



# วิทยานิพนธ์

การพัฒนาอัลกอริทึมการบีบอัดวิดีโอทัศนภาพถ่ายทางอากาศ  
สำหรับกล้องตรวจจับรังสีความร้อน(FLIR)  
ให้มีความสามารถชี้เป้าหมายทางยุทธวิธี

DEVELOPMENT OF VIDEO COMPRESSION ALGORITHM  
FOR AIRBORNE FORWARD LOOKING INFRARED (FLIR)  
EQUIPMENT WITH TACTICAL TARGETING CAPABILITY

นายสุชาติ ชุนทอง

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

พ.ศ. 2550



## ใบรับรองวิทยานิพนธ์

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมการบินและอวกาศ)

ปริญญา

วิศวกรรมการบินและอวกาศ

วิศวกรรมการบินและอวกาศ

สาขา

ภาควิชา

เรื่อง การพัฒนาอัลกอริทึมการบีบอัดวิดีโอที่สนับสนุนภาพถ่ายทางอากาศสำหรับกล้องตรวจจับรังสีความร้อน(FLIR) ให้มีความสามารถชี้เป้าหมายทางยุทธวิธี

Development of Video Compression Algorithm for Airborne Forward Looking Infrared (FLIR) Equipment with Tactical Targeting Capability

นามผู้วิจัย นายสุชาติ ขุนทอง

ได้พิจารณาเห็นชอบโดย

อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก

( นาวาอากาศเอก ผู้ช่วยศาสตราจารย์พารัตน์ สงวนโภคัย, Ph.D. )

อาจารย์ที่ปรึกษาวิทยานิพนธ์ร่วม

( นาวาอากาศตรี ธีรพล นิยมไทย, Ph.D. )

ประธานสาขาวิชา

( ผู้ช่วยศาสตราจารย์อวยชัย จีระชน, M.S. )

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์รับรองแล้ว

( รองศาสตราจารย์วินัย อางคงหาญ, M.A. )

คณบดีบัณฑิตวิทยาลัย

วันที่ ..... เดือน ..... พ.ศ. ....

วิทยานิพนธ์

เรื่อง

การพัฒนาอัลกอริทึมการบีบอัดวิดีโอทัศนภาพถ่ายทางอากาศสำหรับกล้องตรวจจับ  
รังสีความร้อน(FLIR) ให้มีความสามารถชี้เป้าหมายทางยุทธวิธี

Development of Video Compression Algorithm for Airborne Forward Looking Infrared (FLIR)  
Equipment with Tactical Targeting Capability

โดย

นายสุชาติ ขุนทอง

เสนอ

บัณฑิตวิทยาลัย มหาวิทยาลัยเกษตรศาสตร์

เพื่อความสมบูรณ์แห่งปริญญาวิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมการบินและอวกาศ)

พ.ศ. 2550

สุชาติ ขุนทอง 2550: การพัฒนาอัลกอริทึมการบีบอัดวิดีโอภาพถ่ายทางอากาศสำหรับ  
กล้องตรวจจับรังสีความร้อน(FLIR) ให้มีความสามารถชี้เป้าหมายทางยุทธวิธี ปรินญา  
วิศวกรรมศาสตรมหาบัณฑิต (วิศวกรรมการบินและอวกาศ) สาขาวิศวกรรมการบินและ  
อวกาศ ภาควิชาวิศวกรรมการบินและอวกาศ ปรธานกรรมการที่ปรึกษา:  
นาวาอากาศเอก ผู้ช่วยศาสตราจารย์พาหรัณ สงวนโกภัย, Ph.D. 75 หน้า

งานวิจัยชิ้นนี้มีวัตถุประสงค์เพื่อทำบีบอัดภาพวิดีโอ โดยจะต้องคงไว้ซึ่งคุณภาพของ  
วัตถุที่สนใจในทีนี้คือวัตถุที่เคลื่อนที่ส่วนวัตถุอื่น ๆ จะทำการบีบอัดตามมาตรฐานของกลุ่มผู้ชำนาญ  
ภาพเคลื่อนที่(MPEG) เพื่อนำไปใช้ประโยชน์ในทางการทหารกล่าวคือ ทางกองทัพอากาศมีงานวิจัย  
เรื่องอากาศยานไร้คนขับ(UAV) แล้วได้ทำการติดตั้งกล้องอินฟราเรดไปกับอากาศยานด้วย งานวิจัย  
ชิ้นนี้จึงสนับสนุนในเรื่องของการพัฒนาการบีบอัดข้อมูลภาพวิดีโอที่จะส่งลงมาบนภาคพื้น  
เพื่อทำให้ขนาดข้อมูลมีขนาดเล็กแต่คงไว้ซึ่งคุณภาพของวัตถุที่เคลื่อนที่

ในการทดลองเพื่อพัฒนาจะต้องแบ่งแยกระหว่างวัตถุที่เคลื่อนที่และวัตถุที่อยู่นิ่งซึ่งสามารถ  
แบ่งได้โดยใช้คุณสมบัติของการบีบอัดภาพตามมาตรฐานของ MPEG-4 ที่ทำการแบ่งวัตถุออกจาก  
กันก่อนที่จะทำการบีบอัด ซึ่งวัตถุแต่ละตัวจะถูกเรียกว่าพื้นของวัตถุวิดีโอ(Video Object Plane-  
VOP) โดยจะใช้วิธีการแบ่งแยกด้วยคุณสมบัติของเลข 0 และ 1 มาทำการคูณแต่ละพิกเซล จากนั้น  
ทำการบีบอัดแต่ละ VOP แล้วนำมารวมกัน

เมื่อทำการแบ่งแยกวัตถุเรียบร้อยแล้วจะทำการบีบอัดวัตถุที่ไม่สนใจตามขั้นตอนการบีบอัด  
ของ MPEG นั่นคือจะมีขั้นตอนแรกเป็นการแปลงโคไซน์ไม่ต่อเนื่อง (Discrete Cosine Transform)  
จากนั้นจึงนำข้อมูลที่ได้นำมาให้อยู่ในรูปเชิงปริมาณ(Quantization) แล้วจึงนำมาเข้ารหัสเอนโทรปี  
(Entropy Encoding) โดยจะต้องแปลงจากเมตริกซ์ในรูป 8x8 ให้เป็น 1x64 แล้วจึงนำมาเข้ารหัส  
การเข้ารหัสส่วนต่างและการเข้ารหัสแบบ run-length แล้วจึงนำวัตถุเข้ามารวมกันในขั้นตอน  
สุดท้าย

Suchart Khunthong 2007: Development of Video Compression Algorithm for Airborne Forward Looking Infrared (FLIR) Equipment with Tactical Targeting Capability. Master of Engineering (Aerospace Engineering), Major Field: Aerospace Engineering, Department of Aerospace Engineering. Thesis Advisor: Assistant Professor Pahron Sanguanbhokai, Ph.D. 75 pages.

The purpose of this research is to compress a video for military tactical requirement by maintaining the quality of the data of moving objects. Other objects are compressed by conforming to Moving Picture Expert Group (MPEG) compression standard. Royal Thai Air force are developing the Unmanned Aerial Vehicle (UAV) with an infrared camera as the tactical payload. This research supports the development of video compression to minimizes data transmission without losing tactical value.

The experiment isolates between moving objects and background objects by using the MPEG-4 compression standard that performs objects separation before compression. Each object is called Video Object Plane (VOP). Each VOP is compressed with different quality according to tactical value and then recombine together.

After object separation, we compressed disinterested object using MPEG-4 compression standard. First step is to use Discrete Cosine Transform then apply the Quantization to the result and encode data by using Entropy encoding. We have to convert 8x8 matrix to 1x64 matrix, apply the differential encoding method and run-length encoding. The last step is to recombine each object together.

..... / .....  
Student's signature                      Thesis Advisor's signature

## กิตติกรรมประกาศ

ขอกราบขอบพระคุณบิดามารดา รวมถึงญาติพี่น้องที่ช่วยสนับสนุนทั้งกำลังใจและกำลังใจ ให้กระผมได้ศึกษามาตลอดตั้งแต่เรียนระดับประถมศึกษา จนถึงการศึกษาในระดับปริญญาโท รวมถึงการให้คำปรึกษาที่ดีในการใช้ชีวิตให้เป็นคนดีของสังคมและประเทศชาติ

ขอกราบขอบพระคุณคณาจารย์ทุกท่านที่ช่วยอบรมและให้ความรู้ในทุกระดับการศึกษา โดยเฉพาะการศึกษาในระดับปริญญาโทที่ท่านให้ความรู้ทางด้านการบินไม่ว่าจะเป็นอาจารย์ที่ปรึกษา คืออาจารย์พาหุธรณ์ สงวนโกศล ที่ให้คำปรึกษาในการทำวิจัยชิ้นนี้มาตลอด อาจารย์ณัฐพล นิยมไทย ที่เป็นทั้งอาจารย์และพี่ และคณาจารย์ท่านอื่น ๆ ที่ยังไม่ได้เอ่ยนาม ณ ที่นี้

ขอขอบคุณเพื่อน ๆ พี่ ๆ ที่เรียนร่วมโครงการหลักสูตรนี้ไม่ว่าจะเป็น ยัย ต่อ พี่ต๋อง พี่กิ้ง พี่ตั้ม พี่หมู พี่เต๋าและพี่เอ๋ รวมถึงพี่น้องน้ำอ้อย ที่ได้ร่วมกันเรียน ร่วมกันเล่น ร่วมกันทำกิจกรรมตลอดหลักสูตร รวมทั้งเพื่อน ๆ ที่คอยให้กำลังใจทั้งที่สถาบันการศึกษาเดิมและเพื่อน ๆ พี่ ๆ ในที่ทำงานที่อนุญาตให้กระผมมาเรียน

สุชาติ ขุนทอง

เมษายน 2550

## สารบัญ

## หน้า

|                            |     |
|----------------------------|-----|
| สารบัญ                     | (1) |
| สารบัญตาราง                | (2) |
| สารบัญภาพ                  | (3) |
| คำนำ                       | 1   |
| วัตถุประสงค์               | 3   |
| การตรวจเอกสาร              | 4   |
| อุปกรณ์และวิธีการ          | 41  |
| อุปกรณ์                    | 41  |
| วิธีการ                    | 41  |
| ผลและวิจารณ์               | 48  |
| ผล                         | 48  |
| วิจารณ์                    | 56  |
| สรุปและข้อเสนอแนะ          | 57  |
| สรุป                       | 57  |
| ข้อเสนอแนะ                 | 58  |
| เอกสารและสิ่งอ้างอิง       | 59  |
| ภาคผนวก                    | 61  |
| ประวัติการศึกษาและการทำงาน | 75  |

## สารบัญตาราง

| ตารางที่ |                                                                                                        | หน้า |
|----------|--------------------------------------------------------------------------------------------------------|------|
| 1        | ตารางการจัดกลุ่มองค์ประกอบกระแสดวงและกระแสสลับ                                                         | 25   |
| 2        | ตารางการเข้ารหัสฮัฟฟ์แมนสำหรับองค์ประกอบกระแสดวง                                                       | 26   |
| 3        | ตารางแสดงคุณภาพการบีบอัดตามมาตรฐาน MPEG                                                                | 52   |
| 4        | ตารางแสดงคุณภาพการบีบอัดจากผลการวิจัย                                                                  | 53   |
| 5        | ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปร<br>คุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 24 | 54   |
| 6        | ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปร<br>คุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 20 | 54   |
| 7        | ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปร<br>คุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 18 | 55   |

## สารบัญภาพ

| ภาพที่ |                                                                        | หน้า |
|--------|------------------------------------------------------------------------|------|
| 1      | ตัวอย่างภาพลูกสุนัขจากกล้องอิรฟราเรด                                   | 8    |
| 2      | ตัวอย่างภาพกล้อง FLIR ที่ติดกับเครื่อง Cessna                          | 8    |
| 3      | องค์ประกอบของ MPEG-4                                                   | 10   |
| 4      | โครงสร้างภาพใน MPEG-4                                                  | 11   |
| 5      | โครงสร้างของการเข้ารหัส VO ของ MPEG-4                                  | 12   |
| 6      | โครงสร้างของการถอดรหัส VO ของ MPEG-4                                   | 12   |
| 7      | ลำดับชั้นของโครงสร้างข้อมูลใน MPEG-4                                   | 13   |
| 8      | (ก) แสดงบริเวณการค้นหา (ข) และการวิเคราะห์และเทียบแต่ละ MB             | 15   |
| 9      | TSS โดยจะแทน ตัวเลขเป็นการค้นหา โดยให้จุดดำเป็น object ที่ต้องการค้นหา | 16   |
| 10     | TSS รูปแบบของ Diamond Search                                           | 17   |
| 11     | 4:4:4 (Cb/Cr same as luma)                                             | 19   |
| 12     | 4:2:2 (1/2 the Luma Samples)                                           | 20   |
| 13     | 4:1:1 (1/4 the Luma samples)                                           | 20   |
| 14     | 4:2:0 (1/4 the Luma samples)                                           | 21   |
| 15     | สัมประสิทธิ์ความถี่ขนาด หลังผ่านการแปลง DCT                            | 22   |
| 16     | ลำดับการทำ Zig-Zag Scan                                                | 24   |
| 17     | หลักการทางเครื่องเรตินาสแกนของ George                                  | 27   |
| 18     | Paul Nipkow และ Nipkow disk                                            | 27   |
| 19     | ประเภทเทคโนโลยีไอซีอาร์                                                | 33   |
| 20     | อุปกรณ์รู้จำแบบออนไลน์                                                 | 34   |
| 21     | ตัวอักษรที่มีส่วนเชื่อมติดกันและขาดจากกัน                              | 35   |
| 22     | โครงสร้างทั่วไปของระบบไอซีอาร์                                         | 36   |
| 23     | แสดงโครงสร้างของ Neural Network                                        | 40   |
| 24     | ขั้นตอนการบีบอัดข้อมูล                                                 | 41   |
| 25     | การแยกพื้นหน้า, (ก) Original, (ข) Segment, (ค) Foreground              | 42   |
| 26     | การแยกพื้นหลัง, (ก) Original, (ข) Segment, (ค) Foreground              | 42   |

## สารบัญภาพ (ต่อ)

| ภาพที่ |                                                                                                                                                           | หน้า |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|------|
| 27     | ตัวอย่างภาพที่ถ่ายด้วยกล้อง FLIR                                                                                                                          | 43   |
| 28     | การปรับภาพให้อยู่ในรูปความเข้มของแสง (Intensity)                                                                                                          | 44   |
| 29     | Segment ที่ได้จากการกรองภาพ FLIR                                                                                                                          | 44   |
| 30     | การเข้ารหัสพื้นที่หลัง                                                                                                                                    | 45   |
| 31     | Block Diagram ของการเข้ารหัส                                                                                                                              | 45   |
| 32     | Block Diagram ของการถอดรหัส                                                                                                                               | 46   |
| 33     | (ก) Video Input และ (ข) Video Output หลังจาก Compression                                                                                                  | 46   |
| 34     | (ก) Foreground, (ข) Background Compression, (ค) Original Video และ (ง) Video Output                                                                       | 47   |
| 35     | ผลการบีบอัดจากการปรับค่าตัวแปรคุณภาพ เท่ากับ 265 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย                           | 48   |
| 36     | ผลการบีบอัดจากการปรับค่าตัวแปรคุณภาพ เท่ากับ 128 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย                           | 49   |
| 37     | ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 64 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย                                  | 49   |
| 38     | ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 32 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย                                  | 49   |
| 39     | ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 16 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย                                  | 50   |
| 40     | แผนภาพแสดงอัตราการบีบอัดของวิดีโอที่คำนวณจากการปรับค่าตัวแปรคุณภาพตามมาตรฐาน MPEG                                                                         | 50   |
| 41     | แผนภาพแสดงอัตราการบีบอัดของวิดีโอที่คำนวณจากการปรับค่าตัวแปรคุณภาพจากผลการวิจัย                                                                           | 51   |
| 42     | ภาพแสดงความละเอียดของภาพถ่ายวิดีโอที่คำนวณจากการบีบอัดด้วยค่าตัวแปรคุณภาพเท่ากับ 16 (ก) ภาพไม่ถูกบีบอัด (ข) บีบอัดตามมาตรฐาน MPEG (ค) บีบอัดจากผลการวิจัย | 52   |

## สารบัญภาพ (ต่อ)

| ภาพผนวกที่                                                        | หน้า |
|-------------------------------------------------------------------|------|
| 1 โปรแกรมหลัก (Main)                                              | 62   |
| 2 โมดูลการตรวจจับ (Car tracking)                                  | 62   |
| 3 แสดงผล (Display Results)                                        | 63   |
| 4 การบีบอัด (Compress)                                            | 64   |
| 5 พื้นหน้า (Foreground)                                           | 65   |
| 6 พื้นหลัง (Background)                                           | 65   |
| 7 การเข้ารหัส (Encoder)                                           | 66   |
| 8 การหน่วงเวลา (Delay and Pad)                                    | 67   |
| 9 ประมวลผลบล็อก (Block Processing)                                | 67   |
| 10 การเข้ารหัสการชดเชยการเคลื่อนที่ (Motion Compensation Encoder) | 68   |
| 11 เลือกการเข้าคู่ (Select Match)                                 | 69   |
| 12 ตัวแปลงรหัส (Transform Coder)                                  | 70   |
| 13 การถอดรหัส (Decoder)                                           | 70   |
| 14 เปลี่ยนหน่วย (To Unit)                                         | 71   |
| 15 อัตราบิต (Bit Rate)                                            | 72   |
| 16 การแบ่งแยกภาพ                                                  | 73   |
| 17 การแบ่งแยกวิดีโอ                                               | 73   |

## คำอธิบายสัญลักษณ์และคำย่อ

|       |   |                                                |
|-------|---|------------------------------------------------|
| AOVs  | = | Audio/Visual Objects                           |
| ARPS  | = | Adaptive Rood Pattern Search                   |
| BMC   | = | Block-base Motion Compensation                 |
| BPDC  | = | Binary Pixel Difference Classification         |
| BPM   | = | Bit-Plane Mapping Criterion                    |
| DCT   | = | Discrete Cosine Transform                      |
| DPC   | = | Difference Pixel Count Criterion               |
| DS    | = | Diamond Search                                 |
| ES    | = | Exhaustive search                              |
| FLIR  | = | Forward Looking Infrared                       |
| FSS   | = | Four Step Search                               |
| GMC   | = | Global Motion Compensation                     |
| GOP   | = | Group of Picture                               |
| H.261 | = | Video codec for audiovisual services           |
| IDCT  | = | Inverse Discrete Cosine Transform              |
| IEC   | = | Electrotechnical Commission                    |
| ISO   | = | International Organization for Standardization |
| JPEG  | = | Joint Photographic Experts Group               |
| MAE   | = | Mean Absolute Error                            |
| MB    | = | Macroblocks                                    |
| MPEG  | = | Motion Picture Experts Group                   |
| MSE   | = | Mean Square Error                              |
| NTSS  | = | New Three Step Search                          |
| OBMC  | = | Overlapped Block Motion Compensation           |
| OCR   | = | Optical Character Recognition                  |
| PDC   | = | Pixel Difference Classification                |
| PHODS | = | Parallel Hierarchical One-Dimensional Search   |
| RGB   | = | red/green/blue                                 |

|      |   |                                    |
|------|---|------------------------------------|
| SES  | = | Simple and Efficient Search        |
| SPMC | = | Sub-pixel Motion Compensation      |
| TDLS | = | Two-Dimensional Logarithmic Search |
| TSS  | = | Tree Step Search                   |
| VO   | = | Video Object                       |
| VOL  | = | Video Object Layer                 |
| VOPs | = | Video Object Planes                |

**การพัฒนาอัลกอริทึมการบีบอัดวิดีโอภาพถ่ายทางอากาศสำหรับกล้องตรวจจับ  
รังสีความร้อน(FLIR) ให้มีความสามารถชี้เป้าหมายทางยุทธวิธี**

**Development of Video Compression Algorithm for Airborne Forward Looking  
Infrared (FLIR) Equipment with Tactical Targeting Capability**

**คำนำ**

ปัจจุบันการพัฒนาการรับส่งข้อมูลจากอากาศยานสู่อากาศยาน หรืออากาศยานสู่ภาคพื้นมีความสำคัญอย่างยิ่ง เพราะข้อมูลที่ได้สามารถนำมาใช้ประโยชน์ในด้านต่าง ๆ มากมาย เช่น การวิเคราะห์ข้อมูลที่ได้เพื่อปรับสมดุลอากาศยาน การวิเคราะห์ข้อมูลภูมิประเทศจากภาพถ่ายทางอากาศ การสื่อสาร และข้อมูลทางทหาร ฯลฯ เป็นต้น ฉะนั้นข้อมูลที่ต้องการรับส่งข้างต้นนี้จำเป็นต้องให้มีขนาดเล็กที่สุดแต่คงไว้ซึ่งคุณภาพสูงสุดโดยอาศัยวิธีการทางคณิตศาสตร์เข้ามาจัดการเพื่อนำประโยชน์นี้มาทำการบีบอัด(Compress) ข้อมูลก่อนทำการรับส่ง โดยเฉพาะข้อมูลที่เป็นรูปแบบของภาพ เคลื่อนไหว ซึ่งปัจจุบันได้มีเทคโนโลยีเข้ามาสนับสนุนให้ข้อมูลมีความคมชัดมีคุณภาพสูง และขนาดเล็กเช่น ข้อมูลชนิด Moving JPEG (Moving Joint Photographic Experts Group), MPEG (Motion Picture Experts Group) และ H.261 (Video codec for audiovisual services) เป็นต้น โดยในโครงการวิทยานิพนธ์เล่มนี้จะเอาคุณสมบัติของเทคโนโลยีในรูปแบบของข้อมูลชนิด MPEG มาใช้ประโยชน์

รูปแบบของการนำเทคโนโลยี MPEG ซึ่งในที่นี้จะใช้ MPEG-4 มาประยุกต์ใช้คือ การถ่ายภาพจากอากาศยานด้วยกล้องอินฟราเรด ให้เก็บข้อมูลในรูปแบบ MPEG-4 โดยจะแบ่งองค์ประกอบของภาพออกเป็นวัตถุ(Object) โดยจะแยกการบีบอัดแต่ละวัตถุด้วยคุณภาพที่ต่างกัน โดยวัตถุที่เคลื่อนไหว จะทำการบีบอัดด้วยให้มีคุณภาพสูงกว่าวัตถุที่หยุดนิ่ง ซึ่งโดยปกติการบีบอัดข้อมูลตามมาตรฐานของ MPEG 4 จะทำการบีบอัดทุกวัตถุในแต่ละภาพ(Frame) ด้วยคุณภาพที่เท่ากันโดยยึดตามความสามารถการมองเห็นของมนุษย์เป็นหลัก งานวิจัยชิ้นนี้จึงเน้นหาวิธีการพัฒนาอัลกอริทึมบีบอัดวิดีโอในส่วน of วัตถุที่เคลื่อนไหวให้มีความละเอียดมากขึ้น เพื่อเมื่อนำมาขยายข้อมูลที่ได้อาจมีความชัดเจน โดยความสามารถนี้มีความสำคัญอย่างยิ่งกับการนำไปใช้ในยุทธการทางทหาร เช่น ให้อากาศยานทำการบันทึกข้อมูลของผู้ขนส่งสินค้าผิดกฎหมาย จากนั้นเมื่อนำข้อมูลที่ได้จากการถ่ายมาตรวจสอบก็สามารถขยายภาพเพื่อให้เห็นหน้าตาของผู้ขนส่งได้ชัดเจน ซึ่งปัจจุบันโครงการ UAV ของกระทรวงกลาโหมได้สนใจในงานส่วนนี้เนื่องจากขณะนี้ภาพที่ได้มานั้นเมื่อขยายจะทำให้ภาพ

ที่แตก ไม่สามารถมองได้ชัดเจน อันเนื่องมาจากภาพที่ได้เป็นภาพที่ได้จากกล้องวิดีโอที่มีความละเอียดต่ำ เพราะจำเป็นต้องทำการส่งข้อมูลผ่านระบบการเชื่อมโยงข้อมูล(Data Link) ที่มีช่องความกว้างสัญญาณ(Bandwidth) ที่จำกัด ดังนั้นวิธีการที่ดีกว่าคือการบีบอัดสัญญาณวิดีโอ (Video Compression) ให้มีขนาดข้อมูลที่เล็กแต่คงไว้ซึ่งคุณภาพที่ดีก่อนจะส่งข้อมูลผ่านระบบการเชื่อมโยงข้อมูล (Data Link) แล้วจึงนำมาขยาย

งานวิจัยชิ้นนี้มุ่งเน้นการแยกองค์ประกอบของภาพให้เป็นวัตถุที่เรียกว่า Video Object Planes (VOPs) ด้วยวิธีการประเมินการเคลื่อนไหว (Motion Estimation) แล้วนำวัตถุที่ได้มาทำการบีบอัด แต่จะไม่บีบอัดวัตถุที่มีการเคลื่อนไหวให้มีการสูญเสียคุณภาพ (Lossy Compression) ของภาพไป ซึ่งเมื่อนำวัตถุมาขยายจะทำให้เห็นภาพได้อย่างชัดเจน

## วัตถุประสงค์

1. เพื่อพัฒนาการบีบอัดข้อมูลภาพเคลื่อนไหว (Video Compression) ที่เหมาะสมกับการใช้เป้าหมายทางยุทธวิธี
2. เพื่อพัฒนาขั้นตอนวิธี (Algorithm) การขยายภาพเคลื่อนไหวโดยวิธีการทางการประมวลผลภาพ (Image Processing) ตามข้อมูลรูปแบบ MPEG-4
3. เพื่อนำผลการวิจัยที่ได้ไปประยุกต์ใช้ในการถ่ายภาพและรับส่งข้อมูลของโครงการ FLIR

## การตรวจเอกสาร

### 1. Forward Looking Infrared System (FLIR System)

#### 1.1 ภาพรวม

**FLIR** เป็นระบบกล้องที่ใช้ในการตรวจการณ์ และลาดตระเวนทางอากาศ สามารถติดตั้งกับอากาศยาน ได้ทั้ง บ.,ส. และ UAV ระบบ FLIR ติดตั้ง Sensors ได้หลายชนิด แต่โดยทั่วไป Sensors หลัก (Bawornchai, 2006) จะประกอบด้วย

1.1.1 Thermal Imager (TI) หรือมักนิยมเรียกว่า FLIR

1.1.2 กล้องวิดีโอสี ชนิดเลนส์ Spotter หรือ Scope

1.1.3 นอกจากนี้ยังมีอุปกรณ์อื่น ๆ อีก เช่น

ก. Laser Illuminator Pointer

ข. Laser Rangefinder

#### 1.2 ประโยชน์

ดังนั้นจากการที่ระบบ FLIR มี Sensors หลายชนิด จึงทำให้ FLIR ปฏิบัติการลาดตระเวนและบันทึกภาพ ได้ทั้งกลางวันและกลางคืน และมีความเหมาะสมกับการกิจต่าง ๆ เช่น

1.2.1 การลาดตระเวนทางอากาศในเวลากลางวัน กลางคืน

1.2.2 การลาดตระเวนค้นหาและช่วยชีวิต

1.2.3 การลาดตระเวนทางทะเล

1.2.4 การลาดตระเวนใช้อาวุธ

1.2.5 การลาดตระเวนหาผู้กระทำความผิดกฎหมายตามแนวชายแดน

1.2.6 การกิจเกี่ยวกับภัยพิบัติต่าง ๆ

1.2.7 การควบคุมความสงบเรียบร้อยภายในเมือง

### 1.3 พิจารณาความสามารถและสมรรถนะ

FLIR จะประกอบด้วย 2 ส่วน คือ ส่วนที่ติดตั้งกับอากาศยาน (Airborne) และส่วนภาคพื้น (Ground Station)

#### 1.3.1 ส่วนที่ติดกับอากาศยาน จะแบ่งออกเป็น 3 ส่วน คือ

ก. Turret หรือ Gimbal ซึ่งเป็นส่วนที่ติดตั้งภายนอกลำตัวอากาศยาน มีลักษณะเป็นลูกกลม ๆ ยึดติดกับแท่น (ติดตั้งในแนวระดับกับลำตัว อากาศยาน หรือ ใต้อากาศยาน) ภายใน Gimbal จะบรรจุ Sensors ชนิดต่าง ๆ ได้ ตามจำนวนช่องที่เจาะไว้บน Gimbal โดยตัว Gimbal จะมีคุณลักษณะ ดังนี้

- 1) หมุนได้รอบตัว หรือ 360 องศา
- 2) เหยยได้ประมาณไม่เกิน 30 องศา
- 3) ก้มได้ประมาณไม่เกิน 120 องศา
- 4) ทนต่อสภาวะการบินสูง, อุณหภูมิ, ความชื้น และป้องกันฟ้าและหยดน้ำ
- 5) ทนต่อการบินด้วยความเร็ว
- 6) มีระบบปรับแก้การเอียงของภาพ ซึ่งเกิดจากการเอียงของอากาศยาน
- 7) มีระบบป้องกันการสั่นของภาพ (Jitter)

(Stabilization)

ข. ส่วนควบคุมการทำงานหลัก (Master Control Unit) เป็นส่วนควบคุมการทำงานของ Turret Gimbal ส่งสัญญาณภาพไปยังจอรับภาพ การคำนวณทิศทาง ระยะทาง หรืออื่น ๆ การปรับแก้ไขภาพให้ชัดเจน การบันทึกข้อมูลลงในหน่วยความจำ การติดตามเป้าหมายเคลื่อนที่หรืออยู่นิ่งโดยอัตโนมัติ และระบบ Moving Map เป็นต้น

ก. ชุดควบคุมการทำงานด้วยมือ (Hand Control Unit) เป็นชุดควบคุมแยกส่วนออกมาจาก ส่วนควบคุมหลัก จะมีลักษณะเป็นชนิดมือถือ หรือ Join Stick ให้ จนท. Operator หรือนักบิน ควบคุมการทำงานของกล้อง การบันทึก และการบังคับทิศทางของกล้อง เป็นต้น

1.3.2 ส่วนภาคพื้น (Ground Station) เป็นส่วนนำข้อมูลภาพที่บันทึกมาทำการแปลความภาพถ่าย และรายงานให้ผู้เกี่ยวข้องทราบ อีกทั้งยังเป็นส่วนที่ใช้ในการวางแผนการบินอีกด้วย จะประกอบด้วย

ก. ส่วน Exploitation เป็นส่วนที่ใช้ในงานแปลความภาพถ่าย และใช้งานภาพถ่ายต่าง ๆ ซึ่งจะประกอบด้วย คอมพิวเตอร์ เครื่องพิมพ์ภาพ และรายงาน โปรแกรมการแปลความภาพถ่าย และระบบฐานข้อมูล (Data Base)

ข. ส่วน Mission Planning เป็นส่วนที่ใช้ในการวางแผนการบินและการบันทึกภาพ

หมายเหตุ นอกจากนี้ระบบ FLIR ยังสามารถใช้ข้อมูลแบบ Real Time ได้ โดยติดตั้งชุด Data Link ระหว่างอากาศยาน กับ Ground Station ได้

## 1.4 คุณสมบัติทั่วไปและขีดความสามารถของ Sensors ในระบบ FLIR

### 1.4.1 Thermal Imager หรือ FLIR

ก. บันทึกภาพด้วยความแตกต่างระหว่างอุณหภูมิของ Target และ Background โดยสามารถกำหนดให้ บริเวณที่มีอุณหภูมิสูงกว่าแสดงภาพออกมาเป็นสีขาว หรือ ดำ ได้ตามต้องการ บันทึกภาพด้วยความร้อนได้ทั้งเวลากลางวันและกลางคืน

ข. ใช้ Detector เป็นชนิด InSB ซึ่งรับคลื่นความร้อนในช่วง MWIR (Midle Wave IR) มีความยาวคลื่นประมาณ 3-5 ไมโครเมตร (ไมครอน) ซึ่งเป็นช่วงคลื่นความร้อนที่เหมาะสมกับการบันทึกภาพด้วยความร้อน ในภูมิภาคร้อนขึ้น

ค. Detector เป็นรุ่น Generation 3 ชนิด Focal plane array หรือ staring array ซึ่งให้รายละเอียดของภาพสูง (High Resolution) 640 x 480 Pixel

ง. สามารถเลือกมุมมองของภาพได้ หลายระดับ (Field of Vies .FOV)

จ. มีระบบปรับปรุงภาพให้ชัดเจนยิ่งขึ้น (Image Processing)

#### 1.4.2 กล้องวิดีโอชนิดซูมเลนส์

ก. เป็นกล้องวิดีโอสี ชนิด 3-CCD ซึ่งให้รายละเอียดของภาพสูง ประมาณ 750 TV lines หรือ มากกว่า

ข. มีความไวต่อแสงสูงน้อยกว่า 1 lux ซึ่งได้ขีดความสามารถในการถ่ายภาพ บริเวณที่มีแสงน้อย ๆ ได้

ค. ใช้เลนส์ซูมที่มีอัตราการซูมสูง (ประมาณ 15 เท่า หรือมากกว่า) และยังมี ส่วนต่อเพิ่มอัตราการซูมให้มากขึ้น (Switchable Extender)

#### 1.4.3 กล้องวิดีโอชนิด เลนส์ Spotter หรือ Scope

ก. เป็นกล้องวิดีโอสีชนิด 3-CCD หรือชนิด Monochrome

ข. กล้องวิดีโอสีจะให้รายละเอียดของภาพได้สูงกว่า

ค. ขีดความสามารถในการถ่ายภาพได้ไกลขึ้นอยู่กับ ความยาวโฟกัสของเลนส์ (Focal Length)

#### 1.4.4 Laser Illuminator Pointer

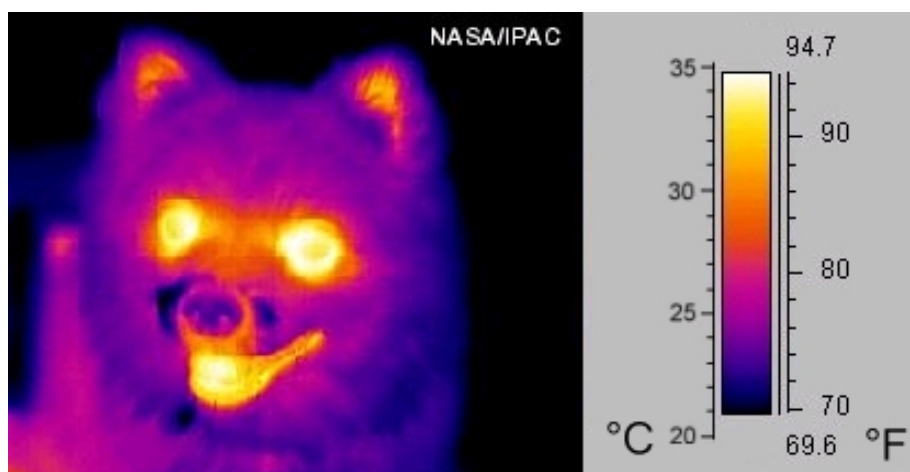
ก. ทำหน้าที่ชี้เป้า หรือ ส่องสว่างในจุดที่ต้องการ

ข. ใช้ควบคู่กับกล้องมองภาพกลางคืน เช่น Night Vision Goggle (NVG) ซึ่ง ถ้ามองด้วยตาเปล่าจะมองไม่เห็น

### 1.4.5 Laser Rangefinder

- ก. ทำหน้าที่ วัดระยะห่างของเป้าหมาย
- ข. วัดระยะได้ไกลสุด ไม่ต่ำกว่า 10 ก.ม.ให้ความถูกต้อง บวก/ลบ 5 ม.
- ค. ใช้ประกอบกับการบินลาดตระเวนใช้อาวุธ

โดยทั่ว ๆ ไปแล้ว กล้อง Thermal Imager หรือ FLIR สามารถจับเป้าหมายบุคคลในระดับ Detection ได้ ในระยะ (Slant Length) ประมาณ 10 ก.ม. ถ้าเป็นเป้าหมายขนาดใหญ่ และมีความร้อนแตกต่างจาก Background มาก ๆ ระยะทางที่ Detect ก็จะได้ไกลขึ้น และในระดับที่สูงขึ้นถึง Recognition และ Identification



ภาพที่ 1 ตัวอย่างภาพลูกสุนัขจากกล้องอินฟราเรด



ภาพที่ 2 ตัวอย่างภาพกล้อง FLIR ที่ติดกับเครื่อง Cessna

## 2. MPEG-4

### 2.1 ภาพรวม

MPEG ย่อมาจาก Moving Picture Expert Group โดยจะทำงานภายใต้ข้อกำหนดของ the International Organization for Standardization (ISO) และ the International Electrotechnical Commission (IEC) ซึ่ง MPEG-4 มีวัตถุประสงค์ที่สร้างมาตรฐานให้มีประสิทธิภาพ ยืดหยุ่นและเพื่อการพัฒนาในอนาคตได้ (Effelsberg and Steinmetz, 1998)

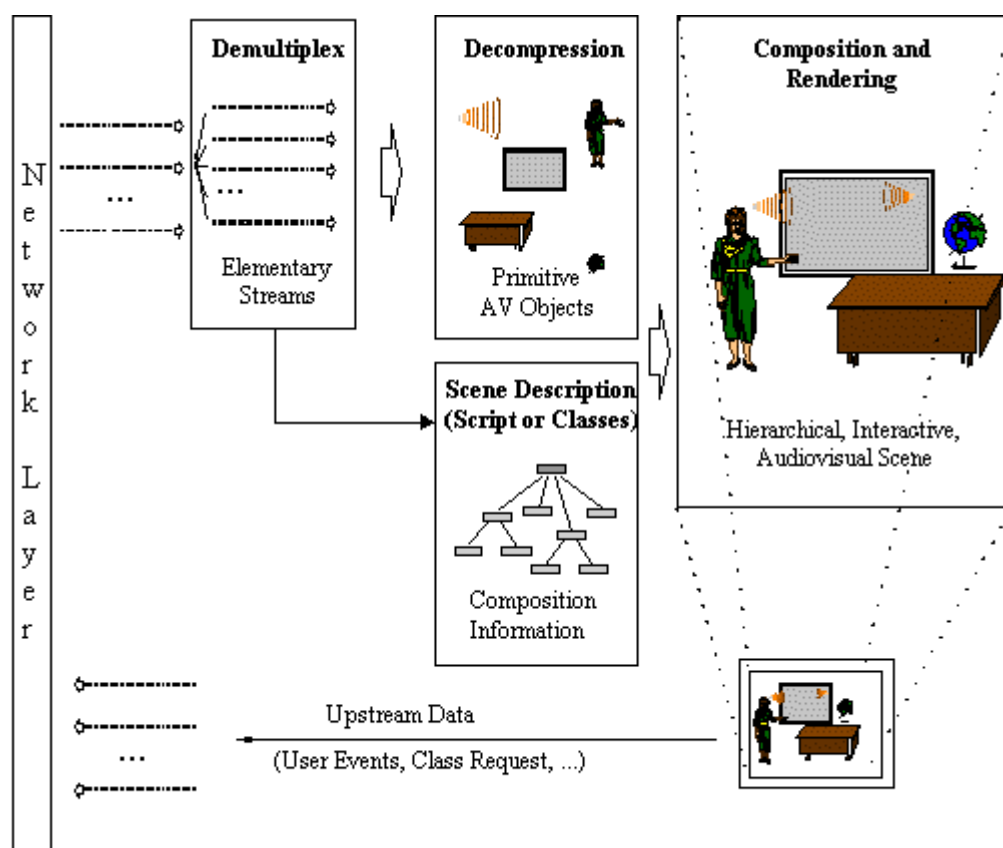
**2.1.1 Support for low bit –rate application:** MPEG-1 และ MPEG-2 จะมีประสิทธิภาพเพียง 1 เมกกะบิตต่อวินาที(Mbps) ซึ่งไม่เพียงพอกับงานที่มีความต้องการถ่ายโอนข้อมูลที่เร็วกว่า

**2.1.2 Support for object-based coding:** มาตรฐาน MPEG-4 จะมีการเขียนในรูปแบบ Object-Base หรือ Content-Base ซึ่งสามารถแบ่งวัตถุ ให้อยู่ในกลุ่มของวัตถุพื้นหน้า และวัตถุพื้นหลัง แทนที่การใช้ทั้งภาพ โดยการเขียนโปรแกรมแบบนี้จะเปิดกว้าง เช่น ไม่ขึ้นอยู่กับวิธีการเขียนโปรแกรม ในวัตถุที่ต่างกัน, สามารถใช้จากเดิมได้, การนำไปประกอบได้หลายส่วน และมีความสัมพันธ์ระหว่างวัตถุ

**2.1.3 Toolkit based coding:** MPEG-1 จะมีข้อจำกัดในเรื่องของความยืดหยุ่น(Flexible) ของการเขียนโปรแกรม ส่วน MPEG-2 เริ่มที่จะพัฒนาให้มีความยืดหยุ่นมากขึ้นด้วยการรวมความแตกต่างของหลาย ๆ Application ไว้ด้วยกัน ในขณะที่ MPEG-4 ได้พัฒนาให้มีความยืดหยุ่นสูงด้วยการกำหนดมาตรฐานของโครงสร้าง(Framework) ด้วยการยอมให้การพัฒนาเครื่องมือ(Tool) ใหม่ ๆ สามารถที่จะเพิ่มเข้าไปในชุดพัฒนา(Toolkit) ได้

### 2.2 โครงสร้างภาพใน MPEG-4

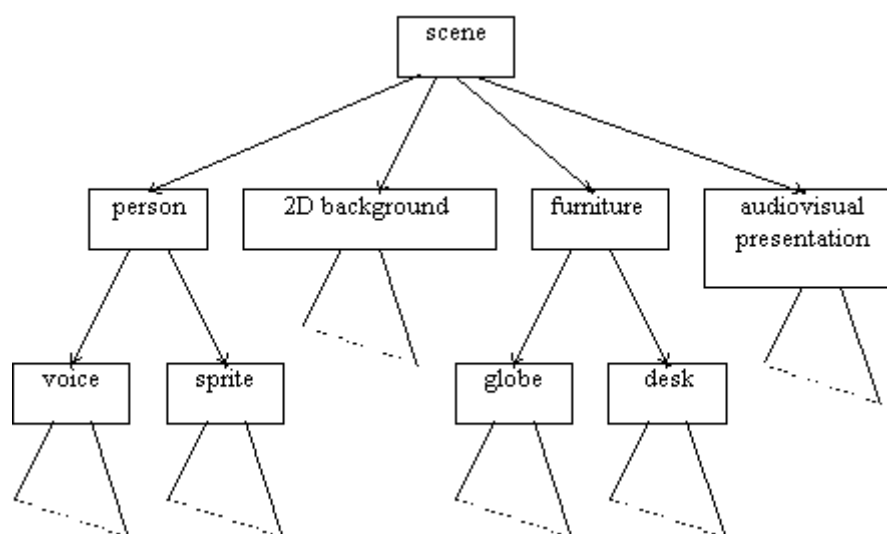
ใน MPEG-4 จะประกอบด้วยสองส่วนหลักคือ ส่วนของการฟัง (Aural) และ ส่วนของการมองเห็น (Visual) หรือรวมเรียกว่าวัตถุภาพและเสียง (Audio/Visual Objects-AVOs) ซึ่งสามารถนำข้อมูลมาจากกล้องและไมโครโฟน โดยจะเห็นได้จากภาพที่ 3 (MPEG-4 Industry Forum, 2002)



ภาพที่ 3 องค์ประกอบของ MPEG-4

ตัวอย่างของวัตถุภาพและเสียงหลัก (Primitive AVOs) ได้แก่ พื้นหลัง, รูปของคนกำลังพูดและ เสียงของคนพูด

องค์ประกอบของภาพ(Scene) ใน MPEG-4 จะประกอบไปด้วย Node ที่สามารถเพิ่มลบ หรือว่าแก้ไขได้ดังภาพที่ 4



ภาพที่ 4 โครงสร้างภาพใน MPEG-4

### 3. การเขียนโปรแกรมวิดิทัศน์ของ MPEG-4

#### 3.1 ภาพรวมของการเขียนโปรแกรมวิดิทัศน์ของ MPEG-4

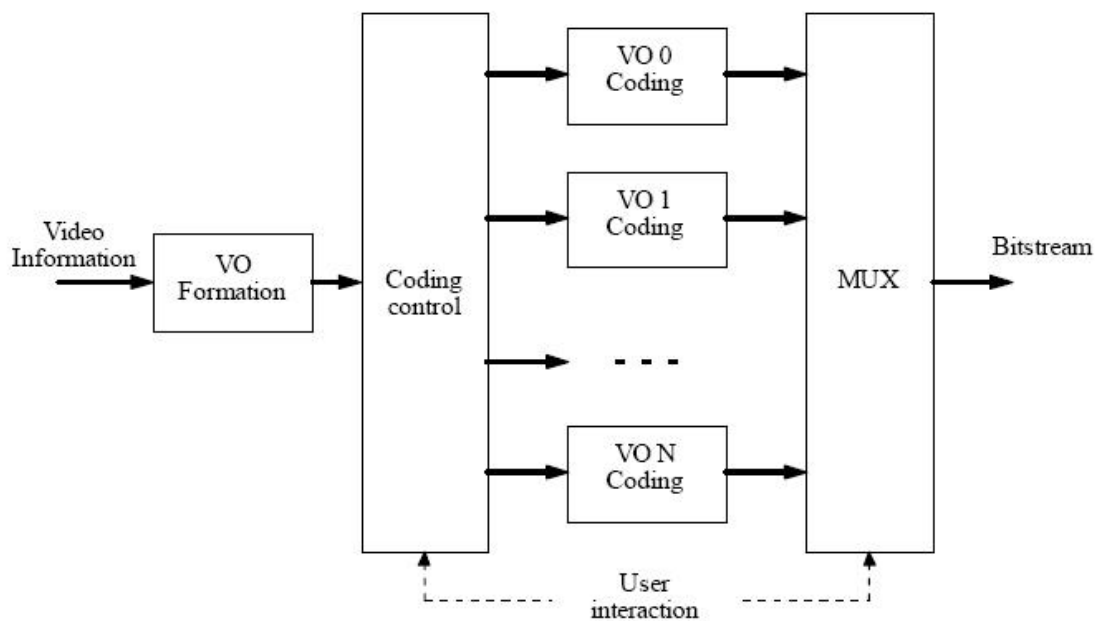
เทคนิคพื้นฐานของการเขียนโปรแกรมวิดิทัศน์ มีสองเรื่องสำคัญคือ การเทียบช่อง (Block Matching) และการแปลงโคไซน์ไม่ต่อเนื่อง (Discrete Cosine Transform - DCT)

ใน MPEG-4 จะทำการแบ่งรูปร่างที่แตกต่างกันในแต่ละ เฟรมออกเป็นวัตถุ ซึ่งเรียกว่า พื้นวัตถุวิดิทัศน์ (Video Object Plan - VOP)(Kee-Yin Ng, 2005; Sikora, 1997; Kadono, 2000)

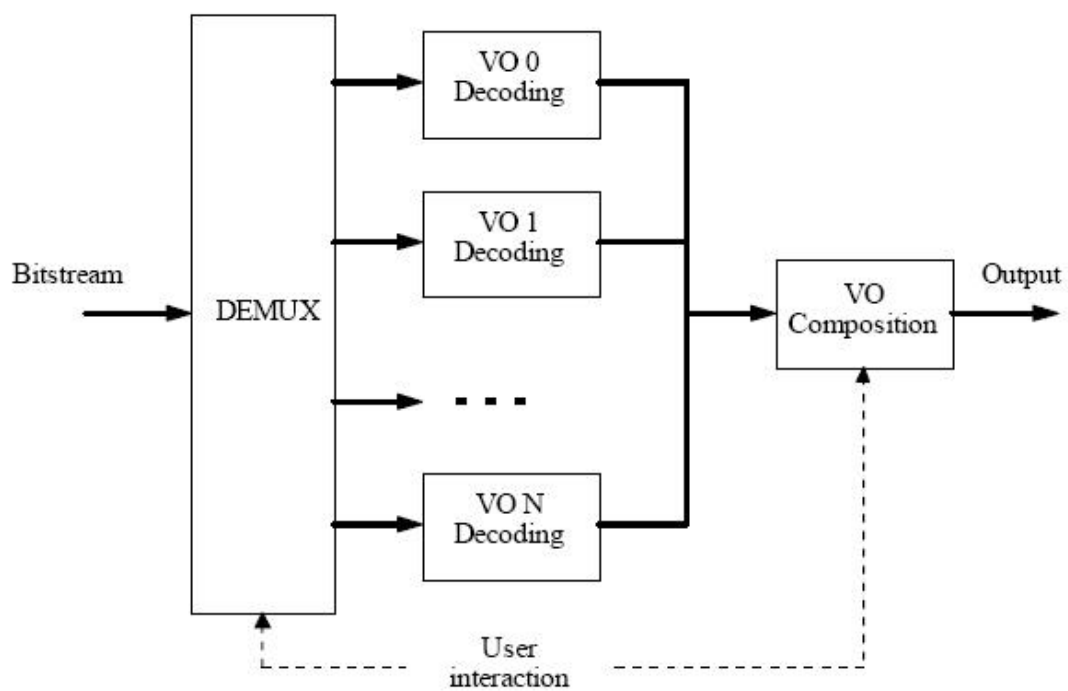
การแบ่งพื้นวัตถุวิดิทัศน์เป็นอัลกอริทึม ที่มีวัตถุประสงค์เพื่อแยกรูปร่างของภาพในแต่ละเฟรมออกเป็นส่วน ๆ

วัตถุวิดิทัศน์ (Video Object - VO)(Paulo Nunes, n.d.) ใน MPEG-4 จะเหมือนกับกลุ่มของภาพ (Group of Picture - GOP) ในมาตรฐาน MPEG-1 และมาตรฐาน MPEG-2

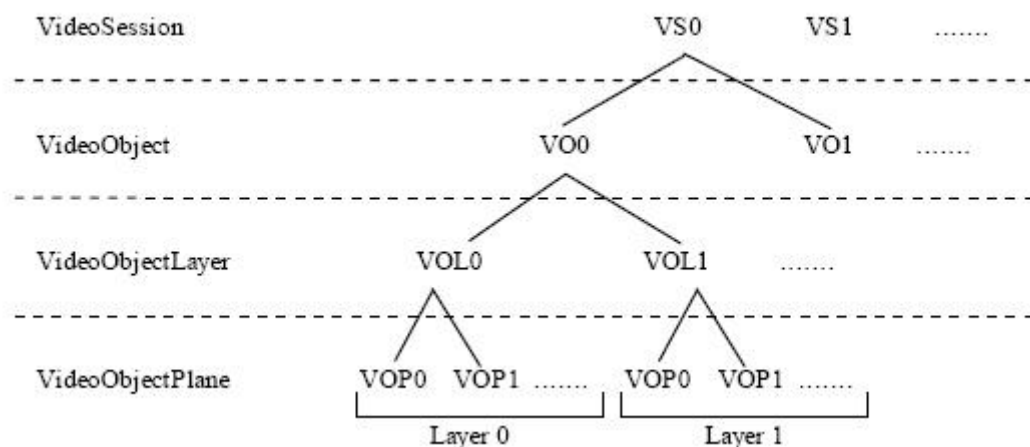
ในการเข้ารหัส ของวัตถุวิดิทัศน์จะทำการแยกการเข้ารหัสออกเป็นแต่ละวัตถุ ที่เรียกว่า ระดับวัตถุวิดิทัศน์ (Video Object Layer - VOL)(Ebrahimi, 1997)



ภาพที่ 5 โครงสร้างของการเข้ารหัส VO ของ MPEG-4



ภาพที่ 6 โครงสร้างของการถอดรหัส VO ของ MPEG-4



ภาพที่ 7 ลำดับชั้นของโครงสร้างข้อมูลใน MPEG-4

การประเมินและชดเชยการเคลื่อนที่ (Motion Estimation and Compensation)

(Chen *et al.*, 2002)

การประเมินการเคลื่อนที่ (Motion Estimation): มีหลายวิธีซึ่งจะขึ้นอยู่กับความเหมาะสมตามเงื่อนไข

3.2 ฟังก์ชันประเมินค่า (Cost Function) คือวิธีการของการนำสองกลุ่มช่อง (Macroblocks MB) มาเทียบกัน และขึ้นอยู่กับเปรียบเทียบคุณสมบัติในช่อง ซึ่งคุณสมบัติหนึ่งคือหนึ่งกลุ่มของเงื่อนไขวิธีการประเมินโดยวิธีคือตัวอย่างของสมการที่กลุ่มช่องปัจจุบัน (Current MB) หมายถึง  $C(x+k, y+l)$  และตำแหน่งพิกเซล ในรูปคือ  $R(x+i+k, y+j+l)$

3.2.1 MAE(Mean Absolute Error) คือค่าความคลาดเคลื่อนกลางเฉลี่ยสมบูรณ์ เมื่อ

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

โดยที่ Coordinate (i,j) ซึ่ง MAE ถูกทำให้เป็นเวกเตอร์การเคลื่อนที่ (Motion Vector)

3.2.2 MSE(Mean Square Error) คือค่าความคลาดเคลื่อนกลางกำลังสอง เมื่อ

$$MSE(i, j) = \frac{1}{MN} \sum_{k=0}^{M-1} \sum_{l=0}^{N-1} |C(x+k, y+l) - R(x+i+k, y+j+l)|^2$$

ความสัมพันธ์ระหว่างช่องอื่น ๆ ในการเปรียบเทียบอัตราตรงกัน

3.2.3 PDC(Pixel Difference Classification) คือการแบ่งกลุ่มความต่างของพิกเซล โดยมีหลักการนับจำนวนของพิกเซล ที่ตรงกันระหว่างสองช่อง โดยเป้าหมายคือจะได้ค่าที่ตรงกันมากที่สุด

$$PDC(i, j) = \sum_k \sum_l T_{i,j}(k, l)$$

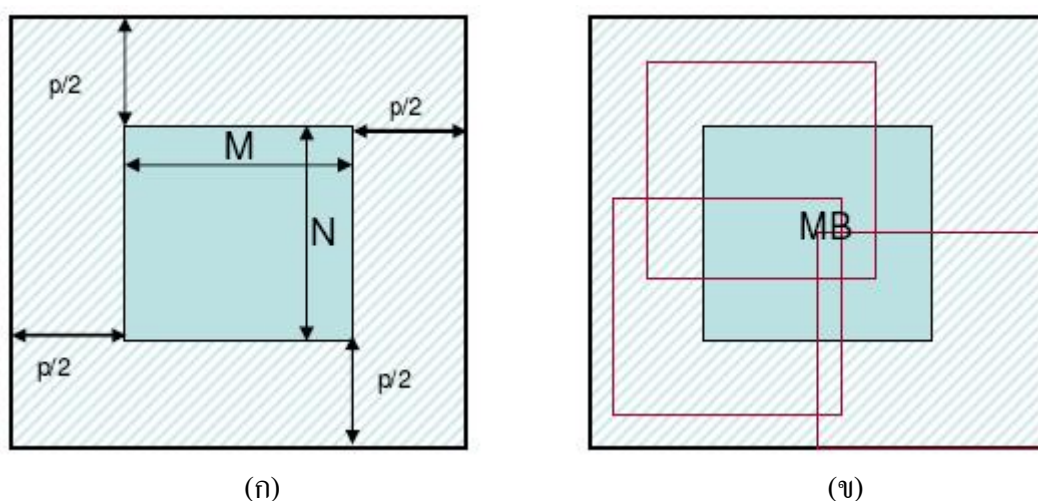
$$\begin{aligned} \text{เมื่อให้ } T_{i,j}(k, l) &= 1 \quad \text{ถ้า } |C(x+k, y+l) - R(x+i+k, y+j+l)| \leq t \\ &= 0 \quad \text{กรณีอื่น ๆ} \end{aligned}$$

3.2.4 BPDC(Binary Pixel Difference Classification) และ BPROP ซึ่ง BPDC คือ PDC ในรูปแบบฐานสอง (Binary) โดยในทุก ๆ พิกเซลที่ตรงกันจะให้น้ำหนักที่เท่ากัน แต่ใน บิต ที่มีความสำคัญสูงจะให้น้ำหนักมากกว่า โดยบรรทัดฐานในการเทียบระดับของ เลขฐานสอง นี้จะถูกเรียกว่า BPROP และมีวัตถุประสงค์เพื่อให้ได้ค่าที่น้อยที่สุด

3.2.5 DPC(Difference Pixel Count Criterion) คือเกณฑ์การนับจำนวนพิกเซล ที่แตกต่างกัน ซึ่งจะต่างจาก BPDC ตรงที่ ใช้การวิเคราะห์เชิงปริมาณ ของการเทียบพิกเซล ในการคำนวณ

3.2.6 BPM(Bit-Plane Mapping Criterion) เมื่อพิกเซล ของทั้งรูปปัจจุบัน (Current frame) และรูปอ้างอิง (Reference frame) ถูกกำหนดให้มีค่าหนึ่งบิต จึงได้นำเอาคุณสมบัตินี้มารวมกับ MAE ในการตรวจหาขอบ (Edges) ของการประเินการเคลื่อนที่

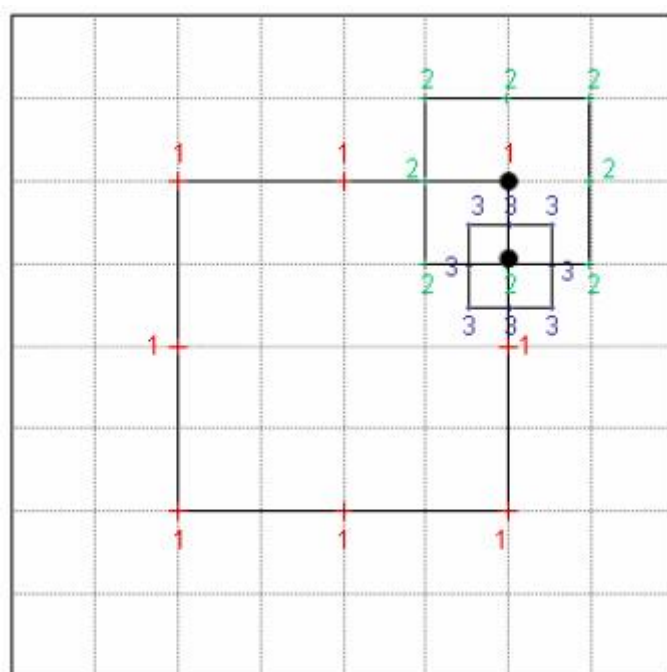
3.3 วิธีการค้นหา (Search Method) ในการค้นหาที่ค่านั้นจะต้องค้นหาภาพทั้งหมดแต่เป็นไปได้ยาก แต่จะสามารถค้นหาบริเวณโดยรอบของจุดเริ่มต้นเท่านั้น



ภาพที่ 8 (ก) แสดงบริเวณการค้นหา (ข) และการวิเคราะห์และเทียบแต่ละ MB

3.3.1 Exhaustive search(ES) or Full search เป็นการค้นหาอย่างละเอียดหรือค้นหาทั้งหมด วิธีการค้นหาแบบนี้เป็นอัลกอริทึมในการเทียบ ได้ดีที่สุด แต่จะใช้เวลาคำนวณสูง นั้นอาจหมายถึงไม่เหมาะกับการนำไปคำนวณฟังก์ชันประเมินค่า แบบ MB

3.3.2 Two-Dimensional Logarithmic Search(TDLS) and Tree Step Search(TSS) เป็นการค้นหาโดยใช้เลขฐานสองและการค้นหาสามขั้น ซึ่ง TDLS จะทำการแบ่งพื้นที่สี่เหลี่ยมที่ให้เป็นสองพื้นที่คือ พื้นที่ภายในและพื้นที่ภายนอก จากนั้นใช้ ฟังก์ชันประเมินค่า ในการคำนวณพื้นที่ทั้งหมด  $[-p/2, p/2]$  นั่นคือตำแหน่ง  $(0,0)$  และทิศโดยรอบทั้งแปด (จะแทนค่า  $S$  เป็นระยะทางจากตำแหน่งหลักที่ถูกวิเคราะห์ในพื้นที่สี่เหลี่ยม) ซึ่งความซับซ้อนของอัลกอริทึมนี้จะเท่ากับ 3.3% เมื่อเทียบกับแบบแรก ส่วน TSS นั้นเป็นการค้นหาเฉพาะ TDLS ที่  $p$  ถึง 7 (ในการประมวลผลจะมี 3 ขั้นตอนก่อนจะถึงพื้นที่ระยะทาง 1 จุด) ซึ่งเพียงพอสำหรับการทำวิดีโอคอนเฟอเรนซ์ (Videoconference) และการเคลื่อนที่หายไปใน้อย



ภาพที่ 9 TSS โดยจะแทน ตัวเลขเป็นการค้นหา โดยให้จุดดำเป็น object ที่ต้องการค้นหา

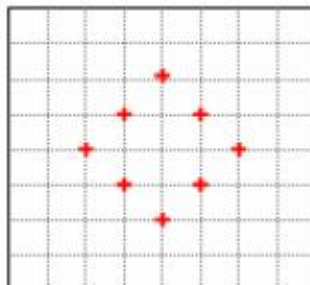
3.3.3 Parallel Hierarchical One-Dimensional Search (PHODS) เป็นการค้นหาหนึ่งทิศทางแบบลำดับชั้นคู่ขนาน จะมีลักษณะคล้าย ๆ กับ TDLS แต่การค้นหาจะไม่ขึ้นอยู่กับสองทิศทางตามกัน นั่นคือการคำนวณฟังก์ชันประเมินค่า จะถูกทำให้คู่ขนานตามการไหลของข้อมูลในพื้นที่

3.3.4 New Three Step Search (NTSS) การค้นหาสามขั้นตอนแบบใหม่ จะมีลักษณะคล้าย ๆ กับ TSS โดยจะทำการเปลี่ยนจุดเริ่มต้นการค้นหาใหม่ทุก ๆ ครั้งที่มีการค้นหาเจอ

3.3.5 Simple and Efficient Search (ESE) การค้นหาแบบง่าย ๆ และมีประสิทธิภาพ จะทำบนสมมติฐานที่มีสองทิศทางที่ต่างกันน้อยสุด โดยมีหลักการคล้าย ๆ กับ TSS แต่ในแต่ละขั้นตอนจะเป็นสองระยะ การค้นหาจะแบ่งออกเป็นสี่ส่วน แต่ละส่วนจะทำงานเหมือน ๆ กัน คือ จะถูกเลือกสามจุด และจะขึ้นอยู่กับ การแบ่งส่วน

3.3.6 Four Step Search (FSS) ค้นหาสี่ขั้นตอน คล้าย ๆ กับ TSS S จะกำหนดเป็น 2 ฟังก์ชันประเมินค่า ทำการคำนวณ 9 พื้นที่ ในมิติ 5 ต่อ 5 หน้าต่าง จะมีการย้ายจุดกึ่งกลางเหมือน NTSS และในขั้นตอนสุดท้ายจะมีขนาดมิติ 3 ต่อ 3 หน้าต่าง

3.3.7 Diamond Search(DS) ขั้นตอนจะเหมือนกับ FSS แต่จะไม่ใช้รูปแบบสี่เหลี่ยมมุมฉาก โดยจะใช้รูป Diamond แทน



ภาพที่ 10 TSS รูปแบบของ Diamond Search

3.3.8 Adaptive Rood Pattern Search(ARPS) หลักการคือจะทำบนพื้นฐานทิศทางการเคลื่อนที่ของ frame

3.4 การประเมินการเคลื่อนที่หลาย ๆ ภาพ (Multipicture Motion Estimation) โดยจะพิจารณาจากภาพที่อ้างอิงที่ถูกรับทำการประเมินการเคลื่อนที่แล้ว

3.5 Motion Compensation: การชดเชยการเคลื่อนที่ในสิ่งที่หายไป โดยอาจใช้วิธีการดังนี้

3.5.1 Global Motion Compensation(GMC) กล้องหลาย ๆ ตัวจะใช้วิธีการนี้ แต่ต้นแบบของวัตถุจะมีคุณภาพไม่ดี

3.5.2 Block-base Motion Compensation(BMC) เป็นวิธีการที่เป็นไปได้สูงในการชดเชย โดยมีวิธีการที่จะชดเชยการเคลื่อนที่ในพื้นที่สี่เหลี่ยม โดยจะใช้อัลกอริทึมในการค้นหาตามวิธีการข้างต้น ซึ่งวิธีการนี้จะได้รับความนิยมมากเนื่องจากจะทำให้ได้วัตถุเคลื่อนที่ที่ถูกต้อง และพอเหมาะกับเฟรมที่เป็นสี่เหลี่ยม

3.5.3 Overlapped Block Motion Compensation(OBMC) จัดการกับปัญหาที่เกิดจากการซ้อนทับกันของ ช่องซึ่งจะให้ 4 พิกเซล และ มี 4 การประเมินค่า แล้วทำการประมวลผลให้มีอิสระต่อกัน

3.5.4 Sub-pixel Motion Compensation (SPMC) การชดเชยชนิดนี้จะทำให้มีคุณภาพสูง แต่มีความซับซ้อนมาก

3.5.5 Region Based Motion Compensation จะทำการชดเชยตามขอบเขตรูปร่าง โดยจะจัดการกับข้อมูลของรูปร่าง (Shape Information)

#### 4. ช่องสี (Color Spaces)

รูปแบบของสีที่เกี่ยวข้องกับ Digital Image และ video นั้นจะใช้ 2 รูปแบบหลัก ๆ คือ RGB และ YCrCb

##### 4.1 RGB (red/green/blue)

ในแต่ละ Pixel จะประกอบด้วยค่าของแม่สีสามสีที่สัมพันธ์กันอยู่คือ แดง เขียว และ น้ำเงิน โดยในแต่ละสีจะเป็นตัวเลข 8 บิต ดังนั้นเมื่อมีสามสีจึงต้องเป็น  $3 \times 8 = 24$  บิต

##### 4.2 YCrCb

เป็นรูปแบบของแสงและสี เมื่อ Y คือค่าความสว่าง (luminance) ที่เป็นน้ำหนักของค่าเฉลี่ยของ R, G และ B ดังนี้

$$Y = k_r R + k_g G + k_b B$$

เมื่อค่า  $k$  น้ำหนัก

และค่า Cr Cb และ Cg เป็นค่าความต่างของสี ซึ่งหาได้โดย

$$Cr = R - Y$$

$$Cb = B - Y$$

$$Cg = G - Y$$

อย่างไรก็ดีจะใช้ค่าความต่างของสีสองตัวเท่านั้นคือ Cr และ Cb

ในขั้นตอนของการแปลงสีจาก RGB ไปเป็น Y:Cr:Cb จะใช้วิธีการดังนี้

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

$$Cb = 0.564(B - Y)$$

$$Cr = 0.713(R - Y)$$

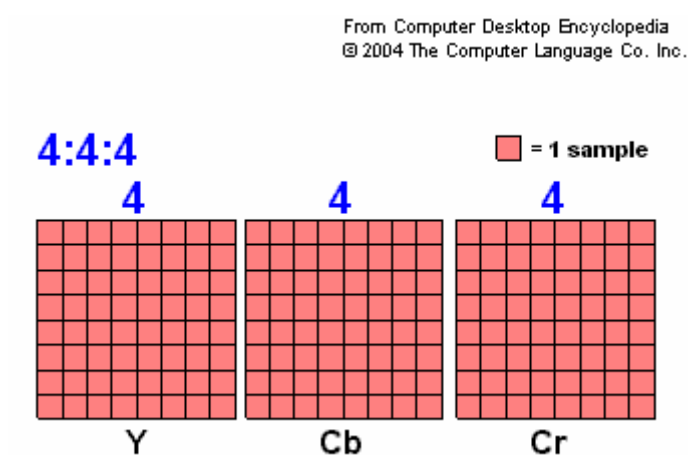
และสามารถแปลงจาก Y:Cr:Cb ไปเป็น RGB ได้โดย

$$R = Y + 1.402Cr$$

$$G = Y + 0.344Cb - 0.714Cr$$

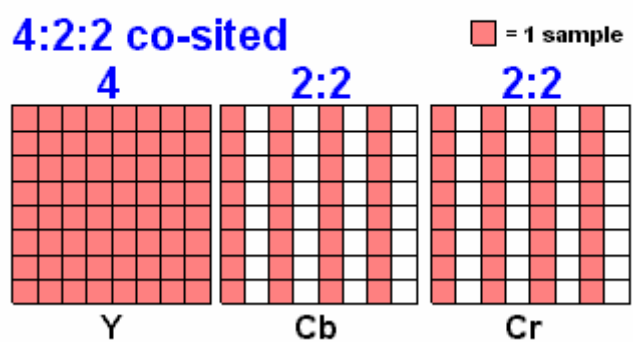
$$B = Y + 1.772Cb$$

อัตราส่วนของ Y:Cb:Cr มีดังต่อไปนี้



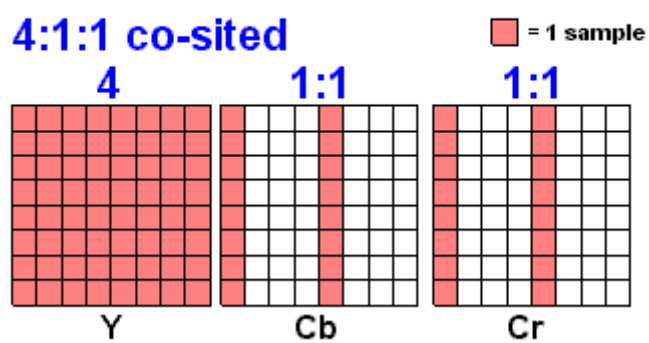
ภาพที่ 11 4:4:4 (Cb/Cr same as luma)

From Computer Desktop Encyclopedia  
© 2004 The Computer Language Co. Inc.

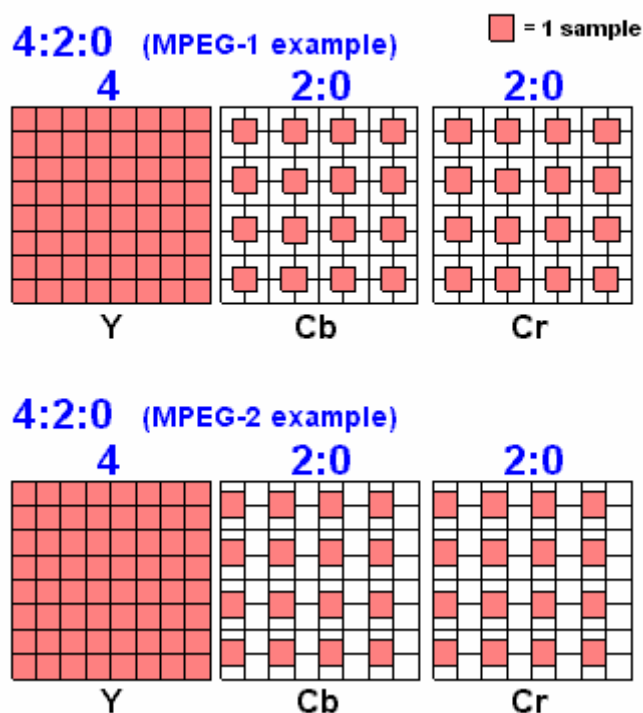


ภาพที่ 12 4:2:2 (1/2 the Luma Samples)

From Computer Desktop Encyclopedia  
© 2004 The Computer Language Co. Inc.



ภาพที่ 13 4:1:1 (1/4 the Luma samples)



ภาพที่ 14 4:2:0 (1/4 the Luma samples)

## 5. ตัวเข้ารหัสการแปลง (Transform Encoder)

### 5.1 การแปลงโคไซน์ไม่ต่อเนื่อง (Discrete Cosine Transform - DCT)

ในขั้นตอนการแปลง DCT นั้น ข้อมูลที่อยู่ในรูปเมตริกซ์ 2 มิติของพิกเซล (spatial domain) จะถูกนำมาแปลงให้อยู่ในโดเมนความถี่ (Frequency Domain) โดยก่อนที่จะทำการแปลง DCT ข้อมูลภาพจะต้องถูกแบ่งออกเป็นบล็อกเล็กๆ ขนาด  $N \times N$  โดยทั่วไปจะใช้  $8 \times 8$  จากนั้นทำการแปลง DCT ที่ละบล็อก สมมติให้ข้อมูลภาพภายในบล็อกแต่ละบล็อกเขียนแทนด้วยฟังก์ชัน  $x(i, j)$  โดย  $0 \leq i, j \leq N-1$  ซึ่งในที่นี้ก็คือ  $0 \leq i, j \leq 7$  สมการที่ใช้ในการแปลง DCT (ถัญณกร วุฒิสัทกุลกิจ และคณะ, 2549)เป็นดังนี้

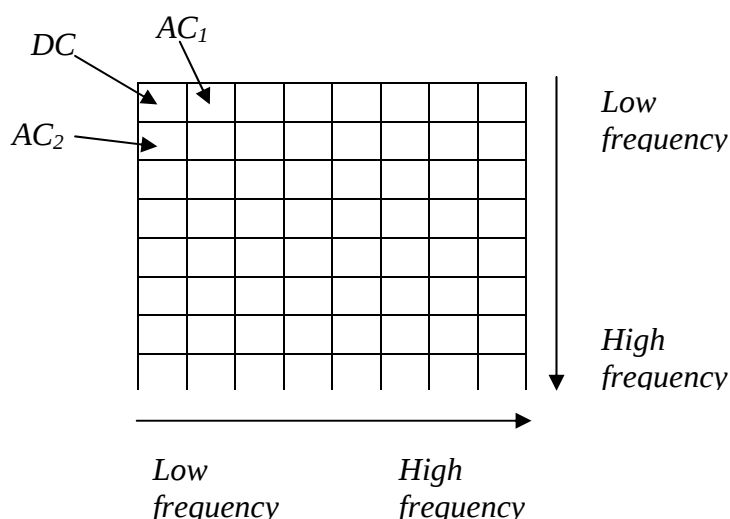
$$X(0,0) = \frac{1}{\sqrt{4N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} x(i, j)$$

เมื่อให้  $u, v = 0$

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

เมื่อ  $1 \leq u, v \leq N-1$

ผลลัพธ์ที่ได้จากการแปลง DCT จะได้บล็อกของสัมประสิทธิ์ความถี่ขนาดเท่าเดิม คือ  $N \times N$  โดย  $x(0,0)$  ซึ่งอยู่ ณ ตำแหน่งมุมด้านซ้ายสุดของเมตริกซ์จะหมายถึงค่าองค์ประกอบกระแสดตรงของภาพต้นฉบับ นั่นคือ  $x(0,0)$  จะแสดงค่าเฉลี่ยความสว่างของพิกเซลภายในบล็อกนั้นๆ และ  $X(u, v)$  ค่าอื่น ๆ แทนองค์ประกอบกระแสดลัดของภาพที่มีความถี่ต่าง ๆ ซึ่งองค์ประกอบกระแสดลัดความถี่ต่ำจะอยู่บริเวณด้านซ้ายของเมตริกซ์แล้วเพิ่มความถี่ขึ้นเมื่อเคลื่อนที่ลงมาทางด้านล่างขวาของเมตริกซ์ โดยองค์ประกอบกระแสดลัดของภาพนั้นจะหมายถึงการเปลี่ยนแปลงค่าของสีในแต่ละพิกเซลนั่นเอง นั่นคือหากค่าสีภายในบล็อกนั้นมีการเปลี่ยนแปลงอย่างรวดเร็ว ค่าองค์ประกอบกระแสดลัดที่มีความถี่สูงจะมีค่ามากนั่นเอง



ภาพที่ 15 สัมประสิทธิ์ความถี่ขนาด  $8 \times 8$  หลังผ่านการแปลง DCT

โดยปกติระดับความสว่างหรือระดับความเข้มลูมิแนนซ์ของค่าพิกเซล ค่าอยู่ระหว่าง 0 ถึง 255 โคน 0 หมายถึงพิกเซลมีความสว่างน้อยที่สุด และ 255 หมายถึงพิกเซลมีความสว่างมาก

## 5.2 การวิเคราะห์เชิงปริมาณ (Quantization)

เนื่องจากพลังงานของสัญญาณภาพโดยทั่วไปมักจะประกอบด้วยองค์ประกอบกระแสดตรง และองค์ประกอบกระแสดลับที่มีความถี่ต่ำเป็นส่วนใหญ่ ดังนั้น สัมประสิทธิ์ของ DCT ที่คำนวณได้แต่ละตัวจึงมีลำดับความสำคัญไม่เท่ากัน กล่าวคือสัมประสิทธิ์ขององค์ประกอบกระแสดตรงและองค์ประกอบกระแสดลับที่มีความถี่ต่ำมีความสำคัญมากกว่าสัมประสิทธิ์ขององค์ประกอบกระแสดลับที่มีความถี่สูง ด้วยเหตุนี้สัมประสิทธิ์ที่มีความสำคัญมากคือสัมประสิทธิ์ขององค์ประกอบกระแสดตรง จะได้รับการวิเคราะห์เชิงปริมาณด้วยระดับความละเอียดที่สูง ในขณะที่สัมประสิทธิ์ที่มีความถี่สูงจะได้รับการวิเคราะห์เชิงปริมาณค่อนข้างหยาบ การกำหนดระดับความละเอียดของสัมประสิทธิ์แต่ละตัวจึงต้องมีการพิจารณาอย่างถี่ถ้วนโดยคำนึงถึงคุณภาพที่ได้ โดยสมการของการวิเคราะห์เชิงปริมาณเป็นดังนี้

$$Fq(u,v) = \text{Integer}(\text{Round}(Fu,v)/Q(u,v))$$

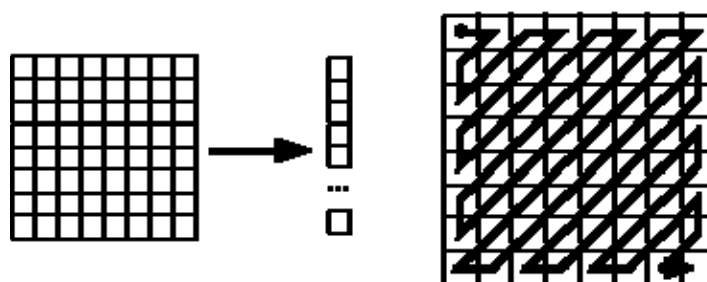
$$Fq^{-1}(u,v) = F(u,v)Q(u,v)$$

เมื่อ  $F(u,v)$  เป็นสัมประสิทธิ์ของ DCT และ  $Q(u,v)$  เป็นค่าควอนไทซ์ในตาราง ซึ่งใน JPEG จะมีการใช้ค่าดังนี้

|    |    |    |    |     |     |     |     |
|----|----|----|----|-----|-----|-----|-----|
| 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  |

### 5.3 การเข้ารหัสเอนโทรปี (Entropy Encoding)

การสแกนแบบซิกแซก ช่องสัมประสิทธิ์ DCT ที่คำนวณได้  $8 \times 8$  ทั้ง 64 ค่าในขั้นตอนที่ผ่านมา จะนำมาเข้ารหัสเอนโทรปีโดยเริ่มจากการจัดเรียงข้อมูลภายในเมทริกซ์ให้อยู่ในลำดับที่เหมาะสมก่อน กรรมวิธีการจัดเรียงที่เลือกใช้คือ การสแกนแบบซิกแซก (Zig-Zag Scan) ซึ่งจะทำให้การเปลี่ยนแปลงค่าเมทริกซ์สองมิติให้เป็นเวกเตอร์ 1 มิติ นั่นคือเปลี่ยนจากเมทริกซ์  $8 \times 8$  เป็น  $1 \times 64$  โดยสัมประสิทธิ์ค่าแรกที่อ่านออกคือ องค์ประกอบกระแสดตรง แล้วตามด้วยองค์ประกอบแอสลับตัวแรก ( $AC_1$ ) ซึ่งอยู่มุมด้านซ้ายของบล็อก จากนั้นจะอ่านค่าองค์ประกอบกระแสดลับตัวที่สอง ( $AC_2$ ) แล้วตามด้วยองค์ประกอบกระแสดลับที่ความถี่สูงขึ้นเรื่อย ๆ ตามเส้นที่แสดงในรูป เหตุผลที่ทำให้การเรียงแบบนี้เพื่อต้องการเรียงค่าข้อมูลที่เป็น 0 ให้อยู่ติด ๆ กันเพื่อที่จะทำให้เกิดประสิทธิภาพในการบีบอัดข้อมูล



ภาพที่ 16 ลำดับการทำ Zig-Zag Scan

การเข้ารหัสส่วนต่างและการเข้ารหัสแบบ run-length สำหรับองค์ประกอบกระแสดตรง (DC) ซึ่งคือ  $X(0,0)$  จะทำการเข้ารหัสส่วนต่าง (Differential encoding) โดยจะนำค่าที่เปลี่ยนแปลงไปจากค่าองค์ประกอบกระแสดตรงจากบล็อกก่อนหน้านี้มาเข้ารหัสเท่านั้น นั่นคือจะนำค่าสัมประสิทธิ์กระแสดตรงจากช่องก่อนหน้านี้มาหักลบออกก่อนแล้วจึงนำค่าที่ได้ไปเข้ารหัส เหตุผลที่ทำเช่นนี้เพราะบล็อกที่อยู่ติดกันมักจะมีค่าเฉลี่ยของสัญญาณ (DC) ใกล้เคียงกัน เพราะฉะนั้น เมื่อนำเฉพาะค่าผลต่างมาเข้ารหัสจะสามารถลดปริมาณบิตข้อมูลลงได้มาก

การเข้ารหัสฮัฟฟ์แมน ในการเข้ารหัสฮัฟฟ์แมนนั้นจะทำการเข้ารหัสโดยการแยกพิจารณาค่าองค์ประกอบกระแสดตรงและค่าองค์ประกอบกระแสดลับแยกกันคือ องค์ประกอบกระแสดตรงนั้น การเข้ารหัสจะประกอบด้วยค่าสองส่วน ส่วนแรกจะเป็นการเข้ารหัสส่วนต่างโดยจะแบ่งกลุ่มส่วนต่าง (CAT. Value) ดังตารางที่ 1

**ตารางที่ 1** ตารางการจัดกลุ่มองค์ประกอบกระแสดรและกระแสดล

| DC       | AC       |                                     |
|----------|----------|-------------------------------------|
| Category | Category | Coefficient                         |
| (SSSS)   | (SSSS)   |                                     |
| 0        | N/A      | 0                                   |
| 1        | 1        | -1 , 1                              |
| 2        | 2        | -3,-2 , 2,3                         |
| 3        | 3        | -7,...,-4 , 4,...,7                 |
| 4        | 4        | -15,...,-8 , 8,...,15               |
| 5        | 5        | -31,...,-16 , 16,...,31             |
| 6        | 6        | -63,...,-32 , 32,...,63             |
| 7        | 7        | -127,...,-64 , 64,...,127           |
| 8        | 8        | -255,...,-128 , 128,...,255         |
| 9        | 9        | -511,...,-256 , 256,...,511         |
| 10       | 10       | -1023,...,-512 , 512,...,1023       |
| 11       | 11       | -2047,...,-1024 , 1024,...,2047     |
| 12       | 12       | -4095,...,-2048 , 2048,...,4095     |
| 13       | 13       | -8191,...,-4096 , 4096,...,8191     |
| 14       | 14       | -16383,...,-8192 , 8192,...,16383   |
| 15       | N/A      | -32767,...,-16384 , 16384,...,32767 |

ลั้วจ้งขัรห้สัฟแมนส่วนต้งตามกลุ่มที่จ้ดไว้ด้งตารางที่ 2

ตารางที่ 2 ตารางการเข้ารหัสฮัฟฟ์แมนสำหรับองค์ประกอบกระแสดรง

| Category | Huffman Code |
|----------|--------------|
| 0        | 00           |
| 1        | 010          |
| 2        | 011          |
| 3        | 100          |
| 4        | 101          |
| 5        | 110          |
| 6        | 1110         |
| 7        | 11110        |
| 8        | 111110       |
| ...      | ....         |

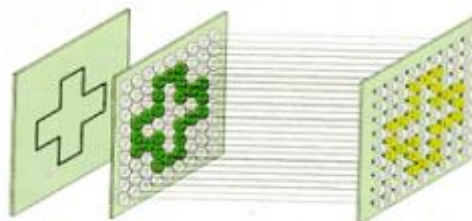
## 6. เครื่องจำอักขระด้วยแสง (Optical character recognition)

โอซีอาร์ (OCR - Optical character recognition) หรือ เครื่องจำอักขระด้วยแสง คือซอฟต์แวร์อย่างหนึ่งที่ออกแบบเพื่อให้แปลความหมายที่ปรากฏในไฟล์รูปภาพ ไปเป็นข้อมูลที่สามารถประมวลผลได้ โดยปกติการนำเข้ารูปภาพเพื่อเข้าสู่กระบวนการแปลสามารถใช้เครื่องสแกนเนอร์ หรือกล้องดิจิทัล หรืออุปกรณ์ที่ออกแบบมาเฉพาะทาง โอซีอาร์เริ่มมีขึ้นโดยเป็นโครงการของการตรวจสอบในระบบปัญญาประดิษฐ์และการมองเห็นของเครื่องจักร และมีการวิจัยต่อไปโดยมุ่งประเด็นไปที่การใช้โอซีอาร์เป็นเครื่องมือในการพิสูจน์ เช่น พิสูจน์อักษร ลายมือ ลายนิ้วมือ ใบหน้า ดวงตา(Sie Bot และคณะ, 2550) เป็นต้น

### 6.1 ประวัติ

ระบบโอซีอาร์ไม่ใช่เรื่องใหม่สำหรับกระบวนการประมวลผลข้อมูล เนื่องจากความพยายามที่จะพัฒนาเครื่องช่วยอ่านตัวอักษรอัตโนมัติเพื่อช่วยงานมนุษย์นั้น มีการคิดค้นกันมากรว่าร้อยปีแล้ว ถ้าหากจะกล่าวถึงประวัติของโอซีอาร์ ก็น่าจะเริ่มต้นกันที่คำว่า optical ซึ่งบอกถึงหลักการทำงานของระบบที่เกี่ยวข้องกับการประมวลผลของภาพอิเล็กทรอนิกส์ที่ได้มาจากเครื่องสแกน

ซึ่งมีจุดเริ่มต้นในปี พ.ศ. 2416 โดยพนักงานโทรเลขชาวไอแลนด์ชื่อ Joseph May ได้ค้นพบคุณสมบัติทางโฟโตอิเล็กทริก (photoelectric) จากซีลีเนียม (selenium) โดยบังเอิญ จากนั้นในปี พ.ศ. 2418 ในประเทศสหรัฐอเมริกา George Carey ได้นำเสนอเครื่องเรตินาสแกนเนอร์ (retina scanner) ที่สามารถถ่ายทอดสัญญาณภาพจากแผงโฟโตอิเล็กทริกเซลล์ที่รับแสงสะท้อนจากภาพแล้วแสดงผลที่แผงอีกด้านหนึ่ง

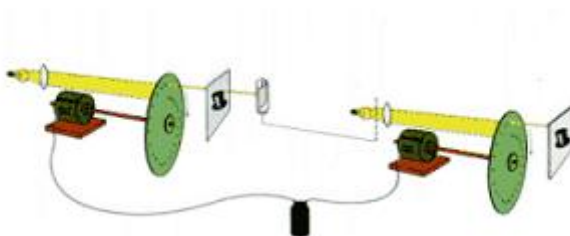


ภาพที่ 17 หลักการทำงานของเครื่องเรตินาสแกนของ George

และต่อมาในปี พ.ศ. 2427 Paul Nipkow ชาวเยอรมันได้คิดค้นเครื่องสแกนภาพแบบใหม่ที่ใช้จานหมุนที่มีรูเล็กๆ เรียงกันในลักษณะเป็นก้นหอยเรียกว่านิพโกว์ดิสก์ (Nipkow disk) ในการอ่านภาพ ซึ่งถือเป็นประดิษฐ์กรรมต้นแบบของเครื่องสแกนรวมทั้งเทคโนโลยีการถ่ายทอดภาพของวงการโทรทัศน์ในปัจจุบันอีกด้วย



Paul Nipkow (1860-1940)



ภาพที่ 18 Paul Nipkow และ Nipkow disk

เครื่องอ่านตัวอักษรเครื่องแรกที่มีการรายงานไว้เป็นเครื่องที่สร้างขึ้นเพื่อช่วยคนพิการทางตาในการอ่านหนังสือซึ่งเป็นผลงานของนักวิทยาศาสตร์ชาวรัสเซียชื่อ Tyurin สร้างขึ้นในปี พ.ศ. 2443 จากนั้นในปี พ.ศ. 2472 Tausheck ได้จดสิทธิบัตรระบบโอซีอาร์เป็นครั้งแรกในประเทศเยอรมนี และต่อมาปี พ.ศ. 2476 ในอเมริกา P.W. Handel ก็ได้จดสิทธิบัตรโอซีอาร์ของตนเองเช่นกัน ถึงแม้เริ่มแรกระบบโอซีอาร์ดังกล่าวจะมีลักษณะและความสามารถที่เทียบไม่ได้กับระบบโอซีอาร์ที่เราใช้กันอยู่ในปัจจุบัน แต่แนวความคิดพื้นฐานของระบบดังกล่าวก็ยังคงใช้กันอยู่จนกระทั่งถึงทุกวันนี้

ความแตกต่างของระบบโอซีอาร์ในยุคแรกๆ ที่เห็นได้ชัดคือ ลักษณะที่เป็นชุดอุปกรณ์ฮาร์ดแวร์ที่พ่วงติดกับเครื่องสแกน แทนที่จะเป็นชุดซอฟต์แวร์ที่ใช้กับเครื่องคอมพิวเตอร์แบบตั้งโต๊ะ (การกล่าวถึงชุดอุปกรณ์ฮาร์ดแวร์โอซีอาร์ผู้เขียนขอใช้คำว่า “ระบบโอซีอาร์” แทนที่จะเป็น “โปรแกรมโอซีอาร์” ซึ่งจะหมายถึงชุดซอฟต์แวร์โอซีอาร์ที่ใช้กับเครื่องคอมพิวเตอร์ ทั้งนี้เพื่อความชัดเจนในการใช้คำ) โดยโอซีอาร์ยุคแรกๆ จะมีลักษณะออกไปทางเครื่องจักรกล ซึ่งต่อมาได้รับการพัฒนาเป็นเครื่องอิเล็กทรอนิกส์แบบอนาล็อก แล้วจึงมาเป็นระบบดิจิทัลอย่างเช่นในปัจจุบัน ทุกวันนี้ชุดอุปกรณ์ฮาร์ดแวร์โอซีอาร์เช่นนี้ก็ยังคงมีการผลิตอยู่และมีราคาสูงมาก เพื่อใช้ในการแปลงเอกสารหรือแบบฟอร์มเฉพาะด้านที่มีปริมาณมากๆ เครื่องพวกนี้จะมีประสิทธิภาพการทำงานสูง ตัวอย่างการใช้งานโอซีอาร์แบบนี้ ได้แก่ ระบบอ่านรหัสไปรษณีย์เพื่อแยกจดหมายอัตโนมัติ และระบบสำหรับอ่านหมายเลขเช็ค เป็นต้น

การวิจัยและพัฒนาโอซีอาร์ได้ก้าวหน้าเป็นอย่างมากหลังจากที่ได้มีการประดิษฐ์คอมพิวเตอร์ขึ้นมาใช้ ศักยภาพเชิงการคำนวณและการทำงานแบบดิจิทัลมีผลต่อการออกแบบและพัฒนาโอซีอาร์ทั้งด้านฮาร์ดแวร์ และอัลกอริธึม ทำให้สามารถนำวิธีการใหม่ๆ ที่ดีกว่ามาใช้ ตัวอย่างโอซีอาร์รุ่นแรกๆ ที่นำเอาเทคนิคทางด้านดิจิทัลมาใช้ได้แก่ ERA (Electric Reading Automation) ของบริษัท Solatron Electronics Group Ltd. ซึ่งใช้ประโยชน์จากพื้นที่หน่วยความจำของคอมพิวเตอร์ที่มีลักษณะสองมิติมาช่วยในการทำงาน ซึ่งก่อนหน้านี้ทำได้ยาก โดยส่วนใหญ่จะต้องแปลงภาพตัวอักษรซึ่งเป็นรูปสองมิติให้เป็นข้อมูลแบบหนึ่งมิติก่อนการประมวลผล ส่วนตัวอักษรที่ ERA อ่านได้นั้น จะเป็นเฉพาะตัวเลขเท่านั้น อีกตัวอย่างหนึ่งของเครื่องแบบดิจิทัลคือระบบโอซีอาร์ของ Electrotechnical Laboratory (ETL) ของญี่ปุ่น โดยคอมพิวเตอร์ ETLMark IV กับเทคนิคที่คล้ายกับ ERA แต่สามารถอ่านได้ทั้งตัวเลขและตัวอักษรทั้งหมด 72 ตัวอักษร นอกจากนั้นยังมีโอซีอาร์ของบริษัท RCA ที่สามารถทำงานได้กับพจนานุกรมภาษาอังกฤษและรัสเซีย โดย RCA ซึ่งเป็นผู้นำทางด้านหลอดอิเล็กทรอนิกส์ (electron tube) ณ เวลานั้น ได้ใช้เทคโนโลยีด้านนี้ของตนร่วมกับเทคโนโลยีด้านอิเล็กทรอนิกส์ในการวิจัยโอซีอาร์ ซึ่งได้ผลเป็นที่น่าพอใจ แต่การวิจัยที่วุ่นวายนี้มิได้มีการสานต่อเพื่อผลิตเป็นผลิตภัณฑ์ออกสู่ตลาด

## 6.2 ประวัติโอชีอาร์เชิงพาณิชย์

เนื่องจากความต้องการของตลาดที่มีต่อโอชีอาร์เพื่อใช้ในการนำเข้าสู่ข้อมูลนั้นเพิ่มสูงขึ้นเรื่อยๆ จึงมีการผลิตระบบโอชีอาร์เป็นสินค้าออกสู่ตลาดอย่างต่อเนื่องมากมายหลายรุ่น ซึ่งระบบต่างๆ เหล่านี้มีคุณสมบัติและความสามารถที่แตกต่างกัน ขึ้นอยู่กับประสิทธิภาพของฮาร์ดแวร์และความก้าวหน้าทางเทคโนโลยีในแต่ละยุคสมัย ซึ่งเราอาจจะแบ่งผลิตภัณฑ์โอชีอาร์ในเชิงพาณิชย์ได้เป็นยุคๆ ตามคุณสมบัติและความสามารถของระบบโอชีอาร์ในแต่ละช่วงเวลาโดยสามารถแบ่งได้เป็น 3 ยุคได้แก่

**ยุคที่ 1** ในช่วงปี พ.ศ. 2503 ถึง พ.ศ. 2508 (ต้นยุค ค.ศ.1960) มีผลิตภัณฑ์โอชีอาร์เชิงพาณิชย์ออกสู่ตลาดมากพอสมควร โดยโอชีอาร์ในยุคนี้จะมีข้อจำกัดในเรื่องฟอนต์และจำนวนตัวอักษรที่ระบบสามารถอ่านได้ ซึ่งจะมีการกำหนดกลุ่มของฟอนต์ที่สามารถใช้กับระบบไว้ หรือไม่ก็สร้างฟอนต์ใหม่ขึ้นมาใช้โดยเฉพาะเลย และมักจะอ่านได้เฉพาะตัวเลขและอักษรเพียงไม่กี่ตัว ตัวอย่างของระบบโอชีอาร์ในกลุ่มนี้ได้แก่ NCR 420 ซึ่งได้ออกแบบฟอนต์พิเศษชื่อ NOF ประกอบด้วยตัวเลขและสัญลักษณ์อีก 5 ตัว, Farrington 3010 เช่นเดียวกับ NCR ที่ต้องใช้กับฟอนต์พิเศษของตัวเอง และบริษัทคอมพิวเตอร์ยักษ์ใหญ่ออย่าง IBM ก็เข้าร่วมวงด้วย เนื่องจากเห็นความสำคัญของโอชีอาร์โดยมี IBM 1418 เป็นโอชีอาร์รุ่นแรกของบริษัทที่ทำออกมาในเชิงพาณิชย์ ใช้งานกับฟอนต์ 407 ของ IBM เอง ซึ่งรูปร่างตัวอักษรของฟอนต์นี้จะมีความเป็นธรรมชาติกว่า 2 รุ่นที่กล่าวมาแล้ว และในช่วงปลายยุค พ.ศ. 2508 (ค.ศ.1960) ในประเทศญี่ปุ่นบริษัทใหญ่ๆ หลายบริษัทได้เปิดตัวระบบโอชีอาร์ซึ่งเข้าอยู่ในกลุ่มยุคแรกเช่นกัน โดยประกอบด้วย Facom 6300A จาก Fujitsu, H-852 จาก Hitachi และ N240D-1 จาก NEC

**ยุคที่ 2** คุณลักษณะที่สำคัญของระบบโอชีอาร์ยุคนี้คือความสามารถในการรู้จำลายมือเขียน โดยในช่วงแรกๆ จะจำกัดไว้เฉพาะตัวเลขเท่านั้น ยุคนี้มีช่วงระยะเวลาอยู่ระหว่างกลางทศวรรษที่ 60 ถึงต้นทศวรรษที่ 70 ระบบโอชีอาร์ที่อยู่ในกลุ่มนี้ก็ว่าได้แก่ IBM 1287 เป็นเครื่องที่รวมฮาร์ดแวร์แบบดิจิทัลกับอนาล็อกไว้ด้วยกัน นอกจากลายมือเขียนแล้วโอชีอาร์ในยุคนี้ยังสามารถอ่านตัวอักษรที่พิมพ์มาจากเครื่องพิมพ์อีกด้วย ซึ่งระบบที่จัดอยู่ในกลุ่มนี้ได้แก่ RETINA โดยตัวอักษรลายมือที่อ่านได้จะเป็นเฉพาะตัวเลขและตัวอักษร C, S, T, X, Z, +, - เท่านั้น ส่วนตัวอักษรจากเครื่องพิมพ์นั้นสามารถอ่านตัวอักษรพิมพ์ใหญ่ ตัวเลข และอักขระพิเศษอีกสี่ตัว รวมเป็น 44 ตัว นอกจากนั้นผู้ใช้ยังสามารถเลือกฟอนต์ที่จะอ่านได้ด้วย RETINA ถือเป็นโอชีอาร์ที่มีประสิทธิภาพมากระบบหนึ่งในขณะนั้น โดยสามารถอ่านตัวอักษรได้ 2,400 ตัวต่อวินาที

**ยุคที่ 3** ยุคนี้ให้ความสนใจกับเอกสารตัวพิมพ์ที่มีคุณภาพต่ำและตัวอักษรลายมือเขียนที่มีกลุ่มตัวอักษรหลากหลายมากขึ้น เช่น ตัวอักษรภาษาจีน เป็นต้น โดยที่โอซีอาร์เชิงพาณิชย์ของยุคนี้ออกสู่ท้องตลาดในช่วงประมาณ พ.ศ. 2518 ถึง พ.ศ. 2528 เราอาจจะแบ่งโอซีอาร์ยุคนี้ออกเป็นกลุ่มย่อยลงไปได้อีกดัง 1. โอซีอาร์สำหรับเอกสารคุณภาพต่ำ เป็นโอซีอาร์ที่สามารถอ่านเอกสารซึ่งมีปัญหาเนื่องมาจากรอยเปื้อน (noise) และตัวอักษรที่ไม่ชัดเจน ตัวอย่างของโอซีอาร์ในกลุ่มนี้ได้แก่ OCR-V100 ของ Toshiba และ IBM 1975 2. ระบบอ่านเอกสารลายมือเขียน โดยเน้นที่เอกสารที่มีทั้งตัวเลขและตัวอักษรเพื่อใช้ในการแปลงข้อมูลที่ได้มาจากแบบฟอร์มต่างๆ 3. ชุดซอฟต์แวร์แทนที่จะพัฒนาระบบที่ประกอบด้วยอุปกรณ์ฮาร์ดแวร์เช่นแต่ก่อน โอซีอาร์สมัยใหม่เริ่มมีการผลิตเป็นซอฟต์แวร์สำหรับใช้กับเครื่องพีซี โดยอาจจะพ่วงมากับเครื่องสแกนแบบตั้งโต๊ะหรือมือถือซอฟต์แวร์เหล่านี้จะมีราคาที่ถูกลงตามลำดับ ซอฟต์แวร์โอซีอาร์ที่ค่อนข้างมีชื่อเสียง ได้แก่ Omnipage ของบริษัท Caere Corporation , WordScan ของ Calera และ Discover ของ Xerox Image System เป็นต้น

### 6.3 ประวัติการวิจัยและพัฒนาโอซีอาร์ภาษาไทย

ในประเทศไทยได้มีการวิจัยและพัฒนาโอซีอาร์สำหรับภาษาไทยมานานประมาณ 10 ปีแล้ว เนื่องจากเทคโนโลยีโอซีอาร์มีความสำคัญต่อกระบวนการประมวลผลข้อมูลในระบบคอมพิวเตอร์เป็นอย่างมาก แต่ภาษาไทยมีลักษณะเฉพาะที่ต่างออกไปจากภาษาอื่นๆ ดังนั้นจึงมีความจำเป็นอย่างมากที่ประเทศไทยจะต้องพัฒนาเทคโนโลยีด้านนี้สำหรับภาษาไทยโดยเฉพาะ การวิจัยส่วนใหญ่เป็นการวิจัยภายในสถาบันการศึกษา และมักจะถูกเลือกเป็นหัวข้อวิทยานิพนธ์ของนักศึกษาทั้งระดับปริญญาโทและปริญญาเอกอยู่เสมอ แต่สถาบันศึกษาที่ให้ความสนใจในการวิจัยอย่างจริงจังและต่อเนื่องนั้น มีอยู่จำนวนจำกัด ซึ่งได้แก่ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังโดย ดร. ชม กิมปาน, สถาบันบัณฑิตพัฒนบริหารศาสตร์ โดย ดร. พิพัฒน์ หิริยวนิชกร และจุฬาลงกรณ์มหาวิทยาลัย โดย ดร. ชิดชนก เหลือสินทรัพย์ นอกจากสถาบันทางการศึกษาดังกล่าวแล้ว ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติซึ่งเป็นหน่วยงานทางการวิจัยภาครัฐก็เล็งเห็นความสำคัญของเทคโนโลยีโอซีอาร์ จึงได้ดำเนินโครงการวิจัยด้านนี้อย่างจริงจังภายใต้การดูแลของ ดร. ทวีศักดิ์ กออนันตกูล และ ดร. จุฬารัตน์ ดันประเสริฐ ซึ่งงานวิจัยนี้ได้รับผลสำเร็จและนำไปสู่การผลิตเป็นโปรแกรมโอซีอาร์ภาษาไทยเชิงพาณิชย์ชื่อ “อ่านไทย” ในเวลาต่อมา

การวิจัยโอซีอาร์ในประเทศไทยนั้นมีทั้งสำหรับแบบตัวพิมพ์และตัวเขียน สำหรับตัวเขียนก็มีการรู้จำทั้งแบบ ออนไลน์และออฟไลน์ ถึงแม้การวิจัยโอซีอาร์ภาษาอื่นๆ เช่น อังกฤษ และญี่ปุ่น เป็นต้น จะมีความก้าวหน้าและประสบความสำเร็จในการนำมาใช้งานอย่างกว้างขวางแล้วก็ตาม แต่การประยุกต์ใช้เทคนิคจากภาษาอื่นมาใช้กับภาษาไทยโดยตรงนั้นไม่สามารถทำได้ จำเป็นต้องมีการคิดค้นวิธีการใหม่ที่เหมาะสมกับภาษาไทย อย่างไรก็ตามงานวิจัยโอซีอาร์สำหรับภาษาไทยก็ประสบความสำเร็จในระดับหนึ่ง และมีซอฟต์แวร์โอซีอาร์เชิงพาณิชย์ออกมาสู่ตลาดแล้วจำนวนหนึ่ง แต่ทั้งหมดยังเน้นหนักไปที่ตัวพิมพ์ และยังมีข้อจำกัดเรื่องฟอนต์ที่สามารถอ่านได้ โดยระดับความถูกต้องของ โอซีอาร์ภาษาไทยจะอยู่ในช่วง 90 – 99 % ซึ่งยังห่างจากความสามารถของโอซีอาร์ภาษาอังกฤษและภาษาอื่นๆ ที่มีระดับความถูกต้องสูงถึง 99.9 % ทั้งนี้ส่วนหนึ่งเนื่องมาจากลักษณะของตัวอักษรภาษาไทยที่มีตัวคล้ายกันแต่ทำหน้าที่แตกต่างกันอยู่หลายตัว รวมทั้งประโยคภาษาไทยที่ประกอบไปด้วยตัวอักษร 4 ระดับ เหล่านี้เป็นปัญหาที่ยากสำหรับการพัฒนาโอซีอาร์ภาษาไทยให้ได้ความถูกต้องที่ 99.9 %

#### 6.4 เทคโนโลยีโอซีอาร์

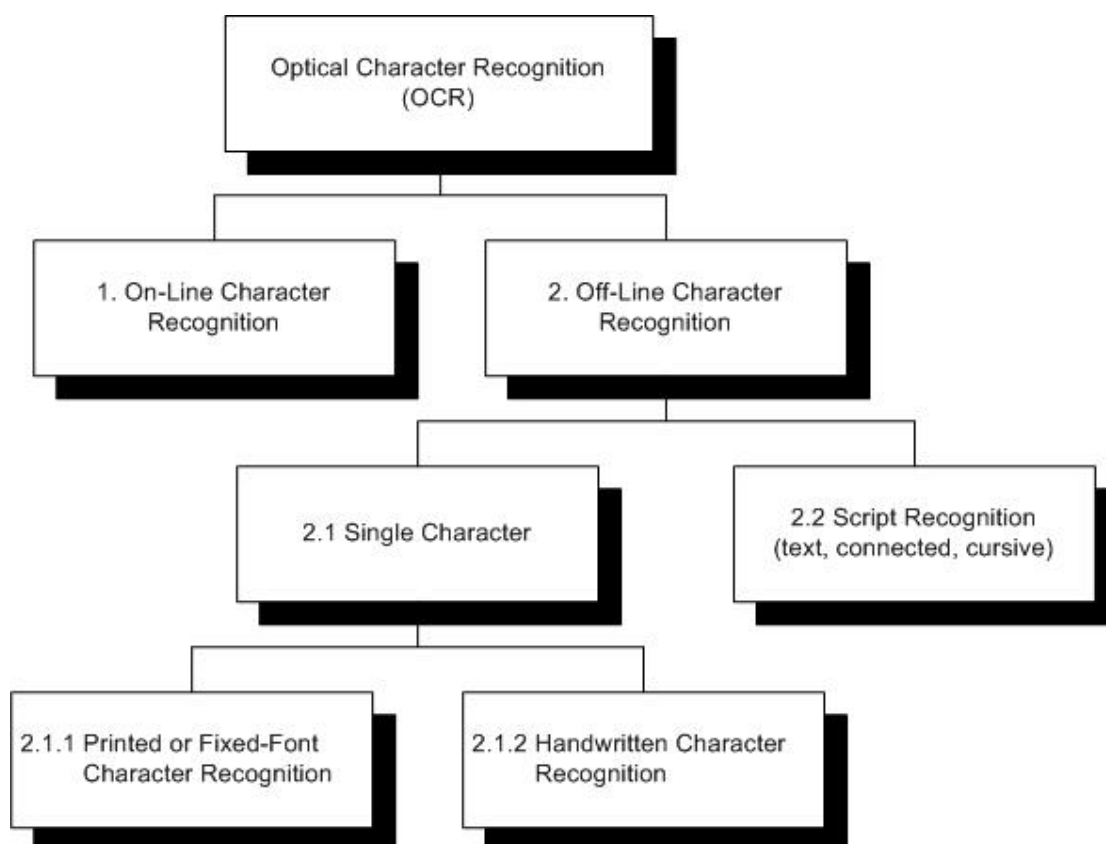
ผลิตภัณฑ์ OCR ที่มีใช้อยู่ในขณะนี้อาศัยเทคโนโลยีที่มีขั้นตอนการทำงานที่ยุ่งยาก ซับซ้อนมาก กว่าเอกสารที่สแกนเข้าไปจะถูกแปลงออกมาเป็นตัวอักษรอย่างที่ต้องการ (ซึ่งบางครั้งก็ได้อะไรที่ไม่ต้องการออกมาด้วย!) รายละเอียดของขั้นตอนการทำงานเหล่านี้จะแตกต่างกันออกไปสำหรับโปรแกรมโอซีอาร์แต่ละตัว ซึ่งส่วนหนึ่งก็เป็นเรื่องรายละเอียดทางด้านเทคนิคที่เป็นกลเม็ดเคล็ดลับของแต่ละบริษัท แต่อีกส่วนก็ขึ้นกับลักษณะงานเฉพาะด้านของโปรแกรมนั้น ทั้งนี้เนื่องจากโปรแกรมประเภทโอซีอาร์อาจแบ่งออกได้เป็นหลายกลุ่ม ซึ่งมีวัตถุประสงค์เพื่อนำไปใช้กับงานที่มีลักษณะแตกต่างกัน ตัวอย่างเช่น โอซีอาร์สำหรับลายมือเขียน (Hand Written OCR) ก็มักจะใช้เทคนิคที่ต่างจากโอซีอาร์สำหรับตัวพิมพ์ (Printed Text OCR) และอื่นๆ ซึ่งเราจะกล่าวถึงรายละเอียดของโอซีอาร์แต่ละกลุ่มในลำดับต่อไป นอกจากนั้นลักษณะของภาษาก็มีอิทธิพลต่อการเลือกใช้เทคนิคด้วย เนื่องจากความยากง่ายและรูปร่างตัวอักษรของแต่ละภาษาแตกต่างกัน ดังนั้นเทคนิคที่ใช้กับภาษาไทยจึงต่างจากภาษาอื่นๆ เช่น ภาษาอังกฤษ และภาษาจีน ซึ่งเงื่อนไขต่างๆ เหล่านี้เป็นเพียงส่วนหนึ่งของความซับซ้อนที่ส่งผลต่อกระบวนการทำงานที่เกิดขึ้นภายในโปรแกรม ทั้งนี้ยังมีส่วนอื่นๆ ที่เป็นข้อกำหนดในการเลือกวิธีการที่เหมาะสมสำหรับจัดการกับปัญหาที่หลากหลาย เพื่อจะได้โปรแกรมโอซีอาร์ที่ดีที่สุด

ก่อนที่จะกล่าวถึงรายละเอียดทางด้านเทคนิคเราควรทำความเข้าใจกับคำว่า “Optical Character Recognition” หรือ OCR กันก่อน ซึ่งถ้าจะแปลเป็นภาษาไทยอย่างตรงไปตรงมา ก็คงจะได้ว่า “การรู้จำตัวอักษรโดยการมองเห็น(ภาพ)” ซึ่งเห็นได้ชัดว่าถ้าจะอธิบายให้คนที่ไม่รู้จักโอซีอาร์ฟัง แล้วเข้าใจด้วยละก็ จำเป็นจะต้องมาแปลให้เป็นภาษาไทยกันอีกที แบบที่จะสามารถเข้าใจกันได้ง่าย ๆ กว่านี้ ฉะนั้นเราน่าจะมาลองแปลกันใหม่ดีกว่า

เริ่มจากคำว่า Optical ซึ่งในที่นี้หมายถึงการอ่านภาพ (ตัวอักษร) ด้วยอุปกรณ์ตรวจจับความเข้มของแสง ซึ่งก็คือเครื่องสแกนหรืออุปกรณ์ที่ทำงานในลักษณะเดียวกันนี้นั่นเอง ที่เป็นตัวรับข้อมูลเข้ามาเป็นรูปภาพ เพื่อนำไปสู่กระบวนการขั้นต่อไป ของโปรแกรม การที่จะต้องมีการประกอบอยู่ด้วยนั้น สืบเนื่องมาจากความพยายามที่จะสร้างเครื่องอ่านเอกสารอัตโนมัติในยุคแรกๆ ซึ่งนอกจากการอ่านข้อมูลแบบ Optical แล้ว ยังมีการใช้วิธีการทางด้าน Magnetic และ Mechanic ด้วย ถึงแม้ในปัจจุบันเราจะไม่ได้ใช้วิธีการทั้งสองแบบหลังนี้แล้วก็ตาม คำว่า Optical ก็ยังคงอยู่เพราะคนส่วนใหญ่จะรู้จักคำว่า OCR ในแง่เป็นคำเฉพาะที่หมายถึงซอฟต์แวร์ประเภทนี้ไปแล้ว ดังนั้นการที่จะเอาตัว O หรือ Optical ออกนั้นคงจะสายเกินไป คำต่อมา Character ก็คือตัวอักษร เพื่อระบุว่าตัวโปรแกรมนั้นทำงานกับอะไร ซึ่งก็คือตัวอักษรที่จะถูกแปลงจากรูปตัวอักษรมาเป็นรหัสตัวอักษร ที่เราสามารถใส่เวิร์ดโปรเซสเซอร์เปลี่ยนแปลงแก้ไขได้ (และต่อจากนี้ขอใช้คำว่ารูปตัวอักษรและรหัสตัวอักษรในการอ้างอิงถึงตัวอักษรทั้งสองแบบดังกล่าว เพื่อให้เกิดความเข้าใจที่ชัดเจนในการสื่อความหมาย) นอกจากนี้ชนิดของตัวอักษรยังใช้เป็นเกณฑ์ในการแบ่งประเภทของโอซีอาร์ด้วย เช่น ตัวพิมพ์ ตัวเขียน เป็นต้น ซึ่งจะกล่าวถึงในหัวข้อถัดไป ส่วนคำสุดท้าย Recognition หมายถึงการบอกว่ารูปภาพของตัวอักษรนั้นเป็นตัวอะไร โดยแปลงให้เป็นรหัสตัวอักษรที่เครื่องคอมพิวเตอร์สามารถนำไปทำงานอื่นๆ ได้ เนื่องจากตัวอักษรตัวหนึ่งๆ สามารถเขียนได้หลากหลายลักษณะ ทำให้งานส่วนนี้กลายเป็นส่วนที่สำคัญที่สุดของระบบ และมีความยุ่งยากซับซ้อนมาก ถึงตอนนี้ลองมาสรุปความหมายของคำว่า “Optical Character Recognition” กันอีกครั้ง ซึ่งน่าจะแปลว่าโปรแกรมที่แปลงภาพตัวอักษรที่ได้จากเครื่องสแกนเนอร์ให้เป็นรหัสตัวอักษรที่เราสามารถเอาไปใช้กับเวิร์ดโปรเซสเซอร์เพื่อที่จะเปลี่ยนแปลงแก้ไขเพิ่มเติมได้

## 6.5 ประเภทของโอซีอาร์

ตามที่ได้กล่าวไว้แล้วข้างต้นเราสามารถแบ่งโปรแกรมโอซีอาร์ได้เป็นกลุ่มๆ ตามลักษณะหรือแหล่งที่มาของตัวอักษรที่ตรงกับความสามารถของโปรแกรม ซึ่งเราสามารถจะแบ่งออกเป็นกลุ่มได้ดังนี้



ภาพที่ 19 ประเภทเทคโนโลยีโอซีอาร์

#### 6.5.1 การรู้จำตัวอักษรแบบออนไลน์ (On-line Character Recognition)

กลุ่มนี้จะต่างจากกลุ่มอื่นๆ ในแง่ของวิธีการอินพุตข้อมูล ซึ่งจะได้มาจากคีย์บอร์ดหรือปากกาอิเล็กทรอนิกส์ที่ใช้กับเครื่องคอมพิวเตอร์มือถือ ส่วนการวิเคราะห์ตัวอักษรจะทำในขณะที่มีการลากเส้นเพื่อเขียนตัวอักษร (ออนไลน์) ซึ่งมีวิธีการทำงานที่ต่างจากวิธีการอื่นๆ ถ้าจะเทียบความยากง่ายกับการรู้จำลายมือเขียนแบบออฟไลน์ กลุ่มนี้จะง่ายกว่าเพราะจะได้ข้อมูลเพิ่มเติมเกี่ยวกับทิศทางและลำดับการลากเส้นมาช่วยด้วย โอซีอาร์กลุ่มนี้มักมาพร้อมกับอุปกรณ์การเขียนที่จะกำหนดพื้นที่ให้อินพุตข้อมูล โดยส่วนใหญ่มักต้องเขียนทีละตัวอักษร ความก้าวหน้าของเทคโนโลยีด้านนี้จะมีผลเป็นอย่างมากต่อการเจริญเติบโตของธุรกิจในการเครื่องคอมพิวเตอร์มือถือที่เป็นคอมพิวเตอร์ขนาดเล็ก ซึ่งต้องอาศัยการใส่อินพุตจากปากกาอิเล็กทรอนิกส์แทนคีย์บอร์ด



ภาพที่ 20 อุปกรณ์รู้จำแบบออนไลน์

#### 6.5.2 การรู้จำตัวอักษรแบบออฟไลน์ (Off-line Character Recognition)

อินพุตของระบบจะเป็นภาพของตัวอักษรที่ได้จากเครื่องสแกน อาจจะเป็นตัวอักษรแบบพิมพ์ หรือแบบเขียน และอาจเป็นตัวอักษรแบบเดี่ยวๆ หรือติดกันเป็นกลุ่มตัวอักษร

ก. ตัวอักษรโดด (Single Character) คือกลุ่มของโปรแกรมไอซีอาร์ที่ทำงานกับภาพตัวอักษรฉบับที่เป็นตัวเดี่ยวๆ ไม่ได้เชื่อมติดกับตัวอื่น โปรแกรมกลุ่มนี้สามารถแบ่งย่อยออกเป็น 2 กลุ่ม ได้แก่

1) การรู้จำตัวพิมพ์แบบฟอนต์เฉพาะ (Printed Fixed-Font Character Recognition) เป็นกลุ่มของโปรแกรมที่ใช้ได้กับตัวอักษรประเภทตัวพิมพ์ที่มีการกำหนดกลุ่มของฟอนต์ ซึ่งถือว่าอยู่ในระดับง่ายที่สุดในด้านการพัฒนาเทคนิคในการทำงานของโปรแกรม ถึงกระนั้นก็ตาม โปรแกรมในกลุ่มนี้ก็ยังประสบกับปัญหาที่เกิดจากเอกสารคุณภาพต่ำ ซึ่งจะส่งผลให้ได้ภาพที่ประอะเปื้อน ทำให้ภาพตัวอักษรติดกันหรือขาดออกจากกัน เหล่านี้เป็นตัวการสำคัญที่ทำให้อัตราความถูกต้องของโปรแกรมลดลง

## เชื่อมติดกัน

ภาพที่ 21 ตัวอักษรที่มีส่วนเชื่อมติดกันและขาดจากกัน

2) การรู้จำลายมือเขียนแบบตัวโดด (Isolated Handprint Character Recognition (ICR)) ตัวอักษรของกลุ่มนี้เป็นลายมือเขียนที่มีจะถูกกำหนดให้เขียนในกรอบที่จัดไว้ โดยเขียนทีละตัวแยกออกจากกัน ตัวอย่างงานที่เข้าข่ายกลุ่มนี้ได้แก่ โปรแกรมโอซีอาร์ที่ใช้แยกจดหมายจากรหัสไปรษณีย์ซึ่งเป็นตัวเลขจากลายมือเขียนเป็นต้น การรู้จำลายมือเขียนเป็นเรื่องยากมาก เพราะตัวอักษรแต่ละตัวที่เขียนโดยคนแต่ละคนมีความหลากหลายมาก ถึงแม้บางครั้งจะเป็นการเขียนโดยคนๆ เดียว การเขียนแต่ละครั้งก็ยังคงแตกต่างกัน ดังนั้นโปรแกรมพวกนี้มักมีข้อกำหนดบางอย่าง เช่น สามารถอ่านได้เฉพาะตัวเลข หรือสามารถอ่านลายมือของคนที่ได้ลองเขียนตัวอักษรตัวอย่างชุดหนึ่งให้โปรแกรมรู้จักก่อนเท่านั้น

ข. การรู้จำลายมือแบบเขียนต่อเนื่อง (Script recognition) กลุ่มนี้ได้โจทย์ในระดับที่ยากที่สุดในจำนวนตระกูลโอซีอาร์ทั้งหลาย เพราะตัวอักษรที่โปรแกรมจะต้องอ่านเป็นตัวอักษรที่เป็นลายมือเขียนโดยไม่มีข้อกำหนดใดๆ ผู้เขียนสามารถเขียนได้ตามธรรมชาติอย่างต่อเนื่อง ดังนั้นตัวอักษรที่ได้อาจมีเส้นที่ลากเชื่อมตัวอักษรหลายๆ ตัวให้ติดกัน และประกอบกับความแตกต่างอย่างมากของลายมือของคนแต่ละคน โปรแกรมในกลุ่มนี้บางที่เรียกว่าการรู้จำลายมือเขียนแบบอิสระ (freestyle handwriting recognition) ถึงแม้จะมีผลิตภัณฑ์โอซีอาร์กลุ่มนี้ออกสู่ตลาดบ้างแล้ว แต่หัวข้อนี้ก็ยังต้องการการวิจัยเพิ่มเติมอีกมาก

### 6.6 การรู้จำแบบ (Pattern Recognition)

โปรแกรมประเภทโอซีอาร์เป็นโปรแกรมที่ประยุกต์ความรู้ด้านการรู้จำแบบ (Pattern Recognition) ซึ่งต้องอาศัยเทคนิคการวิเคราะห์แยกแยะและการระบุชนิด โดยอินพุตของโปรแกรมมักจะต้องผ่านกระบวนการย่อยๆ อีกหลายขั้นขึ้นอยู่กับการประยุกต์ใช้ ได้แก่ การกรองข้อมูลแทรกซ้อน การปรับแต่งข้อมูล การเลือกตัดแบ่งพื้นที่ใช้งาน การสกัดลักษณะสำคัญ การจำแนกแยกแยะการรู้จำและอื่นๆ โดยในแต่ละขั้นตอนก็สามารถจะมีวิธีการได้อีกหลากหลายวิธี ขึ้นอยู่กับตัวข้อมูลและการประยุกต์ใช้

นอกจากโอซีอาร์แล้ว งานประยุกต์ที่แตกแขนงออกมาจากการรู้จำแบบได้แก่ การรับรู้ภาพของคอมพิวเตอร์, การวิเคราะห์การเกิดแผ่นดินไหว, การแยกแยะและวิเคราะห์สัญญาณเรดาร์, การรู้จำใบหน้า, การรู้จำเสียงพูด, การเข้าใจเสียงพูด, การระบุลายพิมพ์นิ้วมือ, การวินิจฉัยทางการแพทย์ และอื่นๆ

## 6.7 โครงสร้างระบบโอซีอาร์

โครงสร้างของระบบโอซีอาร์โดยทั่วไปประกอบไปด้วยขั้นตอนการทำงานหลัก 3 ขั้นตอน



ภาพที่ 22 โครงสร้างทั่วไปของระบบโอซีอาร์

ซึ่งในแต่ละขั้นตอนประกอบด้วยรายละเอียดดังต่อไปนี้

### 6.7.1 ขบวนการประมวลผลขั้นต้น (Pre-Processing)

ในการทำงานของโปรแกรมโอซีอาร์นั้น ก่อนที่โปรแกรมจะสามารถบอกได้ว่ารูปภาพที่ส่งเข้าไปประกอบด้วยตัวอักษรอะไรบ้าง จำเป็นจะต้องผ่านขั้นตอนที่สำคัญหลายขั้นตอนดังกล่าวนี้นี้มักถูกเรียกรวมกันว่า กระบวนการประมวลผลขั้นต้น (Pre-Processing) ซึ่งเป็นขั้นตอนในการปรับแต่งและจัดเตรียมข้อมูลให้เหมาะสมกับขั้นตอนการรู้จำต่อไป ขั้นตอนเหล่านี้มีความสำคัญต่อประสิทธิภาพโดยรวมของระบบ เพราะหากมีความผิดพลาดเกิดขึ้นในส่วนนี้ ก็จะส่งผลกระทบไปยังส่วนถัดไปของระบบด้วย

ขั้นตอนการประมวลผลเบื้องต้นในโปรแกรมโอซีอาร์ที่สำคัญ ได้แก่

ก. การกรองข้อมูลแทรกซ้อน (Noise Filtering) การกรองข้อมูลแทรกซ้อนมีจุดประสงค์เพื่อลดทอนส่วนของรูปภาพที่เป็นสิ่งแปลกปลอมอันไม่พึงประสงค์ออกไป โดยข้อมูลแทรกซ้อนที่เกิดขึ้นส่วนใหญ่มักจะมาจากคุณภาพของเอกสารต้นฉบับที่นำมาทำการอ่าน ซึ่งเป็นต้นเหตุสำคัญที่ทำให้ความถูกต้องของโปรแกรมลดลง จึงจำเป็นที่จะต้องจัดการกับส่วนเกินเหล่านี้

ออกไปให้ได้มากที่สุดเท่าที่จะเป็นไปได้ แต่เรายังไม่มีวิธีการใดที่รับรองได้ว่าสามารถจัดการกับข้อมูลแทรกซ้อนได้ โดยสมบูรณ์ ดังนั้นส่วนการรู้จำของไอซีอาร์ก็จะต้องมีความทนทานต่อการแทรกซ้อนเหล่านี้ได้พอสมควร

ข. การปรับแต่งข้อมูล (Normalization) การปรับแต่งข้อมูลเป็นการปรับภาพตัวอักษรให้อยู่ในรูปแบบที่ระบบต้องการเพื่อนำไปใช้ในขั้นต่อไป ตัวอย่างการปรับแต่งข้อมูลในโปรแกรมไอซีอาร์ต่างๆ ไป อาทิเช่น การปรับขนาดรูปตัวอักษร, การปรับตัวอักษรที่เอียงให้ตรง, การแปลงรูปสี่หรือเกรย์สเกลให้เป็นขาวดำ หรือในทางกลับกัน การแปลงรูปขาวดำให้เป็นสีหรือเกรย์สเกล เป็นต้น

ค. การตัดแบ่งพื้นที่ใช้งาน (Cropping) การตัดแบ่งพื้นที่เป็นการตัดแยกเอาเฉพาะรูปตัวอักษรออกมาจากภาพ เพื่อส่งให้ขั้นตอนการรู้จำในการระบุว่ารูปตัวอักษรนั้นเป็นรหัสอักขระอะไร หลักการพอสั่งเขตที่ใช้สำหรับการตัดรูปตัวอักษรโดยทั่วไปจะใช้พื้นที่สีขาว (สีพื้น) รอบรูปเป็นตัวกำหนดขอบเขตในการตัด ในขั้นตอนนี้มักจะประสบปัญหาที่ส่งผลกระทบต่ออัตราความถูกต้องของระบบโดยรวมอยู่สองปัญหา ปัญหาแรกคือปัญหาตัวติด เกิดจากรูปของตัวอักษรตั้งแต่สองตัวขึ้นไปมีส่วนที่เชื่อมติดกัน ทำให้ไม่สามารถแยกตัวอักษรออกจากกันโดยใช้พื้นที่สีขาวรอบๆ ได้ จำเป็นต้องหาอัลกอริธึมพิเศษมาช่วยในการแยกตัวอักษรออกจากกัน ส่วนปัญหาที่สองในทางตรงกันข้าม เป็นปัญหาตัวขาดที่รูปตัวอักษรหนึ่งๆ ถูกแยกออกเป็นหลายๆ ทำให้เวลาตัดตัวอักษรจากตัวเดียวจะได้เป็นสองตัว ซึ่งก็ต้องหาวิธีการเฉพาะสำหรับมาจัดการอีกเช่นกัน

ง. การสกัดลักษณะสำคัญ (Feature Extraction) การสกัดลักษณะสำคัญเป็นอีกขบวนการหนึ่งที่สำคัญมาก ดำเนินการใหญ่จะแยกส่วนนี้ออกจากการประมวลผลเบื้องต้น ก็จะอยู่ระหว่างขั้นตอนการประมวลผลเบื้องต้นกับขั้นตอนการรู้จำ แต่ในที่นี้ขอรวมไว้ในตอนเดียวกัน การสกัดลักษณะสำคัญเป็นการดึงเอาโครงสร้างพื้นฐานที่สำคัญของตัวอักษรนั้นออกมา โดยโครงสร้างพื้นฐานที่ว่าจะต้องมีการกำหนดไว้ก่อนว่าจะมีอะไรบ้าง มีการนิยามอย่างไร ตัวอย่างเช่น สำหรับภาษาไทยเรากำหนดว่าตัวอักษรภาษาไทยทั้งหมดประกอบด้วยโครงสร้างพื้นฐานคือ เส้นตรง (แนวตั้ง/นอน) เส้นเอียง หัว (วงกลม) ส่วนโค้ง ส่วนเว้า จุดตกกึ่ง จุดตัด เป็นต้น เมื่อเราสามารถแยกเอาองค์ประกอบของตัวอักษรแต่ละตัวออกมาได้แล้ว จากนั้นเราก็นำเสนอรูปภาพของตัวอักษรนั้นในรูปแบบของรายการขององค์ประกอบพื้นฐานต่างๆ แทน ซึ่งจะถูกส่งต่อเป็นอินพุตสำหรับขั้นตอนการรู้จำต่อไป

### 6.7.2 การรู้จำ (Recognition)

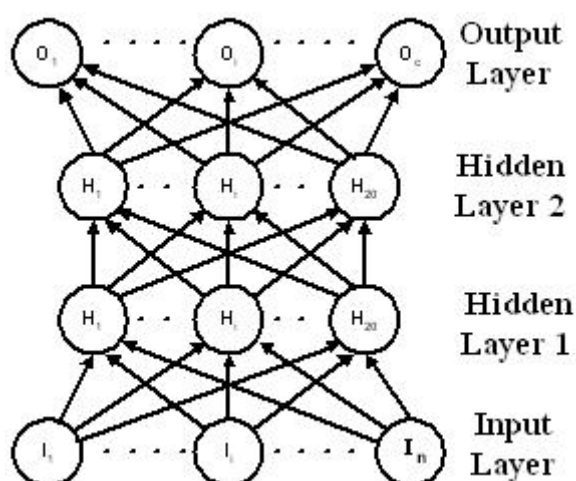
ขั้นตอนนี้ถือเป็นหัวใจของระบบ เพราะเป็นส่วนที่จะตัดสินว่ารูปตัวอักษรที่ส่งเข้าไปเป็นรหัสตัวอักษรอะไร เช่นเดียวกับส่วนอื่นๆ ที่มีวิธีการหลากหลายซึ่งนำมาใช้เพื่อให้ได้ผลการทำงานที่ดีที่สุด เทคนิคใหม่ๆ ได้ถูกพัฒนาขึ้นมาอย่างต่อเนื่องเพื่อรองรับกับปัญหาที่เกิดจากเอกสารที่มีความสลับซับซ้อนมากขึ้น อย่างไรก็ตามเราพอที่จะจัดแบ่งเทคนิคเหล่านี้ออกเป็นกลุ่มตามแนวทางหลักที่ใช้ในการแก้ปัญหา ถึงแม้บ่อยครั้งที่พบว่ามีความคาบเกี่ยวกันของเทคนิคที่นำมาใช้ระหว่างกลุ่มที่ว่านี้ ทั้งนี้เพราะแต่ละแนวทางก็มีข้อดีข้อเสียที่แตกต่างกัน จึงมีความพยายามที่จะรวมเอาเทคนิคเหล่านี้มาใช้ร่วมกันเพื่อเพิ่มความสามารถของระบบเท่าที่จะเป็นไปได้ ดังนั้นการแบ่งกลุ่มในขั้นนี้ เป็นการแบ่งที่เน้นความชัดเจนในแง่ของขอบเขตทางทฤษฎีเป็นหลัก โดยแนวทางการรู้จำสามารถแบ่งออกได้เป็น 4 กลุ่ม ดังนี้

ก. วิธีการเข้ารูปร่างแบบ (Template Matching) วิธีการเข้ารูปร่างแบบเป็นวิธีการแรกๆ ที่มาใช้ในการรู้จำตัวอักษร หลักการโดยทั่วไปคือ จะต้องมียูปร่างแบบ (template) ที่สร้างขึ้นมาสำหรับอ่านตัวอักษร โดยมีการกำหนดตำแหน่งสำคัญที่สามารถใช้แยกแยะความแตกต่างระหว่างตัวอักษรแต่ละตัว เวลาทำงานก็ให้นำรูปภาพที่ต้องการอ่านไปหาบนแบบเพื่อวัดความคล้ายคลึงกันของภาพกับตัวแบบ จากนั้นก็จะระบุว่าเป็นรหัสตัวอักษรอะไร โดยใช้ค่าผ่านระดับหรือวิธีการบางอย่างในการตัดสิน วิธีการนี้จะค่อนข้างอ่อนไหวต่อข้อมูลแทรกซ้อน ขนาด และการเอียงของตัวอักษร จึงจำเป็นต้องมีขั้นตอนการปรับแต่งข้อมูลที่ดี นอกจากนั้นขั้นตอนการเปรียบเทียบก็ไม่ใช่จะสามารถเทียบกันแบบจุดต่อจุดได้ เพราะในทางปฏิบัติตัวอักษรที่ส่งเข้าสามารถมีความแปรปรวนได้หลายรูปแบบ ดังนั้นวิธีการเทียบก็ต้องมีประสิทธิภาพเพียงพอที่จะรองรับกับปัญหาดังกล่าวได้

ข. วิธีการสถิติ (Statistical Approach) วิธีการสถิติเป็นวิธีการที่ใช้หลักการทางสถิติ โดยนำค่าความน่าจะเป็นและ/หรือฟังก์ชันการแจกแจงความน่าจะเป็นมาใช้ในการตัดสินใจ รูปภาพอินพุตที่ได้มาจากขั้นตอนการสกัดลักษณะสำคัญ จะถูกส่งเข้าไปในส่วนการรู้จำเฉพาะของแต่ละตัวอักษร ซึ่งได้ผลลัพธ์ออกมาเป็นค่าความน่าจะเป็นที่อินพุตเป็นตัวอักษรใด เมื่ออินพุตได้ผ่านส่วนการรู้จำครบทุกตัวแล้ว ก็นำเอาผลลัพธ์ที่ได้ทั้งหมดมาเปรียบเทียบกันว่าได้ค่าความน่าจะเป็นของตัวอักษรใดมากที่สุด ผลลัพธ์จะออกเป็นตัวอักษรนั้น

ค. วิธีการวิเคราะห์ทางโครงสร้าง (Structural Analysis) วิธีการวิเคราะห์ทางโครงสร้างคือการวิเคราะห์โครงสร้างตัวอักษร โดยถือว่าตัวอักษรทุกตัวประกอบด้วยองค์ประกอบพื้นฐาน ซึ่งได้มาจากการสกัดลักษณะสำคัญ เช่นเดียวกันกับวิธีการทางสถิติ ต่างกันตรงที่ลักษณะสำคัญ ที่ส่งมาให้กับขั้นตอนการรู้จำแบบการวิเคราะห์ทางโครงสร้างนี้ มักจะใช้เป็นชื่อหรือค่าที่บอกาลักษณะโครงสร้างสำคัญนั้นเป็นอะไร เช่น เส้นตรง วงกลม เป็นต้น แทนที่จะเป็นค่าจำนวนจริง ในขั้นตอนการรู้จำลักษณะสำคัญทั้งหลายที่ประกอบเป็นตัวอักษรนั้น จะถูกส่งเข้าไปให้กับส่วนที่ตรวจวิเคราะห์กฎการเขียนตัวอักษร เช่น ฟอรั่มอลแกรมมาแมชชีน (formal grammar machine) โครงสร้างกราฟ หรือโครงสร้างต้นไม้เป็นต้น เพื่อระบุว่าเป็นตัวอะไร ซึ่งจะตัดสินโดยการดูที่รูปแบบการเชื่อมต่อขององค์ประกอบต่างๆ เข้าเป็นตัวอักษรนั้น วิธีการนี้มีข้อดีตรงที่มีความยืดหยุ่นต่อความหลากหลายของตัวอักษรค่อนข้างมาก อย่างไรก็ตามอัตราความถูกต้องของวิธีนี้ขึ้นอยู่กับ การสร้างกฎและการวิเคราะห์กฎที่มีประสิทธิภาพ ซึ่งเป็นส่วนที่สำคัญที่สุดของวิธีการนี้

ง. วิธีทางโครงข่ายประสาทเทียม (Neural Network) วิธีทางโครงข่ายประสาทเทียมเป็นแนวทางใหม่ที่ได้รับการนิยมนอย่างมากในช่วงหลัง เนื่องจากประสิทธิภาพในด้านการรู้จำแบบ ซึ่งถูกนำไปใช้ในงานหลายๆ ด้าน รวมทั้งโอซีอาร์ด้วย โครงข่ายประสาทเทียมเป็นเทคนิคที่พยายามเลียนแบบการทำงานของสมองมนุษย์ ที่มีโครงข่ายเชื่อมต่อกันของหน่วยความจำย่อยๆ จำนวนมากที่สะสมความรู้เอาไว้ ความรู้เหล่านี้จะได้รับการฝึกสอนไว้ก่อน เช่นการสอนให้รู้จักตัวอักษร “ก” ถึง “ฮ” โดยการส่งภาพตัวอักษรเหล่านี้เข้าไป พร้อมกับบอกว่ามีค่าเป็นรหัสตัวอักษรอะไร โครงข่ายประสาทเทียมจะเรียนรู้ถึงรูปแบบตัวอักษรที่หลากหลายของตัวอักษรตัวนั้น เพื่อว่าเวลาทำงานจริงจะได้มีความสามารถพอที่จะรับมือกับภาพตัวอักษรในหลายๆ รูปแบบ สิ่งที่สอนให้กับโครงข่ายประสาทเทียมไม่จำเป็นต้องเป็นรูปของตัวอักษรอย่างที่เราเห็นกันก็ได้ อินพุตที่ส่งให้มักจะผ่านขั้นตอนการสกัดลักษณะสำคัญ และกระบวนการประมวลผลเบื้องต้นอื่นๆ ก่อนเสมอ



ภาพที่ 23 แสดงโครงสร้างของ Neural Network

### 6.7.3 ขบวนการประมวลผลขั้นปลาย (Post-Processing)

หลังจากที่ผ่านขั้นตอนการรู้จำแล้ว รูปตัวอักษรที่ถูกส่งเข้าไปจะได้ผลลัพธ์ออกมาเป็นรหัสตัวอักษร ซึ่งก็ไม่ได้หมายความว่าเอาท์พุตที่ได้มาจะถูกต้องทั้งหมด ไม่มีผลิตภัณฑ์ไอซีอาร์ตัวใด ไม่ว่าจะเป็นภาษาใดก็ตามที่รับรองความถูกต้อง 100 % ดังนั้นเพื่อเพิ่มความถูกต้องให้แก่โปรแกรมจึงได้มีการเสริมส่วนการตรวจสอบและแก้ไขข้อความเข้ามา โปรแกรมส่วนนี้มักจะทำงานเกี่ยวกับการตรวจสอบความถูกต้องของการสะกดคำและไวยากรณ์ภาษา โดยมักจะใช้พจนานุกรมมาช่วยในการตรวจสอบคำผิด ซึ่งอาจแก้ไขให้โดยอัตโนมัติหรือแสดงเครื่องหมายบางอย่างเพื่อบอกให้ผู้ใช้ทราบว่าคำดังกล่าวอาจไม่ถูกต้อง ซึ่งผู้ใช้อาจแก้ไขหรือไม่แก้ไขขึ้นกับการตัดสินใจของผู้ใช้เอง นอกเหนือไปจากการตรวจสอบความถูกต้องระดับคำแล้ว บางโปรแกรมยังมีความสามารถตรวจสอบไวยากรณ์ในระดับประโยคได้ด้วย

ส่วนขบวนการประมวลผลขั้นปลายในโปรแกรมไอซีอาร์เป็นส่วนที่มีความสำคัญมาก ถ้าพึ่งแต่ความสามารถของส่วนการรู้จำนั้นไม่สามารถไปถึงระดับที่ผู้ใช้ยอมรับได้ (ซึ่งที่ต้องการจริงๆ คือ 100%) ดังนั้นส่วนนี้สามารถเพิ่มอัตราความถูกต้องให้แก่โปรแกรมได้ โดยเฉพาะในส่วนที่นอกเหนือจากความสามารถของส่วนการรู้จำ เช่น ตัวอักษรที่เปราะเปื้อนมากๆ หรือตัวอักษรที่ติดหรือขาด เป็นต้น

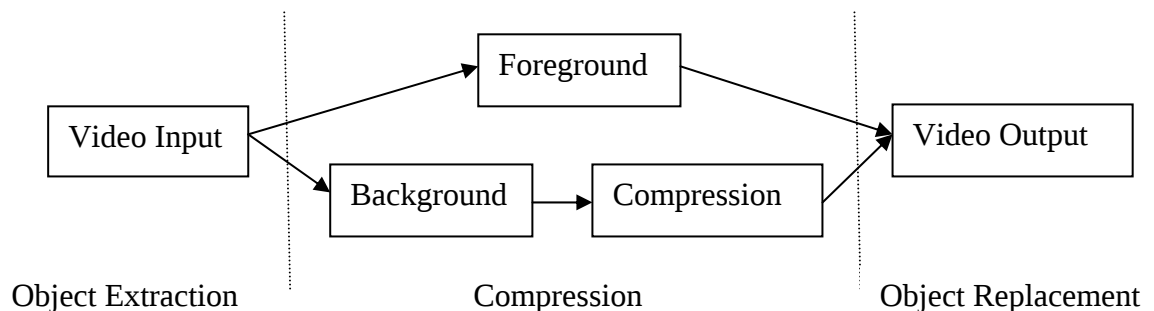
## อุปกรณ์และวิธีการ

### อุปกรณ์

1. กล้องถ่ายภาพวิดีโอ
2. คอมพิวเตอร์
3. โปรแกรม MATLAB
4. เครื่องพิมพ์

### วิธีการ

ขั้นตอนหลัก ๆ ของการทดลองจะแบ่งเป็นสามขั้นตอน นั่นคือ ขั้นตอนแรกคือการแบ่งระดับของวิดีโอชั้น (Video Object Layer - VOL) ให้วัตถุที่เคลื่อนที่เป็นวัตถุพื้นหน้าและให้วัตถุที่อยู่นิ่งเป็นพื้นหลัง จากนั้นนำวัตถุพื้นหลังที่ได้จากขั้นตอนแรกมาทำการบีบอัด (Compression) ตามมาตรฐานของ MPEG ส่วนวัตถุพื้นหน้าจะไม่ทำการบีบอัดเพื่อให้ภาพเกิดการสูญเสียคุณภาพ และขั้นตอนสุดท้ายเอาผลลัพธ์ที่ได้จากขั้นตอนที่สองมารวมกัน โดยในงานวิจัยนี้จะทำการทดลองในขั้นตอนการบีบอัดออกเป็นสองแบบนี้คือ ในขั้นตอนของการแยกระหว่างวัตถุพื้นหน้า (Foreground) และวัตถุพื้นหลัง (Background) จะแยกกัน (Neri *et al.*, 1997) โดยการวิเคราะห์จากการเคลื่อนที่ของวัตถุ และวิเคราะห์จากสีของภาพที่ได้จากกล้อง FLIR



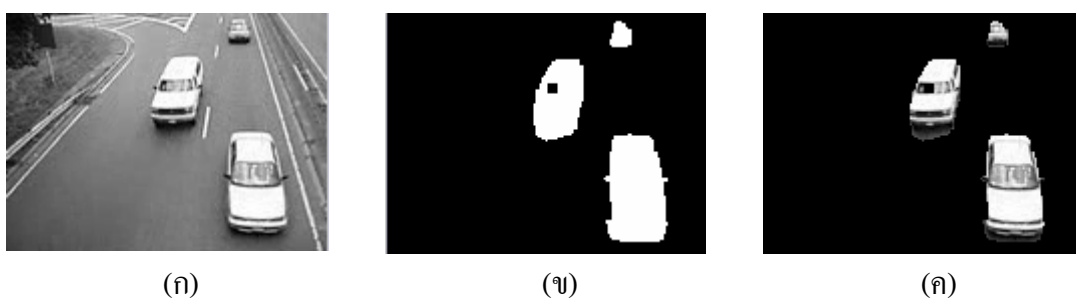
ภาพที่ 24 ขั้นตอนการบีบอัดข้อมูล

## 1. การแยกวัตถุ (Object Extraction)

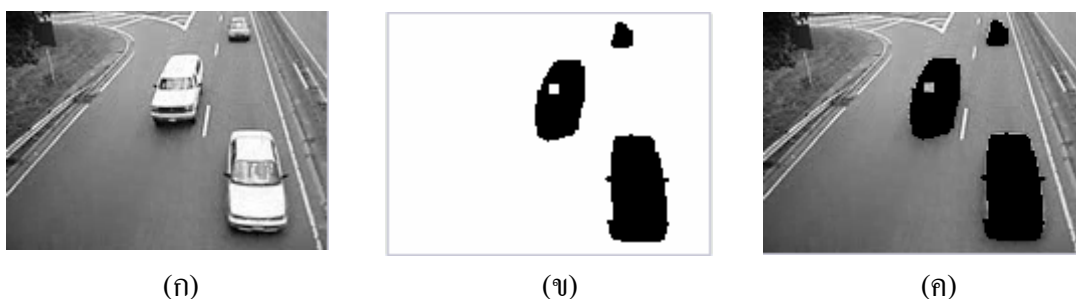
ขั้นตอนการแยกวัตถุนี้มีความสำคัญอยู่ที่หาท่อน (Segment) ของวัตถุที่เป็นพื้นหน้าและเป็นพื้นหลังซึ่งการแบ่งนี้จะทำโดยให้วัตถุเคลื่อนที่ในแต่ละพิกเซล มีค่าเป็น 1 (ขาว) ส่วนที่เหลือมีค่าเป็น 0 (ดำ) จากนั้นนำภาพทำคูณ เพื่อจะได้วัตถุที่ต้องการ

### 1.1 การแยกวัตถุจากการเคลื่อนที่

ในงานวิจัยชิ้นนี้จะวิเคราะห์วัตถุพื้นหน้า ที่เคลื่อนที่และพื้นหลังที่อยู่นิ่งโดยยังไม่สนใจการเคลื่อนที่ของกล้องโดยจะนำภาพการตั้งกล้องที่ท้องถนนแล้วทำการแยกระหว่างรถที่เคลื่อนที่กับวัตถุอื่น ๆ ที่อยู่นิ่ง ได้แก่ ถนน ต้นไม้ข้างทาง และ ป้าย เป็นต้น



ภาพที่ 25 การแยกพื้นหน้า, (ก) Original, (ข) Segment, (ค) Foreground

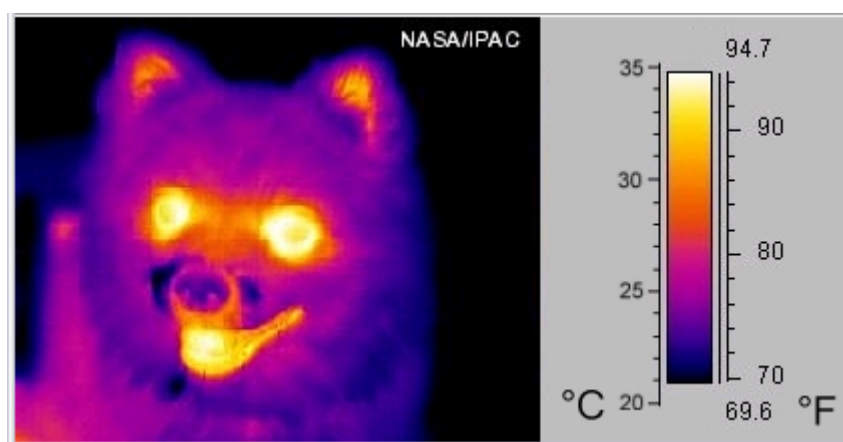


ภาพที่ 26 การแยกพื้นหลัง, (ก) Original, (ข) Segment, (ค) Foreground

ในขั้นตอนของการทำท่อน จะทำการหาพื้นหลังโดยใช้เครื่องมือของ MatLab ที่ชื่อ Temporal Median Estimator จากนั้นก็นำวีดิทัศน์เริ่มต้นมาเปรียบเทียบเพื่อแบ่งแยกวัตถุและให้วัตถุที่สนใจมีค่าเป็น 1(ขาว) วัตถุที่ไม่สนใจมีค่าเป็น 0(ดำ) ดังตัวอย่างภาพที่ 18 (ข) และ ภาพที่ 19 (ข)

## 1.2 การแยกวัตถุจากสีของกล้อง FLIR

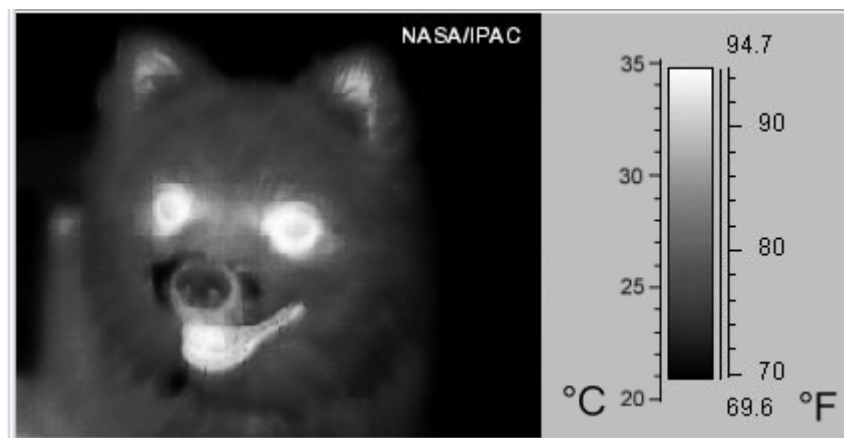
จากคุณสมบัติของภาพที่ถ่ายด้วยกล้อง FLIR ที่มีการแบ่งช่วงของสีตามอุณหภูมิความร้อนของวัตถุ ดังนั้นจึงนำเอาคุณสมบัตินี้มาทำการแบ่งแยกวัตถุได้โดยก่อนที่จะทำการแบ่งนั้น จะต้องปรับรูปแบบของภาพที่แสดงในรูปของสี RGB ให้เป็นภาพที่แสดงความเข้ม (Intensity) เนื่องจาก RGB นั้นจะมีค่าของสีทั้งหมด 3 ค่า คือค่าของสีแดงมีค่าอยู่ระหว่าง 0 ถึง 255 สีเขียวมีค่าอยู่ระหว่าง 0 ถึง 255 และสีน้ำเงินมีค่าอยู่ระหว่าง 0 ถึง 255 เช่นเดียวกัน ซึ่งในการวิเคราะห์แต่ละครั้งจะต้องนำค่าทั้งสามค่ามาคำนวณ ซึ่งมีความยุ่งยากเป็นอย่างมาก แต่ถ้าปรับมาใช้สีในรูปความเข้ม แทนนั้นจะนำมาคำนวณเพียงเลขเดียวเท่านั้น โดยค่าจะอยู่ในช่วง 0 คือสีดำ ถึง 1 คือสีขาว แล้วจึงนำค่าที่ได้ทำการกรองตามความเหมาะสม



ภาพที่ 27 ตัวอย่างภาพที่ถ่ายด้วยกล้อง FLIR

จากภาพจะเห็นว่าสีของภาพจะถูกระบุอุณหภูมิความร้อนของวัตถุทางขวามือของรูปภาพ นั่นคือ ในแสงที่เป็นสีดำจะเป็นส่วนที่มีอุณหภูมิต่ำซึ่งแสดงว่าไม่ใช่พื้นที่ของวัตถุที่สนใจ ในขณะที่แสงสีขาวจะเป็นพื้นที่ของวัตถุที่มีความร้อนสูง

ก่อนที่ทำการแบ่งพื้นหน้าและพื้นหลังของภาพจะทำการปรับรูปแบบของรูปภาพให้แสดงในลักษณะความเข้มของแสงก่อนดังภาพ



ภาพที่ 28 การปรับภาพให้อยู่ในรูปความเข้มของแสง (Intensity)

จากนั้นทำการกรองภาพเพื่อหาท่อน ซึ่งจะนำไปหาภาพพื้นหน้า และภาพพื้นหลังต่อไป

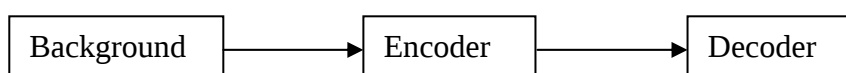


ภาพที่ 29 Segment ที่ได้จากการกรองภาพ FLIR

วิธีการแยกวัตถุนั้นจะนำท่อน ที่ได้มาคูณกับเวกเตอร์เริ่มต้นซึ่งคุณสมบัติการคูณของ 1 เมื่อค่าใด ๆ มาคูณจะทำให้เป็นค่าเดิม ขณะเดียวกันคุณสมบัติของ 0 เมื่อค่าใด ๆ มาคูณจะทำให้มีค่าเป็นศูนย์ ตามเหตุผลข้างต้นนี้จึงสามารถแบ่งวัตถุด้วยการนำมาคูณกับท่อนทุกๆ พิกเซล

## 2. การบีบอัด (Compression)

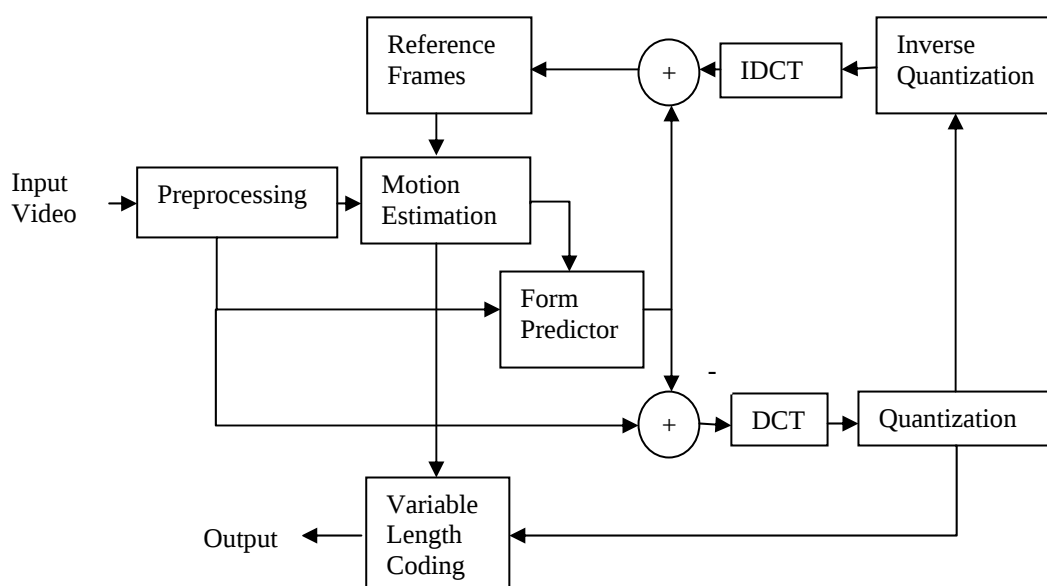
การบีบอัดวิดีโอในงานวิจัยนี้ จะทำการบีบอัดเฉพาะวัตถุพื้นหลังที่อยู่นิ่ง ส่วนวัตถุพื้นหน้าที่เคลื่อนไหวจะไม่ทำการบีบอัด เพื่อไม่ให้คุณภาพของภาพที่เราสนใจสูญหายไป โดยการบีบอัดจะมีขั้นตอนคือการเข้ารหัส (Encoder) และถอดรหัส (Decoder)



ภาพที่ 30 การเข้ารหัสพื้นหลัง

### 2.1 การเข้ารหัส (Encoder)

การเข้ารหัสของ MPEG จะมีขั้นตอนดังภาพ (Gibson *et al.*, 1998)

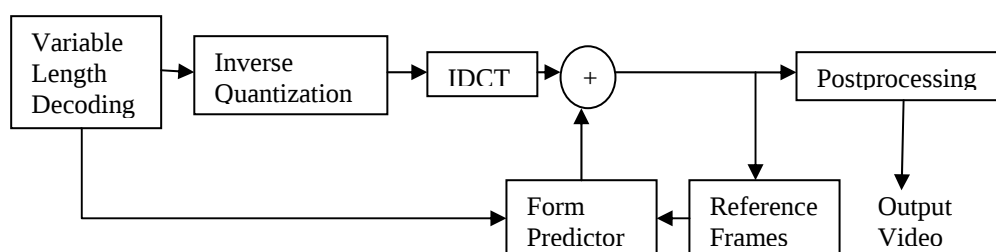


ภาพที่ 31 Block Diagram ของการเข้ารหัส

ก่อนที่จะทำการเข้ารหัสจะต้องทำการเปลี่ยน Color Space จาก RGB เป็น YCbCr โดยให้อัตราส่วนของ Y:Cb:Cr เป็น 4:2:0 จากนั้นเข้าสู่กระบวนการของการเข้ารหัสการชดเชยการเคลื่อนที่ และทำการแปลงรหัส (Transform Coder) ซึ่งมีลำดับของการทำงานคือ DCT, Quantization และ Entropy Encoding ซึ่งจะมีกระบวนการย่อยคือ Zig-Zag Scan และ Variable Length Coding

## 2.2 การถอดรหัส (Decoder)

การถอดรหัสจะมีขั้นตอนดังนี้



ภาพที่ 32 Block Diagram ของการถอดรหัส

หลังจากผ่านกระบวนการเข้ารหัสและถอดรหัสแล้ว Video Input และ Video Output จะเป็นดังนี้



(ก)

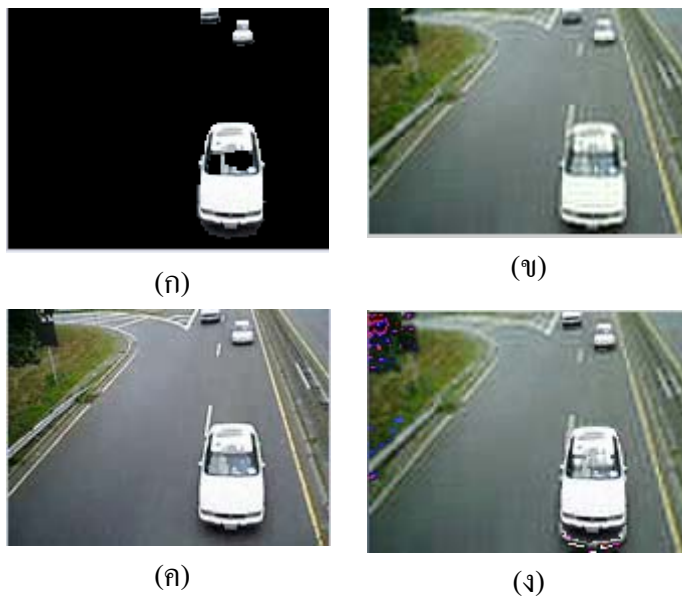


(ข)

ภาพที่ 33 (ก) Video Input และ (ข) Video Output หลังจาก Compression

### 3. การรวมวัตถุ (Object Replacement)

เมื่อนำภาพพื้นหลังมาเข้ารหัสแล้วจึงนำภาพที่ได้เข้ามารวมกับภาพพื้นหน้าที่ได้ทำการผ่านกระบวนการ Segment ในขั้นตอนแรกซึ่งผลลัพธ์ที่ได้จะเป็นดังนี้



ภาพที่ 34 (ก) Foreground, (ข) Background Compression, (ค) Original Video  
และ (ง) Video Output

## ผลและวิจารณ์

### ผล

จากการทดลองการพัฒนาการโปรแกรมโดยใช้ Matlab(The MathWork's Team, 2007) ผลที่ได้วัตถุที่เคลื่อนที่ซึ่งให้เป็นวัตถุพื้นหน้าจะไม่มีการสูญเสียคุณภาพของภาพไปเพราะไม่ได้ผ่านกระบวนการ การบีบอัด ซึ่งต่างจากวัตถุพื้นหลังหรือวัตถุที่อยู่นิ่งที่จะมีการสูญเสียคุณภาพของภาพไปเพราะได้ผ่านกระบวนการบีบอัดด้วยวิธีการตามมาตรฐานของ MPEG เพื่อให้ข้อมูลมีขนาดเล็กลงแต่คงไว้ซึ่งคุณภาพของภาพหรือวัตถุที่สนใจในที่นี้คือวัตถุที่เคลื่อนที่ ทั้งนี้ในส่วนของการแยก วัตถุด้วยการวิเคราะห์ด้วยเงตสีจากภาพถ่ายที่ได้จากกล้องตรวจจับความร้อน สามารถปรับเพื่อควบคุมขอบเขตของวัตถุที่สนใจได้ เพื่อนำไปประกอบกับการบีบอัดต่อไปได้

เมื่อทำการปรับค่าในขั้นตอนของการทำ การวิเคราะห์เชิงปริมาณ เพื่อแสดงอัตราการบีบอัด และผลของการแสดงภาพจะมีค่าดังต่อไปนี้

ค่าตัวแปรคุณภาพ(Gain) = 256



(ก)



(ข)

ภาพที่ 35 ผลการบีบอัดจากการปรับค่าตัวแปรคุณภาพ เท่ากับ 265 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย

ค่าตัวแปรคุณภาพ = 128



(ก)



(ข)

ภาพที่ 36 ผลการบีบอัดจากการปรับค่าตัวแปรคุณภาพ เท่ากับ 128 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย

ค่าตัวแปรคุณภาพ = 64



(ก)



(ข)

ภาพที่ 37 ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 64 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย

ค่าตัวแปรคุณภาพ = 32



(ก)



(ข)

ภาพที่ 38 ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 32 ของขั้นตอนการวิเคราะห์เชิงปริมาณ (ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย

ค่าตัวแปรคุณภาพ = 16



(ก)



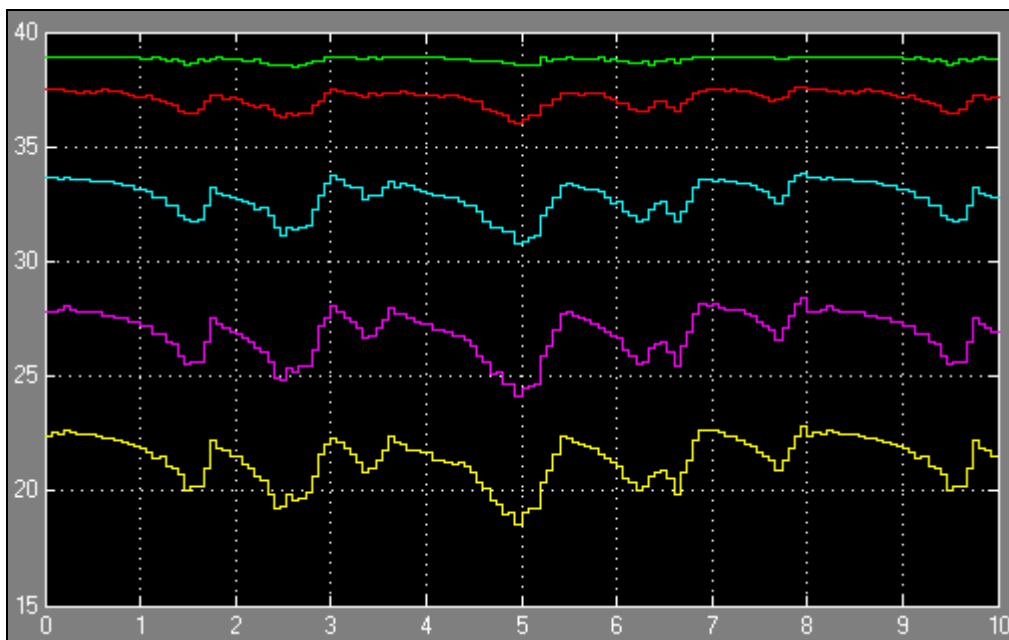
(ข)

ภาพที่ 39 ผลการบีบอัดจากการปรับค่าคุณภาพ เท่ากับ 16 ของขั้นตอนการวิเคราะห์เชิงปริมาณ

(ก) วิธีการตามมาตรฐาน MPEG, (ข) ผลจากการวิจัย

และเมื่อนำอัตราการบีบอัดแสดงในแผนภาพเชิงเส้นแบ่งตามค่าตัวแปรคุณภาพ ในช่วงการแสดงผลวิดีโอทัศน์จะได้แผนภาพดังภาพที่ 33 และ 34

อัตราการบีบอัด

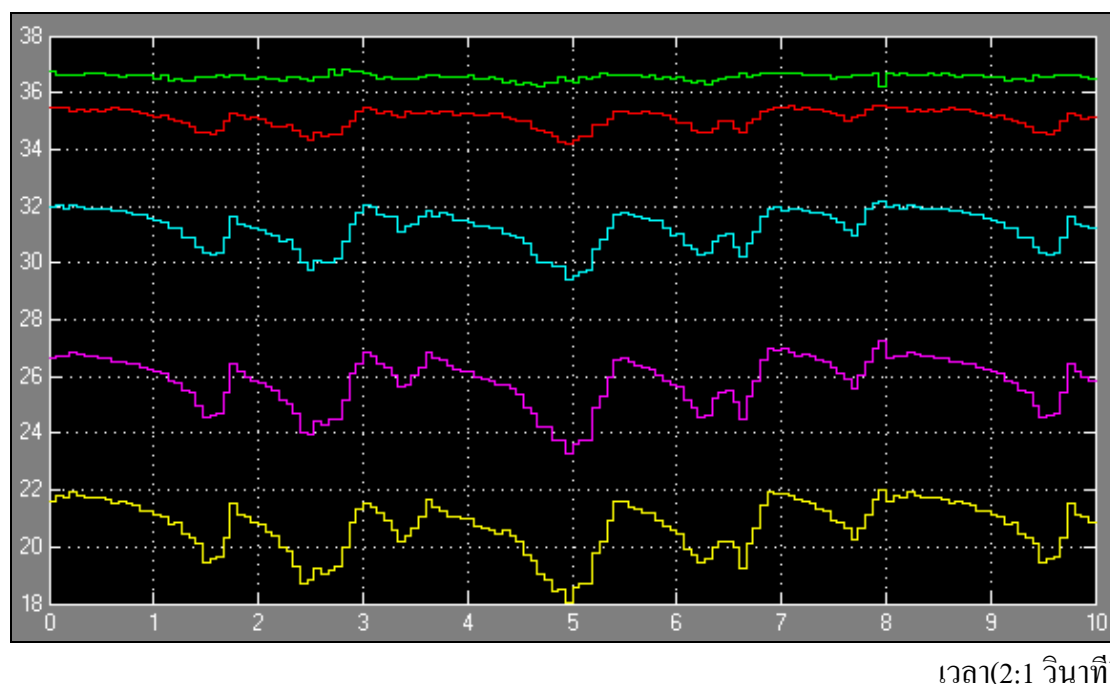


เวลา(2:1 วินาที)

ภาพที่ 40 แผนภาพแสดงอัตราการบีบอัดของวิดีโอทัศน์จากการปรับค่าตัวแปรคุณภาพ

ตามมาตรฐาน MPEG

## อัตราการบีบอัด



**ภาพที่ 41** แผนภาพแสดงอัตราการบีบอัดของวิดีโอจากการปรับค่าตัวแปรคุณภาพ  
จากผลการวิจัย

สีเขียว แสดงอัตราการบีบอัดของค่าตัวแปรคุณภาพที่ 256

สีแดง แสดงอัตราการบีบอัดของค่าตัวแปรคุณภาพที่ 128

สีฟ้า แสดงอัตราการบีบอัดของค่าตัวแปรคุณภาพที่ 64

สีม่วง แสดงอัตราการบีบอัดของค่าตัวแปรคุณภาพที่ 32

สีเหลืองแสดงอัตราการบีบอัดของค่าตัวแปรคุณภาพที่ 16

จากผลการทดลองทำให้เห็นว่าการปรับค่าตัวแปรคุณภาพให้มีค่าสูง ๆ จะทำให้อัตราการบีบอัดแปรผันตาม แต่ภาพที่แสดงออกมาก็จะแปรผันตามค่าตัวแปรคุณภาพ นั่นคือ ยิ่งค่าตัวแปรคุณภาพมีค่าสูง จะทำให้ความละเอียดของภาพมีค่าลดลง ซึ่งเมื่อเปรียบเทียบกับภาพที่ได้จากงานวิจัยชิ้นนี้ ไม่ว่าค่าตัวแปรคุณภาพจะถูกปรับค่าให้สูงขึ้นหรือลดลง แต่ความละเอียดของภาพพื้นหน้ายังมีความละเอียดสูงเสมอ แต่อย่างไรก็ดี เมื่อเปรียบเทียบอัตราการบีบอัดแล้วการบีบอัดด้วยวิธีการตามมาตรฐาน MPEG ยังมีอัตราการบีบอัดสูงกว่าอัตราการบีบอัดจากงานวิจัยในค่าตัวแปรคุณภาพเดียวกัน

หลังจากได้ผลของอัตราการบีบอัดแล้ว งานวิจัยชิ้นนี้จึงได้นำผลการของอัตราการบีบอัด และความละเอียดของภาพมาทำการเปรียบเทียบ ซึ่งจะมีค่าตัวแปรคุณภาพเป็นตัวควบคุม โดยใช้วิธีการดูตัวเลขทะเบียนรถที่วิ่งผ่านในช่วงเวลา 30 วินาที แล้วนับตัวอักษรจากการบีบอัดทั้งสองแบบ ดังนี้



(ก)



(ข)



(ค)

**ภาพที่ 42** ภาพแสดงความละเอียดของภาพถ่ายวิดีโอที่บันทึกจากการบีบอัดด้วยค่าตัวแปรคุณภาพ เท่ากับ 16 (ก) ภาพไม่ถูกบีบอัด (ข) บีบอัดตามมาตรฐาน MPEG (ค) บีบอัดจากผลการวิจัย

และเมื่อนำจำนวนตัวอักษรที่นับได้จากรถที่วิ่งผ่านในช่วงเวลา 30 วินาที จะได้ค่าตามตารางที่ 3 และ 4

**ตารางที่ 3** ตารางแสดงคุณภาพการบีบอัดตามมาตรฐาน MPEG

| ค่าตัวแปรคุณภาพ | อัตราการบีบอัด(เฉลี่ย) | % ตัวอักษรที่นับได้ |
|-----------------|------------------------|---------------------|
| 16              | 21                     | 86.96               |
| 32              | 26.5                   | 65.22               |
| 64              | 33                     | 0                   |
| 128             | 37                     | 0                   |
| 256             | 39                     | 0                   |

ตารางที่ 4 ตารางแสดงคุณภาพการบีบอัดจากผลการวิจัย

| ค่าตัวแปรคุณภาพ | อัตราการบีบอัด(เฉลี่ย) | % ตัวอักษรที่นับได้ |
|-----------------|------------------------|---------------------|
| 16              | 20.0                   | 100                 |
| 32              | 25.5                   | 100                 |
| 64              | 31.0                   | 100                 |
| 128             | 35.0                   | 100                 |
| 256             | 36.5                   | 100                 |

จากตารางทั้งสองจะเห็นว่าอัตรา การบีบอัดที่กำหนดค่าตัวแปรคุณภาพเดียวกัน อัตราการบีบอัดเฉลี่ยตามมาตรฐาน MPEG จะมีค่าสูงกว่าอัตราการบีบอัดเฉลี่ยจากผลการทดลอง แต่เมื่อเปรียบเทียบจำนวนร้อยละของตัวอักษรที่นับได้ ในการบีบอัดตามมาตรฐาน MPEG จะสามารถนับเมื่อค่าตัวแปรคุณภาพตั้งแต่ 32 และต่ำกว่า แต่เมื่อตัวแปรคุณภาพมีค่าสูงกว่านี้ ตัวอักษรจะไม่แสดงรูปทรงให้เห็นชัดเจนเลย ในขณะที่ผลการนับตัวอักษรจากผลการทดลอง สามารถมองเห็นได้ทุกตัวอักษรในทุกค่าตัวแปรคุณภาพ

ในการวัดคุณภาพด้วยการประเมินด้วยสายตามนุษย์อาจเกิดข้อผิดพลาดได้ ดังนั้น ในการทดลองครั้งนี้ ผู้วิจัยจะใช้การอ่านวิธีทัศน์ตัวอักษรแล้วนำมาผ่านกระบวนการบีบอัดตามการทดลองข้างต้น แล้วนำการอ่านอักขระด้วยแสง (OCR) มาอ่านตัวอักษรที่ได้ในค่าตัวแปรคุณภาพต่าง ๆ ดังนี้

ข้อความที่ใช้ คือ ABCDEFGH เป็นชนิดอักษร Cordia New ซึ่งอ่านด้วยโปรแกรม SimpleOCR โดยจะแสดงผลการอ่านอักขระด้วยแสงดังตารางที่ 5-7

ตารางที่ 5 ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปรคุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 24

| ค่าตัวแปร<br>คุณภาพ | ผลการอ่าน<br>อักขระด้วยแสง | % ความถูกต้อง | ผลการอ่านอักขระ<br>ด้วยแสง(MPEG) | % ความถูกต้อง |
|---------------------|----------------------------|---------------|----------------------------------|---------------|
| 8                   | ABCDEFGH                   | 100.0         | ABGDEFGH                         | 87.5          |
| 16                  | ABCDEFGH                   | 100.0         | ABCDEFGH                         | 100.0         |
| 32                  | ABCDEFGH                   | 100.0         | ABCDEFGH                         | 100.0         |
| 64                  | ABCDEFGH                   | 100.0         | KD BH                            | 37.5          |
| 128                 | ABCDEFGH                   | 100.0         |                                  | 0.00          |

ตารางที่ 6 ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปรคุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 20

| ค่าตัวแปร<br>คุณภาพ | ผลการอ่าน<br>อักขระด้วยแสง | % ความถูกต้อง | ผลการอ่าน<br>อักขระด้วยแสง<br>(MPEG) | % ความถูกต้อง |
|---------------------|----------------------------|---------------|--------------------------------------|---------------|
| 8                   | ABCDEFGH                   | 100.0         | ABGDEFGH                             | 87.5          |
| 16                  | ABCDEFGH                   | 100.0         | B DEF                                | 50.0          |
| 32                  | ABCDEFGH                   | 100.0         | AioDEF%k                             | 50.0          |
| 64                  | ABCDEFGH                   | 100.0         | i qp FG.H                            | 37.5          |
| 128                 | ABGDEFGH                   | 87.5          |                                      | 0.0           |

ตารางที่ 7 ตารางแสดงการเปรียบเทียบผลการอ่านอักขระด้วยแสงแยกตามค่าตัวแปรคุณภาพต่าง ๆ ที่ขนาดตัวอักษรเท่ากับ 18

| ค่าตัวแปร<br>คุณภาพ | ผลการอ่าน<br>อักขระด้วยแสง | % ความถูกต้อง | ผลการอ่าน<br>อักขระด้วยแสง<br>(MPEG) | % ความถูกต้อง |
|---------------------|----------------------------|---------------|--------------------------------------|---------------|
| 8                   | CDEFOH                     | 62.5          | In                                   | 0.0           |
| 16                  | A CDEFGH                   | 87.5          | Mill                                 | 0.0           |
| 32                  | A CDEFOH                   | 75.0          | Ed.A .ej                             | 25.0          |
| 64                  | k CDEFOH                   | 62.5          | l.The                                | 0.0           |
| 128                 | ASCDEFOH                   | 75.0          | Hi                                   | 12.5          |

จากสามตารางข้างต้นเมื่อนำผลการอ่านอักขระด้วยแสงมาทำการเปรียบเทียบภาพที่เกิดจากการบีบอัดวิธีที่ 1 ของการวิจัยในขนาดตัวอักษรขนาด 24, 20 และ 18 จะพบว่า ในขนาดตัวอักษรเป็น 24 ผลการอ่านจะมีร้อยละความถูกต้องเป็น 100 ในทุก ๆ ค่าตัวแปรคุณภาพ และเมื่อขนาดตัวอักษรเล็กลงเป็น 20 ที่ค่าตัวแปรคุณภาพ 128 ผลการอ่านอักขระด้วยแสง อ่านค่าผิดไปแค่ 1 ตำแหน่ง จึงทำให้ค่าร้อยละความถูกต้องเป็น 87.5 ในขณะที่ค่าตัวแปรคุณภาพที่มีค่าต่ำกว่านั้น ผลร้อยละความถูกต้องยังเป็น 100 และเมื่อปรับขนาดตัวอักษรเป็น 18 ค่าร้อยละความถูกต้องจะอ่านค่าผิดพลาดมากขึ้น แต่ค่าที่ได้ยังสูงกว่าร้อยละ 50

ในส่วนของผลบีบอัดอักขระด้วยแสงของภาพที่เกิดจากการบีบอัดด้วยวิธี MPEG เมื่อค่าตัวแปรคุณภาพมีค่ามากขึ้นผลการอ่านอักขระก็จะลดลง และยังตัวอักษรมีขนาดเล็กลง ก็ยังทำให้ผลการอ่านอักขระด้วยแสงในแต่ละค่าตัวแปรคุณภาพต่าง ๆ ลดลงไปด้วย

## วิจารณ์

จากผลการทดลอง เมื่อลองทำการเปรียบเทียบการบีบอัดตามแบบมาตรฐาน MPEG และการบีบอัดตามผลของการวิจัย สามารถแบ่งการเปรียบเทียบออกเป็นดังนี้

**อัตราการบีบอัด** เมื่อทำการปรับค่าตัวแปรคุณภาพ ในขั้นตอนการวิเคราะห์เชิงปริมาณ ผลการทดลองแสดงให้เห็นว่า เมื่อมีการปรับค่าตัวแปรคุณภาพ ให้มากขึ้นอัตราการบีบอัดจะมีค่าสูงกว่า อัตราการบีบอัดของค่าตัวแปรคุณภาพที่ต่ำกว่า นั่นคือ อัตราการบีบอัดแปรผันตามค่าตัวแปรคุณภาพที่ปรับ แต่อย่างไรก็ดีในค่าตัวแปรคุณภาพที่เท่ากันอัตราการบีบอัดตามแบบมาตรฐาน MPEG จะมีค่าสูงกว่าผลจากการวิจัย เนื่องจาก ในขณะทำการบีบอัดตามมาตรฐาน MPEG จะใช้การบีบอัดทั้งภาพแต่งงานวิจัยชิ้นนี้ ส่วนที่เป็นพื้นหน้าจะไม่ถูกบีบอัด จะทำการบีบอัดภาพที่เป็นพื้นหลัง จึงทำให้ อัตราการบีบอัดน้อยกว่านั่นเอง

**ความละเอียด** ผลจากการปรับค่าตัวแปรคุณภาพในขั้นตอนการวิเคราะห์เชิงปริมาณ เมื่อปรับค่าตัวแปรคุณภาพให้มีความสูงขึ้น ภาพจากการบีบอัดตามมาตรฐาน MPEG จะมีความละเอียดลดลง นั่นคือ ความละเอียดของภาพจะแปรผกผันกับค่าตัวแปรคุณภาพซึ่งต่างกับความละเอียดของผลจากการวิจัย ภาพที่เป็นพื้นหลังเท่านั้นที่มีความละเอียดลดลงเมื่อปรับค่าตัวแปรคุณภาพให้สูงขึ้น ในขณะที่ภาพพื้นหน้ายังคงซึ่งความละเอียดเท่าเดิม ไม่ว่าจะปรับค่าตัวแปรคุณภาพให้สูงขึ้นหรือลดลงก็ตาม

จากผลการเปรียบเทียบข้างต้น จะเห็นว่าวัตถุประสงค์หลักของของงานวิจัยชิ้นนี้ เพื่อพัฒนาขั้นตอนวิธีการบีบอัดที่เหมาะสมกับการชี้เป้าหมายทางยุทธวิธีที่ถ่ายด้วยกล้องตรวจจับความร้อนนั้น ความละเอียดของวัตถุที่เป็นเป้าหมายจะไม่ถูกบีบอัดให้สูญเสียความละเอียดของภาพไป ส่วนวัตถุอื่น ๆ จะบีบอัดเพื่อให้ขนาดข้อมูลน้อยลงอันเป็นประโยชน์ในการส่งข้อมูลจากอากาศยานสู่สถานีภาคพื้น ซึ่งการบีบอัดตามมาตรฐานของ MPEG ยังไม่สามารถช่วยแก้ไขปัญหานี้ได้ถึงแม้ว่าจะมีอัตราการบีบอัดของข้อมูลที่สูงกว่าอัตราการบีบอัดจากผลการวิจัยก็ตาม

## สรุปและข้อเสนอแนะ

### สรุป

งานวิจัยชิ้นนี้มีวัตถุประสงค์หลักคือการพัฒนาขั้นตอนวิธีการการบีบอัดที่เหมาะสมกับการใช้เป้าหมายทางยุทธวิธีสำหรับการถ่ายด้วยกล้องตรวจจับความร้อน นั่นคือการบีบอัดจะต้องคงไว้ซึ่งคุณภาพของวัตถุที่เป็นเป้าหมาย ส่วนวัตถุอื่น ๆ จะทำการบีบอัดตามมาตรฐาน MPEG โดยสาเหตุที่ไม่ใช่การบีบอัดตามมาตรฐานของ MPEG ทำการบีบอัดทั้งหมดเลยนั้น เนื่องจากภาพที่เกิดจากการบีบอัดตามมาตรฐานของ MPEG จะบีบอัดภาพที่เคลื่อนไหวซึ่งปกติ จะเป็นเป้าหมายของวัตถุที่ถ่ายด้วยกล้องตรวจจับความร้อนด้วยความละเอียดที่สายตาของมนุษย์รับได้ แต่เมื่อทำการขยายเข้าไปในเป้าหมายนั้น ๆ ภาพที่ได้จะแตก คุณภาพของภาพลดลง งานวิจัยชิ้นนี้จึงพัฒนาขั้นตอนวิธีการเพื่อแก้ไขปัญหาของการบีบอัดตามมาตรฐาน MPEG ข้างต้น

ในขั้นตอนของการพัฒนานั้นจะแบ่งออกเป็นสามขั้นตอนคือ การแยกวัตถุ การบีบอัด และการรวมวัตถุ ดังนี้

**การแยกวัตถุ** เป็นขั้นตอนการแยกเป้าหมายออกจากวัตถุอื่น ๆ โดยงานวิจัยชิ้นนี้จะทำการแยกวัตถุด้วยกันสองวิธีคือ การวิเคราะห์จากการเคลื่อนที่ของภาพ และการวิเคราะห์ตามเฉดสีซึ่งสามารถอ้างอิงกับอุณหภูมิความร้อนที่ถ่ายจากกล้องตรวจจับความร้อน โดยหลังจากที่แยกวัตถุแล้วจะแบ่งวัตถุที่เป็นเป้าหมายให้เป็นวัตถุพื้นหน้า ส่วนวัตถุอื่น ๆ ให้เป็นวัตถุพื้นหลัง

**การบีบอัด** เป็นขั้นตอนที่นำเอาวัตถุพื้นหลังที่ได้จากขั้นตอนการแยกวัตถุมาทำการเข้ารหัส โดยมีลำดับของขั้นตอนเข้ารหัสคือ ประเมินและชดเชยการเคลื่อนที่ของวัตถุแล้วทำการแปลงโคไซน์ไม่ต่อเนื่องเพื่อแปลงสัญญาณให้อยู่ในโดเมนความถี่ จากนั้นทำการวิเคราะห์เชิงปริมาณเพื่อลดจำนวนบิต แล้วเข้าสู่กระบวนการเข้ารหัสเอนโทรปี ซึ่งเป็นขั้นตอนสุดท้าย โดยภายในจะมีขั้นตอนการสแกนแบบซิกแซก เพื่อแปลงจากมิติ  $8 \times 8$  เป็น  $64 \times 1$  เพื่อนำไปเข้ารหัสส่วนต่าง เข้ารหัสรันเลนท์ และเข้ารหัสฮัฟแมนต่อไป

**การรวมวัตถุ** หลังจากนำเอาวัตถุพื้นหลังเข้ากระบวนการเข้ารหัสแล้ว นำวัตถุพื้นหน้าเข้าไปรวมเพื่อทำให้ภาพมีความสมบูรณ์และคงไว้ซึ่งความละเอียดของวัตถุพื้นหน้าไว้

เมื่อนำผลการวิจัยไปปรับค่าตัวแปรคุณภาพของการวิเคราะห์เชิงปริมาณเพื่อเปรียบเทียบ อัตราการบีบอัดและความละเอียดของภาพวิดีโอที่ได้บีบอัดตามมาตรฐาน MPEG จะได้ว่า เมื่อมีการปรับค่าตัวแปรคุณภาพ อัตราการบีบอัดที่ได้ทั้งจากผลการวิจัย และมาตรฐาน MPEG จะแปรผันตามค่าตัวแปรคุณภาพ นั่นคือ ถ้าค่าตัวแปรคุณภาพ มีค่ามาก อัตราการบีบอัดจะมากด้วย ซึ่งต่างจากความละเอียดของภาพ คือ ภาพตามมาตรฐาน MPEG จะมีความละเอียดที่แปรผกผันกับค่าตัวแปรคุณภาพ นั่นคือ ยิ่งค่าตัวแปรคุณภาพ มีค่าสูง ความละเอียดของภาพก็จะน้อย ซึ่งต่างจากผลที่ได้จากงานวิจัย พบว่า ค่าตัวแปรคุณภาพ จะมีผลที่แปรผกผันกับวัตถุพื้นหลังเท่านั้น ส่วนวัตถุพื้นหน้า ค่าตัวแปรคุณภาพ ไม่มีผลใด ๆ เลย

### ข้อเสนอแนะ

งานวิจัยชิ้นนี้ยังมีส่วนที่ต้องปรับปรุงแก้ไขเพิ่มเติม เพื่อเพิ่มคุณภาพของผลลัพธ์ได้ดังนี้

1. พัฒนาแล้วนำลงไปติดตั้งกับ DSP Board เพื่อนำไปใช้งานได้จริง
2. พัฒนาขั้นตอนวิธีการสำหรับตรวจสอบการเคลื่อนที่ของกล้องตรวจจับความร้อน เพื่อนำมารวมกับการวิเคราะห์การเคลื่อนที่ของวัตถุ
3. ในส่วนของการแยกวัตถุสามารถแยกวัตถุเป็นรูปทรงอื่น ๆ เช่น สี่เหลี่ยม วงรี วงกลม ฯลฯ ได้นอกเหนือจากรูปที่เป็นของเขตของวัตถุเป้าหมาย

ผลจากงานวิจัยชิ้นนี้สามารถนำไปประยุกต์กับการใช้ประโยชน์ในเรื่องความมั่นคงของประเทศไทย และความสงบเรียบร้อยด้านอื่น ๆ ได้ด้วยเช่น การนำไปติดตั้งเพื่อตรวจจับความเร็วรถติดตั้งตามร้านทอง ติดตั้งตามสถานที่เสี่ยงภัยต่าง ๆ

## เอกสารและสิ่งอ้างอิง

ลัญญกร วุฒิสิทธิกุลกิจ, สุวิทย์ นาคพิรุณ, ปิณฑิร สุทธาโรจน์ และ สมภพ โชคชัยธรรม.

2549. เทคโนโลยีการบีบอัดข้อมูลเบื้องต้น. บริษัทด้านสุทธาการพิมพ์จำกัด, กรุงเทพฯ.

บวรชัย. 2549. **FLIR SYSTEM**. กรมการลาดตระเวนทางอากาศ. แหล่งที่มา:

<http://www.aerial.rtaf.mi.th>, 19 เมษายน 2550.

Chen, J. Ut.Va Koc and K.J. Ray Liu. 2002. **Design od Digital Video Coding System**

**A Complete Compressed Domain Approach**. 10<sup>ed</sup>. Marcel Dekker, New York.

Effelsberg, W. and R. Steinmetz. 1998. **Video Compression Techniques**. dpunk.verlag, Heidelberg.

Gibson, J.D., T. Berger, T. Lookabaugh, D. Lindbergh and R.L.Baker. 1998. **Digital Compression for Multimrdia Principle and Standards**. Morgan Kaufmann Publisher, Inc., California.

Joseph Kee-Yin, NG and C. Kin Cheung Hui. 2004. Implementing a QoS-Aware MPEG-4 Video System for Embedded Computing. **Journal of Embedded Computing** 7 (1):

Kadono, S. Choong, S.B. and M. Etoh. 1998. **Motion Compensation Method for Moving Pictures with Binary Shape**. Matsushita Electric Industrial Co., Ltd., Osaka.

MPEG-4 Industry Forum. 2002. **MPEG-4 – The Media Standard**. Available Source: <http://www.m4if.org>, March 27, 2007.

Neri, A., S. Colonnese, G. Russo and P. Talone. 1998. Automatic moving object and background separation. **Signal Processing** 66 (2): 219 - 232.

Nunes, P., P. Correia and F. Pereira. n.d. **Coding Video Objects with the Emerging MPEG-4 Standard.** Paris.

Sikora, T. 1997. The MPEG-4 video standard verification model. **Circuits and Systems for Video Technology, IEEE Transactions** 7 (1): 19 - 31.

\_\_\_\_\_. and L.Chiariglione. 1997. MPEG-4 video and its potential for future multimedia services. **Proceedings of 1997 IEEE International Symposium on Circuits and Systems** 2: 1468-1471.

SieBot. 2007. **OCR.** OCR – History – Wikipedia. Available Source: <http://th.wikipedia.org/wiki/>, April 30, 2007.

The MathWork's Team. 1994-2007. **Video and Image Processing Blockset 2.3.**

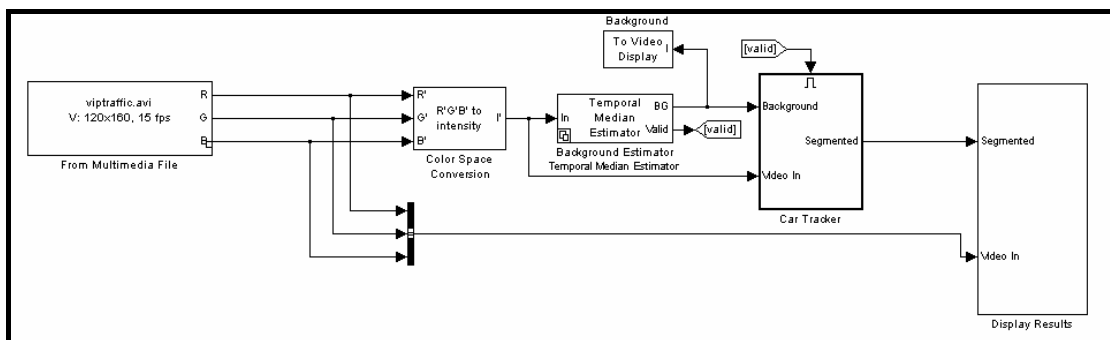
The MathWorks. Available Source: <http://www.mathworks.com/>, March 10, 2007.

T., Ebrahimi. 1997. MPEG-4 video verification model: A video encoding/decoding algorithm based on content representation. **Image Communication** 9 (1): 367-384.

ภาคผนวก

## Programming

โปรแกรม Matlab ของการทดลองที่ 1 การบีบอัดภาพวิดีโอด้วยการแบ่งแยกวัตถุด้วยการวิเคราะห์การเคลื่อนที่



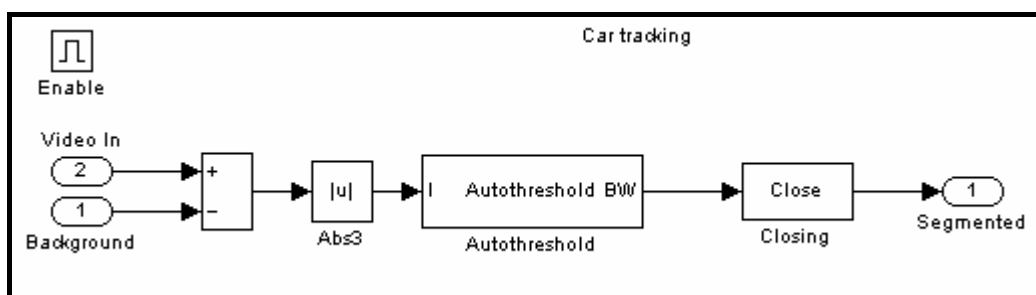
### ภาพผนวกที่ 1 โปรแกรมหลัก (Main)

*Color Space Conversion:* แปลง Color Space จาก RGB เป็น Intensity

*Background Estimator Temporal Median Estimator:* เป็น Block แสดงค่ากลางของข้อมูลวิดีโอทุก ๆ เวลา โดยในการทดลองนี้จะใช้วัด 2 Frame ต่อการคำนวณหนึ่งครั้ง โดย Output ที่ออกมาจะเป็นภาพ Background ของวิดีโอ

*Car Tracker:* เป็น Block ทำการหา Segment ระหว่างพื้นหน้าและพื้นหลัง

*Display Results:* เป็น Block ที่ทำการบีบอัดและแสดงผลการบีบอัด



### ภาพผนวกที่ 2 โมดูลการตรวจจับ (Car tracking)

*Video In:* เป็นภาพวิดีโอต้นเดิม

*Background:* เป็นภาพวิดีโอที่แสดงภาพพื้นหลัง

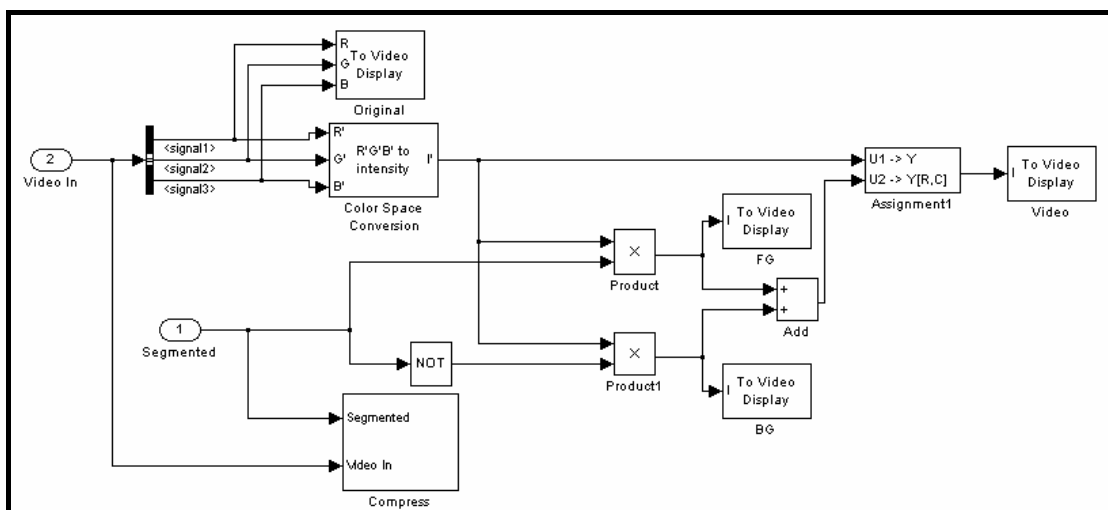
*Sum:* เป็น Block สำหรับเพิ่มหรือลด Input ในที่นี้คือการเพิ่ม Video In และลด Background

*Abs:* เป็น Block เพื่อหาค่าสมบูรณ์ของค่า Input

*Autothreshold:* เป็น Block แปลงจากรูปภาพที่เป็น Intensity อยู่ในรูป Binary โดยใช้วิธีของ Otsu ซึ่งกำหนดค่าเริ่มต้นโดยการแยก Histogram ของภาพ Input นั่นคือแปลงแต่ละกลุ่ม Pixel ให้เป็นค่าที่น้อย

*Closing:* ทำการปิด Intensity หรือ Binary

*Segment:* เป็น Output ที่มีค่าเป็นเมตริก ของ 0 และ 1



### ภาพผนวกที่ 3 แสดงผล (Display Results)

*Segment:* เป็นข้อมูล Input ที่มีค่าเป็นเมตริก ของ 0 และ 1

*Video In:* เป็นข้อมูลของวิดีโอต้นเดิม

*Compress:* เป็น Block สำหรับทำการบีบอัดข้อมูล

*Color Space Conversion:* เป็น Block สำหรับแปลง Color Space ในที่นี้ทำการแปลงจาก RGB เป็น Intensity

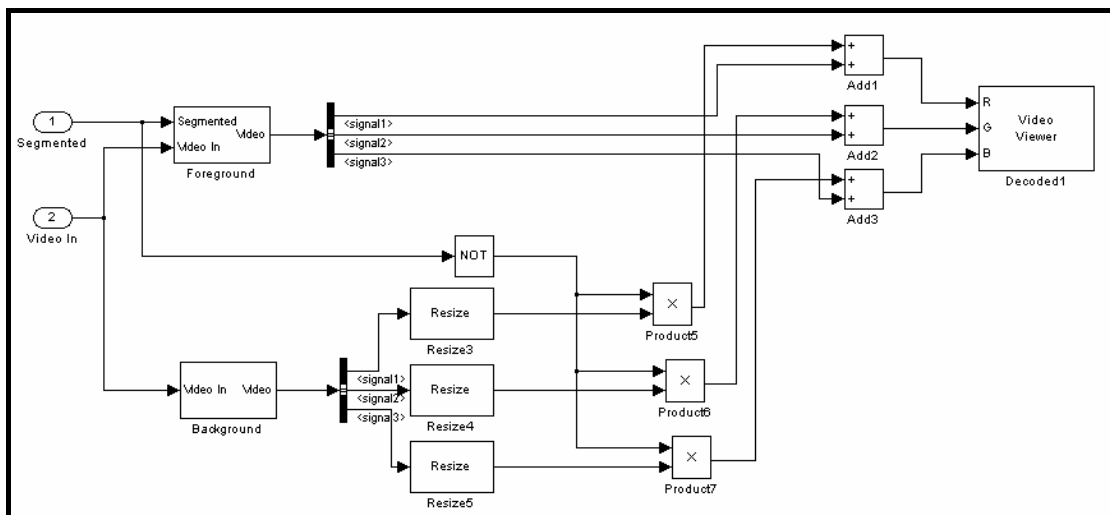
*Logical Operator:* เป็น Block ของการเปรียบเทียบเชิงตรรกศาสตร์ในที่นี้คือ Not นั่นคือการทำนิเสธ

*Product:* เป็น Block สำหรับทำการคูณหรือการหารสำหรับ Input ที่ป้อนเข้ามาสองค่า ในที่นี้จะเป็นการคูณ

*Sum*: เป็น Block ที่ทำการเพิ่มหรือลด Input ที่เข้ามา ในที่นี้ทำการเพิ่ม

*Assignment*: เป็น Block สำหรับกนดการแสดงผล

*Video*: เป็น Block สำหรับแสดง Output



#### ภาพผนวกที่ 4 การบีบอัด (Compress)

*Segment*: เป็นข้อมูล Input ที่มีค่าเป็นเมตริก ของ 0 และ 1

*Video In*: เป็นข้อมูลของวิดีโอที่เริ่มต้น

*Foreground*: เป็น Block สำหรับจัดการกับภาพพื้นหน้า

*Background*: เป็น Block สำหรับทำการบีบอัดภาพพื้นหลัง

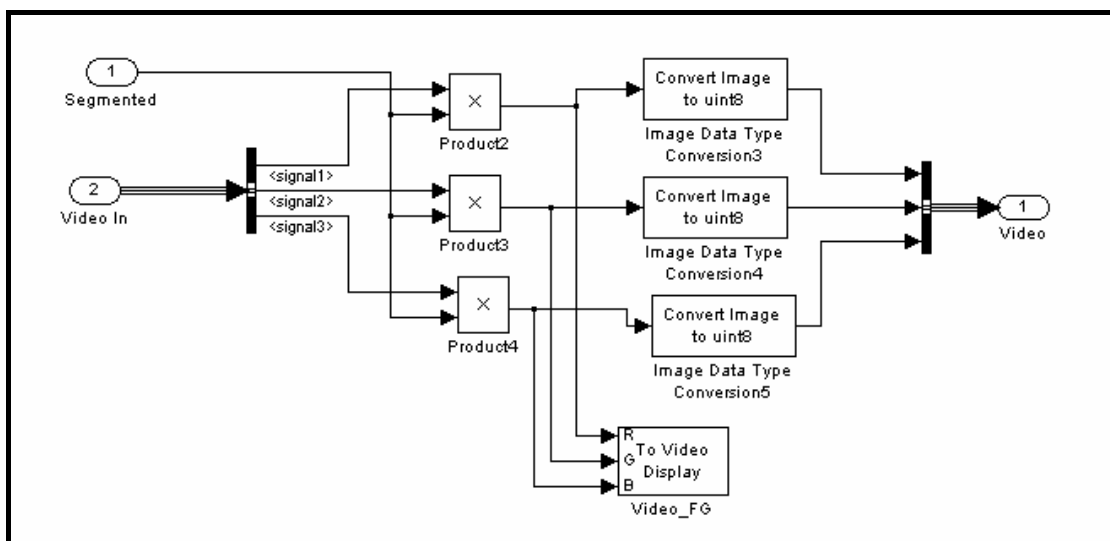
*Resize*: เป็น Block สำหรับปรับขนาดของภาพ

*Logic Operator*: เป็น Block ของการเปรียบเทียบเชิงตรรกศาสตร์ในที่นี้คือ Not นั่นคือการทำนิเสธ

*Product*: เป็น Block สำหรับทำการคูณหรือการหารสำหรับ Input ที่ป้อนเข้ามาสองค่า ในที่นี้จะเป็นการคูณ

*Sum*: เป็น Block ที่ทำการเพิ่มหรือลด Input ที่เข้ามา ในที่นี้ทำการเพิ่ม

*Video*: เป็น Block สำหรับแสดง Output



### ภาพผนวกที่ 5 พื้นหน้า (Foreground)

*Segment:* เป็นข้อมูล Input ที่มีค่าเป็นเมตริก ของ 0 และ 1

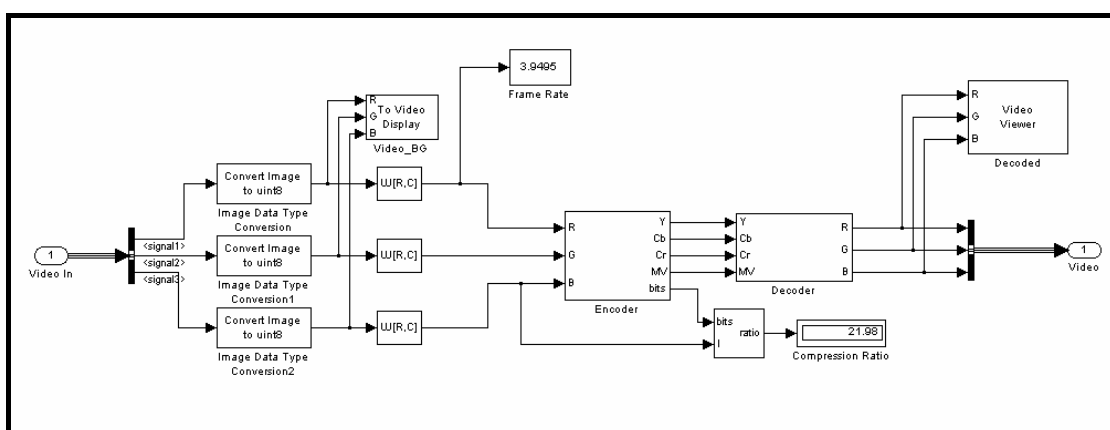
*Video In:* เป็นข้อมูลของวิดีโอที่เริ่มต้น

*Product:* เป็น Block สำหรับทำการคูณหรือการหารสำหรับ Input ที่ป้อนเข้ามาสองค่า ในที่นี้จะเป็นการคูณ

*Image Data Type Conversion:* เป็น Block สำหรับเปลี่ยนแปลงชนิดข้อมูล

*Video:* เป็นข้อมูล Output ของภาพวิดีโอ

*Video FG:* เป็น Block สำหรับแสดง Output



### ภาพผนวกที่ 6 พื้นหลัง (Background)

*Video In:* เป็นข้อมูลของวิดีโอที่เริ่มต้น

*Image Data Type Conversion:* เป็น Block สำหรับเปลี่ยนแปลงชนิดข้อมูล

*Selector:* เป็น Block ที่ระบุ element ของ Vector หรือ Matrix ที่เข้ามา

*Encoder:* เป็น Block สำหรับทำการบีบอัดข้อมูล

*Frame Rate:* เป็น Block แสดงผลการคำนวณอัตราของ Frame ในสัญญาณนำเข้า ในที่นี้จะทำการคำนวณทุก ๆ 10 frame

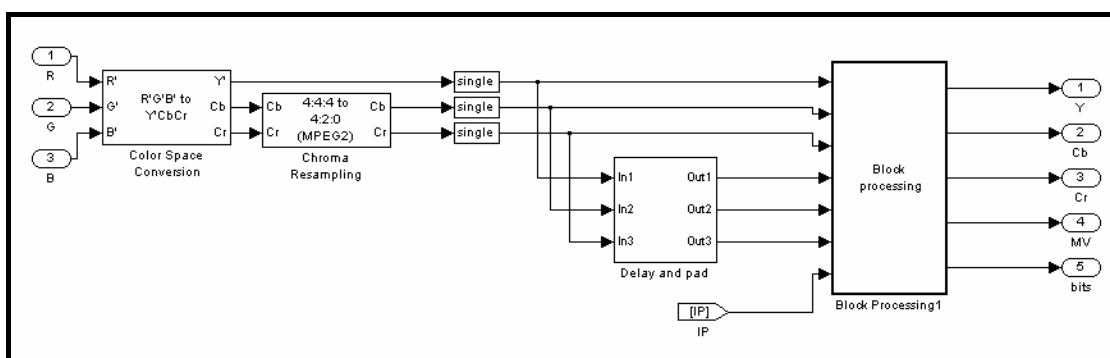
*Bit Rate:* เป็น Block สำหรับคำนวณหาอัตราการบีบอัด

*Compression Ratio:* เป็น Block สำหรับแสดงผลอัตราการบีบอัด

*Decoder:* เป็น Block สำหรับการแยกข้อมูล

*Decoded:* เป็น Block สำหรับแสดงผลการแยกข้อมูล

*Video:* เป็น Video Output



## ภาพผนวกที่ 7 การเข้ารหัส (Encoder)

*R:* เป็นค่า Input ของสีแดง

*G:* เป็นค่า Input ของสีเขียว

*B:* เป็นค่า Input ของสีน้ำเงิน

*Color Space Conversion:* แปลง Color Space จาก RGB เป็น YCbCr

*Chroma Resampling:* เปลี่ยนอัตราส่วนของ Y:Cb:Cr จาก 4:4:4 เป็น 4:2:0

*Signal:* เป็น Block สำหรับเปลี่ยนข้อมูลให้อยู่ในรูปสัญญาณ

*Delay and Pad:* เป็น Block สำหรับหน่วงเวลา

*Block Processing:* เป็น Block สำหรับทำการบีบอัดข้อมูล

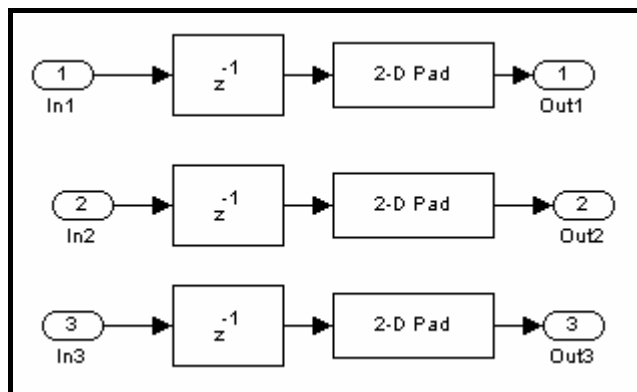
*Y:* ค่าของความสว่าง ที่อยู่ในรูป YCbCr Color Space

*Cb:* ค่าของความต่างของสีน้ำเงิน ที่อยู่ในรูป YCbCr Color Space

*Cr:* ค่าของความต่างของสีแดง ที่อยู่ในรูป YCbCr Color Space

*MV*: คืค่า Motion Vector

*Bits*: จำนวน bit ที่ได้



ภาพผนวกที่ 8 การหน่วงเวลา (Delay and Pad)

*In1*: ค่า Input ตัวที่ 1

*In2*: ค่า Input ตัวที่ 2

*In3*: ค่า Input ตัวที่ 3

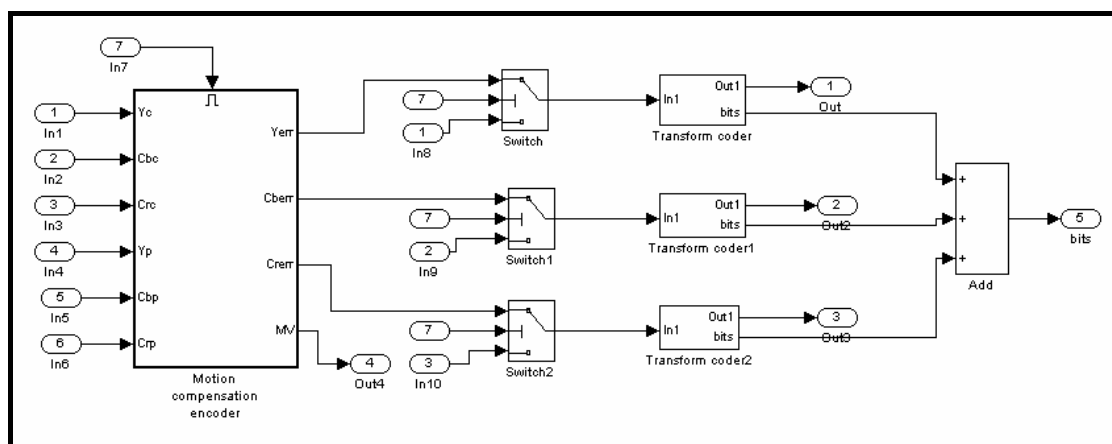
Delay ( $z^{-1}$ ): เป็น Block สำหรับทำการหน่วงเวลาในช่วงเวลาไม่ต่อเนื่อง

2-D Pad: ทำเมตริกให้อยู่ในรูป 2 dimensions

*Out1*: ค่า Output ตัวที่ 1

*Out2*: ค่า Output ตัวที่ 2

*Out3*: ค่า Output ตัวที่ 3



ภาพผนวกที่ 9 ประมวลผลบล็อก (Block Processing)

- 1: เป็นค่า Y ที่อยู่ในรูปสัญญาณ
- 2: เป็นค่า Cb ที่อยู่ในรูปสัญญาณ
- 3: เป็นค่า Cr ที่อยู่ในรูปสัญญาณ
- 4: เป็นค่า Y ที่อยู่ในรูปสัญญาณที่ถูกหน่วงเวลา
- 5: เป็นค่า Cb ที่อยู่ในรูปสัญญาณที่ถูกหน่วงเวลา
- 6: เป็นค่า Cr ที่อยู่ในรูปสัญญาณที่ถูกหน่วงเวลา
- 7: เป็นค่า IP

*Motion Estimation Encoder*: เป็น Block ของการเข้ารหัสการประเมินการเคลื่อนที่

*Switch*: เป็น Block สำหรับทำ Operation Switch

*Transform Coder*: ตัวแปลง

*Out1*: Output ตัวที่ 1

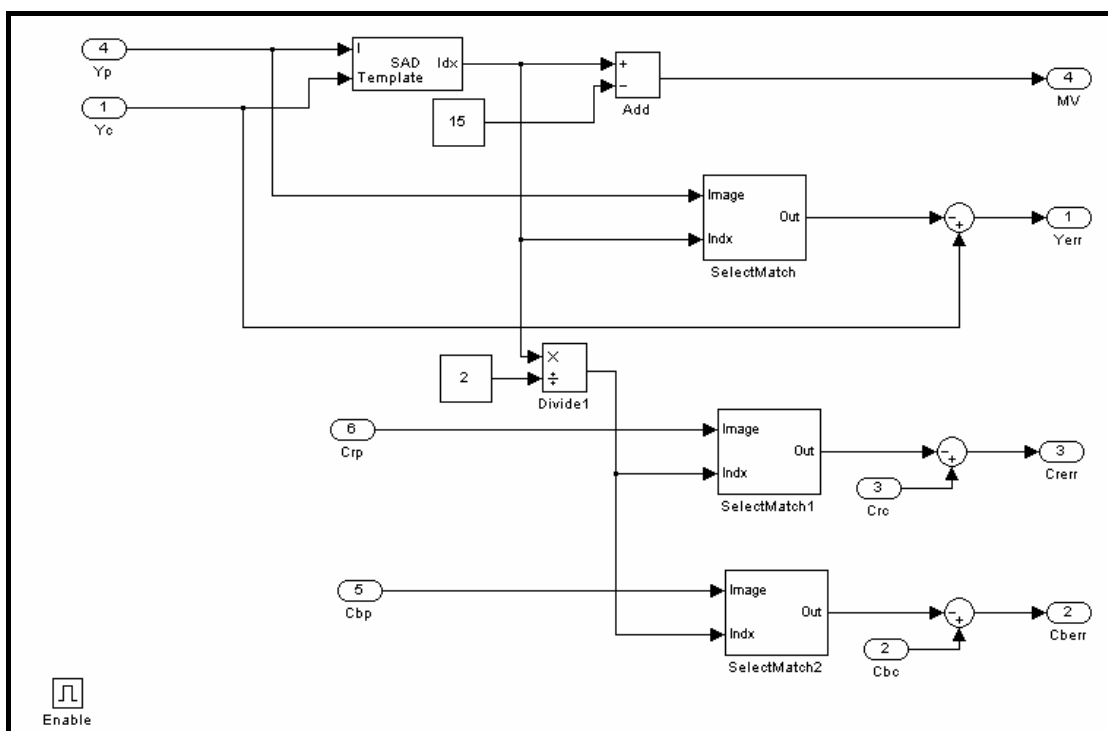
*Out2*: Output ตัวที่ 2

*Out3*: Output ตัวที่ 3

*Out4*: Output ตัวที่ 4

*Sum*: เป็น Block ที่ทำการเพิ่มหรือลด Input ที่เข้ามา ในที่นี้ทำการเพิ่ม

*Bits*: จำนวน bit ที่ได้



ภาพผนวกที่ 10 การเข้ารหัสการชดเชยการเคลื่อนที่ (Motion Compensation Encoder)

*Yp*: เป็นค่า *Y* ที่อยู่ในรูปสัญญาณที่ถูกหน่วงเวลา

*Yc*: เป็นค่า *Y* ที่อยู่ในรูปสัญญาณ

*Crp*: เป็นค่า *Cr* ที่อยู่ในรูปสัญญาณที่ถูกหน่วงสัญญาณ

*Cbp*: เป็นค่า *Cb* ที่อยู่ในรูปสัญญาณที่ถูกหน่วงเวลา

*SAD*: เป็น Block สำหรับคำนวณหาผลรวมของความต่างสมบูรณ์

15: กำหนดค่าคงที่เพื่อการคำนวณไว้เป็นค่า 15

2: กำหนดค่าคงที่เพื่อการคำนวณไว้เป็นค่า 2

*Divide*: เป็น Block ของ Product สำหรับทำการคูณและการหาร

*Add*: เป็น Block ของ Sum สำหรับเพิ่มหรือลดค่า Input

*SelectMatch*: เป็น Block สำหรับเลือกตำแหน่งที่ตรงกัน

*Crc*: เป็นค่า *Cr* ที่อยู่ในรูปสัญญาณ

*Cbc*: เป็นค่า *Cb* ที่อยู่ในรูปสัญญาณ

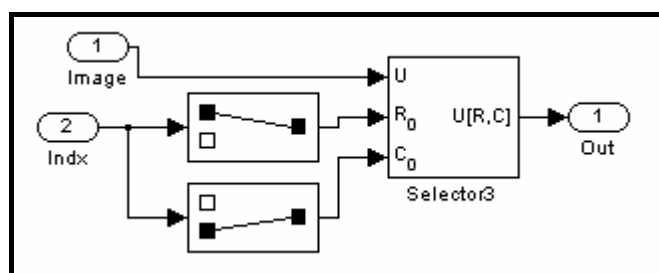
*MV*: เป็นค่า Motion Vector

*Yerr*: เป็นค่า *Y* ที่เปลี่ยนไป

*Crerr*: เป็นค่า *Cr* ที่เปลี่ยนไป

*Cberr*: เป็นค่า *Cb* ที่เปลี่ยนไป

*Enable*: เป็น port สำหรับรับสถานะเพื่อควบคุมการทำงาน



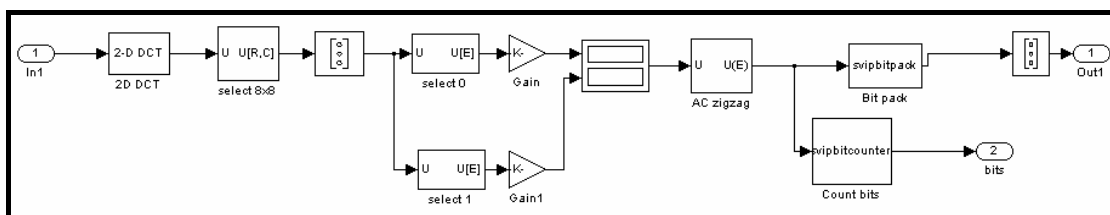
ภาพผนวกที่ 11 เลือกการเข้าคู่ (Select Match)

*Image*: ค่า Input ของภาพเพื่อคำนวณ

*Indx*: ค่าที่ผ่านการคำนวณ SAD

*Selector*: เป็น Block สำหรับจัดการเรื่องการเลือก

*Out*: ผลลัพธ์ที่ได้



ภาพผนวกที่ 12 ตัวแปลงรหัส (Transform Coder)

*In1*: ค่า input สำหรับการคำนวณ

*2D DCT*: เป็นการคำนวณหา Discrete Cosine Transform จากเมตริก 2 Dimension

*Select 8x8*: เป็น Block Selector ให้เมตริกมีขนาดเป็น 8x8

*Reshape*: เปลี่ยน Dimension ให้เป็น 64x1

*Selector0*: เป็น Block Selector โดยกำหนดให้ค่า Element มีค่าเป็น 0

*Selector1*: เป็น Block Selector โดยกำหนดให้ค่า Element มีค่าเป็น 1

*Gain*: เป็น Block การกำหนดค่า Gain โดยกำหนดให้มีความเท่ากับ 1/16

*Concatenate*: การนำค่าสัญญาณของค่า Matrix หรือ Vector มาต่อกัน

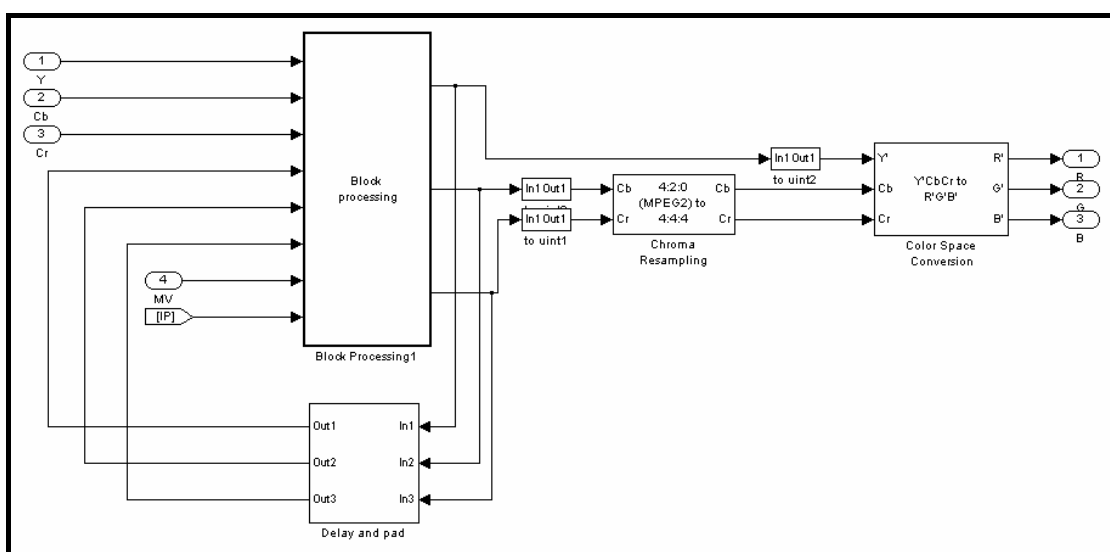
*AC Zigzag*: เป็น Block Selector สำหรับทำกระบวนการ Zigzag

*Count bits*: เป็น Block ของ S-Function สำหรับการหาจำนวน Bit

*Bit pack*: เป็น Block ที่ใช้การทำ run length coding

*Out1*: ผลจากการทำงาน

*Bit*: จำนวน bit



ภาพผนวกที่ 13 การถอดรหัส (Decoder)

*Y*: ค่า Y จากการ Encoder

*Cb*: ค่า Cb จากการ Encoder

*Cr*: ค่า Cr จากการ Encoder

*MV*: ค่า Motion Vector จากการ Encoder

*IP*: ค่า IP

*Block Processing*: เป็น Block สำหรับการถอดรหัสจากการบีบอัด

*Delay and Pad*: เป็น Block สำหรับหน่วงเวลา

*To Unit*: แปลงข้อมูลให้อยู่ในรูป Unit

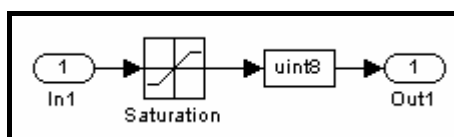
*Chroma Resampling*: เปลี่ยนอัตราส่วนของ Y:Cb:Cr จาก 4:2:0 เป็น 4:4:4

*Color Space Conversion*: แปลง Color Space จาก YCbCr เป็น RGB

*R*: ค่าของสีแดงจากการถอดรหัส

*G*: ค่าของสีเขียวจากการถอดรหัส

*B*: ค่าของสีน้ำเงินจากการถอดรหัส



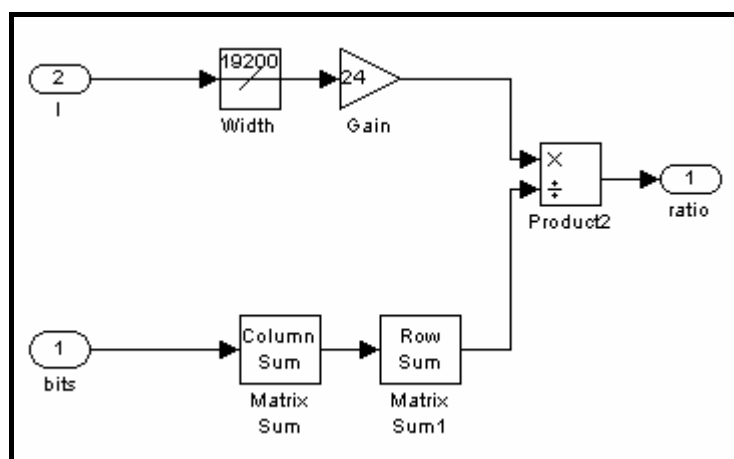
ภาพผนวกที่ 14 เปลี่ยนหน่วย (To Unit)

*In1*: ค่า Input

*Saturation*: เป็น Block สำหรับกำหนดค่าขอบเขตสูงสุดและขอบเขตต่ำสุดในที่นี้ กำหนดให้เป็น 0 ถึง 255 ตามลำดับ

*Unit8*: เป็น Block ของ Data Type Conversion กำหนดให้แปลงชนิดข้อมูลเป็น Unit8

*Out1*: ค่า Output



ภาพผนวกที่ 15 อัตราบิต (Bit Rate)

*I*: ค่า Input ของสีน้ำเงิน

*Bits*: จำนวน bit ที่ได้

*Width*: เป็นการหาความกว้างของชนิดข้อมูลสัญญาณ Input

*Gain*: ค่า Gain กำหนดให้มีค่าเป็น 24

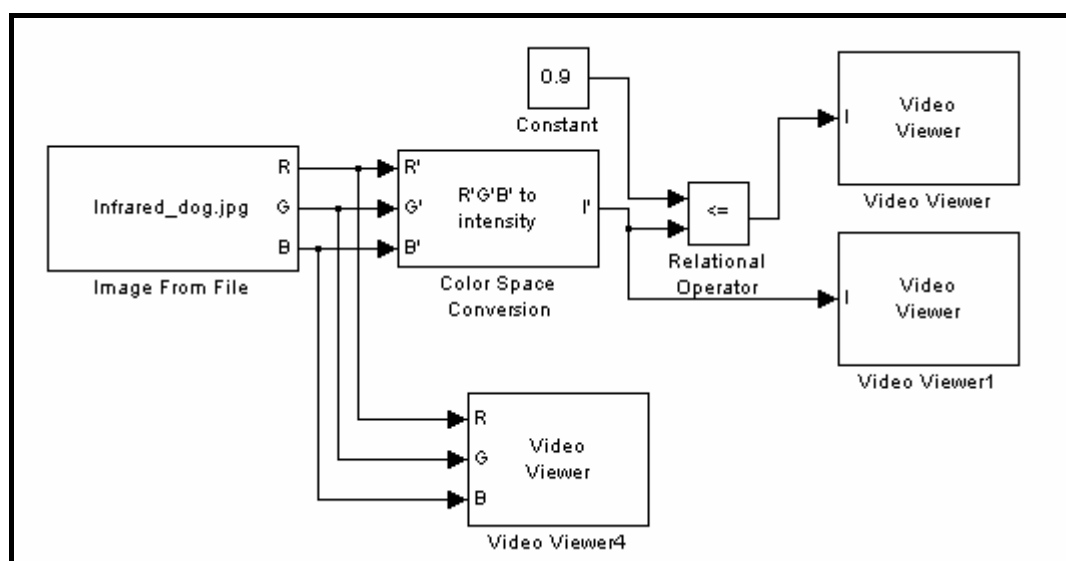
*Column Sum*: เป็น Block ของ Matrix Sum เพื่อหาผลรวมใน Column

*Row Sum*: เป็น Block ของ Matrix Sum เพื่อหาผลรวมใน Row

*Product*: เป็น Block สำหรับการคูณและหาร Input สองค่า

*Ratio*: อัตราส่วนที่ได้จากการ Product

โปรแกรม Matlab ของการทดลองที่ 2 การแบ่งแยกวัตถุด้วยวิเคราะห์ความเข้มของแสง



ภาพผนวกที่ 16 การแบ่งแยกภาพ

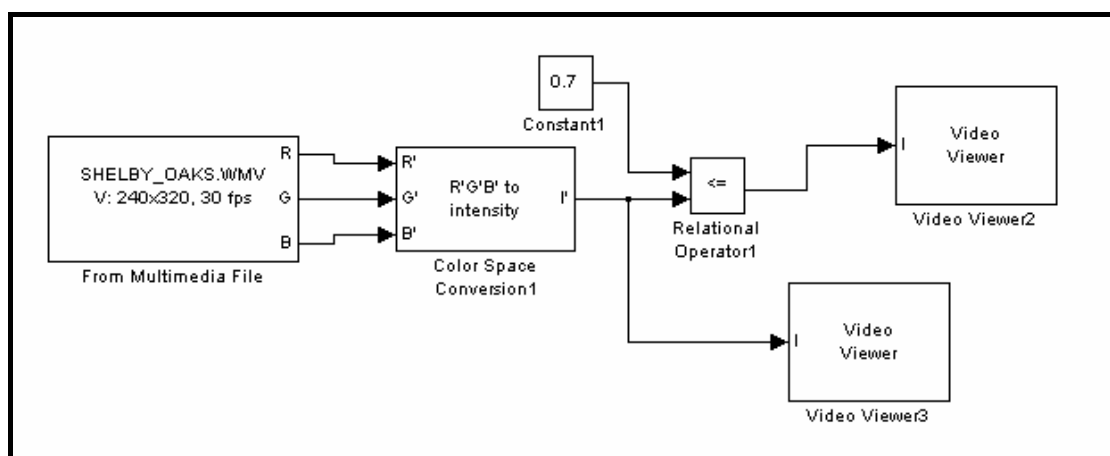
*Image from File:* เป็นภาพที่นำมาทดลอง

*Color Space Conversion:* แปลง Color Space จาก RGB เป็น Intensity

*Constant:* ค่าคงที่มีค่าเป็น 0.9

*Relation Operator:* เป็น Block สำหรับใช้ในกระบวนการทางตรรกศาสตร์ในที่นี้คือน้อยกว่าหรือเท่ากับ

*Video Viewer:* แสดงผลลัพธ์



ภาพผนวกที่ 17 การแบ่งแยกวิดีโอ

*From Multimedia File:* เป็นวีดิทัศน์สำหรับทดลอง

*Color Space Conversion:* แปลง Color Space จาก RGB เป็น Intensity

*Relation Operator:* เป็น Block สำหรับใช้ในกระบวนการทางตรรกศาสตร์ในที่นี้คือน้อยกว่าหรือเท่ากับ

*Constant:* ค่าคงที่มีค่าเป็น 0.7

*Video Viewer:* แสดงผลลัพธ์

## ประวัติการศึกษา และการทำงาน

|                                |                            |
|--------------------------------|----------------------------|
| ชื่อ –นามสกุล                  | นายสุชาติ ชุนทอง           |
| วัน เดือน ปี ที่เกิด           | 22 มีนาคม 2524             |
| สถานที่เกิด                    | จังหวัดพัทลุง              |
| ประวัติการศึกษา                | วท.บ.(วิทยาการคอมพิวเตอร์) |
| ตำแหน่งหน้าที่การงานปัจจุบัน   | Consultant                 |
| สถานที่ทำงานปัจจุบัน           | JSG Co.,Ltd                |
| ผลงานดีเด่นและรางวัลทางวิชาการ | -                          |
| ทุนการศึกษาที่ได้รับ           | -                          |