การศึกษาการถอดรหัสมาตรฐานเอ็มเพ็ค

STUDYING OF MPEG STANDARD DECODING

โดย

นายชาญวิทย์ แสงอรุณบริสุทธิ์

เลขประจำตัว 37014093

อาจารย์ที่ปรึกษา

ดร. สมศักดิ์ ชุ่มช่วย

วัน เดือน ปี 24 ก.ค. 2541

เลขทะเบียน 039142

เลขเรียกหนังสือ T 10381 ช 495ก

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2540

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

039142

ปริญญานิพนธ์ปีการศึกษา 2540

ภาควิชา วิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง : การศึกษาการถอดรหัสมาตรฐานข้อมูลเอ็มเพ็ค

SUBJECT : STUDYING OF MPEG STANDARD DECODING

ผู้จัดทำ

นาย ชาญวิทย์ แสงอรุณบริสุทธิ์ เลขประจำตัว 37014093

ดร. สมศักดิ์ ชุ่มช่วย

อาจารย์ที่ปรึกษา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทคัดย่อ

เมื่อการพัฒนาทาง Digital multimedia มีความก้าวหน้าไปมาก ประกอบกับความต้องการในการใช้งานมัลติมีเดีย ที่มีความสะดวกมากยิ่งขึ้น เป็นสิ่งจูงใจให้ผู้ผลิตทำการพัฒนาผลิตภัณฑ์สู่ตลาดจำนวนมากมาย ในการพัฒนาเหล่านั้นสิ่งที่ถูกพัฒนาอย่างต่อเนื่องก็คือ ระเบียบวิธีการ การประมวลผลรหัสดิจิตอล ซึ่งได้มีผู้คิดค้นวิธีการใหม่ๆจำนวนมาก จึงทำให้เกิดแนวความคิดในการสร้างมาตรฐานเพื่อความเข้ากันได้ในการพัฒนาระบบประมวลผลวีดีโอโดยจัดตั้งกลุ่มที่เรียกว่า MPEG (Moving Picture Expert Group)

โครงงานนี้จะเริ่มด้วยการศึกษาระเบียบวิธีการของมาตรฐาน MPEG-1 ซึ่งเริ่มต้นที่การบีบอัดข้อมูลทั้ง Lossy compression และ Lossless compression การศึกษากระบวนมาตรฐานที่เกี่ยวข้องคือ JPEG ในส่วนที่จะต้องถูกนำไปใช้ในการเข้ารหัสของ MPEG หลังจากนั้นจะเป็นการศึกษากระบวนการวิธีที่จะเข้ารหัสภาพเคลื่อนไหว ที่เกิดจากภาพนิ่งหลายภาพต่อเนื่องกันโดยวิธีต่างๆ เช่น Motion estimation, Motion compensation นำมารวมกับการบีบอัดข้อมูลทั้งสองแบบของภาพนิ่งคือ Lossy และ Lossless ก็จะทำให้ได้ข้อมูลในการส่งที่ลดลงโดยมีการบีบอัดที่สูง และยังคงคุณภาพของภาพตามที่ต้องการ ส่วนสุดท้ายเป็นการศึกษา Syntax ของมาตรฐาน MPEG-1 ประกอบด้วยความหมายของรหัส Binary ใน Bit stream และเขียนโปรแกรม Visual C++ ถอดรหัสจากไฟล์ MPEG ตัวอย่าง แสดงในรูปแบบภาพที่เป็นเฟรมต่อเนื่องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ABSTRACT

In recent year, the digital multimedia has great improvement. The sharply grown market convinces manufacturers to advantage their product for competition. Simultaneously, they develop many new digital data processing format and algorithm. For compatibility, a group named " MPEG " (Moving Picture Expert Group) was founded.

This project offers the studying of MPEG-1 standard which begin by the two basic methods of compression, lossless compression and lossy compression. The other methods that involve video compression, such as JPEG standard. Following by the motion picture processing with two important algorithms are Motion estimation and Motion compensation. The objective is obtaining of decreasing in data amount and bit rate but don't decrease the quality of picture. The project demonstrates results by some picture frames of video sequence from a program, which was written by visual c++.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทที่ 1 บทนำ	1
1.1 บทนำ	1
1.2 วัตถุประสงค์ของโครงงาน	1
1.3 โครงสร้างของโครงงาน	2
1.4 ขั้นตอนการทำงาน	3
1.5 เนื้อหาของวิทยานิพนธ์	3
บทที่ 2 หลักการบีบอัดข้อมูลภาพและมาตรฐาน JPEG	5
2.1 บทนำ	5
2.2 หลักการบีบข้อมูลเบื้องต้น	5
2.3 วิธีการและมาตรฐานการบีบข้อมูลแบบไม่สูญเสีย	9
2.4 หลักการบีบข้อมูลแบบสูญเสีย	13
2.5 มาตรฐาน JPEG	21
บทที่ 3 หลักการบีบอัดข้อมูลวีดีโอและมาตรฐาน MPEG	30
3.1 บทนำ	30
3.2 พื้นฐานการเข้ารหัสสัญญาณวีดีโอ	30
3.3 Motion compensate prediction	33
3.4 อัลกอริธึมสำหรับ Motion estimation	34
3.5 มาตรฐาน MPEG	35
บทที่ 4 รูปแบบรหัสข้อมูล Video ตามมาตรฐาน MPEG-1	53
4.1 บทนำ	53
4.2 Start code	53
4.3 Layer ของรหัสข้อมูล	54

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5 การทดลองและผลการถอดรหัส MPEG 67

5.1 การทดลองการแปลง Discrete Cosine Transform 67

5.2 การทดลองการแปลง Inverse Discrete Cosine Transform 70

5.3 ตัวอย่างการทดสอบข้อมูลจากไฟล์ MPEG- 1 72

5.4 การถอดรหัสไฟล์ MPEG - 1 76

5.5 เฟรมของภาพที่ได้จากการถอดรหัสด้วยโปรแกรมภาษา C 77

5.6 ปัญหาและการแก้ไข 85

บรรณานุกรม 86

ภาคผนวก 87

โปรแกรมสมบูรณ์ที่ใช้ถอดรหัสไฟล์ MPEG-1

ไฟล์ MPEG Decode.c 87

ไฟล์ Dct.h 154

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

หน้า

รูปที่ 2.1 หลักการบีบข้อมูลเบื้องต้น 5

รูปที่ 2.2 สาขาต่างๆ ของการบีบข้อมูล Image , Video และ Audio 7

รูปที่ 2.3 การทำงานร่วมกันของการบีบข้อมูล Lossy และ Lossless 8

รูปที่ 2.4 รูปแบบพื้นฐานของการบีบข้อมูลแบบ Lossless 9

รูปที่ 2.5 ตัวอย่างโครงสร้างของ Huffman code 10

รูปที่ 2.6 DPCM compression method 16

รูปที 2.7 ตัวอย่างการแปลง DCT 18

รูปที่ 2.8 Block diagram of a JPEG encode 22

รูปที่ 2.9 Block diagram of a JPEG decoder 23

รูปที่ 2.10 Huffman coding in baseline JPEG 25

รูปที่ 2.11 conventional and zig-zag ordering in an 8 x 8 matrix 25

รูปที่ 2.12 รายละเอียดการดำเนินตามขั้นตอนในมาตรฐาน JPEG 27

รูปที่ 2.13 Three level hierarchical coder 28

รูปที่ 3.1 Temporal correlation ในภาพที่ต่อเนื่องกัน 31

รูปที่ 3.2 การทำงานโดยทั่วไปของ Video coder และ Video decoder 32

รูปที่ 3.3 Motion-estimation process 33

รูปที่ 3.4 ข้อกำหนดของแมคโครบล็อกในมาตรฐาน MPEG-1 37

รูปที่ 3.5 ความเกี่ยวข้องกันระหว่างภาพแบบ I, P และ B ในอนุกรมวีดีโอ 39

รูปที่ 3.6 Block diagram of an MPEG encoder 40

รูปที่ 3.7 Forward motion compensation 41

รูปที่ 3.8 Bidirectional motion compensation 42

รูปที่ 3.9 Block diagram of an MPEG coder 44

รูปที่ 3.10 Syntax layer in MPEG 1 video coding 46

รูปที่ 3.11 Decision trees for coding macroblocks in I- ,P- and B-pictures 47

รูปที่ 3.12 Block diagram การประมวลผลในระบบ MPEG 49

รูปที่ 3.13 การประมวลผล Intra picture 50

รูปที่ 3.14 การประมวลผล Predicted picture 51

รูปที่ 3.15 การประมวลผล Bidirectionally predicted picture 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 บทนำ

ภาพวีดีโอนับว่ามีบทบาทสำคัญมากในสังคมปัจจุบัน ในการนำข่าวสารความบันเทิงต่างๆ สู่สายตาผู้คนนับล้านทั่วโลก การใช้งานได้แตกแขนงออกไปอย่างกว้างขวาง โดยการอาศัยคุณสมบัติของสัญญาณไฟฟ้าที่ผสมผสานกันทั้งระบบ Analog และ Digital เป็นสื่อของข้อมูล นับตั้งแต่อดีตจนถึงปัจจุบัน การพัฒนาสื่อของข้อมูลนี้เหล่านี้ มีความก้าวหน้าตลอดเวลา เนื่องจากว่าระบบดิจิตอลเป็นระบบที่ยืดหยุ่นในการที่จะประมวลผลข้อมูลตามต้องการได้ง่าย ดังนั้นแนวโน้มของการพัฒนาสื่อของข้อมูลจึงมุ่งพัฒนาในด้านดิจิตอลเป็นหลัก ถึงแม้ว่าจะต้องมีการเพิ่มวงจร ADC และ DAC ก็ตาม จะพบว่าประโยชน์ที่ได้รับคุ้มค่ากับส่วนที่เพิ่มขึ้น ดังนั้นเราจะพบว่าช่วงการประมวลผลและเก็บรักษาข้อมูลจะถูกพัฒนาให้เป็นระบบข้อมูลดิจิตอลทั้งหมด รวมทั้งสื่อต่างๆมากมายที่ถูกนำมาใช้ในระบบดิจิตอลได้มีการพัฒนาทั้งในด้านขนาด ความจุและคุณภาพขึ้นเรื่อย อย่างเช่นการพัฒนา จาก Laser disk เป็น CD-ROM และจะเป็น DVD ในอนาคตต่อไป การพัฒนาจากขนาดของ Diskขนาดใหญ่จนเหลือขนาดมาตรฐานเส้นผ่านศูนย์กลาง 12 เซนติเมตร และจะมีความจุสูงสุดถึง 17 Gbyte การพัฒนาในด้านคุณภาพของภาพและเสียงก็ยิ่งมากขึ้นซึ่งเป็นผลมาจากความจุข้อมูลที่สูงขึ้น และจะส่งผลสืบเนื่องให้ระบบต้องการ การประมวลผลที่รวดเร็วและมีประสิทธิภาพ ดังนั้นเทคโนโลยีที่ถูกพัฒนาควบคู่ไปกับการพัฒนาฮาร์ดแวร์ ก็คือการจัดการกับข้อมูล ซึ่งรวมไปถึง การบีบอัดข้อมูลให้มีขนาดเล็กลง การเข้ารหัส การถอดรหัสจะทำให้เราได้ระบบดิจิตอลวีดีโอ ที่สมบูรณ์แบบทั้งคุณภาพ ความสะดวกและความประหยัด

1.2 วัตถุประสงค์ของโครงงาน

โครงงานนี้เป็นโครงงานการศึกษาระบบประมวลผลภาพเคลื่อนไหว ตามมาตรฐาน ที่เรียกว่า MPEG ซึ่งเป็นกลุ่มที่จัดตั้งขึ้นเพื่อมุ่งสร้างระบบประมวลผลวีดีโอมัลติมีเดียให้เป็นมาตรฐานเดียวกัน โดยอาศัยโปรแกรมภาษา C ทำการถอดรหัสข้อมูลไฟล์ตัวอย่างตามมาตรฐาน MPEG 1 โดยมีเป้าหมายที่สำคัญคือ

1. ศึกษารูปแบบวิธีการแสดงผลภาพโดยทั่วไป
2. ศึกษาวิธีการบีบข้อมูลภาพ
3. ศึกษาวิธีการบีบข้อมูลภาพเคลื่อนไหว
4. ศึกษาการเข้ารหัส ถอดรหัส Syntax และ รหัสชนิดต่างๆ ของมาตรฐาน MPEG
5. เขียนโปรแกรมถอดรหัสภาพวีดีโอจากไฟล์ MPGE 1 แสดงออกเป็นเฟรมของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 โครงสร้างของโครงงาน

1.3.1 ภาคการศึกษาที่ 1

ศึกษาวิธีการแสดงผลภาพ การบีบข้อมูลชนิดต่างๆ ทฤษฎีทั่วไปของการบีบข้อมูลเพื่อให้ได้อัตราการบีบอัดที่ดีและสูญเสียคุณภาพไปน้อยที่สุด การบีบข้อมูลแบบสูญเสียและไม่สูญเสีย การเข้ารหัสแบบฮัฟแมนซึ่งจะทำให้ได้การบีบอัดข้อมูลที่ดีโดยอาศัยความน่าจะเป็น ความจำเป็นที่จะต้องใช้ทั้งระบบ lossy compression และ lossless compression ร่วมกันเนื่องมาจากว่าสื่อข้อมูลไม่มีความสามารถพอ ที่จะส่งข้อมูลได้เร็วเพียงพอต่อการประมวลผลได้ทันเวลา การประมวลผลภาพเคลื่อนไหวที่ใช้ความสัมพันธ์ของช่องว่างของภาพและช่องว่างของเวลาในการเข้ารหัส การทำนายแบบเปรียบเทียบ อาศัยการส่งข้อมูลเป็นเวคเตอร์ที่เรียกว่า motion vector ที่มีขนาดเล็กเป็นตัวชี้ตำแหน่งของข้อมูลขนาดใหญ่ที่ต้องการ ซึ่งถูกหาโดยวิธีการ Search เมื่อต้องทำการเข้ารหัสแบบ Search เราก็จะทำการแบ่งภาพแต่ละภาพออกเป็นชนิดต่างๆตามหน้าที่ที่ต้องใช้ในกระบวนการถอดรหัส การผสานกันในแต่ละส่วนดังกล่าวนั้นทำให้จำนวนข้อมูลที่ส่งจะน้อยลงแต่หลังจากการ Decode แล้วสามารถทำให้ได้ข้อมูลที่ใกล้เคียงกับข้อมูลเดิมมาก

ศึกษามาตรฐาน การเข้ารหัสวีดีโอ MPEG-1 ประกอบไปด้วย Syntax การทำงาน การจัดวางรูปแบบของ Bit stream การแบ่งให้ Bit stream ถูกแยกเป็น Layer เพื่อให้สอดคล้องกับหน้าที่ของข้อมูลที่บรรจุใน Layer นั้นๆ ภายในแต่ละ Layer มีรหัสที่ทำหน้าที่ต่างๆกันจำนวนมากนอกเหนือจากตัวข้อมูลจริงที่เป็นข้อมูลของภาพวีดีโอ เพื่อให้การทำงานเป็นไปอย่างสอดคล้องกัน บิตที่เพิ่มเข้าไปจะเป็นตัวควบคุมสิ่งต่างๆที่จะเกิดขึ้นกับตัวข้อมูลจริงอีกทีหนึ่ง เช่น Bit rate picture rate เป็นต้น และยังถือเป็นการติดต่อกันระหว่าง Encoder และ Decoder อีกด้วย เนื่องจากว่าข้อมูล Bit stream เป็นแบบ serial และมีขนาดไม่แน่นอนทำให้การตรวจสอบความผิดพลาดของระบบการประมวลผลเป็นไปได้ยาก ดังนั้นการใช้ซอฟแวร์ที่มีระบบการ Debug ที่ดี มาทำการทดสอบกับ Bit stream ตัวอย่างจะเป็นสิ่งที่ให้ประโยชน์อย่างมาก ถึงแม้อัลกอริธึมจะต้องมีจำนวนมากกว่าที่จะสามารถถอดรหัสข้อมูลได้ทั้งหมด แต่เป็นวิธีที่เหมาะที่สุดในการทดสอบข้อมูลชนิดนี้ ดังนั้นการศึกษาในหัวข้อนี้ จึงกระทำไปพร้อมการทดลองถอดรหัสพื้นฐานของ MPEG-1 ด้วยโปรแกรมคอมพิวเตอร์ ซึ่งจะทำให้เข้าใจการทำงานและหน้าที่ของรหัสข้อมูลทั้งหมดใน Bit stream ว่ามีการจัดเรียงอย่างไรในแต่ละ Layer และทำหน้าที่อะไรบ้าง เพื่อให้สามารถที่จะออกแบบระบบในขั้นที่สูงต่อไป

1.3.2 ภาคการศึกษาที่ 2

ศึกษาโครงสร้างการทำงานของภาษา C ซึ่งเป็นภาษาที่จะใช้ถอดรหัส MPEG และเก็บแต่ละเฟรมในรูปของไฟล์ภาพ แล้วทำการเขียนโปรแกรมภาษา C โดยเริ่มจาก Procedure เริ่มต้นคือส่วนของข้อมูลตามมาตรฐาน ISO/IEC 11172-1 system ซึ่งเป็นข้อมูลในชั้นระดับไฟล์ ซึ่งจะเป็นส่วนที่เป็นเหมือนตัวแยกข้อมูลชนิดต่างๆคือประกอบไปด้วย ข้อมูลภาพ เสียงและอื่นๆแล้วแต่จะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กำหนดขึ้น ที่แทรกกันอยู่ ให้ตัวประมวลผลแต่ละตัวนำไปประมวลผลได้อย่างถูกต้อง ในส่วนนี้จะทำการแยกเฉพาะข้อมูลที่เป็น video ซึ่งเป็นส่วนที่ทำการศึกษามาทำการประมวลผล ข้อมูลในส่วน video นี้จะเป็นมาตรฐานตาม ISO/IEC 11172-2 video ซึ่งประกอบไปด้วยชั้น Layer ต่างๆที่บรรจุข้อมูลภาพและข้อมูลอื่นเช่นการทำ compensation estimation หรือ การแปลง DCT เป็นต้น

1.4 ขั้นตอนการทำงาน

ภาคการศึกษาที่	การทำงาน	ระยะเวลา
1	- ศึกษาวิธีการบีบข้อมูลภาพ - ศึกษาวิธีการบีบข้อมูลภาพเคลื่อนไหว - ศึกษาระบบมาตรฐาน JPEG - ศึกษาระบบมาตรฐาน MPEG	2 เดือน
	- ศึกษาข้อมูลวีดีโอตามมาตรฐาน MPEG -1 - ศึกษา Layer ต่างๆของข้อมูลอนุกรมวีดีโอ - ทดสอบกับรหัสข้อมูลจริงโดยเขียนโปรแกรมจำลอง การทำงาน	2 เดือน
2	- ศึกษาโครงสร้างการทำงานของภาษา C -เขียนโปรแกรมถอดรหัสไฟล์MPEG ตัวอย่าง - เก็บตัวอย่างเฟรมของภาพที่ต่อเนื่องกัน อธิบายการทำงานของโปรแกรมในส่วนทำงานดังกล่าว	6 เดือน

ตารางที่ 1.1 ขั้นตอนการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 เนื้อหาของวิทยานิพนธ์

ในวิทยานิพนธ์นี้จะประกอบไปด้วยการอธิบายถึงทฤษฎีของการบีบอัดข้อมูลชนิดต่างๆ ซึ่งเป็นพื้นฐานของการประมวลผลภาพวีดีโอ ระบบมาตรฐาน JPEG และ MPEG พร้อมรหัสมาตรฐานที่ MPEG กำหนดเป็น Layer ต่างๆเพื่อให้ทุกระบบที่ผลิตออกมามีความเข้ากันได้ การทดลองเกี่ยวกับ DCT และ การทดสอบโปรแกรมการถอดรหัสข้อมูล MPEG แสดงเป็นเฟรมของภาพต่อเนื่อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

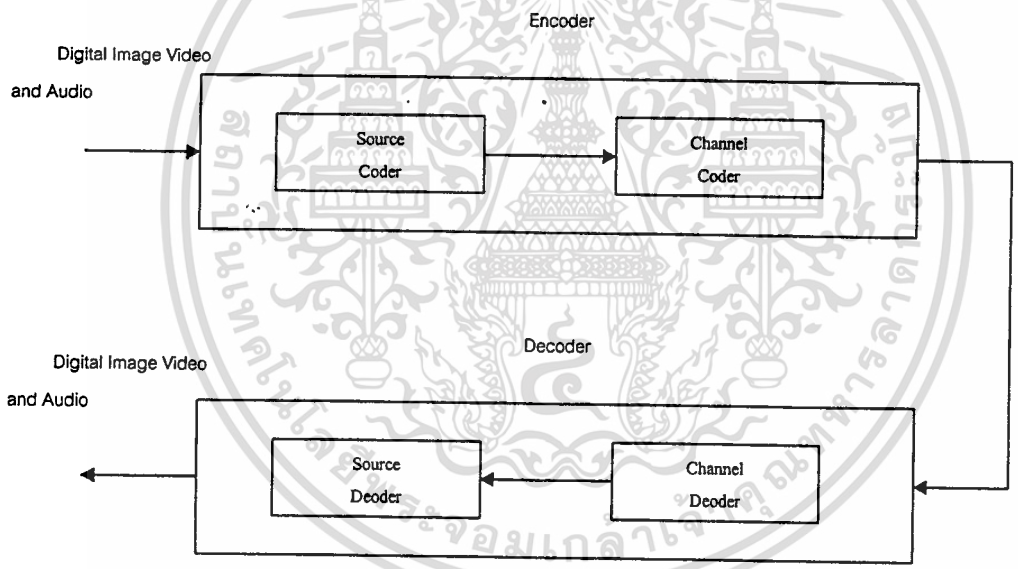
บทที่ 2

หลักการบีบอัดข้อมูลภาพและมาตรฐาน JPEG

2.1 บทนำ

การบีบข้อมูลคือ การทำให้ข้อมูลดิจิตอลมีขนาดเล็กเพื่อให้สามารถจัดการกับข้อมูลได้ง่ายและรวดเร็ว ในงานที่ใช้ทางด้านมัลติมีเดียนั้นส่วนใหญ่แทบจะเป็นไปไม่ได้ ที่จะสามารถทำงานได้ถ้าไม่มีการบีบข้อมูล ปัจจุบันการมีการพัฒนาทางด้านสถาปัตยกรรมการประมวลผลของภาพนิ่ง ภาพเคลื่อนไหว และเสียง โดยจัดการกับข้อมูลดิจิตอลเพื่อลดขนาดของข้อมูลโดยวิธีการบีบอัดข้อมูล ทำให้มีการพัฒนาระบบมัลติมีเดียจำนวนมากเพื่อตอบสนองตลาดผู้บริโภคที่มีขนาดใหญ่ ความแพร่หลายทางด้านการพัฒนาข้อมูลมัลติมีเดียดังกล่าว นำมาซึ่งมาตรฐานและวิธีการเข้ารหัสข้อมูล การเก็บข้อมูล และการสื่อสารข้อมูลให้สามารถมีความ เข้ากันได้ในทุกผลิตภัณฑ์ของผู้ผลิต

2.2 หลักการบีบข้อมูลเบื้องต้น



รูปที่ 2.1 กระบวนการบีบข้อมูลเบื้องต้น

สัญญาณที่ยอมให้มีการบีบข้อมูลด้วยเงื่อนไขดังนี้

1. เมื่อพิจารณาสัญญาณส่วนเกิน (Redundancy) ในทางสถิติจะสามารถ ทำได้ดังนี้คือ

- ข้อมูลเฟรมเดี่ยวในแต่ละเฟรม เมื่อเปรียบเทียบกันจะมีส่วนที่ซ้ำกับเฟรมใกล้ๆ เรียกว่า Spatial correlation

- สำหรับข้อมูลที่มีหลายข้อมูล (เช่นดาวเทียม) ต้องมีการตรวจสอบท่ามกลางข้อมูลเหล่านั้นการเปรียบเทียบนี้เรียกว่า Spectral correlation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สำหรับข้อมูลที่เปลี่ยนแปลงเสมอเช่น Video จะเปรียบเทียบตัวอย่างที่เวลาต่างๆ กัน เรียกการเปรียบเทียบนี้ว่า Temporal correlation

2. บางข้อมูล สัญญาณต้องพิจารณา นอกเหนือจากจุดสิ่งที่มองเห็นและเข้าใจได้

3. บางข้อมูล มีแนวโน้มที่มีคุณลักษณะขั้นสูงที่เป็นส่วนเกินทั้ง ในด้านช่องว่าง (Space) และ เวลา (Time)

ข้อจำกัดที่เป็นปัญหาในระบบการบีบข้อมูล

1. ในการถอดรหัสข้อมูลจะถูกบังคับในเรื่องระดับคุณภาพของสัญญาณ

2. ความซับซ้อนของการถอดรหัสที่ถูกบังคับนั้นส่งผลถึงทั้งการเข้ารหัสและการถอดรหัส

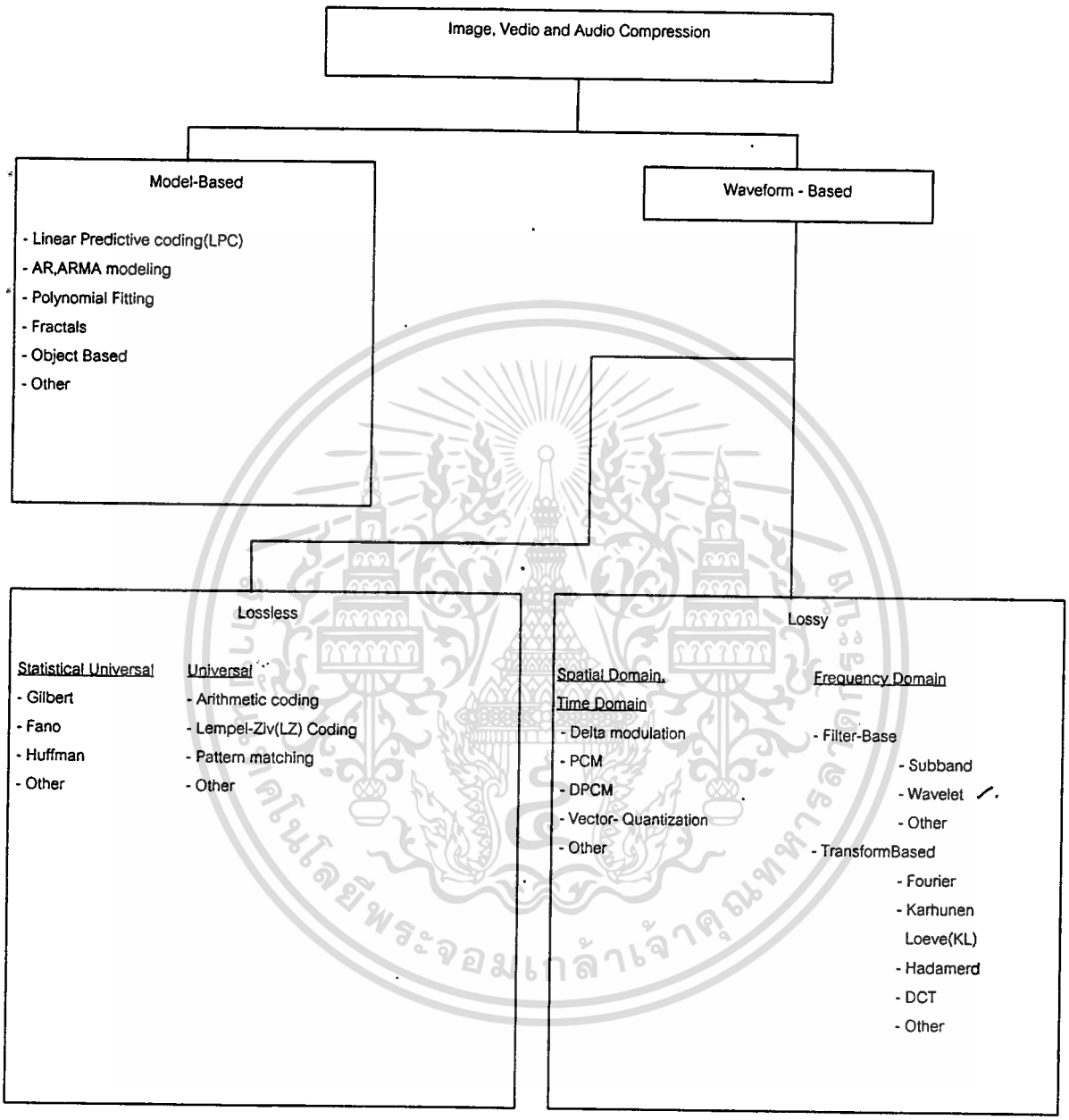
3. การเลื่อนของเวลาในการสื่อสาร (Delay) เป็นการบังคับระยะเวลาตั้งแต่การเข้ารหัสจนถึงถอดรหัส

Application	Data Rate	
	Uncompressed	Compressed
Voice 8 ksamples/s , 8 bits/sample	64 kbps	2 - 4 kbps
Audio conference 8 ksamples/s , 8 bits/sample	64 kbps	16 - 64 kbps
Video conference framesize 352 x 240 , 8 bits/pixel	30.41 Mbps	64 - 768 kbps
Digital audio (stereo) 44.1 ksamples/s , 16 bits/sample	1.5 Mbps	1.28 - 1.5 Mbps
Video file transfer (15 fps) framesize 352 x 240 , 8 bits/pixel	30.41 Mbps	384 kbps
Digital video on CD-ROM (30 fps) framesize 352 x 240 , 8 bits/pixel	60.83 Mbps	1.5 - 4 Mbps
Broadcast video (30 fps) framesize 720 x 48 , 8 bits/pixel	248.83 Mbps	3 - 8 Mbps
HDTV (59.94 fps) framesize 720 x 48 , 8 bits/pixel	1.33 Gbps	20 Mbps

ตารางที่ 2.1 การใช้งานการบีบข้อมูล image video และ audio

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิธีการต่างๆของการบีบข้อมูล



รูปที่ 2.2 สาขาต่างของการบีบข้อมูล Image , Video และ Audio

2.2.1 Lossless compression

การบีบ ข้อมูลแบบสูญเสียเป็นการบีบอัดที่เมื่อทำการถอดรหัส สร้างขึ้นมาใหม่แล้วไม่มีการสูญเสียส่วนใดๆของข้อมูลเดิม ประสิทธิภาพของการบีบข้อมูล (Coding Efficiency) หมายถึงการวัดอัตราบิตต่อนาทีของการทำงาน ซึ่งจะถูกจำกัดด้วยปริมาณของข้อมูลจากแหล่งข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ความซับซ้อนของการบีบข้อมูล (Coding Complexity) เป็นสิ่งที่มีเหตุผลจำเป็นของการเข้ารหัสและถอดรหัสเพื่อการบีบอัดข้อมูลที่มีประสิทธิภาพสูงสุดซึ่งอาจจะวัดจากความต้องการใช้หน่วยความจำหรือจำนวนคำสั่งของหน่วยคำนวณทางคณิตศาสตร์ (MOPS ; Million Operation Per Second)

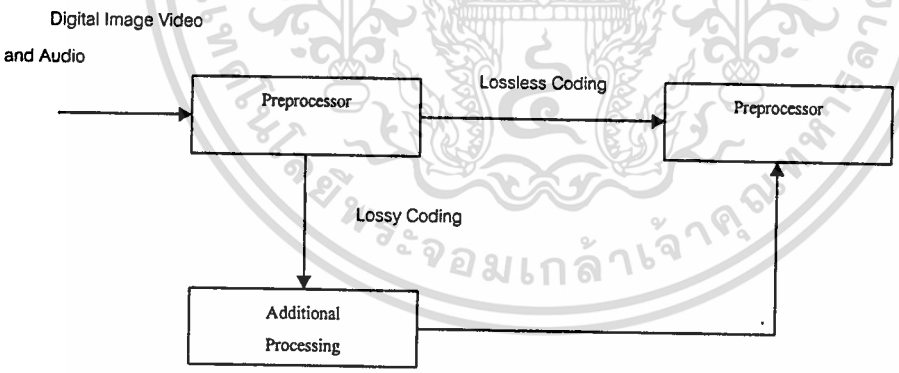
2.2.2 Lossy compression

โดยทั่วไปการทำงานกับข้อมูลภาพ หรือวีดีโอ ไม่จำเป็นจะต้องมีการถอดรหัสข้อมูลให้ได้เหมือนเดิมทั้งหมด การสูญเสียข้อมูลไปบางส่วนเป็นสิ่งที่ยอมรับได้ เราเรียกกระบวนการบีบอัดข้อมูลที่มีการสูญเสียข้อมูลเดิมไปบ้างนี้ว่า การบีบข้อมูลแบบสูญเสีย (Lossy compression) คุณภาพของสัญญาณ (Signal Quality) โดยส่วนใหญ่แล้วจะหมายถึงคุณสมบัติของสัญญาณที่เอาท์พุท

สิ่งหนึ่งที่ถูกอ้างอิงในการวัดคุณภาพของสัญญาณคือ signal to noise ratio (SNR) โดยที่

$$SNR = 10 \cdot \log \frac{encoder\_input\_signal\_energy}{noise\_signal\_energy}$$

เพื่อที่จะทำให้ได้การเข้ารหัสถอดรหัสที่มีประสิทธิภาพสูงจะต้อง อาศัยการทำงานการบีบข้อมูล ทั้งสองแบบโดยสามารถเขียนแผนผังการทำงานในรูปแบบทั่วไปได้ดังนี้

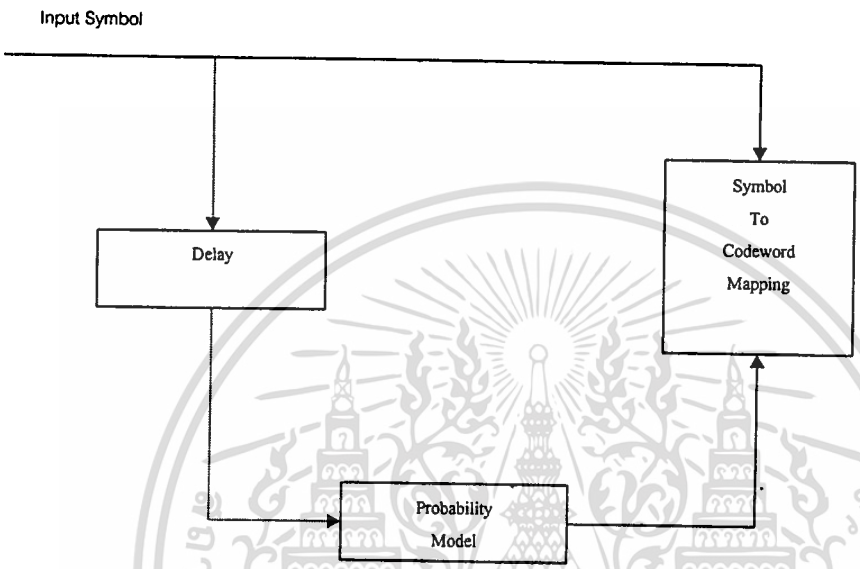


รูปที่ 2.3 การทำงานร่วมกันของการบีบข้อมูลแบบ Lossy และ Lossless

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3 วิธีการและมาตรฐานการบีบข้อมูลแบบไม่สูญเสีย

หลักการบีบอัดข้อมูลแบบไม่สูญเสียคือการที่รับ Symbol เข้ามาแล้วทำการประมาณ (Estimate) การกระจายความน่าจะเป็น (Probability distribution) ซึ่งจะใช้ในการเปลี่ยน Symbol ให้เป็นCodewords ถ้าหากความน่าจะเป็นมาก Codewordsจะสั้น ถ้าหากความน่าจะเป็นน้อย Codewords จะยาว



รูปที่ 2.4 รูปแบบพื้นฐานของการบีบข้อมูลแบบ Lossless

Probability Model จะทำงานโดยอาศัยทั้งข้อมูลที่เข้ามาและการสมมุติฐาน ซึ่งการ Decode ต้องสร้างโมเดลเช่นเดียวกันนี้ด้วย การที่ต้องมีการดีเลย์ก่อนเข้า Probability Model เนื่องจากการประมาณที่เปลี่ยนแปลงของข้อมูล (Dynamically estimation) ถ้าหากว่า model ได้จาก priori assumption แล้วอาจไม่ต้องใช้ Delay block หรือบางทีอาจไม่ต้องใช้ Probability Model เลยก็ได้ Probability Model ไม่ต้องการความแม่นยำมากนักแต่ว่าความแม่นยำจะทำให้ได้การบีบอัดที่ดีกว่า

2.3.1 เทคนิคการทำงานของการบีบอัดข้อมูล

ในการประมวลผลข้อมูลของ Lossless compression จะถูกกำหนดให้ทำงานเป็น 8 , 16 , 32 หรือ 64 บิต แต่ถ้าหากข้อมูลของภาพมี 12 บิตต่อพิกเซล เราจะต้องทำการลดขนาดลงโดยยังคงความสามารถในการประมวลผลให้ได่อัตราส่วนการบีบอัดตามต้องการ . นอกจากนั้นแทนที่เราจะประมวลผลทีละพิกเซล เราจะต้องทำการลดขนาดลงโดยังคงความสามารถในการประมวลผลให้ได้อัตราส่วนการบีบอัดตามต้องการ นอกจากนั้นแทนที่เราจะประมวลผลทีละพิกเซล เราจะทำการจัดข้อมูลให้เป็นกลุ่มของพิกเซลซึ่งทำให้อัตราการบีบอัดสูงขึ้นแต่ก็ทำให้การทำงานซับซ้อนขึ้นเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลภาพนั้นพิกเซลที่อยู่ใกล้เคียงกันมีความสัมพันธ์กันถ้าหากเรานำผลต่างมาใช้หาการกระจายความน่าจะเป็นซึ่งจะได้กราฟที่เป็นเส้นโค้งสมมาตรขึ้น ก็จะทำให้การบีบอัดข้อมูลดีขึ้น

การเข้ารหัสแบบฮัฟแมน (Huffman Encoding)

ในปี ค.ศ. 1952 ฮัฟแมนได้พัฒนาการบีบอัดข้อมูลแบบไม่สูญเสีย ได้เป็นกระบวนการที่รวมเอา Modeling และ Symbol to codeword mapping เข้าด้วยกัน ข้อมูลขาเข้าจะถูกแบ่งแยกเป็นอนุกรมของ Symbol เพื่อสะดวกในการ Modeling

ฮัฟแมนโค๊ด จะประกอบไปด้วยการปฏิบัติการที่ค่อยๆเป็นไปทีละขั้นดังนี้

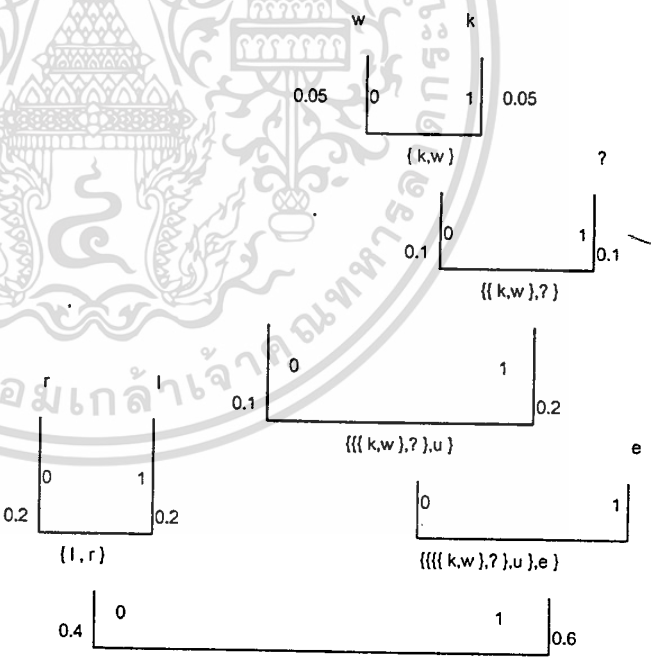
1. เรียง Symbol ตามความน่าจะเป็นของมัน

โครงสร้างของฮัฟแมนโค๊ด สิ่งที่ต้องรู้อันดับแรกคือความถี่ของการปรากฏของ Symbol ซึ่งจะประมาณได้จากข้อมูลทดลองชุดหนึ่ง ซึ่งเป็นตัวแทนของข้อมูลที่ถูกบีบอัดในแบบ Lossless

2. ดำเนินการกับ 2 Symbol ซึ่งมีความน่าจะเป็นน้อยที่สุด โดยใช้ Contraction process

3. ทำซ้ำขั้นตอนข้างต้นจนได้เซตที่เอาท์พุทมีเพียง 1 สมาชิก

Symbol	Probability	Codeword
k	0.05	10101
l	0.2	01
u	0.1	100
w	0.05	10100
e	0.3	11
r	0.2	00
?	0.1	1011



รูปที่ 2.5 ตัวอย่างโครงสร้างของ Huffman code

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณสมบัติของฮัฟแมนโค๊ด

กำหนดให้เอนโทรปีของ S คือ

$$H(s) = \eta = \sum_i p_i \log_2 \frac{1}{p_i}$$

โดยที่ค่าของเอนโทรปีจะแสดงถึงขอบเขตของความยาวเฉลี่ยของ Code word โดยที่เราจะพอใจกับค่าเอนโทรปีเมื่อ $\eta \le l_{avg} \le \eta + 1$ โดยถือว่ามันได้เข้าใกล้ภาวะที่ดีที่สุด

ในการ Decode จะต้องใช้ Lookuped table ซึ่งการที่ความยาวของ Code word ไม่ถูกจำกัดจะทำให้ใช้หน่วยความจำ จำนวนมากในการเก็บตารางดังกล่าวดังนั้นจึงได้มีการพัฒนาฮัฟแมนโค๊ดชนิดต่างๆขึ้น คือ

1. Huffman Codes with Constrained Length
2. Constrained Length Huffman Codes-Ad-Hoc Design
3. Constrained Length Huffman Codes-The Voorhis Method

การ Decode Huffman Code

ในการเข้ารหัสฮัฟแมนจะเป็นลักษณะที่ค่อนข้างตรงไปตรงมา โดยที่ตารางของ Symbol-to-codeword ที่ถูกเตรียมไว้โดยผู้ออกแบบ แต่ว่าในทางตรงข้ามการถอดรหัสเป็นอะไรที่ยากกว่า

1. Bit - Serial Decoding

เราจะทำการสร้างโครงสร้างกิ่งสาขาของรหัสฮัฟแมนขึ้นใหม่โดยใช้ตาราง Symbol-to-codeword โดยกระทำตามขั้นตอนดังนี้

1. อ่านสายบิตข้อมูลที่บีบไว้จนไปถึงปลายสุดของกิ่งสาขา

2.ในขณะที่ แต่ละบิตถูกอ่านแล้วก็จะถูกทิ้งไว้ จนเมื่อถึงปลายสุดก็จะได้ Symbol ออกมา

โดยเราจะทำซ้ำจนกระทั้งทุกอินพุทถูกใส่เข้าไป และเราจะพบว่าวิธีการนี้จะมี Input bit rate ที่ตายตัว แต่มี Output bit rate ที่แปลค่า

2.3.2. Lookup-Table-Based Decoding

วิธีการนี้จะให้ Output bit rate ที่คงที่ โดยจะต้องทำการสร้าง Lookup-table จากขั้นตอน Symbol-to-Codeword mapping ขั้นตอนการสร้าง Lookup-table คือ

- ให้ C_i เป็น Codeword ของ S_i ถ้าให้ C_i มี l_i เราจะต้องสร้างแอดเดรสทั้งหมด L บิต โดยที่ l_i บิตแรกเป็น C_i และที่เหลือ L - l_i อาจจะเป็นหนึ่งหรือ ศูนย์ ดังนั้น S_i จะมีแอดเดรสทั้งหมด2^ (L- l_i)
- ในแต่ละ Entry เราจะสร้าง (S_i , l_i) 2 ชุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำ Lookup-table ทำให้การ Decode ค่อนข้างง่ายและเร็วโดยขั้นตอนดังนี้

- จากชุดข้อมูลอินพุท เราจะอ่าน L บิตเข้าไปยังบัฟเฟอร์
- เราจะใช้ค่า L เป็นแอดเดรสชี้ไปยัง Look-up table จะได้ค่า Symbol ออกมา
- จากนั้นจะทิ้งค่าเดิมแล้วอ่านค่าใหม่มายังบัฟเฟอร์
- ทำซ้ำจนกระทั่งได้ Symbol ทั้งหมดออกมา

มาตรฐานการบีบข้อมูลแบบไม่สูญเสีย

1. มาตรฐานการบีบข้อมูลแบบถ่ายสำเนา (Facsimile Compression Standard)

อาศัยหลักการที่เลขไบนารีมีการซ้ำกันหลายค่า โดยเราจะเปลี่ยนโดยใช้วิธีที่เรียกว่า Run-length coding

2. มาตรฐานการบีบข้อมูลแบบ JBIG

Joint binary Image Experts Group เป็นการบีบข้อมูลที่พัฒนาจากวิธี Compress halftone ที่มีประสิทธิภาพ หรือที่เรียกว่า ISO/IEC IS 11544,ITU-T Rec โดยประกอบด้วย Modeler และ arithmetic coder โดย Modeler จะเป็นตัวประมาณค่า probability ของ symbol เพื่อส่งให้ arithmetic coder

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 มาตรฐานการบีบข้อมูลไม่สูญเสียแบบ JPEG

Joint Photography Experts Group ร่วมกับ ISO และ ITU-T เช่นเดียวกับ JBIG การใช้วิธีบีบข้อมูลแบบ JPEG จะเป็นการแปลงแบบอิสระ โดยใช้ Differential coding จาก Prediction residuals ดังตารางที่ 2.2

Category	Prediction Residual
0	0
1	-1 , 1
2	-3, -2, 2, 3
3	-7, .. , -4, 4, .. , 7
4	-15, .. , -8, 8, .. , 15
5	-31, .. , -16, 16, .. , 31
6	-63, .. , -32, 32, .. , 63
7	-127, .. , -64, 64, .. , 127
8	-255, .. , -128, 128, .. , 255
9	-511, .. , -256, 256, .. , 511
10	-1023, .. , -512, 512, .. , 1023
11	-2047, .. , -1024, 1024, .. , 2047
12	-4095, .. , -2048, 2048, .. , 4095
13	-8191, .. , -4096, 4096, .. , 8191
14	-16383, .. , -8192, 8192, .. , 16383
15	-32767, .. , -16384, 16384, .. , 32767
16	32768

ตารางที่ 2.2 Prediction residual categories for lossless JPEG compression

2.4 หลักการบีบข้อมูลภาพแบบสูญเสีย

การบีบข้อมูลแบบสูญเสียนั้นได้ข้อมูลกลับคืนมาไม่สมบูรณ์นัก ในภาพนิ่งนั้นมีการพัฒนาการใช้ Lossy compression กันอย่างแพร่หลายแสดงในรูปที่ 1.2 ในบทนี้จะกล่าวถึง Concept พื้นฐานของวิธีการนี้เฉพาะหัวข้อที่สำคัญ เริ่มโดยใช้กฎ Rate-distortion ในการหาขอบเขตของการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บีบข้อมูล และตามด้วยการอธิบายถึงแบบแผนการบีบข้อมูลในทางปฏิบัติ โดยเน้นที่การใช้ DCT และสรุปด้วยการพูดถึง DCT ที่ทำงานได้เร็วขึ้น

2.4.1 รูปแบบสำหรับที่ว่างเปรียบเทียบของภาพ

รูปภาพมักจะมีที่ว่างที่ไม่จำเป็นนอกเหนือจากที่เป็นเป้าสายตามนุษย์ ถ้าหากใครกำลังมองภาพที่ถูกบีบอัดจากปลายด้านหนึ่งไปอีกด้านหนึ่ง นั่นคือการสร้างสรรค์สิ่งที่มนุษย์มองเห็นจากข้อมูลทั้งหมดหลังจากการถอดการ์บีบอัดภาพ เป็นการฉลาดที่เราจะทำการลดจำนวนบิตข้อมูลให้สั้นที่สุด โดยใช้ประโยชน์จากส่วนที่ว่างที่ซ้ำกันของภาพ

เราสามารถสร้างรูปสมการจากที่ว่างโดยให้สัมพันธ์กับพิกเซลเป็นฟังก์ชันแบบ 2 แกน เรียกว่า Covariance function คือ

$$Cov_x(i,j) = \sigma^2 e^{-\alpha\sqrt{i^2+j^2}}$$

โดยที่ σ^2 คือค่า Variance ของภาพ

i , j คือระยะทางจากพิกเซลอ้างอิง

เมื่อ Covariance function สามารถหาค่าได้ ถ้าเราให้

$$\rho_1 = \frac{E\left[X(i,j)X(i-1,j)\right]}{E\left[X^2(i,j)\right]}$$

$$\rho_1 = \frac{E\left[X(i,j)X(i,j-1)\right]}{E\left[X^2(i,j)\right]}$$

จะชี้ทิศทางเปรียบเทียบระหว่าง pixel ในแนวนอน และพิกเซลในแนวตั้ง ถ้าเราสมมุติว่า จะไม่มีความแตกต่างระหว่างแนวดิ่งและแนวนอนบริเวณพิกเซลที่ใกล้กันและจะได้ว่า สำหรับการคำนวณทั่วๆ ไป Covariance function จะเสียหายจำนวนมากเมื่อค่า ดังนั้นเราจำเป็นต้องไม่ให้ block of pixel เกิน 8 พิกเซล

รูปแบบทางคณิตศาสตร์สำหรับ Spatial redundancy จะต้องสนับสนุน 2 เป้าหมายคือ

1. มันต้องถูกใช้เป็นส่วนประกอบของระบบบีบอัดภาพ

2. ถ้าหากการบิดเบือนระหว่าง ภาพจริงกับภาพหลังจากถอดการบีบอัดเป็น D มันจะต้องสามารถหา Bit rate ที่น้อยที่สุดได้ในทุกแบบของการบีบอัดข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4.2 การเข้ารหัสการบีบข้อมูลแบบสูญเสีย (Lossy Compression)

Lossy Compression schemes แบ่งเป็น 2 ระดับคือ

1. Sample-base coding

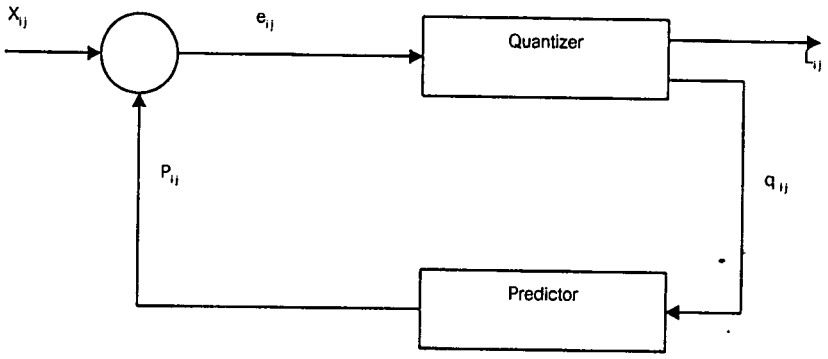
Sample-base coding สามารถใช้ได้ทั้ง spatial domain หรือ frequency domain วิธีการของ Sample-based compression โดยทั่วไปเป็น Scalar predictive coder อย่างเช่น DPCM ซึ่ง แสดงในรูปที่ 3.1 วิธีนี้เหมือนกับที่ใช้ใน Lossless JPEG ต่างตรงที่วิธี Lossless JPEG ไม่มีการ Quantization

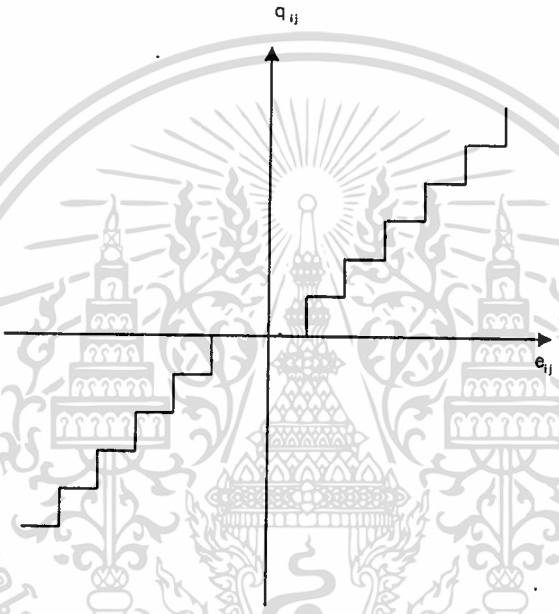
การทำงาน e_{ij} ที่ค้างอยู่จะสร้าง Prediction signal (P_{ij}) เพื่อทำการถ่วงกับอินพุท X_{ij} ซึ่งเป็นอินพุทพิกเซลอันที่อยู่ใกล้อันดับถัดไป ซึ่งถ้าพิกเซลมีความคล้ายคลึงกันมาก ก็จะทำให้ e_{ij} มีค่าเล็กยิ่งขึ้น

2. Block based coding

เนื่องจากวิธีการ DPCM ไม่สามารถจะบรรลุจุดประสงค์ของ R(D) bound ได้ และเรายังมีวิธี Block based coding ที่ให้อัตราการบีบอัดที่ดีกว่าสำหรับความบิดเบือนในระดับเดียวกัน ดังนั้นเทคนิคการบีบอัดภาพจึงถูกพัฒนาในลักษณะเป็น Blockwise โดยแบ่งเป็น 2 Classes คือ spatial domain block และ transform domain block วิธีที่นิยมใช้คือ transfer domain block โดยพิกเซลจะถูกรวมเป็น grouped block หลังจากนั้นจะถูก transform ไปยัง Domain อื่นอย่างเช่น frequency domain วิธีการ transform นั้นมีหลายวิธี วิธีที่ทำให้อัตราการบีบอัดดีที่สุดคือ Karhunen-Loeve transform แต่มันจะไม่เป็นภาพอิสระ (Image dependence) ดังนั้นจึงนิยมใช้ DCT เพราะเป็นวิธีที่เหลือที่ดีที่สุด ในการทำงานเป็น block นั้นขนาดที่เหมาะคือ 8x8 พิกเซล เพราะเป็นขนาดที่ไม่ใช้เมมโมรีมากเกินไปและเป็นขนาดการมองเห็นตามความละเอียดของสายตามนุษย์ที่สามารถสังเกตเห็นความแตกต่างได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





รูปที่ 2.6 DPCM compression method

(a) Encoding block diagram (b) Input-output characteristics of a typical quantizer

2.4.3 Discrete Cosine Transform

DCT เป็นวิธีการหนึ่งของการประมวลผลที่ใช้วิธีการ transform domain block ที่เป็นที่นิยม เนื่องจากมันสามารถเทียบเคียงข้อมูลภาพได้ในอัตราที่สูง

DCT เป็น Orthogonal transform ซึ่งสามารถเขียนในรูป Matrix ได้เป็น

DCT

$$Y = T X T^{t}$$

IDCT

$$X = T^{t} X T$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถเขียนสมการ การ transform จาก $X^{8x8DCT} \rightarrow y$ ได้ดังนี้คือ

$$y_{kl} = \frac{C(k)C(l)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} x_{ij} \cos\left(\frac{(2i+1)k\pi}{16}\right) \cos\left(\frac{(2i+1)l\pi}{16}\right)$$

โดยที่

$$C(k) = \begin{cases} 1/\sqrt{2} & \text{if } k = 0 \\ 1 & \text{otherwise} \end{cases}$$

และ Inverse DCT เขียนเป็นสมการได้เป็น

$$x_{ij} = \frac{C(k)C(l)}{4} \sum_{k=0}^{7} \sum_{l=0}^{7} y_{kl} \cos\left(\frac{(2i+1)k\pi}{16}\right) \cos\left(\frac{(2i+1)l\pi}{16}\right)$$

หรืออาจเขียนอยู่ในรูป Vector matrix ได้เป็น

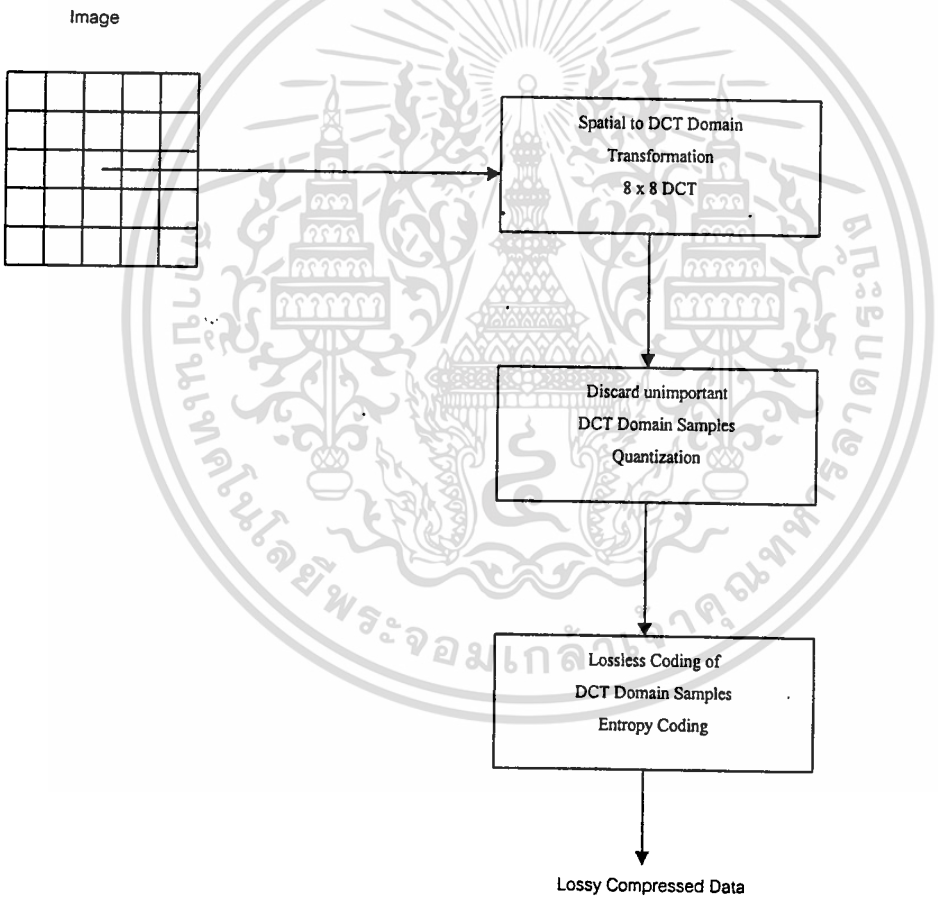
$$y = T\,x$$

โดยที่ T เป็นเมตริกพื้นฐานขนาด 64 x 64 พิกเซล ก็จะได้คำตอบเช่นเดียวกับสมการข้างต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการแปลงภาพแบบ DCT

ขั้นตอนการทำงานแบบพื้นฐานของระบบ DCT- based image coding แสดงดังในรูปที่ 3.2 ซึ่งจะแสดงถึงแกนการคำนวณที่ใช้ในมาตรฐานการบีบอัดข้อมูลภาพและวีดีโอแบบสูญเสีย ค่าสัมประสิทธิ์ของ DCT คือ ตัวถ่วงน้ำหนัก ซึ่งสอดคล้องกับ DCT basis waveform ในโหมดการบีบข้อมูลแบบสูญเสีย ค่าสัมประสิทธิ์บางตัวจะถูกลบออกไปและ waveform จะไม่ถูกใช้ในการ Decompress การสูญเสียส่วนใหญ่จะอยู่ที่การ quantization



รูปที่ 2.7 Block diagram การแปลงภาพแบบ DCT

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$X = \begin{bmatrix} 168 & 161 & 161 & 150 & 154 & 161 & 161 & 154 \\ 171 & 154 & 161 & 150 & 157 & 171 & 150 & 164 \\ 171 & 168 & 147 & 164 & 164 & 161 & 143 & 154 \\ 164 & 171 & 154 & 161 & 157 & 157 & 147 & 132 \\ 161 & 161 & 157 & 154 & 143 & 161 & 154 & 132 \\ 164 & 161 & 161 & 154 & 150 & 157 & 154 & 140 \\ 161 & 168 & 157 & 154 & 161 & 140 & 140 & 132 \\ 154 & 161 & 157 & 150 & 140 & 132 & 136 & 128 \end{bmatrix}$$

$$X = \begin{bmatrix} 214 & 49 & -3 & 20 & -10 & -1 & 1 & -6 \\ 34 & -25 & 11 & 13 & 5 & -3 & 15 & -6 \\ -6 & -4 & 8 & -9 & 3 & -3 & 5 & 10 \\ 8 & -10 & 4 & 4 & -15 & 10 & 6 & 6 \\ -12 & 5 & -1 & -2 & -15 & 9 & -5 & -1 \\ 5 & 9 & -8 & 3 & 4 & -7 & -14 & 2 \\ 2 & -2 & 3 & -1 & 1 & 3 & -3 & -4 \\ -1 & 1 & 0 & 2 & 3 & -2 & -4 & -2 \end{bmatrix}$$

$$Q = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 409 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

$$z_{kl} = round\left(\frac{y_{kl}}{q_{kl}}\right) = \left\lfloor \frac{y_{kl} \pm \left\lfloor \frac{q_{kl}}{2} \right\rfloor}{q_{kl}} \right\rfloor, k,l = 0,1,\ldots,7$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

039142

$$Z = \begin{bmatrix} 13 & 4 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & -2 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ขั้นตอนการ Decompress

$$\hat{Z} = \begin{bmatrix} 208 & 44 & 0 & 16 & 0 & 0 & 0 & 0 \\ 36 & -24 & 14 & 19 & 0 & 0 & 0 & 0 \\ 0 & 0 & 16 & 0 & 0 & 0 & 0 & 0 \\ 14 & -17 & 0 & 0 & 0 & 0 & 0 & 0 \\ -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{X} = \begin{bmatrix} 171 & 160 & 149 & 149 & 158 & 166 & 166 & 162 \\ 174 & 164 & 155 & 154 & 160 & 164 & 161 & 156 \\ 171 & 164 & 157 & 156 & 158 & 158 & 151 & 145 \\ 161 & 157 & 154 & 154 & 155 & 151 & 144 & 137 \\ 156 & 155 & 155 & 156 & 156 & 152 & 145 & 140 \\ 159 & 160 & 160 & 160 & 157 & 153 & 148 & 145 \\ 161 & 161 & 160 & 156 & 150 & 144 & 141 & 139 \\ 159 & 158 & 155 & 148 & 139 & 132 & 129 & 128 \end{bmatrix}$$

สรุป พบความแตกต่างหลังจาก Decompress ทำให้เกิดการบิดเบือนมากพอสมควร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 มาตรฐาน JPEG

จนถึงปัจจุบัน มาตรฐานกลุ่มที่ 3 และ กลุ่มที่ 4 สำหรับ การส่งผ่านแบบถ่ายสำเนา(Facsimile transmission) ยังคงมีมาตรฐานนานาชาติของการบีบข้อมูลภาพเพียงมาตรฐานเดียว อย่างไรก็ตามมาตรฐานนี้ยังกำหนดเพียงภาพแบบสองโทน โดยที่ยังไม่ได้กำหนดการบีบอัดแบบโทนสีต่อเนื่อง หรือภาพโทนขาวดำ (grayscale)

นับตั้งแต่ปี 1980 สมาชิกของ สหภาพโทรคมนาคมนานาชาติ (ITU) และ มาตรฐานองค์กรนานาชาติ (ISO) ได้ทำงานร่วมกันเพื่อสร้าง มาตรฐานร่วมนานาชาติ สำหรับการบีบอัดภาพโทนหลายระดับ(Multilevel still image) ความพยายามนี้รู้จักกันในชื่อว่า JPEG (the Joint Photographic Experts Group) โดยมาตรฐาน JPEG ที่เป็นทางการสอดคล้องกับมาตรฐาน การบีบอัดข้อมูลดิจิตอลของภาพนิ่งโทนต่อเนื่องของ ITU-R และ ISO

ในปัจจุบันมีการพัฒนาระบบบีบข้อมูลภาพแบบใหม่จำนวนมาก รวมทั้งการบีบข้อมูลที่มีการแปลงรหัสใหม่ด้วย โดยเป้าหมายก็เพื่อพัฒนาวิธีการทั่วไปของการบีบข้อมูลภาพ ซึ่งตอบสนองความต้องการที่หลากหลายออกไปดังนี้

- เพื่อให้ผู้ใช้สามารถเลือกได้ง่าย ระหว่างอัตราการบีบอัด และคุณภาพของภาพ
- ทำงานโดยอิสระโดยไม่ขึ้นกับชนิดของภาพ แหล่งของภาพ เนื้อหาของภาพ สี ขนาด ความละเอียดของพิกเซล ฯลฯ
- ใช้พลังในการคำนวณในระดับปานกลาง โดยใช้เครื่องในระดับล่างก็สามารถทำงานได้
- สามารถใช้ได้ทั้ง Single scan และ multiple scans
- ต้องเสนอทางเลือกในการเข้ารหัสแบบ Hierarchical ซึ่งใช้ความละเอียดต่ำได้

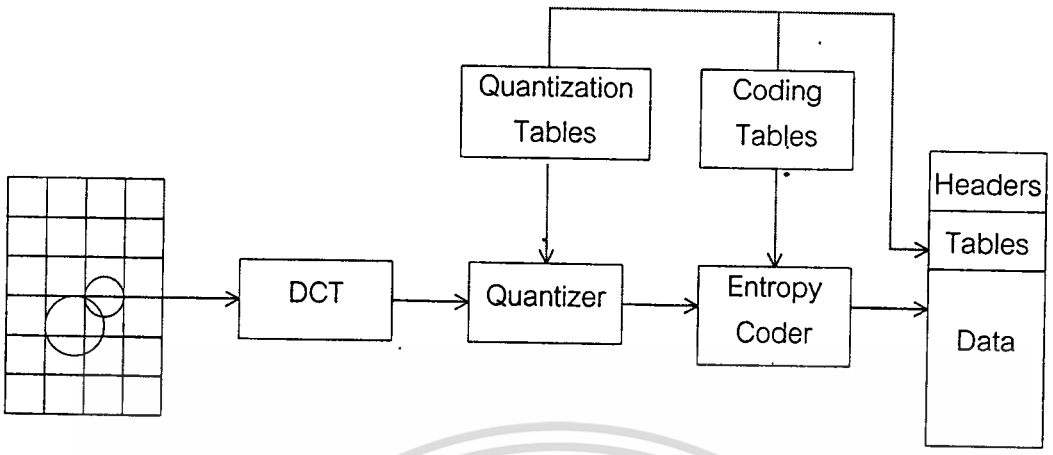
2.5.1 DCT Based Coding

มาตรฐาน JPEG กำหนดการทำงานไว้ทั้งหมด 4 โหมดคือ Sequential DCT- based , Progressive DCT-based , Lossless และ Hierarchical ทั้งหมดนี้ได้กล่าวมาบ้างแล้วในบทที่ 2 คือ Lossless เพื่อจะทำความเข้าใจกับโหมดอื่น จำเป็นจะต้องเข้าใจพื้นฐานของการเข้ารหัส DCT ดังรูปที่ 2.8 ซึ่งแสดงถึงการเข้ารหัส DCT สำหรับรูปที่มีองค์ประกอบสีเดียว หรือภาพขาวดำ (grayscale) สำหรับภาพที่เป็นสีจะต้องมีการทำในแต่ละองค์ประกอบทั้งสามสี ในรูปที่ 2.8 นี้ ขั้นแรกภาพจะถูกแบ่งเป็นบล็อกขนาด 8 x 8 โดยไม่ซ้อนทับกัน ถ้าหากว่าขนาดของภาพไม่พอดีกับบล็อกขนาด 8 x 8 ส่วนที่เหลือส่วนสุดท้ายจะถูกจัดการอย่างเหมาะสม

JPEG Encoder

แต่ละบล็อกที่เข้ามาจะถูกแปลงไปยัง Frequency domain โดยที่มาตรฐานไม่กำหนดว่าจะต้องใช้อัลกอริธึมใด ดังนั้นผู้ใช้ต้องเป็นคนเลือกอัลกอริธึมให้เหมาะสมกับงานที่ทำมากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 Block diagram of a JPEG encoder

จากบล็อกการทำงาน เอาท์พุทที่ได้จาก DCT จะถูก quantized และ entropy code โดยที่ entropy code นี้ประกอบด้วย 2 ส่วนคือ

- Predictive coder(for DC coefficient) , Run - length coder (for AC coefficient)
- Huffman coder Or Arithmetic coder

ถึงแม้ว่า Arithmetic coder จะให้อัตราส่วนการบีบข้อมูลที่สูงกว่าแต่ก็ยังมีระบบ JPEG ที่สนับสนุนเพียงเล็กน้อยเท่านั้นเนื่องมาจาก 3 เหตุผลสำคัญคือ

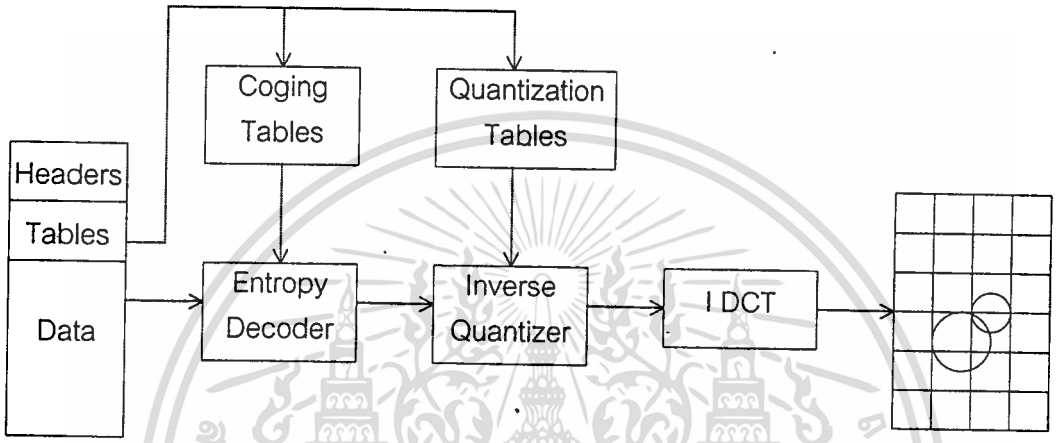
- ในการพัฒนา JPEG ในครั้งแรกๆไม่มีฮาร์ดแวร์ที่สนับสนุนการทำงานที่ซับซ้อน
- Arithmetic coder ส่วนใหญ่ใช้ครอบคลุมในอเมริกาและญี่ปุ่น ดังนั้นผู้ผลิตจึงไม่เต็มใจที่จะจ่ายค่าลิขสิทธิ์ เพื่อให้ได้ประสิทธิภาพที่ดีขึ้นเล็กน้อย

-สายการสร้างหลักที่ใช้อุปกรณ์น้อยที่สุดและเข้ากันได้กับมาตรฐาน JPEG ใช้เพียง Huffman coding เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

JPEG Decoding

ในรูปที่ 2.9 เป็นบล็อกไดอะแกรมของกระบวนการถอดรหัส JPEG หลังทำการกระจายรหัสออกมาผ่านกระบวนการ Entropy decoder และ Inverse quantization แล้วถูกแปลงให้กลับมาอยู่ใน Spatial domain แบบ 2 มิติ โดย IDCT ถือว่าการถอดรหัสได้เสร็จสมบูรณ์



รูปที่ 2.9 Block diagram of a JPEG decoder

2.5.2 การประมวลผลภาพสี

ที่กล่าวมาแล้วนั้นเป็นการกล่าวถึงระบบ JPEG ที่ ประมวลผลองค์ประกอบเดียว แต่ในทางปฏิบัติแล้วภาพจะต้องมีหลายสี ที่ความละเอียดต่างๆกัน ตัวอย่างเช่น สแกนเนอร์ส่วนใหญ่จะกำเนิดภาพที่มีสามองค์ประกอบสี คือ แดง เขียว น้ำเงิน (RGB) มาตรฐาน JPEG ถูกตั้งไว้ให้ไม่มีข้อจำกัดเกี่ยวกับสีที่เข้ามา จำนวนสูงสุดขององค์ประกอบสีของมาตรฐาน JPEG คือ 255 โดยแต่ละองค์ประกอบจะประกอบด้วยกลุ่มพิกเซลรูปสี่เหลี่ยม โดยอาจแทนด้วย p บิต โดย p จะเท่ากับ 8 ถึง 12 ใน DCT-based coders และ p เท่ากับ 2 ถึง 16 ใน Lossless coders และสำหรับภาพที่มีความละเอียดที่ต่างไปจากนี้ก็สามารถเข้ารหัส JPEG ได้เช่นเดียวกัน แต่จะไม่ถูกจัดให้ได้รับการสนับสนุนโดย JPEG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.3 การออกแบบ Quantization Tables

จากหลักการที่ว่า DCT-based coding ต้องการแหล่งข้อมูลเป็นเมตริก ขนาด 8 x 8 ในการทำ Quantization ที่เรียกว่า Quantization table โดยทุกๆองค์ประกอบ (Color component) สามารถใช้ตารางนี้ร่วมกันได้ ในแต่ละส่วนของ Quantization matrix จะเป็นค่าจำนวนเต็มตั้งแต่ 1 ถึง 255 และต้องกำหนดลำดับขั้นของการควอนไตซ์ให้สอดคล้องกับ สัมประสิทธิ์ DCT ในการควอนไตซ์นั้น ถ้าต้องการให้อัตราส่วนการบีบข้อมูลสูงก็อาจทำให้เกิดความบิดเบือนมากขึ้นหลังจากขยายข้อมูลกลับออกมา ดังนั้นเราจะต้องออกแบบ Quantization table ให้มีความสมดุลกันระหว่างความต้องการอัตราการผ่านข้อมูล (Bit rate) ที่ต่ำ และ คุณภาพของภาพ เทคนิคในการออกแบบ Quantization table ถูกแบ่งออกเป็น 2 คลาส คือ

- ใช้การทดลองการรับรู้และการมองเห็นของมนุษย์ (psychovisual) เป็นเกณฑ์
- ใช้ทฤษฎี Rate distortion และ Bit rate control เป็นเกณฑ์

2.5.4 Entropy Coding

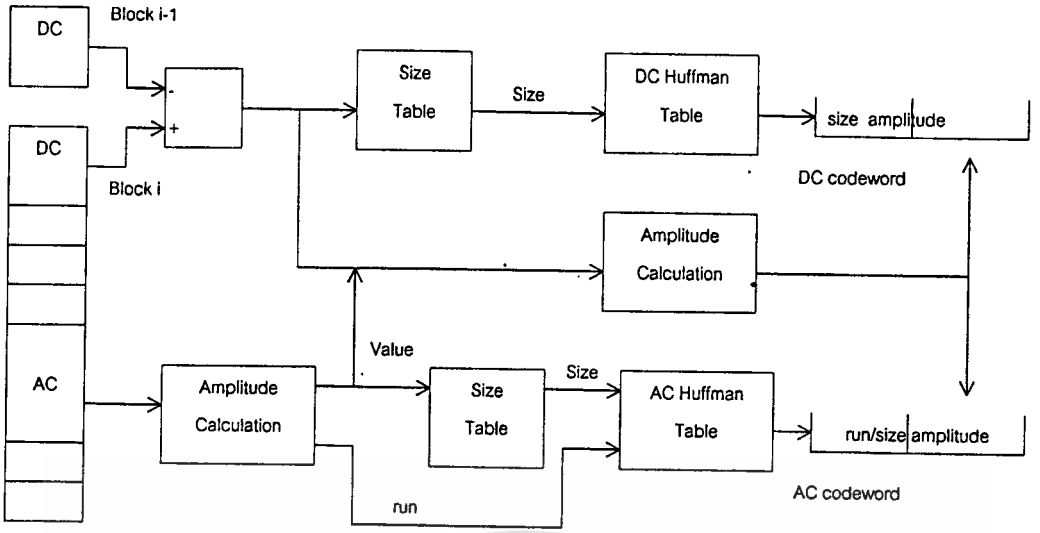
การประมวลผลในส่วนสุดท้ายของ JPEG ก็คือ Entropy coding บล็อกนี้เป็นตัว ปรับปรุงผลงานทั้งหมด โดยปฏิบัติการ Lossless coding บนสัมประสิทธิ์ DCT มันอาศัยวิธี Huffman หรือ Arithmetic โดยทางอ้อมซึ่งอธิบายในบทที่ 2 ถ้าหากว่ามันใช้วิธี Huffman เนื่องจากมันไม่มี Default table ดังนั้นจะต้องมีการกำหนด Huffman table อย่างน้อย 1 ตาราง แต่ส่วนใหญ่ตารางได้ถูกใช้จนเป็นมาตรฐานแล้ว มาตรฐาน JPEG มีข้อจำกัดเกี่ยวกับตารางนี้เพียง 2 ข้อคือ

- ต้องไม่มี Codeword ที่มีความยาวเกิน 16 บิท
- ต้องไม่มี Codeword ที่เป็นเลข 1 ต่อเนื่องทั้งหมด อย่างเช่น FF_{16}

Huffman Coding of the DC Coefficients

ในรูปที่ 2.10 แสดงถึงการเข้ารหัสฮัฟแมน ภายในสายการทำงานของระบบ JPEG ถ้าให้ DC_i และ DC_{i-1} เป็นสัมประสิทธิ์ DC ของบล็อก i และ i -1 เนื่องจากว่ามีการเท่าเทียมกันสูงระหว่างค่า DC ที่ติดต่อกัน JPEGจะใช้ Differential coding สำหรับสัมประสิทธิ์ DC ถ้าเป็นข้อมูลขนาด 8 บิตต่อพิกเซล DC Differential จะมีค่าอยู่ระหว่าง -2047 ถึง 2047 โดยค่าของช่วงนี้จะถูกแบ่งอยู่ในลำดับชั้นที่ 12 หลังจาก Lookuped table มันจะถูกเขียนอยู่ในรูปชุด (size , amplitude) โดย size จะเป็นตัวบอกจำนวนบิตของแอมปริจูด

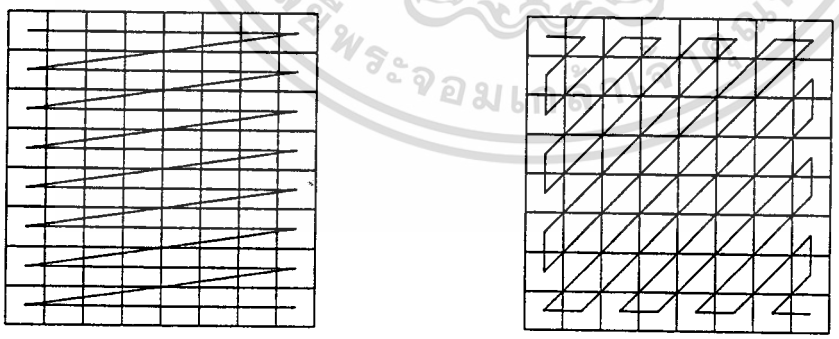
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 Huffman coding in baseline JPEG

Huffman Coding of the AC Coefficients

สำหรับสัมประสิทธิ์ AC ของข้อมูล 8 บิตต่อพิกเซล AC Coefficients จะมีค่าระหว่าง [-1023 ,1023] จากตารางที่ 2.2 ค่านี้จะจัดอยู่ในชั้นที่ 10 และจะถูกแสดงด้วย (size , amplitude) หลังจากการควอนไตซ์ สัมประสิทธิ์ AC ส่วนใหญ่จะเป็นศูนย์ และจะมีค่าที่ไม่ใช่ศูนย์เท่านั้นที่ต้องการการเข้ารหัส เราจะดำเนินการกับค่าสัมประสิทธิ์ AC ด้วยวิธีการซิกแซก (zig-zag order) เพื่อให้ได้ run-length coder ที่มีประสิทธิภาพมากกว่า



(a) Conventional order (b) Zig-zag order

รูปที่ 2.11 Conventional and zig-zag ordering in an 8 x 8 matrix

ตัวอย่างการเข้ารหัส AC Coefficient เช่นมีค่า -18 ซึ่งนำด้วย ศูนย์ 6 ตัวจะสามารถเขียนแทนด้วย (6/5 , 01101) โดยอาศัยตารางที่ 2.1 และวิธี One 's Complement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

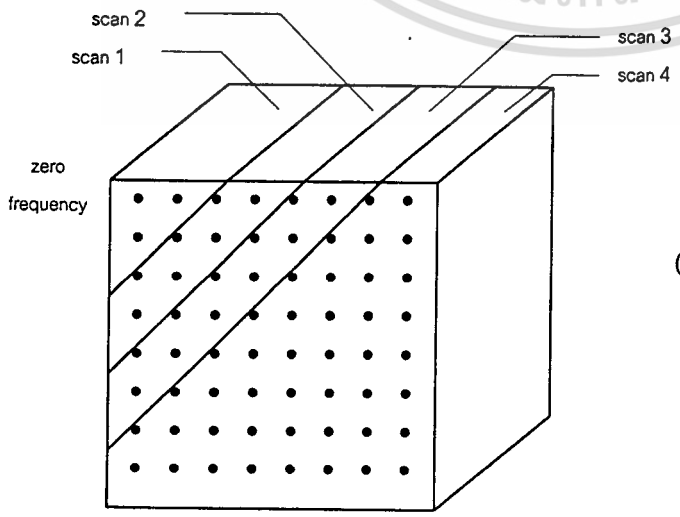
2.5.5 โหมดการทำงานของ JPEG

2.5.5.1 Sequential Coding

วิธีการนี้ เป็นวิธีที่ใช้กันทั่วไปมากที่สุด โดยบล็อกของภาพจะถูกดำเนินการเป็นลำดับจากซ้ายไปขวาและจากบนลงล่าง การทำงานตามลำดับนี้ ภาพนั้นอาจจะถูกส่งไปก่อนที่การถอดรหัสจะเสร็จสิ้นได้ และเช่นกันที่การถอดรหัสจะเริ่มการถอดรหัสได้ถึงไม้ยังไม่ได้รับข้อมูลภาพที่สมบูรณ์ทั้งหมด

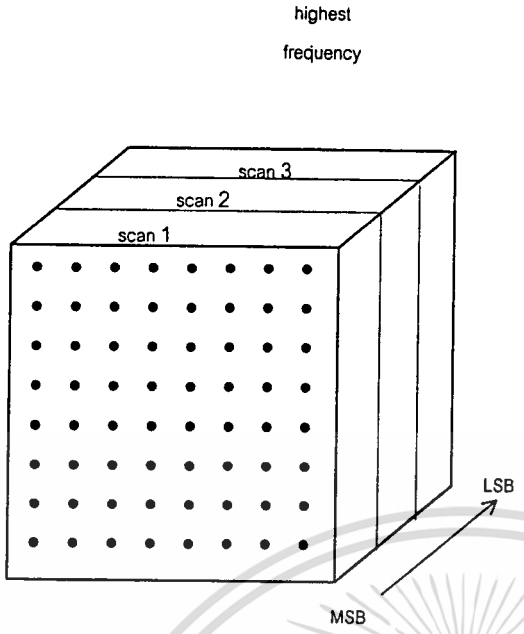
2.5.5.2 Progressive Coding

ในโหมดนี้ภาพจะถูกประมวลผลเป็นบล็อกแบบตามลำดับเช่นกัน แต่การทำงานจะเสร็จโดยการสแกนหลายครั้ง การเลือกการสแกนอาจใช้แบบ Spectral selection คือให้เลือกพิกเซลในแนวเฉียงไปเรื่อยๆ เมื่อพิจารณาบล็อกของ สัมประสิทธิ์ DCT ที่ควอนไตซ์แล้วขนาด 8 x 8 ดังรูปที่ 2.12 (a)โดยพิจารณาให้เป็นวัตถุ 3 มิติ เราจะพบว่าบล็อกจะถูกแบ่งตามแถบความถี่ และแต่ละแถบจะถูกส่งไปตลอดการสแกนที่แตกต่างกัน จะสังเกตุได้ว่าเพียงการสแกนครั้งแรกๆก็จะได้ข้อมูลทั้งไปจำนวนมากแล้วเนื่องจาก DCT Quantized ส่วนใหญ่นั้นจะมีความถี่มากที่มุมบนด้านซ้าย ดังเราจึงได้ภาพที่หยาบๆทางด้านเอาท์พุท แล้วจึงค่อยเติมให้ละเอียดขึ้นเรื่อยๆในการสแกนครั้งต่อๆไป ส่วนในการสแกนแบบ Successive approximation เราจะแบ่งแถบความถี่เป็นการจำนวนยกกำลังของสอง โดยจะส่ง Most significant bit ไปก่อนสำหรับภาพหยาบๆ และตามด้วยบิตรองลงไปจนถึง Least significant bit เพื่อให้ได้รายละเอียดภาพที่สมบูรณ์ ดังแสดงในรูปที่ 2.12 (b) เป็นการทำการสแกนแบบ Successive approximation 3 ครั้ง



(a) Spectral selection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



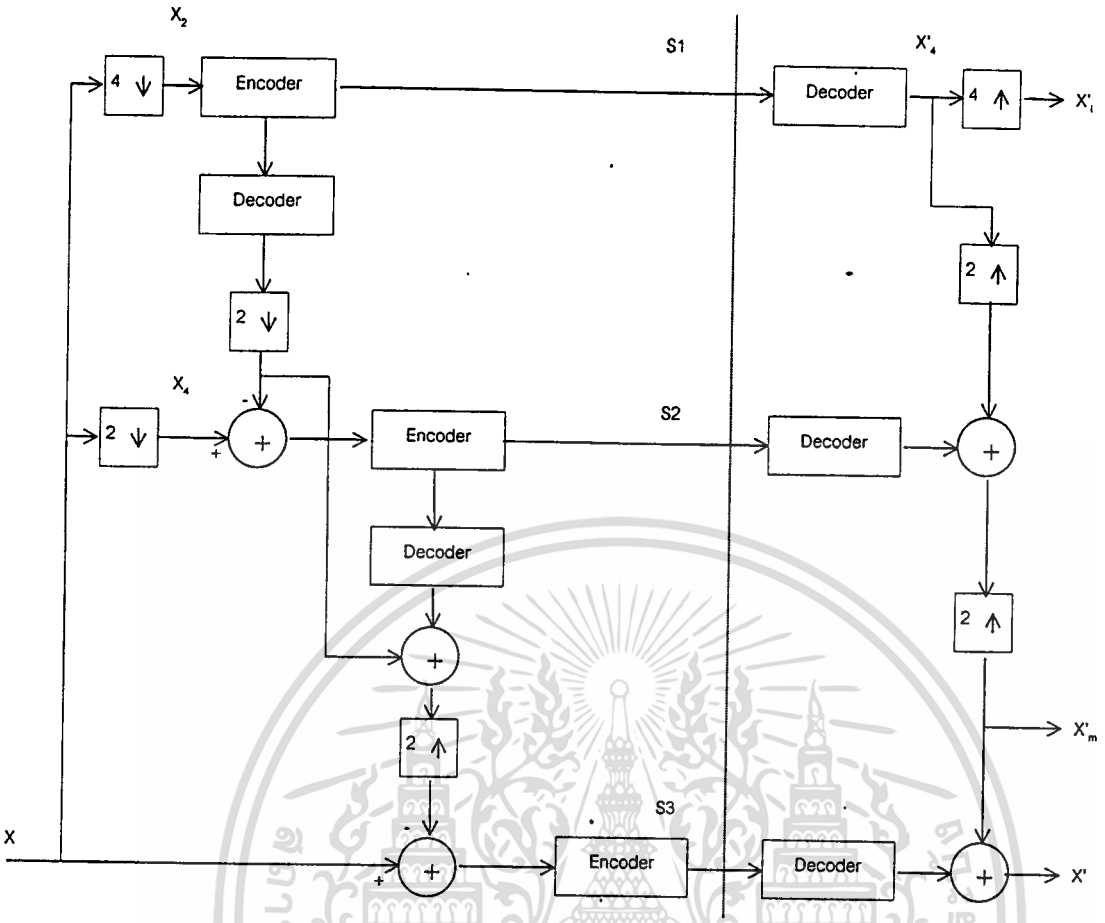
(b) Successive selection

รูปที่ 2.12 รายละเอียดการดำเนินตามขั้นตอน ในมาตรฐาน JPEG

5.6.3 Hierarchical Coding

ในการเข้ารหัสแบบ Hierarchical องค์ประกอบของภาพจะถูกเข้ารหัสเป็นอนุกรมของเฟรม เฟรมแรกมักเป็นภาพหยาบๆของภาพที่เราต้องการ ในเฟรมต่อไปจะเป็นผลต่างระหว่าง Possibly downsampled และ Possibly upsampled โดยเฟรมทั้งหมดสามารถใช้การเข้ารหัส แบบ Lossy JPEG หรือ Lossless JPEG ก็ได้ วิธีการ Hierarchical นี้จะเหมาะแก่ระบบที่ใช้ภาพแบบหลายความละเอียด อย่างเช่นเราต้องการแสดงผลทั้งบนเครื่อง Workstation ที่ใช้ความละเอียดสูงและเครื่องส่วนบุคคลที่ใช้ความละเอียดต่ำ จากรูปที่ 2.13 เป็นตัวอย่างของการเข้ารหัส Hierarchical แบบ 3 ระดับ จากภาพเริ่มต้น X เราจะเริ่มสร้าง subsample เป็น 2 แบบ โดย X_2 เป็น subsample ที่เป็นตัวประกอบของสอง ของทั้งสองมิติ ส่วน X_4 เป็น subsample ที่เป็นตัวประกอบของสี่ของทั้งสองมิติ รหัสข้อมูลที่ได้จะประกอบด้วยเฟรม 3 เฟรม คือ S_1 , S_2 และ S_3 โดยเฟรม S_1 จะบีบข้อมูล X_4 อย่างง่ายก่อน การใช้ S_1 นี้จะช่วยให้ตัวถอดรหัสสามารถแสดงภาพแบบหยาบก่อนได้คือ X'_1 S_2 จะเป็นค่าผลต่างของ X_2 กับ ค่าประมาณของ X_2 หรือ X'_2 หลังจากที่ทำการ upsampling X_4 ด้วยตัวประกอบสองแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.13 Three level hierarchical coder

สรุปคุณลักษณะกระบวนการเข้ารหัส JPEG ชนิดต่าง

1. Baseline Process

   - Coding : DCT-based , sequential , one to four color components
   - Resolution : 8 bit / pixel
   - Huffman coding ; 2 AC and 2 DC
   - Interleaved and noninterleaved scans

2. Extended DCT-based Process

   - Coding : DCT-based , sequential or progressive
   - Resolution : 8 or 12 bit / pixel
   - Huffman coding or arithmetic coding ; 4 AC and 4 DC tables
   - Interleaved and noninterleaved scans

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. Lossless Process

- Coding : Predictive , sequential
- Resolution : 2 - 16 bits / pixel
- Huffman coding or arithmetic coding : 4 DC tables
- Interleaved and noninterleaved scans

4. Hierarchical Process

- Coding : DCT-based or lossless process
- Multiple frames (non differential and differential)
- Interleaved and noninterleaved scans

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการบีบอัดข้อมูลวีดีโอและมาตรฐาน MPEG

3.1 บทนำ

บทนี้จะกล่าวถึงการบีบข้อมูลที่เป็นภาพนิ่งหลายๆภาพที่ต่อเนื่องอย่างเช่นVideo นับว่าเป็นสิ่งจำเป็นอย่างมากในการนำไปใช้งาน อย่างเช่นการจัดการรับส่งข้อมูล 166 Mbits/s กับ TV จะ ต้องทำให้มีอัตราการรับส่งเพียง 4.6 Mbits/s หมายถึงอัตราการบีบข้อมูลเท่ากับ 41:7 หรือในการรับส่งข้อมูลกับ CD ROM ซึ่งมีอัตราการส่งผ่านข้อมูลที่ 1.5 Mbits/s ดังนั้นจะต้องมีอัตราการบีบข้อมูลถึง 110 : 1 โดยทั่งไปถ้าหากเราพูดถึงข้อมูล Video เราจะต้องทำการบีบข้อมูลให้ได้มากถึง 4 เท่าของภาพนิ่ง จึงจะสามารถนำไปใช้งานได้

3.2 พื้นฐานการเข้ารหัสสัญญาณวีดีโอ

3.2.1 รูปแบบของ Image sequence

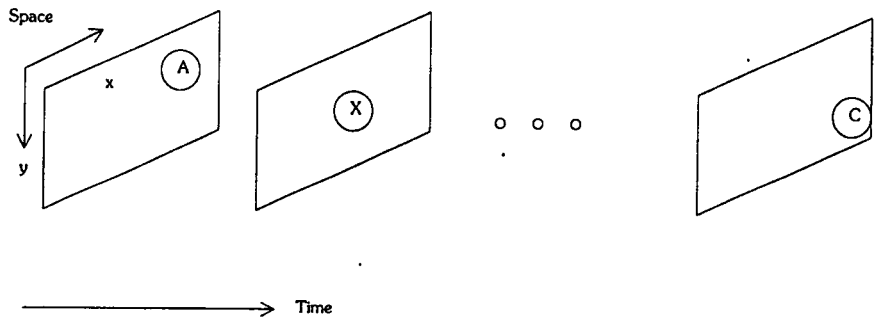
ในภาพนิ่งจะพบว่าข้อมูลนั้นมีส่วนของที่ว่างจำนวนมาก(spatial redundancy) ถ้าหากเราต้องการจับภาพการเคลื่อนไหวของวัตถุ 3 มิติ ในภาพแรกจะจับรายละเอียดภาพทั้งหมดที่รวมทั้งวัตถุและที่ว่าง และในภาพถัดๆไป เราคาดได้ว่าต้องมีส่วนที่เหมือนกัน และถ้าวัตถุเคลื่อนไป ภาพอื่นก็จะมีที่ว่างเช่นกันและเราก็จะคาดได้ว่าจะต้องมีการซ้ำกันของภาพในแต่ละเฟรมในบางพื้นที่ ที่เรียกว่า temporal redundancy โดยเป็นหมายของ Video compression ก็คือการหาประโยชน์จากทั้ง spatial redundancy และ temporal redundancy เพื่อให้ได้การบีบอัดที่ดีที่สุด ส่วนใหญ่แล้วในระบบเข้ารหัส Video จะมี 2 stage ในการทำให้ได้การบีบอัดที่ดีที่สุดคือ

1. Processing for reducing temporal redundancy
2. Processing for reducing spatial redundancy

3.2.2 การลด Temporal redundancy

การประมวลผลในทางอุดมคติที่จะทำการลด Temporal redundancy ก็คือการเข้าถึงทุกๆ พิกเซล จากเฟรมหนึ่งไปยังอีกเฟรมหนึ่ง แต่ว่าการกระทำเป็นพิกเซลนั้นย่อมมี noise เกิดขึ้นด้วย ดังนั้นแทนที่เราจะทำเป็นพิกเซล ๆ เราจะแบ่งข้อมูลเป็นพื้นที่ ขนาด 16 x 16 ที่เรียกว่า macroblocks ซึ่งเป็นขนาดที่เหมาะสมกับประสิทธิภาพในการลด Temporal redundancy และใช้พลังการคำนวณในระดับพอเหมาะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 Temporal correlation ในภาพที่ต่อเนื่องกัน

ถ้าเรามีสองเฟรมต่อกันดังรูปที่ 3.1 โดยสมมุติว่าเป็น frame (t-1) และ frame(t) ในขั้นแรกเรา จะทำการแยก frame(t) ไปเป็น พื้นที่ขนาด 16 X 16 (macroblock) โดยไม่ซ้อนทับกัน จากนั้นเราจะคำนวณ block 16 X 16 ที่สอดคล้องกันของ frame(t-1) ซึ่งโดยใช้การคำนวณที่ได้นี้ ขบวนการ temporal redundancy reduction จะสร้างตัวแทนของ frame(t) ซึ่งมีเพียงส่วนที่เป็นความแตกต่างของทั้งสองเฟรม ทั้งหากทั้งสองเฟรมมีความเหมือนกันมาก (High degree of temporal redundancy) ดังนั้น different frame จะได้ค่าที่ใกล้เคียง 0 จำนวนมาก นั่นหมายถึงมันจะยอมให้ถูกบีบได้มากขึ้น แต่ถ้าหากว่า frame(t) กับ frame(t-1) ต่างกันโดยสิ้นเชิง มันจะไม่สามารถหา different frame ที่สอดคล้องกันได้ และเราก็จะไม่ได้ประโยชน์อะไรจากวิธีการนี้ ในระบบการบีบอัดข้อมูลวีดีโอนั้น วิธีการ ลด temporal redundancy เราจะอ้างถึงกระบวนการที่ เรียกว่า Interframe coder

3.2.3 การลด Spatial redundancy

หลังจากขั้นตอนแรก ในแต่ละเฟรมยังมีความสัมพันธ์กันระหว่างพิกเซลที่เหมือนกัน เราสามารถหาประโยชน์จากพิกเซลที่เหมือนกันเหล่านี้ เพื่อให้ได้ผลการบีบอัดที่เป็นตัวแทนของ frame(t) โดยกระบวนการนี้จะถูกดำเนินการใน stage 2 ซึ่งเป็นขบวนการที่เหมือนกับในบทที่ 3 โดยที่กระบวนการนี้เราจะอ้างถึงกระบวนการที่เรียกว่า Intraframe coder

3.2.4 Motion Compensation

จากขบวนการที่กล่าวมาแล้วนั้น การสร้างความเกี่ยวเนื่องระหว่างเฟรมเราจะอ้างถึงการทำนายที่เรียกว่า motion compensation โดยเราจะให้เป็นกระบวนการที่ชดเชย ระยะทางของวัตถุที่เคลื่อนไหว โดยนำด้วย motion estimation ขบวนการจะทำการหาพิกเซลที่สอดคล้องกันระหว่างเฟรมโดยเขียนสมการ motion estimation ได้เป็น

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

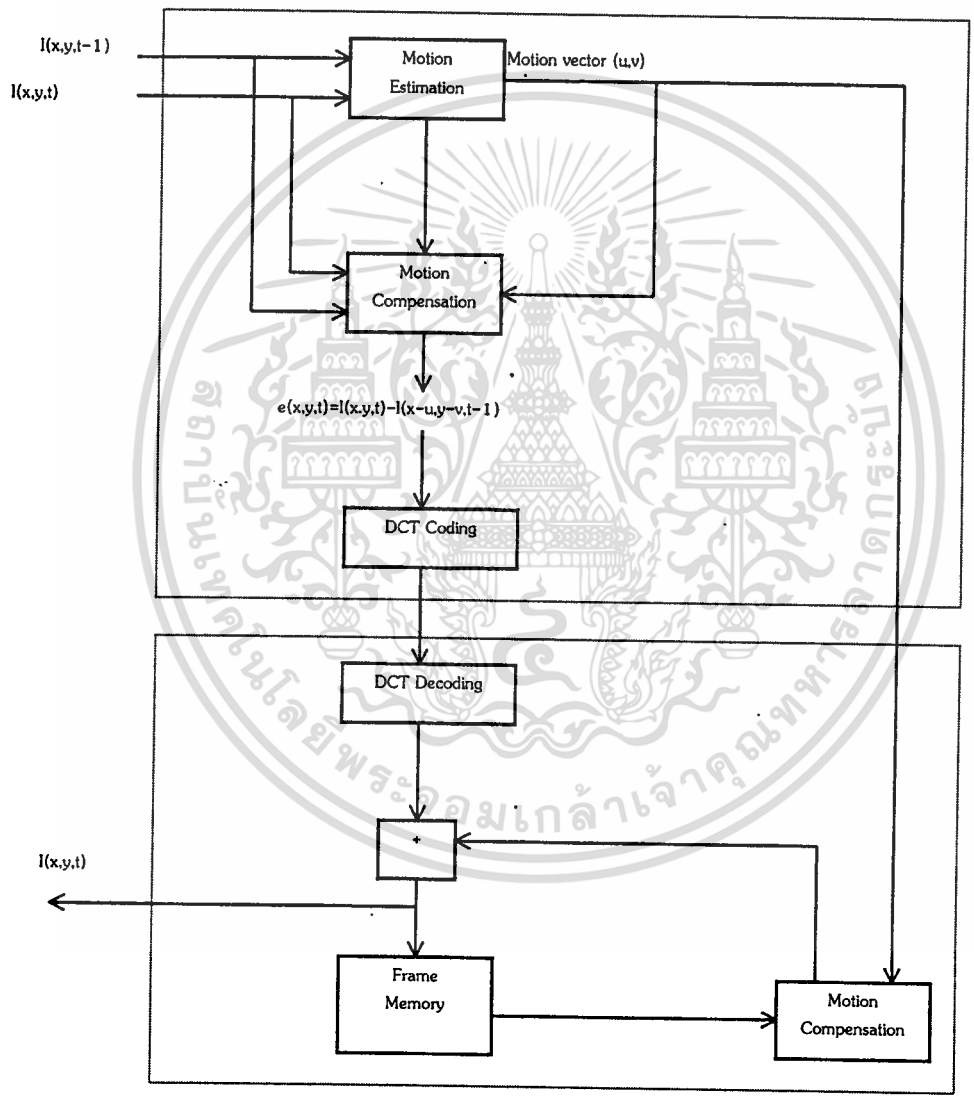
โดยที่ I(x,y,t) คือ ค่าของพิกเซลในตำแหน่งที่มีที่ว่างมากที่จุด (x , y)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(x,y) คือ frame(t)

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

จากกระบวนการนี้จะได้เอาท์พุทที่เป็นความสัมพันธ์ของการเคลื่อนไหวในหนึ่ง block ของเฟรมหนึ่งกับอีกเฟรมหนึ่ง ในรูปเวคเตอร์ (u , v) เมื่อเรารวมการทำงาน ของ Interframe coding และ Intraframe coding เข้าด้วยกันแล้ว มันจะให้อัตราการบีบอัดที่ดีกว่าทำงานเพียงลำพังกระบวนการเดียว



รูปที่ 3.2 การทำงานโดยทั่วไปของ Video coder และ Video decoder

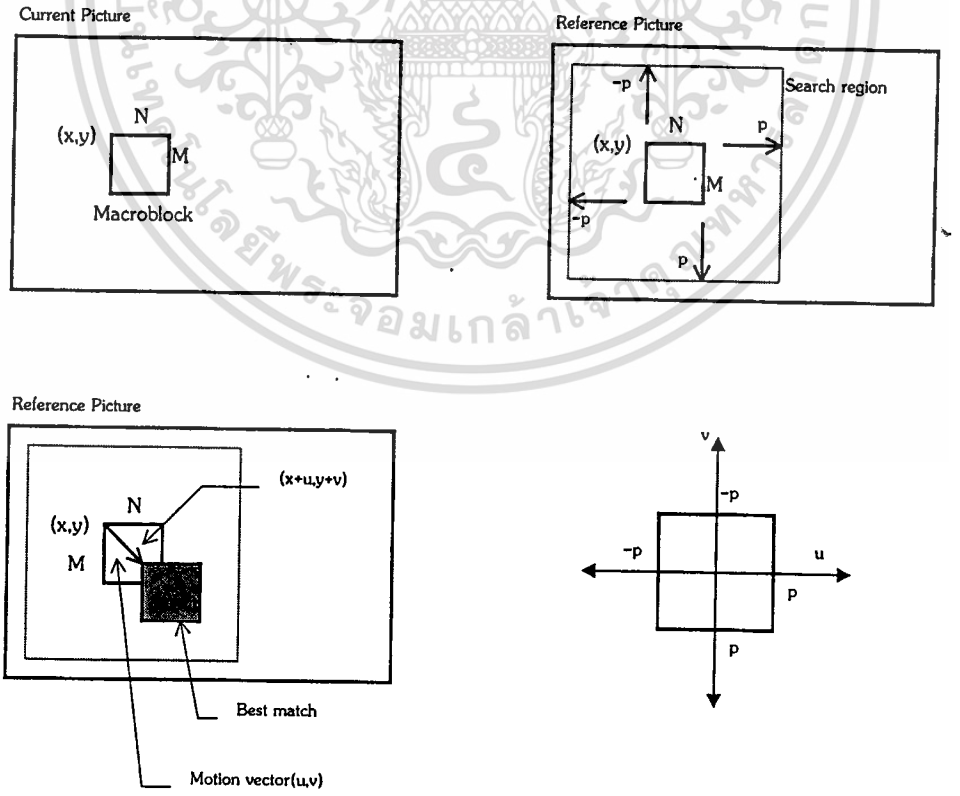
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 Motion Compensated Prediction

จากที่กล่าวมาแล้ววิธีการบีบข้อมูลวีดีโอที่มีประสิทธิภาพที่เรียกว่าไฮบริด(Interframe/Intra frame) สามารถแสดงขบวนการได้ดังรูปที่ 3.2 หนึ่งในส่วนสำคัญของขบวนการนี้คือ motion-estimation ซึ่งเราจะกล่าวถึงต่อไปนี้

ในรูปที่ 3.3 แสดงถึงปัญหาในการทำ motion-estimation ซึ่งอยู่ในมาตรฐานการเข้ารหัส วีดีโอ ถ้าให้รูปอ้างอิงมีขนาด N x M วัตถุประสงค์ของ motion-estimation คือ หา N x M macroblock ในภาพอ้างอิงที่แมทช์มากกว่า macroblock ในภาพปัจจุบัน โดยทั่วไปเรขาคณิตของ matching block ของภาพอ้างอิงไม่จำเป็นต้องเหมือนกับเรขาคณิตของภาพปัจจุบัน ตั้งแต่ตอนที่วัตถุในขนาดจริงเปลี่ยนไป อย่างไรก็ตามเราใช้วิธีการเรขาคณิตเพียงวิธีเดียวในระบบ Video coding เพื่อแปลงรูปแบบของการเคลื่อนไหว ที่เรียกว่า Rectangular geometry ก็เพียงพอ

ตำแหน่งของ macroblock มักจะถูกกำหนดให้เป็น (x,y) ซึ่งเป็นตำแหน่งที่มุมซ้ายด้านบนของพื้นที่ ในทางอุดมคติเราต้องการที่จะหา(search) ภาพอ้างอิงทั้งภาพที่ match มากที่สุด แต่ว่ามันไม่สามารถนำไปใช้ในทางปฏิบัติได้ ดังนั้นเราจะทำการจำกัดการค้าหา ที่พื้นที่ [-p , p] รอบๆจุดกำเนิดของ macroblock



รูปที่ 3.3 Motion-estimation process

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรทัดฐานการเปรียบเทียบ (Matching Criterion)

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

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

โดยที่ $-p \le i \le p$ และ $-p \le j \le p$

จากบรรทัดฐาน MAE เราจะกำหนดไว้ว่าตำแหน่ง Matching block เปรียบเทียบที่ดีที่สุดคือจุดที่ MAE (i , j) มีค่าน้อยที่สุด โดยที่ตำแหน่ง (i , j) นั้นจะเป็นตำแหน่งของ Motion vector ดังนั้นจะนำเราไปสู่ อัลกอริธึมในการ Search เวคเตอร์ (u,v) ที่มีเป้าหมายตามบรรทัดฐานนี้

3.4 อัลกอริธึมสำหรับ Motion estimation

มีวิธีการหลายวิธีที่ใช้ในการทำ Motion-estimation แต่ในมาตรฐาน Video coding ไม่ได้มีการกำหนดให้ใช้วิธีใด ดังนั้นเราสามารถนำแต่ละวิธีไปประยุกต์ใช้ได้ตามต้องการได้ โดยทั่วไป Motion-estimation ชนิดต่างๆสามารถสรุปได้ดังนี้คือ

1 Full-Search

2 Two-Dimensional Logarithmic Search

3 Parallel Hierarchical One-Dimensional Search (PHODS)

4 Hierarchical Motion Estimation

ตารางที่ 3.1 เปรียบเทียบคุณสมบัติของการ Search วิธีต่างๆ

Search Method	Operations per Macroblock	Operations for 720 x 480 pictures at 30 fps	
		p = 15	p = 7
Full search	$(2p+1)^2NM3$	29.89 GOPS	6.99 GOPS
Logarithmic	$(8\lceil \log_2 p \rceil+1)NM3$	1.02 GOPS	777.60 MOPS
PHODS	$(4\lceil \log_2 p \rceil+1)NM3$	528.76 MOPS	404.35 MOPS
Hierarchical	$[(2\lceil p/4 \rceil+1)^2+180]\ NM(3/16)$	507.38 MOPS	398.52 MOPS

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 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) ขึ้นเพื่อพัฒนามาตรฐานการเข้ารหัสข้อมูลภาพเคลื่อนไหวรวมเข้ากับข้อมูลเสียงบนสื่อเก็บข้อมูลดิจิตอล 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 ในเริ่มงานที่สอง นั่นก็คือการพัฒนา MPEG 1 ออกไปอีก โดยให้มีความยืดหยุ่นของรูปแบบข้อมูลเข้ามากขึ้น อัตราส่งผ่านข้อมูลที่ดีขึ้น และ ลดความผิดพลาดในการถอดรหัสลง โดยให้ชื่อว่า มาตรฐาน ISO 13818 (Generic coding of moving pictures and associated audio) หรือ MPEG 2 และได้เริ่มงานใหม่ที่ทำให้ได้ bit rate ต่ำมากคือ MPEG 4 ซึ่งคาดว่าจะเสร็จในปี 1998 มาตรฐาน MPEG ต้นกำเนิดจะเป็นอิสระกับการใช้งาน โดยมันจะกำหนดในเรื่อง วิธีการเข้ารหัส bit stream และ กระบวนการถอดรหัส โดยที่ไม่ได้กำหนดวิธีการเข้ารหัส จึงเป็นวิธีที่ยืดหยุ่นเพียงพอที่ทำให้ผู้ขายแต่ละรายสามารถที่จะพัฒนาแต่ส่วนของระบบของตนเองให้ดีที่สุด มาตรฐาน MPEG ได้ตีพิมพ์แบ่งออกเป็น 4 ส่วน คือ systems , video , audio และ conformance testing ในบทนี้เราจะพูดถึงเฉพาะส่วน video เท่านั้น

3.5.1 มาตรฐานวีดีโอ MPEG 1

3.5.1.1 พื้นฐานเบื้องต้น

มาตรฐานนี้เป็นอัลกอริธึมที่เป็นการบีบอัดข้อมูลแบบสูญเสียที่สามารถประยุกต์ใช้งานได้กับอินพุทฟอร์แมทอย่างมากมาย มันเหมาะสำหรับอุปกรณ์ที่มี bit rate ประมาณ 1.5 Mbits/s อย่างเช่น CD-ROM ในมาตรฐาน MPEG 1 จะใช้คำว่า Picture แทน คำว่าเฟรม โดยมันไม่สามารถรับข้อมูลแบบ Interlace ในระบบ Interlaced video นั้นแต่ละเฟรมจะถูกแบ่งออกเป็น 2 ฟิลด์ คือ top field และ bottom field ภาพในเฟรมของ Interlaced video ในหนึ่งเส้นสแกนจะต้องมีทั้งสองฟิลด์แทรกกันอยู่ ตัวอย่างเช่นถ้าเป็นวีดีโอความเร็ว 30 เฟรมต่อวินาที จะต้องมีการสแกนเส้นทั้งหมด 60 เส้นต่อวินาที ทางด้านในระบบ Noninterlaced video จะต่างกันคือมันจะสแกนต่อเนื่องไปจากมุมซ้ายบนไปจนถึงด้านล่างของเฟรมโดยไม่มีการแบ่งเป็นฟิลด์

กุญแจ สำคัญที่ต้องการตลอดการพัฒนาระบบมาตรฐาน MPEG 1 มีอยู่ 2 ประการคือ

- High compression
- Random access capability

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

การทำงานการบีบข้อมูลใน MPEG จะต้องร่วมด้วย

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

รูปแบบของข้อมูลเข้า (Source input format)

ใน MPEG 1 กำหนดให้ขนาดภาพมีได้สูงสุดที่ 4095 x 4095 พิกเซล อย่างไรก็ตามจำนวนมากที่การบีบข้อมูลวีดีโอที่ใช้มาตรฐาน MPEG 1 ถูกปรับไว้หรับ SIF (source input format) ซึ่งได้มาง่ายๆจากรูปแบบ CCIR 601 ในระบบโทรทัศน์ดิจิตอล คือประกอบไปด้วยสามส่วนคือ 1 องค์ประกอบแสง (Y) และ 2 องค์ประกอบผลต่างของสี (Cb ,Cr) มีสองทางเลือกสำหรับระบบ CCIR 601 คือ

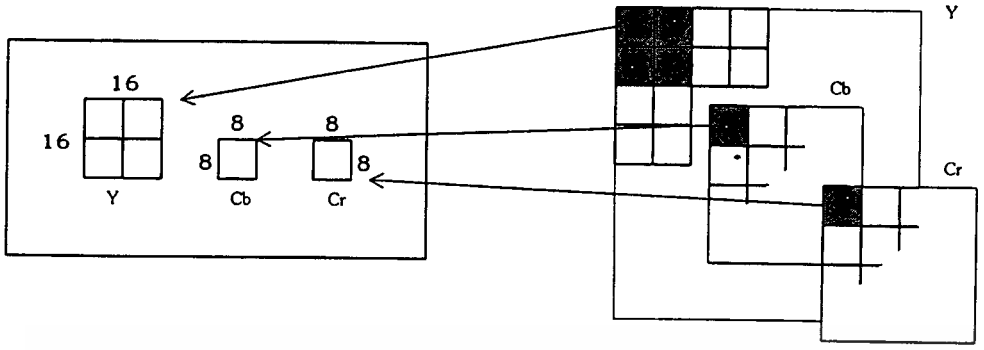
- สำหรับโทรทัศน์ระบบ NTSC เป็นระบบที่ใช้การสแกน 625 เส้นต่อเฟรม และมี 60 เฟรมต่อวินาที เฟรมขององค์ประกอบแสง 720 x 480 พิกเซล และองค์ประกอบสีแต่ละอันเป็น 360 x 480 พิกเซล

- สำหรับโทรทัศน์ระบบ PAL เป็นระบบที่ใช้การสแกน 525 เส้นต่อเฟรม และมี 50 เฟรมต่อวินาที เฟรมขององค์ประกอบแสง 720 x 576 พิกเซล และองค์ประกอบสีแต่ละอันเป็น 360 x 576 พิกเซล

องค์ประกอบ YCbCr ของ MPEG 1 จะต้องมีการแทรก(interleave) เสมอ โดยแมคโครบล็อกที่เล็กที่สุดมี 4 แมคโครบล็อกสำหรับองค์ประกอบแสง (Y) และมี 1 แมคโครบล็อกสำหรับแต่ละองค์ประกอบสี (Cb,Cr) โดยขนาดของแมคโครบล็อกเป็น 8 x 8 และมีขนาดสูงสุดเป็น 16x16

ภาพแต่ละภาพจะถูกแบ่งออกเป็นอนุกรมของแมคโครบล็อกจากซ้ายไปขวาและจากบนลงล่าง ดังนั้นภาพแต่ละภาพจะต้องมีความละเอียดเป็นจำนวนเท่าของ 16 ถ้าหากว่าไม่ตรงตามนี้ตัวเข้ารหัสจะนำส่วนที่เหลือไปไว้ด้านขวาหรือด้านล่างของภาพ และจะถูกตัดทิ้งโดยตัวถอดรหัสในภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ข้อกำหนดของ macroblock ในมาตรฐาน MPEG 1

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

ตารางที่ 3.2 ขนาดของภาพ สำหรับ ระบบ CCIR 601 ,SIF และพิกเซลในพื้นที่ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

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

ตารางที่ 3.3 ตัวแปรที่ถูกบังคับในการเข้ารหัส

จากข้อบังคับในตารางที่ 3.3 นั้นจะพบว่าเมื่อบังคับให้ pixel rate เท่ากับ 396 x 25 ดังนั้น ขนาดของภาพจะถูก จำกัดอยู่ที่ 352 x 288 พิกเซล เฉพาะพื้นที่สำคัญของระบบ SIF แบบ 25 Hz

3.5.1.2 ชนิดของภาพ

โดยทั่วไปแล้วการ ทำงานบางอย่างบน video stream รวมไปถึง editing , random access และ search การทำงานเหล่านี้มักจะเกิดปัญหาบ่อยๆเมื่อต้องบีบข้อมูลในอัตราที่สูง เพื่อให้มีความยืดหยุ่นระหว่างประสิทธิภาพและการเข้าถึงข้อมูลแบบสุ่ม (random access) MPEG 1 ได้แบ่งภาพเป็นชนิดต่างๆกัน 4 แบบ

- Intra pictures (I-pictures) เป็นภาพที่ถูกบีบอัดแบบภายในเฟรม (intraframe) และไม่ต้องอ้างอิงกับภาพอื่นในบิตสตรีม มันเหมาะสำหรับการเข้าถึงอย่างรวดเร็ว (fast random access) แต่จะได้อัตราการบีบอัดในระดับปานกลางเท่านั้น มันจึงเหมาะที่จะประยุกต์ใช้กับภาพ JPEG แบบไม่สูญเสียภาพเดี่ยวๆมาก

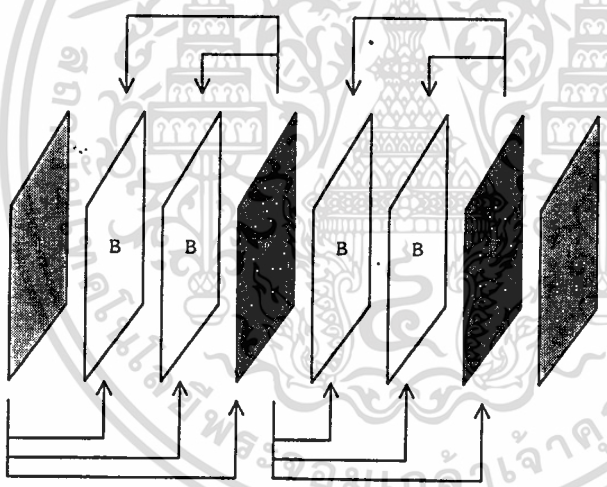
- Predicted pictures (P-pictures) เป็นภาพที่ถูกเข้ารหัสจากวิธี motion compensated prediction จาก P-picture หรือ I-pictures ที่พึ่งผ่านไป ภาพแบบ P-pictures จะมีอัตราการบีบข้อมูลที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สูงกว่าภาพแบบ I-pictures และมันสามารถที่จะเป็นภาพอ้างอิงในการกระบวนการ สร้าง ภาพ motion compensation แบบสองทิศทาง (B-pictured)

- Bidirectionally predicted pictures (B-pictures) เป็นภาพที่ให้ให้อัตราการบีบอัดดีที่สุด และเกิดความบิดเบือนได้ง่าย สร้างจากวิธี motion compensated prediction ของภาพ B-pictures และ P-pictures อื่นๆ ที่เข้ามากก่อนหน้ามันหรือเข้ามาทีหลัง

ในรูปที่ 3.5 แสดงถึงความสัมพันธ์ระหว่างภาพหลักสามแบบในอนุกรมวีดีโอ 8 ภาพ p_1 และ p_8 เป็นภาพแบบ I-pictures p_4 และ p_7 เป็นภาพแบบ P-pictures และที่เหลือเป็น B-pictures ในตัวอย่างนี้ p_3 จะถูกเข้ารหัสโดยวิธี motion compensated prediction จาก p_1 และ p_4 ที่จริงแล้วใน MPEG เราไม่ จำเป็นที่จะต้องใช้ทั้ง B-pictures และ P-pictures พร้อมกัน จากกระบวนการนี้จะพบว่า B-pictures จะมีบิตสตรีมนัยสำคัญต่ำที่สุด แต่การเพิ่ม B-pictures ระหว่าง I-pictures และ P-pictures จะไม่ทำให้อัตราการบีบข้อมูลสูงขึ้นเนื่องจากจะมีการตกลงของ temporal correlation ระหว่าง B-pictures และการโยงความสัมพันธ์กับ B-pictures และ I -pictures ก็จะเพิ่มขึ้นด้วย



รูปที่ 3.5 ความเกี่ยวข้องกันระหว่างภาพแบบ I ,P และ B ในอนุกรมวีดีโอ

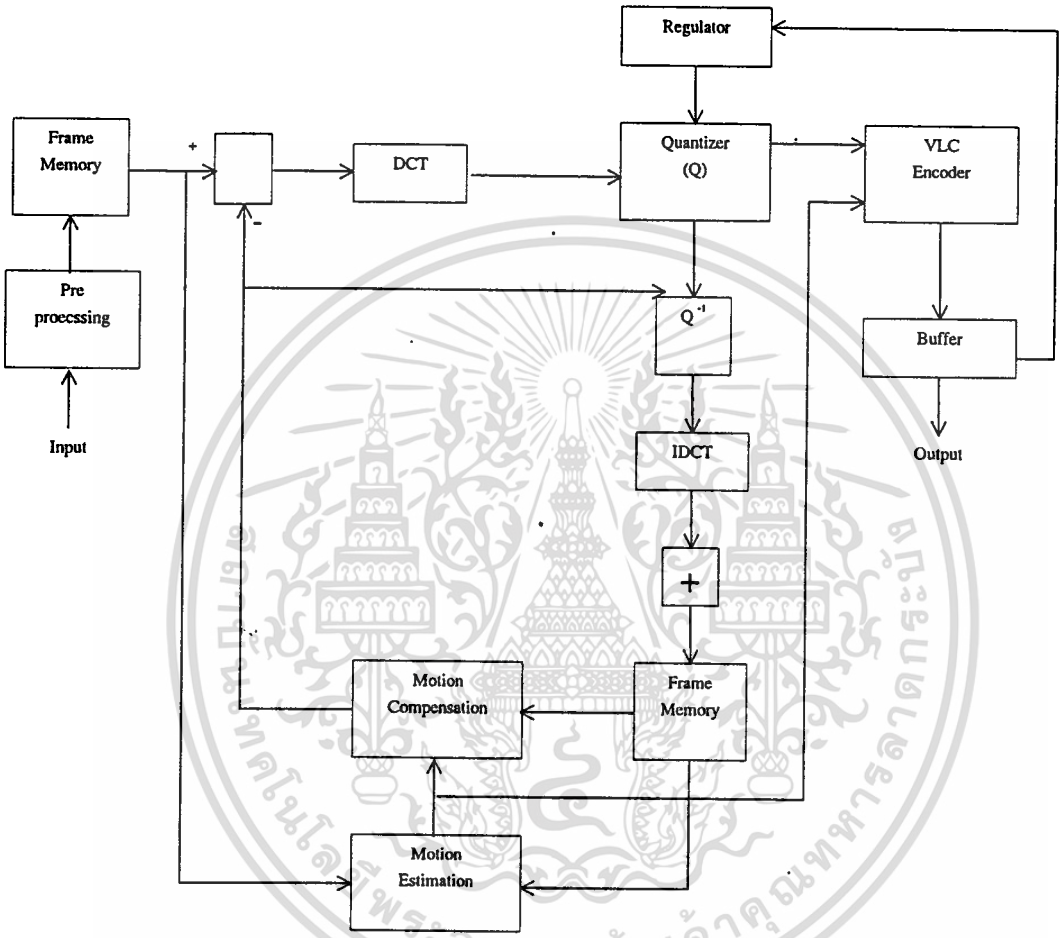
3.5.1.3 การเข้ารหัสวีดีโอ

เนื่องจากในมาตรฐาน MPEG ไม่ได้ กำหนดกระบวนการการเข้ารหัสไว้ มีแต่เพียงกำหนดนิยามของการกระบวนการทำงานกับบิตสตรีมและการถอดรหัสเท่านั้น แต่ว่าการเข้ารหัสเป็นกระบวนการที่จำเป็น รูปที่ 3.6 จะแสดง การดำเนินการที่สำคัญในการเข้ารหัส MPEG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Preprocessing

การเข้ารหัส(encoding) มักเริ่มด้วยการกระบวนการเบื้องต้นบางอย่าง ซึ่งอาจจะหมายถึงการแปรงสีเป็นแบบ YCbCr (interlaced to progressive) , การกรองเบื้องต้น (prefiltering) และ Subsampling กระบวนการเหล่านี้ไม่ถูกกำหนดไว้ในมาตรฐาน



รูปที่ 3.6 Block diagram of an MPEG encoder

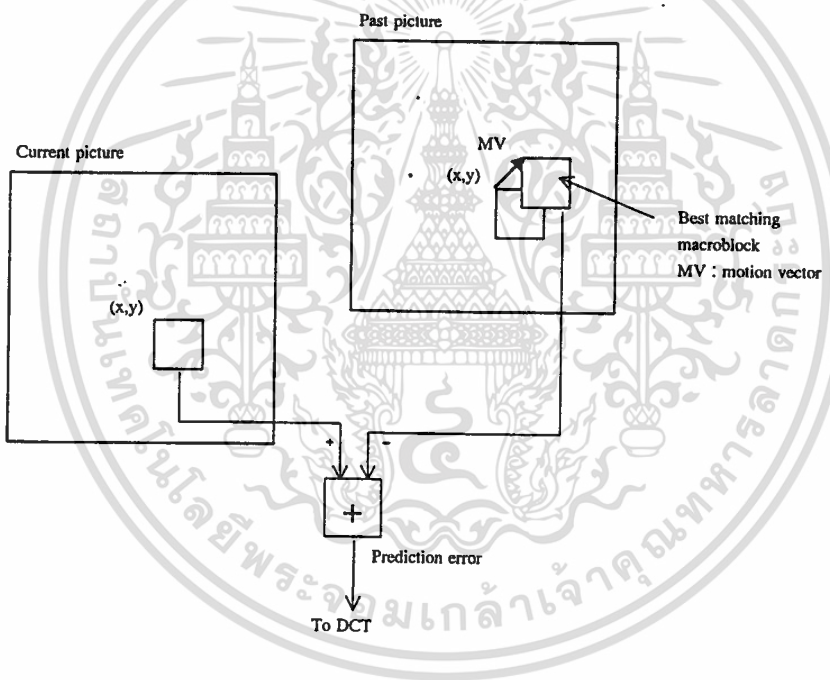
Motion estimation and compensation

หลังจากการทำ Preprocessing ตัวเข้ารหัสจะทำการเลือกชนิดของภาพขาเข้า ถ้าเป็น I-picture ไม่จำเป็นต้องทำกระบวนการ motion estimation หรือ compensation แต่ละ macroblock DCT จะถูกดำเนินการ และสัมประสิทธิ์ DCT จะถูกควอนไตซ์ และผ่านกระบวนการ VLC (variable-length coder) แล้วนำไปเก็บไว้ในบัฟเฟอร์ทางด้านเอาท์พุท ในแต่ละ macroblock จะต้องทำงานกับข้อมูลขนาด 8 x 8 ในงานที่ต้องการ bit rate คงที่ จะต้องมีตัวปรับ(Regulator) จะต้องปรับเมตริกควอนไตซ์เพื่อให้บิตเรตค่อนข้างคงที่ สำหรับงานที่ไม่ต้องการบิตเรตคงที่ฟังก์ชันนี้ก็ไม่จำเป็นต้องใช้ สำหรับกระบวนการควอนไตซ์จะเหมือนกับ Inverse quantized แล้วแปลงให้อยู่ใน

เอกสารนี้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

spatial domain โดยใช้ IDCT การทำงานนี้จะเรียนแบบพฤติกรรมของการถอดรหัสภาพเพื่อให้ได้สำเนาของภาพที่เข้ารหัสและถูกเห็นได้โดยตัวเข้ารหัส สำเนานี้จะถูกเก็บไว้ในหน่วยความจำและถูกใช้สำหรับการทำนายรหัสในอนาคต ผลจากการทำงานนี้ทำให้ตัวถอดรหัสคอยเฝ้าดูคุณภาพของภาพที่ส่งมาไม่ให้บิดเบือนไปจากภาพเดิม เนื่องจากว่า VLC เป็นการเข้ารหัสแบบไม่สูญเสียดังนั้นมันจึงไม่ต้องมีในเส้นทาง feedback

เมื่อข้อมูลอินพุทถูกทำให้เป็น P-pictures หรือ B-pictures แล้วตัวเข้ารหัสจะไม่ดำเนินการกับ picture macroblock โดยตรง แต่มันจะดำเนินการกับ prediction error ใน MPEG เราจะเรียกว่า interframe predictive coding สำหรับ P-pictures กระบวนการนี้อธิบายในรูปที่ 3.7

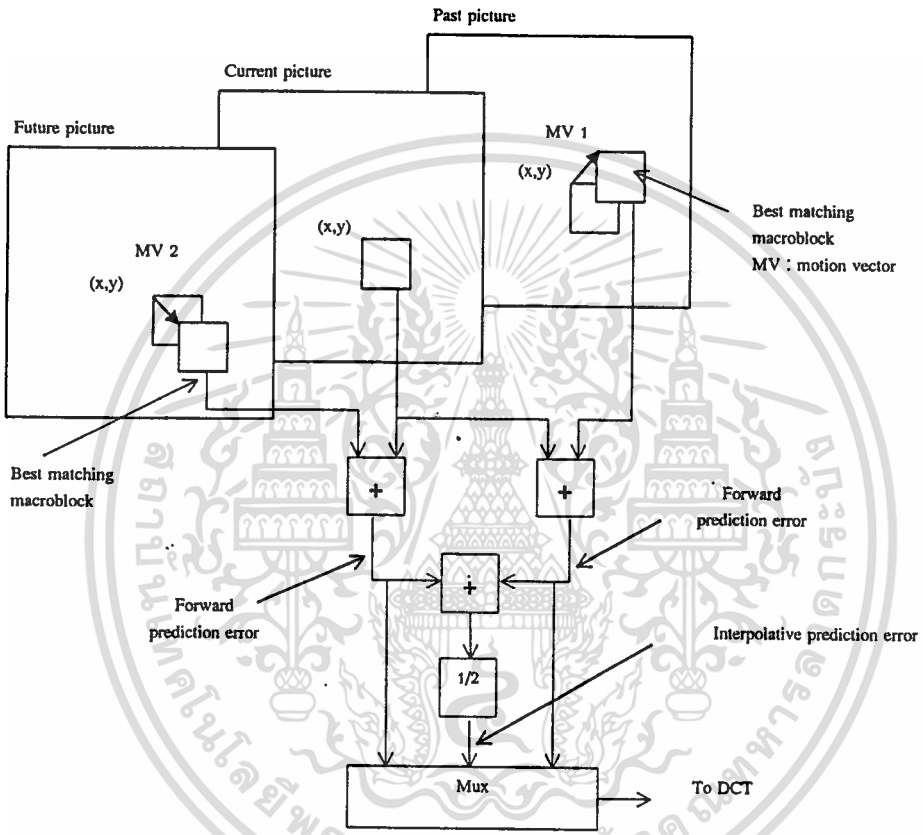


รูปที่ 3.7 Forward motion compensation

กระบวนการ motion estimation นี้จะให้ค่าโคออร์ดิเนตที่เป็นโมชั่นเวคเตอร์ของแต่แมคโครบล็อกซึ่งแมทช์(best match) มากที่สุดกับพื้นที่ค้นหา(search area) ของภาพที่ผ่านมา แมคโครบล็อก 2 บล็อก จะถูกนำมาลบกันเพื่อส่งไปยัง DCT สำหรับภาพชนิด B-pictures การทำ motion estimation ดำเป็น 2 ส่วนคือ สำหรับภาพก่อนและภาพหลัง ดังนั้นจะได้ 2 motion vector ตัวเข้ารหัสจะสามารถสร้าง prediction error macroblock 1 บล็อก จากทั้งสองหรือค่าเฉลี่ยของทั้งคู่ โดยกระบวนการนี้แสดงในรูปที่ 3.8 และใน MPEG จะหมายถึงกระบวนการ interframe

เอกสารนี้มีลิขสิทธิ์ ห้ามนำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

interpolative coding หลังจากนั้น prediction error จะถูกผ่านในยัง block-based DCT และ ควอนไตซ์ ค่าที่ได้นี้จะถูกส่งไปพร้อมกับ motion vector โดยวิธีการมัลติเพล็กและใช้ variable length coder ในระบบ MPEG ไม่เหมือน JPEG เพราะจะใช้การเข้ารหัสแบบฮัฟแมนเท่านั้นสำหรับ variable length coder



รูปที่ 3.8 bidirectional motion compensation

เนื่องจากว่าเป็นการทำนายแบบสองทิศทาง จึงจำเป็นต้องมีการจัดเรียงภาพที่เข้ามาใหม่บ้างเพื่อจะส่งภาพในลำดับที่ถูกต้องให้กับตัวถอดรหัสอย่างเช่นในรูปที่ 3.5 ภาพจะต้องเรียงโดยคำนึงถึงภาพที่จะต้องใช้ทำนายเข้ามาก่อนจะเรียงได้ดังนี้ P_1 , P_4 , P_2 , P_3 , P_7 , P_5 , P_6 และ P_8 และเมื่อถึงขั้นตอนการถอดรหัสจะต้องเรียงใหม่เพื่อให้ได้ลำดับภาพที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.1.4 Video decoder

รูปที่ 3.9 แสดงการทำงานของการถอดรหัสวีดีโอ MPEG ซึ่งเป็นวงจรที่ลักษณะเหมือนส่วนป้อนกลับของตัวเข้ารหัสวีดีโอ ขั้นแรกภาพที่เข้ามาจะถูกถอดรหัสฮัฟแมนพร้อมด้วยข้อมูลชนิดจาก header สำหรับแต่ละแมคโครบล็อกตัวสัมประสิทธิ์จะถูกดีควอนไตซ์แปลงกลับไปสู่ spatial domain 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) สำหรับแต่ละแมคโครบล็อกจะถูกคำนวณ IDCT และเข้ากระบวนการ motion compensation โดยจะบวกพื้นที่ที่ชี้โดย motion vector จากภาพ p_1 แล้วเก็บไว้ที่เอาท์พุทของ IDCT ภาพที่สร้างได้ใหม่นี้จะถูกเก็บไว้ใน future picture store

3. ภาพ p_2 (B-picture) คำนวณ IDCT แล้ว จะดำเนินกระบวนการ bidirectional motion compensation ซึ่งใช้ 2 motion vector เพื่อเข้าถึงพิกเซลที่สอดคล้องกับค่าในตำแหน่งของภาพ future picture และ ภาพ previous picture ที่เก็บไว้ทั้งสองทำการ predicted ให้ได้ภาพ p_2 แล้วรวมกันกับที่ทำ IDCT ถึงตอนนี้ p_2 จะเป็นภาพที่ตาม p_1 ไปและพร้อมที่จะแสดงทางหน้าจอได้ จากการทำงานจะสังเกตุได้ว่า p_2 ซึ่งเป็นภาพแบบ B-pictures จะไม่ต้องถูกเก็บไว้ในหน่วยความจำเนื่องจากไม่ต้องใช้ในการทำนายภาพต่อไป

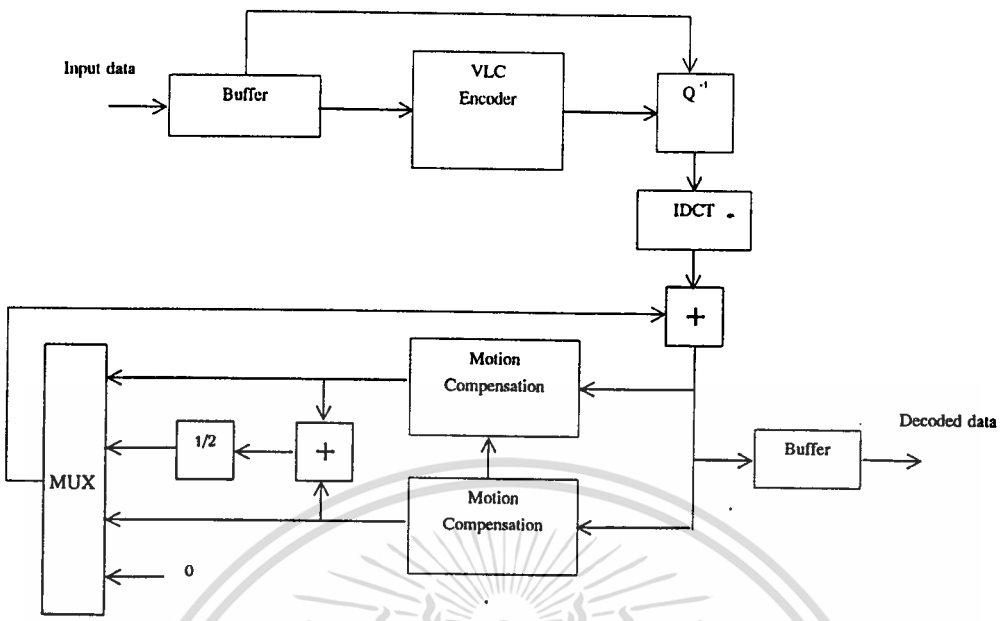
4. ภาพ p_3 (B-picture) จะทำแบบเดียวกับกระบวนการที่ได้ภาพ p_2 ออกมา และหลังจากที่ได้แสดงผลภาพ p_3 แล้ว ภาพ p_4 ก็จะถูกแสดงผลตามไป

5. ภาพ p_7 (P-picture) จะทำซ้ำเหมือนขั้นตอนที่สองและ ทำให้ข้อมูลในหน่วยความจำ previous picture ถูกเขียนทับ

6. ภาพ p_5 , p_6 (B-picture) ทำเหมือนกับในขั้นตอนที่สอง หลังจากที่ p_5 , p_6 ถูกถอดรหัสแล้วเราจะเรียงการแสดงผลออกไปเป็น p_5 , p_6, p_7 , p_8 ตามลำดับ

จากขั้นตอนทั้งหมดนี้ถือว่าการถอดรหัสภาพหนึ่งกลุ่มได้เสร็จสิ้นสมบูรณ์ และจะดำเนินการทำขั้นตอนเดียวกันนี้กับกลุ่มภาพต่อไป จากผังการทำงานจะมีภาพมากสุดสามภาพที่ถูกเก็บไว้พร้อมกันในชั่วขณะหนึ่ง สำหรับมาตรฐาน MPEG 1 จะมีความต้องการหน่วยความจำอย่างน้อยครึ่งเมกกะไบต์ ในการแสดงผลความละเอียดในแบบ SIF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 Block diagram of an MPEG decoder

3.5.1.5 โครงสร้างของ Video bit stream ที่เข้ารหัส

MPEG 1 กำหนด ให้ Video bit stream มีการเข้ารหัสแบบ Hierarchical ซึ่งมีทั้งหมด 6 ชั้น คือ (1) a sequence layer , (2) a group of pictures layer , (3) a picture layer , (4) a slice layer , (5) a macroblock layer และ (6) a block layer ดังแสดงในรูปที่ 3.10

ชั้นที่สูงที่สุดก็คือ sequence layer มันจะรวมถึง header และตามด้วยกลุ่มของภาพหนึ่งกลุ่ม หรือหลายกลุ่ม และจบด้วยรหัส sequence end ข้อมูลที่อยู่ใน header นั้นประกอบไปด้วย ขนาดในแนวตั้งและแนวนอนของแต่ละภาพ , มาตราส่วนของพิกเซล (pixel aspect ratio) , อัตราความเร็วภาพต่อวินาที , บิตเรตในหน่วย 400 บิตต่อวินาที และ ค่าหน่วยความจำต่ำสุดที่ต้องใช้โดยตัวถอดรหัส (หน่วยของ 2,048 ไบต์) นอกจากนั้น header ยังต้องประกอบด้วย DCT quantization matrix ของทั้ง intra-pictures และ nonintra pictures และข้อมูลตัวเลือกของผู้ใช้ สำหรับ intra pictures จะใช้ default quantization matrix ที่กำหนดโดยมาตรฐาน JPEG สำหรับองค์ประกอบแสง สำหรับ nonintra pictures ค่าทั้งหมดใน default quantization matrix จะเป็น 16

ชั้น Group of pictures (GOP) คือ ภาพจำนวนหนึ่งที่แสดงผลต่อเนื่องกันเป็นลำดับและอย่างน้อยต้องมี I-pictures หนึ่งภาพ สามารถเริ่มต้นด้วย I-pictures หรือ B-pictures ก็ได้ และต้องจบด้วย I-pictures หรือ P-pictures อย่างใดอย่างหนึ่ง ถ้าภาพแรกเป็น I-pictures แสดงว่ามันจะไม่เกี่ยวข้องกับ GOP ก่อนหน้ามัน ดังนั้น GOP นี้จะสามารถดำเนินการและแสดงผลได้อิสระ เพียง ลำพัง

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยไม่ขึ้น GOP อื่นๆ และเราจะเรียกว่า closed GOP ดังตัวอย่างในรูปที่ 6.2 ที่ header ของ GOP จะประกอบด้วย a closed-GOP flag ส่วนขยายและ ข้อมูลผู้ใช้

ชั้น Picture ถูกกำหนดให้เป็นข้อมูลการเข้ารหัสสำหรับแต่ละภาพ เพื่อให้สามารถเรียง P-pictures และ B-pictues ได้ โดยที่ header จะเตรียมข้อมูลชั่วคราวในการที่จะเรียงภาพที่จะแสดงผลให้เป็นลำดับถูกต้อง และนอกจากนั้น header ยังประกอบไปด้วยข้อมูล ชนิดของภาพ , การเข้าจังหวะ(synchronization) , ความละเอียดของภาพ และขนาดของ motion vector

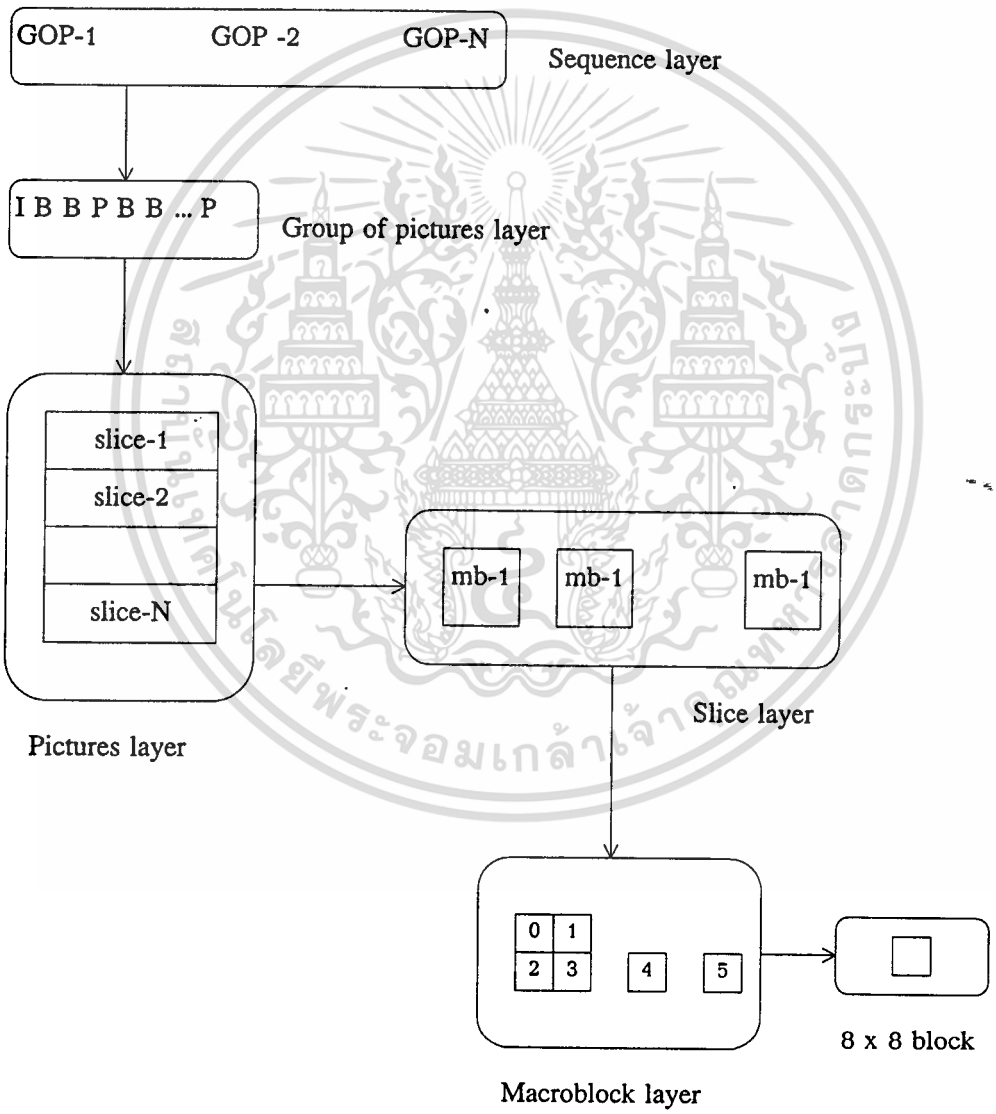
ชั้น Slice โดยขนาดของชั้นนี้ สามารถที่จะใหญ่ถึงภาพทั้งภาพหรือเล็กแค่เพียงหนึ่งแมคโครบล็อกก็ได้ ในกรณีที่เกิดความเสียหายของข้อมูล ตัวถอดรหัสจะสามารถทำการสร้างข้อมูลขึ้นใหม่โดยใช้ข้อมูลจาก Slice headers ตัวอย่างเช่นการส่งข้อมูลเกิดมีปัญหาการผิดพลาด ตัวถอดรหัสจะลดขนาดของ Slice ให้เล็กลงไม่เต็มภาพ ใน Slice headers จะบรรจุข้อมูลตำแหน่งของมันในภาพ และ quantizer scale factor ค่าระหว่าง 1 ถึง 31 เพื่อให้ตัวถอดรหัสใช้ในการ dequantize

ชั้น Macroblock และ block ใน Slice จะถูกแบ่งออกเป็น macroblock ซึ่งที่ header ของ macroblock จะบอก ชนิดของแมคโครบล็อก , ตำแหน่งของข้อมูล , รหัสของ motion vectors ในแนวตั้งและแนวนอน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

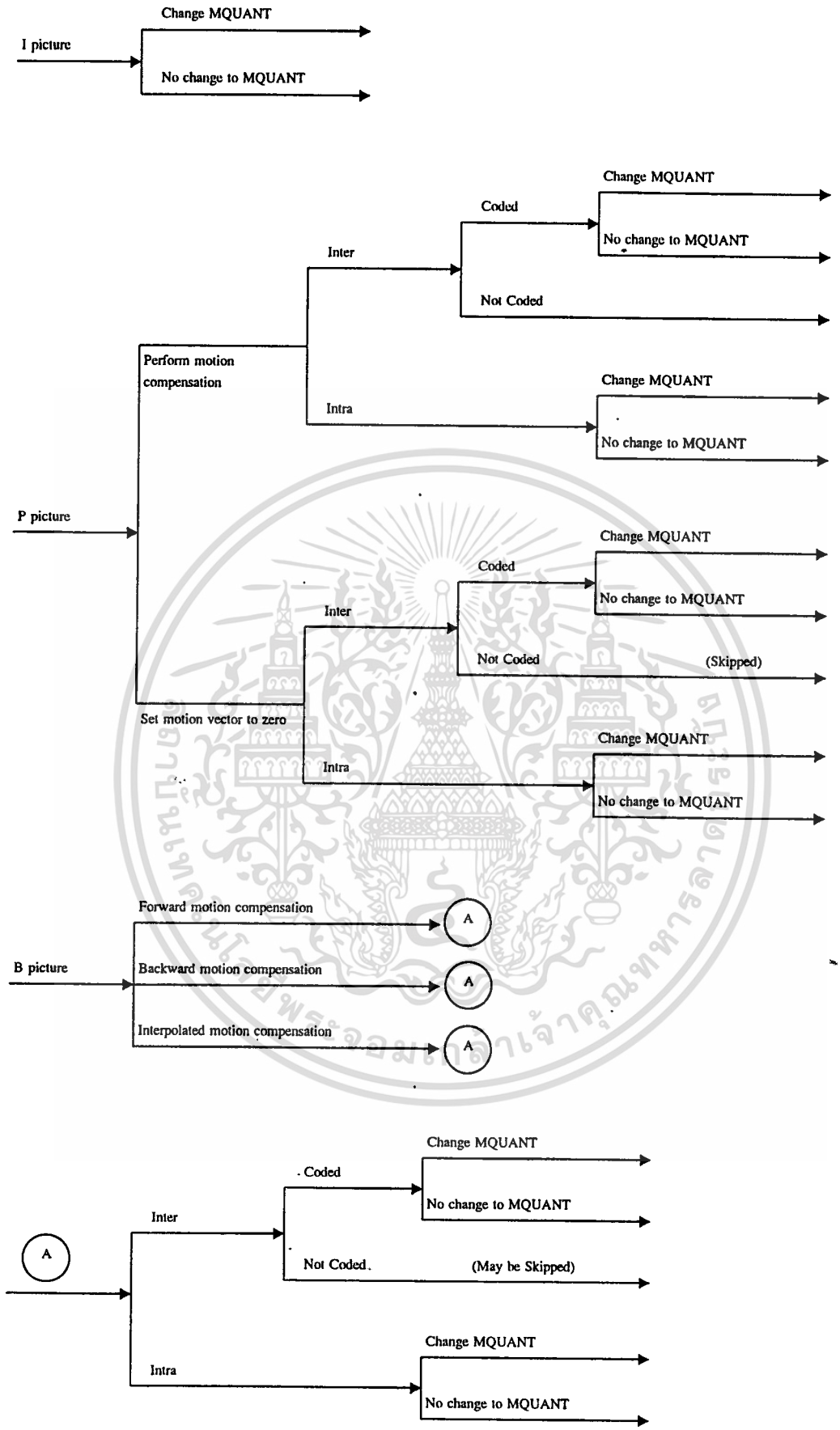
3.5.1.6 Macroblock Coding

จากที่กล่าวมาแล้วเราแบ่งภาพเป็นภาพหลัก 3 แบบ คือ I- , B- และ P-pictures แต่อย่างไรก็ตามในแต่ละแบบนั้น การดำเนินการกับรหัสในภาพชนิดนั้นๆ จะต่างกันออกไป ซึ่งสามารถแสดงเป็นแผนภูมิสาขาตามรูปที่ 3.11



รูปที่ 3.10 Syntax layers in MPEG 1 video coding

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 Decision trees for coding macroblocks in I- ,P- and B-pictures.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 มาตรฐานวีดีโอ MPEG 2

MPEG 2 เป็นผลงานในของการ ทำงานในเฟสที่สองของกลุ่มความร่วมมือ MPEG ต้นกำเนิดจริงๆของ MPEG 2 จะกำหนดมาตรฐานที่สามารถนำไปประยุกต์ใช้งานต่างๆได้กว้างมากที่สุดเท่าที่จะทำได้ เดิมนั้นเรายังมีความต้องการการพัฒนาเพื่อนำไปใช้งานดังเช่น

- อัตรา bit stream สูงถึง 5 Mbits / s สำหรับ NTSC/PAL และ 10 Mbits / s สำหรับงานที่ใกล้เคียงคุณภาพระดับในสตูดิโอ
- ใช้งานเข้ากับ MPEG 1 ได้
- ภาพที่มีคุณภาพสูง
- ความยืดหยุ่นต่อข้อมูลเข้า
- ความสามารถการเข้าถึงแบบสุ่ม
- การแสดงผลแบบ fast forward , reverse play และ slow motion
- คุณสมบัติ bit stream scalability
- การหน่วง (low delay) เพื่อการสื่อสารแบบสองทาง
- ลด bit error ในกระบวนการย้อนกลับ

MPEG ได้ตระหนักและเล็งเห็นว่า

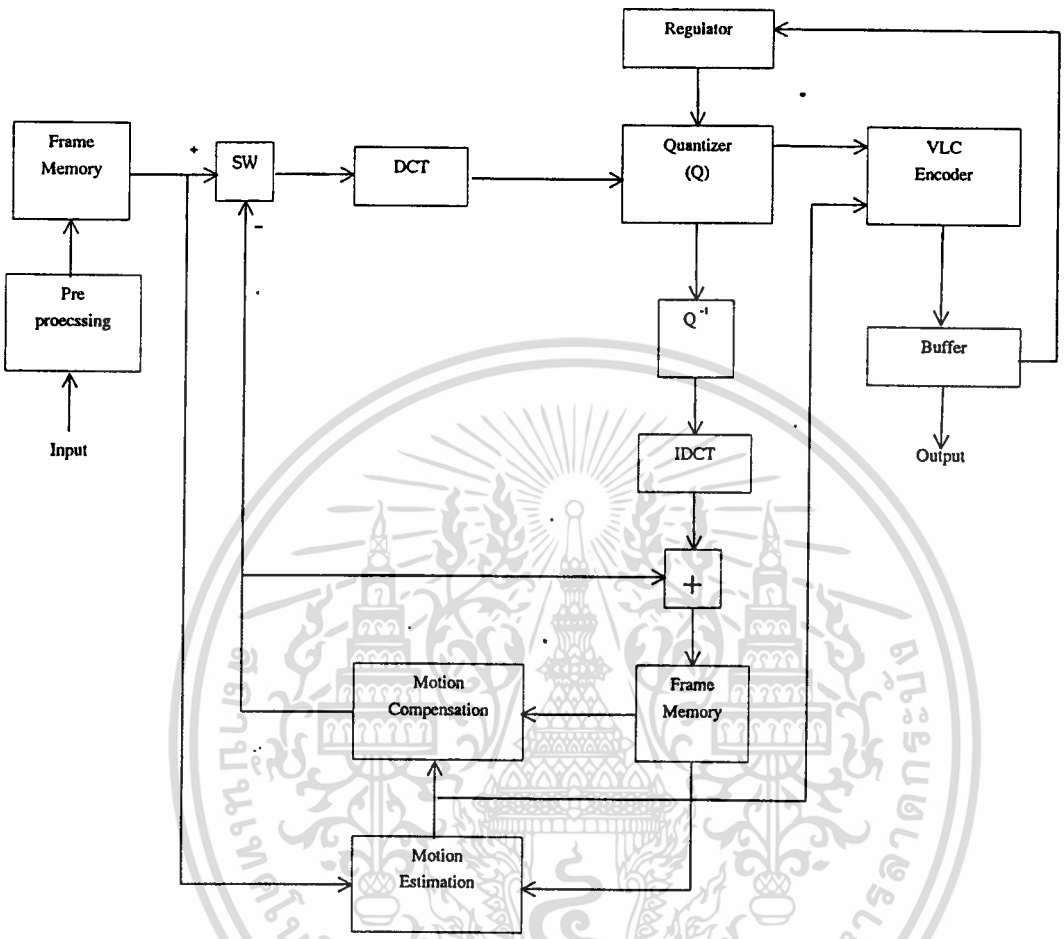
- ไม่ควร จำกัดค่าการเข้ารหัสบิตสูงสุดที่ 10 Mbits / s แต่จะต้องสามารถทำให้สนับสนุนการดำเนินการรหัสบิตเรตที่สูงกว่า ตัวอย่างเช่น 80 หรือ 100 10 Mbits / s สำหรับ HDTV
- การใช้มาตรฐานเพียงมาตรฐานเดียวไม่สามารถบรรลุความต้องการทั้งหมดนี้ได้
- งานส่วนใหญ่จะใช้เพียงบางส่วนของมาตรฐานเท่านั้น ดังนั้น MPEG 2 ได้พัฒนาเครื่องมือเฉพาะสำหรับงานแต่ละประเภท

สิ่งที่ได้มีการเพิ่มเติมและเปลี่ยนแปลงในมาตรฐาน MPEG 2

- สนับสนุนภาพแบบ Interlaced pictures
- สนับสนุนการทำ color subsampling ได้เพิ่มขึ้นคือ จาก MPEG 1 มีเพียงแบบ 4 : 2 : 0 ในมาตรฐาน MPEG 2 จะมีการทำ color subsampling แบบ 4 : 2 : 2 และ 4 : 4 : 4
- สามารถทำการ prediction ได้ทั้งกลุ่ม field pictures และ flame pictures โดยให้ชื่อว่า field prediction และ flame prediction ตามลำดับ
- Motion vector ถูกกำหนดให้เป็น half pixel resolution
- Scalalable bit streams

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2.3 กระบวนการเข้ารหัสภาพชนิดต่างๆ ในมาตรฐาน MPEG

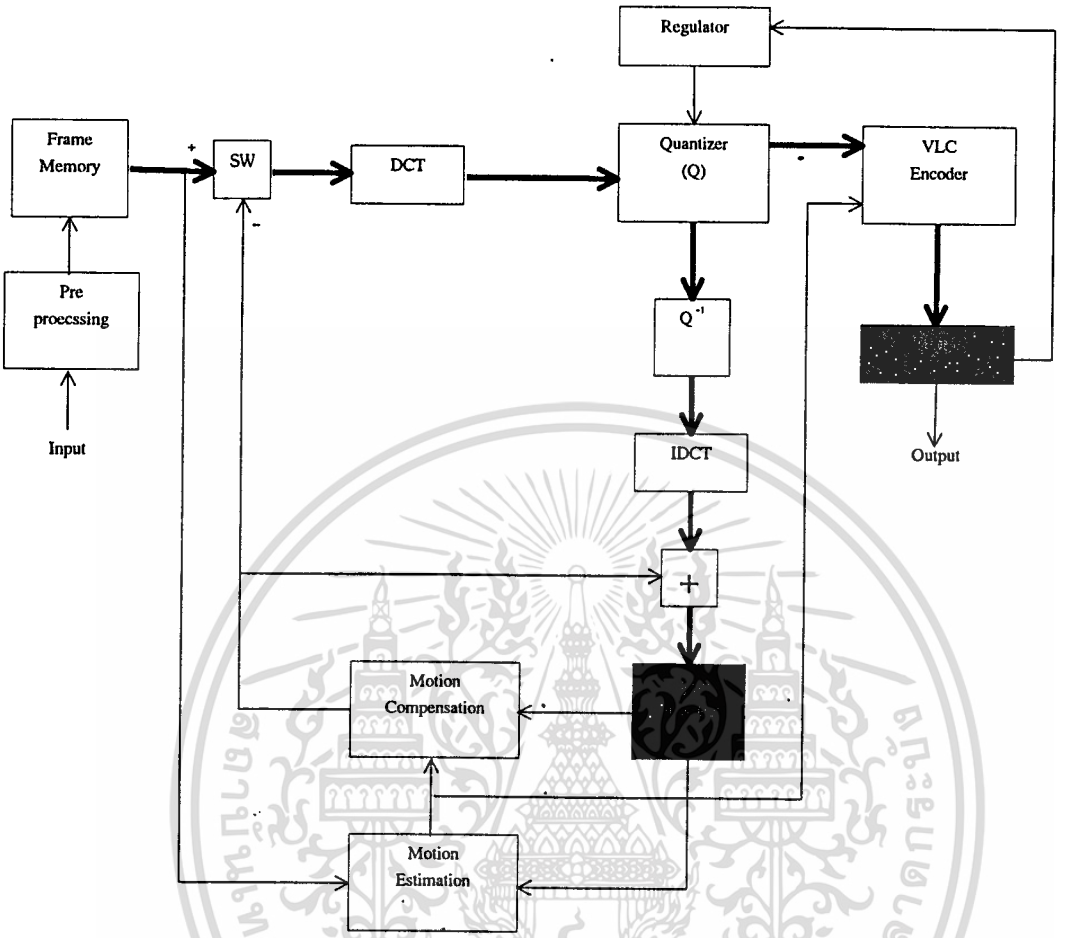


รูปที่ 3.12 Block diagram การประมวลผลในระบบ MPEG

ภาพที่ส่งเข้ามาภาพแรกหลังจากถูกจัดเรียงเรียบร้อยแล้วจะถูกส่งผ่านกระบวนการ DCT , Quantize และ การเข้ารหัส VLC ได้ข้อมูลออกไปที่เอาท์พุท และอีกทางหนึ่งข้อมูลนี้จากถูกเข้ากระบวนการย้อนกลับเช่นเดียวกับการ decode เพื่อให้ได้ข้อมูลที่เหมือนเดิมหรือใกล้เคียง แล้วนำกลับมาเก็บไว้ใน Frame memory นำ macroblock ของภาพถัดไปมาทำการ search หา motion vector ส่งไปยัง VLC โดย ขั้นตอน Motion estimation และแต่ละ macroblock จะถูกส่งผ่านมายัง กระบวนการ Motion compensation เพื่อนำมา compensate กับ ภาพเดิมใน Frame memory โดยมีลักษณะกระบวนการเหมือนการลบเพื่อให้ได้ส่วนต่างกันของภาพที่มีข้อมูลที่ไม่มี Temporal redundancy ซึ่งส่วนต่างนี้จะถูกส่งเป็นข้อมูลเฟรมต่อไปส่งไปในกระบวนการเข้ารหัสส่งออกไปที่เอาท์พุทต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้ารหัส I-pictures

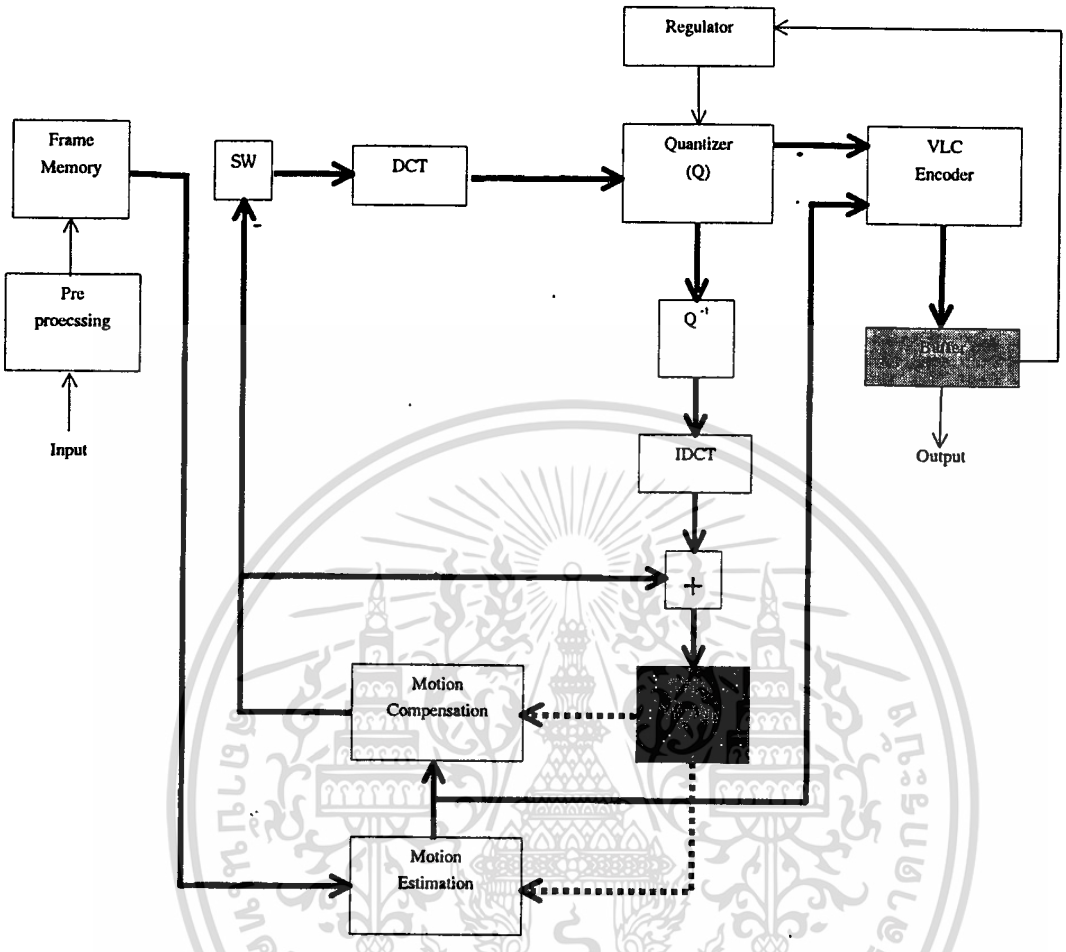


รูปที่ 3.13 การประมวลผล Intra picture

การดำเนินการกับ I-Picture ภาพชนิด Intraframe จะเป็นภาพที่ไม่ต้องมีกระบวนการ estimation และ compensation มักจะเป็นภาพแรกของ GOP ภาพนั้นจะทำการทำ DCT ทีละบล็อก แล้วส่งไปยัง Quantizer ซึ่งเป็นกระบวนการบีบข้อมูลแบบสูญเสีย และจากนั้นเข้ากระบวนการ Lossless compression คือ การทำ VLC (variable length code) ได้เป็นข้อมูลขนาด compact ส่งออกทางเอาท์พุทเพื่อการ storage หรือ ส่งไปยังอุปกรณ์อื่นๆแล้วแต่การใช้งาน ในระหว่างกระบวนการหลังจากการ Quantizing จะนำข้อมูลส่วนนั้นเข้ากระบวนการย้อนกลับด้วย โดยเก็บทีละบล็อกไว้ใน Frame memory จนครบเฟรม เพื่อนำไปใช้ในกระบวนการ Motion compensation ซึ่งเป็นกระบวนการที่ต้องมีการเปรียบเทียบเฟรมที่ผ่านไปแล้วกับเฟรมใหม่ที่เข้ามาให้ได้ ข้อมูล prediction error และ motion vector ของเฟรมถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเข้ารหัส P-pictures



รูปที่ 3.14 การประมวลผล Predicted picture

การดำเนินการกับ P-pictures จะต้อง นำข้อมูลภาพชนิด P นี้มาทำการ Estimation และ Compensation โดยอาศัยข้อมูลภาพก่อนหน้าที่เก็บไว้ใน Frame memory โดยมีวิธีการดังนี้คือ

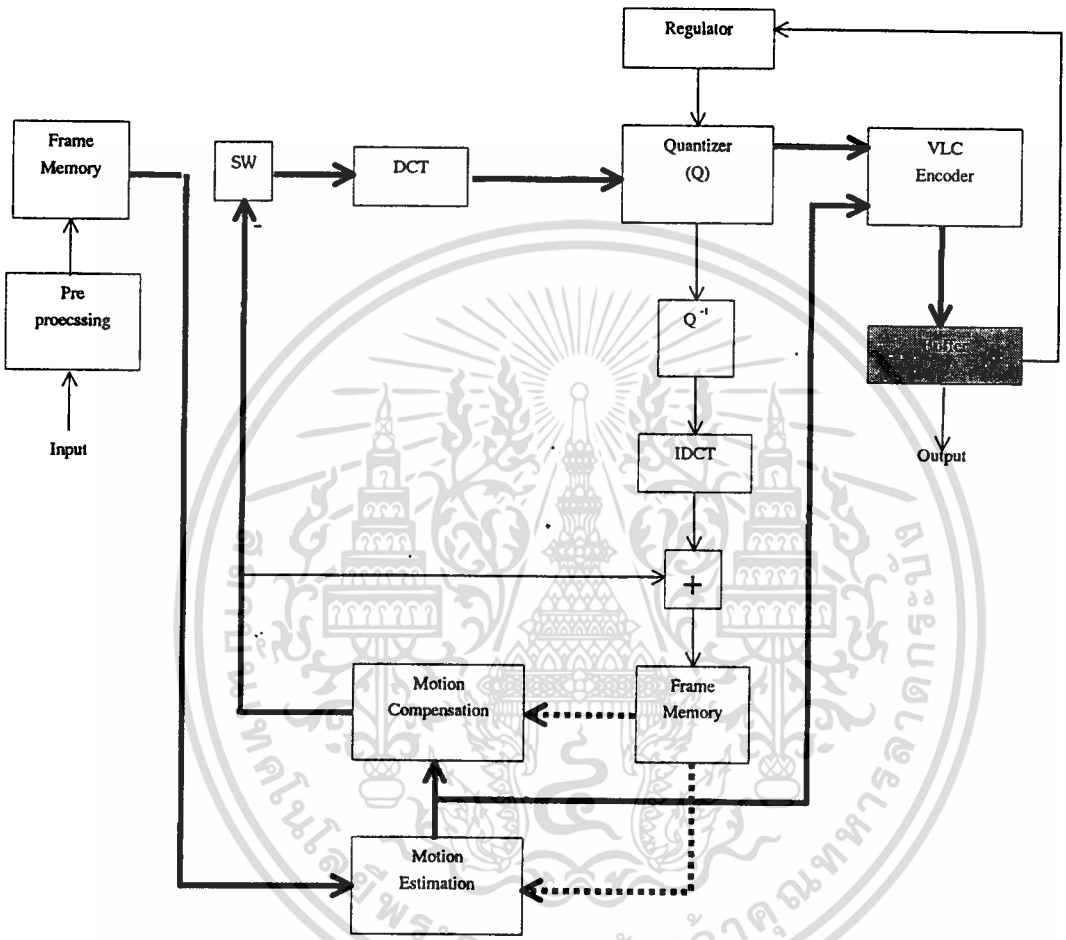
- Estimation คือ การประเมินหาแมคโครบล็อกของภาพ(search) จาก Frame memory ที่เหมือนกับ แมคโครบล็อกของภาพปัจจุบันมากที่สุดเมื่อได้แล้วก็สร้างเวคเตอร์ที่เรียกว่า motion vecrtor ซึ่งชี้จากจุดกำเนิดของบล็อกอดีตไปยังจุดกำเนิดของบล็อกปัจจุบัน ค่าของเวคเตอร์นี้จะถูกส่งไปยัง VLC เพื่อเก็บไว้เป็นรหัสเฉพาะของบล็อคนั้นๆ เพื่อใช้ตอนการถอดรหัส

- Compensation เมื่อเรามี motion vector แล้วเราไม่จำเป็นต้องนำข้อมูลที่ซ้ำกันกับส่วนอื่นของเฟรมเดิมส่งไปอีกเพราะสามารถใช้ motion vector ในการหาข้อมูลที่ต้องการนั้นได้ในการถอดรหัส การ compensate จะช่วยลดปริมาณข้อมูลให้น้อยลงที่สุด คือเราจะนำข้อมูลที่ผ่านไปแมคโครบล็อกที่เรา ทำการ search มาได้ ลบกับข้อมูลปัจจุบันเพื่อให้ได้ค่า prediction error ได้เป็นข้อมูลของภาพชนิด P ที่มีขนาดเล็กกว่าภาพชนิด I

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อทำขั้นตอนทั้งสองแล้วข้อมูล prediction error จะถูกเข้ากระบวนการ DCT , Quantiazing และ VLC ตามปรกติ

การเข้ารหัส B-pictures



รูปที่ 3.15 การประมวลผล Bidirectionally predicted picture

การดำเนินการกับ B-Pictures จะมีขั้นตอน motion compensation คล้ายกับภาพชนิด P จะต่างกันที่ไม่มีการนำเอาข้อมูลในขั้นตอน quantizing เข้ากระบวนการย้อนกลับมาเก็บไว้อีกเนื่องจากมันไม่ได้ถูกใช้ในขั้นตอนใดเหมือนกับภาพสองชนิดแรก

การทำ motion compensation ในภาพชนิด B นั้นใน frame memory จะต้องมีที่สำหรับเก็บข้อมูล 2 ชุด คือ ข้อมูลอดีตและข้อมูลอนาคต โดยจะเป็นการดำเนินการแบบสองทิศทางซึ่งจะได้ข้อมูล prediction error macroblock สามชุดคือ backward motion compensation ,forward motion compensation และ interporated macroblock (average) โดยการเลือกว่าจะเอา macroblockไหนเป็นข้อมูลส่งไปใช้วิธี คำนวณ macroblock ที่มีค่า mean square ต่างกับ macroblock ปัจจุบันน้อยที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

รูปแบบรหัสข้อมูล Video ตามมาตรฐาน MPEG - 1

4.1 บทนำ

จากการทำงานของระบบประมวลผลภาพเคลื่อนไหวตามมาตรฐาน MPEG ในบทที่ 3 นั้น เป็นเพียงโครงสร้างการทำงานของระบบเท่านั้น แต่รหัสข้อมูลจริงๆนั้น ถูกกำหนดไว้ใน ISO/IEC 11172-2 ชุดของ Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s ซึ่งจะแบ่งออกเป็น 4 ส่วนคือ Systems, Video, Audio และ Compliance testing ซึ่งจะทำให้ผู้ออกแบบระบบการประมวลผลภาพเคลื่อนไหว สามารถพัฒนาบนมาตรฐานเดียวกัน

4.2 Start code

Start code คือ รหัสที่ใช้เป็นรหัส ที่บอกการเริ่มของกระบวนการใดกระบวนการหนึ่ง ซึ่งในมาตรฐานได้กำหนดไว้ดังในตารางที่ 4.1

Table 4.1 System start codes are defined in ISO/IEC 11172-1

Name	Hexadecimal value
picture_start_code	00000100
slice_start_codes	00000101- 000001AF
reserved	000001B0 , 000001B1
user_data_start_code	000001B2
sequecnce_header_code	000001B3
sequecnce_error_code	000001B4
extension_start_code	000001B5
reserved	000001B6
sequence_end_code	000001B7
group_start_code	000001B8
system start codes	000001B9 - 000001FF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. 3 Layer ของรหัสข้อมูล

เนื่องจากข้อมูลที่ใช้ในการประมวลผลภาพเคลื่อนไหวมีจำนวนมาก ดังนั้นจะทำการแบ่งข้อมูลออกเป็นชั้นตามหน้าที่การทำงานเพื่อให้สามารถตรวจสอบได้ง่าย มีทั้งหมด 6 ชั้นคือ

4.3.1 Video layer

มี Syntax ของ bit stream ตามอัลกอริธึมดังนี้ คือ

Syntax	No. of bits	Mnemonic
video_sequence() {		
next_start_code()		
do{		
sequence_header()		
do {		
group_of_pictures()		
} while (nextbits() == group_start_code)		
} while (nextbits() == sequence_header_code)		
sequence_end_code	32	bslbf
}		

ในชั้น Video sequence นี้จะมี procedure ที่สำคัญคือ sequence header ซึ่งมีการทำงานดังนี้คือ

Syntax	No. of bits	Mnemonic
sequence_header() {		
sequence_header_code	32	bslbf
horizontal_size	12	uimsbf
vertical_size	12	uimsbf
pel_aspect_ratio	4	uimsbf
picture_rate	4	uimsbf
bit_rate	18	uimsbf
marker_bit	1	"1"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        vbv_buffer_size                                          10        uimsbf
        constrained_parameter_flag                               1
        load_intra_quantizer_matrix                              1
        if ( load_intra_quantizer_matrix)
                intra_quantizer_matrix [ ]                       8*64      uimsbf
        load_non_intra_quantizer_matrix                          1
        if ( load_non_intra_quantizer_matrix)
                non_intra_quantizer_matrix [ ]                   8*64      uimsbf
        if ( nextbits() == extension_start_code ) {
                extension_start_code                             32        bslbf
                while ( nextbits () != "0000 0000 0000 0000 0000 0001"){
                        sequence_extension_data                  8
                }
                next_start_code()
        }
        if ( nextbits() == extension_start_code ) {
                user_data_start_code                             32        bslbf
                while ( nextbits () != "0000 0000 0000 0000 0000 0001"){
                        user_data                                8
                }
                next_start_code()
        }

}
```

horizontal_size คือ ความกว้างของส่วนที่แสดงผลได้ของ Y ในหน่วยพิกเซล

vertical_size คือ ความสูงของส่วนที่แสดงผลได้ของ Y หน่วยพิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

pel_aspect_ratio คือ ค่าที่แสดงอัตราส่วนของความสูงต่อความกว้างอ่านค่าได้ตามตารางที่ 4.2

pel_aspect_ratio	height/width	example
0000	forbidden	
0001	1.0000	VGA etc.
0010	0.6735	
0011	0.7031	16:9, 625 line
0100	0.7615	
0101	0.8055	
0110	0.8437	16:9 ,525 line
0111	0.8935	
1000	0.9157	CCIR601, 625 line
1001	0.9815	
1010	1.0255	
1011	1.0697	
1100	1.0950	
1101	1.1575	CCIR601, 525line
1110	1.2015	
1111	reserved	

ตารางที่ 4.2 รหัสข้อมูลของ pel_aspect_ratio

picture_rate คือ อัตราของภาพต่อนาที

picture_rate	picture per second
0000	forbidden
0001	23.976
0010	24
0011	25
0100	29.97
0101	30

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

0110	50
0111	59.94
1000	60
.....	reserved
1111	reserved

ตารางที่ 4.3 รหัสข้อมูลของ picture_rate

bit_rate คือค่าจำนวนเต็มซึ่งกำหนดอัตราความเร็วของบิตในหน่วยของ 400 bits/s(ห้ามเป็น 0)

marker_bit คือ บิตจำนวนหนึ่งบิตถูกเซตให้เป็น "1"

vbv_buffer_size คือ ค่าจำนวนเต็มขนาด 10 บิต ใช้กำหนดขาดของ Video Buffering Verifier โดยค่าขนาดของ VBV ต่ำสุดที่ยังสามารถใช้ถอดรหัสได้คือ 16*1024*vbv_buffer_size

constrained_parameters_flag เป็นแฟลกที่ถ้าถูกเซตเป็น "1" ค่าต่างๆจะถูกบังคับให้เป็น

horizontal_size <= 768 pels

vertical_size <= 576 pels

((horizontal_size+15)/16)*((vertical_size+15)/16)< = 396

((horizontal_size+15)/16)*((vertical_size+15)/16)*picture_rate <= 396

picture_rate <= 30 pictures/s

forward_f_code <= 4

backward_f_code <= 4

load_intra_quantizer_matrix ถ้าเป็น 1 ค่าที่ตามมาจะเป็น intra_quantizer_matrix ถ้าถูกเซตเป็น 0 จะใช้ค่า quantizer matrix แบบ default ตามรูปเมตริกข้างล่าง และจะใช้ไปจนกว่า จะถึง sequence ถัดไป

$$\begin{bmatrix} 8 & 16 & 19 & 22 & 26 & 27 & 29 & 34 \\ 16 & 16 & 22 & 24 & 27 & 29 & 34 & 37 \\ 19 & 22 & 26 & 27 & 29 & 34 & 34 & 38 \\ 22 & 22 & 26 & 27 & 29 & 34 & 37 & 40 \\ 22 & 26 & 27 & 29 & 32 & 35 & 40 & 48 \\ 26 & 27 & 29 & 32 & 35 & 40 & 48 & 58 \\ 26 & 27 & 29 & 34 & 38 & 46 & 56 & 69 \\ 27 & 29 & 35 & 38 & 46 & 56 & 69 & 83 \end{bmatrix}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

intra_quantizer_matrix เป็นข้อมูลจำนวน 64 กลุ่มของบิตจำนวน 8 บิต แบบจำนวนเต็มไม่มีเครื่องหมาย โดยจะสามารถใส่ลงในเมตริกในแบบ zigzag scanning order (ห้ามมีค่าใดเป็น 0) ค่าของ 8 บิตแรกต้องเป็น 8

load_non_intra_quantizer_matrix ถ้าเป็น1ค่าที่ตามมาจะเป็น non_ntra_quantizer_matrix ถ้าถูกเซตเป็น 0 จะใช้ค่า quantizer matrix แบบ default ตามรูปเมตริกข้างล่าง และจะใช้ไปจนกว่าจะถึง sequence ถัดไป

$$\begin{bmatrix} 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \\ 16 & 16 & 16 & 16 & 16 & 16 & 16 & 16 \end{bmatrix}$$

non_intra_quantizer_matrix เป็นข้อมูลจำนวน 64 กลุ่มของบิตจำนวน 8 บิต แบบจำนวนเต็มไม่มีเครื่องหมาย โดยจะสามารถใส่ลงในเมตริกในแบบ zigzag scanning order (ห้ามมีค่าใดเป็น 0)

4.3.2 Group of pictures layer

Syntax	No. of bits	Mnemonic
group_of_pictures() {		
group_start_code()	32	bslbf
time_code	25	
closed_gop	1	
broken_link	1	
next_start_code ()		
if (nextbits() == extension_start_code) {		
extension_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
group_extension_data	8	
}		
next_start_code ()		
}		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    if ( nextbits() == user_data_start_code ) {
        user_data_start_code                                    32          bslbf
        while ( nextbit() != '0000 0000 0000 0000 0000 0001') {
            user_data                                           8
        }
        next_start_code()
    }
    do {
        picture()
    } while ( nextbits() == picture_start_code)
}
```

time_code คือ ข้อมูลจำนวน 25 บิต ซึ่งจะถูกตามด้วย drop_frame_flag, time_code_hours, time_code_minutes, marker_bit, time_code_seconds และ time_code_pictures ซึ่งค่าเหล่านี้เป็นค่าที่สอดคล้องกับมาตรฐาน IEC ในเรื่อง "time and control codes for video tape recorders" ค่า drop_frame_flag จะเป็น "1" เมื่อ picture rate = 29.97 Hz เท่านั้น การนับจำนวนภาพจะตัดภาพที่ 0 และ 1 ของตอนเริ่มนาที ยกเว้นนาที ที่ 0, 10, 20, 30, 40, 50 แต่ถ้า drop_frame_flag ถูกเซตเป็นศูนย์มันจะปัดให้ picture rate เท่ากับจำนวนเต็มค่าที่ใกล้ที่สุด

time_code	range of value	bits	
drop_frame_flag		1	
time_code_hours	0 - 23	5	uimsbf
time_code_minutes	0 - 59	6	uimsbf
marker_bit	1	1	"1"
time_code_seconds	0 - 59	6	uimsbf
time_code_pictures	0 - 59	6	uimsbf

ตารางที่ 4.4 รหัสข้อมูลหลัง time_code

close_gop ถ้าถูกเซตเป็น "1" แสดงว่า GOP นี้ถูกเข้ารหัสโดยไม่ได้ใช้ การชี้ของ motion vector จาก GOP ก่อนหน้านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

broken_link ควรจะถูกเซตเป็น "0" เสมอ แต่ถ้าเซตเป็นหนึ่งจะแสดงว่าไม่สามารถถอดรหัส B-pictures ได้เนื่องมาจากภาพที่จะใช้ทำนายชนิด I หรือ P ไม่มีอยู่

4.3.3 Picture layer

Syntax	No. of bits	Mnemonic
pictures() {		
picture_start_code()	32	bslbf
temporal_reference	10	uimsbf
picture_coding_type	3	uimsbf
vbv_delay	16	uimsbf
if (picture_coding_type == 2) || (picture_coding_type == 3))		
{		
full_pel_forward_vector	1	
forward_f_code	3	uimsbff
}		
if (picture_coding_type == 3) {	8	
extra_bit_picture	1	"1"
extra_information_picture	8	
}		
extra_bit_picture	1	"0"
next_start_code()		
if (nextbits() == extension_start_code) {		
extension_start_code	32	bslbf
while (nextbit() != '0000 0000 0000 0000 0000 0001') {		
picture_extention_data	8	
}		
next_start_code()		
}		
if (nextbits() == user_data_start_code) {		
user_data_start_code	32	bslbf

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        while ( nextbit() != '0000 0000 0000 0000 0000 0001') {
            user_data                                          8
        }
        next_start_code()
    }
    do {
        slice()
    } while ( nextbits() == slice_start_code)
}
```

temporal_reference เป็นค่าที่ใช้บอกจำนวนของภาพ โดยจะเพิ่มขึ้นหนึ่งครั้งต่อหนึ่งภาพในแบบ modulo 1024 ภาพแรกของ GOP ให้ค่าเป็น 0

picture_coding_type เป็นค่าที่บอกชนิดของภาพดังตารางที่ 4.5

picture_coding_type	coding method
000	forbidden
001	intra-coded (I)
010	predictive-coded (P)
011	bidirectionally-predictive-coded (B)
100	dc intra-coded (D)
101	reserved
....	....
11	reserved

ตารางที่ 4.5 รหัสข้อมูลของ picture_coding_type

vbv_delay เป็นค่าจำนวนเต็มแบบไม่มีเครื่องหมาย ขนาด 16 บิต สำหรับการทำงานแบบบิตเรตคงที่ มันจะถูกใช้สำหรับตั้งค่าเริ่มต้น บัฟเฟอร์ของตัวถอดรหัส เมื่อเริ่มการถอดรหัสภาพเพื่อให้บัฟเฟอร์ไม่เกิดการ Overflow หรือ Underflow โดยจะทำการคำนวณเวลาที่จะต้องเติม VBV buffer หลังจากที่ว่างในช่วงเริ่มต้นตั้งให้บิตเรตเท่ากับ R เพื่อให้สามารถแก้ไขได้ในทันทีก่อนที่ภาพขณะนั้นถูกลบไปจากบัฟเฟอร์

ค่าของ vbv_delay จะเป็นค่าของ ช่วงเวลาของ clock 90 kHz เพื่อรอหลังจากรับ ไบต์สุดท้ายของ picture_start_code โดยสามารถคำนวณได้จาก

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$vbv\_delay_n = 90000*B_n / R$

$n > 0$

B_n = VBV occupancy ในหน่วย bits

R = bitrate ในหน่วย bits/s

สำหรับการทำงานแบบบิตเรตไม่คงที่ vbv_delay จะถูกตั้งให้เป็นค่า $FFFF_H$

full_pel_forward_vector ถ้าเป็น 1 แสดงว่าค่าของ forward motion vector เป็นจำนวนเต็ม

forward_f_code เป็นตัวที่นำไปใช้ในการหา forward motion vector เป็นจำนวนเต็มค่า 1 ถึง 7

full_pel_forward_vector ถ้าเป็น 1 แสดงว่าค่าของ backward motion vector เป็นจำนวนเต็ม

backward_f_code เป็นตัวที่นำไปใช้ในการหา backward motion vector เป็นจำนวนเต็มค่า 1 ถึง 7

extra_bit_picture ถ้าเป็น 1 แสดงว่าจะมีข้อมูล extra information ตามมา

4.3.4 Slice layer

Syntax	No. of bits	Mnemonic
slice() {		
slice_start_code()	32	bslbf
quantizer_scale	5	uimsbf
while (nextbits() == '1'() {		
extra_bit_slice	1	"1"
extra_information_slice	8	
}		
extra_bit_slice	1	"0"
do {		
macroblock()		
} while (nextbits() != '000 0000 0000 0000 0000 0000')		
next_start_code()		
}		

slice_start_code เป็นข้อมูล 32 บิต แสดงการเริ่มต้นของชั้น slice โดยมี 24 บิตแรกเป็น 000001H และ 8 บิตหลังเป็น ตำแหน่งของ slice ค่าตั้งแต่ 01H-AFH

เอกสารนี้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

slice_vertical_position เป็นค่า 8 บิตสุดท้ายของ slice_start_code โดยที่ slice แรกมีค่า เป็น 1 อาจมีหลายslice ที่มีค่าเดียวกัน ซึ่งแสดงว่าขนาดของ sliceไม่จำเป็นต้องเท่ากัน อาจจะเริ่มที่ ตำแหน่ง vertical หนึ่ง และไปจบที่อีก vertical หนึ่ง แต่ที่แน่นอนคือ macroblock แรกของภาพจะอยู่ใน slice แรก และ macroblock สุดท้ายของภาพจะอยู่ใน slice สุดท้าย

quantizer_scale เป็นค่าจำนวนเต็ม ตั้งแต่ 1-31 สำหรับคูณกับค่า DCT coefficient ที่ได้จาก stream ข้อมูล ค่านี้จะถูกใช้จนกว่าจะรับมาได้ใหม่

extra_bit_slice ถ้าเป็น 1 จะมีข้อมูล extra_information_slice ตามมา

4.3.5 Macroblock layer

Syntax	No. of bits	Mnemonic
macroblock() {		
while (nextbits() == '0000 0001 111')		
macroblock_stuffing	11	vlclbf
while (nextbits() == '0000 0001 000')		
macroblick_escape	11	vlclbf
macroblock_address_increment	1-11	vlclbf
macroblock_type	1- 6	vlclbf
if (macroblock_quant)		
quantizer_scale	5	uimsbf
if (macroblock_motion_forward) {		
motion_horizontal_forward_code	1- 11	vlcbf
if ((forward_f != 1) &&		
(motion_horizontal_forward_code != 0)		
)		
motion_horizontal_forward_r	1-6	uimsbf
motion_vertical_forward_code	1-11	vlclbf
if ((forward_f != 1) &&		
(motion_vertical_forward_code != 0))		
motion_vertical_forward_r	1- 6	uimsbf
}		
if (macroblock_motion_backward) {		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        motion_horizontal_backward_code                    1- 11        vlclbf
        if ( (backward_f != 1 ) &&
                        (motion_horizontal_backward_code != 0) )
            motion_horizontal_backward_r                   1 - 6        uimsbf
        motion_vertical_back_code                          1- 11        vlclbf
        if ( (backward_f != 1 ) &&
                        (motion_vertical_backward_code != 0) )
            motion_vertical_forward_r                      1- 6         uimsbf
    }
    if ( macroblock_pattern )
        code_block_pattern                                 3 - 9        vlclbf
    for ( i=0 ; i<6 ; i++ )
        block( i )
    if ( picture_coding_type == 4 )
        end_of_macroblock                                  1            "1"
}
```

macroblock_stuffing เป็นค่าคงที่ "0000 0001 111" ใส่โดย encoder เพื่อเพิ่มบิตเรต

macroblock_escape เป็นค่าคงที่ "000 0001 000" เพื่อเป็นการบอกค่าในกรณี macroblock address increment มีค่าเกิน 33 โดยจะนำ 33 มาคูณด้วยจำนวนของ macroblock_escape แล้วบวกเข้ากับค่าที่ให้มา

macroblock_address_increment เป็นค่า vlc บอกค่าความแตกต่างระหว่างตำแหน่งของมันกับแมคโครบล็อกก่อนหน้า

macroblock_type เป็นค่าแสดงชนิดของ macroblock

quantizer_scale เป็นค่าเดียวกับที่อยู่ในชั้น slice จะเปลี่ยนเมื่อได้รับค่าใหม่เข้ามา

motion_horizontal_forward_code เป็นค่า vlc ที่บอกว่ามี motion_horizontal_forward_r อยู่ใน stream หรือไม่

motion_horizontal_forward_r เป็นจำนวนเต็มไม่มีเครื่องหมายใช้สำหรับการหา forward motion vector

motion_vertical_forward_code เป็นค่า vlc ที่บอกว่ามี motion_vertical_forward_r อยู่ใน stream หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

motion_vertical_forward_r เป็นค่าจำนวนเต็มไม่มีเครื่องหมายใช้สำหรับการหา forward motion vector

motion_horizontal_backward_code เป็นค่า vlc ที่บอกว่ามี motion_horizontal_backward_r อยู่ใน stream หรือไม่

motion_horizontal_backward_r เป็นค่าจำนวนเต็มไม่มีเครื่องหมายใช้สำหรับการหา backward motion vector

motion_vertical_backward_code เป็นค่า vlc ที่บอกว่ามี motion_vertical_backward_r อยู่ใน stream หรือไม่

motion_vertical_backward_r เป็นค่าจำนวนเต็มไม่มีเครื่องหมายใช้สำหรับการหา backward motion vector

code_block_pattern เป็นค่า vlc บอกว่ามีการรับ block ตำแหน่งใดเข้ามาหรือว่าเป็น skipped block

4.3.6 Block layer

Syntax	No. of bits	Mnemonic
block(i) {		
if (pattern_code[i]) {		
if (macroblock_intra) {		
if (i < 4) {		
dct_dc_size_luminance	2 - 7	vlclbf
if (dc_size_luminance != 0)		
dct_dc_differential	1 - 8	uimsbf
}		
else {		
dct_dc_size_chrominance	2 - 8	vlclbf
}		
}		
else {		
dct_coeff_first	2 - 28	vlclbf
}		
if (picture_coding_type != 4) {		
while (nextbits() != '10')		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

dct_coeff_next	3 - 28	vlclbf
end_of_block	2	vlcbf
}		
}		
}		

dct_dc_size_luminance เป็นจำนวนบิตของ ค่า dct_dc_differential ของ component Y ที่ตามมา

dct_dc_size_chrominance เป็นจำนวนบิตของค่า dct_dc_differential ของ component cb และ cr ที่ตามมา

dct_dc_differential เป็นค่า vlc จะไม่มีใน stream ข้อมูลถ้า dct_dc_size ทั้งสองเป็น 0

dct_coeff_first เป็นค่า vlc ที่เป็น coefficient ตัวแรกของ block

dct_coeff_next เป็นค่า vlc ของ coefficient ตัวถัดไปของ block

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

การทดลองและผลการถอดรหัส MPEG

5.1 การใช้โปรแกรม Mathcad คำนวณ Discrete Cosine Transform

ขั้นตอนการทำงาน

- ให้ข้อมูลที่จะทำการทดสอบเป็น matrix ขนาด 8 x 8 เก็บไว้ในตัวแปล X
- ให้ข้อมูล Quantization เก็บไว้ใน matrix q

$$x := \begin{bmatrix} 168 & 161 & 161 & 150 & 154 & 168 & 164 & 154 \\ 171 & 154 & 161 & 150 & 157 & 171 & 150 & 164 \\ 171 & 168 & 147 & 164 & 164 & 161 & 143 & 154 \\ 164 & 171 & 154 & 161 & 157 & 157 & 147 & 132 \\ 161 & 161 & 157 & 154 & 143 & 161 & 154 & 132 \\ 164 & 161 & 161 & 154 & 150 & 157 & 154 & 140 \\ 161 & 168 & 157 & 154 & 161 & 140 & 140 & 132 \\ 154 & 161 & 157 & 150 & 140 & 132 & 136 & 128 \end{bmatrix} \quad Q := \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

- ทำการหักออกให้ค่าเฉลี่ยเป็น 0 โดยการลบด้วย 128
- กำหนดค่าสัมประสิทธิ์ของ DCT ตามเงื่อนไขทฤษฎีของ DCT

$$X := X - 12$$

$$C(N) := \begin{cases} \frac{1}{\sqrt{2}} & \text{IF } N= \\ 1 & \text{OTHERWIS} \end{cases}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สร้างโปรแกรมการคำนวณ DCT ตามวิธีของ Mathcad เก็บค่าที่คำนวณได้ไว้ในเมตริก y

$$
Y := \left|\begin{array}{l}
M \leftarrow 0 \\
\text{FOR } K \in 0..7 \\
\quad \text{FOR } L \in 0..7 \\
\quad\quad S \leftarrow 0 \\
\quad\quad \text{FOR } I \in 0..7 \\
\quad\quad\quad \text{FOR } J \in 0..7 \\
\quad\quad\quad\quad S \leftarrow S + \frac{C(K)\cdot C(L)}{4}\cdot X_{I,J}\cdot \cos\left[\frac{(2\cdot I+1)\cdot K\cdot \pi}{16}\right]\cdot \cos\left[\frac{(2\cdot J+1)\cdot L\cdot \pi}{16}\right] \\
\quad\quad M_{K,L} \leftarrow S \\
M
\end{array}\right.
$$

$$
Y = \begin{bmatrix}
214.5 & 49.224 & -2.732 & 19.661 & -10.25 & -0.911 & 0.59 & -6.083 \\
34.441 & -24.953 & 10.573 & 12.737 & 4.774 & -2.977 & 14.909 & -5.873 \\
-6.061 & -3.585 & 7.855 & -9.172 & 2.618 & -2.638 & 5.014 & 9.996 \\
8.351 & -10.394 & 4.408 & 3.699 & -15.141 & 10.135 & 5.563 & 6.38 \\
-12.5 & 4.988 & -1.412 & -1.565 & -14.75 & 9.304 & -5.368 & -1.409 \\
4.721 & 9.396 & -7.653 & 2.604 & 4.255 & -6.752 & -14.283 & 2 \\
1.699 & -1.763 & 2.764 & -0.868 & 1.082 & 2.811 & -3.105 & -4.197 \\
-1.087 & 0.597 & -0.355 & 1.583 & 3.04 & -1.539 & -4.073 & -2.494
\end{bmatrix}
$$

- ทำการปัดเศษไปเป็นจำนวนเต็ม

$$
Y := \begin{bmatrix}
214 & 49 & -3 & 20 & -10 & -1 & 1 & -6 \\
34 & -25 & 11 & 13 & 5 & -3 & 15 & -6 \\
-6 & -4 & 8 & -9 & 3 & -3 & 5 & 10 \\
8 & -10 & 4 & 4 & -15 & 10 & 6 & 6 \\
-12 & 5 & -1 & -2 & -15 & 9 & -5 & -1 \\
5 & 9 & -8 & 3 & 4 & -7 & -14 & 2 \\
2 & -2 & 3 & -1 & 1 & 3 & -3 & -4 \\
-1 & 1 & 0 & 2 & 3 & -2 & -4 & -2
\end{bmatrix}
$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการควอนไตซ์ ค่าของเมตริก y ได้ เมตริก z ที่ทำการควอนไตซ์แล้ว

$$Z = \left| \begin{array}{l} \text{FOR } K \in 0..7 \\ \quad \text{FOR } L \in 0..7 \\ \quad\quad S \leftarrow \text{FLOOR}\left(\dfrac{Y_{K,L} + \text{FLOOR}\left(\dfrac{Q_{K,L}}{2}\right)}{Q_{K,L}}\right) \\ \quad\quad M_{K,L} \leftarrow S \\ M \end{array} \right.$$

$$Z = \begin{bmatrix} 13 & 4 & 0 & 1 & 0 & 0 & 0 & 0 \\ 3 & -2 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

ดังนั้นข้อมูลที่ได้มาจะเป็นข้อมูลที่ทำการบีบอัดด้วยวิธีการ DCT และ Quantizing ซึ่งมีปริมาณข้อมูลน้อยลงและมีศูนย์อยู่จำนวนมากซึ่งสามารถตัดทิ้งได้เมื่อทำการเข้ารหัสข้อมูลลงใน stream

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การใช้โปรแกรม Mathcad คำนวณ Inverse Discrete Cosine Transform

ขั้นตอนการทำงาน

- ทำการ Dequantizing ตามสูตรในบทที่ 2 เก็บค่าที่ได้ไว้ใน zo

```
ZO := | FOR K ∈ 0..7
      |   FOR L∈ 0..7
      |     S← Z_{K,L}·Q_{K,}
      |     M_{K,L}← S
      | M
```

$$ZO = \begin{bmatrix} 208 & 44 & 0 & 16 & 0 & 0 & 0 & 0 \\ 36 & -24 & 14 & 19 & 0 & 0 & 0 & 0 \\ 0 & 0 & 16 & 0 & 0 & 0 & 0 & 0 \\ 14 & -17 & 0 & 0 & 0 & 0 & 0 & 0 \\ -18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- ทำ IDCT ตามสูตรในบทที่ 2 เก็บค่าที่ได้ไว้ใน xo

```
XO := | M ← 0
      | FOR I∈ 0..7
      |   FOR J∈ 0..7
      |     S← 0
      |     FOR K ∈ 0..7
      |       FOR L∈ 0..7
      |         S← S + (C(K)·C(L))/4 · ZO_{K,L} · COS[((2·I + 1)·K·π)/16] · COS[((2·J + 1)·L·π)/16]
      |     M_{I,J}← S
      | M
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$XO = \begin{bmatrix} 43.252 & 31.953 & 21.067 & 20.984 & 29.943 & 37.576 & 37.602 & 34.019 \\ 46.346 & 36.444 & 26.722 & 25.896 & 32.015 & 35.997 & 33.072 & 27.977 \\ 42.734 & 35.612 & 28.623 & 27.619 & 30.333 & 29.777 & 23.422 & 16.747 \\ 32.752 & 29.188 & 26.069 & 26.063 & 26.736 & 23.47 & 15.739 & 8.914 \\ 27.881 & 27.314 & 27.354 & 28.157 & 27.7 & 23.764 & 17.102 & 11.796 \\ 30.898 & 31.548 & 32.143 & 31.602 & 29.103 & 24.8 & 20.076 & 16.964 \\ 32.943 & 33.058 & 31.701 & 27.668 & 21.764 & 16.237 & 12.769 & 11.391 \\ 31.038 & 30.207 & 26.699 & 19.58 & 10.836 & 4.001 & 0.896 & 0.349 \end{bmatrix}.$$

- ค่าที่ได้จะต้องบวกด้วย 128 จึงจะได้ข้อมูลเดิมกลับคืนมา

$$XO + 128 = \begin{bmatrix} 171.252 & 159.953 & 149.067 & 148.984 & 157.943 & 165.576 & 165.602 & 162.019 \\ 174.346 & 164.444 & 154.722 & 153.896 & 160.015 & 163.997 & 161.072 & 155.977 \\ 170.734 & 163.612 & 156.623 & 155.619 & 158.333 & 157.777 & 151.422 & 144.747 \\ 160.752 & 157.188 & 154.069 & 154.063 & 154.736 & 151.47 & 143.739 & 136.914 \\ 155.881 & 155.314 & 155.354 & 156.157 & 155.7 & 151.764 & 145.102 & 139.796 \\ 158.898 & 159.548 & 160.143 & 159.602 & 157.103 & 152.8 & 148.076 & 144.964 \\ 160.943 & 161.058 & 159.701 & 155.668 & 149.764 & 144.237 & 140.769 & 139.391 \\ 159.038 & 158.207 & 154.699 & 147.58 & 138.836 & 132.001 & 128.896 & 128.349 \end{bmatrix}.$$

- เปรียบเทียบกับข้อมูลเดิมคือ x

$$x := \begin{bmatrix} 168 & 161 & 161 & 150 & 154 & 168 & 164 & 154 \\ 171 & 154 & 161 & 150 & 157 & 171 & 150 & 164 \\ 171 & 168 & 147 & 164 & 164 & 161 & 143 & 154 \\ 164 & 171 & 154 & 161 & 157 & 157 & 147 & 132 \\ 161 & 161 & 157 & 154 & 143 & 161 & 154 & 132 \\ 164 & 161 & 161 & 154 & 150 & 157 & 154 & 140 \\ 161 & 168 & 157 & 154 & 161 & 140 & 140 & 132 \\ 154 & 161 & 157 & 150 & 140 & 132 & 136 & 128 \end{bmatrix}$$

สรุป ข้อมูลหลังจากการเข้ารหัสและถอดรหัสได้ค่าที่ใกล้เคียงข้อมูลเดิม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ตัวอย่างการทดสอบข้อมูลจาก ไฟล์ MPEG-1

000000000000000001000004B26C000000000000001011A87E1F4A528740EA3E47B94A5206A
2062018006E529487D0D8780737294A43E86C042E529487DA19026E52948DF4300B6E52948
E9C382E36D7294A40DD0D8C235CA5290370E3C54360217294A40D40A86C2AE529487C02
C86D7294A4383000C7460217294A40D1CC14D1B6B94A52068A8EDAE529481B8700710DB
DCA5290370E850E0AD7294A40DC3A022E529481B874322E529487C3A1B1C0B7FF7294A4
0D8198085CA5290FA1B2964EB94A5261BD5D530D247CC1E1F000000001020BF87D294A33
59B2B5CA529251B3F855CA5291C600FE19624DB94A526104586DC0DDCA5292081E06A2
E52948C321AE5294854300D2E52948501186101772 94A43EED0D7294A43E36182EE529498
442855CA5290111B60237294A43E0370D87DCA5291FE1401E4318F7294A42B430B235CA5
291D8505C2C4DCA5290D0D7294A42A1977294A43A0742AE52948740118019DCA5290F85
C329AE529480D6E8EB986E08E1C448000001030BF87D294A3A0D7D47DC7DCA5292C801
FC2CEB94A520665C3705EFFB94A5263342C4855CA5290337836E5294819A0F1C46B94A52
0661E4F03B1F295EE529490C639E17CDB94A520663A1AE52948198D86E046E529486D0D7
294A4DB46B94A520959C4C300F80DDCA5292C522143EE52949B633C30BB94A524330F86
E2EE52949E4430BD7294A4863AA1B8F70FB94A52063C37177294A40C705EFF8016DCA52
90C8E0330B86582F7FDCA52930D4619FB874C37047E03380000001040BF87D294A0671E46
B3599EE529482571F3CF7294A4129273C04DCA5290C9150E9B94A52062407C4081BC7007
7AE5294864C2C17806E5294819FB1B0FEE5294819C08C00CEE529490CE3E163EE529498C
48F2091A1D22EE52948E5247E0F809B94A5232130E9B94A521BC0482F7FDCA5291D1E1C
17C06E529486F0B021EE52948673A1FDCA52903123FAE162C2EE52948E8E3E1606C0D5C
A5290C91F0E977B94A5206247C04B9F7294A40C4C6E90237294A417B8CF30DC919DF8707
C000001050BF87D294A06A408598D157294A43F9FFAE52948581187E65CA529013D970E03
EB94A5202701AF0E22E529487480FE327B01BB94A5206A417C0606FB94A521D3DC443FF
7294A43F05C0181B7294A42E1EE529487F0FEE5294809E6F87F7294A404C3FEE75CA52927
4B8F81BC513EE52949B963A06F1D7294A43A4193FE06FD7294A471B0EE7DCA5290E9876
1D7294A404B8307FC0DB94A5238D05F0181B7294A404C3FB94A5261BD3D1D662E61BD2
33E300EC0000001060BF87D294A3A4559B8AB94A52068A1337C03EB94A52096289D0F9F7
3EE52948C93202017BFEE529482580FE1F0FB94A524379BA2E52949B9F0F7294A464F8761
D7294A40DE380E438AB94A52096E7E0F8FCCB94A5231672A157294A40D06EFF874050BB
94A520681032330F6B94A5206F05F01850FB94A5206F87C26E529481BC178061C07F7294A

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ห้ามนำไปใช้ในการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

462F8E87D7294A412C183FE14F7294A4748F05BFF8FF5CA5290370E87D7294A40D05FFF8
7C203EE52949969E3393E61A908F8411E000001070BF87D294A1F0F2759B6EB7B94A5206
ED361789F7294A43E05EFF86C3C8A47D7294A40DC03E86CB7235CA5290350389D1F601F
DCA52903510D87DCA52927D0D8793EE52949BE1E4E0370D7294A43E021A1B10157294A4
64ED184DCA52903503A14046E529487C03E8F7294A40DDD6A0A8FB5CA52903510D879C
6BDCA5290370B86C3EE52948DF42AE529481B80841C6DCA5290FA14E67B94A5206E1D0
1C01C5CA5290370E86B94A524350389D1DAE5294A8B4EE8ED32D895C781D8000001080B
F87D294A19006ACDB19C9F7294A4866620CC60BF7294A4320CFA067B94A5236C2A1BB0
1DB94A521B2E1B82F80DCA5290C8C3E1BDCA5290C547B94A521900BE030C0D9FF7294
A474207C300FDEE529498C40F786177294A48620505430BB94A526DA1BDCA5290D807D0
CD7294A4361C4F8603B7294A486686016DCA52936468FB5CA52903110D808DCA52926D0
C2FDCA5293640A8601F2AE529486C04606010B94A521B435CA52930D6C8F30D619C7A88
F0000001090BF87D294A0651F00D6E5294819A3EB94A520942E17EE5294962A1B81B03772
94A412B8F8EE2AE529498C85B63AE529482538742D77294A4860B2342EE52948E8380E912
17EE529498C86E2AE529490C7348F0BB94A5264AE3E1B811B94A524338F86F7294A4C95C
79061B811B94A520951F25EE529490C86FEE529498CD01D7294A40CD01D7294A41285F86
3CFB94A520942E1BDCA5290305C371F7294A406A0239B32F72570E23C0000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000010A1A87E1F4A528D920B80343FB94A521B9
F0E91D7294A4370A016C3F05B01B94A521B80D21C06444B94A521B81187E0BFFF7294A4
327ADE1F8BB94A521930FDEE52948648FDA1FDCA5290C9F0E9B94A521924487FFDCA52
90B170FCDC6DCA5290DE3DCA5290313CD87581DB94A5206247F8748BB94A5206261FD
CA5290C901743FB94A52193C7807B0137294A40C4E3CD87E6DCA5290C9D1BFB94A521B
C2C0DDCA5290338F85811B94A5261A848CE7CC37719CF80000010B0BF87D294B3DD366
EA0397294A40D48998D32E52948C94EE0761C53DCA52903523A068AB94A527FF65C3E1D
7294A43F87B94A5243781181AAB94A5206F01F43B8BB94A523F4C3DCA52931A981BEE52
9490D48E0A8759D7294A4C6F81AA36E529481BC740DD80D5CA5290E90230370E27DCA5
290FE068AB94A521D2C7C6F7294A43F05EFF87B94A5206A61F82FFFDCA5290E98FC51C0
2FB94A523A531D22AE529481BE1C513AE5294986FE33CC37847C03380000010C0BF87D2
94A0680FE857D7294A412CF3C4C285912E529481BC0840EB94A5206888062FFB94A52435
CCD0A7EAB94A5264B14468FF0BB94A5232733B43ED7294A474EE642AE522A02617A00F

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

FFFFFFFFFFFFFFFFFF00301250020801620F0801620F000001BA2100033EC1801B9100000
1E00902FF0F9481A7284C6605FFFB94A523A60FB94A5209603E8380D5CA52904BE0462E5
294912DCC8B94A5264B150F7294A412F0E850EB94A52097817C0614A55CA5290374287807
37294A47F01F7294A40D8300E5CA5290378E27C055CA5290378B87157294A406A5D1DE61
BF0CE1C4780000010D0BF87D294A1F010B3FB7C65CA52920801FCDB01AB94A521F40C6
36E52948DF0F819BBAEE5294852A1B0FB94A526E81F803B86D7294A43E86DC0DDCA529
20801FC30B235CA5291FA2066B94A526E87786C22E5294874300D49F0336B94A52022060B
B94A521F00CE1AE529493A05435CA5290F9E1B6E75CA52935413AE52948FD03C9D0DB9
F7294A43A236C3EE529490446DAE52949BEFE1AE529492879163EB94A5261BF0CF32D89
5D603F800000010E0BF87D294A19CDB3FBB0FB94A521B301D9B7177294A40C443717729
4A412910D7294A40C40F27C7428FB94A524330F27E03B0305916E529481987C0CE3EE5294
8198793C0EC371F7294A4C661601CC3244DCA52903368DBDCA5290334300FAE52948F911
F705F01B94A5231481F0C03FB94A5259187C0C73793AE52948198793A185DCA5291D1879
3E067B94A5263340CE3C0EDCA5291B6130CE3EE52948188861372 94A4320D8FAE5294864
0F81803FB94A520351867EE2261A8647EEE711E00000010F0BF87D294A0671F67DC7DCA5
2904A016CDFAEE529490CE261FB9D7294A4C609507C3F375CA529031E3BDCA52904A8D
EE52948E88FFFB94A520628F8B94A52095C7C2C5DCA529219C7C2F2EE52949928E8581A
B94A52067171FF177294A40C8B94A5209501C4FB94A5209403F8712A016DCA52904A17E1
665CA52904AE3E006B7294A43787F7294A40CF0FC05F7294A40CF0FC5DCA52904A1F0EF
7294A4C350C87986E08FDC74000001100BF87D294A1BF349F46E6116E52948091FE0680D2
E5294809150FC4875CA5290120461EE529480987F7294A404F2243FB94A5217E1C51F7294A
43261C897294A42C07E41836E52948EE2C2E1F80FAE5294864AE3A7A8817294A46C91F1B
AC177FEE52948648B87E388B7294A42E068F55CA529012B87B94A5238D501D87B94A5216
418751983EE52948093D7002EB94A520247C2DC0E5CA5290122A1FDCA529005C07E07637
FAE52949868D6EE0B60330DE9E8E380EC000001110BF87D294A3A785D0A36E529481BC7
43A2E5294825F1D0F8817294A40DF0F7294A474F87D7294A40D4C0DC2AE529481BC711C
2A1C07F7294A40D4E1F0F84055CA529034570A8735805B7294A40D15A3A7730038B94A52
06F8DF011B94A5206F0204E87D7294A40DF0E2EE5294825A474388B94A5206A4711E37B9
4A5206F1DC8B0EA035EE52948E9E06D606E1F010F7294A40D111FE0357294A40D144F87
D7294A40DFE3FB94A521FC055CA52932D3C67986F08F8411F4000001120BF87D294A306E
1E4FD40D47D82705DCA529006503F13A09681FAE52948E1A8703B034A1F803BB94A5209
650F2340D287E00EAE5294803250FD034A1F82EE52948E1A81F82E1CA02782F5CA529006

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

503F13988D035004F5CA52904B40FCD03503F5CA529006503F05C0D401327DCA5290064A
1F04B287EB94A5200C942A09650FB94A5200C943F4C6943F7B94A5200C95CDD21A50FD7
294A401940F27C0D2809DCA529006503F40D4703FAE5294803281FA06A07EB94A5200CA0
E7C17034A1E4FB94A52006E801F412CA1F7294A400DCA1504B287DCA5291C3503F412C
A1FAE52948032881A531FAE5294996947475E1C7E70A3C3E61A5A3A3EDC07F0000000000
00000000000000000000000000000000

ตัวอย่างข้างต้นเป็นตัวอย่างของการทดสอบหา slice_start_code ของภาพหนึ่งภาพซึ่งมีทั้งหมด 18 slice ซึ่งค่าที่วงสี่เหลี่ยมไว้มีดังนี้

00000100	picture_start_code
00000101	slice_vertical_position = 1
00000102	slice_vertical_position = 2
00000103	slice_vertical_position = 3
00000104	slice_vertical_position = 4
00000105	slice_vertical_position = 5
00000106	slice_vertical_position = 6
00000107	slice_vertical_position = 7
00000108	slice_vertical_position = 8
00000109	slice_vertical_position = 9
0000010A	slice_vertical_position = 10
0000010B	slice_vertical_position = 11
0000010C	slice_vertical_position = 12
0000010D	slice_vertical_position = 13
0000010E	slice_vertical_position = 14
0000010F	slice_vertical_position = 15
00000110	slice_vertical_position = 16
00000111	slice_vertical_position = 17
00000112	slice_vertical_position = 18

และจะสังเกตได้ว่าในระหว่าง slice ที่ 12 และ 13 จะมีการเริ่ม packet ใหม่โดยมี packet_start_code เป็น 000001BA ซึ่งเป็นส่วนที่ตัวโปรแกรมจะต้องแยกข้อมูลออกมาว่าเป็น packet ชนิดใดซึ่งมีที่สำคัญ 2 ชนิดคือ audio stream และ video stream

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.4 การถอดรหัส ไฟล์ MPEG-1

การถอดรหัสข้อมูลไฟล์ MPEG –1 กระทำโดยการอ่านไฟล์ข้อมูล MPEG บางส่วน ขึ้นมาเก็บไว้ในบัฟเฟอร์แล้วทำการประมวลผล พร้อมกัลการอ่านข้อมูลส่วนใหม่ขึ้นมาอีกเมื่อประมวลผลข้อมูลเดิมเสร็จแล้ว ในส่วนของการประมวลผลข้อมูลมีส่วนประกอบมากมายสามารถแบ่งตามประเภทของการทำงานได้เป็น Procedure ต่างๆคือ

- Main procedure เป็นตัวที่ดำเนินการอ่านและเก็บข้อมูลไปตามมาตรฐาน ISO/IEC 11172 – 2 มีดังนี้คือ iso11172_stream, pack, system, packet, video_seuence, sequence_header, group_of_pictures, picture, slice, macroblock, block, sub_pack, sub_packet

- Bit stream procedure เป็นส่วนย่อยที่ลงมืออ่านหรือเขียนข้อมูลในระดับล่างเพื่อส่งให้ตัว main procedure ก่อนนำไปเรียบเรียงต่อไป มีดังนี้คือ buffer, nextkeyword, nextbit, nextbits, next_start_code, sub_buffer, sub_nextkeyword, shift_left, shift_right

- Variable length code table procedure เป็นส่วนที่ใช้ถอดรหัสข้อมูลแบบ vlc ซึ่งเป็นตารางที่กำหนดไว้ให้โดย มาตรฐาน MPEG มีดังนี้คือ vlc_macroblock_address_increment, vlc_motion_code, vlc_code_block_pattern, vlc_dct_coeff, vlc_dct_dc_size_luminance, vlc_dct_dc_size_chrominance, vlc_dct_level_(2-11), escape_sequence, obtainvalue

- Processing procedure เป็นส่วนที่ใช้คำนวณค่าต่างตามมาตรฐาน MPEG เช่น DCT , IDCT Motion reconstruction ได้แก่ dequantize_ipicture, dequantize_nonipicture, block_search, motion_rebuild, idct

- Procedure อื่นๆที่ทำหน้าที่ประกอบการทำงาน เช่น keep_picture สำหรับประมวลภาพเก็บเป็นไฟล์ลง disk หรือ slice_start_code สำหรับตรวจสอบว่าข้อมูล slice_start_code ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.5 เฟรมของภาพที่ได้จากการถอดรหัสด้วยโปรแกรมภาษา C

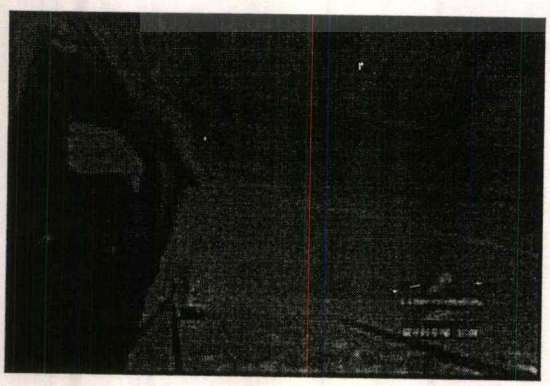
ในการทดสอบจะใช้ไฟล์ Test.mpg ซึ่งเป็นไฟล์ Demo ของโปรแกรม Xing MPEG Player เป็นภาพวีดีโอขนาด 352 x 240 มีทั้งหมด 10 GOPโดยจะนำมาแสดงเพียง 2 GOPโดยเราจะสังเกตได้ว่าภาพที่อยู่ใกล้เคียงกันจะมีความแต่งต่างกันน้อยมาก ดังนั้นภาพในแต่ละGOP จะมีความสัมพันธ์กันเพื่อการลดข้อมูลจะอาศัย motion vector เป็นตัวเชื่อมโยงภาพแต่ละชนิด และพบว่าภาพชนิด P และชนิด B จะมีความคมชัดน้อยกว่าภาพชนิด I โดยเฉพาะเมื่อทำการเปิดภาพให้มีขนาดใหญ่กว่าขนาดพิกเซลที่ให้มาเช่นในกรณีเปิดเป็น Full screen แต่ error เหล่านั้นเมื่อนำมาฉายต่อกันด้วยความเร็วแล้ว จะไม่สังเกตเห็นในส่วนรายละเอียด

GOP 1 มี 13 ภาพ

ภาพเมื่อเรียงตามลำดับการแสดงผลแล้วได้เป็น IBBPBBPBBPBBP

Picture 1

Type : I - Picture



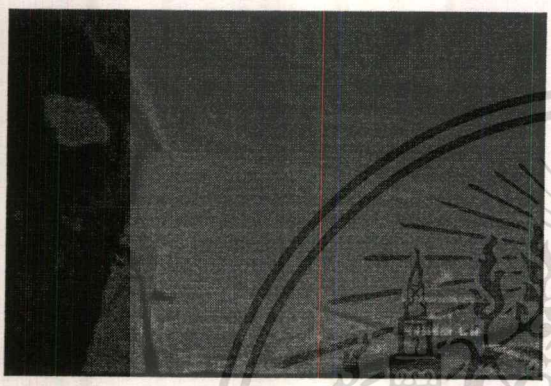
Picture 2

Type : B - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

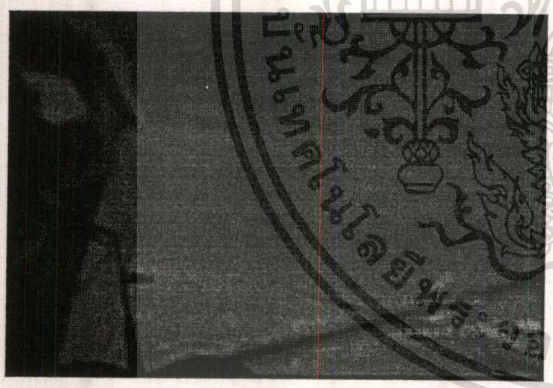
Picture 3

Type : B - Picture



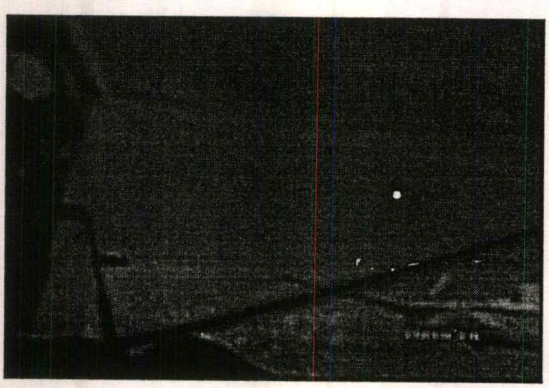
Picture 4

Type : P - Picture



Picture 5

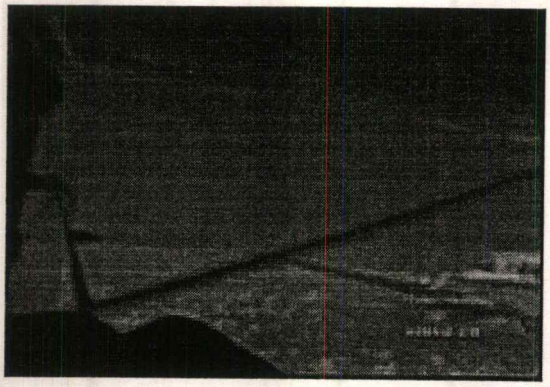
Type : B - Picture



Picture 6

Type : B - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

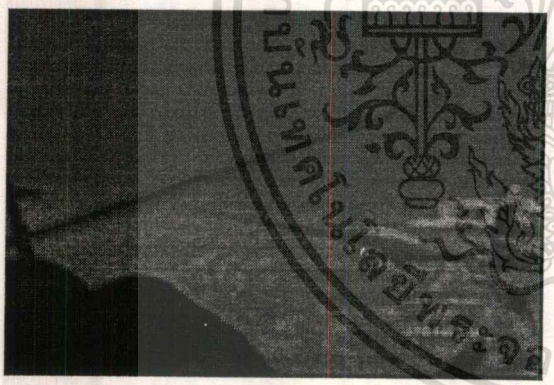


Picture 7

Type : P - Picture

Picture 8

Type : B - Picture



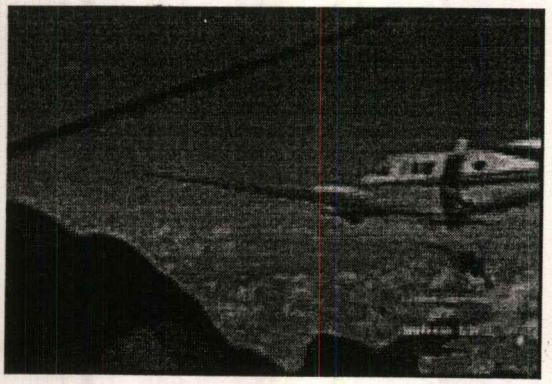
Picture 9

Type : B - Picture

Picture 10

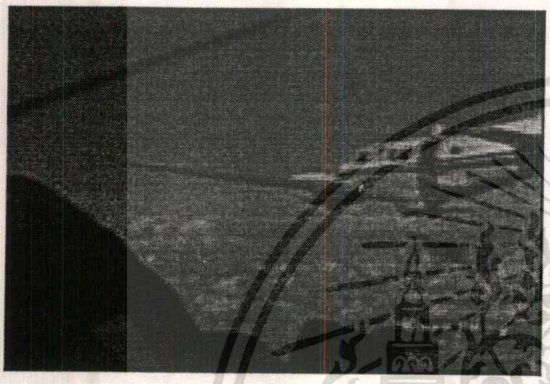
Type : P - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



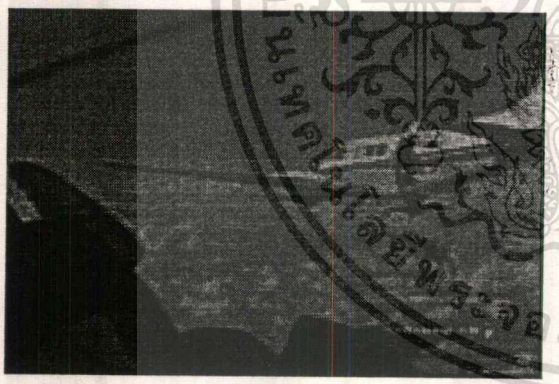
Picture 11

Type : B - Picture



Picture 12

Type : B - Picture



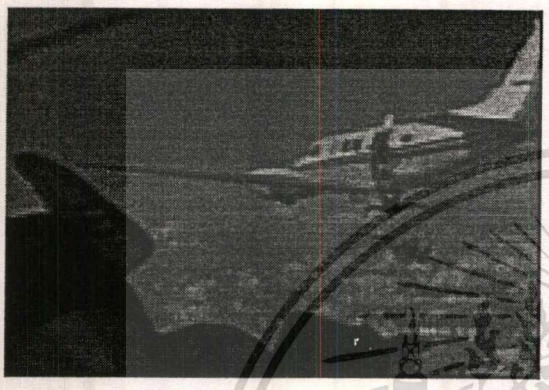
Picture 13

Type : P - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

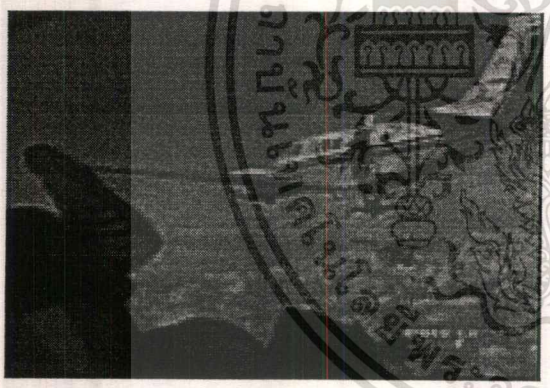
GOP 2 มี 15 ภาพ

ภาพเมื่อเรียงตามลำดับการแสดงผลแล้วได้เป็น BBIBBPBBPBBPBBP



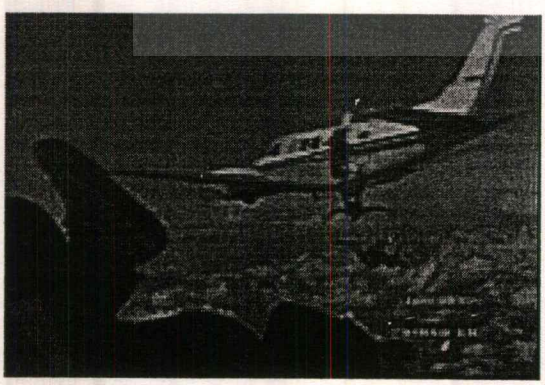
Picture 1

Type : B - Picture



Picture 2

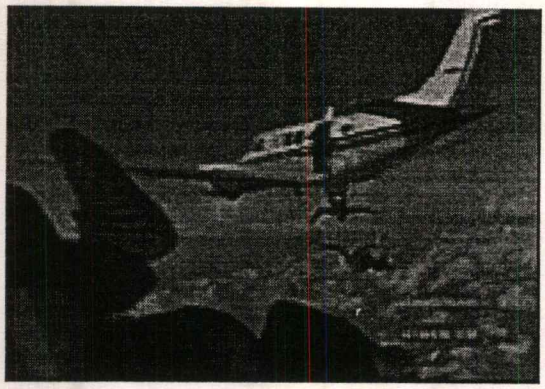
Type : B - Picture



Picture 3

Type : I - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

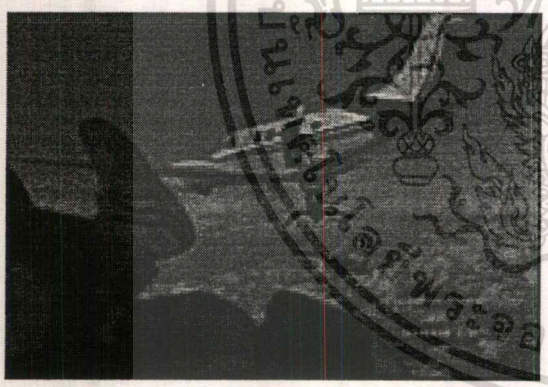


Picture 4

Type : B - Picture

Picture 5

Type : B - Picture



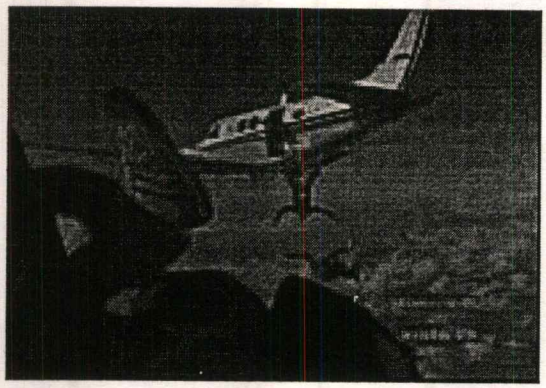
Picture 6

Type : P - Picture

Picture 7

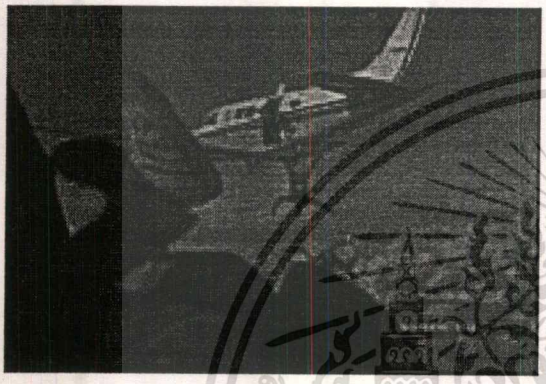
Type : B - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Picture 8

Type : B - Picture

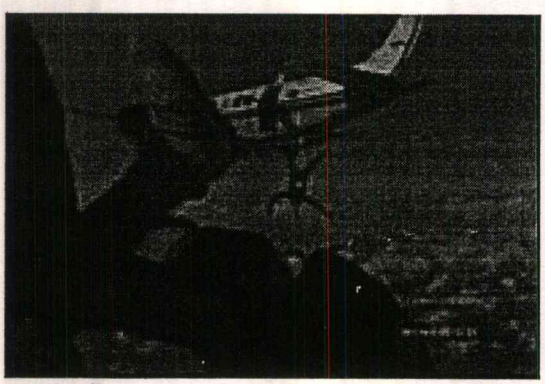


Picture 9

Type : P - Picture

Picture 10

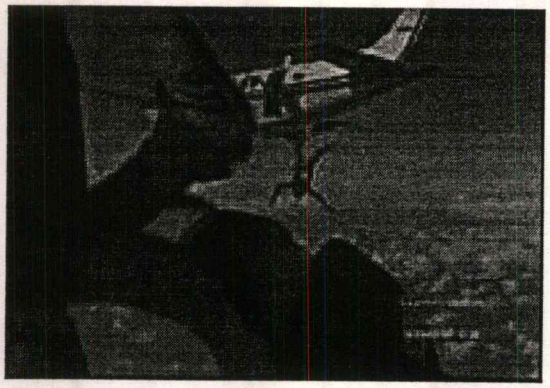
Type : B - Picture



Picture 11

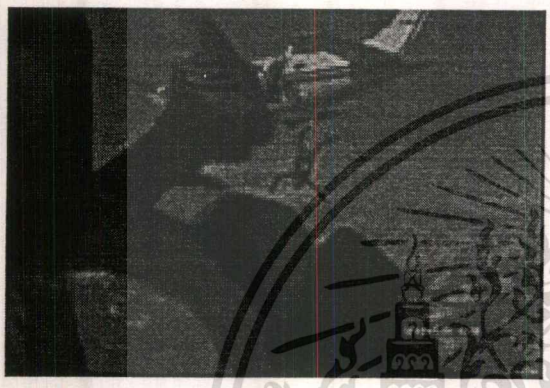
Type : B - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



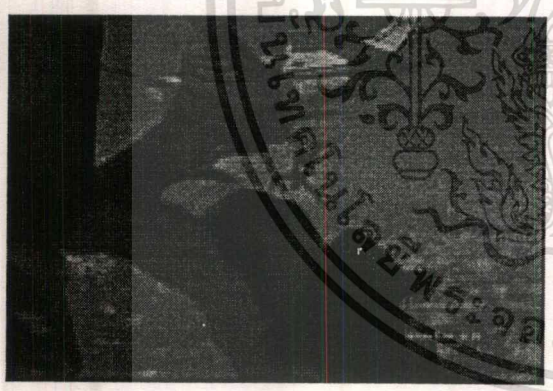
Picture 12

Type : P - Picture



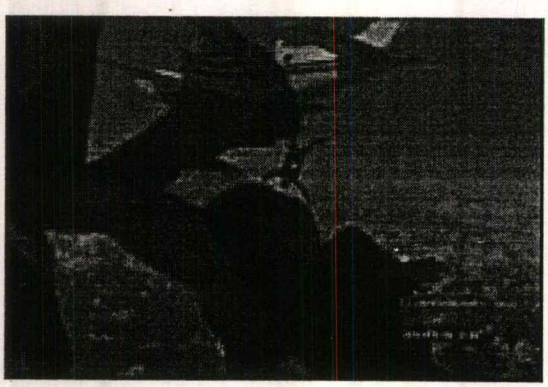
Picture 13

Type : B - Picture



Picture 14

Type : B - Picture



Picture 15

Type : P - Picture

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.6 ปัญหาและการแก้ไข

1.เนื่องจาก Bit stream ถูกส่งออกมาแบบ Serial bits ดังนั้นหากบิตใดบิตหนึ่งผิดพลาด ก็จะไม่สามารถ รู้ได้ในทันทีเนื่องจากบิตที่ตามมาอาจจะตรงกับรูปแบบของข้อมูลถัดไปได้

2. หากโปรแกรมตรวจพบความผิดพลาด ก็จะไม่สามารถทราบได้แน่นอนว่า ผิดพลาดจากจุดใดและเมื่อใด ดังนั้น การ debug โปรแกรมจึงทำได้ยากมาก แก้ไขได้โดยการทำสัญลักษณ์ไว้แล้วให้แสดงแต่ละบิตบนหน้าจอแล้วตรวจทีละบิต

3. ข้อมูลมีรูปแบบที่ซับซ้อน ขนาดความยาวไม่แน่นอนแตกต่างกันออกไปมากมาย ทำให้การตรวจสอบด้วยมือแทบจะเป็นไปไม่ได้เลย ดังนั้นการแก้จึงทำได้เพียงการใช้เครื่องมือ การ debug ของโปรแกรม และเทคนิคของผู้ที่มีความชำนาญในการเขียนโปรแกรมในการตรวจสอบ

4. รายละเอียดของมาตรฐานบางส่วนไม่สมบูรณ์ หรือเข้าใจยาก จึงทำให้โปรแกรมมี จุดปัญหาจำนวนมาก

5. เนื่องจากต้องใช้ memory จำนวนมากและตัวของมาตรฐานถูกเขียนไว้ในรูปของภาษา C จึงเหมาะสมที่ต้องใช้ Visual C++ เป็น compiler ซึ่งมีความซับซ้อนตามแบบของโปรแกรมที่สร้าง Application บน windows 95 ต้องใช้เวลาในการศึกษาค่อนข้างมากทำให้การทำงานล่าช้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. Kluwer Academic Publisher, " IMAGE AND VIDEO COMPRESSION STANDARD Algorithms and Architectures", 1995.

2. ISO/IEC, "INTERNATIONAL STANDARD ISO/IEC 11172-1 Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s-Part 1: System ", 1993.

3. ISO/IEC, "INTERNATIONAL STANDARD ISO/IEC 11172-2 Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s Part 2 : Video", 1993.

4. Brian W.Kernighan, Dennis M.Ritchie, "THE C PROGRAMMING LANGUAGE", 1988.

5. ธันวา ศรีประมง, " การเขียนโปรแกรมภาษาซีสำหรับวิศวกรรม", 2521.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

โปรแกรมสมบูรณ์ที่ใช้ถอดรหัสไฟล์ MPEG-1

1. ไฟล์ MPEG Decode.c

```
#include "Dct.h"
FILE *textfile;
FILE *testfile;

char stopchar;
int picture_start_code=0x100;
int user_data_start_code=0x1b2;
int sequence_header_code=0x1b3;
int sequence_error_code=0x1b4;
int extension_start_code=0x1b5;
int sequence_end_code=0x1b7;
int group_start_code=0x1b8;
int system_start_code=0x1b9;

void iso11172_stream(void);
void pack(void);
void system_header(void);
void packet(void);

void video_sequence(void);
void sequence_header(void);
void group_of_pictures(void);
void picture(void);
void slice(void);
void macroblock(void);
void block(int a);

int slice_start_code(void);
void macroblock_type(void);
int pattern_code(int a);
int nextkeyword(int a);
int nextbit(int a);
int nextbits(int a);
int shift_left(int a,int b);
int shift_right(int a,int b);
void Type_of_stream(void);
char buffer(void);
void next_start_code(void);
void sub_next_start_code(void);
void sub_pack(void);
void sub_packet(void);
int sub_nextkeyword(int a);
char sub_buffer(void);

int vlc_macroblock_address_increment(void);
int vlc_motion_code(void);
int vlc_code_block_pattern(void);
int vlc_dct_dc_size_luminance(void);
int vlc_dct_dc_size_chrominance(void);
```

สงวนลิขสิทธิ์ใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void vlc_dct_coeff(int *out);
void obtainvalue(int a,int b,int *z);
void escape_sequence(int *z);
void dequantize_ipicture(int block_number);
void dequantize_nonipicture(int block_number);
void keep_picture(void);
void block_search(int block_number);
void motion_rebuild(void);

void dct_level_2(int *out);
void dct_level_3(int *out);
void dct_level_4(int *out);
void dct_level_6(int *out);
void dct_level_7(int *out);
void dct_level_8(int *out);
void dct_level_9(int *out);
void dct_level_10(int *out);
void dct_level_11(int *out);

int packet_length,packet_data_length,packet_data_byte;
int stream_id,reserved_stream=188,private_stream_1=189,padding_stream=190,private_stream_2=191;
int macroblock_address_increment,code_block_pattern_value,macroblock_address=-1;
int forward_f_code,forward_r_size,forward_f,full_pel_forward_vector;
int backward_f_code,backward_r_size,backward_f,full_pel_backward_vector;
int motion_horizontal_forward_code,motion_vertical_forward_code;
int motion_horizontal_backward_code,motion_vertical_backward_code;
int motion_horizontal_forward_r,motion_vertical_forward_r;
int motion_horizontal_backward_r,motion_vertical_backward_r;
int past_intra_address,dct_dc_y_past,dct_dc_cb_past,dct_dc_cr_past;
int picture_coding_type;
int recon_right_for,recon_down_for;
int recon_right_back,recon_down_back;
int recon_right_for_prev=0,recon_down_for_prev=0;
int recon_right_back_prev=0,recon_down_back_prev=0;
int past_picture[3][240][352],next_picture[3][240][352];
int recent_picture[3][240][352],pel[8][8];
int picture_out[15][22][6][8][8],block_out[8][8];
int block_idct_out[8][8],data_out[64];

int video_stream_flag=0,stream_jump_flag=0;
int close_gop_flag,coeff_first_flag,recon_ppic_flag=0;

int start_code[4],quantizer_scale;
int data_code[8],data_byte;
int byte_no=0,bit_out,macroblock_number=0,slice_number=0,picture_number=0,GOP_number=0;
char buffer_code[64];
int video_buffer[3][81920],buffer_length[3];
int write_mem=0,read_mem,mem_point,bit_amout;
int bit_test=0,test_space=0;
int ipicture_number=0,ppicture_number=0,bpicture_number=0;
int macroblock_quant,macroblock_motion_forward;
int macroblock_motion_backward,macroblock_pattern,macroblock_intra;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int intra_quantizer_matrix[8][8]=
{
                {8,16,19,22,26,27,29,34},
                {16,16,22,24,27,29,34,37},
                {19,22,26,27,29,34,34,38},
                {22,22,26,27,29,34,37,40},
                {22,26,27,29,32,35,40,48},
                {26,27,29,32,35,40,48,58},
                {26,27,29,34,38,46,56,69},
                {27,29,35,38,46,56,69,83}
};

int non_intra_quantizer_matrix[8][8]=
{
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16},
                {16,16,16,16,16,16,16,16}

};


int zzscan[8][8]=
{
                {0,1,5,6,14,15,27,28},
                {2,4,7,13,16,26,29,42},
                {3,8,12,17,25,30,41,43},
                {9,11,18,24,31,40,44,53},
                {10,19,23,32,39,45,52,54},
                {20,22,33,38,46,51,55,60},
                {21,34,37,47,50,56,59,61},
                {35,36,48,49,57,58,62,63}

};


void main(void)
{
        int k,i,l,j;


        textfile=fopen("Test.mpg","rb");
        testfile=fopen("Ptest.raw","wb");
        byte_no=0;bit_out=0;

/*
        for(i=0;i<1000;i++)
        {
                scanf("%d %d",&k,&l);
                dct_coeff(k,l,&z);
                printf("Level=%0x Length=%d\n",z[0],z[1]);
                stopchar=getch();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }

        for(i=0;i<1000;i++)
        {
                k=sub_nextkeyword(32);

                if(k<0x110)
                        printf("%0x\n",k);

                if((k!=0x1ba)&&(k!=0x1bb)&&(k!=0x1b3)&&(k!=0x1b8)&&(k!=0x100)&&
 (k!=0x101))
                {
                        l=sub_nextkeyword(16);
                        printf("Packet length = %d\n",l);
                        for(j=0;j<l;j++)
                        {
                                k=sub_nextkeyword(8);
                        }
                        k=sub_nextkeyword(32);
                        printf("Is there error here> %0x    byte_no=%d\n\n",k,byte_no);
 //                     sub_next_start_code();
 //                     printf("Byte no= %d",byte_no);
 //                     stopchar=getch();
                }
                sub_next_start_code();
                stopchar=getch();
        }
 */
 /*
        sub_next_start_code();
        k=sub_nextkeyword(32);
        for(i=0;i<6;i++)
        {
                k=0;
                while(k!=0x1b8)
                {
                        sub_nextkeyword(8);
                        sub_next_start_code();
                        k=nextbit(32);
                }
        }

        k=sub_nextkeyword(32);
        for(i=0;i<1;i++)
        {
                k=0;
                while(k!=0x100)
                {
                        sub_nextkeyword(8);
                        sub_next_start_code();
                        k=nextbit(32);
                }
        }

        picture();
 */
 /*
        k=nextbit(32);
```

เอกสารนี้เป็นเอกสารสงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        for(i=0;i<4;i++)
        {
                ch=sub_nextkeyword(8);
                fputc(ch,testfile);
        }
        printf("%0x",k);
        stopchar=getch();
        k=0;
        while(k!=0x100)
        {
                ch=sub_nextkeyword(8);
                fputc(ch,testfile);
                k=nextbit(32);
                printf("%0x ",k);
        }

        for(i=0;i<4;i++)
        {
                ch=sub_nextkeyword(8);
                fputc(ch,testfile);
        }

        stopchar=getch();

        fclose(testfile);
        fclose(textfile);
*/

        iso11172_stream();

        printf("\nByte number = %d\n",byte_no);
        stopchar=getch();

}


void iso11172_stream(void)
{

        do
        {
                pack();
                if(nextbit(32)==439)
                        nextkeyword(32);
        }while(nextbit(32)==442);

}



void pack(void)
{


        if(sub_nextkeyword(32)!=442)
        //pack_start_code
        {
                printf("pack_start_code ERROR !!!\n");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                stopchar=getch();
        }

        if(sub_nextkeyword(4)!=2)
        {
                printf("pack error\n");
                stopchar=getch();
        }
        sub_nextkeyword(3);
        sub_nextkeyword(1);
        sub_nextkeyword(15);
        sub_nextkeyword(1);
        sub_nextkeyword(15);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(22);
        sub_nextkeyword(1);
        if(nextbit(32)==443)
                system_header();

        while((nextbit(24)==1)&&(nextbit(32)!=442)&&(nextbit(32)!=441)&&(nextbit(32)!=439))
                packet();
        }


void system_header(void)
{

        sub_nextkeyword(32);
        sub_nextkeyword(16);
        sub_nextkeyword(1);
        sub_nextkeyword(22);
        sub_nextkeyword(1);
        sub_nextkeyword(6);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(5);
        sub_nextkeyword(8);

        while(nextbit(1)==1)
        {
                sub_nextkeyword(8);
                if(sub_nextkeyword(2)!=3)
                {
                        printf("header error");
                        stopchar=getch();
                }

                sub_nextkeyword(1);
                sub_nextkeyword(13);
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void packet(void)
{

        int i,j,k;
        char ch;

        if(sub_nextkeyword(24)!=1)
        {
                printf("packet_start_code_prefix  ERROR!! \n");
                stopchar=getch();
        }

        stream_id=sub_nextkeyword(8);
        packet_length=sub_nextkeyword(16);

        Type_of_stream();
        printf("Packet length = %d\n\n",packet_length);

        k=byte_no;

        if(stream_id!=private_stream_2)
        {
                while(nextbit(8)==255)
                        sub_nextkeyword(8);
                if(nextbit(2)==1)
                {
                        sub_nextkeyword(2);
                        sub_nextkeyword(1);
                        sub_nextkeyword(13);
                }
                if(nextbit(4)==2)
                {
                        sub_nextkeyword(4);
                        sub_nextkeyword(3);
                        sub_nextkeyword(1);
                        sub_nextkeyword(15);
                        sub_nextkeyword(1);
                        sub_nextkeyword(15);
                        sub_nextkeyword(1);
                }
                else if(nextbit(4)==3)
                {
                        sub_nextkeyword(4);
                        sub_nextkeyword(3);
                        sub_nextkeyword(1);
                        sub_nextkeyword(15);
                        sub_nextkeyword(1);
                        sub_nextkeyword(15);
                        sub_nextkeyword(1);

                        if(sub_nextkeyword(4)!=1)
                        {
                                printf("Packet  ERROR");
                                stopchar=getch();
                        }

                        sub_nextkeyword(3);
                        sub_nextkeyword(1);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                sub_nextkeyword(15);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
        }
        else
        {
                if(sub_nextkeyword(8)!=15)
                {
                        printf("Packet ERROR !!");
                        stopchar=getch();
                }

        }
}

k=byte_no-k;

packet_data_length=packet_length-k;
packet_data_byte=-1;

if(video_stream_flag)
{
        for(i=0;i<packet_data_length;i++)
        {

                if(i<7)
                {
                        for(j=0;j<8;j++)
                                video_buffer[write_mem][i*8+j]=buffer_code[i*8+j];
                }
                else
                {
                        ch=fgetc(textfile);
                        for(j=0;j<8;j++)
                        {
                                k=ch&1;
                                video_buffer[write_mem][8*(i+1)-(j+1)]=k;
                                ch>>=1;
                        }
                }

        }

        for(i=0;i<57;i=i+8)
        {
                ch=fgetc(textfile);
                for(j=8+i;j>i;j--)
                {
                        k=ch&1;
                        buffer_code[j-1]=k;
                        ch>>=1;
                }
        }
        bit_out=1;
        buffer_length[write_mem]=packet_data_length;
        write_mem++;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                video_stream_flag=0;
                if(write_mem==2)
                {
                        read_mem=0;
                        mem_point=0;
                        bit_amout=buffer_length[read_mem]*8;
                        video_sequence();
                }
        }
        else
        {
                for(i=0;i<packet_data_length;i++)
                        sub_nextkeyword(8);
        }

}



void video_sequence(void)
{

        next_start_code();
        do
        {
                sequence_header();
                do
                {
                        group_of_pictures();
                        GOP_number++;
                        picture_number=1;
                }while(nextbits(32)==group_start_code);
        }while(nextbits(32)==sequence_header_code);
        nextkeyword(32);

}


void sequence_header(void)
{
        if(nextkeyword(32)!=0x1b3)
        {
                printf("/n sequence_header code ERROR !! /n");
                stopchar=getch();
        }

        nextkeyword(12);
        nextkeyword(12);
        nextkeyword(4);
        nextkeyword(4);
        nextkeyword(18);
        nextkeyword(1);
        nextkeyword(10);
        nextkeyword(1);
        if(nextkeyword(1)==1)
                nextkeyword(8*64);
        if(nextkeyword(1)==1)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                nextkeyword(8*64);

        next_start_code();
        if(nextbits(32)==extension_start_code)
        {
                nextkeyword(32);
                while(nextbits(24)!=1)
                        nextkeyword(8);
                next_start_code();
        }

        if(nextbits(32)==user_data_start_code)
        {
                nextkeyword(32);
                while(nextbits(24)!=1)
                        nextkeyword(8);
                next_start_code();
        }

}


void group_of_pictures(void)
{

        if(nextkeyword(32)!=group_start_code)
        {
                printf("/n group_start_code ERROR !! /n");
                stopchar=getch();
        }

        printf("\nThis is start of GOP %d",GOP_number);
        stopchar=getch();

        nextkeyword(25);
        close_gop_flag=nextkeyword(1);
        nextkeyword(1);

        next_start_code();

        if(nextbits(32)==extension_start_code)
        {
                nextkeyword(32);
                while(nextbits(24)!=1)
                        nextkeyword(8);
                next_start_code();
        }
        if(nextbits(32)==user_data_start_code)
        {
                nextkeyword(32);
                while(nextbits(24)!=1)
                        nextkeyword(8);
                next_start_code();
        }

        do
        {
                picture();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            printf("Number of slice = %d\n",slice_number);
            slice_number=0;
            picture_number++;

        }while(nextbits(32)==picture_start_code);

}


void picture(void)
{

        int k,l,m,n;

        if(nextkeyword(32)!=picture_start_code)
        {
                printf("/n picture_start_code ERROR !! /n");
                stopchar=getch();
        }

        printf("\n\nThis  is  start of picture %d  of  GOP >> %d\n",picture_number,GOP_number);
        nextkeyword(10);
        picture_coding_type=nextkeyword(3);
        printf("This  picture 's type is %d\n",picture_coding_type);
        nextkeyword(16);
        stopchar=getch();

        if((picture_coding_type==2)&&(recon_ppic_flag))
        {
                for(m=0;m<240;m++)
                {
                        for(n=0;n<352;n++)
                        {
                                k=floor(m/2);
                                l=floor(n/2);
                                past_picture[0][m][n]=next_picture[0][m][n];
                                past_picture[1][k][l]=next_picture[1][k][l];
                                past_picture[2][k][l]=next_picture[2][k][l];
                        }
                }
        }


        if((picture_coding_type==2)||(picture_coding_type==3))
        {
                full_pel_forward_vector=nextkeyword(1);
                forward_f_code=nextkeyword(3);
                forward_r_size=forward_f_code-1;
                forward_f=1<<forward_r_size;
        }

        if(picture_coding_type==3)
        {
                full_pel_backward_vector=nextkeyword(1);
                backward_f_code=nextkeyword(3);
                backward_r_size=backward_f_code-1;
                backward_f=1<<backward_r_size;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while(nextbits(1)==1)
{
        nextkeyword(1);
        nextkeyword(8);
}

nextkeyword(1);
next_start_code();


if(nextbits(32)==extension_start_code)
{
        nextkeyword(32);
        while(nextbits(24)!=1)
                nextkeyword(8);
        next_start_code();
}

if(nextbits(32)==user_data_start_code)
{
        nextkeyword(32);
        while(nextbits(24)!=1)
                nextkeyword(8);
        next_start_code();
}

do
{
        slice();
        slice_number++;
}while(slice_start_code());

if(picture_coding_type==3)
        bpicture_number++;


keep_picture();


if(picture_coding_type==1)
{
        for(m=0;m<240;m++)
        {
                for(n=0;n<352;n++)
                {
                        k=floor(m/2);
                        l=floor(n/2);
                        past_picture[0][m][n]=recent_picture[0][m][n];
                        past_picture[1][k][l]=recent_picture[1][k][l];
                        past_picture[2][k][l]=recent_picture[2][k][l];
                }
        }
}
else if(picture_coding_type==2)
{
        for(m=0;m<240;m++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                        for(n=0;n<352;n++)
                        {
                                k=floor(m/2);
                                l=floor(n/2);
                                next_picture[0][m][n]=recent_picture[0][m][n];
                                next_picture[1][k][l]=recent_picture[1][k][l];
                                next_picture[2][k][l]=recent_picture[2][k][l];
                                recon_ppic_flag=1;
                        }
                }
        }

}


void slice(void)
{
        int k;

        past_intra_address=-2;
        dct_dc_y_past=128*8;
        dct_dc_cr_past=128*8;
        dct_dc_cb_past=128*8;
        recon_right_for_prev=0;
        recon_down_for_prev=0;
        recon_right_back_prev=0;
        recon_down_back_prev=0;


        k=nextkeyword(32);
        printf("\nslice start code = %0x \n",k);
//      stopchar=getch();

        quantizer_scale=nextkeyword(5);
        while(nextbits(1)==1)
        {
                nextkeyword(1);
                nextkeyword(8);
        }
        nextkeyword(1);
        macroblock_number=0;

        do
        {
                code_block_pattern_value=0;
                macroblock();
                macroblock_number++;

        }while(nextbits(23)!=0);
        printf("Number of macroblock is %d \n",macroblock_number);
        next_start_code();

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void macroblock(void)
{
        int i,j,k,l,m,n;

        while(nextbits(11)==15)
                nextkeyword(11);
        while(nextbits(11)==8)
                nextkeyword(11);

        macroblock_address_increment=vlc_macroblock_address_increment();

        if(macroblock_address_increment>1)
        {
                if(picture_coding_type==2)
                {
                        recon_right_for=0;
                        recon_down_for=0;
                        m=macroblock_address_increment-1;
                        for(n=0;n<m;n++)
                        {
                                for(i=0;i<6;i++)
                                {
                                        block_search(i);
                                        for(k=0;k<8;k++)
                                        {
                                                for(l=0;l<8;l++)
picture_out[slice_number][macroblock_number][i][k][l]=pel[k][l];
                                        }
                                }
                                macroblock_number++;
                        }
                }
                else if(picture_coding_type==3)
                {
                        m=macroblock_address_increment-1;
                        for(n=0;n<m;n++)
                        {
                                for(i=0;i<6;i++)
                                {
                                        block_search(i);
                                        for(k=0;k<8;k++)
                                        {
                                                for(l=0;l<8;l++)
picture_out[slice_number][macroblock_number][i][k][l]=pel[k][l];
                                        }
                                }
                                macroblock_number++;
                        }
                }
                else
                {
                        printf("\n  skip macroblock ERROR !! \n");
                        stopchar=getch();

                }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/*
        k=macroblock_address_increment;
        printf("\n\nmacroblock number %d\n",macroblock_number);
        printf("macroblock address increment= %d\n",k);
*/
        macroblock_address=macroblock_address+macroblock_address_increment;
        macroblock_type();
/*
        printf("\nmacroblock quant= %d\n",macroblock_quant);
        printf("macroblock intra= %d\n",macroblock_intra);
        printf("macroblock pattern= %d\n",macroblock_pattern);
        printf("macroblock motion forward= %d\n",macroblock_motion_forward);
        printf("macroblock motion backward= %d\n",macroblock_motion_backward);
        printf("forward_r_size= %d      ",forward_r_size);
        printf("forward_f= %d\n",forward_f);
*/
//      if(picture_coding_type==3)
//              stopchar=getch();

        if(!macroblock_intra)
        {
                dct_dc_y_past=128*8;
                dct_dc_cr_past=128*8;
                dct_dc_cb_past=128*8;
        }


        if(macroblock_quant)
                quantizer_scale=nextkeyword(5);

        if(macroblock_motion_forward)
        {
                motion_horizontal_forward_code=vlc_motion_code();
                if((forward_f!=1)&&(motion_horizontal_forward_code!=0))
                        motion_horizontal_forward_r=nextkeyword(forward_r_size);

                motion_vertical_forward_code=vlc_motion_code();
                if((forward_f!=1)&&(motion_vertical_forward_code!=0))
                        motion_vertical_forward_r=nextkeyword(forward_r_size);
        }


        if(macroblock_motion_backward)
        {
                motion_horizontal_backward_code=vlc_motion_code();
                if((backward_f!=1)&&(motion_horizontal_backward_code!=0))
                        motion_horizontal_backward_r=nextkeyword(backward_r_size);

                motion_vertical_backward_code=vlc_motion_code();
                if((backward_f!=1)&&(motion_vertical_backward_code!=0))
                        motion_vertical_backward_r=nextkeyword(backward_r_size);
        }


        if(picture_coding_type==2)
        {
                if((!macroblock_motion_forward)||(macroblock_address_increment>1))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
                recon_right_for_prev=0;
                recon_down_for_prev=0;
        }

        if((!macroblock_motion_backward)||(macroblock_address_increment>1))
        {
                recon_right_back_prev=0;
                recon_down_back_prev=0;
        }
}

if(picture_coding_type==3)
{
        if(macroblock_intra)
        {
                recon_right_for_prev=0;
                recon_down_for_prev=0;
                recon_right_back_prev=0;
                recon_down_back_prev=0;
        }
}


if(!macroblock_intra)
        motion_rebuild();


if(macroblock_pattern)
        code_block_pattern_value=vlc_code_block_pattern();

for(i=0;i<6;i++)
{
        for(k=0;k<64;k++)
                data_out[k]=0;
        for(k=0;k<8;k++)
        {
                for(l=0;l<8;l++)
                        block_out[k][l]=0;
        }

        block(i);

        if(macroblock_intra)
                idct(block_out,&block_idct_out);
        else
        {
                idct(block_out,&block_idct_out);
                block_search(i);
                for(k=0;k<8;k++)
                {
                        for(l=0;l<8;l++)
                                block_idct_out[k][l]=block_idct_out[k][l]+pel[k][l];
                }
        }

/*
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                if(picture_coding_type==2)
                {
                        printf("\n\n");
                        for(k=0;k<8;k++)
                        {
                                for(l=0;l<8;l++)
                                {
                                        printf("%d ",block_idct_out[k][l]);
                                }
                                printf("\n");
                        }
                        stopchar=getch();
                }
*/


                for(k=0;k<8;k++)
                {
                        for(l=0;l<8;l++)
                        {
                                j=block_idct_out[k][l];
                                if(j<0)

                                        j=0;
                //limit to range 0-255
                                else if(j>255)
                                        j=255;
                                picture_out[slice_number][macroblock_number][i][k][l]=j;
                        }
                }

        }

        if(macroblock_intra)
                past_intra_address=macroblock_address;

        if(picture_coding_type==4)
                nextkeyword(1);

}


void block(int block_number)
{
        int k,l,data_out_number,z[2];

        if(pattern_code(block_number))
        {
                data_out_number=0;
                if(macroblock_intra)
                {
                        if(block_number<4)
                        {
                                k=vlc_dct_dc_size_luminance();
                                if(k!=0)
                                {
                                        if(nextbits(1)==1)
                                                data_out[0]=nextkeyword(k);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        else
                        {
                                l=nextkeyword(k);
                                l=(l^0xffffffff);
                                l=shift_right(l,k);
                                data_out[0]=-1*l;
                        }
                        data_out_number++;
                }
                else
                        data_out_number++;
        }
        else
        {
                k=vlc_dct_dc_size_chrominance();
                if(k!=0)
                {
                        if(nextbits(1)==1)
                                data_out[0]=nextkeyword(k);
                        else
                        {
                                l=nextkeyword(k);
                                l=(l^0xffffffff);
                                l=shift_right(l,k);
                                data_out[0]=-1*l;
                        }
                        data_out_number++;
                }
                else
                        data_out_number++;
        }
}
else
{
        coeff_first_flag=1;
        vlc_dct_coeff(&z);
        data_out[z[1]]=z[0];
        data_out_number=z[1]+1;
        coeff_first_flag=0;
}

if(picture_coding_type!=4)
{
        while(nextbits(2)!=2)
        {
                vlc_dct_coeff(&z);
                data_out[data_out_number+z[1]]=z[0];
                data_out_number=data_out_number+z[1]+1;
        }

        if(data_out_number>64)
        {
                printf("BLOCK DATA OVER !!\n");
                stopchar=getch();
        }

        if(nextkeyword(2)!=2)
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        printf("End of block ERROR !!\n");
                        stopchar=getch();
                }

        }

/*
        if(picture_coding_type==2)
        {
                printf("\n\n");
                for(k=0;k<8;k++)
                {
                        for(l=0;l<8;l++)
                        {
                                printf("%d  ",data_out[zzscan[k][l]]);
                        }
                        printf("\n");
                }
                stopchar=getch();
        }
*/

        if(macroblock_intra)
                dequantize_ipicture(block_number);
        else
                dequantize_nonipicture(block_number);

    }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int vlc_macroblock_address_increment(void)
{

        if(nextkeyword(1))
                return 1;
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 2;
                else
                        return 3;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 4;
                else
                        return 5;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 6;
                else
                        return 7;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 8;
                        else
                                return 9;
                }
                else if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 10;
                        else
                                return 11;
                }
                else
                {
                        if(nextkeyword(1))
                                return 12;
                        else
                                return 13;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 14;
                        else
                                return 15;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            else if(nextkeyword(1))
            {
                    if(nextkeyword(1))
                    {
                            if(nextkeyword(1))
                                    return 16;
                            else
                                    return 17;
                    }
                    else
                    {
                            if(nextkeyword(1))
                                    return 18;
                            else
                                    return 19;
                    }
            }
            else  if(nextkeyword(1))
            {
                    if(nextkeyword(1))
                            return 20;
                    else
                            return 21;
            }
            else  if(nextkeyword(1))
            {
                    if(nextkeyword(1))
                            return 22;
                    else
                            return 23;
            }
            else
            {
                    if(nextkeyword(1))
                            return 24;
                    else
                            return 25;
            }

}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        return 26;
                                else
                                        return 27;
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        return 28;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                else
                                        return 29;
                        }
                }
                else if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 30;
                        else
                                return 31;
                }
                else
                {
                        if(nextkeyword(1))
                                return 32;
                        else
                                return 33;
                }
        }
        else
        {
                printf("vlc_macroblock_address_increment code ERROR!!\n");
                printf("At GOP>%d Picture>%d Slice>%d
Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
                stopchar=getch();
        }
    }
    else
    {
        printf("vlc_macroblock_address_increment OVERZERO ERROR!!\n");
        printf("At GOP>%d Picture>%d Slice>%d
Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
        stopchar=getch();
    }

}


int vlc_motion_code(void)
{

    if(nextkeyword(1))
        return 0;
    else if(nextkeyword(1))
    {
        if(nextkeyword(1))
            return -1;
        else
            return 1;
    }
    else if(nextkeyword(1))
    {
        if(nextkeyword(1))
            return -2;
        else
            return 2;
    }
    else if(nextkeyword(1))
```

เอกสารนี้ใช้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
        if(nextkeyword(1))
                return -3;
        else
                return 3;
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return -4;
                else
                        return 4;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return -5;
                else
                        return 5;
        }
        else
        {
                if(nextkeyword(1))
                        return -6;
                else
                        return 6;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return -7;
                else
                        return 7;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return -8;
                        else
                                return 8;
                }
                else
                {
                        if(nextkeyword(1))
                                return -9;
                        else
                                return 9;
                }
        }
        else if(nextkeyword(1))
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                if(nextkeyword(1))
                        return -10;
                else
                        return 10;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return -11;
                else
                        return 11;
        }
        else
        {
                if(nextkeyword(1))
                        return -12;
                else
                        return 12;
        }

}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        return -13;
                                else
                                        return 13;
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        return -14;
                                else
                                        return 14;
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        return -15;
                                else
                                        return 15;
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        return -16;
                                else
                                        return 16;
                        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                }
            }
        }
        else
        {
            printf("\n vlc_motion_code OVERZERO ERROR!!\n");
            printf("At GOP>%d Picture>%d Slice>%d
Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
            stopchar=getch();
        }

}


int vlc_code_block_pattern(void)
{

    if(nextkeyword(1))
    {
        if(nextkeyword(1))
        {
            if(nextkeyword(1))
                return 60;
            else
            {
                if(nextkeyword(1))
                    return 4;
                else
                    return 8;
            }
        }
        else if(nextkeyword(1))
        {
            if(nextkeyword(1))
                return 16;
            else
                return 32;
        }
        else if(nextkeyword(1))
        {
            if(nextkeyword(1))
                return 12;
            else
                return 48;
        }
        else
        {
            if(nextkeyword(1))
                return 20;
            else
                return 40;
        }
    }
    else if(nextkeyword(1))
    {
        if(nextkeyword(1))
        {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 28;
                        else
                                return 44;
                }
                else
                {
                        if(nextkeyword(1))
                                return 52;
                        else
                                return 56;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 1;
                else
                        return 61;
        }
        else
        {
                if(nextkeyword(1))
                        return 2;
                else
                        return 62;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 24;
                        else
                                return 36;
                }
                else
                {
                        if(nextkeyword(1))
                                return 3;
                        else
                                return 63;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 5;
                        else
                                return 9;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                else
                {
                        if(nextkeyword(1))
                                return 17;
                        else
                                return 33;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 6;
                else
                        return 10;
        }
        else
        {
                if(nextkeyword(1))
                        return 18;
                else
                        return 34;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        return 7;
                                else
                                        return 11;
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        return 19;
                                else
                                        return 35;
                        }
                }
                else if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 13;
                        else
                                return 49;
                }
                else
                {
                        if(nextkeyword(1))
                                return 21;
                        else
                                return 41;
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            }
            else if(nextkeyword(1))
            {
                    if(nextkeyword(1))
                    {
                            if(nextkeyword(1))
                                    return 14;
                            else
                                    return 50;
                    }
                    else
                    {
                            if(nextkeyword(1))
                                    return 22;
                            else
                                    return 42;
                    }
            }
            else if(nextkeyword(1))
            {
                    if(nextkeyword(1))
                            return 15;
                    else
                            return 51;
            }
            else
            {
                    if(nextkeyword(1))
                            return 23;
                    else
                            return 43;
            }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 25;
                        else
                                return 37;
                }
                else
                {
                        if(nextkeyword(1))
                                return 26;
                        else
                                return 38;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 29;
                else
                        return 45;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                }
                else
                {
                        if(nextkeyword(1))
                                return 53;
                        else
                                return 57;
                }

        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 30;
                        else
                                return 46;
                }
                else
                {
                        if(nextkeyword(1))
                                return 54;
                        else
                                return 58;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                return 31;
                        else
                                return 47;
                }
                else
                {
                        if(nextkeyword(1))
                                return 55;
                        else
                                return 59;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                        return 27;
                else
                        return 39;
        }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int vlc_dct_dc_size_luminance(void)
{

        if(nextkeyword(1)==0)
        {
                if(nextkeyword(1)==0)
                        return 1;
                else
                        return 2;
        }
        else
        {
                if(nextkeyword(1)==0)
                {
                        if(nextkeyword(1)==0)
                                return 0;
                        else
                                return 3;
                }
                else
                {
                        if(nextkeyword(1)==0)
                                return 4;
                        else
                        {
                                if(nextkeyword(1)==0)
                                        return 5;
                                else
                                {
                                        if(nextkeyword(1)==0)
                                                return 6;
                                        else
                                        {
                                                if(nextkeyword(1)==0)
                                                        return 7;
                                                else
                                                {
                                                        if(nextkeyword(1)==0)
                                                                return 8;
                                                        else
                                                        {
                                                                printf("dct_dc_size_luminance
ERROR \n");
                                                                printf("At GOP>%d  Picture>%d
Slice>%d  Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
                                                                stopchar=getch();
                                                        }
                                                }
                                        }
                                }
                        }
                }
        }

//      k=packet_data_length*8-(packet_data_byte*8+bit_out-1);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int vlc_dct_dc_size_chrominance(void)
{

        if(nextkeyword(1)==0)
        {
                if(nextkeyword(1)==0)
                        return 0;
                else
                        return 1;
        }
        else
        {
                if(nextkeyword(1)==0)
                        return 2;
                else
                {
                        if(nextkeyword(1)==0)
                                return 3;
                        else
                        {
                                if(nextkeyword(1)==0)
                                        return 4;
                                else
                                {
                                        if(nextkeyword(1)==0)
                                                return 5;
                                        else
                                        {
                                                if(nextkeyword(1)==0)
                                                        return 6;
                                                else
                                                {
                                                        if(nextkeyword(1)==0)
                                                                return 7;
                                                        else
                                                        {
                                                                if(nextkeyword(1)==0)
                                                                        return 8;
                                                                else
                                                                {
        printf("dct_dc_size_chrominance ERROR \n");
                                                                        printf("At GOP>%d
Picture>%d  Slice>%d
Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
                                                                        stopchar=getch();
                                                                }
                                                        }
                                                }
                                        }
                                }
                        }
                }
        }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void vlc_dct_coeff(int *out)
{

        test_space=1;

        if(nextkeyword(1))
        {
                if(coeff_first_flag)
                {
                        coeff_first_flag=0;
                        obtainvalue(0,1,out);
                }
                else
                {
                        nextkeyword(1);
                        obtainvalue(0,1,out);
                }
        }
        else
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                obtainvalue(1,1,out);
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(2,1,out);
                                else
                                        obtainvalue(0,2,out);
                        }
                }
                else
                        dct_level_2(out);
        }

        test_space=0;

}

void dct_level_2(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                obtainvalue(3,1,out);
                        else
                                obtainvalue(4,1,out);
                }
                else
                {
                        if(nextkeyword(1))
                                obtainvalue(0,3,out);
                        else
                                {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(10,1,out);
                                        else
                                                obtainvalue(0,5,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(1,3,out);
                                        else
                                                obtainvalue(3,2,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(11,1,out);
                                        else
                                                obtainvalue(12,1,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,6,out);
                                        else
                                                obtainvalue(13,1,out);
                                }
                        }
                }
        }
}
else
        dct_level_3(out);
}


void dct_level_3(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                obtainvalue(5,1,out);
                        else
                                obtainvalue(1,2,out);
                }
                else
                {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        if(nextkeyword(1))
                                obtainvalue(6,1,out);
                        else
                                obtainvalue(7,1,out);
                }
        }
        else
                dct_level_4(out);
}

void dct_level_4(int *out)
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                obtainvalue(8,1,out);
                        else
                                obtainvalue(0,4,out);
                }
                else
                {
                        if(nextkeyword(1))
                                obtainvalue(9,1,out);
                        else
                                obtainvalue(2,2,out);
                }
        }
        else
        {
                if(nextkeyword(1))
                        escape_sequence(out);
                else
                        dct_level_6(out);
        }
}

void dct_level_6(int *out)
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        obtainvalue(4,2,out);
                                else
                                        obtainvalue(14,1,out);
                        }
                        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        if(nextkeyword(1))
                                obtainvalue(15,1,out);
                        else
                                obtainvalue(1,4,out);
                }
        }
        else
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                                obtainvalue(2,3,out);
                        else
                                obtainvalue(0,7,out);
                }
                else
                {
                        if(nextkeyword(1))
                                obtainvalue(5,2,out);
                        else
                                obtainvalue(16,1,out);
                }
        }
}
else
        dct_level_7(out);

}


void dct_level_7(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(17,1,out);
                                        else
                                                obtainvalue(6,2,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,8,out);
                                        else
                                                obtainvalue(3,3,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        if(nextkeyword(1))
                                                obtainvalue(1,5,out);
                                        else
                                                obtainvalue(18,1,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(19,1,out);
                                        else
                                                obtainvalue(0,9,out);
                                }
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(20,1,out);
                                        else
                                                obtainvalue(21,1,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(7,2,out);
                                        else
                                                obtainvalue(2,4,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,10,out);
                                        else
                                                obtainvalue(4,3,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(8,2,out);
                                        else
                                                obtainvalue(0,11,out);
                                }
                        }
                }
        }
        else
                dct_level_8(out);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void dct_level_8(int *out)
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(22,1,out);
                                        else
                                                obtainvalue(23,1,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(24,1,out);
                                        else
                                                obtainvalue(25,1,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(26,1,out);
                                        else
                                                obtainvalue(0,12,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,13,out);
                                        else
                                                obtainvalue(0,14,out);
                                }
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,15,out);
                                        else
                                                obtainvalue(1,6,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(1,7,out);
                                        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        obtainvalue(2,5,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(3,4,out);
                                        else
                                                obtainvalue(5,3,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(9,2,out);
                                        else
                                                obtainvalue(10,2,out);
                                }
                        }
                }
        }
        else
                dct_level_9(out);

}


void dct_level_9(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,16,out);
                                        else
                                                obtainvalue(0,17,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,18,out);
                                        else
                                                obtainvalue(0,19,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,20,out);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                else
                                        obtainvalue(0,21,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(0,22,out);
                                else
                                        obtainvalue(0,23,out);
                        }
                }
        }
        else
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        obtainvalue(0,24,out);
                                else
                                        obtainvalue(0,25,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(0,26,out);
                                else
                                        obtainvalue(0,27,out);
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        obtainvalue(0,28,out);
                                else
                                        obtainvalue(0,29,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(0,30,out);
                                else
                                        obtainvalue(0,31,out);
                        }
                }
        }
}
else
        dct_level_10(out);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void dct_level_10(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(1,8,out);
                                        else
                                                obtainvalue(1,9,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(1,10,out);
                                        else
                                                obtainvalue(1,11,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                if(nextkeyword(1))
                                                obtainvalue(1,12,out);
                                        else
                                                obtainvalue(1,13,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(1,14,out);
                                        else
                                                obtainvalue(0,32,out);
                                }
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,33,out);
                                        else
                                                obtainvalue(0,34,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,35,out);
                                        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        obtainvalue(0,36,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,37,out);
                                        else
                                                obtainvalue(0,38,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(0,39,out);
                                        else
                                                obtainvalue(0,40,out);
                                }
                        }
                }
        }
        else
                dct_level_11(out);

}


void dct_level_11(int *out)
{

        if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(27,1,out);
                                        else
                                                obtainvalue(28,1,out);
                                }
                                else
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(29,1,out);
                                        else
                                                obtainvalue(30,1,out);
                                }
                        }
                        else
                        {
                                if(nextkeyword(1))
                                {
                                        if(nextkeyword(1))
                                                obtainvalue(31,1,out);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                else
                                        obtainvalue(11,2,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(12,2,out);
                                else
                                        obtainvalue(13,2,out);
                        }
                }
        }
        else
        {
                if(nextkeyword(1))
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        obtainvalue(14,2,out);
                                else
                                        obtainvalue(15,2,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(16,2,out);
                                else
                                        obtainvalue(6,3,out);
                        }
                }
                else
                {
                        if(nextkeyword(1))
                        {
                                if(nextkeyword(1))
                                        obtainvalue(1,15,out);
                                else
                                        obtainvalue(1,16,out);
                        }
                        else
                        {
                                if(nextkeyword(1))
                                        obtainvalue(1,17,out);
                                else
                                        obtainvalue(1,18,out);
                        }
                }
        }
}
else
{
        printf("\nDCT Variable length code ERROR !!\n");
        printf("At GOP>%d Picture>%d Slice>%d
Macroblock>%d",GOP_number,picture_number,slice_number,macroblock_number);
        stopchar=getch();
}
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void obtainvalue(int a,int b,int *z)
{
        int k;

        k=nextkeyword(1);

        if(k==1)
        {
                *z=-b;
                z++;
                *z=a;
        }
        else
        {
                *z=b;
                z++;
                *z=a;
        }

}


void escape_sequence(int *z)
{
        int k,l,i,j,run,level;

        i=0;j=0;
        k=0;l=0;

        while((nextbits(1)!=1)&&(k<6))
        {
                nextkeyword(1);
                k++;
        }

        run=nextkeyword(6-k);

        k=nextkeyword(8);
        if(k==128)
                level=-256+nextkeyword(8);
        else if(k==0)
                level=nextkeyword(8);
        else if(k>128)
                level=-256+k;
        else if(k<128)
                level=k;
        else
        {
                printf("Escape code ERROR !!\n");
                stopchar=getch();
        }

        *z=level;
        z++;
        *z=run;

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int pattern_code(int a)
{

        int value;

        value=0;
        if((code_block_pattern_value)&(1<<(5-a)))
                value=1;
        if(macroblock_intra)
                value=1;

        return value;
}



int slice_start_code(void)
{

        if(nextbits(32)==0x101)
                return 1;
        else if(nextbits(32)==0x102)
                return 1;
        else if(nextbits(32)==0x103)
                return 1;
        else if(nextbits(32)==0x104)
                return 1;
        else if(nextbits(32)==0x105)
                return 1;
        else if(nextbits(32)==0x106)
                return 1;
        else if(nextbits(32)==0x107)
                return 1;
        else if(nextbits(32)==0x108)
                return 1;
        else if(nextbits(32)==0x109)
                return 1;
        else if(nextbits(32)==0x10a)
                return 1;
        else if(nextbits(32)==0x10b)
                return 1;
        else if(nextbits(32)==0x10c)
                return 1;
        else if(nextbits(32)==0x10d)
                return 1;
        else if(nextbits(32)==0x10e)
                return 1;
        else if(nextbits(32)==0x10f)
                return 1;
        else if(nextbits(32)==0x110)
                return 1;
        else if(nextbits(32)==0x111)
                return 1;
        else if(nextbits(32)==0x112)
                return 1;
        else if(nextbits(32)==0x113)
                return 1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        else if(nextbits(32)==0x114)
                return 1;
        else if(nextbits(32)==0x115)
                return 1;
        else
                return 0;

}


void macroblock_type(void)
{

        if(picture_coding_type==1)
        {
                if(nextkeyword(1))
                {
                        macroblock_quant=0;
                        macroblock_motion_forward=0;
                        macroblock_motion_backward=0;
                        macroblock_pattern=0;
                        macroblock_intra=1;
                }
                else if(nextkeyword(1))
                {
                        macroblock_quant=1;
                        macroblock_motion_forward=0;
                        macroblock_motion_backward=0;
                        macroblock_pattern=0;
                        macroblock_intra=1;
                }
                else
                {
                        printf("Macroblock  type 1 ERROR !!\n");
                        stopchar=getch();
                }

        }
        else if(picture_coding_type==2)
        {
                if(nextkeyword(1))
                {
                        macroblock_quant=0;
                        macroblock_motion_forward=1;
                        macroblock_motion_backward=0;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
                else if(nextkeyword(1))
                {
                        macroblock_quant=0;
                        macroblock_motion_forward=0;
                        macroblock_motion_backward=0;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
                else if(nextkeyword(1))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        {
                macroblock_quant=0;
                macroblock_motion_forward=1;
                macroblock_motion_backward=0;
                macroblock_pattern=0;
                macroblock_intra=0;
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        macroblock_quant=0;
                        macroblock_motion_forward=0;
                        macroblock_motion_backward=0;
                        macroblock_pattern=0;
                        macroblock_intra=1;
                }
                else
                {
                        macroblock_quant=1;
                        macroblock_motion_forward=1;
                        macroblock_motion_backward=0;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
        }
        else if(nextkeyword(1))
        {
                macroblock_quant=1;
                macroblock_motion_forward=0;
                macroblock_motion_backward=0;
                macroblock_pattern=1;
                macroblock_intra=0;
        }
        else if(nextkeyword(1))
        {
                macroblock_quant=1;
                macroblock_motion_forward=0;
                macroblock_motion_backward=0;
                macroblock_pattern=0;
                macroblock_intra=1;
        }
        else
        {
                printf("Macroblock type 2 ERROR !!\n");
                stopchar=getch();
        }

}
else if(picture_coding_type==3)
{
        if(nextkeyword(1))
        {
                if(nextkeyword(1)==0)
                {
                        macroblock_quant=0;
                        macroblock_motion_forward=1;
                        macroblock_motion_backward=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                macroblock_pattern=0;
                macroblock_intra=0;
        }
        else
        {
                macroblock_quant=0;
                macroblock_motion_forward=1;
                macroblock_motion_backward=1;
                macroblock_pattern=1;
                macroblock_intra=0;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1)==0)
        {
                macroblock_quant=0;
                macroblock_motion_forward=0;
                macroblock_motion_backward=1;
                macroblock_pattern=0;
                macroblock_intra=0;
        }
        else
        {
                macroblock_quant=0;
                macroblock_motion_forward=0;
                macroblock_motion_backward=1;
                macroblock_pattern=1;
                macroblock_intra=0;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1)==0)
        {
                macroblock_quant=0;
                macroblock_motion_forward=1;
                macroblock_motion_backward=0;
                macroblock_pattern=0;
                macroblock_intra=0;
        }
        else
        {
                macroblock_quant=0;
                macroblock_motion_forward=1;
                macroblock_motion_backward=0;
                macroblock_pattern=1;
                macroblock_intra=0;
        }
}
else if(nextkeyword(1))
{
        if(nextkeyword(1))
        {
                macroblock_quant=0;
                macroblock_motion_forward=0;
                macroblock_motion_backward=0;
                macroblock_pattern=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        macroblock_intra=1;
                }
                else
                {
                        macroblock_quant=1;
                        macroblock_motion_forward=1;
                        macroblock_motion_backward=1;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
        }
        else if(nextkeyword(1))
        {
                if(nextkeyword(1))
                {
                        macroblock_quant=1;
                        macroblock_motion_forward=1;
                        macroblock_motion_backward=0;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
                else
                {
                        macroblock_quant=1;
                        macroblock_motion_forward=0;
                        macroblock_motion_backward=1;
                        macroblock_pattern=1;
                        macroblock_intra=0;
                }
        }
        else if(nextkeyword(1))
        {
                macroblock_quant=1;
                macroblock_motion_forward=0;
                macroblock_motion_backward=0;
                macroblock_pattern=0;
                macroblock_intra=1;
        }
        else
        {
                printf("Macroblock type 3 ERROR !!\n");
                stopchar=getch();
        }

}
else
{
        printf("Macroblock type ERROR !!\n");
        stopchar=getch();
}

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int nextkeyword(int a)
{
        int i,k,l;

        k=0;
        for(i=0;i<a;i++)
        {
                l=buffer();
                k<<=1;
                k=k+l;
        }
        return k;

}


int nextbit(int a)
{
        int i,k,l;

        k=0;
        for(i=0;i<a;i++)
        {
                l=buffer_code[i];
                k<<=1;
                k=k+l;
        }
        return k;
}


void next_start_code(void)
{

        while((mem_point%8)!=0)
                buffer();

        while(nextbits(24)!=1)
                nextkeyword(8);

}

void sub_next_start_code(void)
{

        while(bit_out!=9)
                sub_buffer();

        while(nextbit(24)!=1)
                sub_nextkeyword(8);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int nextbits(int a)
{
        int i,k,l,m,n;

        k=0;
        m=read_mem;
        for(i=mem_point;i<(mem_point+a);i++)
        {
                if(i<bit_amout)
                        n=i;
                else
                {
                        if(i==bit_amout)
                        {
                                n=0;
                                if(read_mem==0)
                                        m=1;
                                else if(read_mem==1)
                                        m=2;
                                else
                                        m=0;
                        }
                        else
                                n++;
                }
                l=video_buffer[m][n];
                k<<=1;
                k=k+l;
        }
        return k;
}

char buffer(void)
{
        char out;

        out=video_buffer[read_mem][mem_point];
        mem_point++;

/*
        if(test_space)
        {
                printf(" ");
        }
        printf("%d",out);
*/

        if(mem_point==bit_amout)
        {

                        printf("\n\n  ********** This is start of PACKET **********\n");
                        video_stream_flag=0;
                        sub_pack();

                        mem_point=0;
                        if(read_mem==0)
                                read_mem=1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                else if(read_mem==1)
                        read_mem=2;
                else
                        read_mem=0;

                bit_amout=buffer_length[read_mem]*8;

        }

        return out;

}


void Type_of_stream(void)
{
        int k,l;

        printf("\nstream_id = %0x\n",stream_id);

        if(stream_id==188)
                printf("This is reserve stream\n");
        else if(stream_id==189)
                printf("This is private stream 1 stream\n");
        else if(stream_id==190)
                printf("This is padding stream\n");
        else if(stream_id==191)
                printf("This is private stream 2 stream\n");
        else
        {
                l=stream_id;
                k=l>>5;
                if(k==6)
                        printf("This is audio stream\n");
                else
                {
                        k=l>>4;
                        if(k==14)
                        {
                                video_stream_flag=1;
                                printf("This is video stream\n");
                        }
                        else
                        {
                                printf("Type of stream ERROR or RESERVE\n");
                                stopchar=getch();
                        }

                }
        }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int sub_nextkeyword(int a)
{
        int i,k,l;

        k=0;
        for(i=0;i<a;i++)
        {
                l=sub_buffer();
                k<<=1;
                k=k+l;
        }
        return k;
}


char sub_buffer(void)
{
        int i,j,k;
        char ch,out;

        if(bit_out==0)
        {
                for(i=0;i<57;i=i+8)
                {
                        ch=fgetc(textfile);
                        for(j=8+i;j>i;j--)
                        {
                                k=ch&1;
                                buffer_code[j-1]=k;
                                ch>>=1;
                        }
                }
                bit_out=1;
        }

        if(bit_out==9)
        {
                ch=fgetc(textfile);
                for(i=64;i>56;i--)
                {
                        k=ch&1;
                        buffer_code[i-1]=k;
                        ch>>=1;
                }
                bit_out=1;
                byte_no++;
        }
        out=buffer_code[0];
        for(i=0;i<63;i++)
        {
                buffer_code[i]=buffer_code[i+1];
        }
        bit_out++;
        return out;

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void sub_pack(void)
{

loop1:

        if((nextbit(24)==1)&&(nextbit(32)!=442))
                goto loop2;

        if(sub_nextkeyword(32)!=442)
        //pack_start_code
        {
                printf("pack_start_code  ERROR !!!\n");
                stopchar=getch();
        }


        if(sub_nextkeyword(4)!=2)
        {
                printf("pack error\n");
                stopchar=getch();
        }

        sub_nextkeyword(3);
        sub_nextkeyword(1);
        sub_nextkeyword(15);
        sub_nextkeyword(1);
        sub_nextkeyword(15);
        sub_nextkeyword(1);
        sub_nextkeyword(1);
        sub_nextkeyword(22);
        sub_nextkeyword(1);

        if(nextbit(32)==443)
                system_header();

loop2:

        sub_packet();

        if(stream_jump_flag)
        {
                stream_jump_flag=0;
                goto loop1;
        }

}


void sub_packet(void)
{
        int i,j,k;
        char ch;

        if(sub_nextkeyword(24)!=1)
        {
                printf("packet_start_code_prefix  ERROR!! \n");
                stopchar=getch();
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
stream_id=sub_nextkeyword(8);
packet_length=sub_nextkeyword(16);

Type_of_stream();
printf("Packet length = %d\n\n",packet_length);

k=byte_no;
if(stream_id!=private_stream_2)
{
        while(nextbit(8)==255)
                sub_nextkeyword(8);
        if(nextbit(2)==1)
        {
                sub_nextkeyword(2);
                sub_nextkeyword(1);
                sub_nextkeyword(13);
        }
        if(nextbit(4)==2)
        {
                sub_nextkeyword(4);
                sub_nextkeyword(3);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
        }
        else if(nextbit(4)==3)
        {
                sub_nextkeyword(4);
                sub_nextkeyword(3);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
                if(sub_nextkeyword(4)!=1)
                {
                        printf("Packet  ERROR");
                        stopchar=getch();
                }
                sub_nextkeyword(3);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
                sub_nextkeyword(15);
                sub_nextkeyword(1);
        }
        else
        {
                if(sub_nextkeyword(8)!=15)
                {
                        printf("Packet ERROR !!");
                        stopchar=getch();
                }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
    }

    k=byte_no-k;
    packet_data_length=packet_length-k;
    packet_data_byte=0;

    if(video_stream_flag)
    {
        for(i=0;i<packet_data_length;i++)
        {

            if(i<7)
            {
                for(j=0;j<8;j++)
                    video_buffer[write_mem][i*8+j]=buffer_code[i*8+j];
            }
            else
            {
                ch=fgetc(textfile);
                for(j=0;j<8;j++)
                {
                    k=ch&1;
                    video_buffer[write_mem][8*(i+1)-(j+1)]=k;
                    ch>>=1;
                }
            }
        }

        for(i=0;i<57;i=i+8)
        {
            ch=fgetc(textfile);
            for(j=8+i;j>i;j--)
            {
                k=ch&1;
                buffer_code[j-1]=k;
                ch>>=1;
            }
        }
        bit_out=1;
        buffer_length[write_mem]=packet_data_length;
        if(write_mem==0)
            write_mem=1;
        else if(write_mem==1)
            write_mem=2;
        else
            write_mem=0;

    }
    else
    {
        stream_jump_flag=1;
        for(i=0;i<packet_data_length;i++)
            sub_nextkeyword(8);
    }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void dequantize_ipicture(int block_number)
{

        int i,j,k;

        if(block_number==0)
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                        {
                                k=zzscan[i][j];
                                block_out[i][j]=(2*data_out[k]*quantizer_scale*intra_quantizer_matrix
[i][j])/16;

                                if((block_out[i][j]&1)==0)
                                        block_out[i][j]=block_out[i][j]-Sign(block_out[i][j]);
                                if(block_out[i][j]>2047)
                                        block_out[i][j]=2047;
                                if(block_out[i][j]<-2047)
                                        block_out[i][j]=-2048;
                        }
                }
                block_out[0][0]=data_out[0]*8;
                if((macroblock_address-past_intra_address)>1)
                        block_out[0][0]=128*8+block_out[0][0];
                else
                        block_out[0][0]=dct_dc_y_past+block_out[0][0];
                dct_dc_y_past=block_out[0][0];
        }
        else if((block_number>0)&&(block_number<4))
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                        {
                                k=zzscan[i][j];
                                block_out[i][j]=(2*data_out[k]*quantizer_scale*intra_quantizer_matrix
[i][j])/16;

                                if((block_out[i][j]&1)==0)
                                        block_out[i][j]=block_out[i][j]-Sign(block_out[i][j]);
                                if(block_out[i][j]>2047)
                                        block_out[i][j]=2047;
                                if(block_out[i][j]<-2048)
                                        block_out[i][j]=-2048;
                        }
                }
                block_out[0][0]=dct_dc_y_past+(data_out[0]*8);
                dct_dc_y_past=block_out[0][0];
        }
        else if(block_number==4)
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                        {
                                k=zzscan[i][j];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        block_out[i][j]=(2*data_out[k]*quantizer_scale*intra_quantizer_matrix
[i][j])/16;

                        if((block_out[i][j]&1)==0)
                                block_out[i][j]=block_out[i][j]-Sign(block_out[i][j]);
                        if(block_out[i][j]>2047)
                                block_out[i][j]=2047;
                        if(block_out[i][j]<-2047)
                                block_out[i][j]=-2048;
                    }
                }
                block_out[0][0]=data_out[0]*8;
                if((macroblock_address-past_intra_address)>1)
                        block_out[0][0]=128*8+block_out[0][0];
                else
                        block_out[0][0]=dct_dc_cb_past+block_out[0][0];
                dct_dc_cb_past=block_out[0][0];
        }
        else if(block_number==5)
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                        {
                                k=zzscan[i][j];
                                block_out[i][j]=(2*data_out[k]*quantizer_scale*intra_quantizer_matrix
[i][j])/16;

                                if((block_out[i][j]&1)==0)
                                        block_out[i][j]=block_out[i][j]-Sign(block_out[i][j]);
                                if(block_out[i][j]>2047)
                                        block_out[i][j]=2047;
                                if(block_out[i][j]<-2047)
                                        block_out[i][j]=-2048;
                        }
                }
                block_out[0][0]=data_out[0]*8;
                if((macroblock_address-past_intra_address)>1)
                        block_out[0][0]=128*8+block_out[0][0];
                else
                        block_out[0][0]=dct_dc_cr_past+block_out[0][0];
                dct_dc_cr_past=block_out[0][0];
        }
}


void dequantize_nonipicture(int block_number)
{
        int m,n,i;

        for(m=0;m<8;m++)
        {
                for(n=0;n<8;n++)
                {
                        i=zzscan[m][n];

        block_out[m][n]=((2*data_out[i]+Sign(data_out[i]))*quantizer_scale*non_intra_quantizer_matrix
[m][n])/16;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        if((block_out[m][n]&1)==0)
                                block_out[m][n]=block_out[m][n]-Sign(block_out[m][n]);
                        if(block_out[m][n]>2047)
                                block_out[m][n]=2047;
                        if(block_out[m][n]<-2048)
                                block_out[m][n]=-2048;
                        if(data_out[i]==0)
                                block_out[m][n]=0;
                }
        }

}


void block_search(int block_number)
{
        int i,j,k;
        int right_half_for,down_half_for;
        int right_half_back,down_half_back;
        int right_for,down_for;
        int right_back,down_back;
        int b_row,b_column,sub_sampling;
        int pel_for[8][8],pel_back[8][8];


        if(block_number<4)
        {
                right_for=recon_right_for>>1;
                down_for=recon_down_for>>1;
                right_half_for=recon_right_for-(2*right_for);
                down_half_for=recon_down_for-(2*down_for);

                right_back=recon_right_back>>1;
                down_back=recon_down_back>>1;
                right_half_back=recon_right_back-(2*right_back);
                down_half_back=recon_down_back-(2*down_back);

        }
        else
        {
                right_for=(recon_right_for/2)>>1;
                down_for=(recon_down_for/2)>>1;
                right_half_for=recon_right_for/2-(2*right_for);
                down_half_for=recon_down_for/2-(2*down_for);

                right_back=(recon_right_back/2)>>1;
                down_back=(recon_down_back/2)>>1;
                right_half_back=recon_right_back/2-(2*right_back);
                down_half_back=recon_down_back/2-(2*down_back);
        }

        sub_sampling=16;
        if(block_number<4)
                k=0;
        else if(block_number==4)
                k=1;
        else
                k=2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(block_number==0)
{
        b_row=0;
        b_column=0;
}
else if(block_number==1)
{
        b_row=1;
        b_column=0;
}
else if(block_number==2)
{
        b_row=0;
        b_column=1;
}
else if(block_number==3)
{
        b_row=1;
        b_column=1;
}
else
{
        b_row=0;
        b_column=0;
        sub_sampling=8;
}


for(i=0;i<8;i++)
{
        for(j=0;j<8;j++)
        {
                if((!right_half_for)&&(!down_half_for))

    pel_for[i][j]=past_picture[k][slice_number*sub_sampling+b_column*8+i+down_for]
[macroblock_number*sub_sampling+b_row*8+j+right_for];

                if((!right_half_for)&&(down_half_for))

    pel_for[i][j]=floor((past_picture[k][slice_number*sub_sampling+b_column*8+i+down_for]
[macroblock_number*sub_sampling+b_row*8+j+right_for]+past_picture[k][slice_number*sub_sampling
+b_column*8+i+down_for+1][macroblock_number*sub_sampling+b_row*8+j+right_for])/2+0.5);

                if((right_half_for)&&(!down_half_for))

    pel_for[i][j]=floor((past_picture[k][slice_number*sub_sampling+b_column*8+i+down_for]
[macroblock_number*sub_sampling+b_row*8+j+right_for]+past_picture[k][slice_number*sub_sampling
+b_column*8+i+down_for][macroblock_number*sub_sampling+b_row*8+j+right_for+1])/2+0.5);

                if((right_half_for)&&(down_half_for))

    pel_for[i][j]=floor((past_picture[k][slice_number*sub_sampling+b_column*8+i+down_for]
[macroblock_number*sub_sampling+b_row*8+j+right_for]+past_picture[k][slice_number*sub_sampling
+b_column*8+i+down_for+1][macroblock_number*sub_sampling+b_row*8+j+right_for]+past_picture[k]
[slice_number*sub_sampling+b_column*8+i+down_for][macroblock_number*sub_sampling+b_row*8+j
+right_for+1]+past_picture[k][slice_number*sub_sampling+b_column*8+i+down_for+1]
[macroblock_number*sub_sampling+b_row*8+j+right_for+1])/4+0.5);
```

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                }
        }


        if(picture_coding_type==3)
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                        {
                                if((!right_half_back)&&(!down_half_back))

        pel_back[i][j]=next_picture[k][slice_number*sub_sampling+b_column*8+i+down_back]
[macroblock_number*sub_sampling+b_row*8+j+right_back];

                                if((!right_half_back)&&(down_half_back))

        pel_back[i][j]=floor((next_picture[k][slice_number*sub_sampling+b_column*8+i+down_back]
[macroblock_number*sub_sampling+b_row*8+j+right_back]+next_picture[k][slice_number*sub_samplin
g+b_column*8+i+down_back+1][macroblock_number*sub_sampling+b_row*8+j+right_back])/2+0.5);

                                if((right_half_back)&&(!down_half_back))

        pel_back[i][j]=floor((next_picture[k][slice_number*sub_sampling+b_column*8+i+down_back]
[macroblock_number*sub_sampling+b_row*8+j+right_back]+next_picture[k][slice_number*sub_samplin
g+b_column*8+i+down_back][macroblock_number*sub_sampling+b_row*8+j+right_back+1])/2+0.5);

                                if((right_half_back)&&(down_half_back))

        pel_back[i][j]=floor((next_picture[k][slice_number*sub_sampling+b_column*8+i+down_back]
[macroblock_number*sub_sampling+b_row*8+j+right_back]+next_picture[k][slice_number*sub_samplin
g+b_column*8+i+down_back+1][macroblock_number*sub_sampling+b_row*8+j+right_back]+next_pict
ure[k][slice_number*sub_sampling+b_column*8+i+down_back][macroblock_number*sub_sampling+b_r
ow*8+j+right_back+1]+next_picture[k][slice_number*sub_sampling+b_column*8+i+down_back+1]
[macroblock_number*sub_sampling+b_row*8+j+right_back+1])/4+0.5);

                        }
                }

        }

        if(picture_coding_type==2)
        {
                for(i=0;i<8;i++)
                {
                        for(j=0;j<8;j++)
                                pel[i][j]=pel_for[i][j];
                }

        }
        else
        {
                if((macroblock_motion_forward)&&(macroblock_motion_backward))
                {
                        for(i=0;i<8;i++)
                        {
                                for(j=0;j<8;j++)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                pel[i][j]=floor((pel_for[i][j]+pel_back[i][j])/2+0.5);
                        }
                }
                else if(macroblock_motion_forward)
                {
                        for(i=0;i<8;i++)
                        {
                                for(j=0;j<8;j++)
                                        pel[i][j]=pel_for[i][j];
                        }
                }
                else
                {
                        for(i=0;i<8;i++)
                        {
                                for(j=0;j<8;j++)
                                        pel[i][j]=pel_back[i][j];
                        }
                }
        }

}


void motion_rebuild(void)
{

        int complement_horizontal_forward_r,complement_vertical_forward_r;
        int complement_horizontal_backward_r,complement_vertical_backward_r;
        int right_little,right_big,down_little,down_big;
        int max,min;
        int new_vector;


        if(macroblock_motion_forward)
        {
                if((forward_f==1)||(motion_horizontal_forward_code==0))
                        complement_horizontal_forward_r=0;
                else
                        complement_horizontal_forward_r=forward_f-1-motion_horizontal_forward_r;

                if((forward_f==1)||(motion_vertical_forward_code==0))
                        complement_vertical_forward_r=0;
                else
                        complement_vertical_forward_r=forward_f-1-motion_vertical_forward_r;

                right_little=motion_horizontal_forward_code*forward_f;

                if(right_little==0)
                        right_big=0;
                else
                {
                        if(right_little>0)
                        {
                                right_little=right_little-complement_horizontal_forward_r;
                                right_big=right_little-(32*forward_f);
                        }
                        else
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                {
                        right_little=right_little+complement_horizontal_forward_r;
                        right_big=right_little+(32*forward_f);
                }
        }

        down_little=motion_vertical_forward_code*forward_f;
        if(down_little==0)
                down_big=0;
        else
        {
                if(down_little>0)
                {
                        down_little=down_little-complement_vertical_forward_r;
                        down_big=down_little-(32*forward_f);
                }
                else
                {
                        down_little=down_little+complement_vertical_forward_r;
                        down_big=down_little+(32*forward_f);
                }
        }

        max=(16*forward_f)-1;
        min=(-16*forward_f);

        new_vector=recon_right_for_prev+right_little;
        if((new_vector<=max)&&(new_vector>=min))
                recon_right_for=recon_right_for_prev+right_little;
        else
                recon_right_for=recon_right_for_prev+right_big;
        recon_right_for_prev=recon_right_for;
        if(full_pel_forward_vector)
                recon_right_for=recon_right_for<<1;

        new_vector=recon_down_for_prev+down_little;
        if((new_vector<=max)&&(new_vector>=min))
                recon_down_for=recon_down_for_prev+down_little;
        else
                recon_down_for=recon_down_for_prev+down_big;
        recon_down_for_prev=recon_down_for;
        if(full_pel_forward_vector)
                recon_down_for=recon_down_for<<1;
}
else
{
        if(picture_coding_type==2)
        {
                recon_right_for=0;
                recon_down_for=0;
        }
        else
        {
                recon_right_for=recon_right_for_prev;
                recon_down_for=recon_down_for_prev;
        }
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(macroblock_motion_backward)
{

        if((backward_f==1)||(motion_horizontal_backward_code==0))
                complement_horizontal_backward_r=0;
        else
                complement_horizontal_backward_r=backward_f-1-
motion_horizontal_backward_r;

        if((backward_f==1)||(motion_vertical_backward_code==0))
                complement_vertical_backward_r=0;
        else
                complement_vertical_backward_r=backward_f-1-motion_vertical_backward_r;

        right_little=motion_horizontal_backward_code*backward_f;
        if(right_little==0)
                right_big=0;
        else
        {
                if(right_little>0)
                {
                        right_little=right_little-complement_horizontal_backward_r;
                        right_big=right_little-(32*backward_f);
                }
                else
                {
                        right_little=right_little+complement_horizontal_backward_r;
                        right_big=right_little+(32*backward_f);
                }
        }

        down_little=motion_vertical_backward_code*backward_f;
        if(down_little==0)
                down_big=0;
        else
        {
                if(down_little>0)
                {
                        down_little=down_little-complement_vertical_backward_r;
                        down_big=down_little-(32*backward_f);
                }
                else
                {
                        down_little=down_little+complement_vertical_backward_r;
                        down_big=down_little+(32*backward_f);
                }
        }


        max=(16*backward_f)-1;
        min=(-16*backward_f);

        new_vector=recon_right_back_prev+right_little;
        if((new_vector<=max)&&(new_vector>=min))
                recon_right_back=recon_right_back_prev+right_little;
        else
                recon_right_back=recon_right_back_prev+right_big;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                recon_right_back_prev=recon_right_back;
                if(full_pel_backward_vector)
                        recon_right_back=recon_right_back<<1;

                new_vector=recon_down_back_prev+down_little;
                if((new_vector<=max)&&(new_vector>=min))
                        recon_down_back=recon_down_back_prev+down_little;
                else
                        recon_down_back=recon_down_back_prev+down_big;
                recon_down_back_prev=recon_down_back;
                if(full_pel_backward_vector)
                        recon_down_back=recon_down_back<<1;

        }
        else
        {
                recon_right_back=recon_right_back_prev;
                recon_down_back=recon_down_back_prev;
        }

}


void keep_picture(void)
{
        int i,j,k,l,m,n;
        int     cb_macroblock[22][16][16],cr_macroblock[22][16][16];
        int y_slice_block[16][352],cb_slice_block[16][352],cr_slice_block[16][352];
        int r_slice_block[16][353],g_slice_block[16][352],b_slice_block[16][352];
        char ch;

        for(i=0;i<15;i++)
        {
                for(l=0;l<2;l++)
                {
                        for(j=0;j<22;j++)
                        {
                                for(k=0;k<2;k++)
                                {
                                        for(m=0;m<8;m++)
                                        {
                                                for(n=0;n<8;n++)

        y_slice_block[m+l*8][n+k*8+j*16]=picture_out[i][j][k+l*2][m][n];
                                        }
                                }
                        }
                }

                for(j=0;j<22;j++)
                {
                        for(m=0;m<8;m++)
                        {
                                for(n=0;n<8;n++)
                                {
                                        cb_macroblock[j][m*2][n*2]=picture_out[i][j][4][m][n];
                                        cb_macroblock[j][m*2][n*2+1]=picture_out[i][j][4][m][n];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                cb_macroblock[j][m*2+1][n*2]=picture_out[i][j][4][m][n];
                                cb_macroblock[j][m*2+1][n*2+1]=picture_out[i][j][4][m][n];

                                cr_macroblock[j][m*2][n*2]=picture_out[i][j][5][m][n];
                                cr_macroblock[j][m*2][n*2+1]=picture_out[i][j][5][m][n];
                                cr_macroblock[j][m*2+1][n*2]=picture_out[i][j][5][m][n];
                                cr_macroblock[j][m*2+1][n*2+1]=picture_out[i][j][5][m][n];
                        }
                }
        }

        for(j=0;j<22;j++)
        {
                for(m=0;m<16;m++)
                {
                        for(n=0;n<16;n++)
                        {
                                cb_slice_block[m][n+j*16]=cb_macroblock[j][m][n]-128;
                                cr_slice_block[m][n+j*16]=cr_macroblock[j][m][n]-128;
                        }

                }
        }

/*
        printf("\n\n");
        for(m=0;m<16;m++)
        {
                for(n=0;n<16;n++)
                {
                        printf("%d ",cr_slice_block[m][n]);
                        if(l==15)
                                printf("\n");
                }
        }
        stopchar=getch();
*/

        for(m=0;m<16;m++)
        {
                for(n=0;n<352;n++)
                {
                        k=floor((16*i+m)/2);
                        l=floor(n/2);
                        recent_picture[0][16*i+m][n]=y_slice_block[m][n];
                        recent_picture[1][k][l]=cb_slice_block[m][n]+128;
                        recent_picture[2][k][l]=cr_slice_block[m][n]+128;
                }
        }

        for(m=0;m<16;m++)
        {
                for(n=0;n<352;n++)
                {
                        r_slice_block[m][n]=y_slice_block[m][n]+cr_slice_block[m][n]*1.402;
                        g_slice_block[m][n]=y_slice_block[m][n]+cb_slice_block[m][n]*(-
0.344)+cr_slice_block[m][n]*(-0.714);
                        b_slice_block[m][n]=y_slice_block[m][n]+cb_slice_block[m][n]*1.772;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                if(r_slice_block[m][n]>255)
                                        r_slice_block[m][n]=255;
                                if(g_slice_block[m][n]>255)
                                        g_slice_block[m][n]=255;
                                if(b_slice_block[m][n]>255)
                                        b_slice_block[m][n]=255;
                                if(r_slice_block[m][n]<0)
                                        r_slice_block[m][n]=0;
                                if(g_slice_block[m][n]<0)
                                        g_slice_block[m][n]=0;
                                if(b_slice_block[m][n]<0)
                                        b_slice_block[m][n]=0;

                        }
                }

                if(bpicture_number==8)
                {
                        for(m=0;m<16;m++)
                        {
                                for(n=0;n<352;n++)
                                {
/*
                                        if(m==15)
                                        {
                                                r_slice_block[m][n]=200;
                                                g_slice_block[m][n]=200;
                                                b_slice_block[m][n]=200;
                                        }

                                        if((n%16)==0)
                                        {
                                                r_slice_block[m][n]=200;
                                                g_slice_block[m][n]=200;
                                                b_slice_block[m][n]=200;
                                        }
*/
                                        ch=r_slice_block[m][n];
                                        fputc(ch,testfile);
                                        ch=g_slice_block[m][n];
                                        fputc(ch,testfile);
                                        ch=b_slice_block[m][n];
                                        fputc(ch,testfile);
                                }
                        }
                }
        }

        if(bpicture_number==8)
                fclose(testfile);

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int shift_right(int a,int b)
{
        int i,j,k,l,code[8];

        i=0;j=0;k=0;l=0;
        for (i=0;i<8;i++)
        {
                j=a&1;
                code[i]=j;
                a>>=1;
        }

        for(i=b;i>0;i--)
        {
                j=code[i-1];
                k<<=1;
                k=k+j;
        }
        return k;
}


int shift_left(int a,int b)
{
        int i,j,k,l,code[8];

        i=0;j=0;k=0;l=0;
        for (i=0;i<8;i++)
        {
                j=a&1;
                code[i]=j;
                a>>=1;
        }

        for(i=0;i<b;i++)
        {
                j=code[7-i];
                k<<=1;
                k=k+j;
        }
        return k;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.ไฟล์ dct.h

```
#include <io.h>
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <math.h>
#define pi 3.1415926

double Cal(int m);
void dct(int q[8][8],int x[8][8],int *z);
void idct(int z[8][8],int *zout);
int Sign(int a);


void dct(int q[8][8],int x[8][8],int *z)
{
        int k,l,i,j;
        double y[8][8];
        char m;

        for (i=0;i<8;i++){
                for (j=0;j<8;j++){
                        x[i][j]=x[i][j]-128;
                }
        }

        for (k=0;k<8;k++){
                for(l=0;l<8;l++){
                        y[k][l]=0;
                }
        }


        for (k=0;k<8;k++){
                for(l=0;l<8;l++){

                        for (i=0;i<8;i++){
                                for (j=0;j<8;j++){
                                y[k][l]=y[k][l]+Cal(k)*Cal(l)/4*x[i][j]*cos((2*i+1)*k*pi/16)*cos
((2*j+1)*l*pi/16);
                                }
                        }
                        y[k][l]=ceil(y[k][l]-0.5);
                }
        }

        for (k=0;k<8;k++){
                for(l=0;l<8;l++){
                        *z=floor((y[k][l]+floor(q[k][l]/2))/q[k][l]);
                        z++;
                }

        }

}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void idct(int z[8][8],int *zout)

{
        int k,l,i,j;
        double zi;

        zi=0;
        for (i=0;i<8;i++){
                for(j=0;j<8;j++){

                        for (k=0;k<8;k++){
                                for (l=0;l<8;l++){
                                zi=zi+Cal(k)*Cal(l)/4*z[k][l]*cos((2*i+1)*k*pi/16)*cos
((2*j+1)*l*pi/16);
                                }
                        }
                        *zout=ceil(zi-0.5);
                        zout++;
                        zi=0;
                }
        }

}



double Cal(int m)
{
        double a;

        if (m==0)
                a=1/sqrt(2);
        else
                a=1;

        return a;
}


int Sign(int a)
{
        if(a>0)
                return 1;
        else if(a<0)
                return -1;
        else if(a==0)
                return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้