับกัน.....16

2.6 แสดงทิศทางการประมวลผลข้อมูลแต่ละคอมโพเนนท์
แบบที่ความละเอียดของแต่ละคอมโพเนนท์ไม่เท่ากัน..... 17

2.7 แสดงตัวอย่างการเข้ารหัสการแปลงจากโดเมนหนึ่งไปยังอีกโดเมนหนึ่ง 18

2.8 แสดงการแปลงดีสครีตโคโซน์แบบ 2 มิติ.....20

2.9 แสดงการแปลงดีสครีตโคโซน์ที่ทำในบล็อกขนาด 8×8.....20

2.10 แสดงแพทเทิร์นของความเข้มของสัมประสิทธิ์การถ่วงน้ำหนัก
การแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง.....21

2.11 การกระจายของส่วนประกอบทางความถี่ (ประมาณ)
ของการแปลงดิสครีตโคซายน์.....22

2.12 แสดงการกระจายค่าของสัมประสิทธิ์ที่ใช้ในการควอนไตซ์..... 23

2.13 แสดงเมทริกซ์ของค่าควอนตัมที่แฟกเตอร์คุณภาพเท่ากับ 2..... 24

2.14 แสดงทิศทางการจัดลำดับแบบซิกแซก.....25

2.15 Temporal correlation ในภาพที่ต่อเนื่องกัน..... 29

2.16 การทำงานโดยทั่วไปของ Video coder และ Video decoder.....31

2.17 Motion – estimation process..... 32

2.18 ตัวอย่างของวิธีการค้นหาแบบ 3 ขั้นตอน..... 35

2.19 ตัวอย่างของวีการค้นหาแบบ PHODS 37

2.20 วิธีการทั่วไปของการค้นหาเวคเตอร์ของการเคลื่อนที่
ในวิธีการค้นหาแบบลำดับชั้น.....38

2.21 แสดงให้เห็นการสแกนสลับกันและสแกนแบบเรียงลำดับ.....41

สารบัญรูป (ต่อ)

รูปที่ หน้า

2.22 ข้อกำหนดของ macroblock ในมาตรฐาน MPEG....43
2.23 แสดงการเข้ารหัสของ Intra Frame....45
2.24 แสดงการเข้ารหัสของ P-Frames....46
2.25 แสดงการเข้ารหัส B-Frame....47
2.26 ความเกี่ยวข้องกันระหว่างภาพแบบ I, P และB ในอนุกรมวีดีโอ....47
2.27 การกระจายบิทระหว่างภาพ I, P และ B ในการเข้ารหัสบิทสตรีม....48
2.28 Block diagram of an MPEG encoder....49
2.29 Forward motion compensation....50
2.30 Bidirectional motion compensation....51
2.31 Block diagram of an MPEG decoder....53
2.32 Syntax layers in MPEG video coding....55
2.33 Decision trees for coding macroblock in I- P- and B-picture....56
3.1 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมการเข้ารหัสของเฟรมภาพ....58
3.2 การบีบอัดข้อมูลแบบ Chroma 2 ทิศทาง 4:1:1....60
3.3 กระบวนการ FDCT, ควอนไตซ์และการเข้ารหัส....61
3.4 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมการถอดรหัสของเฟรมภาพ....63
3.5 การขยายข้อมูลแบบ Chroma 2 ทิศทาง 4:1:1....64
3.6 กระบวนการ IDCT, ดีควอนไตซ์และการถอดรหัส....65
3.7 แสดงขั้นตอนการทำงานของ Motion estimation ของเฟรม P....67
3.8 แสดงขั้นตอนการทำงานของ Motion Compensated ของเฟรม P....69
3.9 แสดงขั้นตอนการทำงานของ Motion estimation ของเฟรม B....70
4.1 แสดงถึงรูปต้นฉบับที่เป็น RGB ยังไม่ได้การลดขนาด....71
4.2 แสดงถึงรูปต้นฉบับที่เป็น RGB ยังไม่ได้การลดขนาดและมี Grid....71
4.3 แสดงถึงการถอดเอาเฉพาะสัญญาณ Y ที่แยกออกมาจากภาพต้นฉบับ....72
4.4 แสดงถึงการถอดเอาเฉพาะสัญญาณ Cr ที่แยกออกมาจากภาพต้นฉบับ....72
4.5 แสดงถึงการถอดเอาเฉพาะสัญญาณ Cb ที่แยกออกมาจากภาพต้นฉบับ....73
4.6 แสดงถึงการถอดเอาเฉพาะสัญญาณ R ที่แยกออกมาจากภาพต้นฉบับ....73

สารบัญรูป (ต่อ)

รูปที่ หน้า

4.7 แสดงถึงการถอดเอาเฉพาะสัญญาณ G ที่แยกออกมาจากภาพต้นฉบับ.....74
4.8 แสดงถึงการถอดเอาเฉพาะสัญญาณ B ที่แยกออกมาจากภาพต้นฉบับ.....74
4.9 แสดงถึงรูปที่ผ่านการ Sampling 4:1:1 และ Quantization = 5.....75
4.10 แสดงถึงรูปที่ผ่านการ Sampling 4:4:4 และ Quantization = 5..... 75
4.11 แสดงถึงรูปที่ผ่านการ Sampling 8:1:1 และ Quantization = 5..... 76
4.12 แสดงการ Zoom ของ Pixel ในรูปที่ตำแหน่ง X=304 และ Y=96 76
4.13 แสดงค่าของสัญญาณ Y.....76
4.14 แสดงค่าที่ผ่านการ DCT และ Quantization = 5.....77
4.15 แสดงการผ่าน Inverse DCT ของสัญญาณ Y ข้างต้น.....77
4.16 แสดงค่าของ Cb ก่อนทำการ Sampling ที่ได้จากภาพ.....77
4.17 แสดงค่าของ Cb หลังทำการ Sampling ในอัตรา 4:1:1 ที่ได้จากภาพ.....77
4.18 แสดงค่าของ Cb หลังทำการ Sampling ในอัตรา 8:1:1 ที่ได้จากภาพ.....78
4.19 แสดงค่าของ Cr ก่อนทำการ Sampling ที่ได้จากภาพ.....78
4.20 แสดงค่าของ Cr หลังทำการ Sampling ในอัตรา 4:1:1 ที่ได้จากภาพ.....78
4.21 แสดงค่าของ Cr หลังทำการ Sampling ในอัตรา 8:1:1 ที่ได้จากภาพ..... 78
4.22 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม B1 ทั้ง Forward และ Backward..... 79
4.23 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม B2 ทั้ง Forward และ Backward.....79
4.24 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม P ที่มีแต่เฉพาะ Forward เท่านั้น.....80
4.25 แสดงถึง Motion Compensated ของเฟรม B1.....80
4.26 แสดงถึง Motion Compensated ของเฟรม B2.....81
4.27 แสดงถึง Motion Compensated ของเฟรม P..... 81
4.28 แสดงถึงค่าที่ค้นการใน Area Search ที่ Match กันกับ Block ค้นหา..... 82
4.29 แสดงการ Zoom ของ Macro block.....82

สารบัญตาราง

ตารางที่ หน้า

1.1 Gang Chart..... 4
2.1 เปรียบเทียบคุณสมบัติของการ Search วิธีต่าง ๆ.....39
2.2 ขนาดของภาพ สำหรับระบบ CCIR 601 , SIF และพิกเซลในพื้นที่ใช้งาน.....43
2.3 ตัวแปรที่ถูกบังคับในการเข้ารหัส.....44

บทที่ 1
บทนำ

1.1 บทนำ

การมองเห็นของมนุษย์เป็นสิ่งที่สำคัญที่สุดและเป็นกลไกการรับภาพที่ซับซ้อนอย่างหนึ่ง ซึ่งจะให้มนุษย์ได้รับข้อมูลที่มีความจำเป็นสำหรับใช้งานง่าย ๆ และสำหรับงานที่มีความซับซ้อน ดังนั้นรูปภาพจึงมีบทบาทมากสำหรับองค์กรต่าง ๆ เช่น หนังสือพิมพ์ โทรทัศน์ ภาพยนตร์ซึ่งได้ใช้ทั้งภาพนิ่ง และ ภาพเคลื่อนไหว เป็นสื่อในการนำเสนอข้อมูลข่าวสารต่าง ๆ สิ่งที่น่าสนใจของข้อมูลเกี่ยวกับการมองเห็นหรือข้อมูลภาพนั้นก็คือกระบวนการประมวลผลภาพโดยใช้ดิจิตอลคอมพิวเตอร์ และในการชมภาพยนตร์ผ่านทางวีดีโอ (VDO) ซึ่งเป็นระบบแบบอนาล็อก (Analog) เมื่อเราต้องการนำสัญญาณนี้มาเข้าสู่คอมพิวเตอร์หรือนำมาทำเป็นสัญญาณดิจิตอล (Digital) ข้อมูลวีดีโอจะมีขนาดใหญ่มากจึงต้องใช้หน่วยความจำในการเก็บจำนวนมากถึงขนานประมาณ 69 จิกะไบต์ ต่อ วีดีโอเทปหนึ่งชั่วโมง และแล้วต่อมามีผู้คิดค้นแผ่นซีดีรอม (CD-ROM) ขึ้นมีขนาด 600 เมกะไบต์ ทำให้มนุษย์ใช้งานคอมพิวเตอร์ (Computer) กันมากขึ้น โดยเฉพาะงานทางด้านมัลติมีเดีย (Multimedia) นั้น มีการพัฒนาซอร์ฟแวร์ (Software) ในการบีบอัดกันมากขึ้น ซึ่งการบีบอัดนี้ มีทั้งซอร์ฟแวร์ (Software) และฮาร์ดแวร์ (Hardware) ต่อมาก็ได้เกิดการบีบอัดขึ้นอันหนึ่งคือ MPEG ซึ่งย่อมาจาก Motion Picture Experts Group เป็นชื่อของกลุ่มทำงานกลุ่มหนึ่งที่ก่อตั้งขึ้นมาเมื่อปี พ.ศ. 2531 เป็นคณะกรรมการที่จัดตั้งขึ้นจากการสนับสนุนของ องค์การมาตรฐานสากล (International Standards Organization : ISO) โดยมีหน้าที่ในการสร้างมาตรฐานสำหรับการบีบข้อมูลภาพ (ที่ไม่ใช่ภาพนิ่ง) และเสียงในระบบดิจิตอล ซึ่งเป็นเป้าหมายของกลุ่มก็คือการสร้างมาตรฐานที่ระบุการเข้ารหัสชุดข้อมูล และความต้องการในการถอดรหัสเพื่อใช้กับภาพและเสียงดิจิตอลคุณภาพสูงโดยที่มาตรฐานของ MPEG เป็นที่ยอมรับอย่างแพร่หลายในการทำงานกับภาพและเสียง MPEG ได้สร้างการประยุกต์ของการเข้ารหัสไว้หลายอย่างตั้งแต่ ภาพวิดีโอและซีดีแบบสื่อประสมสำหรับคอมพิวเตอร์ทีวีระบบโต้ตอบไปจนถึงเครือข่ายดาวเทียมดิจิตอลการบีบข้อมูลสามารถกระทำผ่านกระบวนวิธีหรืออัลกอริทึมที่แบ่งเฟรมของภาพออกเป็นตารางสี่เหลี่ยม แล้วมองหาข้อมูลที่สามารถตัดออกไปได้ในกรอบสี่เหลี่ยมนั้น วิธีการนี้เป็นการวิเคราะห์ภายในเฟรม (intraframe analysis) และจะมีการเปรียบเทียบเฟรมภาพแต่ละเฟรมที่ต่อเนื่องกันซึ่งจะทำให้ตัดสินใจได้ว่าจะตัดข้อมูลใดทิ้งไป หลักการนี้เรียกว่า การบีบอัดข้อมูลส่วนเกิน (lossy compression) ซึ่งเป็นหลักที่ใช้ในระบบของ MPEG ในปริญญานิพนธ์ฉบับนี้จึงมุ่งเน้นในการศึกษา

การทำงานของมาตรฐาน MPEG ที่ทำการบีบอัดข้อมูลมีการเปลี่ยนแปลงไปอย่างไร โดยยังคงแสดงภาพออกมาเหมือนเดิมแต่ความละเอียดน้อยลง

1.2 วัตถุประสงค์ของโครงงาน

โครงงานนี้เป็นการวิเคราะห์การบีบอัดข้อมูลจากข้อมูลวีดีโอ โดยใช้วิธีการของ MPEG – 1 ซึ่งใช้โปรแกรม Visual Basic 6 ในการเขียนโปรแกรม โดยมีวัตถุประสงค์ดังนี้

1. ศึกษารูปแบบการแสดงของภาพให้ตรงกับมาตรฐาน JPEG
2. ศึกษาวิธีการบีบอัดข้อมูลภาพ
3. ศึกษาการเข้ารหัสโดยให้ตรงกับการทำงานของ MPEG - 1
4. สามารถเขียนโปรแกรมแสดงขั้นตอนการบีบอัดข้อมูลให้อยู่ในการทำงานของ MPEG - 1 ได้

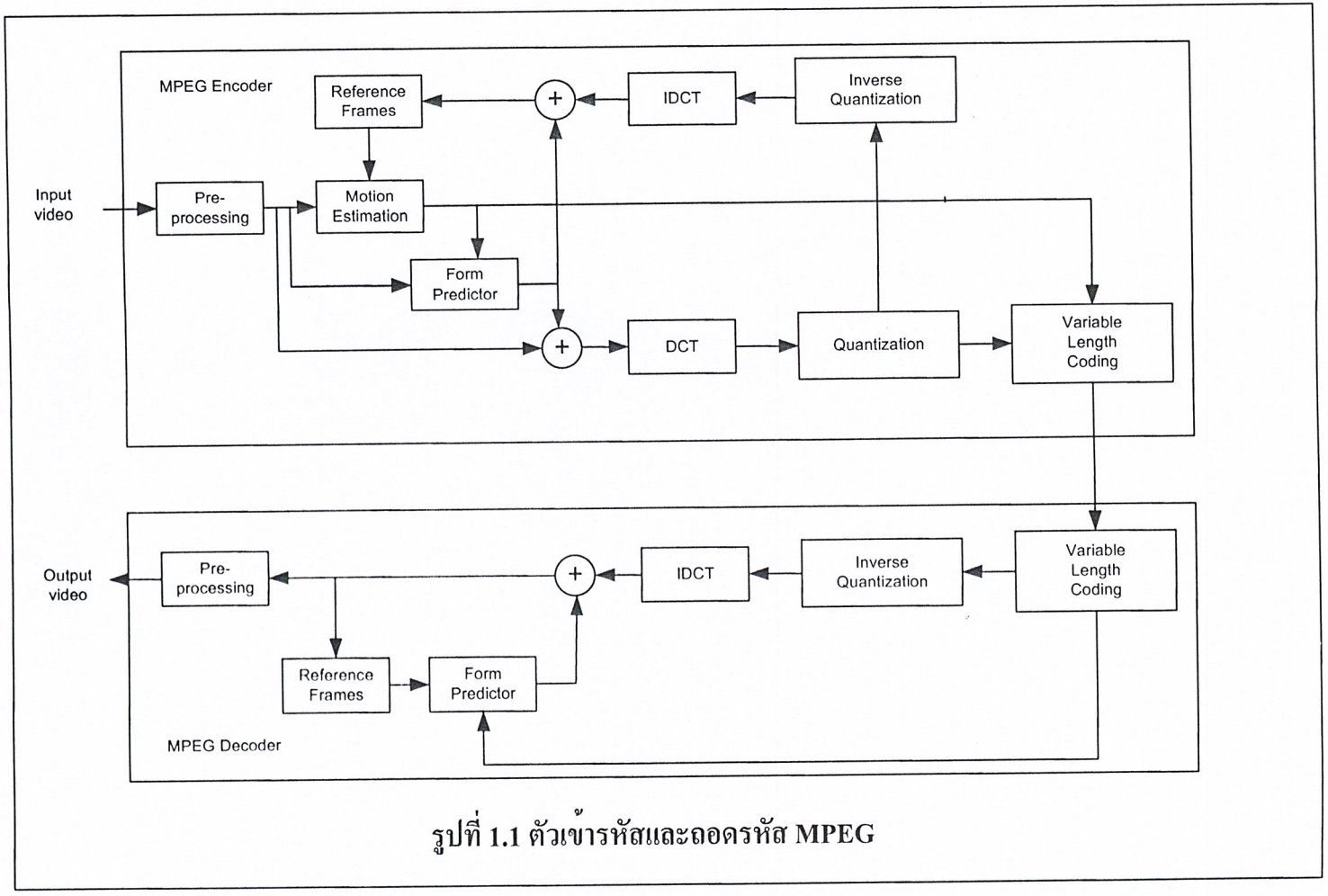
1.3 ขอบเขตของโครงงาน

การเขียนโปรแกรมที่สามารถบีบอัดข้อมูลให้ได้ตามมาตรฐาน MPEG - 1 โดยใช้โปรแกรม Visual Basic 6 ในการทำงานบีบอัดข้อมูล

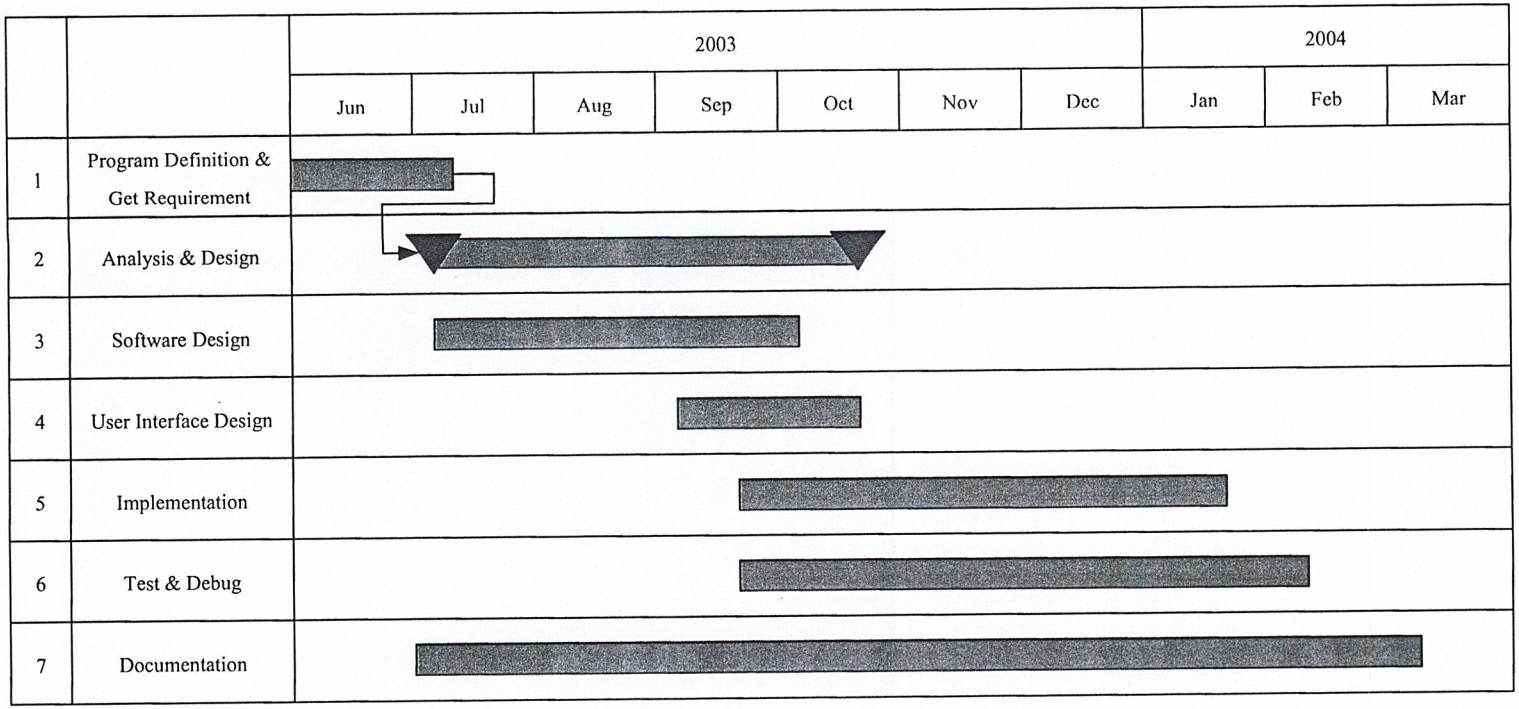
1. ศึกษาการแปลงข้อมูลโดยใช้ DCT (Discrete Cosine Transform)
2. ศึกษาวิธีการบีบอัดข้อมูลภาพแบบ JPEG ได้
3. ศึกษาวิธีการบีบอัดข้อมูลโดยให้ตรงตามการทำงานของ MPEG - 1
4. ศึกษาการหาเวกเตอร์เคลื่อนที่ (Motion Vector) ซึ่งใช้ในการเข้ารหัสแบบ MPEG-1
5. ศึกษาโปรแกรม Visual Basic 6 ซึ่งเป็นโปรแกรมที่ใช้เขียนในการบีบอัดข้อมูล
6. เขียนโปรแกรมแสดงขั้นตอนและรายละเอียดการเข้ารหัสของเฟรมภาพได้
7. แสดงภาพที่ถูกเข้ารหัสซึ่งถูกกระทำโดยโปรแกรม

1.4 ผลที่คาดว่าจะได้รับ

เราสามารถรู้จักวิธีการทำงานการบีบอัดข้อมูล แบบ JPEG, แบบ MPEG – 1 และเขียนโปรแกรมในการวิเคราะห์ การทำงานของการบีบอัดนี้ได้ ทำให้เราสามารถที่จะพัฒนาขึ้นไปเป็นการบีบอัดที่มีมาตรฐานเพิ่มขึ้นไปอีกได้ โดยใช้พื้นฐานจากโครงงานนี้เป็นความรู้เบื้องต้น



รูปที่ 1.1 ตัวเข้ารหัสและถอดรหัส MPEG



ตารางที่ 1.1 Gang chart

บทที่ 2
ทฤษฎีและหลักการ

2.1 บทนำ

การลดขนาดข้อมูลภาพเป็นการลดจำนวนของข้อมูลที่จะใช้แทนภาพนั้น ๆ โดยมีหลักการคือตัดข้อมูลส่วนที่เกินความจำเป็นออกไป หรือส่วนที่มีการซ้ำซ้อนกันออก จึงทำให้ข้อมูลนั้นเหลือน้อยลงเพื่อให้สามารถจัดการเก็บข้อมูลได้ง่ายและน้อยลง ในงานด้านมัลติมีเดีย (Multimedia) ส่วนใหญ่แทบจะเป็นไปไม่ได้เลยที่จะสามารถทำงานได้ถ้าหากปราศจากขบวนการบีบอัดข้อมูล ปัจจุบันได้มีการพัฒนาทางด้านสถาปัตยกรรมการประมวลผลภาพนิ่ง ภาพเคลื่อนไหวและเสียงโดยจัดการกับข้อมูลดิจิตอล (Digital) เพื่อลดขนาดของข้อมูลโดยการบีบอัดข้อมูล

2.1.1 Codec

Codec คือเทคนิคในการบีบอัดข้อมูลมาจากคำว่า COmpression / DECompression ในการบีบอัดนี้สามารถทำได้โดยใช้ซอฟต์แวร์ หรือใช้ฮาร์ดแวร์ที่เป็นไอซีที่ออกแบบมาเป็นพิเศษมาช่วยก็ได้ Codec สำหรับการจับวีดีโอ (Capture) ที่นิยมใช้กับคือ

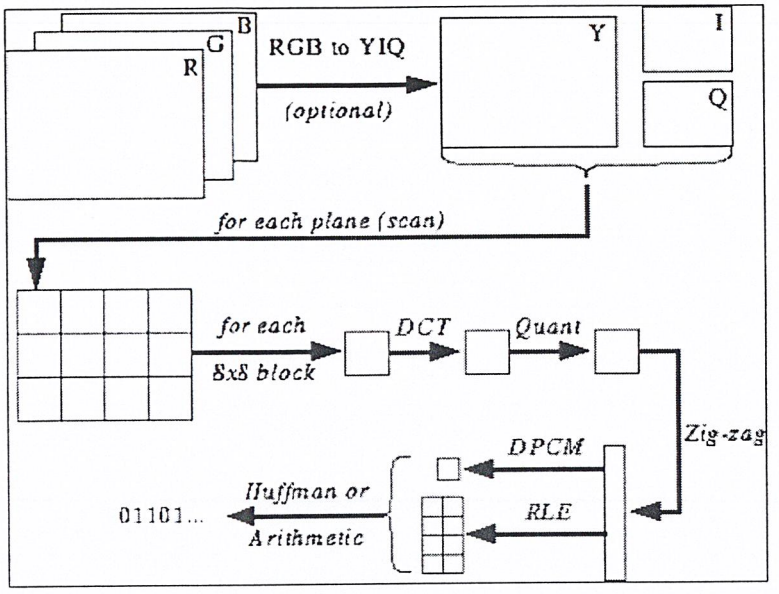
2.1.2 Motion-JPEG

Motion-JPEG เป็นรูปแบบการบีบอัดข้อมูลที่นิยมกันมากในการตัดต่อวิดีโอเป็นการบีบอัดข้อมูลที่ยังให้คุณภาพที่สูงมาก และเป็นการบีบอัดที่ใช้วิธีการบีบอัดภาพนิ่งแบบ JPEG (Joint Photographic Experts Group) ซึ่งจะลดความซ้ำซ้อนภาพในภาพเดียวกัน (Intra-Frame) โดยใช้หลักการคณิตศาสตร์ Discrete Cosine Transform (DCT)

2.1.3 M-JPEG

M-JPEG เป็นการบีบอัดแบบ Lossy ถ้าการบีบอัดน้อยมาก เช่น น้อยกว่า 2:1 ก็เป็นแบบ Lossless ได้ ดังนั้นถ้าเราใช้การบีบอัดแบบน้อย ๆ ลดข้อมูลจริงมาเพียงนิดเดียว เช่น 3:1, 4:1 เราก็จะได้ขนาดของไฟล์มีขนาดใหญ่มากและคุณภาพก็ดีด้วย แต่ปัญหาก็คือไฟล์จะมีขนาดใหญ่และจะมีข้อมูลมหาศาลที่เข้าและออกระหว่างวีดีโอกับฮาร์ดดิสก์ (Hard disk) ของคอมพิวเตอร์ (Computer) ดังนั้นฮาร์ดดิสก์จะต้องมีความสามารถรับส่งข้อมูลได้เร็วเพียงพอกับจำนวนข้อมูลมหาศาลนี้ ความสามารถนี้เราหมายถึง Transfer Rate ของฮาร์ดดิสก์จะต้องทำงานเร็วพอ ถ้าทำงานไม่ทันวิดีโอที่เรา Capture เข้ามาฮาร์ดดิสก์ อาจเกิดอาการ Frame Drop หรือภาพกระตุก ในการบีบอัดแบบ M-JPEG นี้ เราสามารถกำหนดค่าของการบีบอัดและค่าขนาดความละเอียดของวิดีโอได้

M-JPEG เป็นระบบวิดีโอแบบปิดคือในการเล่นวิดีโอแบบ M-JPEG จะต้องมี Codec ของ M-JPEG ซอร์ฟแวร์หรือฮาร์ดแวร์ติดอยู่ที่ตัวเครื่องที่เล่นนี้ด้วย



รูปที่ 2.1 แสดงการEncode ของ JPEG

2.1.4 MPEG

MPEG มาจากคำว่า Motion Picture Experts Group ใช้หลักการลดความซ้ำซ้อนกันระหว่างภาพ (Inter Frame) ในกลุ่มภาพ (Group of Picture) เดียวกัน เป็นมาตรฐานการบีบอัดสัญญาณภาพและเสียงของภาพเคลื่อนไหวโดยใช้ระบบ (DCT) หรือ Discrete Cosine Transform ซึ่งคล้ายกับขั้นตอนการเข้ารหัสภาพนิ่งแบบ JPEG แต่ลดจำนวนข้อมูลที่ซ้ำ ๆ กันของภาพต่อ ๆ ไป การบีบอัดข้อมูลระบบ MPEG เป็นแบบไม่สมมาตร เพราะขั้นตอนในการเข้ารหัสเพื่อบีบอัดสัญญาณวิดีโอจะนานกว่าขั้นตอนการถอดรหัสข้อมูลออกไปเป็นภาพและเสียง MPEG เป็นระบบวิดีโอแบบเปิด คือไม่ขึ้นอยู่กับซอฟต์แวร์หรือฮาร์ดดิสก์ ดังนั้นจึงสามารถนำไปเล่นที่เครื่องไหนก็ได้

2.2 ทฤษฎีการบีบอัดข้อมูล

การบีบอัดข้อมูลทำเพื่อลดขนาดไฟล์ให้มีขนาดเล็กลง เช่น ข้อมูลขนาด 704 × 576 ลดขนาดลงเป็น352 ×288 หรือลดจำนวนเฟรมต่อวินาทีลงมาให้น้อยลงเช่น 15 เฟรมต่อวินาที สำหรับการบีบอัดข้อมูลนั้นสามารถแบ่งออกได้เป็น 2 ประเภทใหญ่คือ

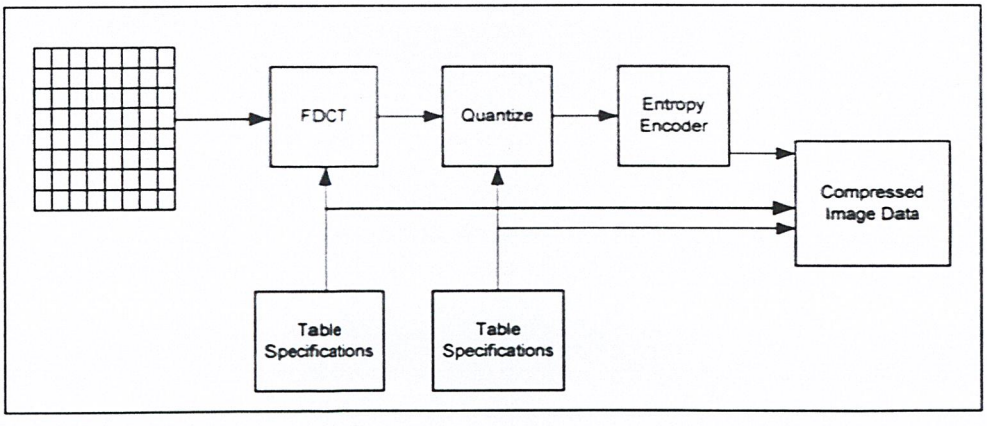
2.2.1 การบีบอัดข้อมูลแบบไม่มีการสูญเสีย (Lossless Compression)

เป็นเทคนิคการลดขนาดข้อมูลที่รักษาความถูกต้องของข้อมูลได้ 100% ดังนั้นไฟล์ที่ถูกลดขนาดข้อมูลโดยเทคนิคนี้จะต้องมีความถูกต้องของข้อมูลเช่นเดียวกันกับข้อมูลต้นฉบับทุกประการ เทคนิคส่วนใหญ่จะถูกนำไปประยุกต์ใช้กับข้อมูลชนิดตัวอักษรเช่น ไฟล์ประเภทฐานข้อมูล (Database) สเปรดชีต (Spreadsheet) หรือข้อมูลที่ส่งตามเครื่องแฟกซ์ (Fax) เป็นต้น ซึ่งไฟล์ข้อมูลประเภทนี้ล้วนมีความสำคัญทุก ๆ ตัวอักษร (Byte) ดังนั้นถ้าหากมีการสูญหายหรือเปลี่ยนแปลงข้อมูลในไฟล์ที่ถูกลดขนาดแล้วเพียงแค่บิต (Bit) เดียว ก็อาจทำให้ข้อมูลของไฟล์ที่ถูกขยายขนาดแล้วมีความคลาดเคลื่อนไป ซึ่งอาจจะไม่สามารถถูกประมวลผลได้

2.2.2 การบีบอัดข้อมูลแบบมีการสูญเสีย (Lossy Compression)

เป็นเทคนิคการลดขนาดข้อมูลที่ยอมให้ความถูกต้องของข้อมูลสูญเสียหายไปบ้าง ทั้งนี้ก็เพื่อความเร็วของการลดขนาดข้อมูล ซึ่งวิธีนี้ได้ถูกนำไปใช้ในงานด้านกราฟิกอิมเมจ (Graphic Image) ซึ่งเป็นเทคนิคที่ใช้สำหรับการลดขนาดของข้อมูลประเภทนี้ สามารถปรับให้มีความถูกต้องหลายระดับ ซึ่งถ้าหากถูกกำหนดให้มีความแม่นยำสูงก็จะมีผลทำให้ขนาดไฟล์ที่ลดขนาดใหญ่ขึ้น และใช้เวลาในการประมวลผลนานขึ้นโดยเปรียบเทียบกับกรณีที่กำหนดให้มีความแม่นยำลดลง ในปัจจุบันได้มีซอร์ฟแวร์ (Software) หลายประเภทที่เสนอการลดขนานข้อมูลโดยใช้เทคนิคนี้

2.3 กระบวนการเข้ารหัสแบบเจเป็ก (JPEG CODEC)



รูปที่ 2.2 แสดงการเข้ารหัสแบบเจเป็ก

ลำดับของการเข้ารหัสแบบเจเป็กเป็นไปตามรูปที่ 2.2 โดยที่ภาพต้นฉบับจะถูกแบ่งออกเป็นบล็อก (Block) เล็ก ๆ ขนาด 8×8 พิกเซล ถ้าเป็นภาพที่แสดงผลด้วยจำนวน n บิทต่อพิกเซลแล้ว ค่าข้อมูลพิกเซลจะมีค่าตั้งแต่ 0 ถึง 2^n - 1 ก่อนที่จะผ่าน การแปลงฟอร์เวิร์ดดิสครีตโคซายน์ (Forward Discrete Cosine Transform หรือ FDCT) จะมีการสร้างค่าของพิกเซลให้เป็นจำนวนเต็มแบบมีเครื่องหมาย ซึ่งมีค่าตั้งแต่ -2^n ถึง 2^n - 1 ตัวอย่างเช่น ภาพแบบเกรย์สเกลที่มี 8 บิทต่อพิกเซล (ความเข้ม 2^8 = 256 ระดับ) มีค่า 0 – 255 เมื่อจะผ่านการแปลงต้องทำข้อมูลให้เป็น -128 ถึง 127 ก่อน

เมื่อผ่านการแปลงโคซายน์แบบไม่ต่อเนื่อง (DCT) แล้วเอาท์พุทที่ได้จะเป็นค่าสัมประสิทธิ์ทางความถี่ ต่อมาจะเป็น การควอนไตซ์ (Quantized) ค่าสัมประสิทธิ์เหล่านี้ ในขั้นตอนนี้เองที่จะมีการลดขนาดของข้อมูลและเกิดมีความผิดพลาดจากการควอนไตซ์ขึ้น ในการควอนไตซ์นี้จะเป็นแบบไม่เป็นยูนิฟอร์ม (Non - Uniform) กล่าวคือค่าสัมประสิทธิ์แต่ละค่าจะถูกควอนไตซ์ด้วยระดับการควอนไตซ์ (Step Size) ที่ไม่เท่ากันเสมอ ขนาดของระดับขั้นหรือสเตป (Step) ในการควอนไตซ์จะถูกกำหนดโดยค่าในตารางเรียกว่าเป็นตารางค่าควอนตัม (Quantum) นั่นเอง ซึ่งค่าในตารางนี้จะเป็นค่าที่ใช้เฉพาะกับสัมประสิทธิ์ ที่ได้จากการแปลงโคซายน์แบบไม่ต่อเนื่องแต่ละค่าเท่านั้น

ค่าสัมประสิทธิ์ที่ถูกควอนไตซ์และจะถูกนำมาจัดตามลำดับซิกแซก (Zigzag) เพื่อความสะดวกในการเข้ารหัสซึ่งอาจใช้วิธีของฮัฟแมน (Huffman encoding) หรือวิธีการเข้ารหัสแบบความยาวของรหัสไม่คงที่ (Variable Length Encoding) เอาท์พุทจากกระบวนการนี้จะเป็นข้อมูลภาพที่ถูกลดขนาดเรียบร้อยแล้ว

จากบล็อกไดอะแกรม การเข้ารหัสแบบเจเป็กประกอบด้วยบล็อกที่สำคัญ 3 บล็อก คือ

1. ฟอร์เวิร์ดดิสครีตโคซายน์ทรานสฟอร์ม (Forward Discrete cosine Transform (FDCT))
2. ควอนไตเซชัน (Quantization)
3. การเข้ารหัสเอนโทรปี (Entropy encoding)

ที่อินพุทของวงจรจะเข้ารหัสแซมเปิลของข้อมูลภาพที่ไม่มีเครื่องหมาย (Original unsigned samples) ที่อยู่ในช่วง $[0, 2^{P-1}]$ จะถูกเปลี่ยนให้เป็นจำนวนเต็มแบบมีเครื่องหมายซึ่งอยู่ในช่วง $[-2^{P-1}, 2^{P-1}-1]$ เช่น ภาพแบบเกรย์สเกลมี P=8 แซมเปิล (sample) ซึ่งอยู่ในช่วง [0, 255] จะถูกเปลี่ยนให้อยู่ในช่วง [-128, +127]

จากนั้นข้อมูลภาพจะถูกแบ่งเป็นบล็อกแบบ 8×8 และแซมเปิลจากแต่ละบล็อกจะถูกแปลงให้อยู่ในโดเมนของความถี่ โดยใช้การแปลงโดเมนของความถี่ โดยมีสมการของการแปลงค่าเป็น

$$F(u,v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(i,j) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

และมีสมการการแปลงกลับเป็น

$$f(i,j) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u,v) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

โดย $F(u,v)$ คือ ค่าตัวแปร 2 มิติของ $N \times N$ DCT

$f(i,j)$ คือ ค่าของระดับสีในพิกเซล(Pixel)

$u,v,x,y = 0, 1, 2,..., N-1$

i,j เป็น Spatial coordinates ใน pixel data block

u,v เป็น Frequency coordinates ใน DCT coefficient block

ทั้งนี้เนื่องจากสมการ ที่ใช้ในการแปลงค่าสัมประสิทธิ์การแปลงแบบดิสครีตโคซายน์ (DCT) ถ้าจำนวนข้อมูลยิ่งมาก (N ยิ่งมาก) จะทำให้ใช้เวลาและกำลังในการคำนวณมาก ดังนั้นเจเป็ก (JPEG) จึงกำหนดการแปลงภาพโดยแบ่งการแปลงเป็นบล็อกย่อย ๆ ของภาพขนาดบล็อกละ 8×8 พิกเซล (N=8)

ขั้นตอนต่อไปคือการควอนไตเซชัน (Quantization) ซึ่งเป็นการปรับลดค่าสัมประสิทธิ์ที่ได้จากการทำการแปลงแบบดิสครีตโคซายน์ (DCT) เพื่อลดจำนวนบิทที่ใช้ในการเก็บซึ่งมีผลทำให้ความแม่นยำ (Precision) ของข้อมูลลดลง คือ ทำให้เกิดการสูญเสียข้อมูลบางส่วนไป โดยสัมประสิทธิ์จากการแปลงแบบดิสครีตโคซายน์ทั้ง 64 ค่าจะถูกทำการควอนไตซ์ (Quantized) โดยใช้ตารางการควอนไตซ์ขนาด 64 อิลิเมนต์ (element) การควอนไตเซชันจะลดค่าแอมปลิจูดของสัมประสิทธิ์ที่มีผลกับคุณภาพของภาพน้อยที่สุด เพื่อให้เป็นไปตามหลักการเพิ่มค่าสัมประสิทธิ์ที่มีค่าศูนย์ให้มากขึ้น และลดค่าสัมประสิทธิ์ที่ไม่มีความสำคัญลง การควอนไตเซชันกระทำได้ดังสมการต่อไปนี้

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

และมีสมการดิควอนไตเซชันดังนี้

$$T^{\circ}(u,v) = \hat{T}(u,v)Q(u,v)$$

เมื่อ $T(u,v)$ คือ ค่าสัมประสิทธิ์ที่ได้จากการแปลงแบบดิสครีตโคซายน์

$\hat{T}(u,v)$ คือ ค่าสัมประสิทธิ์ที่ได้จากการแปลงแบบดิสครีตโคซายน์ เมื่อถูกควอนไตซ์

$T^{\circ}(u,v)$ คือ ค่าสัมประสิทธิ์ที่ได้จากการแปลงแบบดิสครีตโคซายน์ เมื่อถูกดิควอนไตซ์

$Q(u,v)$ คือ ค่าควอนไตเซชัน

และ *round* คือ การหาจำนวนเต็มที่มีค่าใกล้เคียงที่สุดโดย $0 \le u \le 7, 0 \le v \le 7$

หลังจากผ่านการควอนไตเซชันแล้ว ขั้นตอนต่อไปคือการเรียงข้อมูลแบบซิกแซก โดยเป็นการเรียงค่าสัมประสิทธิ์ที่ได้จากการแปลงแบบดิสครีตโคซายน์ให้เป็นในบล็อกข้อมูล 8×8 ค่าให้เป็นแถวข้อมูล 64 ค่าโดยเรียงลำดับจากค่าสัมประสิทธิ์ของความถี่ต่ำไปหาค่าสัมประสิทธิ์ของความถี่สูงซึ่งทำให้ค่าสัมประสิทธิ์ที่มีค่าศูนย์ส่วนใหญ่ในช่วงความถี่สูงอันเนื่องมาจากการควอนไตเซชันมาอยู่ติดกันซึ่งจะเป็นประโยชน์ในการเข้ารหัสในขั้นตอนต่อไป ขั้นตอนสุดท้ายคือ การเข้ารหัสแบบเอนโทรปี (entropy encoding) ซึ่งเป็นการเข้ารหัสค่าที่ได้จากการควอนไตเซชันให้อยู่ในรูปลำดับเลขฐานสองที่กระชับขึ้น ซึ่งมาตรฐานเจเป็กมีการเข้ารหัสเอนโทรปี 2 แบบ คือ การเข้ารหัสฮัฟแมน (Huffman Encoding) และ การเข้ารหัสเชิงอริธเมติก (Arithmetic Encoding) ซึ่งในระบบซีเควนเชียลเบสไลน์ จะใช้การเข้ารหัสแบบฮัฟแมนเท่านั้น โดยวิธีการเข้ารหัสฮัฟแมน นั้นเป็นวิธีการเข้ารหัส (encode) ข้อมูลด้วยการสร้างรหัสขนาดต่าง ๆ ขึ้นจากสถิติของจำนวนข้อมูลที่เกิดขึ้น เพื่อนำรหัสที่สร้างขึ้นนั้นไปใช้แทนตัวข้อมูลเดิม โดยพยายามสร้างรหัส ที่มีขนาดสั้นสำหรับใช้แทนตัวข้อมูลที่เกิดขึ้นจำนวนมากและรหัส ที่มีขนาดยาวขึ้นเพื่อแทนตัวข้อมูลที่เกิดขึ้นน้อยลงลดหลั่นกันไป สำหรับวิธีการเข้ารหัสฮัฟแมน ที่ใช้ใน JPEG นั้นได้มีการอาศัยเทคนิคการเข้ารหัส วิธีอื่นมาช่วยด้วยคือ การเข้ารหัสแบบวาริเอเบิลเลนจ์ (variable-length encoding) และการเข้ารหัสแบบรันเลนจ์ (runlength encoding) ซึ่งจะแบ่งการเข้ารหัส เป็น 2 ส่วนคือ การเข้ารหัส ค่า DC และค่า AC ตามลำดับ

2.4 ทฤษฎีสี

มนุษย์สามารถมองเห็นสีได้เมื่อความยาวคลื่นแสงที่สามารถมองเห็นได้บางช่วงตกกระทบที่เรตินาของดวงตาบางคนสามารถเห็นสีเดียวกันได้จากการรวมกันของความยาวคลื่นที่ตากต่างกันจำนวนมาก จำนวนของการรวมกันของความยาวคลื่นต่าง ๆ ไม่มีขีดจำกัดแต่จำนวนของสีที่สามารถแยกแยะได้โดยมนุษย์มีขีดจำกัดดังนั้น "รูปแบบของสี" จึงเป็นที่ต้องการเพื่อที่จะแทนรหัสที่เป็นเอกลักษณ์สำหรับแต่ละสีที่มนุษย์สามารถเห็นได้ รูปแบบของสีที่มีอยู่มีเป็นจำนวนมาก แต่มีเพียงแค่ 3 รูปแบบที่ถูกใช้ในทางภาพสีซึ่งได้แก่

- รูปแบบสีของ RGB (Red-Green-Blue) ซึ่งถูกใช้สำหรับระบบการแผ่ของแสงสว่างมากกว่ารูปแบบอื่น ๆ เช่น โทรทัศน์และจอคอมพิวเตอร์
- รูปแบบสีของ CMY (Cyan-Magenta-Yellow) ใช้สำหรับระบบการดูดกลืนแสง เช่น การพิมพ์
- รูปแบบสีของ RYB (Red-Yellow-Blue) สำหรับการผสมในทางศิลปะ

2.4.1 รูปจำลองของสี (Color model)

รูปจำลองสีมีลักษณะเป็นรูประบบพิกัดแบบสามมิติ แต่ละสีเกิดจากการผสมระหว่างแกนสีทั้งสามโดยรูปแบบจำลองของสีในปัจจุบันได้หันเหไปตามฮาร์ดแวร์ เช่น จอภาพสี เครื่องพิมพ์ และการประยุกต์ใช้งานต่าง ๆ รูปแบบจำลองของสีที่นิยมใช้กันมากที่สุดสำหรับมอนิเตอร์และกล้องวีดีโอ คือ RGB model (RED, GREEN, BLUE) สำหรับรูปจำลองของสีแบบ CMY model (CYAN, MAGENTA, YELLOW) นิยมใช้กับเครื่องพิมพ์สี ส่วนรูปแบบการส่งแบบ YIQ model เป็นมาตรฐานสำหรับการส่งกระจายภาพโทรทัศน์สี โดย Y คือ ค่าการเปลี่ยนแปลงความเข้มข้นของแสง และ I กับ Q คือ ส่วนของค่าการเปลี่ยนแปลงความเข้มของสี ที่เป็น inphase และ quadrature ตามลำดับ ส่วนรูปจำลองกลางที่ใช้ในการเปลี่ยนมาระหว่างรูปจำลองสีแบบต่าง ๆ คือระบบรูปจำลองแบบ HIS (Hue, Saturation, Intensity) และรูปจำลองแบบ HSV (Hue, Saturation, Value)

2.4.2 รูปจำลองของสีแบบอาร์จีบี (RGB color model)

ลำของแสงเมื่อผ่านแก้วปริซึมแล้ว ลำของแสงที่ออกมาไม่ได้มีสีขาว แต่ประกอบไปด้วยสเปกตรัมที่ต่อเนื่องของสี มีย่านจากสีม่วง (violet) ไปจนถึงสีแดง(red) โดยพื้นฐานแล้วสีที่มนุษย์สามารถมองเห็นวัตถุเกิดมาจากธรรมชาติของแสงในการสะท้อนมาจากวัตถุ วัตถุที่สะท้อนแสงทั้งหมดทุกแสงจะมองเห็นเป็นสีขาว วัตถุที่สะท้อนบางส่วนของสเป็กตรัมจะให้บางสีเฉดของสี ตัวอย่างเช่น วัตถุสีเขียวสะท้อนช่วงที่มีความยาวคลื่น 500 ถึง 570 nm โดยดูดซับเอาความยาวคลื่นที่เหลือนอกนั้นไว้ทั้งหมด ทำให้เราสามารถมองเห็นวัตถุชิ้นนั้นเป็นสีเขียวนั่นเอง ในย่านที่

มนุษย์สามารถมองเห็นได้นั้นความยาวคลื่นของแสงจะเป็นตัวกำหนดสี ที่อยู่ในหน่วยนาโนเมตร ในระบบรูปแบบจำลองแบบอาร์จีบี สีแดง สีเขียวและสีฟ้าจะถูกเรียกว่าสีปฐมภูมิ (primary color) แสงที่มีความยาวคลื่นประมาณ 430 นาโนเมตรคือสีฟ้า ความยาวคลื่นประมาณ 550 นาโนเมตรคือสีเขียว แบะที่มีความยาวคลื่นประมาณ 700 นาโนเมตรคือสีแดง ในการผสมของแสงสีแดง เขียว และฟ้า เข้าด้วยกันในหลาย ๆ รูปแบบ ทำให้สามารถได้สีต่าง ๆ อื่น ๆ ออกมามากมาย ถึงแม้จะมีสีอีกมากที่ไม่สามารถสร้างโดยวิธีการนี้ได้ก็ตามแต่วิธีการนี้ก็มีประสิทธิภาพดีพอเพียงที่จะใช้ในการแสดงผลภาพ และมีความนิยมใช้กันอย่างมาก เช่นการแสดงผลในมอนิเตอร์สีสำหรับคอมพิวเตอร์ทั่วไป เป็นต้น

2.4.3 แซมเปิ้ลของสีต่อพิกเซล (pixel)

การแสดงสีโดยทั่วไปใช้หนึ่งแซมเปิลต่อพิกเซลสำหรับภาพขาวดำและสามแซมเปิลต่อพิกเซลสำหรับค่าสี RGB ค่าสีแสดงเปอร์เซ็นต์ของความเข้มแสงและสี โดยแซมเปิลหนึ่งแสดงเปอร์เซ็นต์ความเข้มของแสงและอีกสองแซมเปิลแสดงเปอร์เซ็นต์ความเข้มของสี

1. ตาของมนุษย์ไม่ไวต่อจำนวนการเปลี่ยนแปลงจริง ๆ ระหว่างความเข้มของแสงสองความเข้มแต่จะไวต่ออัตราส่วนของการเปลี่ยนแปลงเท่านั้น เช่น การเปลี่ยนแปลงของความเข้มของแสงจาก 0.1 เป็น 0.2 และจาก 0.2 เป็น 0.4 นั้นมนุษย์จะเห็นจำนวนของความเข้มที่เปลี่ยนแปลงเท่ากันเพราะอัตราส่วนเท่ากับ 2 เท่ากัน เมื่อกลุ่มของความเข้มของแสงไม่เปลี่ยนแปลงด้วยจำนวนที่คงที่ดังนั้นมันจึงถูกเรียกว่าไม่เป็นเชิงเส้น ซึ่งระบบที่เป็นเส้นทั้งหมดไม่เป็นเชิงเส้นรวมทั้งตาของมนุษย์ด้วย ดังนั้นในระบบสีจึงแทนค่าด้วยค่าการเปลี่ยนแปลงความเข้มของแสง

2. การที่เจเป็ก (JPEG) เลือกใช้ระบบสี YCbCr เนื่องจากระบบสี YCbCr นั้นมีการแยกสีออกเป็นว่าการเปลี่ยนแปลงความเข้มของแสง (Iuminance) และค่าการเปลี่ยนแปลงความเข้มของสี (chrominance) อย่างชัดเจนและในระบบการมองเห็นของตามนุษย์นั้น ตาของมนุษย์จะมีความไวต่อการเปลี่ยนแปลงความเข้มของแสงมากกว่าการเปลี่ยนแปลงความเข้มของสี ดังนั้นระบบสี YCbCr จึงมีประโยชน์ต่อการลดจำนวนข้อมูลสีต่อจำนวนจุดพิกเซลโดยการลดจำนวนสี ค่าการเปลี่ยนแปลงความเข้มของสี ด้วยวิธีซับแซมปลิง (Sub Sampling) ซึ่งจะไม่ทำให้คุณภาพของภาพเปลี่ยนไปมากนักเพราะตามนุษย์สังเกตไม่เห็น

เนื่องจากบิตแมพไฟล์ซึ่งเป็นรูปแบบข้อมูลภาพที่เราจะนำมาทำการบีบอัด ใช้ระบบสี RGB แต่ในระบบของเจเป็กจะใช้รูปแบบสีของระบบสี YCbCr ดังนั้นจึงต้องมีการแปลง ระบบสีจากระบบสี RGB มาเป็นระบบสี YcbCr ซึ่งการแปลงระบบสีสามารถทำได้ 2 วิธี ดังนี้

1. การแปลงระบบสีโดยตารางสี

สิ่งที่ใช้กันมากอย่างหนึ่งในเรื่องของทฤษฎีสีก็คือตารางสีซึ่งเป็นการแทนค่าสีของแต่ละพิกเซลด้วยค่าตัวเลขค่าหนึ่งในตารางสี (เรียกว่าค่าดัชนีสี) ซึ่งตารางสีสำหรับภาพสีและภาพขาวดำก็จะแตกต่างกัน

ภาพขาวดำใช้ตารางสีเพียงตารางเดียว (เนื่องจากมีค่าความเข้มค่าเดียวต่อ 1 พิกเซล) สำหรับภาพสีบ่อยครั้งที่จะใช้ตารางสีแยกกันสำหรับแต่ละส่วนของความเข้มต่อพิกเซล เช่น ตารางหนึ่งสำหรับสีแดง, ตารางหนึ่งสำหรับสีเขียว และอีกตารางหนึ่งสำหรับสีน้ำเงิน

สาเหตุที่เราใช้ตารางสีก็เพื่อลดขนาดของหน่วยความจำที่ต้องการสำหรับภาพ เนื่องจากจำนวนบิทที่เราใช้แทนแต่ละสีทำให้ต้องการหน่วยความจำสำหรับแทนแต่ละสี

2. การแปลงระบบสีโดยตรง (True color)

เมื่อค่าของสีค่าหนึ่งถูกใช้เป็นความเข้มหนึ่งของสีโดยตรง ซึ่งบ่อยครั้งที่จะหมายถึงสีจริง (true color) ซึ่งชื่อนี้อาจทำให้เข้าใจผิดเล็กน้อยเพราะว่ามันไม่ได้หมายความว่าค่าของสีนั้นเป็นค่าจริง แต่หมายถึงว่าค่าของสีนั้นแทนค่าของสีเลย โดยไม่ต้องใช้ตารางสี

2.5 การแปลงระบบสี RGB เป็นระบบสี YCbCr

เนื่องจากระบบสีในบิทแมพไฟล์ นั้นจะเป็นระบบ RGB แต่ระบบสีที่ใช้ใน JPEG นั้นจะเป็นระบบสี YCbCr ซึ่งจะประกอบด้วยการเปลี่ยนแปลงความเข้มของแสง (luminance) ซึ่งแทนด้วย Y = 0.30R + 0.59G +0.11B และการเปลี่ยนแปลงความเข้มของสี (chrominance) ซึ่งแทนด้วย Cb = 0.493(B-Y) และ Cr = 0.877(R-Y) โดยมีสมการการแปลงสีในระบบสี RGB เป็นระบบสี YCbCr ดังนี้

$$
\begin{aligned}
Y &= 0.299\ R + 0.587\ G + 0.114\ B \\
Cb &= -0.1687R - 0.3313\ G + 0.5\ B \\
Cr &= 0.5000R - 0.4187\ G - 0.0813\ B
\end{aligned}
$$

และสามารถเขียนให้อยู่ในรูปเมทริกต์(Matrix) ได้ดังนี้

$$
\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.1687 & -0.3313 & 0.5 \\ 0.5 & -0.4187 & -0.0813 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}
$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1.402 \\ 1 & -0.34414 & -0.71414 \\ 1 & 1.772 & 0 \end{bmatrix} \begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix}$$

โดย ค่า Y อยู่ในช่วง 0 ถึง 255

ค่า Cb อยู่ในช่วง 0 ถึง 255

ค่า Cr อยู่ในช่วง 0 ถึง 255

โดยการแปลงสีจากระบบสี RGB เป็นระบบสี YCbCr ของระบบสีในบิทแมพไฟล์มี 2 แบบคือ

2.5.1 การแปลงสีจากระบบ RGB ในบิทแมพไฟล์เป็นระบบสี YCbCr โดยใช้ตารางสี

เนื่องจากบิทแมพไฟล์ ชนิด 1, 4 และ 8 บิท/พิกเซล สีที่ใช้ทั้งหมดจะเก็บไว้ในตารางสี ซึ่งจะมีจำนวนสีเท่ากับ 2, 16, 256 สีตามลำดับ ซึ่งจุด pixel แต่ละจุดในบิทแมพไฟล์ ชนิดนี้จะเก็บค่าดัชนีของสีที่ใช้ในตารางสี ดังนั้นการแปลงสีจากระบบสี RGB ไปเป็นระบบสี YCbCr จะทำโดยการแปลงสีแต่ละสีตามลำดับในตารางสีซึ่งประกอบด้วยค่าสี R, G, B ไปเป็นค่าสี Y, Cb, Cr และเก็บไว้ในตารางสีตารางใหม่ที่สร้างขึ้นอีกตารางหนึ่ง ด้วยวิธีนี้ค่าดัชนีที่เก็บไว้ในจุดพิกเซลสามารถใช้เลือกสีในระบบสี YCbCr จากตารางสีใหม่ได้ทันที

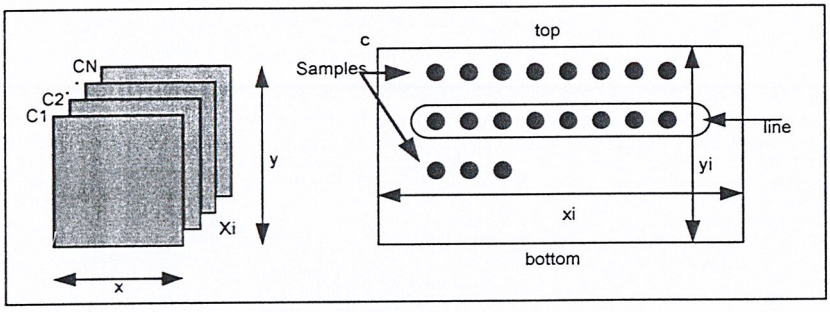
2.5.2 การแปลงสีจากระบบ RGB ในบิทแมพไฟล์เป็นระบบสี YCbCr โดยตรงซึ่งเรียกว่าสีจริง

เนื่องจากบิทแมพไฟล์ ชนิด 24 บิท/พิกเซล จะเก็บค่าสี R, G, B ไว้ในจุดพิกเซลโดยตรง ดังนั้นการแปลงระบบสีในบิทแมพไฟล์ ชนิดนี้จึงต้องทำการแปลงสีของจุด พิกเซล ทีละจุดโดยนำค่าสี R, G, B จากจุดพิกเซลมาทำการแปลงสีเป็นค่าสี Y, Cb, Cr และเก็บค่าสีที่ได้ไว้ตามตำแหน่งของจุดพิกเซล นั้น ๆ

2.6 การซับแซมปลิง (Subsampling)

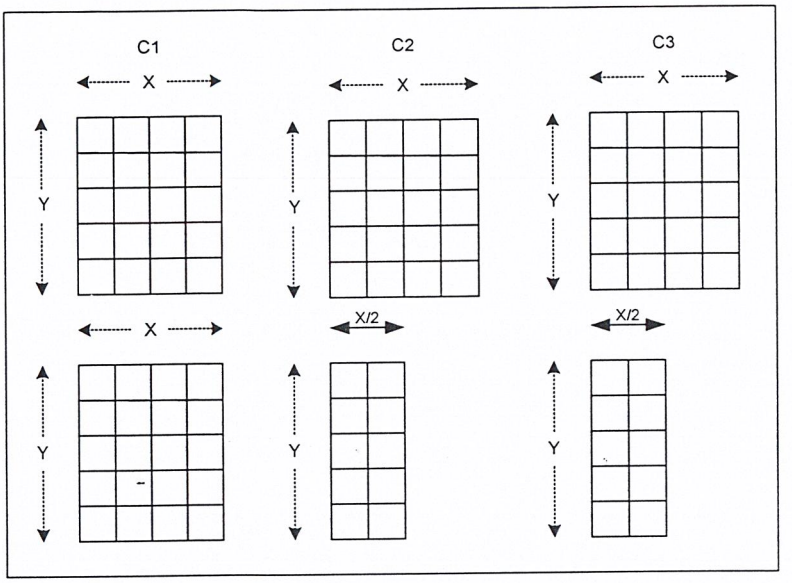
การซับแซมปลิง เป็นการลดจำนวนข้อมูลด้วยการแทนกลุ่มของข้อมูลด้วยค่าเฉลี่ยของกลุ่มข้อมูลนั้น วิธีของการซับแซมปลิง ข้อมูลสีสำหรับข้อมูลภาพดิจิตอลมีหลายวิธี วิธีหนึ่งถูกแสดงโดยชุดของค่า 3 ค่า ซึ่งแยกโดยเครื่องหมายโคลอน (:) ซึ่งค่าเหล่านี้จะแสดงจำนวนความสัมพันธ์ของตัวอย่างจากแต่ละส่วนประกอบของภาพ ค่าแรกแทน Y ซึ่งคือ การเปลี่ยนแปลงของความเข้มแสง ค่าที่สองแสดง Cr (R-Y) และค่าที่สามแสดง Cb (B-Y) ซึ่งค่าที่สองและสามคือ การเปลี่ยนแปลงของความเข้มสีสองรูปแบบที่ใช้กันโดยทั่วไปใช้อัตราส่วนการซับแซมปลิง เป็น 4:2:2 และ 4:1:1 รูปแบบ 4:2:2 แสดงว่าสำหรับทุก ๆ สี่ตัวอย่างของ ค่าการเปลี่ยนแปลงความเข้มของแสง จะมีสอง

ตัวอย่างของแต่ละแบบของ ค่าการเปลี่ยนแปลงความเข้มของสี ซึ่งอาจกล่าวได้ว่าทุก ๆ สอง ตัวอย่างของค่าการเปลี่ยนแปลงความเข้มของแสง มีหนึ่งตัวอย่างของแต่ละ ค่าการเปลี่ยนแปลงความเข้มสี ส่วนรูปแบบ 4:1:1 แสดงว่าสำหรับทุก ๆ สี่ตัวอย่างของ ค่าการเปลี่ยนแปลงความเข้มของแสงจะมีหนึ่งตัวอย่างของแต่ละ ค่าการเปลี่ยนแปลงความเข้มของสี ซึ่งในที่นี้เราใช้การซับแซมปลิง แบบ 4:1:1 ซึ่งการซับแซมปลิง สามารถทำได้ใน 2 ทิศทาง คือ ในแนวตั้ง (Vertical:V) และในแนวนอน (Horizontal:H) จากอัลกอริทึมของเจเป็กสำหรับภาพเกรย์สเกล ถ้าเรานำมาบีบอัดข้อมูลภาพสี อาจกล่าวได้ว่าจะต้องเป็นการบีบอัดข้อมูลแบบหลาย ๆ คอมโพเนนท์ของภาพ จากโมเดล (Model) ของภาพต้นฉบับของเจเป็กจะประกอบไปด้วย 1 ถึง 255 คอมโพเนนท์ ของภาพที่เรียกกันว่า สี หรือ แบนด์ ของสเปกตรัม (Spectrum Band) ดังรูปที่ 2.3



รูปที่ 2.3 แสดงคอมโพเนนท์ของภาพสี

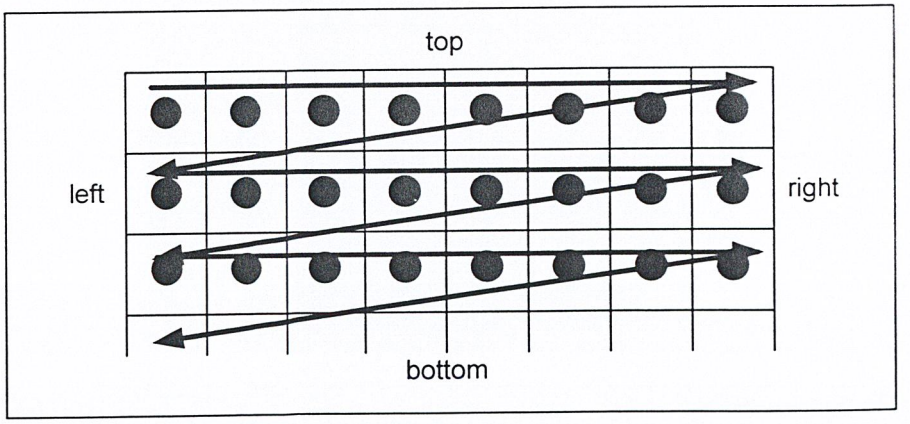
จากพื้นฐานของสีที่กล่าวว่า สีต่าง ๆ ประกอบขึ้นจากสีหลัก คือ แดง สีเขียว และสีน้ำเงิน ที่อัตราส่วนผสมต่าง ๆ กัน เราอาจพิจารณาได้ ตัวอย่างเช่น การแสดงผลแบบ RGB และ YCbCr ก็จะประกอบไปด้วย 3 คอมโพเนนท์ ซึ่งแต่ละคอมโพเนนท์อาจจะมีจำนวนของพิกเซลในแนวตั้ง (X_i) และแนวนอน (Y_i) ที่ไม่เท่ากันก็เป็นได้ ดังรูปที่ 2.4 แสดงให้เห็นถึง 2 กรณี สำหรับภาพสีที่มี 3 คอมโพเนนท์กรณีแรกทุก ๆ คอมโพเนนท์มีความละเอียดในการแสดงผลเท่ากัน แต่ในกรณีที่สอง แต่ละคอมโพเนนท์มีการแสดงผลที่ความละเอียดไม่เท่ากัน



รูปที่ 2.4 ความละเอียดของคอมโพเนนท์ในการประมวลผลข้อมูลคอมโพเนนท์

คอมโพเนนท์ต่าง ๆ ของสีสามารถถูกประมวลผลได้ 2 ลักษณะ ด้วยกันคือ

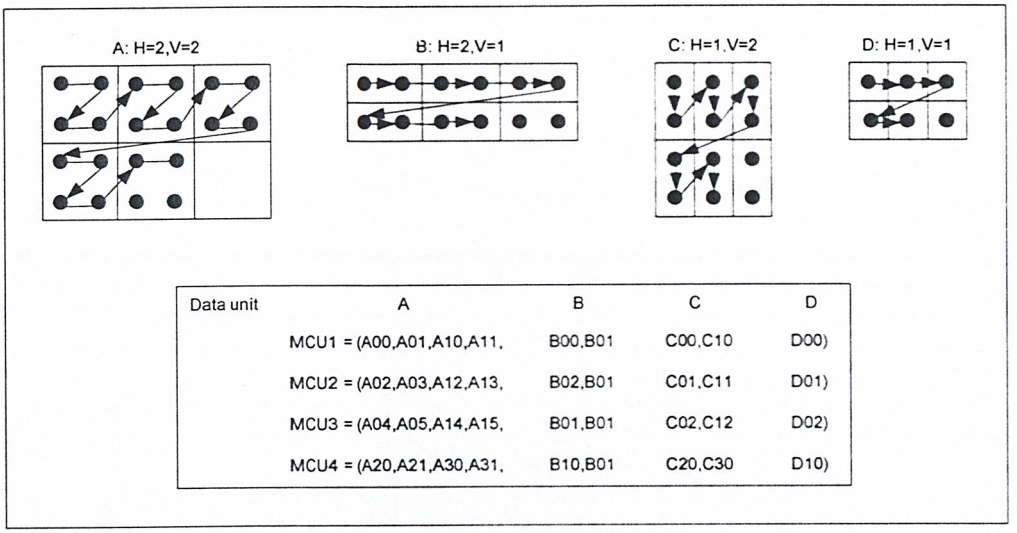
1. แบบลำดับการประมวลผลข้อมูลของแต่ละคอมโพเนนท์ตามลำดับ คือ จะทำการบีบอัดข้อมูลทีละคอมโพเนนท์ จากซ้ายไปขวา และจากบนลงล่าง สมมติเป็นโหมด RGB เราก็จะทำการบีบอัด R (สีแดง) ก่อน G (สีเขียว) และ B (สีน้ำเงิน) ตามกันไปเป็นลำดับ แสดงตามรูปที่ 2.5 ซึ่งในการแสดงผลการขยายข้อมูลกลับคืนนั้นจะเห็นหน้าจอเป็น สีแดงจนเต็มจอก่อน แล้วค่อยแสดงสีเขียว ซ้อนจนเต็มจอ และสีน้ำเงิน ซ้อนทับเป็นลำดับสุดท้าย เป็นต้น



รูปที่ 2.5 แสดงการประมวลผลข้อมูลแต่ละคอมโพเนนท์แบบไม่ซ้อนกันทับกัน

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

2. แบบประมวลผลแต่ละคอมโพเนนท์ไปพร้อม ๆ กัน โดยที่คอมโพเนนท์ต่าง ๆ เหล่านั้นจะถูกนำมารวมกัน เรียกว่า หน่วยที่ถูกเข้ารหัสแบบน้อยที่สุด (Minimum coded Units หรือ MCUs) วิธีนี้เหมาะสำหรับงานที่จะเป็นต้องแสดงผลหรือพิมพ์ พร้อม ๆ ไปกับกระบวนการขยายข้อมูลคืนในทันที เนื่องจากคอมโพแนนท์ต่าง ๆ จะต้องซ้อนทับเข้าด้วยกัน ดังนั้นจึงจำเป็นจะต้องทำการแบ่งแต่ละคอมโพเนนท์แบบสี่เหลี่ยม ตาม H_i และ V_i ตัวอย่างแสดงตามรูปที่ 2.6 ซึ่งภาพประกอบด้วย 4 คอมโพเนนท์ คือ $C_1C_2C_3C_4$ ซึ่งแต่ละคอมโพเนนท์ที่ไม่เท่ากัน



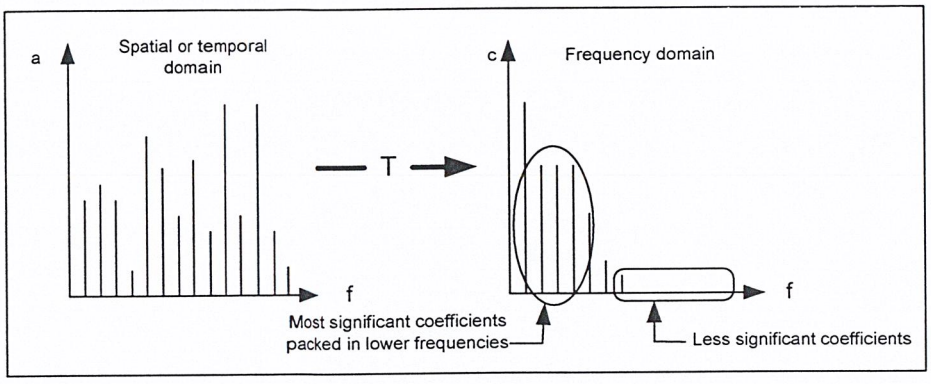
Data unit	A	B	C	D
	MCU1 = (A00,A01,A10,A11,	B00,B01	C00,C10	D00)
	MCU2 = (A02,A03,A12,A13,	B02,B01	C01,C11	D01)
	MCU3 = (A04,A05,A14,A15,	B01,B01	C02,C12	D02)
	MCU4 = (A20,A21,A30,A31,	B10,B01	C20,C30	D10)

รูปที่ 2.6 แสดงทิศทางการประมวลผลข้อมูลแต่ละคอมโพเนนท์ แบบที่ความละเอียดของแต่ละคอมโพเนนท์ไม่เท่ากัน

2.7 การแปลงแบบโคซายน์

การแปลงรูปสัญญาณก็คือการนำข้อมูลเริ่มต้นซึ่งเป็นข้อมูลที่อยู่ในสปาเชี่ยลโดเมน (Spatial domain) มาผ่านกระบวนการทางคณิตศาสตร์ เพื่อแปลงให้อยู่ในรูปของโดเมนอื่น ๆ ที่เหมาะสมสำหรับวิธีการบีบอัดนั้น ๆ ซึ่งโดยส่วนใหญ่แล้วการแปลงจะอยู่ในโดเมนของความถี่ กระบวนการแปลงแต่ละแบบสามารถทำกลับให้อยู่ในรูปแบบเดิมได้โดยการแปลงข้อมูลกลับ (Inverse transform) ดังตัวอย่างในรูปที่ 2.7

55722



รูปที่ 2.7 แสดงตัวอย่างการเข้ารหัสการแปลงจากโดเมนหนึ่งไปยังอีกโดเมนหนึ่ง (a เป็นค่าขนาดสัญญาณ และ c เป็นค่าขนาดกลุ่มพลังงาน)

การแปลงจากโดเมนสปาเชียลไปสู่โดเมนของความถี่ โดยหลังจากการแปลงแล้วจะได้สัมประสิทธิ์ที่มีนัยสำคัญสูง (Most significant coefficients) และนัยสำคัญต่ำ (Less significant coefficients)

รูปแบบของการแปลงมีด้วยกันหลายแบบ เช่น การแปลงแบบฟูริเยร์ (Fourier transform) การแปลงแบบโคซายน์ (Cosine transform) การแปลงแบบฮาร์ (Haar transform) การแปลงแบบฮาร์ดามาร์ด (Hadamard transform) เป็นต้น และรูปแบบที่นิยมมากในการนำมาใช้สำหรับการบีบอัดข้อมูลภาพก็คือ การแปลงแบบโคซายน์

การแปลงแบบโคซายน์เป็นการแปลงข้อมูลจากโดเมนของภาพ (Spatial domain) ไปเป็นโดเมนทางความถี่ (Frequency domain) โดยมีสมการการแปลงดังนี้

$$F(u) = \int f(x)\cos(2\pi fx)dx$$

ซึ่งจะเห็นได้ว่าการแปลงแบบโคซายน์คือ การแปลงแบบฟูเรียร์โดยที่ฟังก์ชั่น ที่จะนำมาทำการแปลงนั้นต้องเป็นฟังก์ชั่นคู่ซึ่งจะทำให้ผลการแปลงมีเฉพาะสัมประสิทธิ์ของฟังก์ชั่นโคซายน์ การแปลงดิสครีตโคซายน์ทรานฟอร์ม (DCT)

การแปลงโคซายน์นั้น เป็นวิธีที่นิยมมากในเรื่องของการบีบอัดข้อมูลที่ต้องการความรวดเร็วและมีประสิทธิภาพ วิธีการนี้เป็นกระบวนการของการบีบอัดข้อมูลที่สูญเสีย (Lossy compression) โดยวิธีที่ใช้เป็นมาตรฐานอยู่ขณะนี้ก็คือ JPEG (Joint Photographics Expert Group) โดยมีพื้นฐานการแปลงดีสครีตโคซายน์เป็นหลัก

เนื่องจากภาพโดยทั่ว ๆ ไปประกอบขึ้นจากฟังก์ชัน 2 ฟังก์ชันดังนั้นในการกระทำกระบวนใด ๆ กับภาพจำเป็นต้องทำในลักษณะ 2 แนวพร้อม ๆ กัน ในการทำการแปลงรูปสัญญาณก็

เช่นเดียวกันจำเป็นต้องทำใน 2 ทิศทางดังนั้นจึงต้องทำการแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่องในลักษณะ 2 ทิศทางด้วย การแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง 2 ทิศทาง คือ การนำภาพเริ่มต้นมาแบ่งเป็นบล็อกของพิกเซลขนาด 8×8 พิกเซล ซึ่งเป็นลักษณะของอาร์เรย์ของพิกเซล 2 มิติ โดยในแต่ละบล็อกของ 8×8 นั้นจะมีด้วยกันทั้งหมด 64 ค่า และแต่ละค่าจะแสดงขนาด (Amplitude) ของแต่ละค่าที่เข้ามาขนาดของสัญญาณนี้จะเป็นฟังก์ชันของจุดพิกัด 2 จุด โดยสมมุติให้ a = f(x,y) โดย x,y จะเป็นมิติของสปาเชียลโดเมน เมื่อพิจารณาอาร์เรย์ของพิกเซลขนาด 8×8 พิกเซล การแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง 1 ทิศทางจะถูกนำมาใช้แยกกันสำหรับแต่ละแถวและแต่ละหลัก การแปลงแบบดิสครีตโคซายน์ทรานสฟอร์มจะทำการแปลงข้อมูลในลักษณะของพิกเซลไปเป็นสัมประสิทธิ์ทางความถี่ ซึ่งผลจากการแปลงจะมีลักษณะเป็นจุด ๆ เช่นเดียวกับจุดพิกเซลก่อนที่จะทำการแปลงแต่จุดต่าง ๆ นี้จะแทนค่าสัมประสิทธิ์ทางความถี่แต่ละค่า สมการที่ใช้ในการแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง 2 ทิศทางมีสมการดังนี้

สมการการแปลงดีสครีตโคซายน์ (Forward DCT)

$$F(u,v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

สมการการแปลงกลับดีสครีตโคซายน์ (Inverse DCT)

$$f(i,j) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u,v) \cos\left[\frac{(2x+1)u\pi}{2N}\right] \cos\left[\frac{(2y+1)v\pi}{2N}\right]$$

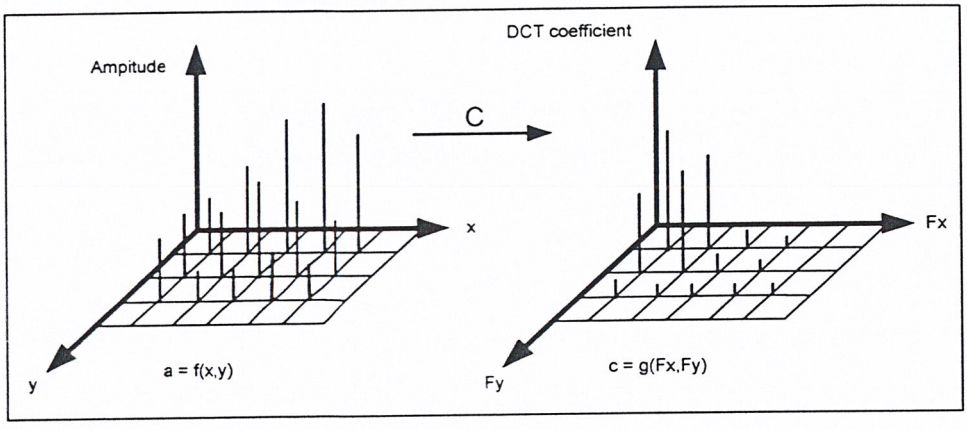
โดย

$$C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & ; u,v = 0 \\ 1 & ; other \end{cases}$$

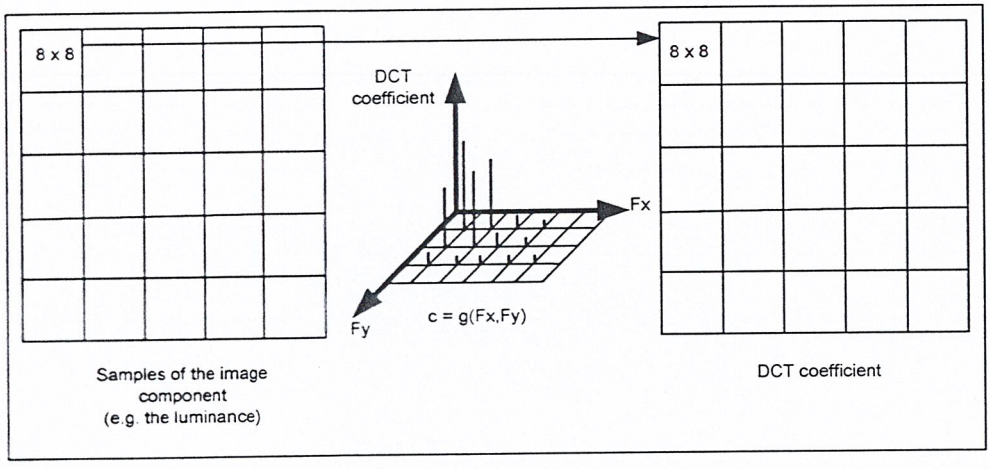
และ $f(i,j)$ เป็นข้อมูลภาพเริ่มต้นและภาพผลลัพธ์

$F(u,v)$ เป็นสัมประสิทธิ์ของการแปลง

โดยตัวอย่างในรูปที่ 2.8 และ 2.9 แสดงภาพในลักษณะ 3 มิติ ของการแปลงแบบดิสครีตโคซายน์ของบล็อกขนาด 8 แถว × 8 หลัก



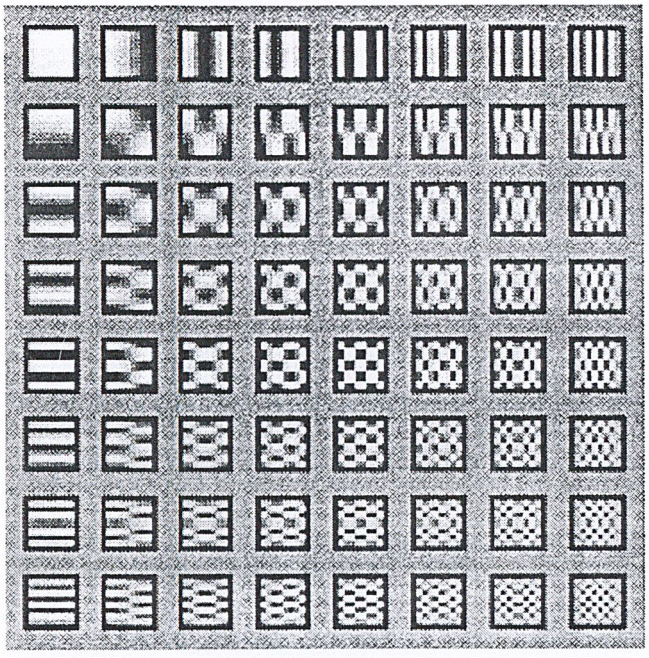
รูปที่ 2.8 แสดงการแปลงดีสครีตโคโซน์แบบ 2 มิติ



รูปที่ 2.9 แสดงการแปลงดีสครีตโคโซน์ที่ทำในบล็อกขนาด 8×8

หลังการผ่านกระบวนการแปลงด้วยดีสครีตโคไซน์มาจะได้รูปแบบของฟังก์ชันที่อยู่ในโดเมนของความถี่ ซึ่งจะได้สัมประสิทธิ์ทางความถี่ที่อยู่ในรูปของบล็อกขนาด 8×8 ค่า (8 แถว 8 หลัก) โดยสมมติให้มีฟังก์ชันเป็น $c = g(F_x, F_y)$ โดย c จะเป็นค่าของสัมประสิทธิ์ ส่วนค่าของ F_x และ F_y จะแสดงถึงค่าที่เกิดขึ้นในแกนของความถี่ในแต่ละทิศทาง สัมประสิทธิ์ของ g(0,0) นั้นจะเป็นสัมประสิทธิ์ที่ค่าความถี่ศูนย์ซึ่งจะเรียกว่าสัมประสิทธิ์ DC ซึ่งจะเป็นค่าเฉลี่ยของค่าในแต่ละบล็อก ค่าต่าง ๆ ในแถวที่ศูนย์จะมีส่วนประกอบทางความถี่เป็นศูนย์ในทิศทางหนึ่งและค่าทั้งหมดในคอลัมน์ที่ศูนย์ก็จะมีส่วนประกอบทางความถี่เป็นศูนย์ในอีกทิศทางหนึ่ง เมื่อแถวและคอลัมน์เพิ่มขึ้นในทิศทางที่ห่างออกจากจุดเริ่มต้นสัมประสิทธิ์ในเมทริกซ์ที่ได้จากการแปลงแบบดิสครีตโคซายน์ทรานสฟอร์มจะแทนความถี่ที่สูงขึ้น และความถี่ที่สูงที่สุดคือที่ตำแหน่ง N-1 ของบล็อกข้อมูล

ซึ่งโดยปกติพิกเซลจะมีการเปลี่ยนแปลงน้อยลงจากจุดถึงจุด ดังนั้นขนาดของความถี่ที่ต่ำที่สุดจะสูงที่สุดส่วนค่าของความถี่กลางและสูงจะมีค่าที่น้อยหรืออาจจะเป็นศูนย์ซึ่งในส่วนนี้จะไม่นำมาใช้อาจจะตัดทอนลงไปดังจะเห็นจากรูปที่ 2.9 ซึ่งจะเป็นประโยชน์ต่อการลดขนาดของภาพลงได้ในขั้นตอนของการควอนไตเซชัน และเมื่อนำแต่ละสัมประสิทธิ์ทางความถี่มาผ่านการแปลงกลับดิสครีตโคซายน์โคซายน์แบบไม่ต่อเนื่องแบบ 2 ทิศทางแล้ว จะได้ค่าแอมปลิจุดเป็นพื้นที่ที่มีความขาวดำด้วยอัตราความคมชัด (Constrast Ratio) ซึ่งกำหนดโดยแอมปลิจุดซึ่งในรูปที่ 2.10 แสดง ถึงแพทเทิร์นที่สร้างจากค่าสัมประสิทธิ์ทางความถี่ของการแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่องของ 8×8 พิกเซล

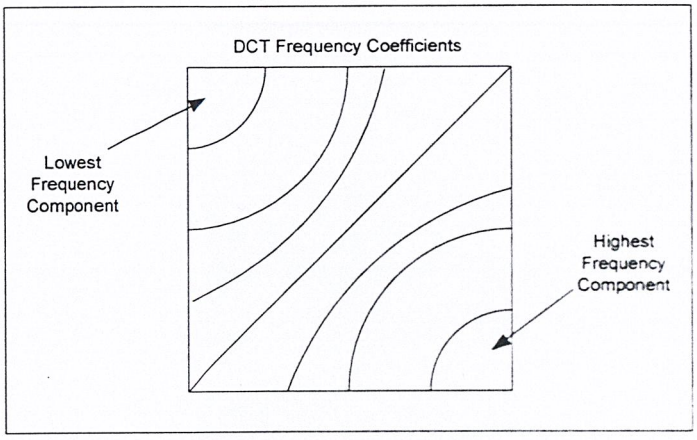


รูปที่ 2.10 แสดงแพทเทิร์นของความเข้มของสัมประสิทธิ์การถ่วงน้ำหนักการแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง

สมมุติว่าแต่ละค่าเป็นค่าสูงที่สุดแล้ว และจะเห็นได้ว่าบล็อกบนซ้ายมือจะเป็นสีเทาทึบแสดงถึงค่าเฉลี่ยของความเข้มทั้งบล็อกขนาด 8×8 พิกเซล ส่วนบล็อกที่อยู่มุมล่างขวามือจะมีลักษณะเป็นตารางหมากรุก เพราะมีส่วนคอมโพเนนท์ทางความถี่ที่สูงสุดตามแนวตั้งและแนวนอน บล็อกที่มุมขวามือด้านบนและซ้ายมือด้านล่างแสดงแถบตามแนวนอนและแนวตั้งตามลำดับ เพราะมีความถี่สูงสุดทางแนวนอนแต่ต่ำสุดตามแนวตั้งและความถี่สูงสุดทางแนวตั้งแต่ต่ำสุดทางแนวนอนตามลำดับ ดังรูปที่ 2.10

การแบ่งภาพต้นฉบับออกเป็นบล็อกของพิกเซลที่ขนาดเล็กลงจะทำให้จำนวนครั้งในการคำนวณลดลงอย่างมาก แต่ข้อเสียที่อาจได้รับ คือ กระบวนการสร้างภาพกลับคืนอาจมองเห็นผลต่างของแต่ละบล็อกได้อย่างชัดเจน และในทางปฏิบัติ ดีกรีในการควอนไตซ์ ที่ใช้กับคอมโพเนนท์ทางความถี่นั้น ขึ้นอยู่กับจำนวนของเฟกเตอร์ที่รวมไว้ในคุณลักษณะของภาพ อัตราการลดขนาดข้อมูลที่ต้องการ และคุณภาพของภาพที่ต้องการหลังจากทำการสร้างภาพกลับคืนมา สำหรับภาพสีจะทำการควอนไตซ์ส่วนที่เป็นสีมากกว่าส่วนของความสว่างและเมื่อมีการคำนวณ การแปลงโคซายน์แบบไม่ต่อเนื่องให้กับบล็อกของภาพมักใช้เลขจำนวนเต็มมากกว่าทศนิยมเพราะใช้เวลาในการคำนวณและความยุ่งยากในการคำนวณน้อยกว่า

ในรูปที่ 2.11 จะแสดงถึงการกระจายของข้อมูล (คอมโพเนนท์ทางความถี่) ที่ผลลัพธ์จากการแปลงโคซายน์



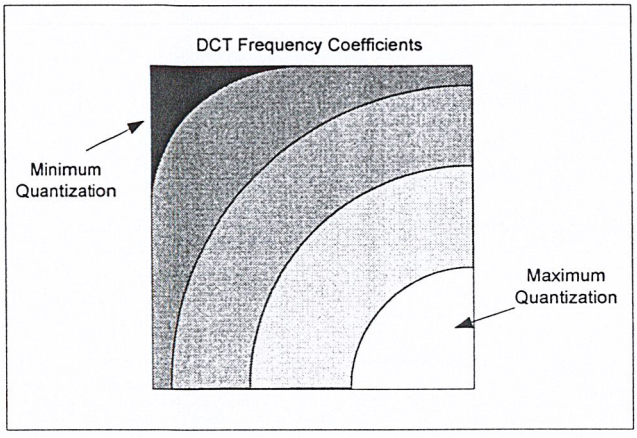
รูปที่ 2.11 การกระจายของส่วนประกอบทางความถี่ (ประมาณ) ของการแปลงดิสครีตโคซายน์

2.8 การควอนไตซ์ (Quantized)

การควอนไตซ์ (Quantized) จะเป็นการลดจำนวนบิตที่จำเป็นที่จะใช้เก็บเลขจำนวนเต็ม ของค่าสัมประสิทธิ์ที่ได้จาก การแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง ซึ่งจะทำเป็นเมทริกซ์ โดยที่ทุก ๆ อีเลเมนท์ในเมทริกซ์จะมีค่าที่ใช้ในการควอนไตซ์เป็นค่าเฉพาะ เรียกว่าค่าสัมประสิทธิ์การควอนไตซ์หรือค่าควอนตัม (Quantum) ลักษณะเป็นค่าที่มากขึ้นตามแนวเส้นทแยงมุม ตามรูปที่ 2.12 หรือเป็นแบบซิกแซก (zigzag) โดยประมาณ โดยที่ค่าควอนตัมจะเป็นค่าที่บอกขนาดของสเตป ในการควอนไตซ์ ซึ่งส่วนใหญ่มีค่า 1 – 256

อีเลเมนท์ที่อยู่ด้านมุมบนซ้ายมือ จะมีส่วนประกอบสำคัญของภาพ จะถูกควอนไตซ์ด้วยสเตปแคบ ๆ (ให้ค่าผิดพลาด (Error) น้อย) เพราะต้องการความถูกต้องและความละเอียดมาก ส่วนที่

ตำแหน่งที่ห่างออกไปจากจุดนี้ตามแนวทแยงมุมค่าสเตปจะกว้างขึ้น เพื่อปัดค่าส่วนที่ไม่สำคัญของภาพให้ใกล้เคียงหรือเท่ากับ 0 และที่มุมขวาล่างจะมีสเตปกว้างที่สุด เพราะค่าผิดพลาดที่เกิดขึ้นมีผลต่อภาพน้อย



รูปที่ 2.12 แสดงการกระจายค่าของสัมประสิทธิ์ที่ใช้ในการควอนไตซ์

$$(\text{Quantized value})(i,j) = \frac{\text{ค่าที่ได้จากการแปลงดิสครีตโคซายน์แบบไม่ต่อเนื่อง } (\text{DCT } (i,j))}{\text{ค่าควอนตัม}(i,j)}$$

ในกระบวนการของการดีควอนไตซ์ (Dequantized) จะทำการหาค่าสัมประสิทธิ์ทางความถี่โดย

$$(\text{DCT } (i,j)) = \text{ค่าที่ได้จากการควอนไตซ์ } (i,j) \times \text{ค่าควอนตัม } (i,j)$$

2.8.1 การเลือกเมทริกซ์ที่จะนำมาทำการควอนไตซ์

ค่าควอนตัมมีค่ามากจะสามารถลดขนาดข้อมูลได้มากแต่ภาพจะผิดเพี้ยนมาก ในขณะที่ค่าควอนตัมน้อย ๆ จะลดขนาดได้น้อยแต่ผิดเพี้ยนน้อย เพื่อให้คุณภาพของภาพยืดหยุ่นได้ตามความต้องการของผู้ใช้จึงมีการกำหนดค่าเฟกเตอร์คุณภาพ (Quality Factor) ปกติมีค่า 1 – 25 (ค่าที่มากกว่านี้ภาพจะเพี้ยนมากเกินที่จะยอมรับได้) ซึ่งค่านี้จะถูกนำไปคำนวณเพื่อเลือกค่าควอนตัมที่เหมาะสมในการควอนไตซ์ ตามคุณภาพที่ต้องการ

ค่าเฟกเตอร์คุณภาพจะมีค่าความสัมพันธ์กับค่าควอนตัมตามแนวทแยงมุมของเมทริกซ์ แสดงตามรูปที่ 2.13 โดยที่สมการที่ใช้ในการคำนวณค่าควอนตัมจากค่าแฟกเตอร์คุณภาพคือ

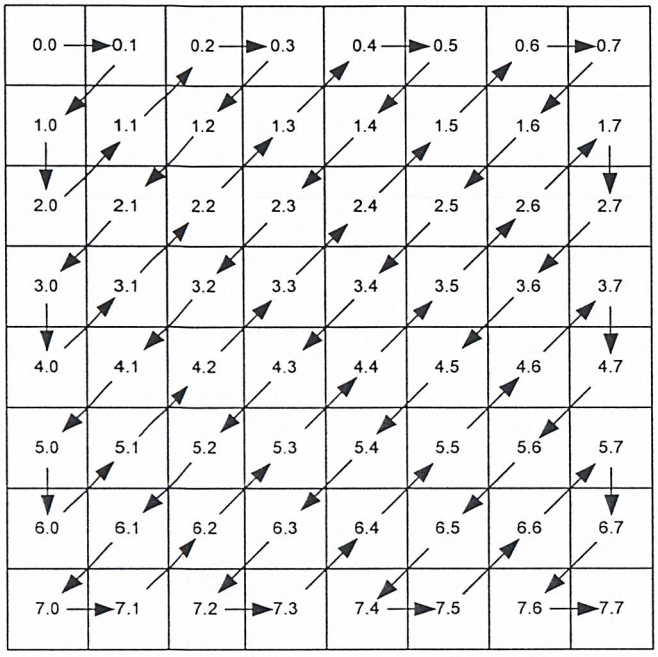
$$\text{ค่าควอนตัม } (i,j) = 1 + [(1 + i + j) * \text{ค่าเฟกเตอร์คุณภาพ}]$$

3	5	7	9	11	13	15	17
5	7	9	11	19	15	17	19
7	9	11	13	15	17	19	21
9	11	13	15	17	19	21	23
11	13	15	17	19	21	23	25
13	15	17	19	21	23	25	27
15	17	19	21	23	25	27	29
17	19	21	23	25	27	29	31

รูปที่ 2.13 แสดงเมทริกซ์ของค่าควอนตัมที่แฟกเตอร์คุณภาพเท่ากับ 2

2.9 การควอนไตซ์เป็นลำดับแบบซิกแซก

เจเป็ก (JPEG) เป็นวิธีทีทำให้ส่วนประกอบทางความถี่สูงเป็น 0 เป็นส่วนใหญ่ จึงเป็นการสะดวกที่จะเข้ารหัสด้วยวิธีอาศัยความต่อเนื่องของข้อมูล ขบวนการควอนไตซ์ (Quantization) เป็นลำดับแสดงตามรูปที่ 2.14 ซึ่งก็คือการอ่านข้อมูลแบบซิกแซก (zigzag scan) เป็นการอ่านข้อมูลที่ได้จากการควอนไตเซชั่นซึ่งอยู่ในรูปแบบของบล็อกขนาด 8 แถว 8 คอลัมน์ ให้อยู่ในรูปของชุดข้อมูลที่เรียงต่อกันไป โดยลักษณะของการอ่านข้อมูลนี้ก็เพื่อให้สอดคล้องกับข้อมูลที่ได้จากการแปลงแบบดิสครีตโคซายน์ทรานสฟอร์มและการควอนไตเซชั่น ผลที่ได้คือข้อมูลก็จะลดลงตามแนวเส้นทแยงมุมของบล็อก และในช่วงความถี่สูง ๆ ค่าจะถูกลดลงเป็นศูนย์มาก ดังนั้นการอ่านข้อมูลแบบซิกแซกก็มีความพยายามที่จะนำค่าที่เป็นค่าศูนย์ มาเรียงให้ติดกันเพื่อประโยชน์ในการบีบอัดข้อมูลขั้นต่อไป



รูปที่ 2.14 แสดงทิศทางการจัดลำดับแบบซิกแซก

2.10 การเข้ารหัสเจเป็ก (JPEG Coding)

กระบวนการสุดท้ายของเจเป็ก คือ การเข้ารหัสข้อมูลที่ผ่านการควอนไตซ์แล้วด้วยกระบวนการ 3 ขั้นตอน

1. การเก็บค่าสัมประสิทธิ์กระแสตรงของบล็อกพิกเซล (8×8) บล็อกแรกก่อน แล้วทำการเปลี่ยนค่าสัมประสิทธิ์กระแสตรงที่จุด (0, 0) ของบล็อกต่อมาเป็นค่าที่สัมพัทธ์กับค่าแรกหรือเป็นค่าผลต่างนั่นเองแล้วจึงทำการเข้ารหัสผลต่างนี้แทน เพื่อประโยชน์คือสามารถเข้ารหัสโดยใช้ผลต่างที่ถือว่าน้อย ด้วยจำนวนบิทที่น้อยกว่าการเข้ารหัสค่าจริง (เพราะจะถือว่าบล็อกของพิกเซลที่อยู่ใกล้กันจะมีส่วนที่แตกต่างกันน้อยมาก และสัมประสิทธิ์กระแสตรงก็ได้รวมข้อมูลที่สำคัญของภาพไว้แล้ว)

2. การจัดลำดับสัมประสิทธิ์กระแสสลับเป็นลำดับ แบบซิกแซก เพื่อให้ค่าศูนย์ที่เกิดขึ้นจากกระบวนการควอนไตซ์มีความต่อเนื่องกันยาว ๆ ทำให้สามารถเข้ารหัสได้ง่าย

3. เลือกทำการเข้ารหัสสัมประสิทธิ์กระแสสลับที่จัดลำดับแล้ว อาจเป็น

3.1 ด้วยวิธีความต่อเนื่องของข้อมูล RLE (Run-Length Encoding) ซึ่งข้อดีของวิธีนี้คือถ้าค่าศูนย์ที่เรียงติดกันมีความต่อเนื่องกันยาวมาก ๆ จะทำให้เข้ารหัสได้ง่าย

3.2 การเข้ารหัสเอนโทรปี (Entropy Encoding) ซึ่งทำได้โดยวิธีการฮัฟแมน (Huffman) หรือการประมวลผลทางคณิตศาสตร์ (Arithmetic) ก็ได้

2.10.1 การเข้ารหัสเอนโทรปี (Entropy Encoding)

เป็นกรรมวิธีบีบอัดข้อมูลที่ไม่มีการสูญหาย (Lossless Compression) ใช้พื้นฐานจากคุณสมบัติทางสถิติของรูปภาพหรือกระแสข้อมูลที่ถูกบีบอัด พื้นฐานของเอนโทรปีประกอบด้วยการเข้ารหัสรูปแบบที่เกิดขึ้นบ่อยบนตารางด้วยจำนวนบิตที่น้อยที่สุด ด้วยวิธีนี้ข้อมูลจะถูกบีบอัดด้วยอัตราที่เพิ่มขึ้นอีก 3 หรือ 4 เท่า การเข้ารหัสเอนโทรปีสำหรับงานการบีบอัดวีดีโอมีสองขั้นตอน คือ Zero Run-Length Coding (RLC) และ Huffman Coding หลังจากที่ได้ลำดับข้อมูลมาแล้วเราจะเห็นได้ว่าลำดับข้อมูลด้านท้ายทั้งหมดที่มีค่าเป็นศูนย์ตลอดโดยไม่มีค่าอื่นคั่นนั้นสามารถละทิ้งได้โดยใส่เครื่องหมาย End of Block (EOB) ปิดท้ายได้เลยตัวถอดรหัสจะรู้ว่าต้องนำค่าศูนย์ไปต่อท้ายส่วนที่เหลือให้เต็มบล็อกเอง ในกรณีที่ค่าสัมประสิทธิ์ท้าย ๆ ยังมีค่าอื่นหลงเหลืออยู่ทำให้ไม่สามารถตัดช่วงท้ายทิ้งไปได้ การใช้ Run length Encoding แทนค่าสัมประสิทธิ์ที่มีค่าเดียวกันซึ่งอยู่ติดกันจะช่วยลดข้อมูลลงโดยไม่สูญเสียคุณภาพ (Lossless) การลดข้อมูลแบบ Run length จะแทนด้วยสัญลักษณ์ (Skip , Value) โดยที่ Skip หมายถึงค่าต่อไปที่ไม่ใช่ศูนย์ เช่น รหัส RLC (5,7) จะแทนลำดับข้อมูลด้วย (0, 0, 0, 0, 0, 7) การเข้ารหัสแบบ ฮัฟฟ์แมน (Huffman) เป็นการให้รหัสที่มีความยาวแปรผันได้กับข้อมูลที่ผ่าน RLC มาแล้ว จึงเกิดข้อมูลกระแสบิตความยาวแปรผันด้วยตารางฮัฟฟ์แมน (Huffman)สามารถคำนวณล่วงหน้าได้โดยอาศัยคุณสมบัติทางสถิติของภาพ (ใน JPEG) หรือสามารถกำหนดล่วงหน้าได้หากใช้ตารางที่ถูกเลือกไว้แล้ว (Sent in Header) และตารางนี้จะนำมาใช้ในขั้นตอนการถอดรหัสของบิตเหล่านี้ด้วย

2.10.2 การเข้ารหัสแบบรันเลนจ์ (Run Length)

เป็นวิธีการลดความซ้ำซ้อนของข้อมูล (Interpixel Redundancy) สำหรับการข้อมูลที่เป็นแบบหนึ่งมิติ (1-D) จะสามารถอธิบายได้คือในกรณีที่ระดับความสว่างของภาพมีค่าคงที่ติดต่อกันค่ารหัสที่ใช้แทนข้อมูลระดับแสงสว่างนี้ก็จะมีค่าซ้ำ ๆ กันไป ในการเก็บรหัสเหล่านี้ลงไปทั้งหมดก็จะทำให้ใช้เนื้อที่มาก วิธีการของรันเลนจ์ (Run Length Coding) จะเก็บข้อมูลนี้ในลักษณะของรหัสระดับความสว่างกับจำนวนที่อยู่ติดกันเป็นคู่ลำดับของ (รหัสข้อมูล, จำนวน)

อย่างไรก็ตามรหัสที่ได้หลังจากผ่านขบวนการทำ Run-Length Coding แล้วอาจมีขนาดหรือจำนวนที่มากขึ้นกว่าเดิมก็ได้ ถ้าภาพนั้นมีรายละเอียดมากซึ่งข้อมูลภาพเหล่านี้ค่าระดับความสว่างของภาพจะไม่คงที่หรือคงที่ติดต่อกันไม่มากนัก

ดังนั้นในการใช้ Run-Length Coding โดยทั่วไปจะใช้กับข้อมูลเกี่ยวกับระดับความสว่างของภาพ ซึ่งได้แปลงให้อยู่ในรูปแบบที่เหมาะสมแล้ว จึงจะให้อัตราส่วนการลดข้อมูลที่น่าพอใจ

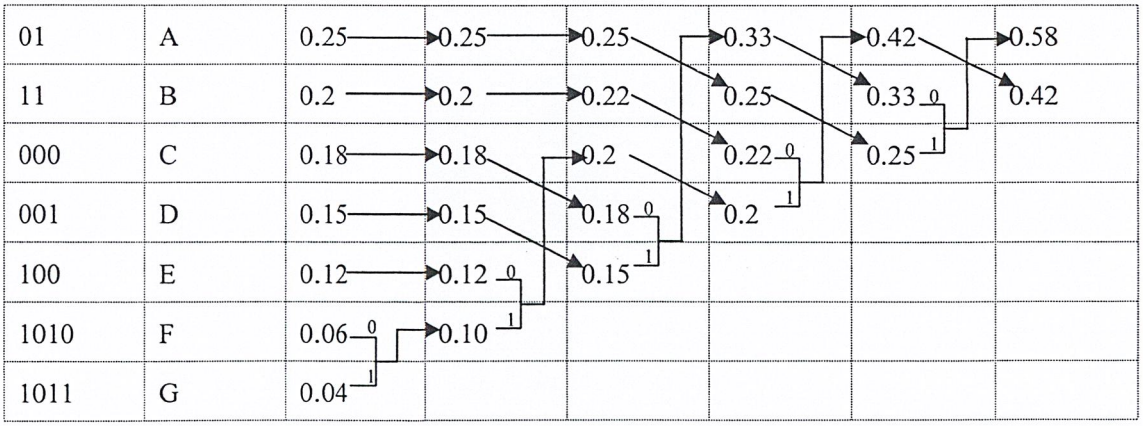
2.10.3 การเข้ารหัสฮัฟแมน (Huffman Coding)

เป็นวิธีการเข้ารหัสแบบความยาวแปรได้ที่ให้ขนาดความยาวบิตเฉลี่ยที่ดีที่สุด การเข้ารหัสฮัฟแมน (Huffman Coding) จะทำการสร้างรหัสจากปลายกิ่งของต้นไม้ไปหารากซึ่ง ฮัฟแมน (Huffman) ได้กำหนดให้รหัสด้วยความยาวบิตมากสำหรับสัญลักษณ์ที่เกิดน้อยและแทนรหัสด้วยความยาวบิตน้อยสำหรับสัญลักษณ์ที่เกิดมากและแทนรหัสของสัญลักษณ์ที่เกิดน้อยที่สุด 2 ตัวสุดท้ายด้วยบิตที่มีความยาวมากที่สุดโดยมีบิตต่างกัน 1 บิต ซึ่งมีขั้นตอนการทำดังนี้

ถ้าให้ความน่าจะเป็นในการที่จะพบ ค่าระดับความสว่างต่าง ๆ ในข้อมูลรูปภาพมีขนาด m ตัว

1. เรียงลำดับสัญลักษณ์ตามความน่าจะเป็นนี้จากมากไปน้อย
2. กำหนดค่า 0 และ 1 ให้กับสัญลักษณ์ที่มีความน่าจะเป็นน้อยที่สุด 2 ตัวสุดท้ายตามลำดับ ทำการสร้างสัญลักษณ์ใหม่ขึ้นโดยได้จากการรวมความน่าจะเป็นของสัญลักษณ์ 2 ตัวที่แทนค่าไบนารี่ไปแล้วเข้าด้วยกัน
3. ทำการเรียงลำดับสัญลักษณ์ตามขนาดความน่าจะเป็นในการเกิดใหม่
4. กระทำซ้ำข้อ 2 , 3 จนกระทั่งเหลือสัญลักษณ์ 2 ตัว ค่ารหัสได้จากการอ่านค่า ไบนารี 0 หรือ 1 บนกิ่งที่กำหนดให้แต่ละสัญลักษณ์ในข้อ 2 โดยทำการอ่านตั้งแต่การรวมครั้งสุดท้ายถึงสัญลักษณ์เริ่มต้น

ตัวอย่าง



ค่าเฉลี่ยของบิทต่อสัญลักษณ์คือ

$(0.25\times2) + (0.20\times2) + (0.018\times3) + (0.15\times3) + (0.12\times3) + (0.06\times4) + (0.04\times4) = 2.65$ bits

วิธีการของ Huffman coding นี้ จะเป็นวิธีการที่มีประสิทธิภาพมากในการสร้างรหัสแทนข้อมูลที่ต้องการหลังจากที่หารหัสของ Huffman ได้แล้วขบวนการเข้ารหัสข้อมูลหรือถอดรหัสข้อมูลสามารถทำได้โดยการเปิดตารางเทียบข้อมูลที่ต้องการกับรหัสที่ได้

ข้อมูลที่เข้ารหัสแบบ Huffman นี้จะมีลักษณะที่พิเศษคือ ในการถอดรหัสข้อมูลจะสามารถทำได้โดยที่ไม่ต้องมีข้อมูลอ้างอิงขึ้นระหว่างรหัสแต่ละตัว รหัสของ Huffman นี้จะสามารถแยกออกมาได้โดยไม่ต้องซ้ำกันและจะแยกรหัสนี้ได้เพียงวิธีเดียวเท่านั้นจากการถอดรหัสจากซ้ายไปขวา

2.11 หลักการบีบข้อมูลวีดีโอและมาตรฐาน MPEG

พื้นฐานการเข้ารหัสสัญญาณวีดีโอ

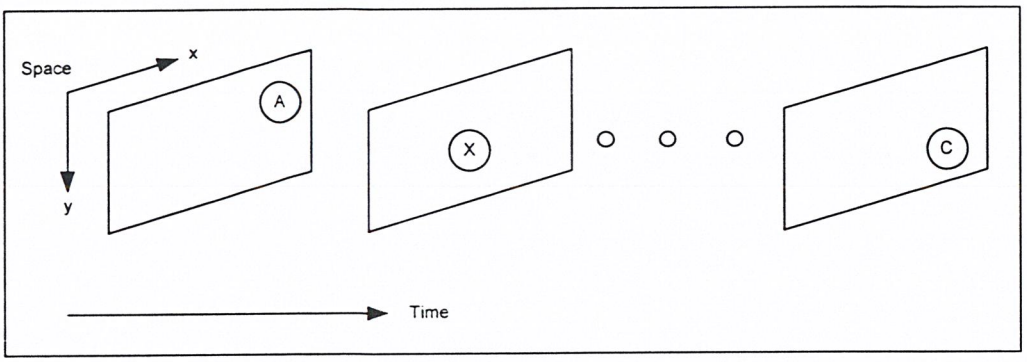
2.11.1. รูปแบบของการลำดับภาพนิ่ง (Image sequence Model)

ในภาพนิ่งจะพบว่าข้อมูลนั้นมีส่วนของที่ว่างจำนวนมาก (spatial redundancy) ถ้าหากเราต้องการจับภาพการเคลื่อนไหวของวัตถุ 3 มิติ เป็นดังรูปที่ 2.15 จากรูปที่ 2.15 ในภาพแรกจะจับรายละเอียดภาพทั้งหมดที่รวมทั้งวัตถุและที่ว่าง และในภาพถัด ๆ ไป เราคาดได้ว่าต้องมีส่วนที่เหมือนกัน และถ้าวัตถุเคลื่อนที่ไป ภาพอื่นก็จะมีที่ว่างเช่นกันและเราก็จะคาดได้ว่าจะต้องมีการซ้ำกันของภาพในแต่ละเฟรมในบางพื้นที่ ที่เรียกว่า temporal redundancy โดยเป็นเป้าหมายของ Video compression คือการหาประโยชน์จากทั้ง spatial redundancy และ temporal redundancy เพื่อให้ได้การบีบอัดที่ดีที่สุด ส่วนใหญ่แล้วในระบบเข้ารหัส Video จะมี 2 stage ในการทำให้ได้การบีบอัดที่ดีที่สุดคือ

1. กระบวนการสำหรับการลดที่ว่างที่ไม่จำเป็นที่เกิดขึ้นชั่วคราว (Processing for reducing temporal redundancy)
2. กระบวนการสำหรับการลดที่ว่างที่ไม่จำเป็น (Processing for reducing spatial redundancy)

2.11.2. การลดที่ว่างที่ไม่จำเป็นที่เกิดขึ้นชั่วคราว (Reducing Temporal Redundancy)

การประมวลผลในทางอุดมคติที่จะทำการลด Temporal redundancy ก็คือการเข้าถึงทุก ๆ จุดพิกเซล จากเฟรมหนึ่งไปยังอีกเฟรมหนึ่ง แต่ว่าการกระทำเป็นพิกเซลนั้นย่อมมี noise เกิดขึ้นด้วย ดังนั้นแทนที่เราจะทำเป็นพิกเซล ๆ เราจัดแบ่งข้อมูลเป็นพื้นที่ ขนาด 16 ×16 ที่เรียกว่า มาโครบล็อก (macroblocks) ซึ่งเป็นขนาดที่เหมาะสมกับประสิทธิภาพในการลด Temporal redundancy และใช้พลังการคำนวณในระดับพอเหมาะ



รูปที่ 2.15 Temporal correlation ในภาพที่ต่อเนื่องกัน

ถ้าเรามีสองเฟรมต่อกันดังรูปที่ 2.15 โดยสมมติว่าเป็น frame (t-l) และ frame(t) ในขั้นแรกเราจะทำการแยก frame(t) ไปเป็น พื้นที่ขนาด 16×16 (macroblock) โดยไม่ซ้อนทับกัน จากนั้นเราจะคำนวณ block 16×16 ที่สอดคล้องกันของ frame(t-l) ซึ่งโดยใช้การคำนวณที่ได้นี้ ขบวนการ Temporal redundancy reduction จะสร้างตัวแทนของ frame(t) ซึ่งมีเพียงส่วนที่เป็นความแตกต่างของทั้งสองเฟรม ถ้าหากทั้งสองเฟรมมีความเหมือนกันมาก (High degree of temporal redundancy) ดังนั้นจะทำให้เฟรมที่แสดงความแตกต่าง (different frame) ได้ค่าที่ใกล้เคียง 0 จำนวนมาก นั่นหมายถึงมันจะยอมให้ถูกบีบได้มากขึ้นแต่ถ้าหากว่า frame(t) กับframe(t-l) ต่างกันโดยสิ้นเชิง มันจะไม่สามารถหา different frame ที่สอดคล้องกันได้ และเราก็จะไม่ได้ประโยชน์อะไรจากวิธีนี้ ในระบบการบีบอัดข้อมูลภาพเคลื่อนไหวนั้น วิธีการบีบอัดที่ใช้ในการลด Temporal redundancy เราจะอ้างถึงกระบวนการที่ เรียกว่า Interframe coder

2.11.3. การลดที่ว่างที่ไม่จำเป็น (Reducing Spatial redundancy)

หลังจากที่ เราได้เฟรมที่แสดงถึงความแตกต่างของ 2 เฟรมติดกัน ทำให้เราจะสามารถหาประโยชน์จากพิกเซลที่เหมือนกันเหล่านั้นได้ เพื่อให้ได้ผลของการบีบอัดที่เป็นตัวแทนของ Frame (t) โดยกระบวนการนี้จะแสดงในขั้นตอนที่ 2 (stage 2) มาตรฐานการเข้ารหัสภาพเคลื่อนไหวจะใช้การเข้ารหัสแบบ DCT เพื่อลดที่ว่างที่ไม่จำเป็น เราจะเรียกว่า ตัวเข้ารหัสภายในเฟรม (Intra coder) การรวมเอาการเข้ารหัสระหว่างเฟรม Interframe coding และการเข้ารหัสภายในเฟรม (Intraframe coding) เข้าด้วยกัน นั้น เราเรียกว่าการเข้ารหัสแบบไฮบริด (ภายในเฟรม / ระหว่างเฟรม)[a hybrid (intraframe / interframe) coding method]

2.11.4. การชดเชยการเคลื่อนไหว (Motion Compensation)

จากกระบวนการที่เปรียบเทียบเฟรม 2 เฟรมที่ติดกันเพื่อหาความแตกต่างของ 2 เฟรม เราอ้างถึงการทำนายที่ต้องใช้การชดเชยการเคลื่อนที่ โดยเราจะสามารถอธิบายการชดเชยการเคลื่อนที่ได้ว่าเป็นกระบวนการที่ชดเชย ระยะทางของวัตถุที่เคลื่อนไหวจากเฟรมหนึ่งไปยังอีกเฟรมหนึ่ง ในทางปฏิบัติการชดเชยการเคลื่อนที่จะถูกดำเนินการด้วยการประมาณค่าการเคลื่อนที่ (motion estimation) ซึ่งเป็นกระบวนการที่หาพิกเซลที่สอดคล้องกันระหว่างเฟรมที่ต้องการ โดยสมการการชดเชยการเคลื่อนที่สารมารถเขียนได้เป็น

$$e(x, y, t) = I(x, y, t) - I(x - u, y - v, t - 1)$$

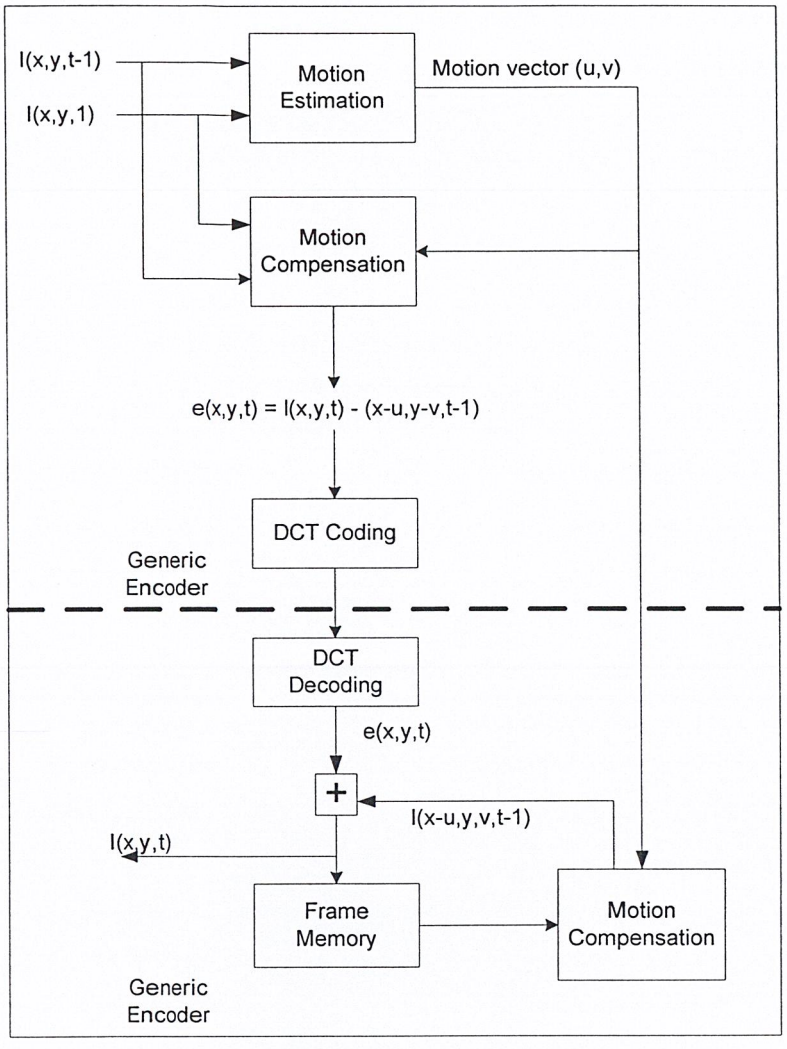
โดยที่ I(x, y, t) คือ ค่าของพิกเซลในที่ตำแหน่ง(x, y) ใน frame (t)

I(x – u, y – v, t - 1) คือพิกเซลที่สอดคล้องกับ I(x, y, t) ที่ตำแหน่ง (x – u, y – v) ที่ frame (t-l)

จากกระบวนการนี้จะได้เอาท์พุตที่เป็นความสัมพันธ์ของการเคลื่อนไหวในหนึ่ง Block ของเฟรมหนึ่งกับอีกเฟรมหนึ่ง ในรูปเวคเตอร์ (u, v) จากรูปที่ 2.16

แสดงบล็อกไดอะแกรม (Block diagram) ของการเข้ารหัสแบบไฮบริด (Hybrid Coding) สังเกตได้ว่าการชดเชยการเคลื่อนที่นั้นจะมีอยู่ทั้งในตัวเข้ารหัสและตัวถอดรหัส แต่การประมาณค่าเคลื่อนที่นั้นจะอยู่ในตัวเข้ารหัสเท่านั้น ในทางปฏิบัติ ตัวเข้ารหัส (Coder) จะต้องรวมบล็อกของการทำงานของหน่วยความจำของบัฟเฟอร์ (Buffer memory) กระบวนการที่เพิ่มเติมก่อนกระบวนการจริง (additional preprocessing) และกระบวนการหลังจากกระบวนการจริง (Post processing) เอาไว้ด้วย ซึ่งไม่ได้แสดงไว้ในรูป

เมื่อเรารวมการทำงาน ของ Interframe coding และ Intraframe coding เข้าด้วยกันแล้ว มันจะได้อัตราการบีบอัดที่ดีกว่าทำงานเพียงลำพังกระบวนการเดียว



รูปที่ 2.16 การทำงานโดยทั่วไปของ Video coder และ Video decoder

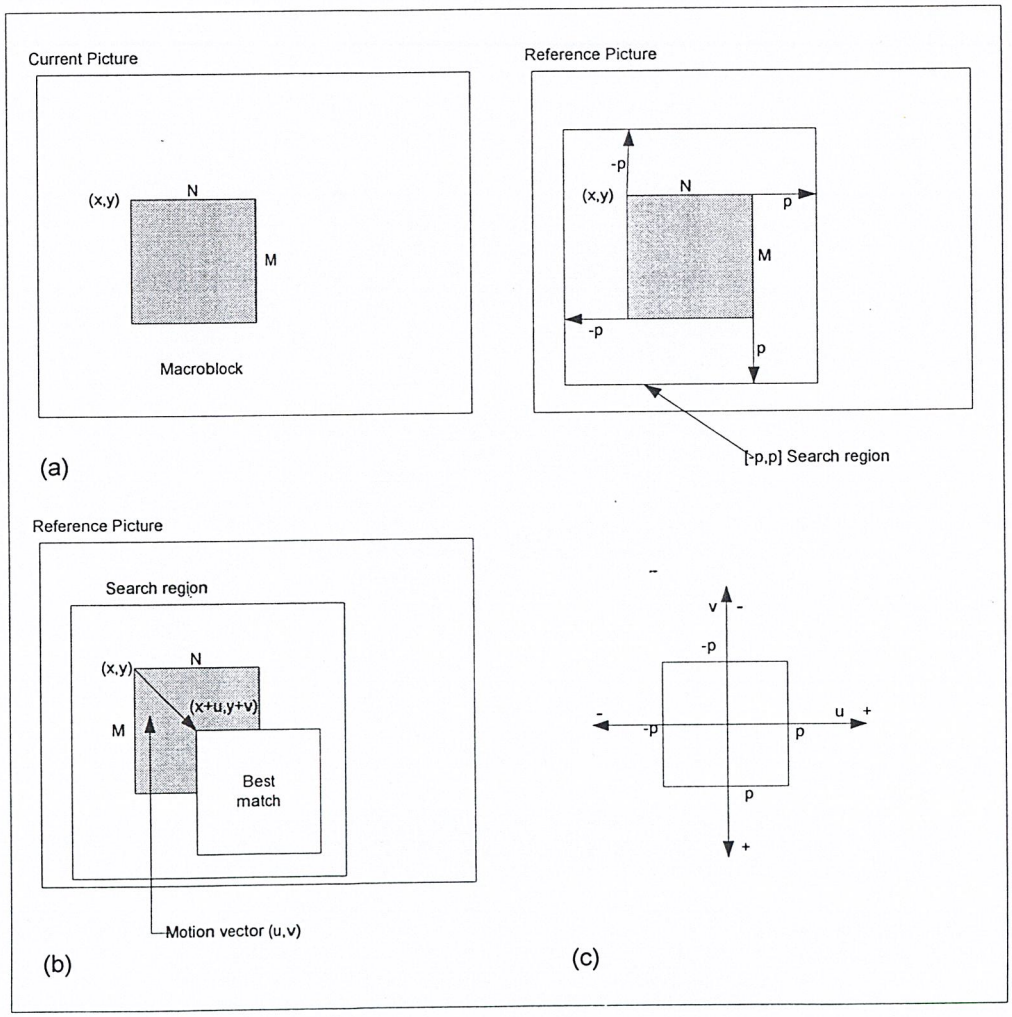
2.12 การทำนายสำหรับการชดเชยการเคลื่อนที่ (Motion Compensated Prediction)

จากที่กล่าวมาแล้ววิธีการบีบข้อมูลวีดีโอทีมีประสิทธิภาพที่เรียกว่า ไฮบริด (Interframe / Intraframe) สามารถแสดงขบวนการได้ดังรูปที่ 2.16 หนึ่งในส่วนสำคัญของขบวนการนี้คือ Motion - estimation ซึ่งเราจะกล่าวถึงต่อไปนี้

จากรูปที่ 2.17 แสดงให้เห็นถึงปัญหาของกระบวนการประมาณค่าการเคลื่อนที่ในมาตรฐานการเข้ารหัสภาพเคลื่อนไหว จากรูปเราให้ภาพอ้างอิงมา 1 ภาพ รวมทั้งมาโครบล็อกขนาด N×M ของภาพปัจจุบันมาด้วย การประมาณการเคลื่อนที่จะเป็นตัวกำหนดบล็อกขนาด N×M ที่อยู่ในภาพอ้างอิงซึ่งสอดคล้องกัน (Match) กับลักษณะของมาโครบล็อกของภาพปัจจุบันมากที่สุด โดยภาพปัจจุบันเราให้เป็นภาพนิ่งหรือเฟรมที่เวลา t ส่วนภาพอ้างอิงเราจะกำหนดให้เป็นภาพนิ่งหรือ เฟรม

ทั้งเวลาที่เป็นอดีตคือ t-n สำหรับการประมาณค่าการเคลื่อนที่แบบไปข้างหน้า (Forward motion estimation) และเวลาที่เป็นอนาคต t-k สำหรับการประมาณค่าการเคลื่อนที่แบบย้อนกลับ (Backward motion estimation) ในกรณีนอกเหนือจากนี้ เรขาคณิตของบล็อกที่สอดคล้องกัน (matching block) ในภาพอ้างอิงไม่จำเป็นต้องมีรูปทรงเหมือนบล็อกที่อยู่ปัจจุบัน แต่อย่างไรก็ตาม ในมาตรฐานการเข้ารหัสภาพเคลื่อนไหว เพียงแค่ใช้รูปทรงสี่เหลี่ยมเป็นเรขาคณิตของบล็อกที่สอดคล้องกันก็เพียงพอแล้ว

ตำแหน่งของ macroblock มักจะถูกกำหนดให้เป็น (x, y) ซึ่งเป็นตำแหน่งที่มุมซ้ายด้านบนของพื้นที่ ในทางอุดมคติเราต้องการที่จะหา (search) ภาพอ้างอิงทั้งภาพที่ match มากที่สุด แต่ว่ามั่นไม่สามารถนำไปใช้ในทางปฏิบัติได้ ดังนั้นเราจะทำการจำกัดการค้นหา ที่พื้นที่ [-p, p] รอบ ๆ จุดกำเนิดของ macroblock



รูปที่ 2.17 Motion – estimation process

กำหนดให้ (x+u, y+v) เป็นตำแหน่งของบล็อกที่สอดคล้องมากที่สุดของภาพอ้างอิง ดังแสดงในรูปที่ 2.17 (b) เวคเตอร์จากตำแหน่ง (x, y) ถึงตำแหน่ง (x+u, y+v) หมายถึงเวคเตอร์ของการเคลื่อนที่ (Motion vector) ที่บอกตำแหน่งของมาโครบล็อกซึ่งก็คือ ตำแหน่ง (x, y) นั้นเอง เรามักสมมุติให้ตำแหน่ง (x, y) มีค่า (0, 0) ดังนั้นเวคเตอร์ของการเคลื่อนที่จึงถูกกำหนดให้เป็น (u, v)

ในมาตรฐานการเข้ารหัสภาพเคลื่อนไหว N=M=16 จากรูปที่ 2.17(a) สำหรับพื้นที่ของการค้นหาจะเห็นว่า $-p \le u \le p$ และ $-p \le v \le p$

2.12.1 บรรทัดฐานการเปรียบเทียบ (Matching Criterion)

ถ้าให้ พิกเซลในเฟรมปัจจุบันถูกแสดงด้วย C(x+k , y+1) และพิกเซลในภาพอ้างอิงแสดงด้วย R(x+I +k , y+j+1) เราสามารถเขียนสมการ MAE (Absolute Mean Error) ได้ว่า

$$MAE(i,j) = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}\left|C(x+k,y+1) - R(x+i+k,y+j+l)\right|$$

จากบรรทัดฐาน MAE เรากำหนดไว้ว่าตำแหน่ง Matching block เปรียบเทียบที่ดีที่สุด คือ จุดที่ MAE (i, j) นั้นจะเป็นตำแหน่งของ Motion vector ดังนั้นจะนำเราไปสู่ อัลกอริธึมในการ Search เวคเตอร์ (u, v) ที่มีเป้าหมายตามบรรทัดฐานนี้

2.13 อัลกอริธึมสำหรับ Motion estimation

มีวิธีหลายวิธีที่ใช้ในการทำ Motion estimation แต่ในมาตรฐาน Video coding ไม่ได้มีการกำหนดให้ใช้วิธีใด ดังนั้นเราสามารถนำแต่ละวิธีไปประยุกต์ใช้ได้ตามต้องการได้ โดยทั่วไป Motion estimation ชนิดต่าง ๆ สามารถสรุปได้ดังนี้ คือ

1. วิธีการค้นหาแบบเต็มรูปแบบ (Full-Search Method)
2. วิธีการค้นหาแบบล็อกการิทึมสองมิติ (Two-Dimensional Logarithmic search)
3. วิธีการค้นหาแบบลำดับชั้นคู่ขนานหนึ่งมิติ (Parallel Hierarchical One-Dimensional Search (PHODS))
4. วิธีการประมาณค่าการเคลื่อนที่แบบลำดับชั้น (Hierarchical Motion Estimation)

2.13.1 วิธีการค้นหาแบบเต็มรูปแบบ (Full-Search Method)

วิธีที่ง่ายที่สุดในการค้นหาเวคเตอร์ของการเคลื่อนที่สำหรับแต่ละมาโคร-บล็อกก็คือ การคำนวณหา MAE (i, j) ของแต่ละตำแหน่งในพื้นที่ของการค้นหา (search space) ซึ่งเรียกว่าอัลกอริทึมของการค้นหาแบบเต็มรูปแบบ แต่วิธีนี้มีการคำนวณที่ยุ่งยากมาก แต่ก็ได้รับการรับรอง

ว่าจะสามารถหาค่า MAE ที่น้อยที่สุดได้ ดังนั้นเราจึงหาวิธีปรับปรุงและพัฒนาวิธีการใหม่ ๆ ขึ้นมา เพื่อลดความยุ่งยากในการคำนวณดังกล่าว แต่ก็ยังไม่สามารถรับรองได้ว่าจะได้ค่า MAE ที่น้อยทีสุด หรือไม่

2.13.2 วิธีการค้นหาแบบลอการิทึมสองมิติ (Two- Dimensional Logarithmic Search)

วิธีการค้นหาแบบนี้คล้ายกับการค้นหาแบบไบนารี่มาก ขั้นแรกเราจะแบ่งพื้นที่ที่ต้องการค้นหา [-p, p] ออกเป็น 2 ส่วน ส่วนแรกจะอยู่ภายในพื้นที่รูปสี่เหลี่ยม $[\frac{-p}{2}, \frac{p}{2}]$ (ที่ความละเอียดเป็นจำนวนเต็มของจุดสี) ส่วนที่ 2 จะอยู่ภายนอกพื้นที่ดังกล่าว นอกจากนั้นแทนที่เราจะค้นหาภายในพื้นที่ $[\frac{-p}{2}, \frac{p}{2}]$ ทั้งหมด เราจะคำนวณหาการทำงาน (function) ของ MAE สำหรับ 9 ตำแหน่งเท่านั้นกล่าวคือ ที่ตำแหน่ง (0, 0) และอีก 8 จุดที่สำคัญภายในพื้นที่ $[\frac{-p}{2}, \frac{p}{2}]$ ซึ่งถ้าระยะทางระหว่างจุดเหล่านี้คือ d_1 เราจะสามารถหา MAE ที่มีน้อยที่สุดได้จากการคำนวณ MAE ที่จุด (0, 0), (0, d_1), (0, $-d_1$), ($-d_1$, 0), (d_1, 0), (d_1, d_1), (d_1, $-d_1$), ($-d_1$, d_1) และ ($-d_1$, $-d_1$) โดยระยะ d_1 จะถูกกำหนดเป็น

$$d_1 = 2^{k-1}$$

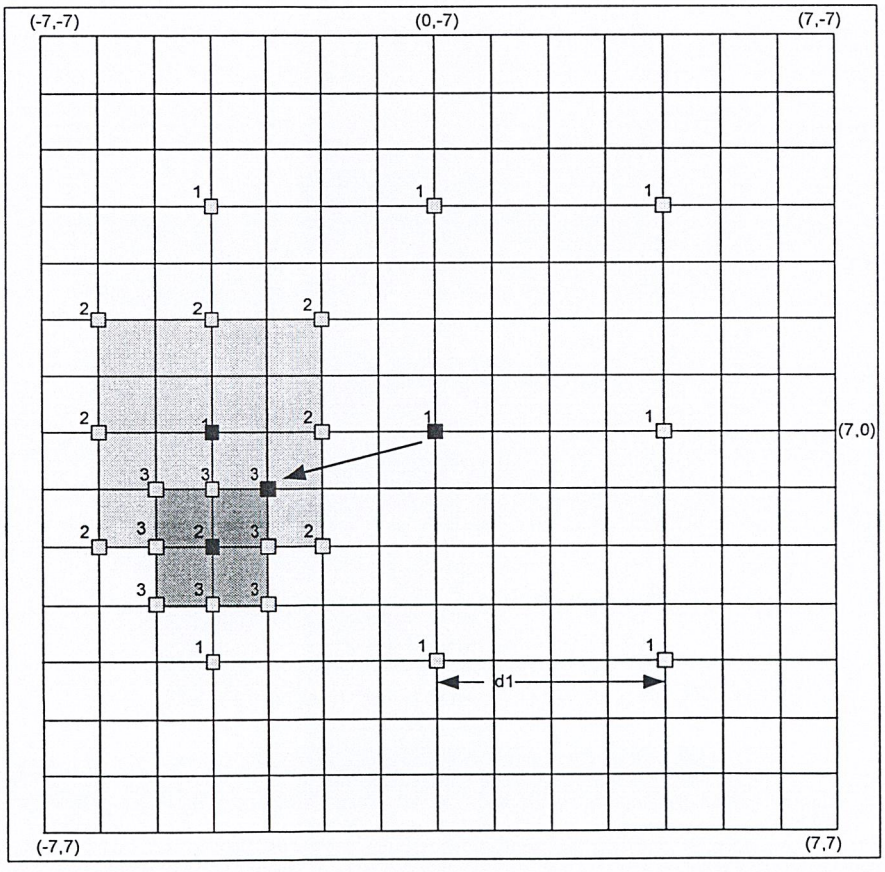
โดยที่ $k = [\log_2 p]$

ตัวอย่างเช่น สำหรับค่า p = 7, k = 3 และ $d_1 = 4$ เราจะใช้ตำแหน่งที่สอดคล้องกันมากที่สุดเป็นจุดกึ่งกลาง แล้วเราจะทำการค้นหาพื้นที่ที่สอดคล้องกันมากที่สุดต่อไปภายใน 8 จุดที่ถูกกำหนดโดยระยะ $d_2 = \frac{d_1}{2}$ เราจะทำการคำนวณและค้นหาไปจนกว่าจะครบการค้นหาครั้งที่ k ซึ่ง 8 จุดภายในพื้นที่ที่ค้นหานั้นจะถูกกำหนดให้ระยะเป็น 1 หลังจากที่ 8 จุดนั้นได้ถูกกำหนดแล้ว เราก็จะได้ค่า MAE ที่น้อยที่สุดนั่นเอง

จากรูปที่ 2.18 แสดงให้เห็นกระบวนการการค้นหาแบบลอการิทึมสองมิติที่นิยมใช้กันมาก ซึ่งเรียกว่า การค้นกาแบบ 3 ขั้นตอน (Three – step search หรือ TSS) ซึ่งเราให้ k = 3 และ p = 7 ในการประยุกต์ใช้งานสำหรับข้อมูลภาพเคลื่อนไหวนั้นค่า p = 7 ก็เพียงพอแล้ว

การค้นหาที่แบ่งออกเป็น 3 ขั้นตอนดังกล่าว แสดงให้เห็นโดยตัวเลขทีกำกับเป็น 1,2 และ 3 ในรูปที่ 2.18 นั่นเอง ซึ่งรายละเอียดมีดังต่อไปนี้

- ขั้นแรก เราจะเริ่มที่ตำแหน่ง (0, 0) เป็นจุดกึ่งกลาง จากนั้นเราจะคำนวณหา 8 จุดที่เหลือ ซึ่งแสดงได้ดังรูป ตรงตำแหน่งที่มีเลขกำกับว่า 1 ระยะระหว่างจุดเหล่านั้น คือ d_1 และสามารถสันนิษฐานได้ว่า ค่า MAE ที่น้อยที่สุดภายในพื้นที่ที่ค้นหาคือ (-4, 0)
- ขั้นที่สอง เราจะใช้ตำแหน่ง (-4, 0) เป็นจุดกึ่งกลางในการค้นหา 8 จุดที่เหลือนั้นต่อไป โดยมีเลขกำกับเป็นตัวเลข 2 ดังรูป ซึ่งกำหนดให้ระยะระหว่างจุดเหล่านั้นคือ $d_2 = d_1/2$ และสามารถสันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดอยู่ตำแหน่ง (-4, 2)
- ขั้นที่สาม เราจะใช้ตำแหน่ง (-4, 2) เป็นจุดกึ่งกลางในการค้นหา 8 จุดที่เหลือเช่นกัน โดยระยะระหว่างจุดเหล่านั้นคือ d3 = 1 และเราสามารถสันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดอยู่ที่ตำแหน่ง (-3, 1) กระบวนการค้นหาจะสิ้นสุดที่จุด ๆ นี้ เพราะการแบ่งพื้นที่ของการค้นหาไม่สามารถกระทำได้อีกต่อไปแล้ว และเราจะได้เวคเตอร์ของการเคลื่อนทีที่ถูกกำหนดเป็นคู่ลำดับ (-3, 1)



รูปที่ 2.18 ตัวอย่างของวิธีการค้นหาแบบ 3 ขั้นตอน (three – step search หรือ TSS)

2.13.3 วิธีการค้นหาแบบลำดับชั้นคู่ขนานหนึ่งมิติ (Parallel Hierarchical One – Dimensional Search หรือ PHODS)

สิ่งที่แตกต่างจากวิธีการค้นหาแบบลอการิทึมก็คือ การค้นหาแบบนี้สามารทำได้โดยอิสระภายใน 2 มิติ โดยมีอัลกอริทึม ดังนี้

1. สำหรับพื้นที่ค้นหา [-p, p] จากรูปที่ 2.17 ให้ $s = 2^{[\log_2 p]}$ และให้จุดกำเนิดของพื้นที่ที่ต้องการค้นหาคือ (0, 0) สิ่งที่ควรตระหนักถึงคือ จุดกำเนิดดังกล่าวนั้นเสมือนจุด (di, dj)

2. ในขณะเดียวกันเราจะคำนวณ

- แกน i ที่น้อยที่สุด กล่าวคือ ระหว่างจุด 3 จุด ต่อไปนี้ (di, dj - S), (di, dj), (di, dj + S) เราจะหาตำแหน่งที่ให้ค่า MAE ที่น้อยที่สุด โดยเราจะทำให้ dj อยู่ในคู่ลำดับของ j ของตำแหน่งนี้ และให้ S = S / 2

ทำซ้ำขั้นตอนที่สอง จนกระทั่งค่า S = 0 ค่า (di, dj) ที่ได้จะเป็นเวคเตอร์ของการเคลื่อนที่ที่ทำให้ได้ค่ามาโครบล็อกที่สอดคล้องที่สุดในภาพปัจจุบัน

กระบวนการที่แสดงในรูปที่ 2.19 เราจะให้ค่า p = 7 ซึ่งเราอาจจะต้องการเปรียบเทียบกับกระบวนการการค้นหาแบบที่แล้วในรูปที่ 2.18 ได้เช่นกัน

เริ่มต้นที่ค่า S = 4

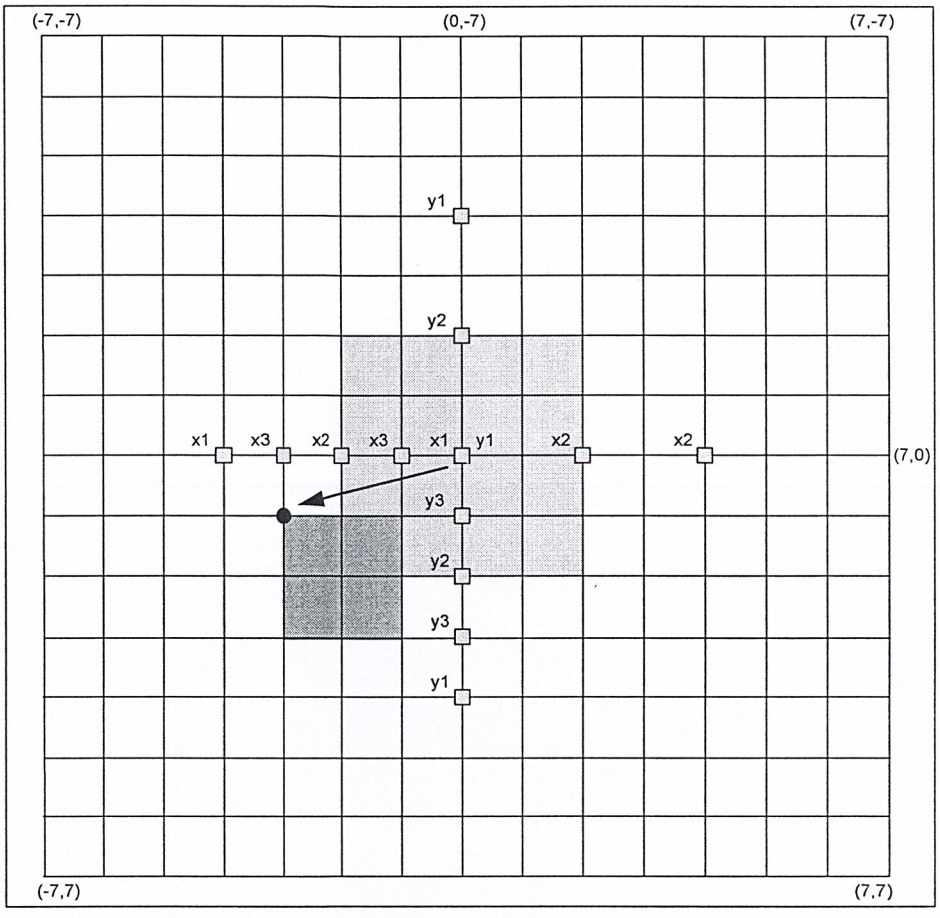
1. ขั้นแรกเราต้องหาแกน i น้อยที่สุดก่อน โดยการคำนวณค่า MAE ของ 3 จุดดังแสดงได้ด้วยสัญลักษณ์ x_1 ในรูปที่ 2.19 สันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดถูกกำหนดด้วยค่า i = 0 ในขณะเดียวกันเราก็ต้องคำนวณหาแกน j ที่น้อยที่สุดเช่นกัน ซึ่งในรูปที่ 2.19 เราจะแสดงด้วยสัญลักษณ์ y_1 และสันนิษฐานได้ว่าค่า MAE ทีน้อยที่สุดถูกกำหนดด้วยค่า j = 0 ดังนั้นจุดกำเนิดใหม่จะอยู่ที่ (0, 0)

2. ระยะห่างระหว่างจุดถูกลดลงให้เหลือ 2 ค่าแกน i ที่น้อยที่สุดจะถูกกำหนดจากจุดกึ่งกลางใหม่ที่หาได้จากขั้นตอนที่แล้ว โดยจะแสดงได้ด้วยสัญลักษณ์ x_2 ดังรูปที่ 2.19 สันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดอยู่ที่ค่า i = -2 และสำหรับการหาค่าแกน j ที่น้อยที่สุดนั้น เราจะแสดงด้วยสัญลักษณ์ y_2 และสันนิษฐานได้ว่า ค่า MAE ที่น้อยที่สุดอยู่ที่ค่า j = 2 ดังนั้นจุดกำเนิดของการค้นหาใหม่อยู่ที่ตำแหน่ง (-2, 2)

3. สำหรับขั้นตอนนี้ ค่า S – 1 และค่า MAE ที่น้อยที่สุดตามแนวแกน i จะแสดงด้วยสัญลักษณ์ x_3 และในทำนองเดียวกัน ค่า MAE ที่น้อยทีสุดตามแนวแกน j ก็จะถูกแสดงด้วยสัญลักษณ์ y_3 เดียวกัน ดังนั้นเราสามารถสันนิษฐานได้ว่า ค่า MAE ที่น้อยที่สุดอยู่ที่ค่า (-3, 1)

ด้วยเหตุผลที่ว่าค่า S = 1 คือค่าเริ่มต้นของขั้นตอน การที่จะลดหรือแบ่งพื้นที่จึงไม่สามารถกระทำได้ เราจึงสรุปได้ว่า ค่า MAE ที่น้อยที่สุดอยู่ที่ตำแหน่ง (-3, 1) และดังนั้นเวคเตอร์ของการ

เคลื่อนที่ที่ต้องการสำหรับมาโครบล็อกที่สอดคล้องที่สุดในภาพปัจจุบัน ก็คือ ค่าของตำแหน่งนี้นั่นเอง



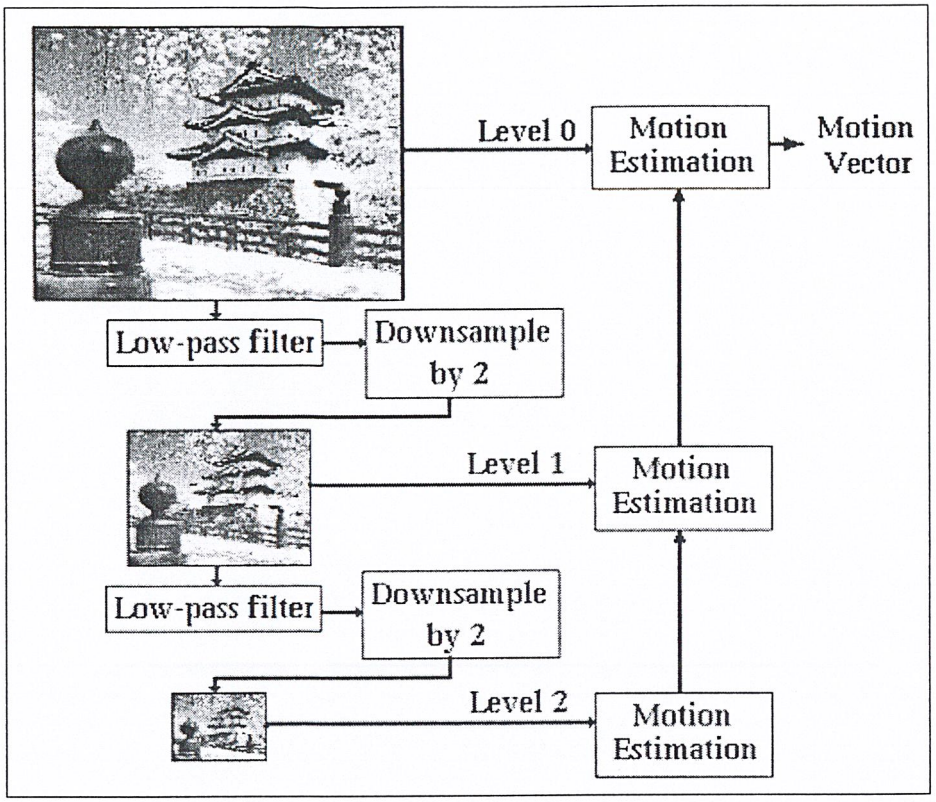
รูปที่ 2.19 ตัวอย่างของวีการค้นหาแบบ PHODS

PHODS มีข้อดีกว่า TSS อยู่ 1 ประการคือ

1. มีการทำงานที่เป็นระเบียบ เพราะการค้นหาตำแหน่งที่ต้องการนั้นจะกระทำไปตามแนวแกน i และ j เสมอ

2.13.4 วิธีการประมาณค่าการเคลื่อนที่แบบลำดับชั้น (Hierarchical Motion Estimation)

วิธีการประมาณค่าการเคลื่อนที่หลายวิธีได้นำการค้นหาตำแหน่งที่น้อยลงมารวมกันกับการคำนวณหาพิกเซลที่น้อยลง โดยใช้การคำนวณหา MAE ซึ่งเราจะเรียกวิธีการนี้ว่า วิธีค้นหาแบบลำดับชั้น วิธีการนี้มีกระบวนการที่ทำได้หลายแบบ



รูปที่ 2.20 วิธีการทั่วไปของการค้นหาเวคเตอร์ของการเคลื่อนที่ในวิธีการค้นหาแบบลำดับชั้น

กระบวนการค้นหาเวคเตอร์ของการเคลื่อนที่ มีขั้นตอนดังต่อไปนี้

1. เราจะกำหนดให้ภาพปัจจุบันและภาพอ้างอิงมีค่าความละเอียดต่ำหลายค่า ดังรูปที่ 2.20 แสดงให้เห็นถึงภาพอ้างถึงที่มีค่าความละเอียดต่ำ 2 ค่า คือที่ ระดับที่ 1 (Level 1) และระดับที่ 2 (Level 2) ให้เราสันนิษฐานว่า มาโครบล็อกในภาพปัจจุบันมีตำแหน่งอยู่ที่ (x,y) ดังนั้นในระดับที่ 1 และ ระดับที่ 2 มาโครบล็อกที่มีความคล้ายคลึงกัน จะอยู่ที่ตำแหน่ง $\left(\frac{x}{2},\frac{y}{2}\right)$ และ $\left(\frac{x}{4},\frac{y}{4}\right)$ ตามลำดับ และให้เราสันนิษฐานว่าในระดับที่ 0 (level 0) มาโครบล็อกจะมีขนาด 16×16 และมีเวคเตอร์ของการเคลื่อนที่อยู่ในช่วงพิกเซลที่ $\pm p$

2. เราจะเริ่มต้นหาเวคเตอร์ของการเคลื่อนที่ที่ระดับที่ 2 เป็นอันดับแรก เราจะใช้พารามิเตอร์ของการค้นหาเป็น $\frac{p}{4}$ แทนที่จะใช้ p ดังนั้นวิธีการหาเวคเตอร์ของการเคลื่อนที่ในระดับที่ 2 จึงสามารถใช้ได้ทั้งวิธีการค้นหาแบบเต็มรูปแบบและการค้นหาแบบลอการิทึม ซึ่งทั้ง 2 วิธีการนี้จะใช้มาโครบล็อกขนาด 4×4 ใช้พารามิเตอร์ในการค้นหาคือ $\frac{p}{4}$ และจุดเริ่มต้นของการ

ค้นหาอยู่ที่ตำแหน่ง $\left(\frac{x}{4},\frac{y}{4}\right)$ ทำให้เราสามารถสันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดอยู่ที่ตำแหน่ง (u_2,v_2)

3. ที่ระดับที่ 1 เราจะสามารถค้นหาเวคเตอร์ของการเคลื่อนที่ได้โดยใช้มาโครบล็อกขนาด 8×8 ซึ่งจุดเริ่มต้นการค้นหาอยู่ที่ $\left(\frac{x}{2}+2u_2,\frac{y}{2}+2v_2\right)$ และพื้นที่ของการค้นหาอยู่ที่ตำแหน่ง $[-1,1]$ รอบ ๆ จุดเริ่มต้น และสามารถสันนิษฐานได้ว่าค่า MAE ที่น้อยที่สุดอยู่ตำแหน่ง (u_1,u_1)

4. ที่ระดับที่ 0 เรากำหนดให้จุดเริ่มต้นของการค้นหาอยู่ที่ $(x+2u_1\ ,y+2v_1)$ และพื้นที่ของการค้นหาอยู่ที่ตำแหน่ง $[-1,1]$ รอบ ๆ จุดเริ่มต้นซึ่งเราจะใช้มาโครบล็อกขนาด 16×16 โดยใช้วิธีการค้นหาแบบใดก็ได้ ซึ่งก็รวมทั้งวิธีการค้นหาแบบเต็มรูปแบบด้วย ค่า MAE ที่น้อยที่สุดที่ได้นั้นจะเป็นค่าเวกเตอร์ของการเคลื่อนที่สุดท้ายที่เอาท์พุท

การประมาณค่าการเคลื่อนที่ของวิธีการแบบลำดับชั้นนี้ช่วยลดความยุ่งยากเมื่อเปรียบเทียบกับวิธีการค้นหาแบบเต็มรูปแบบลง 29.89 GOPS ซึ่งแสดงให้เห็นว่าวิธีการนี้มีประสิทธิภาพมากแต่อย่างไรก็ตามวิธีการนี้ยังต้องการการเก็บข้อมูลที่ดีพอ ยิ่งไปกว่านั้นเราอาจจะได้เวกเตอร์ของการเคลื่อนที่ที่ไม่เที่ยงตรงนักโดยเฉพาะในบริการที่เป็นวัตถุขนาดเล็ก เพราะวิธีการนี้เราต้องเริ่มจากภาพที่มีความละเอียดต่ำก่อน ทำให้บริเวณที่เป็นวัตถุขนาดเล็กอาจถูกละเลยไป แต่ในทางกลับกันการที่เรากำหนดให้ภาพเริ่มแรกมีค่าความละเอียดต่ำจะเป็นผลทำให้ไม่เกิดสัญญาณรบกวนด้วยเช่นกัน

Search Method	Operation per Macroblock	Operation for 720 x 480 picture at 30 fps	
		p = 15	p = 7
Full search	$(2p+1)^2 NM3$	29.89 GOPS	6.99 GOPS
Logarithmic	$(8\lfloor\log_2 p\rfloor+1)NM3$	1.02 GOPS	777.60 MOPS
PHODS	$(4\lfloor\log_2 p\rfloor+1)NM3$	528.76 MOPS	404.35 MOPS
Hierarchical	$\left[\left(2\left\lfloor\frac{p}{4}\right\rfloor+1\right)^2+180\right]NM\left(\frac{3}{16}\right)$	507.38 MOPS	398.52 MOPS

ตารางที่ 2.1 เปรียบเทียบคุณสมบัติของการ Search วิธีต่าง ๆ

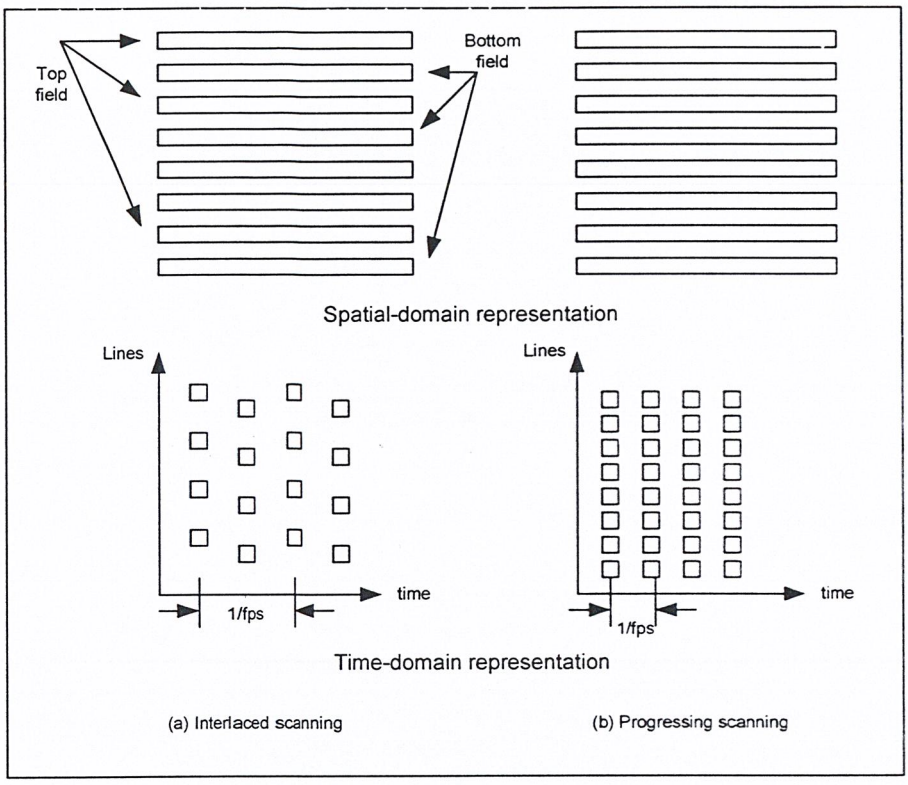
2.14 มาตรฐาน MPEG

ความก้าวหน้าของเทคโนโลยีทางด้าน Digital video และ Digital storage จึงทำให้ระบบ Digital video ถูกใช้งานในมัลติมีเดียได้ ในปี 1988 ได้มีการตื่นตัวที่จะสร้างรูปแบบทั่วไปของการเข้ารหัสและเก็บข้อมูล Digital video มาตรฐานองค์กรนานาชาติ (ISO) ได้ก่อตั้ง MPEG (Moving Picture expert Group) ขึ้นเพื่อพัฒนามาตรฐานการเข้ารหัสข้อมูลภาพเคลื่อนไหวรวมเข้ากับข้อมูลเสียงบนสื่อเก็บข้อมูล ดิจิตอล เดิมนั้น MPEG รู้จักกันในชื่อว่า ISO-IEC / JTC1 SC29 / WG11 ซึ่งการทำงานครั้งแรกได้เสร็จสิ้นเมื่อปี 1991 ด้วยการพัฒนามาจากระบบมาตรฐาน ISO 11172 (Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbits / s) มาตรฐานแรกนี้รู้จักกันในชื่อว่า MPEG 1

ในปี 1990 MPEG ได้เริ่มงานที่สอง นั่นก็คือการพัฒนา MPEG 1 ออกไปอีก โดยให้มีความยืดหยุ่นของรูปแบบข้อมูลมากขึ้น อัตราส่งผ่านข้อมูลที่ดีขึ้นและลดความผิดพลาดในการถอดรหัสลง โดยใช้ชื่อว่า มาตรฐาน ISO 13818 (Generic coding of moving pictures and associated audio) หรือ MPEG 2 และได้เริ่มงานใหม่ที่ทำให้ได้ bit rate ต่ำมากคือ MPEG 4 ซึ่งคาดว่าจะเสร็จในปี 1998 มาตรฐาน MPEG ต้นกำเนิดอิสระกับการใช้งาน โดยมันจะกำหนดในเรื่อง วิธีการเข้ารหัส bit stream และกระบวนการถอดรหัส โดยที่ไม่ได้กำหนดวิธีการเข้ารหัส จึงเป็นวิธีที่ยืดหยุ่นเพียงพอที่ทำให้ผู้ขายแต่ละรายสามารถที่จะพัฒนาแต่ละส่วนของระบบของตนเองให้ดีที่สุด มาตรฐาน MPEG ได้ตีพิมพ์แบ่งออกเป็น 4 ส่วน คือ systems, video, audio และ conformance testing ในบทนี้เราจะพูดถึงเฉพาะส่วน video เท่านั้น

2.14.1 พื้นฐานเบื้องต้นของ MPEG

มาตรฐานนี้เป็นอัลกอลิธึมที่เป็นการบีบอัดข้อมูลแบบสูญเสียที่สามารถประยุกต์ใช้งานได้กับอินพุทฟอร์แมทอย่างมากมาย มันเหมาะสำหรับอุปกรณ์ที่มี bit rate ประมาณ 1.5 Mbits / s อย่างเช่น CD-ROM ในมาตรฐาน MPEG จะใช้คำว่า Picture แทน คำว่า เฟรม โดยมันไม่สามารถรับข้อมูลแบบ Interlace ในระบบ Interlaced video นั้นแต่ละเฟรมจะถูกแบ่งออกเป็น 2 ฟิลด์ คือ top field และ bottom field ของภาพในเฟรมของ Interlaced video ในหนึ่งเส้นสแกนจะต้องมีทั้งสองฟิลด์แทรกกันอยู่ ดังรูปที่ 2.21 ตัวอย่างเช่นถ้าเป็นวีดีโอความเร็ว 30 เฟรมต่อวินาที จะต้องมีการสแกนเส้นทั้งหมด 60 เส้นต่อวินาที ทางด้านในระบบ Noninterlaced video จะต่างกันคือมันจะสแกนต่อเนื่องไปจากมุมซ้ายบนไปถึงด้านล่างของเฟรมโดยไม่มีการแบ่งเป็นฟิลด์



รูปที่ 2.21 แสดงให้เห็นการสแกนสลับกันและสแกนแบบเรียงลำดับ

กุญแจ สำคัญที่ต้องการตลอดกาพัฒนาระบบมาตรฐาน MPEG มีอยู่ 2 ประการคือ

1. High compression
2. Random access capability

ในการเข้ารหัส Intraframe เพียงลำพังนั้นจะสามารถบรรลุได้เพียง Random access capability เท่านั้น เพื่อให้บรรลุเป้าหมายสองประการนี้จะต้องใช้เทคนิคของ Intraframe และ Interframe ร่วมกันนอกจากนั้นการที่จะปรับปรุงอัตราการบีบข้อมูลจะต้องใช้ Predictive และ Interporative ด้วย

การทำงานการบีบข้อมูลใน MPEG จะต้องร่วมด้วย

1. การลดอัตราการสุ่มใน Spatial domain และ temporal domain ทั้งองค์ประกอบแสงและสี
2. Block-based DCT สำหรับ อินทราเฟรม และอินเทอร์เฟรม
3. Block-based motion compensation สำหรับ predictive interframe และ interpolative interframe
4. การเข้ารหัสฮัฟแมน สำหรับ motion vector และ quantized DCT coefficients

รูปแบบของแหล่งกำเนิดของข้อมูลเข้า (Source input format)

ใน MPEG กำหนดให้ขนาดภาพมีได้สูงสุดที่ 4095 × 4095 พิกเซล อย่างไรก็ตามการประยุกต์ใช้งานจำนวนมากที่การบีบข้อมูลวีดีโอที่ใช้มาตรฐาน MPEG ถูกปรับไว้สำหรับ SIF (source input format) ซึ่งได้มาง่าย ๆ จากรูปแบบ CCIR 601 ในภาพเคลื่อนไหวในโทรทัศน์ระบบดิจิตอล จากรูปแบบของ CCIR601 แหล่งกำเนิดของภาพเคลื่อนไหวที่เป็นสี คือประกอบไปด้วยสามส่วนคือ 1 องค์ประกอบแสง (Y) และ 2 องค์ประกอบผลต่างของสี (Cb, Cr)

มี2ทางเลือกสำหรับค่าความละเอียดที่ต่างกันของระบบ CCIR 601 คือ

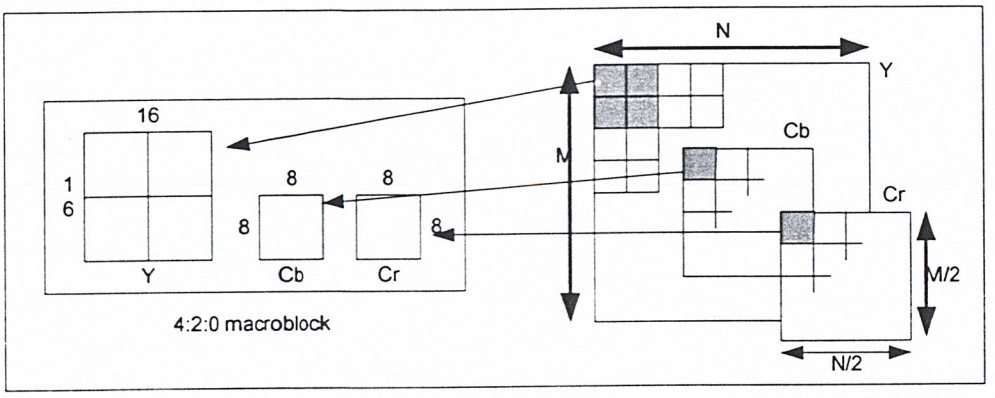
- สำหรับโทรทัศน์ระบบ NTSC เป็นระบบที่ใช้การสแกน 625 เส้นต่อเฟรม และมี 60 เฟรมต่อวินาที เฟรมขององค์ประกอบแสง 720 × 480 พิกเซล และองค์ประกอบสีแต่ละอันเป็น 360 × 480 พิกเซล

- สำหรับโทรทัศน์ระบบ PAL เป็นระบบที่ใช้การสแกน 525 เส้นต่อเฟรม และมี 50 เฟรมต่อวินาที เฟรมขององค์ประกอบแสง 720 × 576 พิกเซล และ องค์ประกอบสีแต่ละอันเป็น 360 × 576 พิกเซล

เพื่อที่จะทำให้ได้อัตราการบีบอัดที่ต่ำถึง 1.5 เมกะบิทต่อวินาที MPEG-1 จำเป็นต้องกำหนดรูปแบบของแหล่งกำเนิดของข้อมูลเข้า (SIF) ซึ่งลำดับของ SIF มีความละเอียดขององค์ประกอบแสงที่ 360×240 พิกเซลต่อภาพ และมีอัตราเร็ว 30 ภาพต่อวินาที หรือที่ 360×288 พิกเซลต่อภาพและมีอัตราเร็ว 25 ภาพต่อวินาที

ทั้ง 2 กรณีนี้ความละเอียดขององค์ประกอบสีจะเป็นครึ่งหนึ่งของความละเอียดขององค์ประกอบแสง ทั้งในแนวนอนและแนวตั้ง หรืออาจกล่าวได้ว่าเป็นรูปแบบของการแบ่งข้อมูลเป็นส่วนย่อย ๆ ที่มีอัตราส่วน 4:2:0 ภาพ SIF สามารถได้มาง่าย ๆ จากเฟรมของ CCIR601 โดยที่ใช้การกรองและการแบ่งข้อมูลเป็นส่วนย่อย ๆ

องค์ประกอบ YCbCr ของ MPEG จะต้องมีการแทรก (interleave) เสมอโดยมาโครบล็อกที่เล็กที่สุดมี 4 มาโครบล็อกสำหรับองค์ประกอบแสง (Y) และมี 1 มาโครบล็อกสำหรับแต่ละองค์ประกอบสี (Cb, Cr) โดยขนาดของมาโครบล็อกเป็น 8 × 8 และมีขนาดสูงสุดเป็น 16 × 16 ดังแสดงในรูปที่ 2.22



รูปที่ 2.22 ข้อกำหนดของ macroblock ในมาตรฐาน MPEG

ภาพแต่ละภาพจะถูกแบ่งออกเป็นอนุกรมของมาโครบล็อกจากซ้ายไปขวาและจากบนลงล่าง ดังนั้นภาพแต่ละภาพจะต้องมีความละเอียดเป็นจำนวนเท่าของ 16 ถ้าหากว่าไม่ตรงตามข้อตกลงนี้ ตัวเข้ารหัสจะทำให้ค่าพิเซลเป็น 0 แล้วนำค่าไปไว้ด้านขวาหรือด้านล่างของภาพ และจะถูกตัดทิ้งโดยตัวถอดรหัสในภายหลัง

Picture Rate (Hz)	30	25
CCIR 601		
Y	720 × 480	720 × 576
Cb , Cr	360 × 480	360 × 576
SIF		
Y	360 × 240	360 × 288
Cb , Cr	180 × 120	180 × 144
Significant Pixel Area for SIF		
Y	352 × 240	352 × 288
Cb , Cr	176 × 120	176×144

ตารางที่ 2.2 ขนาดของภาพ สำหรับระบบ CCIR 601 , SIF และพิกเซลในพื้นที่ใช้งาน

MPEG-1 จะให้ความยืดหยุ่นในการนำไปใช้งานได้เป็นอย่างดี แต่อย่างไรก็ตามก็ไม่สามารถคาดหวังได้ว่าตัวถอดรหัสทั้งหมดของ MPEG จะสามารถสนับสนุนการเข้ารหัสทุกรูปแบบได้ ดังนั้น MPEG-1 จึงกำหนด ข้อจำกัดของพารามิเตอร์บิทสตรีม (constrained parameters bit stream) ขึ้นซึ่งก็คือบิทสตรีมที่ตัวถอดรหัสทั้งหมดของ MPEG จะสามารถสนับสนุนได้ จากตารางที่ 2.3 แสดงให้เห็นถึงพารามิเตอร์ดังกล่าว แต่อย่างไรก็ตามพารามิเตอร์เหล่านี้ก็ยังไม่เป็นที่ต้องการในมาตรฐาน

จากข้อบังคับในตารางที่ 2.3 นั้นจะพบว่าเมื่อบังคับให้ pixel rare เท่ากับ 396 × 25 ดังนั้นขนาดของภาพจะถูกจำกัดอยู่ที่ 352 × 288 พิกเซล เฉพาะพื้นที่สำคัญของระบบ SIF แบบ 25 Hz

Coding Parameter	**Maximum Value**
Horizontal picture size	768 pixels
Vertical picture size	576 lines
Macroblocks	396
Pixel rate	396 × 25 marcroblocks / s
Picture rate	30 picture / s
Range of motion vectors	± 64 pixels (half-pixel resolution)
Size of input buffer	327,680 bits
Bite rate	1,856 kbits / s

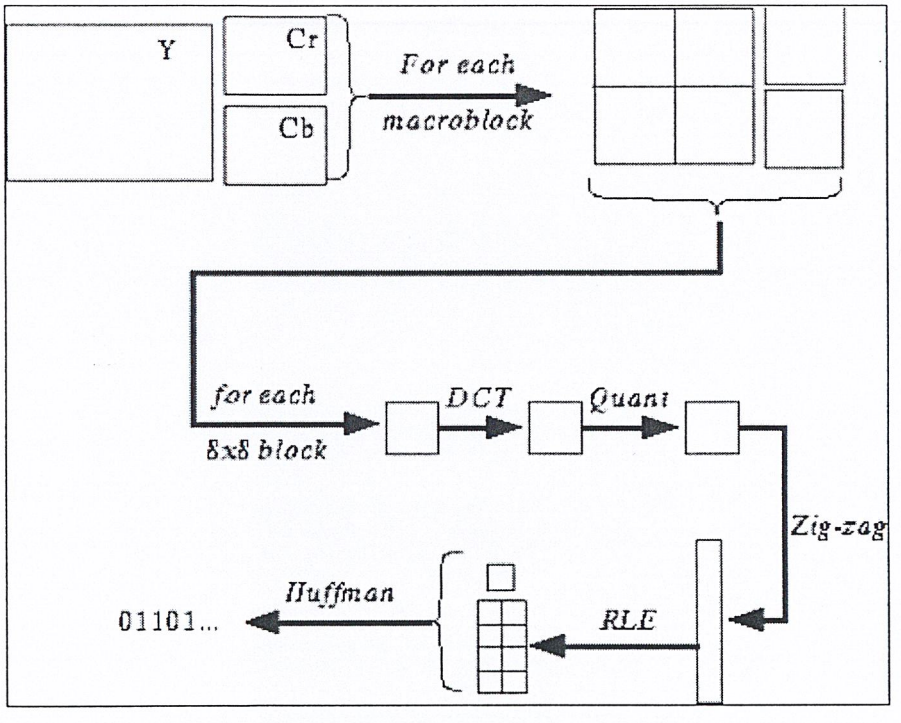
ตารางที่ 2.3 ตัวแปรที่ถูกบังคับในการเข้ารหัส

2.15 ชนิดของภาพ

โดยทั่วไปแล้วการทำงานบางอย่างบน Video stream รวมไปถึง editing, random access และ search การทำงานเหล่านี้มักจะเกิดปัญหาบ่อย ๆ เมื่อต้องบีบข้อมูลในอัตราที่สูง เพื่อให้มีความยืดหยุ่นระหว่างประสิทธิภาพและการเข้าถึงข้อมูลแบบสุ่ม (random access) MPEG ได้แบ่งภาพเป็นชนิดต่าง ๆ กัน 4 แบบ

2.15.1 Intra pictures (I-pictures)

Intra pictures (I-pictures) เป็นภาพที่ถูกบีบอัดโดยการเข้ารหัสแบบภายในเฟรม (intraframe) และไม่ต้องอ้างอิงกับภาพอื่นในบิตสตรีม มันเหมาะสำหรับการเข้าถึงข้อมูลแบบสุ่มอย่างรวดเร็ว (fast random access) แต่จะได้อัตราการบีบอัดในระดับปานกลางเท่านั้น มันจึงเหมาะที่จะประยุกต์ใช้กับภาพ JPEG แบบไม่สูญเสียภาพเดี่ยว ๆ มาก

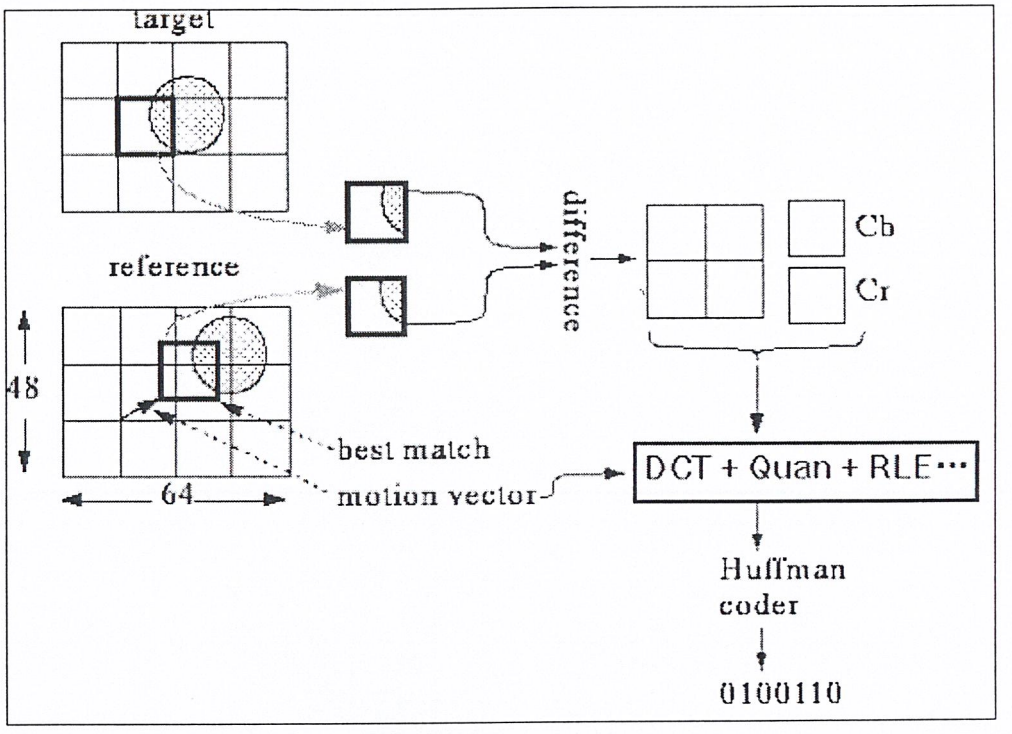


รูปที่ 2.23แสดงการเข้ารหัสของ Intra Frame

2.15.2 Predicted pictures (P-pictures)

Predicted pictures (P-pictures) เป็นภาพที่ถูกเข้ารหัสจากวิธี motion compensated prediction จาก P-picture หรือ I-pictures ที่พึ่งผ่านไป ภาพแบบ P-picture จะมีอัตราการบีบข้อมูลที่

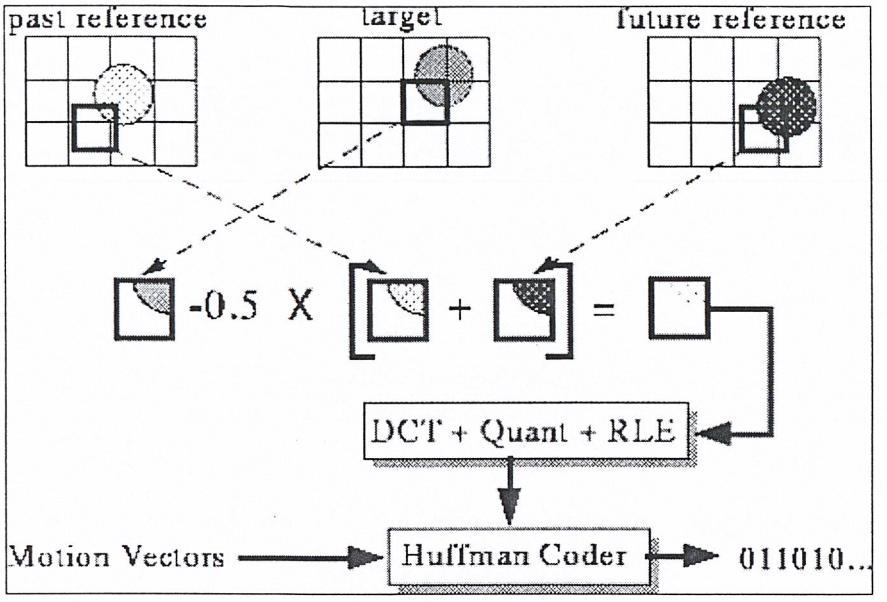
สูงกว่าภาพแบบ I-pictures และมันสามารถที่จะเป็นภาพอ้างอิงในกระบวนการ สร้างภาพ motion compensation และสองทิศทาง (B-pictures)



รูปที่ 2.24 แสดงการเข้ารหัสของ P-Frames

2.15.3 Bidirectionally predicted pictures (B-pictures)

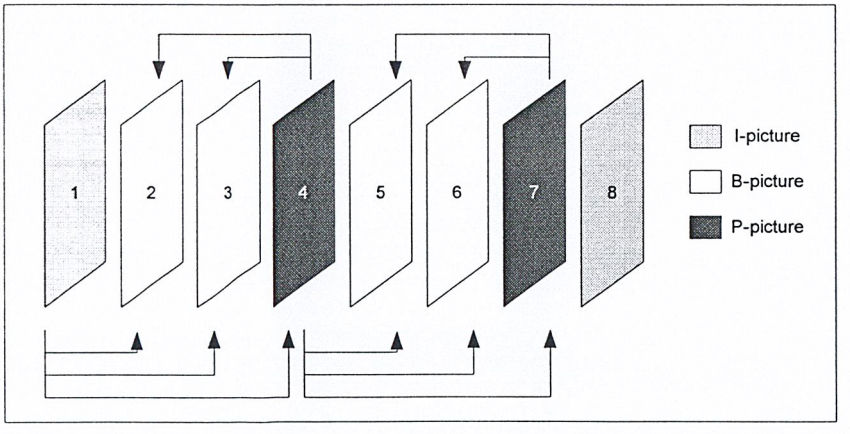
Bidirectionally predicted pictures (B-pictures) เป็นภาพที่ให้อัตราการบีบอัดที่ดีที่สุดและเกิดความบิดเบือนได้ง่าย และเป็นภาพที่ถูกเข้ารหัสโดยวิธี motion compensation prediction จากทั้งภาพ I-pictures(ที่เพิ่งผ่านไป และ/หรือ ที่กำลังจะเกิดขึ้น) หรือ P-picture เพราะ B-picture ไม่สามารถใช้ในการทำนายภาพ B หรือภาพ P ใด ๆ ได้ จึงทำให้มันเป็นภาพที่บิดเบือนได้ง่ายและมีอัตราการบีบอัดที่สูงกว่า I-pictures หรือ P-picture



รูปที่ 2.25 แสดงการเข้ารหัส B-Frame

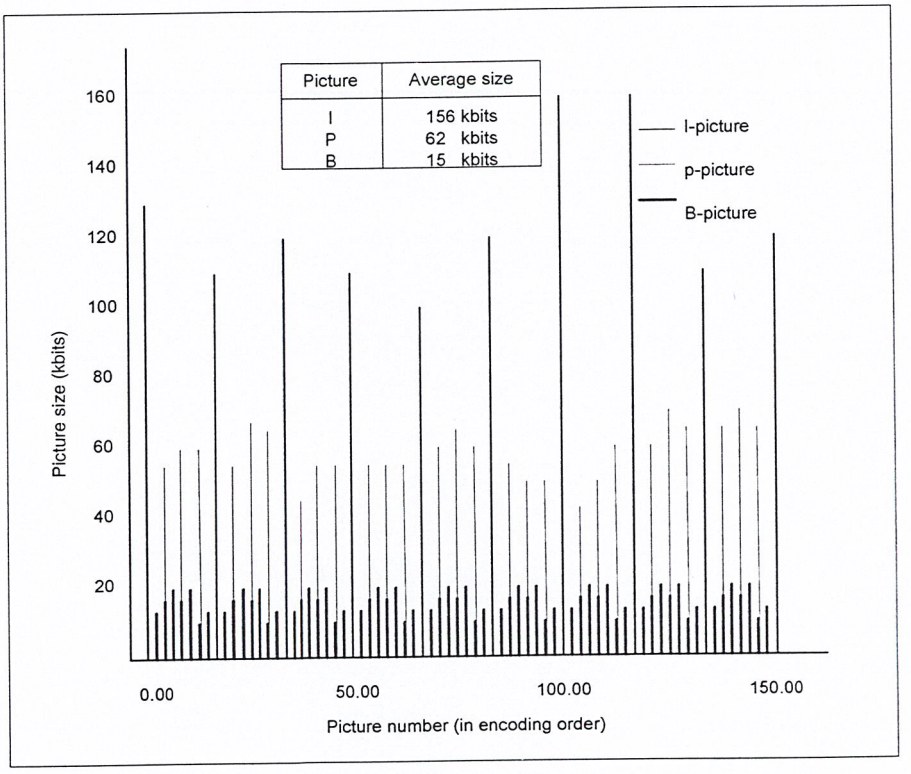
2.15.4 DC-coded picture หรือ D-picture

ภาพที่ได้จากการเข้ารหัส DC หรือภาพ D (DC – coded picture หรือ D-picture) มีลักษณะคล้ายกับภาพ I แต่ต่างกันตรงที่ภาพ D จะแสดงเฉพาะในส่วนของสัมประสิทธิ์ DC เท่านั้น ภาพ D ไม่สามารถนำมารวมกับภาพชนิดอื่น ๆ ได้ อีกทั้งยังไม่สามารถใช้ใน MPEG-2 ได้อีกด้วย แต่สามารถใช้ได้โดยตรงกับการค้นหาแบบรวดเร็ว



รูปที่ 2.26 ความเกี่ยวข้องกันระหว่างภาพแบบ I, P และB ในอนุกรมวีดีโอ

ในรูปที่ 2.26 แสดงถึงความสัมพันธ์ระหว่างภาพหลักสามแบบในอนุกรมวีดีโอ 8 ภาพ P_1 และ P_8 เป็นภาพแบบ I-pictures P_4 และ P_7 เป็นภาพแบบ P-picture และที่เหลือเป็น B-pictures ในตัวอย่างนี้ P_3 จะถูกเข้ารหัสโดยวิธี motion compensation prediction จาก P_1 และ P_4 ที่จริงแล้วใน MPEG เราไม่ จำเป็นที่จะต้องใช้ทั้ง B-pictures และ P-picture พร้อมกัน จากกระบวนการนี้จะพบว่า B-pictures จะมีบิตสตรีมนัยสำคัญต่ำที่สุด แต่การเพิ่ม B-pictures ระหว่าง I-pictures และ P-picture จะไม่ทำให้อัตราการบีบข้อมูลสูงขึ้นเนื่องจากจะมีการตกลงของ temporal correlation ระหว่าง B-pictures และการโยงความสัมพันธ์กับ B-pictures และ I-pictures ก็จะเพิ่มขึ้นด้วยส่วนขนาดของภาพ I, P และ B นั้นโดยปกติในอนุกรมของภาพเคลื่อนไหวจะถูกเข้ารหัสที่ความละเอียด 1.15 เมกะบิทต่อวินาทีนั่นเอง ดังแสดงในรูปที่ 2.27



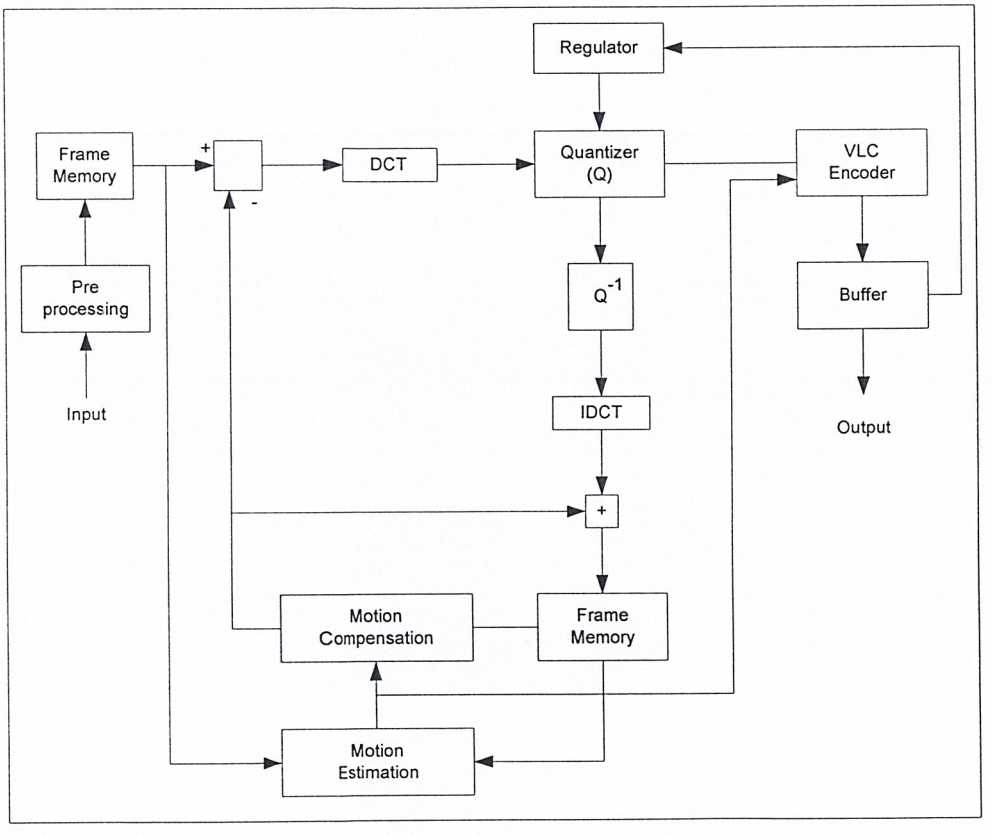
รูปที่ 2.27 การกระจายบิทระหว่างภาพ I, P และ B ในการเข้ารหัสบิทสตรีมของ MPEG-1 โดยมีลำดับของภาพที่เข้ารหัสดังนี้ IPBBPBBPBBPBBPBB

2.16 การเข้ารหัสวีดีโอ

เนื่องจากในมาตรฐาน MPEG ไม่ได้ กำหนดกระบวนการการเข้ารหัสไว้ มีแต่เพียงกำหนดนิยามของกระบวนการทำงานกับบิตสตรีมและการถอดรหัสเท่านั้น แต่ว่าการเข้ารหัสเป็นกระบวนการที่จำเป็น 3.6 จะแสดง การดำเนินการที่สำคัญในการเข้ารหัส MPEG

2.16.1 กระบวนการก่อนกระบวนการจริง (Preprocessing)

การเข้ารหัส (Encoding) มักเริ่มด้วยการกระบวนการเบื้องต้นบางอย่าง ซึ่งอาจจะหมายถึงการแปลงสีเป็นแบบ YcbCr (interlaced to progressive) การกรองเบื้องต้น (prefiltering) และ Subsampling กระบวนการเหล่านี้ไม่ถูกกำหนดไว้ในมาตรฐาน



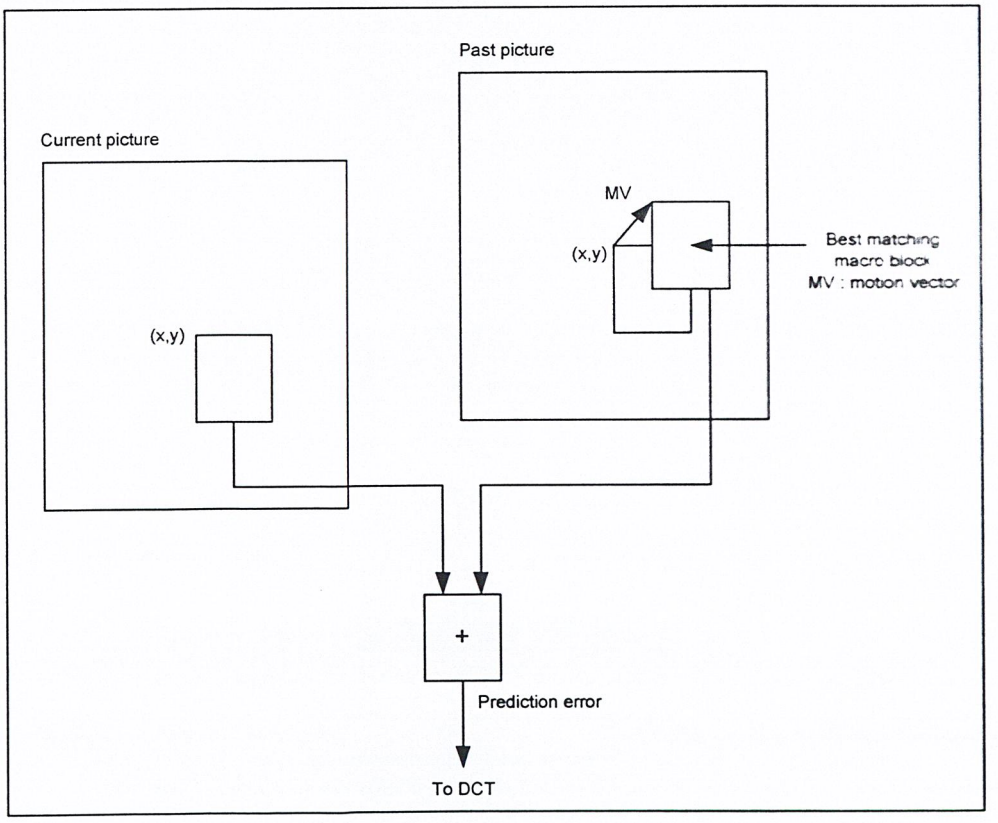
รูปที่ 2.28 Block diagram of an MPEG encoder

2.16.2 การประมาณการเคลื่อนที่และการชดเชยการเคลื่อนที่ (Motion estimation and compensation)

หลังจากการทำ Preprocessing ตัวเข้ารหัสจะทำการเลือกชนิดของภาพขาเข้า ถ้าเป็น I-picture ไม่จำเป็นต้องทำกระบวนการ Motion estimation หรือ compensation แต่ละ macroblock จะ

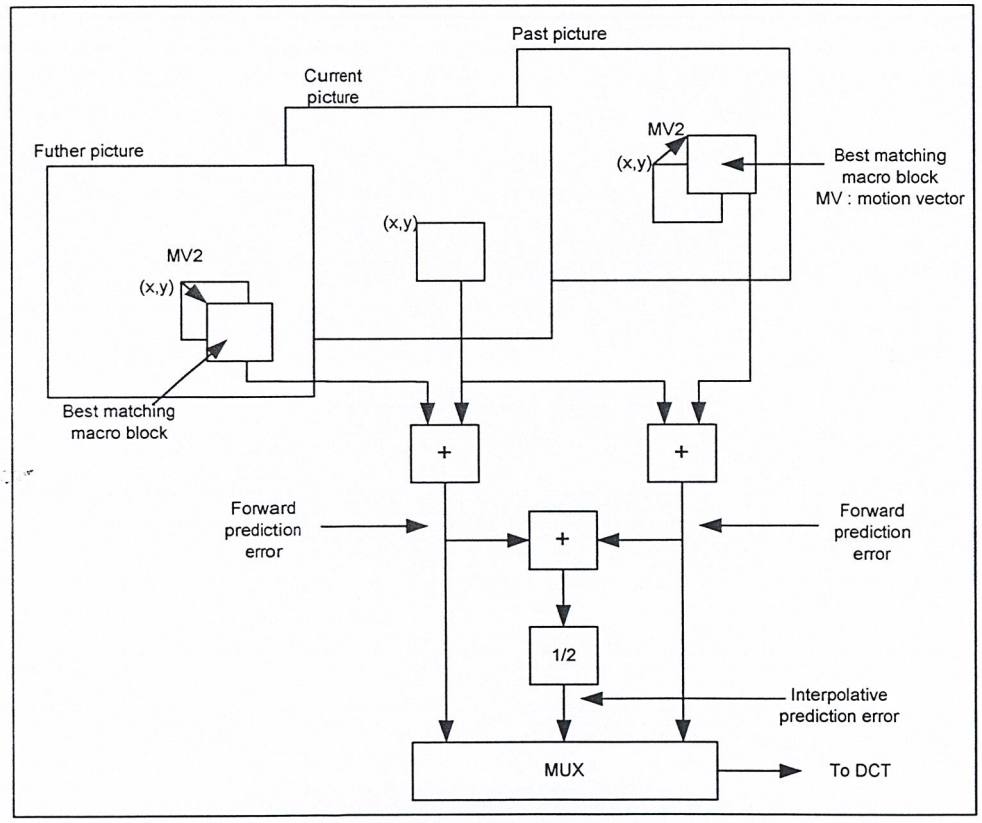
เข้ารหัสแบบ DCT จากนั้นสัมประสิทธิ์ DCT จะถูกควอนไตซ์ และผ่านกระบวนการ VLC (variable-length coder) แล้วนำไปเก็บไว้ในบัฟเฟอร์ทางด้านเอาท์พุท ในแต่ละ macroblock จะต้องทำงานกับข้อมูลขนาด 8 × 8 ในงานที่ต้องการ bit rate คงที่ จะต้องมีตัวปรับ (Regulator) จะต้องปรับเมทริกซ์ควอนไตซ์เพื่อให้บิตเรตค่อนข้างคงที่ สำหรับงานที่ไม่ต้องการบิตเรตคงที่ฟังก์ชันนี้ก็ไม่จำเป็นต้องใช้ สำหรับกระบวนการควอนไตซ์จะเหมือนกับ Inverse quantized (Q^{-1}) แล้วแปลงให้อยู่ใน spatial domain IDCT การทำงานนี้จะเรียนแบบพฤษติกรรมของการถอดรหัสภาพเพื่อให้ได้สำเนาของภาพที่เข้ารหัสและถูกเห็นได้โดยตัวเข้ารหัส สำเนานี้จะถูกเก็บไว้ในหน่วยความจำและถูกใช้สำหรับการทำนายรหัสในอนาคต ผลจากการทำงานนี้ทำให้ตัวถอดรหัสคอยเฝ้าดูคุณภาพของภาพที่ส่งมาไม่ให้บิดเบือนไปจากภาพเดิม เนื่องจากว่า VLC เป็นการเข้ารหัสแบบไม่สูญเสียดังนั้นมันจึงไม่ต้องมีในเส้นทาง feedback

เมื่อเข้าอินพุทถูกทำให้เป็น P-pictures หรือ B-pictrues แล้วตัวเข้ารหัสจะไม่ดำเนินการกับ picture macroblock โดยตรง แต่มันจะดำเนินการกับ prediction error ใน MPEG เราจะเรียกว่า interframe predictive coding สำหรับ P-pictures กระบวนการนี้อธิบายในรูปที่ 2.29



รูปที่ 2.29 Forward motion compensation

กระบวนการ Motion estimation นี้จะให้ค่าโคออร์ดิเนตที่เป็นโมชั่นเวคเตอร์ของแต่ละมาโครบล็อกซึ่งแมทช์ (best match) มากที่สุดกับพื้นที่ค้นหา (search area) ของภาพที่ผ่านมา มาโครบล็อก 2 บล็อก จะถูกนำมาลบกันเพื่อส่งไปยัง DCT สำหรับภาพชนิด B-pictrues การกระทำ motion compensation จำเป็น 2 ส่วนคือ สำหรับภาพก่อนและภาพหลัง ดังนั้นจะได้ 2 motion vector ตัวเข้ารหัสจะสามารถสร้าง prediction error macroblock 1 บล็อก จากทั้งสองหรือค่าเฉลี่ยของทั้งคู่ โดยกระบวนการนี้แสดงในรูปที่ 2.30 และใน MPEG จะหมายถึงกระบวนการ interframe interpolative coding หลังจากนั้น prediction error จะถูกผ่านไปยัง block-based DCT และควอนไตซ์ ค่าที่ได้นี้จะถูกส่งไปพร้อมกับ motion vector โดยวิธีการมัลติเพล็กและใช้ variable length coder ในระบบ MPEG ไม่เหมือน JPEG เพราะจะใช้การเข้ารหัสแบบฮัฟแมนเท่านั้นสำหรับ variable length coder



รูปที่ 2.30 bidirectional motion compensation

เนื่องจากว่าเป็นการทำงานแบบสองทิศทาง จึงจำเป็นต้องมีการจัดเรียงภาพที่เข้ามาใหม่บ้างเพื่อจะส่งภาพในลำดับที่ถูกต้องให้กับตัวถอดรหัสอย่างเช่นในรูปที่ 2.26 ภาพจะต้องเรียง

โดยคำนึงถึงภาพที่จะต้องใช้ทำงานเข้ามาก่อนจะได้ดังนี้ P_1, P_4, P_2, P_3, P_7, P_5, P_6 และ P_8 และเมื่อถึงขั้นตอนการถอดรหัสจะต้องเรียงใหม่เพื่อให้ได้ลำดับภาพที่ถูกต้อง

2.17 Video decoder

รูปที่ 2.31 แสดงการทำงานของการถอดรหัสวีดีโอ MPEG ซึ่งเป็นวงจรที่ลักษณะเหมือนส่วนป้อนกลับของตัวเข้ารหัสวีดีโอ ขั้นแรกภาพที่เข้ามาจะถูกถอดรหัสฮัฟแมนพร้อมด้วยข้อมูลชนิดจาก header สำหรับแต่ละมาโครบล็อกตัวสัมประสิทธิ์จะถูกดีควอนไตซ์แปลงกลับไปสู่ spatial header ถ้าเรามาพิจารณาอนุกรมของภาพตัวอย่างคือ P_1, P_4, P_2, P_3, P_7, P_5, P_6 และ P_8 กระบวนการจะดำเนินด้วยขั้นตอนต่อไปนี้

1. ภาพ P_1 (I-picture) ไม่มีการทำ motion compensation โดยจะทำการ คำนวณ IDCT แล้วเก็บไว้ในบัฟเฟอร์แสดงผลและในบล็อก Previous picture store

2. ภาพ P_4 (P-picture) สำหรับแต่ละมาโครบล็อกจะถูกคำนวณ IDCT และเข้ากระบวนการ motion compensation โดยจะบวกพื้นที่ที่ชี้โดย future picture store

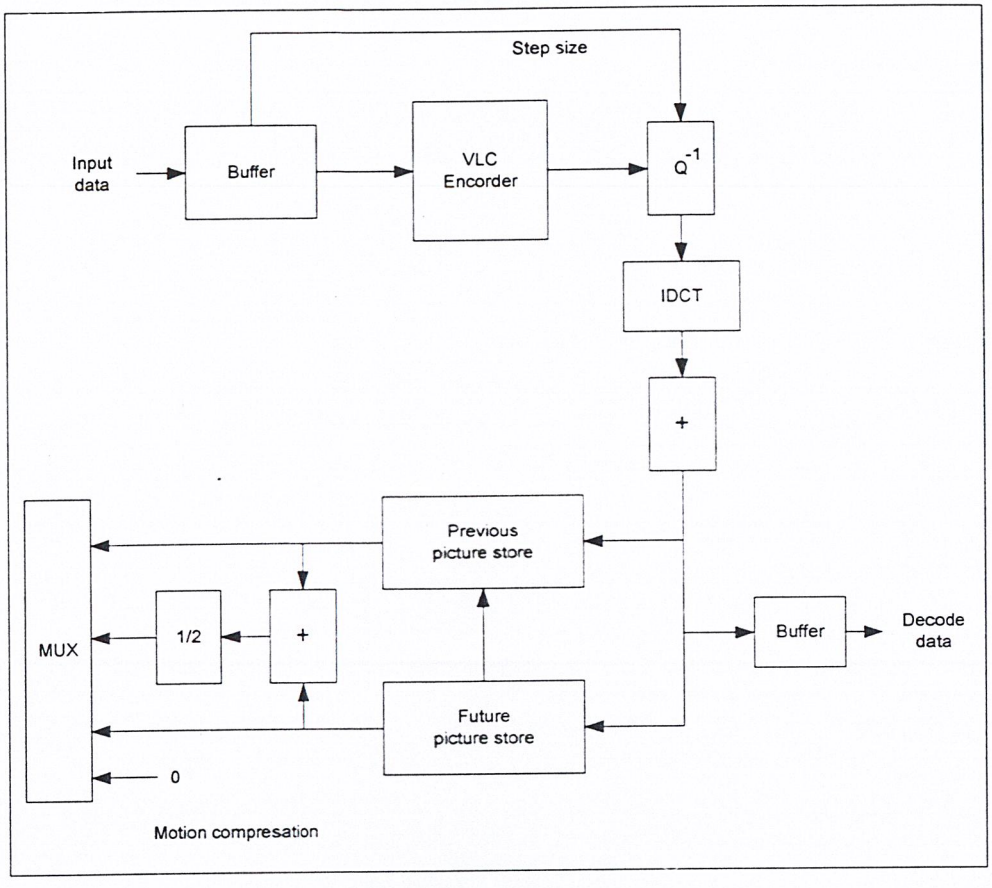
3. ภาพ P_2 (B-picture) คำนวณ IDCT แล้ว จะดำเนินกระบวนการ bidirectional motion compensation ซึ่งใช้ 2 motion victor เพื่อเข้าถึงพิกเซลที่สอดคล้องกับค่าในตำแหน่งของภาพ future picture และ ภาพ previous picture ที่เก็บไว้ทั้งสองทำการ predicted ให้ได้ภาพ P_2 แล้วรวมกันกับที่ทำ IDCT ถึงตอนนี้ P_2 จะเป็นภาพที่ตาม P_1 ไปและพร้อมที่จะแสดงทางหน้าจอได้ จากการทำงานจะสังเกตได้ว่า P_2 ซึ่งเป็นภาพแบบ B-picture จะไม่ต้องถูกเก็บไว้ในหน่วยความจำเนื่องจากไม่ต้องใช้ในการทำนายภาพต่อไป

4. ภาพ P_3 (B-picture) จะทำแบบเดียวกับกระบวนการที่ได้ภาพ P_2 ออกมา และหลังจากที่ได้แสดงผลภาพ P_3 แล้ว ภาพ ภาพ P_4 ก็จะถูกแสดงผลตามไป

5. ภาพ P_7 (P-picture) จะทำซ้ำเหมือนขั้นตอนที่สองและทำให้ข้อมูลในหน่วยความจำ previous picture ถูกเขียนทับ

6. ภาพ P_5, P_6 (B-picture) ทำเหมือนกับในขึ้นตอนที่สอง หลังจากที่ P_5, P_6 ถูกถอดรหัสแล้วเราจะเรียงการแสดงผลออกไปเป็น P_5, P_6, P_7, P_8 ตามลำดับ

จากขั้นตอนทั้งหมดนี้ถือว่าการถอดรหัสภาพหนึ่งกลุ่มได้เสร็จสิ้นสมบูรณ์ และจะดำเนินการทำขั้นตอนเดียวกันนี้กับกลุ่มภาพต่อไป จากผังการทำงานจะมีภาพมากสุดสามภาพที่ถูกเก็บไว้พร้อมกันในชั่วขณะหนึ่ง สำหรับมาตรฐาน MPEG จะมีความต้องการหน่วยความจำอย่างน้อยครึ่งแมกกะไบต์ ในการแสดงผลความละเอียดในแบบ SIF



รูปที่ 2.31 Block diagram of an MPEG decoder

2.18 โครงสร้างของในการเข้ารหัสของบิทสตรีมที่เป็นภาพเคลื่อนไหว (Structure of the Coded Video bit stream)

MPEG-1 กำหนดให้ Video bit stream มีการเข้ารหัสแบบ Hierarchical ซึ่งมีทั้งหมด 6 ชั้น คือ (1) a sequence layer (2) a group of picture layer (3) a picture layer (4) a slice layer (5) a macroblock layer (6) a block layer ดังแสดงในรูปที่ 2.32

1. ชั้นที่สูงที่สุดก็คือ sequence layer มันจะรวมถึง header และตามด้วยกลุ่มของภาพหนึ่งกลุ่มหรือหลายกลุ่ม และจบด้วยรหัส sequence end ข้อมูลที่อยู่ใน header นั้นประกอบไปด้วยขนาดในแนวตั้งและแนวนอนของแต่ละภาพ มาตราส่วนของพิกเซล (pixel aspect ratio) อัตราความเร็วภาพต่อวินาที บิตเรตในหน่วย 400 บิตต่อวินาที และค่าหน่วยความจำต่ำสุดที่ต้องใช้โดยตัวถอดรหัส (หน่วยของ 2,048) นอกจากนั้น header ยังต้องประกอบด้วย DCT quantization matrix ของทั้ง intra-pictures และ nonintra picrures และข้อมูลตัวเลือกของผู้ใช้ สำหรับ intra-pictures จะ

ใช้ default quantization matrix ที่กำหนดโดยมาตรฐาน JPEG สำหรับองค์ประกอบแสง สำหรับ nonintra pictures ค่าทั้งหมดใน default quantization matrix จะเป็น 16

2. ชั้น Group of pictures (GOP) คือภาพจำนวนหนึ่งที่แสดงผลต่อเนื่องกันเป็นลำดับและอย่างน้อยต้องมี I-picture หนึ่งภาพ สามารถเริ่มต้นด้วย I-picture หรือ B-picture ก็ได้ และต้องจบด้วย I-picture หรือ P-picture อย่างใดอย่างหนึ่ง ถ้าภาพแรกเป็น I-picture แสดงว่ามันจะไม่เกี่ยวข้องกับ GOP ก่อนหน้ามัน ดังนั้น GOP นี้จะสามารถดำเนินการและแสดงผลได้อิสระ เพียงลำพังโดยไม่ขึ้น GOP อื่น ๆ และเราจะเรียกว่า closed GOP

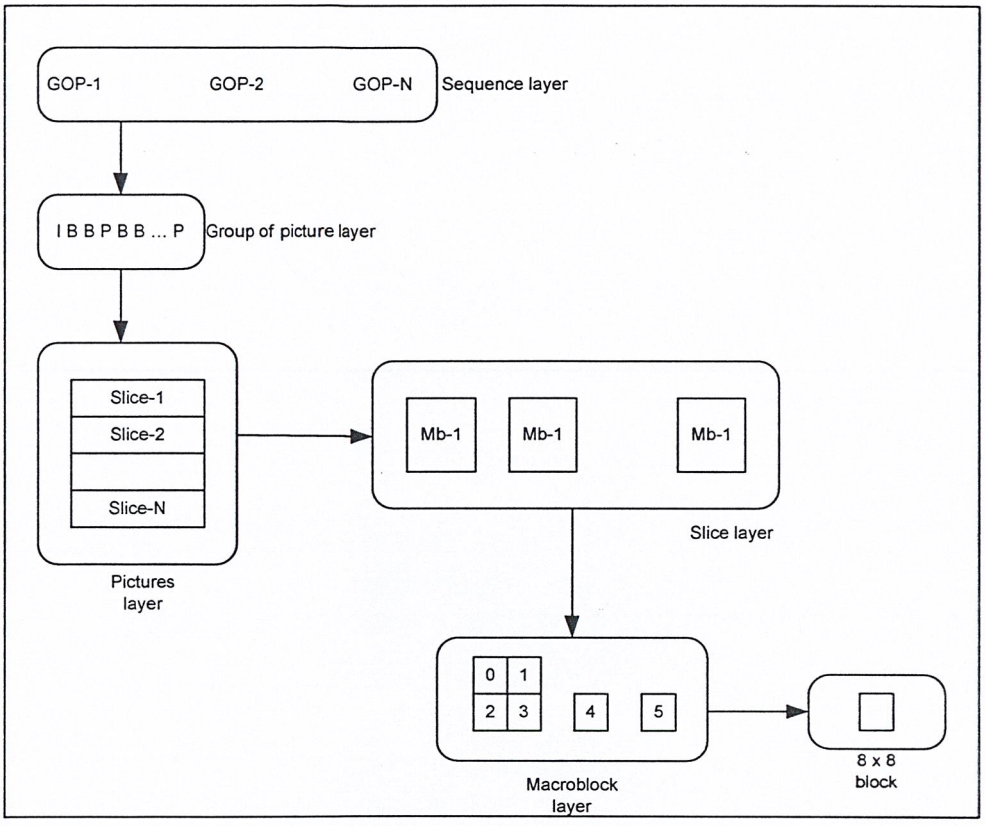
3. ชั้น Picture ถูกกำหนดให้เป็นข้อมูลการเข้ารหัสสำหรับแต่ละภาพ เพื่อให้สามารถเรียง P-picture ได้ โดยที่ header จะเตรียมข้อมูลชั่วคราวในการที่จะเรียงภาพที่จะแสดงผลให้เป็นลำดับถูกต้อง และนอกจากนั้น header ยังประกอบไปด้วยข้อมูล ชนิดของภาพ การเข้าจังหวะ (synchronization) ความละเอียดภาพ และขนาดของ motion vector

4. ชั้น Slice โดยขนาดของชั้นนี้ สามารถที่จะใหญ่ถึงภาพหรือเล็กแค่เพียงหนึ่งมาโครบล็อกได้ ในกรณีที่เกิดความเสียหายของข้อมูล ตัวถอดรหัสจะสามารถทำการสร้างข้อมูลขึ้นใหม่โดยใช้ข้อมูลจาก Slice headers ตัวอย่างเช่นการส่งข้อมูลเกิดมีปัญหาการผิดพลาด ตัวถอดรหัสจะลดขนาดของ Slice ให้เล็กลงไม่เต็มภาพ ใน Slice headers จะบรรจุข้อมูลตำแหน่งของมันในภาพและ quantizer scale factor ค่าระหว่าง 1 ถึง 31 เพื่อให้ตัวถอดรหัสใช้ในการ Dequantize

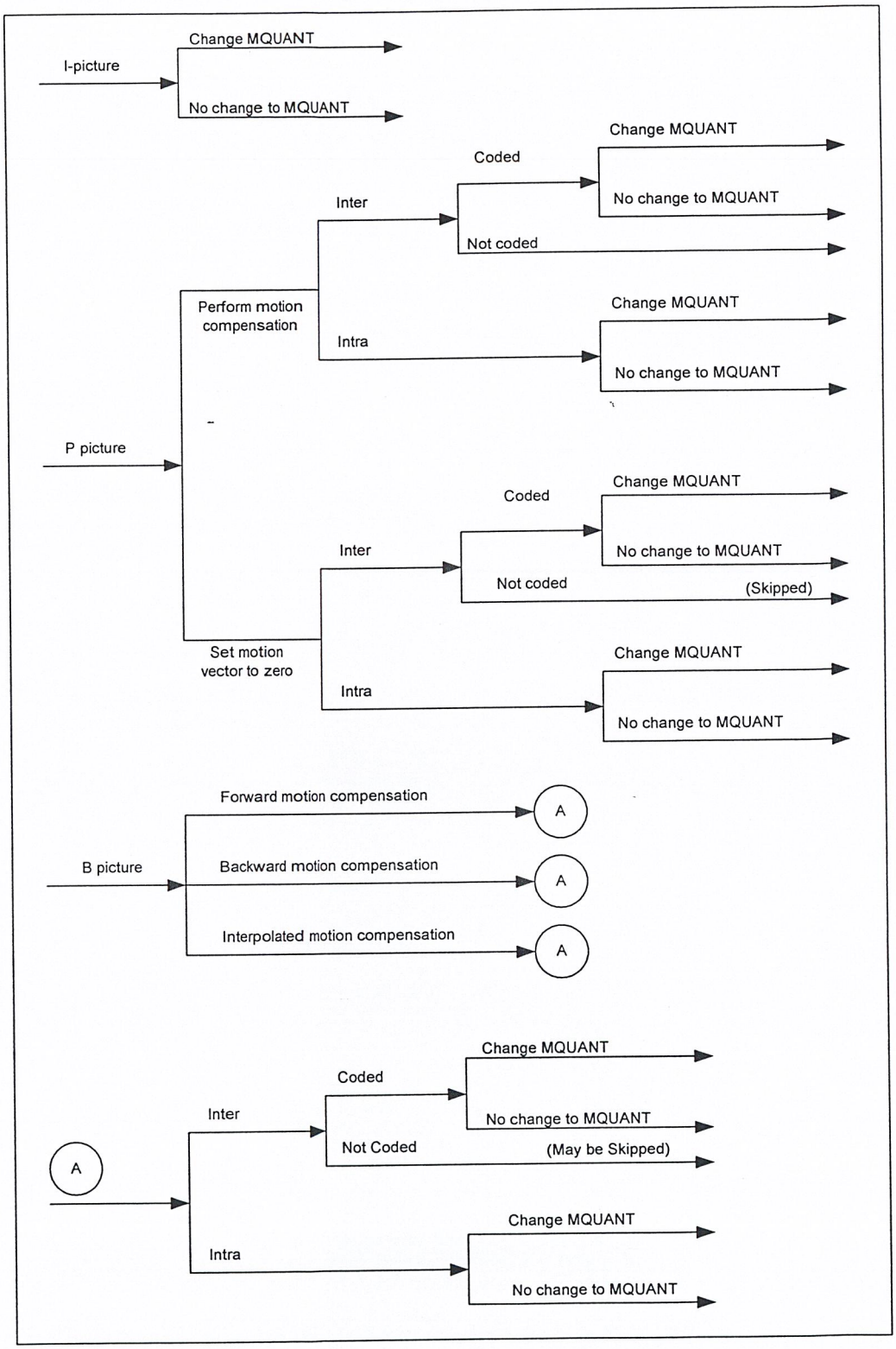
5. ชั้น Macroblock และ block ใน slice จะถูกแบ่งออกเป็น macroblock ซึ่งที่ header ของ macroblock จะบอก ชนิดของมาโครบล็อก ตำแหน่งของข้อมูล รหัสของ motion vectors ในแนวตั้งและแนวนอน

2.18.1 Mocroblock coding

จากที่กล่าวมาแล้วเราแบ่งภาพเป็นภาพหลัก 3 แบบ คือ I- picture, B-picture และ P-picture แต่อย่างไรก็ตามในแต่ละแบบนั้น การดำเนินการกับรหัสในภาพชนิดนั้น ๆ จะต่างกันออกไป ซึ่งสามารถแสดงเป็นแผนภูมิสาขาตามรูปที่ 2.33



รูปที่ 2.32 Syntax layers in MPEG video coding



รูปที่2.33 Decision trees for coding macroblock in I- P- and B-picture

บทที่ 3
การออกแบบระบบ

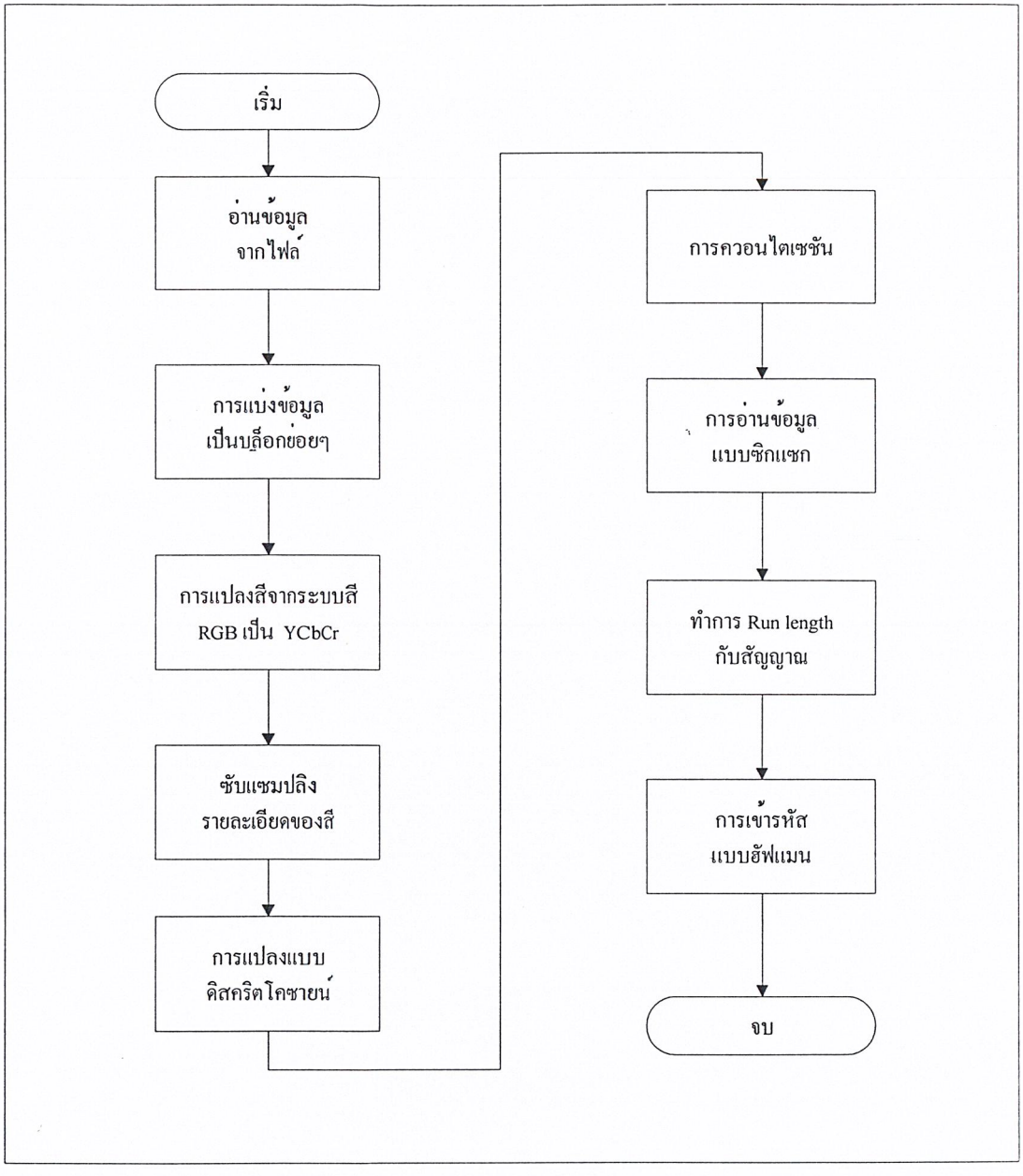
ผู้เขียนได้ออกแบบและเขียนโปรแกรมสำหรับแสดงการบีบอัดภาพเพื่อใช้ทดลองในโครงงานนี้ โดยโฟลว์ชาร์ต(Flowchart) การทำงานคร่าว ๆ ของโปรแกรม ในการบีบอัดและขยายภาพแต่ละ เฟรม (Frame) เป็นดังนี้

3.1 กระบวนการทำภาพ JPEG

ภาพต้นฉบับจะถูกแบ่งออกเป็นบล็อก (Block) เล็ก ๆ ขนาด 8×8 พิกเซล ถ้าเป็นภาพที่แสดงผลด้วยจำนวน n บิทต่อพิกเซลแล้ว ค่าข้อมูลพิกเซลจะมีค่าตั้งแต่ 0 ถึง 2^n - 1 ก่อนที่จะผ่าน การแปลงฟอร์เวิร์ดดิสครีตโคซายน์ (Forward Discrete Cosine Transform หรือ FDCT) จะมีการสร้างค่าของพิกเซลให้เป็นจำนวนเต็มแบบมีเครื่องหมาย ซึ่งมีค่าตั้งแต่ -2^n ถึง 2^n - 1 ตัวอย่างเช่น ภาพแบบเกรย์สเกลที่มี 8 บิทต่อพิกเซล (ความเข้ม 2^8 = 256 ระดับ) มีค่า 0 – 255 เมื่อจะผ่านการแปลงต้องทำข้อมูลให้เป็น -128 ถึง 127 ก่อน

เมื่อผ่านการแปลงโคซายน์แบบไม่ต่อเนื่อง (DCT) แล้วเอาท์พุทที่ได้จะเป็นค่าสัมประสิทธิ์ทางความถี่ ต่อมาจะเป็น การควอนไตซ์ (Quantized) ค่าสัมประสิทธิ์เหล่านี้ ในขั้นตอนนี้เองที่จะมีการลดขนาดของข้อมูลและเกิดมีความผิดพลาดจากการควอนไตซ์ขึ้น ในการควอนไตซ์นี้จะเป็นแบบไม่เป็นยูนิฟอร์ม (Non - Uniform) กล่าวคือค่าสัมประสิทธิ์แต่ละค่าจะถูกควอนไตซ์ด้วยระดับการควอนไตซ์ (Step Size) ที่ไม่เท่ากันเสมอ ขนาดของระดับขั้นหรือสเตป (Step) ในการควอนไตซ์จะถูกกำหนดโดยค่าในตารางเรียกว่าเป็นตารางค่าควอนตัม (Quantum) นั่นเอง ซึ่งค่าในตารางนี้จะเป็นค่าที่ใช้เฉพาะกับสัมประสิทธิ์ ที่ได้จากการแปลงโคซายน์แบบไม่ต่อเนื่องแต่ละค่าเท่านั้น

ค่าสัมประสิทธิ์ที่ถูกควอนไตซ์และจะถูกนำมาจัดตามลำดับซิกแซก (Zigzag) เพื่อความสะดวกในการเข้ารหัสซึ่งอาจใช้วิธีของฮัฟแมน (Huffman encoding) หรือวิธีการเข้ารหัสแบบความยาวของรหัสไม่คงที่ (Variable Length Encoding) เอาท์พุทจากกระบวนการนี้จะเป็นข้อมูลภาพที่ถูกลดขนาดเรียบร้อยแล้ว



ผังงานที่ 3.1แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมการเข้ารหัสของเฟรมภาพ

3.2 การแปลงข้อมูลจาก RGB ไปเป็น YCrCb

มนุษย์สามารถมองเห็นสีได้เมื่อความยาวคลื่นแสงที่สามารถมองเห็นได้บางช่วงตกกระทบที่เรตินาของดวงตาบางคนสามารถเห็นสีเดียวกันได้จากการรวมกันของความยาวคลื่นที่ตากต่างกันจำนวนมาก จำนวนของการรวมกันของความยาวคลื่นต่าง ๆ ไม่มีขีดจำกัดแต่จำนวนของสีที่สามารถแยกแยะได้โดยมนุษย์มีขีดจำกัดดังนั้น "รูปแบบของสี" จึงเป็นที่ต้องการเพื่อที่จะแทนรหัสที่เป็นเอกลักษณ์สำหรับแต่ละสีที่มนุษย์สามารถเห็นได้ รูปแบบของสีที่มีอยู่มีเป็นจำนวนมาก

เนื่องจากระบบสีในบิทแมพไฟล์ นั้นจะเป็นระบบ RGB แต่ระบบสีที่ใช้ใน JPEG นั้นจะเป็นระบบสี YCbCr ซึ่งจะประกอบด้วยการเปลี่ยนแปลงความเข้มของแสง (luminance) ซึ่งแทนด้วย Y = 0.30R + 0.59G +0.11B และการเปลี่ยนแปลงความเข้มของสี (chrominance) ซึ่งแทนด้วย Cb = 0.493(B-Y) และ Cr = 0.877(R-Y) โดยมีสมการการแปลงสีในระบบสี RGB เป็นระบบสี YCbCr ดังนี้

$$Y = 0.299\ R + 0.587\ G + 0.114\ B$$

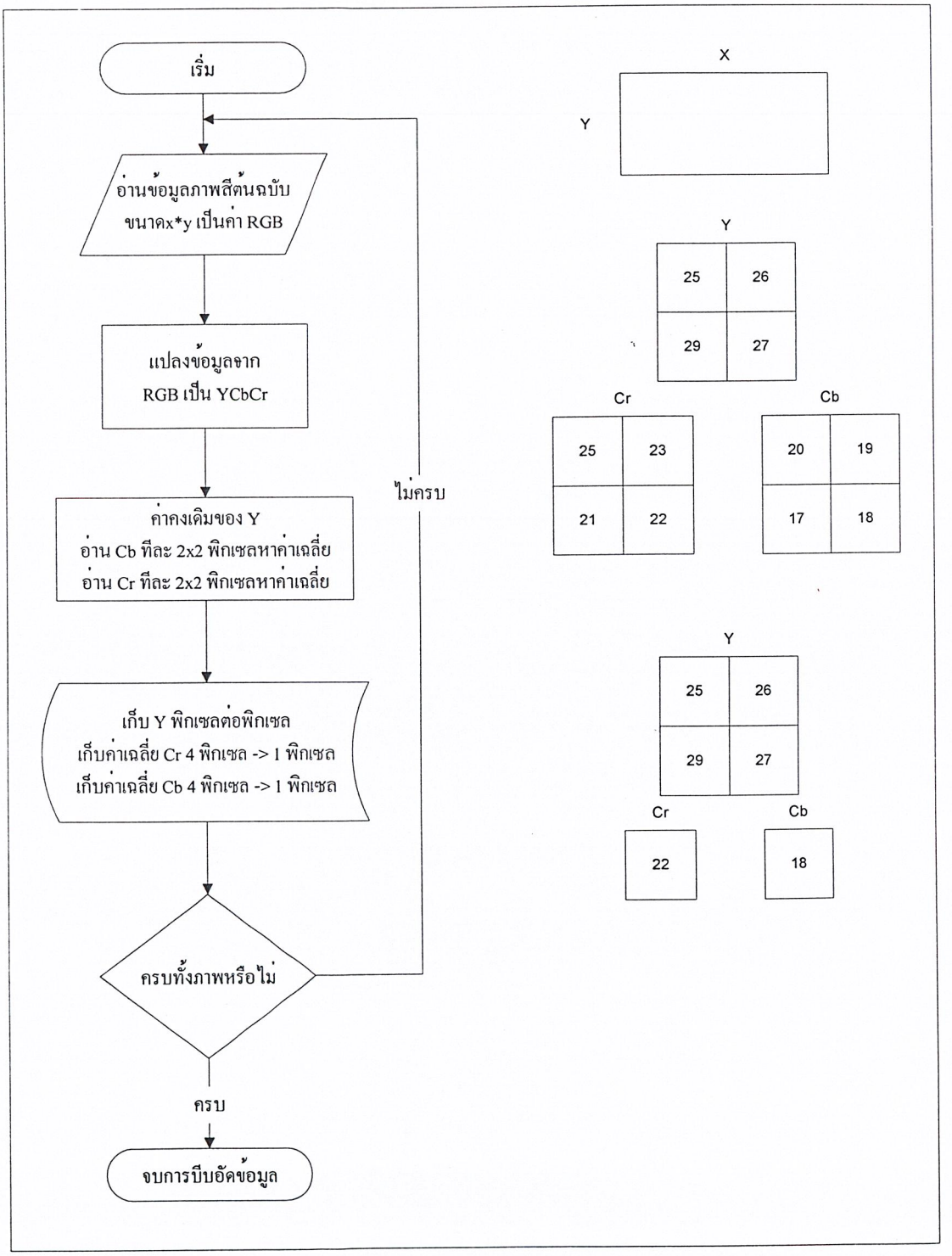
$$Cb = -0.1687R - 0.3313\ G + 0.5\ B$$

$$Cr = 0.5000R - 0.4187\ G - 0.0813\ B$$

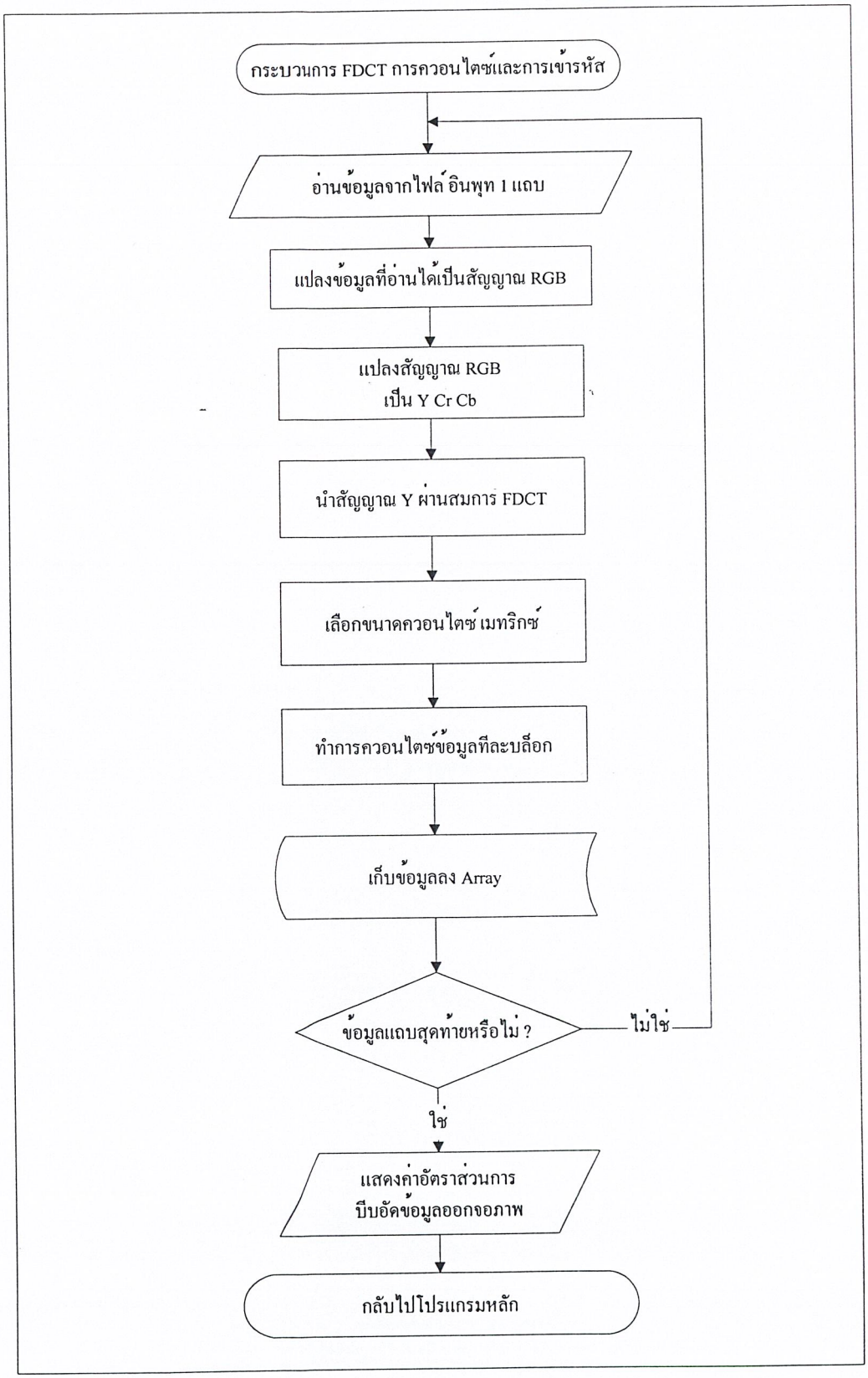
โดยที่ ค่า Y อยู่ในช่วง 0 ถึง 255, ค่า Cb อยู่ในช่วง 0 ถึง 255, ค่า Cr อยู่ในช่วง 0 ถึง 255

3.3 ซับแซมปลิ้ง

การซับแซมปลิง เป็นการลดจำนวนข้อมูลด้วยการแทนกลุ่มของข้อมูลด้วยค่าเฉลี่ยของกลุ่มข้อมูลนั้น วิธีของการซับแซมปลิง ข้อมูลสีสำหรับข้อมูลภาพดิจิตอลมีหลายวิธี วิธีหนึ่งถูกแสดงโดยชุดของค่า 3 ค่า ซึ่งแยกโดยเครื่องหมายโคลอน (:) ซึ่งค่าเหล่านี้จะแสดงจำนวนความสัมพันธ์ของตัวอย่างจากแต่ละส่วนประกอบของภาพ ค่าแรกแทน Y ซึ่งคือ การเปลี่ยนแปลงของความเข้มแสง ค่าที่สองแสดง Cr (R-Y) และค่าที่สามแสดง Cb (B-Y) ซึ่งค่าที่สองและสามคือ การเปลี่ยนแปลงของความเข้มสีสองรูปแบบที่ใช้กันโดยทั่วไปใช้อัตราส่วนการซับแซมปลิง เป็น 4:2:2 และ 4:1:1 รูปแบบ 4:2:2 แสดงว่าสำหรับทุก ๆ สี่ตัวอย่างของ ค่าการเปลี่ยนแปลงความเข้มของแสง จะมีสองตัวอย่างของแต่ละแบบของ ค่าการเปลี่ยนแปลงความเข้มของสี ซึ่งอาจกล่าวได้ว่าทุก ๆ สองตัวอย่างของค่าการเปลี่ยนแปลงความเข้มของแสง มีหนึ่งตัวอย่างของแต่ละ ค่าการเปลี่ยนแปลงความเข้มสี ส่วนรูปแบบ 4:1:1 แสดงว่าสำหรับทุก ๆ สี่ตัวอย่างของ ค่าการเปลี่ยนแปลงความเข้มของแสงจะมีหนึ่งตัวอย่างของแต่ละ ค่าการเปลี่ยนแปลงความเข้มของสี ซึ่งในที่นี้เราใช้การซับแซมปลิง แบบ 4:1:1



ผังงานที่ 3.2 การบีบอัดข้อมูลแบบ Chroma 2 ทิศทาง 4:1:1



ผังงานที่ 3.3 กระบวนการ FDCT, ควอนไตซ์และการเข้ารหัส

3.4 การถอดรหัสของเฟรมภาพ

1. นำข้อมูลที่เข้ารหัสแล้วมาทำการถอดรหัสจะทำให้ได้ข้อมูลกลับมาเป็นบล๊อกขนาดเท่าเดิมซึ่งจะเป็นข้อมูลที่มีการลดขนาดและมีการสูญเสียข้อมูลบางส่วนไปด้วย

2. เมื่อได้ข้อมูลกลับมาเป็นบล๊อกย่อย ๆ แล้วก็นำไปทำการดีควอนไตซ์ เพื่อให้ข้อมูลที่เป็นศูนย์ส่วนใหญ่กลับมาเป็นข้อมูลที่มีส่วนประกอบต่างๆของภาพโดยนำไปคูณกับค่าควอนไตซ์เมทริกซ์อีกครั้งก็จะได้ข้อมูลที่พร้อมจะนำไปทำการทรานฟอร์มแบบย้อนกลับ

3. นำข้อมูลที่ได้จากการดีควอนไตซ์ ไปเข้าสมาการเพื่อทำการทรานฟอร์มแบบย้อนกลับเพื่อให้ได้ข้อมูลที่เป็นสัญญาณความสว่างและส่วนที่เป็นส่วนประกอบของสีกลับคืนมา

4. นำข้อมูลในส่วนที่เป็น Cb และ Cr มาทำการดีแซมปลิงเพื่อให้ได้ส่วนประกอบของสีที่ทำการลดขนาดลงไปแบบ 4:1:1 ซึ่งจะมีค่าเพียงช่องเดียวจาก 4 ช่องให้กลับมาเป็น 4 ช่องเหมือนเดิม

5. เปลี่ยนระบบสีแบบ YCbCr ให้กลับมาเป็นระบบสีแบบ RGB ทีละจุด

6. นำข้อมูลสีที่แปลงเสร็จแล้วมารวมเป็นบล๊อกย่อย ๆ เมื่อเดิมเพื่อที่จะนำไปแสดงออกทางจอภาพให้เห็น

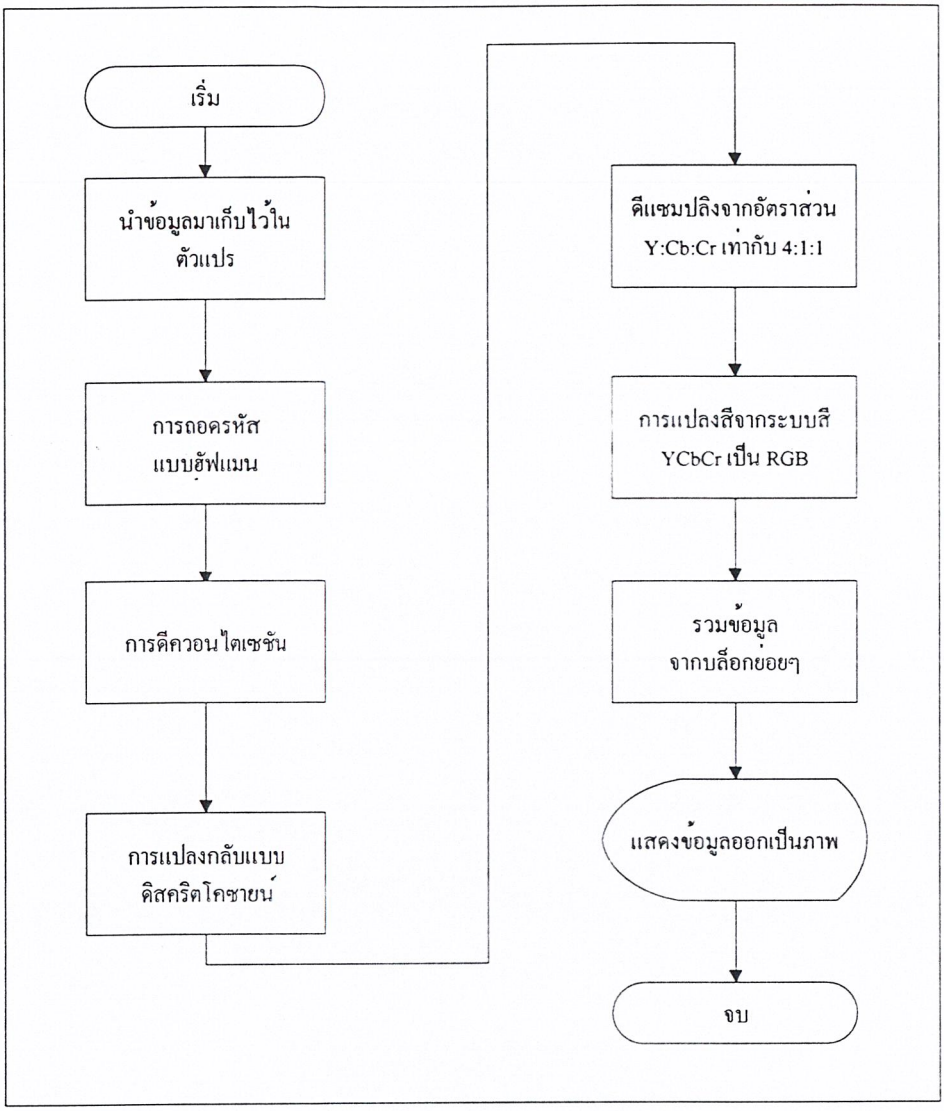
3.5 การขยายข้อมูลแบบ Chorma 2 ทิศทาง 4:1:1

1. อ่านข้อมูลที่เป็นส่วนประกอบของสีที่ได้จากการทรานฟอร์มแบบย้อนกลับทีละจุด

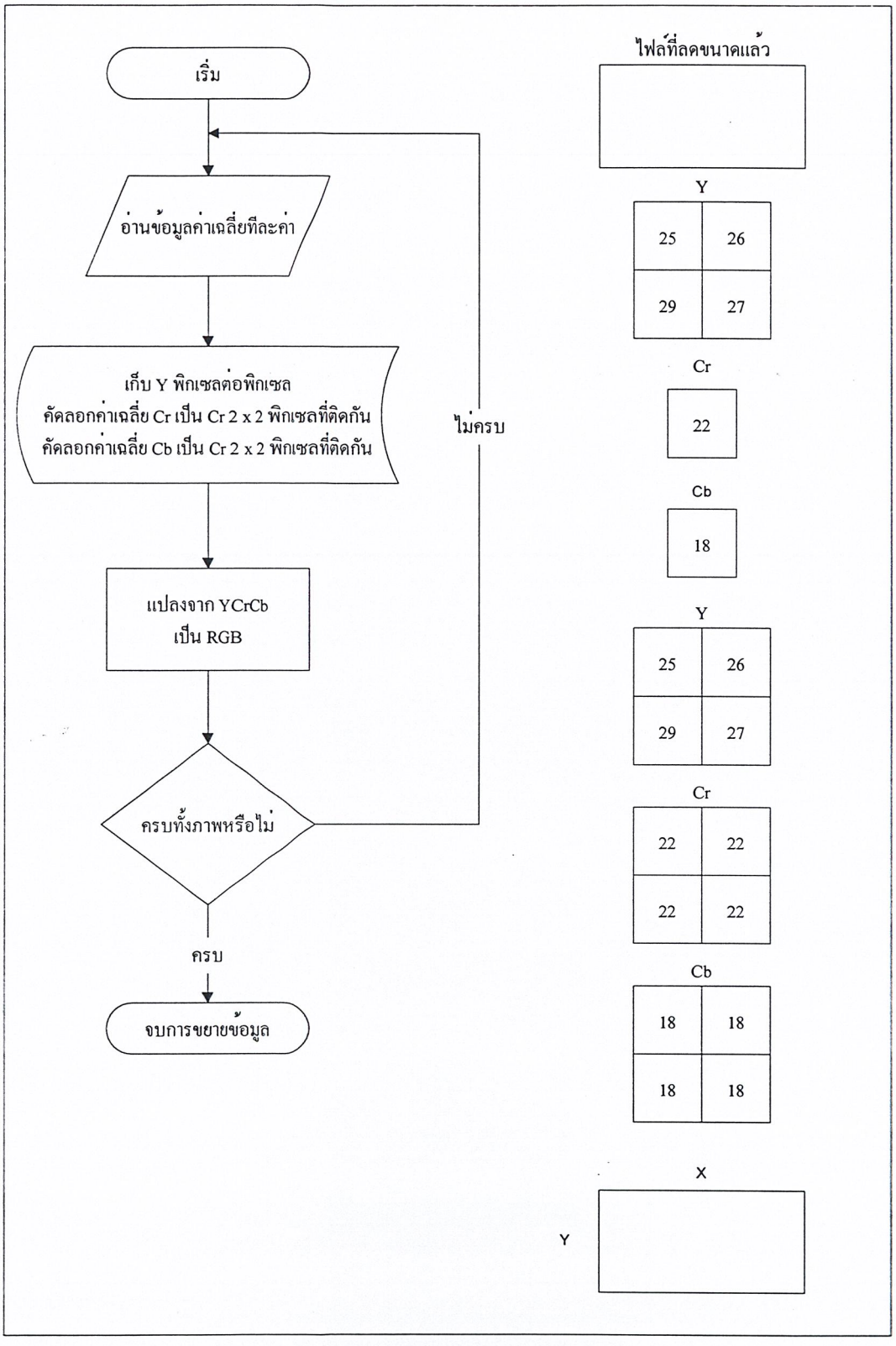
2. นำข้อมูลที่ได้ แต่ละจุดเก็บลงในตัวแปล 4 ตัวโดย 1 จุดจะเก็บไว้ในตัวแปล 4 จุดโดยจะมีค่าเดียวกันทั้ง 4 จุด ทั้ง Cb และ Cr

3. แปลงจากสัญญาณ YCbCr เป็น RGB ที่ละจุดโดยจะแปลงจากค่า 4 จุดที่ซ้ำกัน

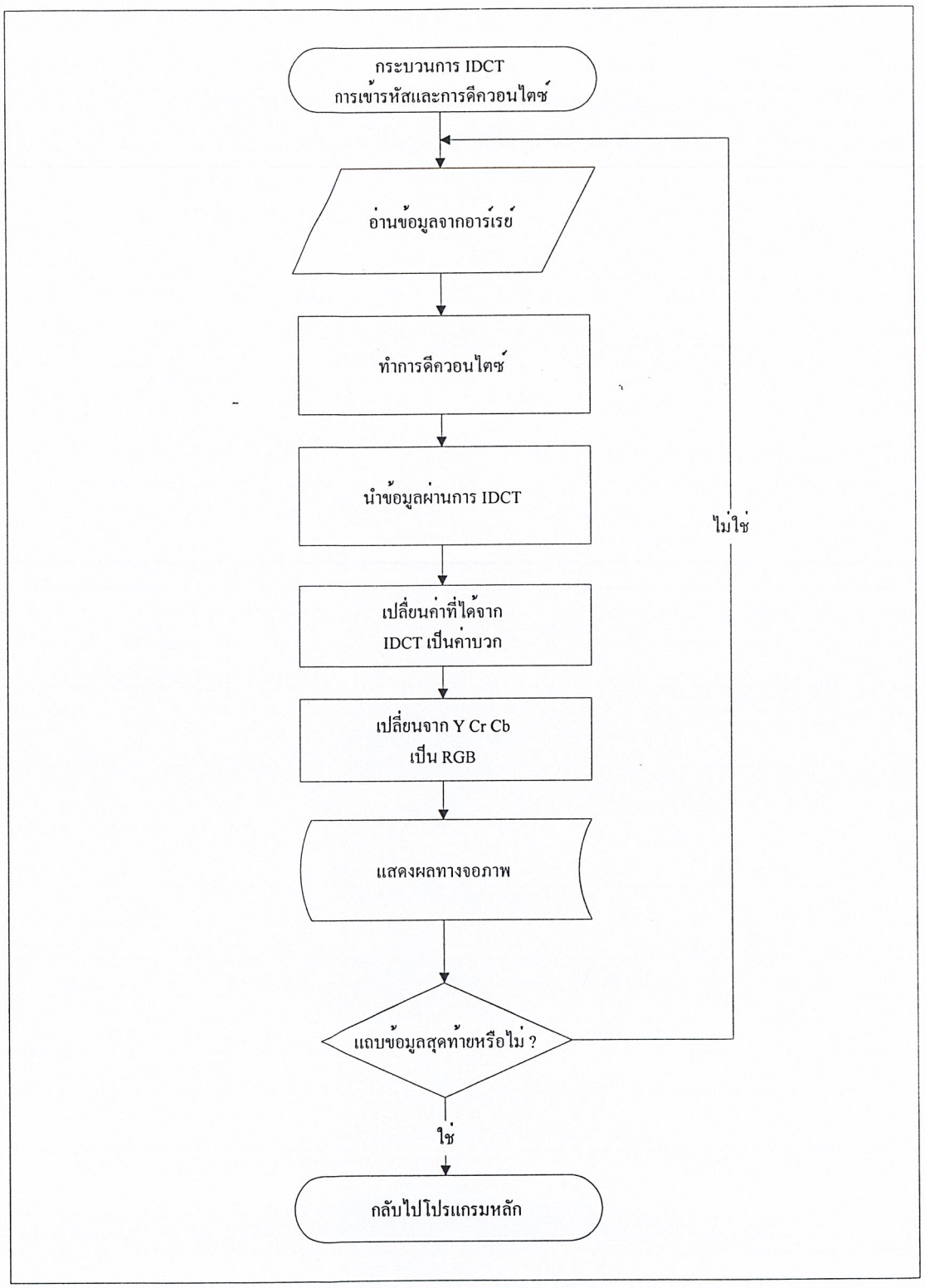
4. ตรวจสอบว่าครบทุกจุดหรือไม่ ถ้ายังให้กลับไปอ่านจุดต่อไปถ้าครบแล้วให้จบการทำงานการ ดีซับแซมปลิง



ผังงานที่ 3.4 แสดงโฟลว์ชาร์ตการทำงานของโปรแกรมการถอดรหัสของเฟรมภาพ



ผังงานที่ 3.5 การขยายข้อมูลแบบ Chroma 2 ทิศทาง 4:1:1



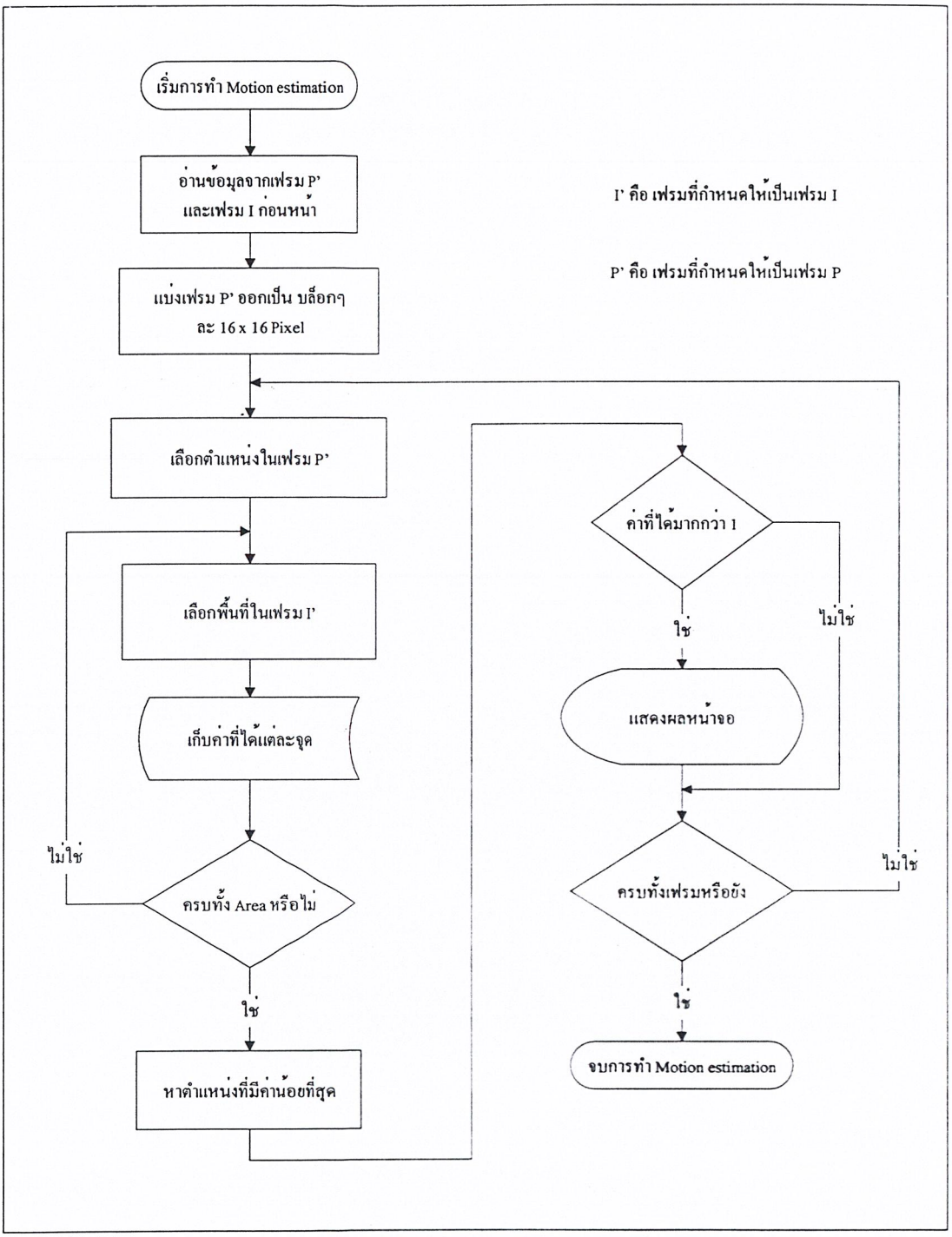
ผังงานที่ 3.6 กระบวนการ IDCT, ดีควอนไตซ์และการถอดรหัส

3.4 Motion Estimate

วิธีการค้นหาแบบเต็มรูปแบบ (Full-Search Method) คือ วิธีที่ง่ายที่สุดในการค้นหาเวคเตอร์ของการเคลื่อนที่สำหรับแต่ละมาโคร-บล็อกก็คือ การคำนวณหา MAE (i, j) ของแต่ละตำแหน่งในพื้นที่ของการค้นหา (search space) ซึ่งเรียกว่า อัลกอริทึมของการค้นหาแบบเต็มรูปแบบ แต่วิธีนี้มีการคำนวณที่ยุ่งยากมาก แต่ก็ได้รับการรับรองว่าจะสามารถหาค่า MAE ที่น้อยที่สุดได้ ดังนั้นเราจึงหาวิธีปรับปรุงและพัฒนาวิธีการใหม่ ๆ ขึ้นมาเพื่อลดความยุ่งยากในการคำนวณดังกล่าว แต่ก็ยังไม่สามารถรับรองได้ว่าจะได้ค่า MAE ที่น้อยทีสุดหรือไม่เราสามารถเขียนสมการ MAE (Absolute Mean Error) ได้ว่า

$$MAE(i,j) = \frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}\left|C(x+k,y+1) - R(x+i+k,y+j+l)\right|$$

จากบรรทัดฐาน MAE เรากำหนดไว้ว่าตำแหน่ง Matching block เปรียบเทียบที่ดีที่สุด คือ จุดที่ MAE (i, j) นั้นจะเป็นตำแหน่งของ Motion vector ดังนั้นจะนำเราไปสู่ อัลกอริธึมในการ Search เวคเตอร์ (u, v) ที่มีเป้าหมายตามบรรทัดฐานนี้

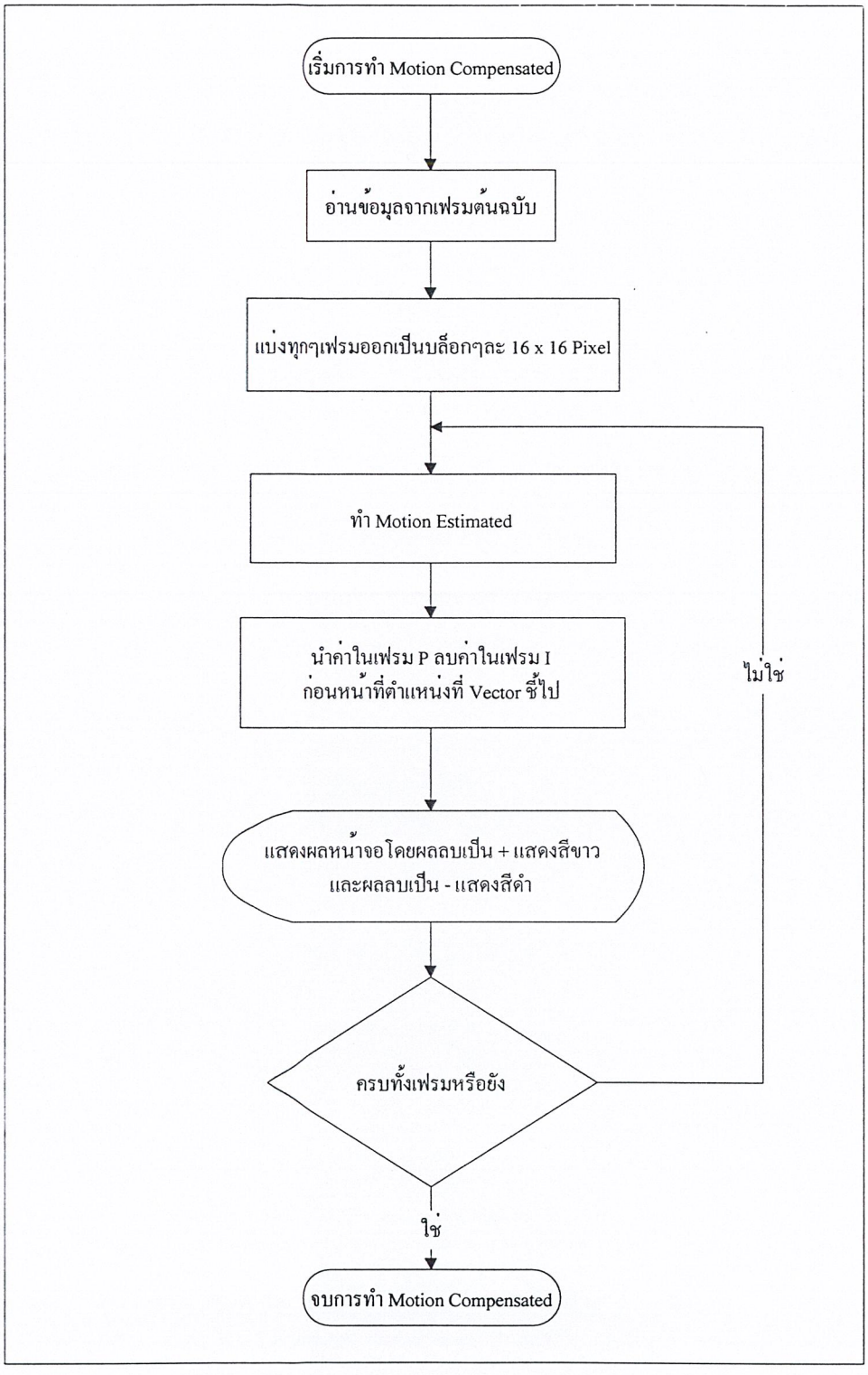


ผังงานที่ 3.7 แสดงขั้นตอนการทำงานของ Motion estimation ของเฟรม P

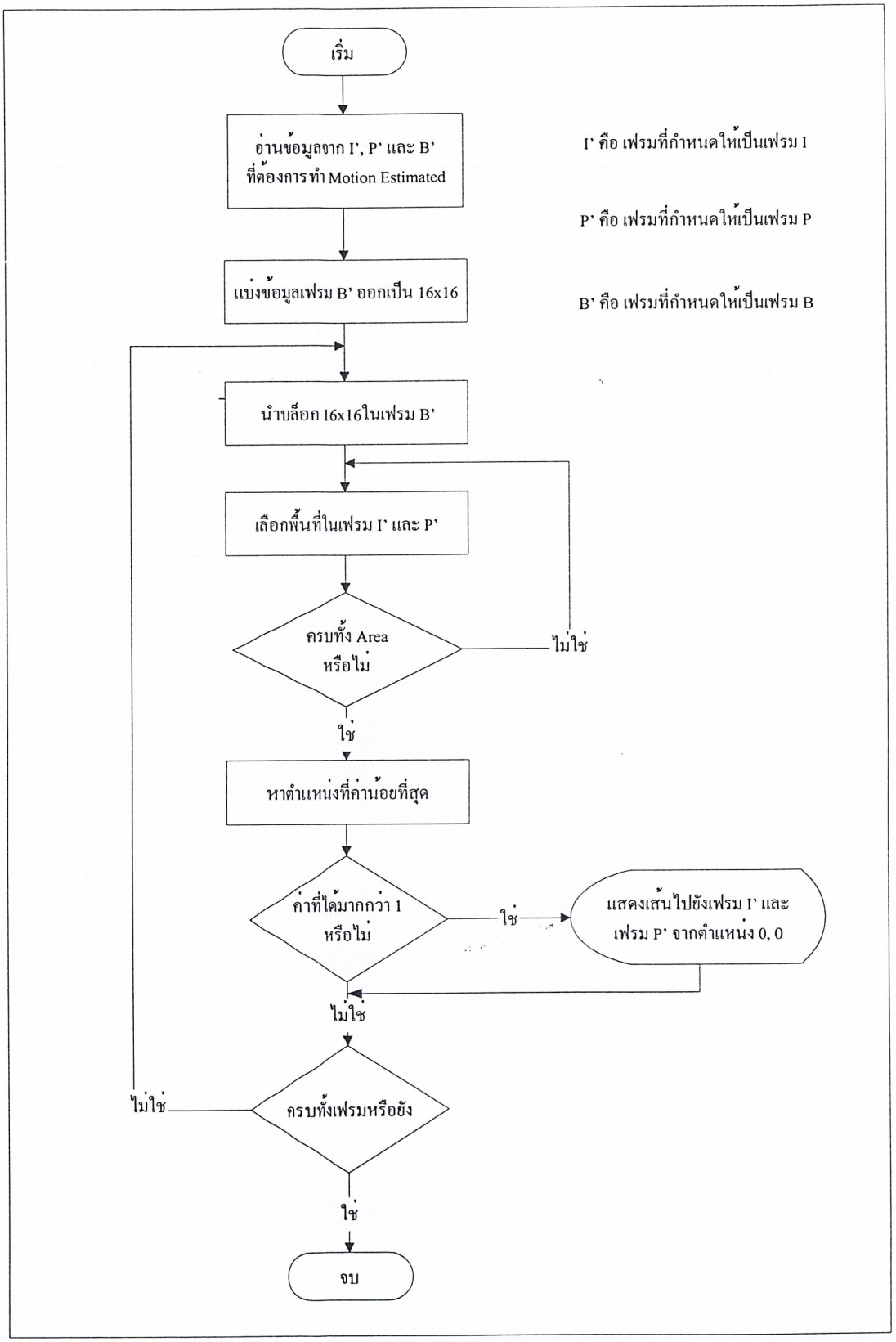
3.5 Motion Compensated

ด้วย การประมาณการเคลื่อนที่จะเป็นตัวกำหนดบล็อกขนาด N×M ที่อยู่ในภาพอ้างอิงซึ่งสอดคล้องกัน (Match) กับลักษณะของมาโครบล็อกของภาพปัจจุบันมากที่สุด โดยภาพปัจจุบันเราให้เป็นภาพนิ่งหรือเฟรมที่เวลา t ส่วนภาพอ้างอิงเราจะกำหนดให้เป็นภาพนิ่งหรือ เฟรมทั้งเวลาที่เป็นอดีตคือ t-n สำหรับการประมาณค่าการเคลื่อนที่แบบไปข้างหน้า (Forward motion estimation) และเวลาที่เป็นอนาคต t-k สำหรับการประมาณค่าการเคลื่อนที่แบบย้อนกลับ (Backward motion estimation) ในกรณีนอกเหนือจากนี้ เรขาคณิตของบล็อกที่สอดคล้องกัน (matching block) ในภาพอ้างอิงไม่จำเป็นต้องมีรูปทรงเหมือนบล็อกที่อยู่ปัจจุบัน แต่อย่างไรก็ตามในมาตรฐานการเข้ารหัสภาพเคลื่อนไหว เพียงแค่ใช้รูปทรงสี่เหลี่ยมเป็นเรขาคณิตของบล็อกที่สอดคล้องกันก็เพียงพอแล้ว

ตำแหน่งของ macroblock มักจะถูกกำหนดให้เป็น (x, y) ซึ่งเป็นตำแหน่งที่มุมซ้ายด้านบนของพื้นที่ ในทางอุดมคติเราต้องการที่จะหา (search) ภาพอ้างอิงทั้งภาพที่ match มากที่สุด แต่ว่ามั่นไม่สามารถนำไปใช้ในทางปฏิบัติได้ ดังนั้นเราจะทำการจำกัดการค้นหา ที่พื้นที่ [-p, p] รอบ ๆ จุดกำเนิดของ macroblock



ผังงานที่ 3.8 แสดงขั้นตอนการทำงานของ Motion Compensated ของเฟรม P



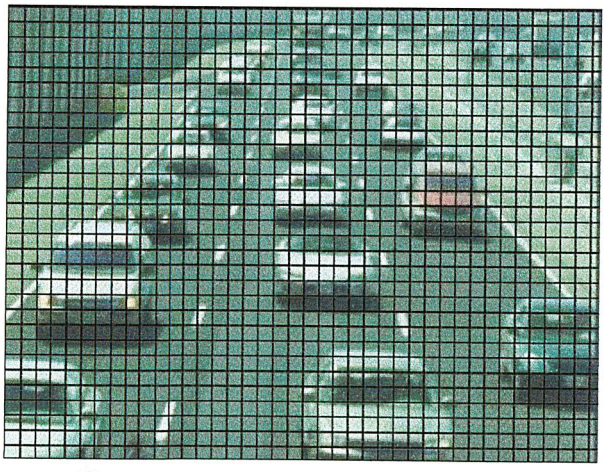
ผังงานที่ 3.9 แสดงขั้นตอนการทำงานของ Motion estimation ของเฟรม B

บทที่ 4

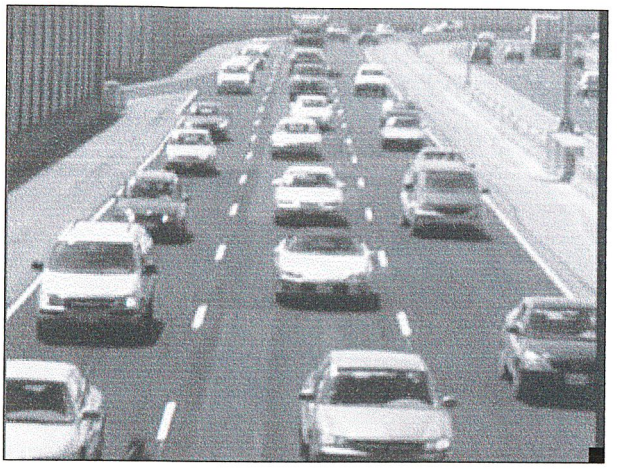
ผลการทดลอง

ในบทนี้จะกล่าวถึงผลการทดลองในการแสดงผลของการหาของเวกเตอร์และความแตกต่างของสีที่เกิดขึ้นในขั้นตอนการทำงาน ซึ่งรวมไปถึงการแสดงการทำงานของการบีบอัดภาพแบบ JPEG ซึ่งเป็นส่วนหนึ่งในการทำงานแบบ MPEG-1

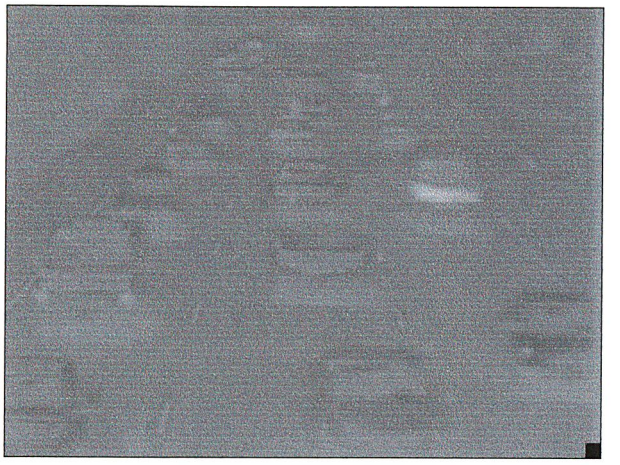
รูปที่ 4.1 แสดงถึงรูปต้นฉบับที่เป็น RGB ยังไม่ได้การลดขนาด



รูปที่ 4.2 แสดงถึงรูปต้นฉบับที่เป็น RGB ยังไม่ได้การลดขนาดและมี Grid

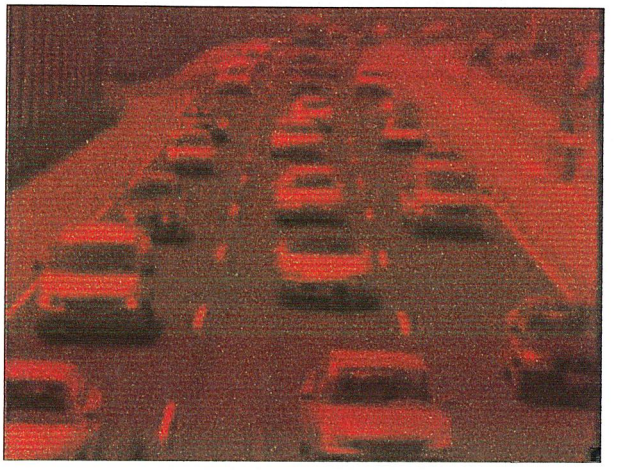


รูปที่ 4.3 แสดงถึงการถอดเอาเฉพาะสัญญาณ Y ที่แยกออกมาจากภาพต้นฉบับ



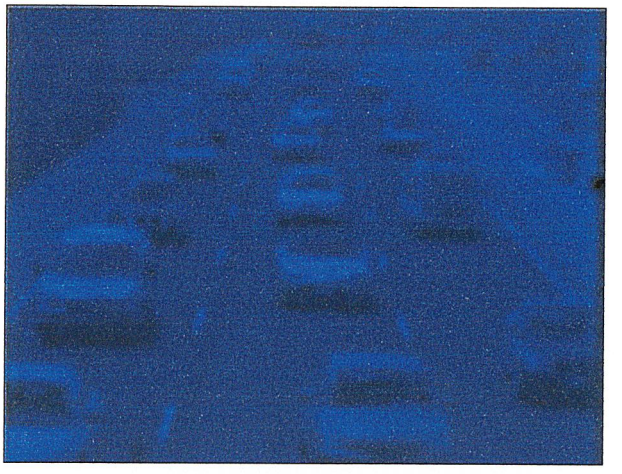
รูปที่ 4.4 แสดงถึงการถอดเอาเฉพาะสัญญาณ Cr ที่แยกออกมาจากภาพต้นฉบับ

รูปที่ 4.5 แสดงถึงการถอดเอาเฉพาะสัญญาณ Cb ที่แยกออกมาจากภาพต้นฉบับ

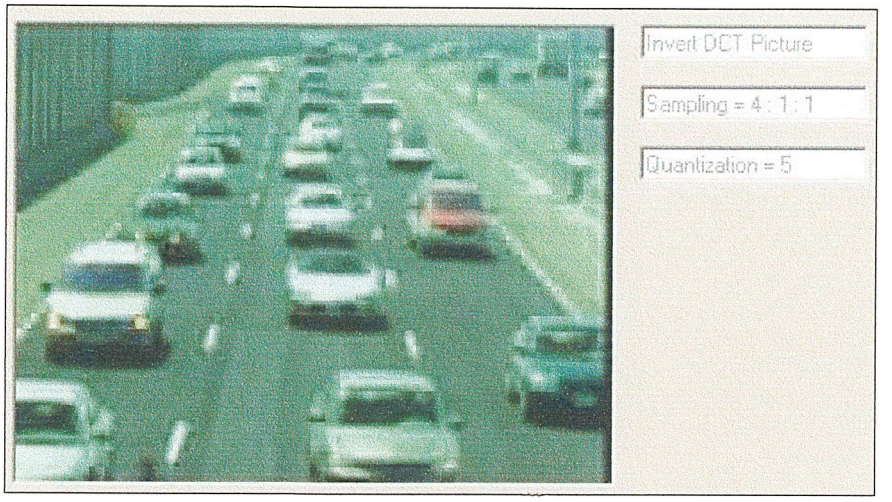


รูปที่ 4.6 แสดงถึงการถอดเอาเฉพาะสัญญาณ R ที่แยกออกมาจากภาพต้นฉบับ

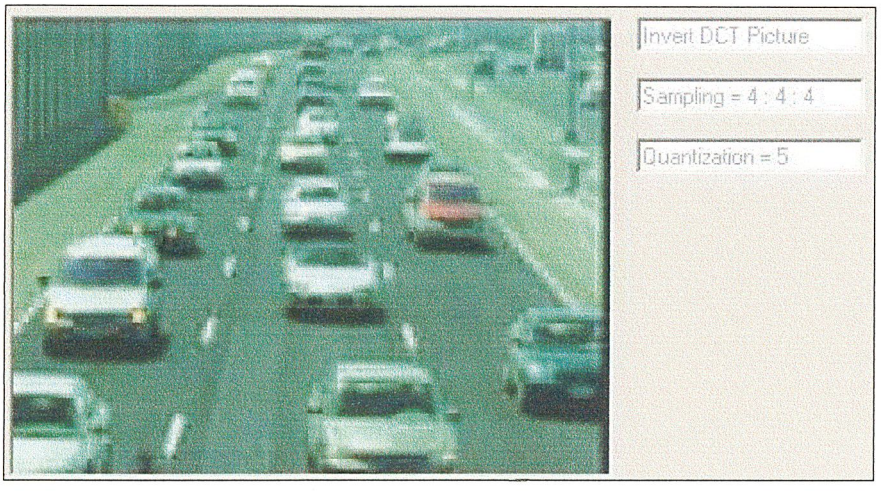
รูปที่ 4.7 แสดงถึงการถอดเอาเฉพาะสัญญาณ G ที่แยกออกมาจากภาพต้นฉบับ



รูปที่ 4.8 แสดงถึงการถอดเอาเฉพาะสัญญาณ B ที่แยกออกมาจากภาพต้นฉบับ

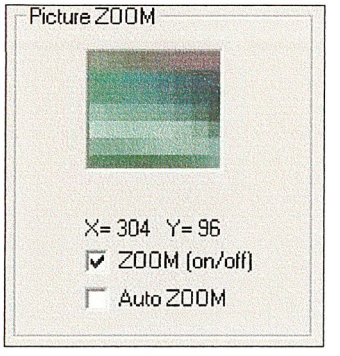


รูปที่ 4.9 แสดงถึงรูปที่ผ่านการ Sampling 4:1:1 และ Quantization = 5



รูปที่ 4.10 แสดงถึงรูปที่ผ่านการ Sampling 4:4:4 และ Quantization = 5

รูปที่ 4.11 แสดงถึงรูปที่ผ่านการ Sampling 8:1:1 และ Quantization = 5



รูปที่ 4.12 แสดงการ Zoom ของ Pixel ในรูปที่ตำแหน่ง X=304 และ Y=96

242	242	242	243	244	245	246	244
241	241	241	242	244	245	245	243
241	240	240	239	242	245	245	241
240	240	240	237	242	245	244	239
240	239	240	238	241	244	241	239
241	241	240	242	242	242	241	241
239	239	240	240	240	238	238	238
221	221	223	223	222	213	198	182

รูปที่ 4.13 แสดงค่าของสัญญาณ Y

1902	6	-9	0	0	0	0	0
47	-20	0	0	0	0	0	0
-33	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
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

รูปที่ 4.14 แสดงค่าที่ผ่านการ DCT และ Quantization = 5

235	237	239	241	243	243	243	243
238	239	241	243	244	245	244	244
241	243	244	246	246	246	245	245
243	244	245	246	246	245	244	243
242	243	243	243	243	241	239	238
238	238	238	238	236	234	232	230
232	232	232	231	229	226	224	222
229	229	228	227	225	222	219	217

รูปที่ 4.15 แสดงการผ่าน Inverse DCT ของสัญญาณ Y ข้างต้น

-8	-9	-9	-8	-7	-6	-4	-6
-7	-8	-8	-8	-6	-5	-5	-6
-9	-8	-8	-9	-7	-5	-6	-9
-11	-10	-9	-12	-9	-7	-8	-12
-10	-10	-9	-10	-8	-7	-9	-11
-8	-9	-8	-7	-8	-8	-9	-11
-9	-8	-8	-8	-8	-10	-11	-13
-12	-12	-11	-11	-11	-14	-15	-14

รูปที่ 4.16 แสดงค่าของ Cb ก่อนทำการ Sampling ที่ได้จากภาพ

-4	-4	-1	-1	3	3	5	5
-4	-4	-1	-1	3	3	5	5
-7	-7	-3	-3	1	1	3	3
-7	-7	-3	-3	1	1	3	3
-11	-11	-7	-7	-3	-3	0	0
-11	-11	-7	-7	-3	-3	0	0
-15	-15	-11	-11	-6	-6	-3	-3
-15	-15	-11	-11	-6	-6	-3	-3

รูปที่ 4.17 แสดงค่าของ Cb หลังทำการ Sampling ในอัตรา 4:1:1 ที่ได้จากภาพ

-7	-7	-7	-7	-8	-8	-8	-8
-7	-7	-7	-7	-8	-8	-8	-8
-7	-7	-7	-7	-8	-8	-8	-8
-7	-7	-7	-7	-8	-8	-8	-8
-9	-9	-9	-9	-10	-10	-10	-10
-9	-9	-9	-9	-10	-10	-10	-10
-9	-9	-9	-9	-10	-10	-10	-10
-9	-9	-9	-9	-10	-10	-10	-10

รูปที่ 4.18 แสดงค่าของ Cb หลังทำการ Sampling ในอัตรา 8:1:1 ที่ได้จากภาพ

-14	-14	-14	-12	-12	-11	-11	-12
-16	-16	-16	-15	-13	-11	-11	-14
-16	-17	-17	-18	-15	-11	-11	-15
-16	-16	-17	-20	-15	-11	-12	-16
-16	-17	-17	-19	-15	-13	-15	-18
-15	-16	-17	-14	-14	-15	-16	-14
-18	-18	-18	-18	-18	-19	-18	-18
-23	-24	-24	-24	-23	-21	-19	-19

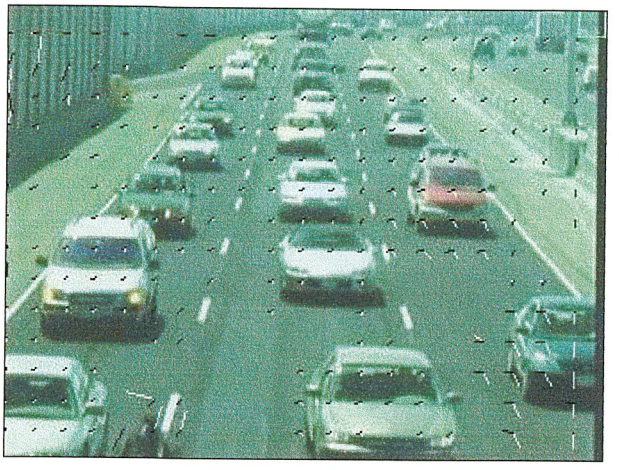
รูปที่ 4.19 แสดงค่าของ Cr ก่อนทำการ Sampling ที่ได้จากภาพ

-6	-6	-5	-5	-4	-4	-4	-4
-6	-6	-5	-5	-4	-4	-4	-4
-4	-4	-3	-3	-2	-2	-3	-3
-4	-4	-3	-3	-2	-2	-3	-3
-2	-2	-2	-2	-2	-2	-3	-3
-2	-2	-2	-2	-2	-2	-3	-3
-2	-2	-1	-1	-1	-1	-3	-3
-2	-2	-1	-1	-1	-1	-3	-3

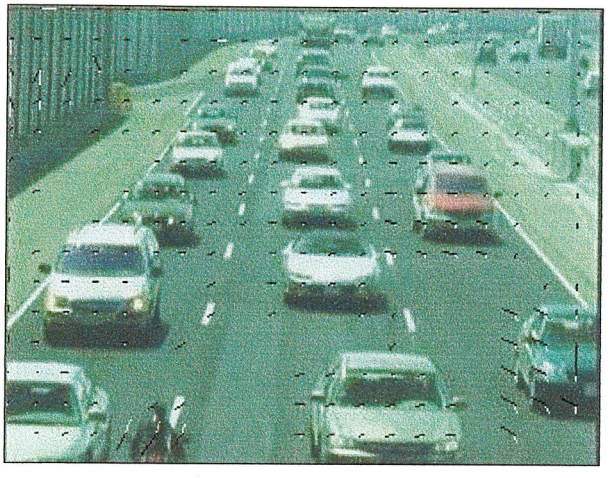
รูปที่ 4.20 แสดงค่าของ Cr หลังทำการ Sampling ในอัตรา 4:1:1 ที่ได้จากภาพ

-17	-17	-17	-17	-18	-18	-18	-18
-17	-17	-17	-17	-18	-18	-18	-18
-17	-17	-17	-17	-18	-18	-18	-18
-17	-17	-17	-17	-18	-18	-18	-18
-19	-19	-19	-19	-20	-20	-20	-20
-19	-19	-19	-19	-20	-20	-20	-20
-19	-19	-19	-19	-20	-20	-20	-20
-19	-19	-19	-19	-20	-20	-20	-20

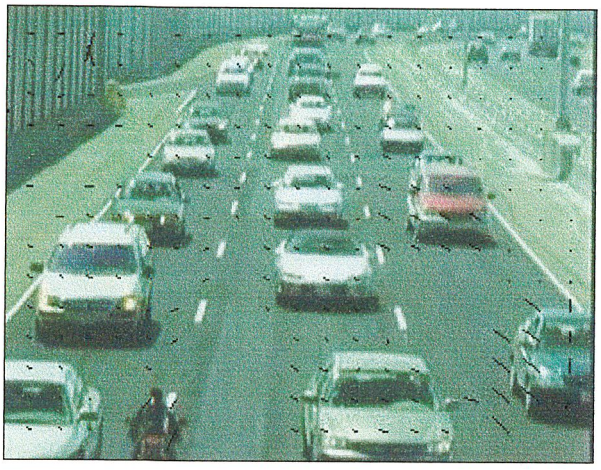
รูปที่ 4.21 แสดงค่าของ Cr หลังทำการ Sampling ในอัตรา 8:1:1 ที่ได้จากภาพ



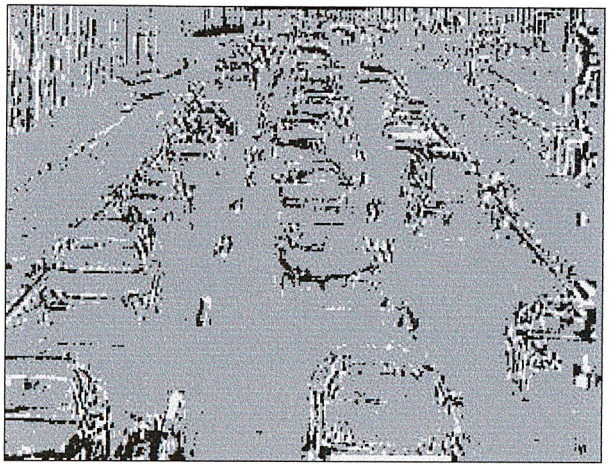
รูปที่ 4.22 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม B1 ทั้ง Forward และ Backward



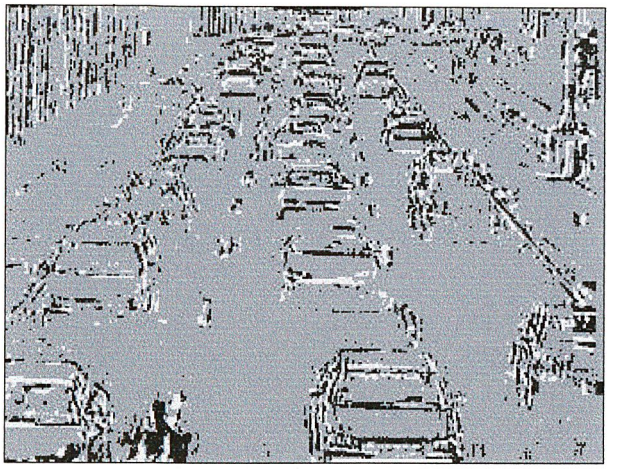
รูปที่ 4.23 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม B2 ทั้ง Forward และ Backward



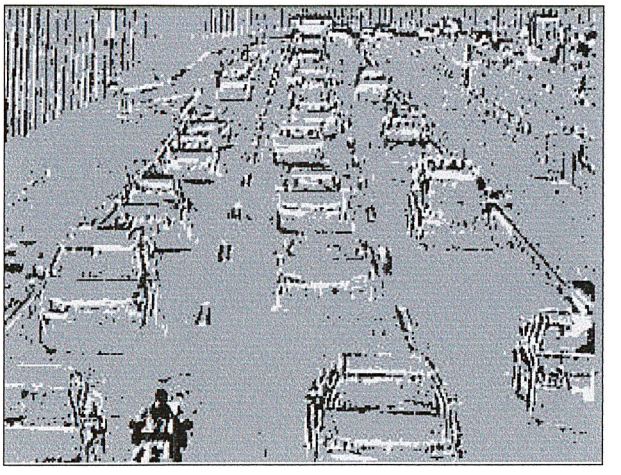
รูปที่ 4.24 แสดงเวกเตอร์การเคลื่อนที่ของเฟรม P ที่มีแต่เฉพาะ Forward เท่านั้น



รูปที่ 4.25 แสดงถึง Motion Compensated ของเฟรม B1



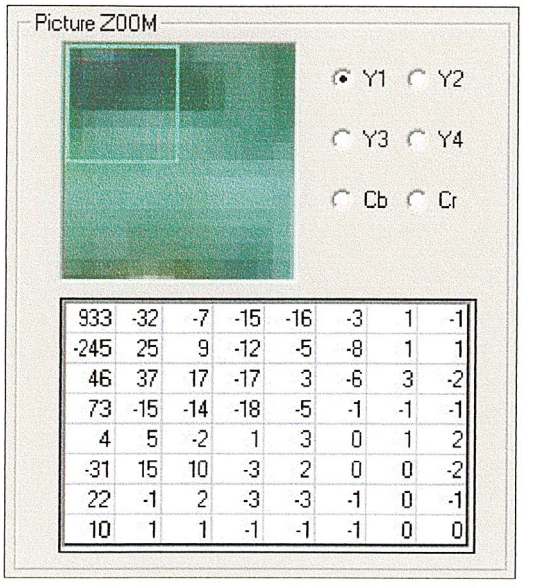
รูปที่ 4.26 แสดงถึง Motion Compensated ของเฟรม B2



รูปที่ 4.27 แสดงถึง Motion Compensated ของเฟรม P

125	125	127	127	127	123	127	127	125	125	130	130	133	127	123	115
126	126	127	127	127	124	124	124	123	130	130	130	128	119	116	112
126	126	127	128	127	125	124	122	124	130	125	123	119	114	115	115
125	127	127	127	127	127	124	123	126	126	123	121	120	119	122	122
126	127	126	127	127	126	125	125	126	125	124	122	123	125	124	124
127	126	127	127	127	127	127	127	126	127	126	124	125	127	126	124
126	126	126	127	126	127	127	127	126	127	127	125	125	127	126	125
126	126	126	128	126	128	127	127	126	125	125	124	125	127	126	125
126	126	127	129	127	127	127	127	125	124	124	123	125	127	127	124
127	125	125	128	127	127	129	130	126	124	125	121	124	125	126	121
127	123	121	125	126	131	132	130	126	125	127	124	122	120	123	122
124	125	124	125	126	131	133	128	124	127	127	125	122	119	123	124
122	127	128	126	127	130	131	129	133	134	125	121	121	122	124	125
123	127	127	129	127	127	131	154	173	157	127	119	122	126	126	125
125	124	125	129	130	125	145	212	229	193	132	120	126	130	129	126
124	119	120	127	130	125	169	232	232	197	128	121	127	128	130	126

รูปที่ 4.28 แสดงถึงค่าที่ค้นการใน Area Search ที่ Match กันกับ Block ค้นหา



รูปที่ 4.29 แสดงการ Zoom ของ Macro block

บทที่ 5
สรุป

โปรแกรมแสดงการวิเคราะห์การบีบอัดสัญญาณภาพ (Visual MPEG Analysis) นี้จากที่ทำการลดขนาดการบีบอัดตามแบบ JPEG แล้วโดยการกระทำ ต่างๆที่สำคัญคือ การแปลงจากฟอร์แมตภาพของอาร์จีบีเป็นสัญญาณ YCbCr จากนั้นก็ทำการทรานสฟอร์มของดิสครีตโคซายน์ ต่อมาก็ทำการควอนไตเซชัน โดยตารางที่กำหนดขึ้น และสุดท้ายก็เข้ารหัสเอนโทรปี ซึ่งใช้รันเลนจ์และการเข้ารหัสฮัฟแมน ซึ่งทำให้ภาพที่ได้ลดขนาดลงซึ่งเกิดการสูญเสียเกิดขึ้น แต่ยังคงสามารถเห็นเป็นกับภาพเหมือนต้นฉบับมาก ซึ่งเหตุที่ต้องทำ JPEG ก่อนเพราะ ในเฟรมของ MPEG นั้นมีเฟรมหนึ่งที่มีการกระทำเป็นแบบ JPEG คือเฟรม I หรือ Intra Frame

เมื่อทำการบีบอัดแบบ JPEG ไปแล้วก็มาทำต่อในส่วนที่ยากที่สุดของ MPEG นั้นคือการวิเคราะห์การเคลื่อนที่ (Motion estimation) และ การวิเคราะห์ความแตกต่างของสี (Motion compensation) ซึ่งโปรแกรมสามารถที่จะแยกการทำงานทั้งสองส่วนนี้ได้ทำให้สามารถมองเห็นถึงวิธีการทำงานของการลดความซ้ำซ้อนของภาพ โดยโปรแกรมจะแสดงให้เห็นถึงแต่ละขั้นตอนการทำงานทั้งเวกเตอร์ทั้ง การเคลื่อนที่ Forward และ Backward รวมทั้งการทำงานของ Motion compensation ได้

ปัญหาที่เกิดขึ้นในด้านเทคนิค คือ การทำงานนั้นใช้เวลาในการค้นหาและทำการสร้างทั้งเวกเตอร์และความแตกต่างระหว่างภาพ พอสมควรเพราะต้องทำการค้นหาทุกจุดในเฟรมแต่ละอันเปรียบเทียบกัน จึงควรใช้ ตัวประมาณผลกลางที่ทำงานเร็วพอสมควร และหน่วยความจำที่มากพอ เฟรมที่นำมาทำการวิเคราะห์นั้นทำได้แค่ 4 เฟรมเท่านั้น คือ เฟรม I เฟรม B1 เฟรม B2 และเฟรม P เท่านั้น และเฟรมภาพที่นำเข้ามานั้นควรจะมีความแตกต่างกันในเฟรมแต่ละเฟรมพอสมควรจึงจะสามารถเห็นเวกเตอร์การเคลื่อนที่ ถ้าหากความแตกต่างของเฟรมภาพน้อยจะทำให้เห็นเวกเตอร์เป็นจุดมองไม่เห็นเป็นเส้นแสดงการเคลื่อนที่ของการเปลี่ยนแปลงภาพได้ชัดเจน และภาพอินพุทที่นำเข้ามานั้นต้องเป็นฟอร์แมตไฟล์ AVI ที่บันทึกเข้ามาโดยยังไม่มีการลดขนาด และต้องมี ขนาดของภาพเท่ากับ 320 × 240 pixel

แนวทางในการพัฒนาต่อ คือ สามารถทำให้โหลดเฟรมได้มากขึ้นในการวิเคราะห์แต่ละครั้ง เพื่อให้ สามารถเลือกได้หลายเฟรมมากขึ้น และในการพัฒนาต่อสามรถทำให้เฟรมภาพที่นำเข้ามานั้นสามารถที่จะปรับขนาดของภาพได้หรือทำให้สามารถวิเคราะห์กับขนาดของภาพที่มากขึ้นหรือน้อยลงได้ และอาจจะเพิ่มความสามารถของโปรแกรมให้ทำงานกับ ภาพได้หลายขนาดหรืออาจทำให้ใช้ได้กับภาพทุกขนาดโดยจะทำงานได้ตามเฟรมภาพที่รับเข้ามา

บรรณานุกรม

1. Kluwer Academic Publisher "Image And Video compression Standard algorithms and Architecture" , 1995
2. Nelsos M. , "The data compression book" , MAT publishing company, Inc 1991
3. อ.กฤดากร กล่อมการ "การสื่อสารข้อมูล(Data Communications)" สภาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2545
4. K.R. Rao and P.Yip. "Discrete Cosine Transform." Academic Press. 1990.
5. กิตติ ภักดีวัฒนะกุล-จำลอง ครูอุตสาหะ "Visual Basic 6 ฉบับโปรแกรมเมอร์" บริษัท เคทีพี คอมพ์ แอนด์ คอนซัลท์ จำกัด, 2545
6. Yue Chen, Yu Wang, Ying Lu "A New Fast Motion Estimation Algorithm - Literature Survey" , 1998
7. Jan Peter Berns, Tobias G. Noll "A 3.2 TOPS Enhanced Motion Estimation System" Chair of Electrical Engineering and Computer Systems,University of Technology RWTH Aachen, Germany
8. Yung-Kai Lai, Jin Li, and C.-C. Jay Kuo, "Image Enhancement for Low Bit-rate JPEG and MPEG Coding via Postprocessing" ,University of Southern CaliforniaLos Angeles