

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การนับแยกจำนวนรถยนต์โดยใช้การประมวลผลภาพ

VEHICLE CLASSIFICATION BY USING DIGITAL IMAGE PROCESSING



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
คณะวิศวกรรมศาสตร์ สาขาวิชาอิเล็กทรอนิกส์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**การนับแยกจำนวนรถยนต์โดยใช้การประมวลผลภาพ**  
**VEHICLE CLASSIFICATION BY USING DIGITAL IMAGE PROCESSING**



ปริญญาานิพนธ์สำหรับปริญญาวิศวกรรมศาสตรบัณฑิต  
สาขาวิชาอิเล็กทรอนิกส์  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2548

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญาานิพนธ์ ปีการศึกษา 2548

ภาควิชา อิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การนับแยกจำนวนรถยนต์โดยใช้การประมวลผลภาพ

ผู้จัดทำ

1. นายอำนาจ ปรารักษ์แสงวิไล รหัส 45010974
2. นายอุกฤษฏ์ กาศิ๊บ รหัส 45010978



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# การนับแยกจำนวนรถยนต์โดยใช้การประมวลผลภาพ

นายอำนาจ ปรามค์แสงวิไล รหัส 45010974

นายอุกฤษฏ์ กาศิ๊บ รหัส 45010978

ผศ.ดร. ชูชาติ ปิณฑวิรุจน์ (อาจารย์ที่ปรึกษา)

ภาคเรียนที่ 2 ปีการศึกษาที่ 2548

## บทคัดย่อ

โครงการที่นำเสนอนี้ เป็นการออกแบบ การนับแยกจำนวนรถยนต์ เพื่อใช้ในการควบคุม ปริมาณจำนวนรถยนต์ที่ขับผ่านแยกจราจร โดยการออกแบบจะสร้าง Graphic User Interface (GUI) สำหรับแสดงภาพที่ได้จากกล้องและทำการ Capture ภาพ Video พร้อมกับสร้าง Active Area และ Window ขึ้นมาใหม่ เพื่อแสดงค่าเฉลี่ยความเข้มของแสงที่ได้จากการลบกันของภาพ Background ของพื้นถนนกับภาพที่ต้องการประมวลผล จากนั้นจะทำการนับและแยกประเภทของยานพาหนะ และแสดงผลลัพธ์ออกมาด้วยกราฟเพื่อให้ง่ายต่อการพิจารณา

# VEHICLE CLASSIFICATION BY USING DIGITAL IMAGE PROCESSING

Mr.Amnarj Prangsangvilai ID.45010974

Mr.Augrit Katib ID.45010978

Assist.Prof.Dr. Chuchart Pintavirooj (Advisor)

2<sup>nd</sup> Semester, Educational Year 2005

## Abstract

This project presents vehicle classification by using digital image processing for controlling the quantity of vehicle which pass the traffic light. GUI is made to show and capture the vision from the video camera. This program can also re-doing the active area and new window for indicating the average of light density from road background deduct by image processing. After that the program will count and classification of the vehicle and show the graph for the easy reference.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา 01044057 Project1 จัดทำขึ้นเพื่อประกอบการศึกษาในการ ออกแบบระบบการนับแยกจำนวนรถยนต์โดยใช้การประมวลผลภาพ VEHICLE CLASSIFICATION BY USING DIGITAL IMAGE PROCESSING โดยการออกแบบจะนำภาพจากไฟล์วิดีโอเข้ามา ประมวลผลทีละ frame และนำข้อมูลที่ได้มาวิเคราะห์จำนวนและคัดแยกประเภทของยานพาหนะ กับความ หนาแน่นของการจราจร

ผู้จัดทำหวังเป็นอย่างยิ่งว่าโครงการการออกแบบระบบการนับแยกจำนวนรถยนต์โดยใช้ การประมวลผลภาพ VEHICLE CLASSIFICATION BY USING DIGITAL IMAGE PROCESSING และรายงานฉบับนี้ จะมีประโยชน์ต่อผู้ต้องการค้นคว้าเกี่ยวกับ “Digital Image Processing” เพื่อที่จะสามารถนำไปประยุกต์ใช้ในงานเกี่ยวกับจราจรได้

.....  
 (นายอำนาจ ปรารักษ์แสงวิไล)

.....  
 (นายอุกฤษฏ์ กาศดับ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
บทคัดย่อ	ก
Abstract	ข
กิตติกรรมประกาศ	ค
สารบัญ	ง
บทที่ 1 บทนำ	
1.1 ความเป็นมาของปริญญาานิพนธ์	1
1.2 จุดมุ่งหมายของปริญญาานิพนธ์ของภาคเรียนที่ 1	1
1.3 ขอบเขตของโครงการงาน	1
ประโยชน์และผลที่คาดว่าจะได้รับจากโครงการงาน	2
บทที่ 2 ทฤษฎี	
2.1 การได้มาซึ่งภาพ (Image Acquisition)	3
2.2 กระบวนการประมวลผลภาพ (Image Processing)	3
2.2.1 การแทนภาพด้วยข้อมูลแบบดิจิทัล	3
2.2.2 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิทัล	4
2.3 ไฟล์ข้อมูลภาพชนิดบิตแมป	5
2.3.1 รูปแบบของไฟล์ข้อมูลภาพชนิดบิตแมป	5
2.3.2 โครงสร้างของไฟล์ของข้อมูลภาพชนิดบิตแมป	5
2.4 การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป	6
2.5 ระดับสีเทา (Grey Scale)	6
2.6 การสร้างภาพไบนารี	7
2.6.1 ฮิสโตแกรม (Histogram)	9
2.6.2 การแปลงระดับเกรย์ (Grey-level Transformation)	9
2.7 การแยกวัตถุจากภาพ (Segmentation)	11
2.7.1 การแยกภาพด้วยการพิจารณาการต่อเนื่องของข้อมูล	11
2.7.2 การแยกภาพด้วยวิธี Region Labeling	11
2.8 กลุ่มสีที่ใช้แสดงผล	12
2.8.1 คุณสมบัติของกลุ่มสีที่แสดงผล	12
2.8.2 กลุ่มสีแสดงผลแบบ RGB	13
บทที่ 3 หลักการทำงานและการออกแบบ	
3.1 หลักการทำงานของโปรแกรม	18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบระบบ	19
3.2.1 โครงสร้างโดยรวมของโปรแกรม	19
3.2.2 Capture From File AVI	20
3.2.3 Load First Frame For Background	20
3.2.4 Select Active Area	20
3.2.5 Subtract each Active Area	20
3.2.6 Thresholding each Active Area	21
3.2.7 Find Average value of Summation Absolute for each Active Area	22
3.2.8 Process Information	23
3.2.9 Plot Graph	24
<b>บทที่ 4 ผลการทดลอง</b>	
4.1 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนซ้าย โดยรถ CAR	25
4.2 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนกลาง โดยรถ PICK UP	27
4.3 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนขวา โดยรถ MOTOCYCLE	29
4.4 การนับแยกจำนวนยานพาหนะในช่องถนนจำนวน 2 เลนพร้อมกัน โดยรถ CAR	30
4.5 การนับแยกจำนวนยานพาหนะในช่องถนนจำนวน 3 เลนพร้อมกัน โดยรถ CAR,VAN วิ่งในช่องทางซ้ายกับกลาง และ MOTOCYCLE ในช่องทางขวา	32
4.6 การนับแยกจำนวนยานพาหนะของรถบรรทุก (TRUCK)	33
4.7 ผลการทดลองที่เกิด Error ขึ้น	35
4.8 การหา % error ของโปรแกรม	37
4.9 การทดลองวัดความเร็วของยานพาหนะในช่องทางเลนกลาง	38
<b>บทที่ 5 สรุปผลและวิจารณ์</b>	
5.1 สรุปและวิจารณ์ผลการทดลอง	40
5.2 ข้อดีและข้อเสีย	40
5.3 การนำไปประยุกต์การใช้งาน	41
5.4 แนวทางการพัฒนา	41
<b>บรรณานุกรม</b>	42

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาของปัญญาประดิษฐ์

เนื่องจากในปัจจุบันปัญหาการจราจรติดขัดมีแนวโน้มที่จะทวีความรุนแรงเพิ่มมากขึ้นเรื่อยๆ โดยเฉพาะในเมืองใหญ่ๆ ซึ่งเกิดจากการเพิ่มขึ้นของประชากร กลุ่มผู้ดำเนินโครงการได้สังเกตเห็นถึงปัญหาข้อนี้จึงได้พยายามสร้างระบบตรวจสอบสภาพการจราจร ให้สามารถนับปริมาณรถบนท้องถนน และหาความหนาแน่นบนท้องถนน เพื่อที่จะได้นำข้อมูลเหล่านี้ไปใช้ในการแก้ไขปัญหารถติดต่อไป การทำงานของระบบจะทำการรับภาพเข้ามาผ่านขบวนการประมวลผลภาพ และนำมาทำการวิเคราะห์หาจำนวนรถ และความหนาแน่นของรถ โดยที่ระบบตรวจสอบสภาพการจราจรนี้สามารถทำงานได้ 2 แบบ คือทำงานในส่วนของเรียลไทม์ (Real-time) รับภาพจากกล้องวิดีโอเข้ามาประมวลในขณะนั้นเลยและการทำงานในส่วนของออฟไลน์ (Offline) ซึ่งจะนำภาพที่มีการบันทึกไว้ก่อนมาทำการประมวลผล

ถึงแม้ว่าปัจจุบันนี้คอมพิวเตอร์จะเข้ามามีบทบาทต่อชีวิตคนเรามากขึ้น สามารถช่วยเหลืองานที่ซับซ้อนยุ่งยากได้อย่างรวดเร็ว และช่วยให้มนุษย์มีความสะดวกสบายมากขึ้นกว่าแต่ก่อน แต่คอมพิวเตอร์ในปัจจุบันนี้ยังไม่สามารถช่วยในงานที่มีลักษณะที่ต้องอาศัยความฉลาดของมนุษย์ในการตัดสินใจได้ดั่งนี้ ซึ่งมีความไม่แน่นอนของการแก้ปัญหาเข้ามาเกี่ยวข้องกับเทคนิคทางค่านเทรชโฮล (Thresholding Technique) ในการคำนวณหาค่าของภาพที่มีลักษณะแตกต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background) และการแยกวัตถุออกจากภาพ (Segmentation) เพื่อนำข้อมูลเหล่านี้ไปสู่กระบวนการนับวัตถุ ที่สามารถตรวจจับได้

### 1.2 จุดมุ่งหมายของปัญญาประดิษฐ์ของภาคเรียนที่ 2

- เพื่อสร้าง โปรแกรมใช้งานที่สามารถประมวลผลภาพของยานพาหนะ ซึ่งใช้ในการตรวจสอบนับจำนวนและแยกประเภทของยานพาหนะบนถนนแต่ละช่องทางวิ่งได้
- เพื่อศึกษากระบวนการแยกวัตถุออกจากภาพ (Segmentation)
- เพื่อศึกษากระบวนการทำเทรชโฮล (Thresholding Technique) เพื่อให้ได้ภาพที่มีลักษณะแตกต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background)

### 1.3 ขอบเขตของโครงการ

โครงการนี้จะทำการรับข้อมูลรูปภาพเข้ามาทีละ frame โดยชนิดของไฟล์รูปภาพที่เข้ามาจะได้มาจากไฟล์วิดีโอซึ่งมีนามสกุลเป็น .avi จากนั้นทำการโหลดภาพที่เป็น Background และนำภาพทั้งหมดมาลบกับภาพ Background เพื่อให้ได้ภาพค่าเฉลี่ยของผลรวมของ Absolute Difference เพื่อนำข้อมูลนี้ไปวิเคราะห์หาจำนวนและแยกประเภทของยานพาหนะ และปริมาณความหนาแน่นของการจราจร

### 1.4 ประโยชน์และผลที่คาดว่าจะได้รับจากโครงการ

- ได้ทราบถึงหลักการประมวลผลภาพโดยใช้เทคโนโลยีเทคนิคทางด้านทรนซ์โซล (Thresholding Technique) และการแยกวัตถุออกจากภาพ (Segmentation)
- ได้ศึกษาการเขียนโปรแกรมประยุกต์ใช้งานด้วย Visual C++.NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎี

กลไกระบบการมองเห็นภาพนั้น จะหมายความรวมถึงทุกสิ่งที่เป็นต่อการได้มาซึ่งรหัสทางดิจิทัลที่ใช้แทนภาพนั้น ๆ การปรับปรุงเปลี่ยนแปลงข้อมูล และรวมทั้งการนำเสนอภาพที่ใดหลังการปรับปรุงแล้วด้วย ซึ่งระบบการมองเห็นภาพ (Vision System) นี้อาจมีความยุ่งยากซับซ้อนต่างกันตามลักษณะการใช้งาน

#### 2.1 การได้มาซึ่งภาพ (Image Acquisition)

การได้มาซึ่งภาพนี้ จะหมายถึงการแปลงภาพที่เราเห็น ในลักษณะทางกายภาพนั้นให้เป็นเซตของข้อมูลทางดิจิทัล เช่นการรับรูปภาพผ่านทางสแกนเนอร์ การรับรูปผ่านกล้องดิจิทัล จากนั้นก็จะเก็บค่าความสว่างของแต่ละพิกเซลอยู่ในรูปแบบเลขฐานสองจำนวน 8 บิต (หรือมากกว่า) ซึ่งเซตของพิกเซลเหล่านี้จะถูกส่งไปยังส่วนประมวลผลภาพต่อไป

#### 2.2 กระบวนการประมวลผลภาพ (Image Processing)

หน้าที่พื้นฐานของกระบวนการประมวลผลภาพ ได้แก่ การสร้างภาพใหม่โดยแยกแยะส่วนที่เราต้องการหรือสนใจกับสิ่งรบกวนออกจากกัน (Noise elimination), การหาขอบภาพ (Edge enhancement), การกรอง (Filtering), การปรับปรุงเปลี่ยนแปลงค่าระดับเกรย์ (Grey scale modification)

การประมวลผลภาพเชิงตัวเลข (Digital Image Processing) หมายถึง การนำภาพมาประมวลผลด้วยเครื่องคอมพิวเตอร์ โดยภาพที่นำมาประมวลผลนี้จะถูกแทนที่ด้วยตัวเลขให้อยู่ในรูปแบบของเมตริกซ์ แต่ภาพที่ได้โดยส่วนมากแล้วจะเป็นภาพที่ได้จากตัวรับสัญญาณ ซึ่งอยู่ในรูปของฟังก์ชัน  $f(x, y)$  ที่ต่อเนื่องในระนาบสองมิติ (คือแกน  $x$  และแกน  $y$ ) โดยจะเป็นสัดส่วนกับความสว่างหรือความเข้มของภาพ ที่ตำแหน่ง  $(x, y)$  ซึ่งเรียกว่า ระดับสีเทา (Grey Level)

##### 2.2.1 การแทนภาพด้วยข้อมูลแบบดิจิทัล

ภาพข้อมูลแบบดิจิทัล (Digital Image) เป็นภาพที่ถูกแปลงมาจากภาพอนาล็อกให้อยู่ในรูปแบบของตัวเลข โดยการนำภาพอนาล็อกมาแบ่งเป็นพื้นที่ที่สี่เหลี่ยมเล็ก ๆ ที่เรียกว่า พิกเซล (pixel) ในแต่ละพิกเซลจะถูกระบุตำแหน่งโดย  $(x, y)$  และค่าระดับสีเทาของพิกเซล โดยเราสามารถแปลงภาพเป็นข้อมูลแบบดิจิทัลได้ โดยมีขั้นตอนและวิธีการดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเรานำสัญญาณอนาล็อกที่ต้องการประมวลผลมาผ่านส่วนที่เรียกว่า ดิจิไทเซอร์ (Digitizer) ซึ่งจะมีหน้าที่ในการเปลี่ยนสัญญาณอนาล็อกให้เป็นสัญญาณดิจิทัล จากนั้นทำการควอนไทซ์ (Quantizing) เพื่อที่จะประมวลสัญญาณด้วยระบบคอมพิวเตอร์ ฟังก์ชันของภาพ  $f(x, y)$  จะถูกทำให้เป็นสัญญาณไม่ต่อเนื่องทั้งระนาบของภาพ ซึ่งเราเรียกว่าการสุ่มภาพ (Image Sampling) ของฟังก์ชันที่ได้เรียกว่า การควอนไทซ์ระดับสีเทา (Grey level quantization) ก็จะได้ข้อมูลที่เป็นดิจิทัล

### 2.2.2 ลักษณะการจัดเก็บข้อมูลภาพแบบดิจิทัล

โดยทั่วไปแล้ว ข้อมูลภาพจะมีค่าความเข้มตั้งแต่ 2 ระดับขึ้นไป แต่ที่ใช้กันมากจะใช้กันที่ค่าระดับความเข้มของจุดภาพเท่ากับ 256 ระดับ ซึ่งจะทำให้ค่าของจุดภาพอยู่ในช่วง (0-255) โดยใช้เนื้อที่การเก็บข้อมูลภาพขนาด 1 ไบต์ หรือ 8 บิต สำหรับข้อมูล 1 จุดภาพ ( $2^8 = 256$ ) ในกรณีที่ต้องการภาพที่มีความละเอียดของระดับความเข้มสูงๆ อาจจะต้องการจำนวนบิตสำหรับการเก็บข้อมูลมากกว่า 8 บิต คือ อาจจะเป็น 16 หรือ 24 บิต โดยค่าความเข้มของจุดภาพจะเท่ากับ  $2^{16}$  และ  $2^{24}$  โดยจะแยกให้เห็นชัดเจนดังนี้

1. ภาพ 2 ระดับ คือ มีเพียงแค่จุดขาวกับจุดดำเท่านั้น โดยแต่ละจุดภาพเป็นข้อมูลขนาด 1 บิต
2. ภาพ 16 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 4 บิต ซึ่งทำให้สามารถแสดงได้ 16 ระดับสี หรือ 16 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
3. ภาพ 256 ระดับ คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 8 บิต ซึ่งทำให้สามารถแสดงภาพได้ 256 ระดับสี หรือ 256 เกรย์สเกล ขึ้นอยู่กับภาพนั้นเป็นภาพสีหรือภาพขาวดำ
4. ภาพทิวทัศน์ (True Color) คือ ในแต่ละจุดภาพจะมีขนาดของข้อมูล 24 บิต ทำให้สามารถแสดงผลภาพได้เหมือนภาพจริงที่สุด เพราะสามารถแสดงสีได้ถึง 16,777,216 สี

การแสดงผลภาพนี้ใช้วิธีตั้งค่าของแม่สีในตารางสี โดยอาจเลือกสีเป็นแบบ 16 สี จาก 64 สี หรือ 16 สี จาก 262,144 สี หรือ 256 สี จาก 262,144 สี ขึ้นอยู่กับโหมดการแสดงผล สำหรับทิวทัศน์ จะไม่มีการเลือกสีแสดงผลโดยการส่งค่าสี RGB ผ่าน D/A ทีละ 8 บิต ออกไป ความแตกต่างของการแสดงผลสีและขาวดำ คือภาพขาวดำจะต้องตั้งให้แม่สีทั้งสามสี มีค่าเท่ากัน เนื่องจาก VGA กำหนดให้แม่สีแต่ละสีใช้ได้เพียง 64 ระดับเท่านั้น หากต้องการให้เห็นจริงทั้ง 256 ระดับ ต้องแสดงในโหมดทิวทัศน์แล้วให้ RGB มีค่าเท่ากัน ซึ่งในโหมดนี้ จะสามารถใช้ รีจิสเตอร์ ได้ 8 บิต สำหรับแต่ละแม่สี

โดยทั่วไปวิธีการประมวลผลภาพเชิงตัวเลขที่ทำให้คอมพิวเตอร์สามารถรู้จักวัตถุภายในภาพได้นั้นพอจะแบ่งได้สองระดับด้วยกันคือ การประมวลผลภาพในระดับต่ำ (Low-Level Image Processing) และการประมวลผลภาพในระดับสูง (High-Level Image Processing) การประมวลผลในระดับต่ำจะ

เป็นการประมวลผลเชิงตัวเลขทั้งหมด เพื่อหาตัวแปรต่างๆ มาอธิบายข้อมูลภาพ โดยมีจุดประสงค์เพื่อนำตัวแปรเหล่านั้นไปใช้ในการประมวลผลระดับสูงต่อไป โดยทั่วไปแล้วการประมวลผลภาพระดับต่ำจะประกอบไปด้วย การประมวลผลภาพก่อน (Preprocessing) การกำจัดสัญญาณรบกวนหรือการทำให้ภาพคมชัด การหาขอบภาพ เป็นต้น

การประมวลผลภาพในระดับสูง เป็นการนำผลลัพธ์หรือสัญลักษณ์ที่ได้จากการประมวลผลระดับต่ำมาตีความหรือประมวลผลเพื่อให้คอมพิวเตอร์สามารถรู้จักและเข้าใจภาพได้สำหรับความแตกต่างของการประมวลผลภาพระดับต่ำและการประมวลผลภาพระดับสูงนั้นคือ ข้อมูลที่นำมาใช้ในการประมวลผลภาพ โดยที่การประมวลผลภาพระดับต่ำจะใช้ค่าความสว่างของจุดโดยตรง ส่วนการประมวลผลภาพระดับสูงนั้นข้อมูลภาพที่นำมาประมวลผลจะถูกแสดงในรูปของสัญลักษณ์ ซึ่งสัญลักษณ์เหล่านี้จะแสดงถึงสิ่งต่างๆ ที่มีอยู่ในภาพ เช่น ขนาดของวัตถุ รูปร่าง และความสัมพันธ์กันระหว่างวัตถุภายในภาพ

## 2.3 ไฟล์ข้อมูลภาพชนิดบิตแมป

### 2.3.1 รูปแบบของไฟล์ข้อมูลภาพชนิดบิตแมป

รูปแบบของไฟล์ข้อมูลชนิดบิตแมป เป็นฟอร์แมตของวินโดวส์บิตแมป ซึ่งเป็นมาตรฐานสำหรับไฟล์กราฟิกบนวินโดวส์ ซึ่งจะใช้ในการตัดต่อ หรือสำเนาภาพต่างๆ ลงบนคลิปบอร์ด (Clipboard) เมื่อเวลาจัดเก็บไฟล์ที่มีสกุล BMP

### 2.3.2 โครงสร้างของไฟล์ของข้อมูลภาพชนิดบิตแมป

โครงสร้างของไฟล์ข้อมูลภาพชนิดบิตแมป จะประกอบด้วย 3 ส่วนคือ

2.3.2.1 ข้อมูลเฮดเดอร์ (Header)

2.3.2.2 ข้อมูลจานสี (Palette)

2.3.2.3 ข้อมูลภาพ (Data)

**2.3.2.1 ข้อมูลเฮดเดอร์** คือ ข้อมูลที่อยู่บริเวณส่วนหัวของไฟล์ซึ่งประกอบไปด้วยข้อมูลที่บอกรายละเอียดต่างๆ ของภาพ เช่น ความกว้าง ความยาวภาพ จำนวนสี จำนวนบิต ความละเอียด เป็นต้น

**2.3.2.2 ข้อมูลงานสี** คือ ข้อมูลที่บอกถึงชุดของงานสีที่เกิดจากการผสมแม่สีทั้งสาม คือ แดง เขียว และน้ำเงินมาผสมกันได้เป็นสีต่างๆ ตามจำนวนสีของภาพ เช่น รูปขนาด 4 บิตจะมี 16 ระดับสี รูป 8 บิต จะมีขนาด 256 สี เป็นต้น ซึ่งถ้ามีจำนวนสีน้อยๆ ก็จะมีการเก็บค่างานสีนี้ลงไฟล์ด้วย แต่ถ้ารูป ประเภท 24 บิต จะไม่มีค่างานสี แต่จะใช้วิธีเก็บค่าแม่สีทั้งสามลงไปเป็นข้อมูลแทนเพราะถ้าเก็บค่างานสีที่มีถึง 16.7 ล้านสีลงไปด้วยจะเปลืองพื้นที่มาก ข้อแตกต่างที่สำคัญของบิตแมปขนาดนี้คือ ไฟล์บิตแมปจะเก็บงานสีชุดละ 4 ไบต์ แต่ก็ใช้แค่ 3 ไบต์ เช่นกันคือ แดง เขียว และน้ำเงิน

**2.3.2.3 ข้อมูลภาพ** คือ ข้อมูลสีของภาพแต่ละจุดที่มาประกอบกันเป็นรูปภาพ ซึ่งค่าที่เก็บนี้จะเป็นค่าที่ใช้ในการชี้ตาราง Palette หมายเลขอะไร เช่น จุดแรกมีค่าเป็น 10 ก็ให้ไปเปิดตาราง Palette หมายเลข 10 สมมุติว่าของแม่สีเป็น  $R = 0$ ,  $G = 0$ , และ  $B = 100$  ก็จะได้จุดนี้เป็นสีน้ำเงิน ซึ่ง ถ้าเป็นกรณีของรูป 24 บิต จะเป็นการอ่านข้อมูลขึ้นมา 3 ค่า เป็นค่าของแม่สี RGB แล้วนำไปผสมบนจอภาพแทน

## 2.4 การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป

การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป มีการเก็บอยู่ 2 แบบ คือ

### 2.4.1 แบบบีบอัดข้อมูล

- RLE 4 เป็นการบีบอัดข้อมูลแบบ Run-length Encoder แบบ 4 บิต
- RLE 8 เป็นการบีบอัดข้อมูลแบบ Run-length Encoder แบบ 8 บิต

### 2.4.2 แบบไม่ได้บีบอัดข้อมูล

เป็นการเก็บข้อมูลจริงของสีของพิกเซล ซึ่งทำให้ขนาดของไฟล์ค่อนข้างใหญ่ แต่จะทำการแสดงภาพได้เร็วกว่า เพราะไม่ต้องเสียเวลาในการคลายข้อมูล

## 2.5 ระดับสีเทา (Grey Scale)

หากเราต้องการค่าข้อมูลที่ละเอียดมากขึ้น ก็จำเป็นที่จะต้องเพิ่มจำนวนบิตการแสดงค่าของแต่ละพิกเซล ยกตัวอย่างเช่น หากแบ่งความเข้มของการส่องสว่างให้มี 4 ระดับ ก็ต้องใช้เลขฐานสองจำนวน 2 บิต และจำนวน 4 บิตสำหรับ 16 ระดับ, 8 บิตสำหรับ 256 ระดับ ซึ่งจำนวนระดับที่ใช้ในระดับสีเทาหรือระดับเกรย์นี้มักเป็นเลขยกกำลังของ 2, ค่าที่ต่ำที่สุดคือ 0 กำหนดให้เป็นสีดำ และ 1 หรือ

ตัวเลขที่น้อยกว่าค่าสูงสุดของระดับเกรย์อยู่ 1 (เช่น 15 สำหรับระดับเกรย์ 16 ระดับ) แทนสีขาว ค่าที่กำหนดให้ในแต่ละพิกเซลนี้นิยมใช้เป็นจำนวนเต็ม

ในยุคแรก ๆ ของระบบการมองเห็นภาพจะใช้ระบบเลขฐานสอง แต่ในปัจจุบันเทคโนโลยีไมโครโปรเซสเซอร์เข้ามามีบทบาทมากขึ้น การแบ่งระดับเป็น 16, 64 หรือ 256 เป็นเรื่องธรรมดาแต่ทั้งนี้ในการมองเห็นของมนุษย์จะสามารถแยกแยะความแตกต่างได้เพียง 10-15 ระดับเท่านั้น การแบ่งโดยละเอียดเป็น 64 หรือ 256 ระดับอาจนำไปประยุกต์ใช้กับงานการประมวลผลภาพหรือกระบวนการทางอุตสาหกรรมอื่น ๆ

ซึ่งจะเห็นว่าจำนวนระดับเกรย์จะเป็นตัวจำกัดรายละเอียดของภาพ โดยทั่วไปแล้วยิ่งแบ่งระดับเกรย์เป็นหลายระดับก็จะเป็นการเพิ่มคุณภาพของภาพด้วย และการเพิ่มจำนวนพิกเซล เช่น จาก  $35 * 35$  เป็น  $250 * 250$  ก็จะเป็นการเพิ่มความละเอียด (Resolution) และรายละเอียด (Detail) ของภาพเช่นกัน จะเห็นว่าจะแตกต่างกับการขยายภาพ (Zoom) โดยการขยายภาพนี้เป็นการขยายแต่ละพิกเซลให้ใหญ่ขึ้น ไม่ได้เป็นการเพิ่มจำนวนพิกเซล

## 2.6 การสร้างภาพไบนารี

อุปกรณ์ที่มีความสามารถในการแสดงผลได้แค่ 2 ระดับ หรือ 2 สี คือ สีขาวกับสีดำยังมีการใช้กันอย่างแพร่หลาย เช่น เครื่องพิมพ์ (Printer), เครื่องโทรสาร (Fax), จอภาพแสดงผลแบบโมโนโครม (Monochrome Monitor) เป็นต้น เนื่องจากอุปกรณ์เหล่านี้เป็นอุปกรณ์ที่มีราคาถูก ดังนั้นจึงเกิดปัญหาในการแสดงผลขึ้น การที่จะแสดงผลหรือพิมพ์รูปภาพที่มีระดับความเข้มของภาพหลายระดับ ซึ่งมีมากกว่าความสามารถในการแสดงผลของอุปกรณ์เหล่านั้นที่มีเพียงแค่ 2 ระดับ เราจะต้องทำการแปลงข้อมูลภาพให้เป็นภาพไบนารี (Binary Image) เสียก่อน ซึ่งการสร้างภาพไบนารีนั้นก็หมายถึงการแปลงข้อมูลภาพที่มีระดับความเข้มหลายระดับ (Multilevel Image) ให้เป็นภาพที่มีระดับความเข้มเพียง 2 ระดับ นั่นคือ 1 จุดภาพ มีได้ 2 ค่าเท่านั้น คือ 0 กับ 1 โดยจุดภาพที่แทนด้วย 0 หมายถึงจุดภาพที่มีสีดำ ส่วนจุดที่แทนด้วย 1 จะหมายถึงจุดภาพที่มีสีขาว เมื่อทำการแปลงเป็นภาพไบนารีแล้ว จึงสามารถนำภาพนั้นไปแสดงผลบนอุปกรณ์เหล่านั้น จะเห็นได้ว่าการแปลงข้อมูลภาพหลายระดับเป็นภาพไบนารีจึงมีความจำเป็นและมีประโยชน์มากในการแสดงผลภาพที่มีความเข้มของภาพหลายระดับบนอุปกรณ์ที่มีความสามารถในการแสดงผลได้ 2 ระดับ สำหรับประโยชน์อีกประการหนึ่งในการแปลงข้อมูลภาพนั้นเป็นภาพไบนารี คือการลดเนื้อที่การเก็บข้อมูลภาพ ภาพไบนารีนี้จะสามารถลดเนื้อที่ลงได้ถึง 8 เท่า นั่นคือถ้าเดิม 1 จุดภาพใช้เนื้อที่ในการเก็บ 8 บิต เมื่อทำการแปลงเป็นภาพไบนารีแล้วจะใช้เนื้อที่ในการเก็บ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1 จุดภาพนั้นเพียง 1 บิตเท่านั้น อีกทั้งยังสามารถนำไปประยุกต์ใช้งานได้อย่างแพร่หลาย เช่น นำไปประยุกต์ใช้ในการวิเคราะห์เอกสารในขั้นตอนที่เรียกว่าการประมวลผลขั้นต้น (Preprocessing) เป็นต้น

ในการสร้างภาพไบนารี สามารถทำได้โดยใช้เทคนิคการทำเทรชโฮล (Thresholding Technique) โดยพิจารณาว่าจุดภาพใดควรจะเป็นจุดขาวหรือจุดดำ จะกระทำโดยการเปรียบเทียบระหว่างจุดภาพเริ่มต้นกับค่าคงที่ค่าหนึ่งซึ่งเรียกว่า ค่าเทรชโฮล (Threshold Value) เทคนิคนี้ใช้กันมากในกรณีที่ข้อมูลภาพมีลักษณะแตกต่างกันระหว่างวัตถุ (Object) และพื้นหลัง (Background) โดยค่าของจุดภาพใด ๆ ที่มีค่าน้อยกว่าค่าเทรชโฮลจะถูกกำหนดให้เป็น 0 (จุดดำ) และถ้าค่าของจุดภาพใด ๆ ที่มีค่ามากกว่าหรือเท่ากับค่าเทรชโฮล จะถูกเปลี่ยนให้เป็น 1 (จุดขาว)

ในการสร้างภาพไบนารีโดยใช้เทคนิคเทรชโฮลเพื่อให้ได้ผลลัพธ์ที่เหมาะสมและคมชัด สิ่งที่สำคัญที่สุดคือ ค่าเทรชโฮล เนื่องจากถ้าเลือกค่าเทรชโฮลที่ไม่เหมาะสม (ค่าเทรชโฮลที่มีค่าน้อยเกินไป หรือ มากเกินไป) ภาพที่ได้อาจจะไม่เหมาะสม ขาดความคมชัด และอาจทำให้รายละเอียดบางส่วนขาดหายไป กล่าวคือภาพที่ได้อาจจะมืดเกินไปหรือสว่างเกินไป หรือภาพที่ได้มีสิ่งรบกวน (Noise) เกิดขึ้น อันเป็นผลทำให้ภาพผลลัพธ์ที่ได้ไม่สวยงามเท่าที่ควร ดังนั้นปัญหาของการสร้างภาพไบนารีโดยวิธีเทรชโฮลนี้คือ ทำอย่างไรจึงจะสามารถกำหนดค่าเทรชโฮลที่เหมาะสมสำหรับแต่ละภาพที่จะนำมาทำการสร้างภาพไบนารี ซึ่งมีวิธีกำหนดค่าเทรชโฮลหลายวิธี โดยแต่ละวิธีเหมาะสมกับลักษณะการทำงานที่แตกต่างกันไป เช่น การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Pre-assigned Threshold Value) การหาค่าเทรชโฮลจากค่ากลาง (Mid-range Threshold Value) ซึ่งแต่ละวิธีอธิบายได้ดังนี้

#### • การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า (Pre-assigned Threshold Value)

การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้านี้เป็นวิธีที่ง่ายที่สุด เป็นการกำหนดค่าเทรชโฮลโดยการกำหนดเองจากผู้ใช้ ซึ่งการกำหนดนี้จะขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้น ๆ ทำได้โดยการเลือกค่าคงที่ค่าหนึ่ง ซึ่งเรียกค่านั้นว่า “ค่าเทรชโฮล” โดยค่าที่เลือกมานี้จะเป็นค่าที่อยู่ระหว่างค่าต่ำสุดและค่าสูงสุดของระดับความเข้มของข้อมูลภาพอินพุท เช่น ภาพข้อมูลอินพุทมีเกรย์สเกล 256 ระดับ จะมีค่าเกรย์สเกลได้ตั้งแต่ 0-255

#### • การหาค่าเทรชโฮลจากค่ากลาง (Mid-Range Threshold Value)

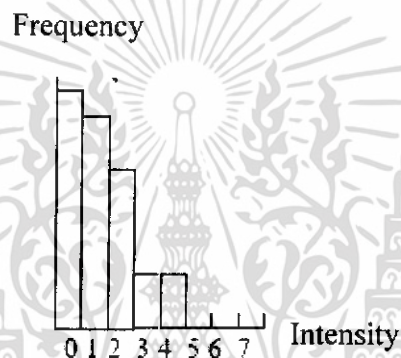
การหาค่าเทรชโฮลโดยพิจารณาจากค่ากลาง เป็นการหาค่าเทรชโฮลที่แตกต่างจากการหาค่าเทรชโฮลวิธีแรก สำหรับวิธีนี้จะเป็นการคำนวณหาค่าเทรชโฮลโดยอัตโนมัติโดยผู้ใช้ไม่ต้องเป็นผู้กำหนดเอง การคำนวณหาค่าได้อาศัยการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลาง หรือค่าเฉลี่ย (Mean) มาประยุกต์ใช้ ซึ่งค่าเทรชโฮลที่คำนวณได้จะเป็นค่าที่เหมาะสมสำหรับภาพนั้น ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.6.1 ฮิสโตแกรม (Histogram)

ฮิสโตแกรมเป็นการแสดงให้เห็นถึงความถี่ของการนับจำนวนพิกเซล ที่มีค่าความเข้มแต่ละค่าหนึ่ง ๆ ในภาพระดับเกรย์ โดยแกน x ในกราฟแสดงค่าระดับเกรย์ และแกน y แสดงค่าจำนวนพิกเซล ขั้นตอนการสร้างฮิสโตแกรม มีดังนี้

1. ทำการดิจิไทซ์ (Digitizing) ภาพ
2. นับจุดพิกเซลในแต่ละระดับเกรย์
3. พล็อตกราฟระหว่างจำนวนจุดกับระดับเกรย์



รูปที่ 2.1 ภาพฮิสโตแกรมของ 8 ระดับเกรย์

รูปร่างของฮิสโตแกรมสามารถบอกลักษณะบางประการของภาพได้ เช่น หากฮิสโตแกรมมีลักษณะที่แคบ จะหมายถึงภาพนั้นขาดการแยกแยะความผิดเพี้ยนของสี (Contrast) เป็นต้น ฮิสโตแกรมนี้มีประโยชน์ในการกำหนดค่าเทรซโฮล เพื่อสำหรับการเปลี่ยนภาพระดับเกรย์ให้เป็นภาพไบนารี

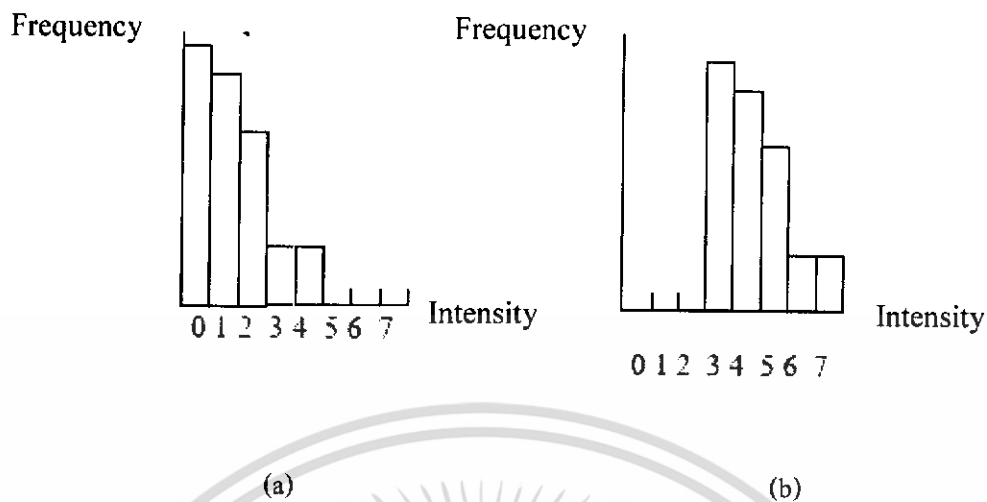
### 2.6.2 การแปลงระดับเกรย์ (Grey-level Transformation)

ประโยชน์ของการแปลงระดับเกรย์

1. การเปลี่ยนแปลงความสว่าง (Global alterations in brightness)

ทำได้โดยการใช้ค่าคงที่บวกหรือลบออกจากทุกพิกเซลของภาพ เพื่อเพิ่มหรือลดความสว่างของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 (a) ฮิสโตแกรมจากภาพต้นฉบับ

(b) ฮิสโตแกรมของภาพหลังจากเพิ่มความสว่างเข้าไปอีก 3 ให้กับทุกจุดภาพ

## 2. การทำเทรชโฮลด์ (Threshold)

เป็นการเปลี่ยนแปลงหรือหาแนวโน้มของค่าระดับเกรย์ในภาพเพื่อให้ค่าระดับเกรย์เกิดความไม่ต่อเนื่อง (discrete) มากขึ้น โดยจะนำภาพมาทำฮิสโตแกรม แล้วกำหนดค่าระดับเกรย์ที่แน่นอนขึ้นมาเพื่อที่จะเป็นจุดตัดในการทำเทรชโฮลด์ จากนั้นจะทำการตัดหรือปิดส่วนของระดับเกรย์ที่เราไม่ต้องการออกไป

## 3. บันช์ซิง (Bunching) และการควอนไทซ์ (Quantize)

ใช้เพื่อลดระดับเกรย์ที่ไม่ต้องการในภาพลง ในบางครั้งบันช์ซิงจะมีความหมายรวมถึงการควอนไทซ์ด้วย

## 4. สปลิตติง (Splitting)

เป็นการเพิ่มความแตกต่างระหว่าง 2 กลุ่มของระดับเกรย์ ยกตัวอย่างเช่น หากเรามีตัวอักษรเขียนอยู่บนฉาก โดยตัวอักษรมีระดับเกรย์ที่ 98 และฉากมีระดับเกรย์ที่ 99 ซึ่งตาของมนุษย์ไม่สามารถแยกแยะความแตกต่างเพียงเท่านี้ได้ , ฮิสโตแกรมที่ได้ก็จะมีลักษณะแบน ดังนั้นจึงแก้ปัญหาด้วยการทำการสปลิตฮิสโตแกรม โดยทำการดึงค่า 99 ขึ้นเป็น 120 และดึงค่า 98 ลงมาเป็น 80 ซึ่งก็จะทำให้ระหว่างตัวเลขและตัวอักษรมีความแตกต่างกันมากขึ้นจนสามารถสังเกตเห็นแยกแยะได้

เทคนิคนี้มีประโยชน์มากในกรณีที่เราต้องการดึงเฉพาะบางส่วนของภาพที่มีความสว่างใกล้เคียงกันมากออกมา

## 2.7 การแยกวัตถุจากภาพ(Segmentation)

กระบวนการสำคัญอีกขั้นตอนหนึ่งในการประมวลผลเบื้องต้นก่อนจะนำไปสู่ขั้นตอนการจดจำรูปแบบ ก็คือ กระบวนการแยกวัตถุออกจากพื้นหลัง ซึ่งในที่นี้จะเป็นการแยกข้อมูลภาพที่เป็นยานพาหนะออกจากข้อมูลภาพทั้งหมดโดยจะนำรูปภาพนั้นมาลบกัน เพื่อนำไปสู่กระบวนการนับจำนวนวัตถุซึ่งสามารถประมวลผลได้ดีเมื่อมีวัตถุผ่านเข้ามาทีละหนึ่ง

### 2.7.1 การแยกภาพด้วยการพิจารณาการต่อเนื่องของข้อมูล

เมื่อรับข้อมูลภาพที่ได้จากการเปลี่ยนข้อมูลเป็นรูปแบบไบนารีที่มีค่า 0 กับ 1 แล้ว ซึ่งข้อมูล 0 จะแทน ส่วนที่เป็นพื้นหลัง และ ข้อมูล 1 จะแทนส่วนที่เป็นวัตถุ หลักการเบื้องต้นคือ หากค่าพิกเซลที่เป็น 0 ที่ต่อเนื่องกันตลอดทั้งแนวตั้งและแนวนอนทำให้ได้ขนาดของกรอบ(Block) ข้อมูลภาพวัตถุที่มีขนาดต่างๆกัน จากนั้นจะพิจารณาเลือกขนาดของกรอบที่ต้องการจากความแตกต่างของจำนวนพิกเซล ความสูงความกว้าง และตำแหน่ง เป็นต้น

### 2.7.2 การแยกภาพด้วยวิธี Region Labeling

ในการจำแนกภาพ โดยวิธีนี้ได้ถือว่าบริเวณที่อยู่ข้างเคียงเป็นบริเวณที่สำคัญมาก จุดภาพที่อยู่ข้างเคียงกันมักจะมีคุณสมบัติทางสถิติที่คล้ายกันหรือใกล้เคียงกันสำหรับจุดรอบข้างที่มาเชื่อมต่อกัน ในวิธีนี้จะทำการพิจารณาภาพบริเวณย่อยๆจำนวนมากจากนั้นพื้นที่ติดกันจะถูกนำมาพิจารณาถึงความ เป็นเนื้อเดียวกันร่วมกัน การรวมตัวกันจะสิ้นสุดเมื่อพื้นที่ข้างเคียงไม่สามารถพิจารณาถึงความเป็นเนื้อเดียวกันได้ แต่ถ้าจุดของภาพที่อยู่ใกล้เคียงกันนั้นตรวจสอบแล้วไม่อยู่ในเกณฑ์การรวม จุดภาพนั้นจะไม่ถูกรวมเข้าไปในส่วนนั้นของภาพแต่จะถูกเลือกให้เป็นจุดเริ่มต้นของส่วนอื่นๆต่อไป และหลังจากที่จุดภาพทุกจุดได้รวมตัวกันเป็นกลุ่มเรียบร้อยแล้ว

ในกรณีนี้จะกล่าวถึงภาพที่มีวัตถุในภาพมาก วิธีที่จะแยกวัตถุออกจากกันจะทำได้โดยพิจารณาจากการติดกันของพิกเซลที่เป็น 1 โดยสามารถพิจารณาได้ดังนี้

- การติดกันแบบ 4 จุด จะพิจารณา 4 พิกเซลรอบข้างทางด้านแนวนอน และแนวตั้ง
- การติดกันแบบ 8 จุด จะพิจารณา 8 พิกเซลรอบทางด้านแนวนอนและแนวตั้ง

วิธีการแยกวัตถุแบบ Region Labeling นี้ จะพิจารณาแบบไบนารีเฉพาะพิกเซลที่มีค่า 1 ทีละแถวจากซ้ายไปขวาและจากบนลงล่าง ซึ่งเมื่อพิจารณาที่พิกเซลใดพิกเซลหนึ่ง ถ้าพิกเซลแถวบนและทางซ้ายซึ่งผ่านการกำหนด Label แล้ว มีค่าไม่เป็น 1 ก็จะกำหนด Label เหมือนกับพิกเซลข้างเคียงที่เป็น 1 แต่ถ้าในกรณีที่พิกเซลข้างเคียงมีค่า 1 มากกว่า 1 พิกเซล และแต่ละพิกเซลมี Label ต่างกัน ก็จะ

กำหนด Label ให้กับพิกเซลที่พิจารณาอยู่ให้เหมือนกับ Label ที่สมมูลกันให้เหมือนกันซึ่งจะสามารถทราบถึงความแตกต่างของแต่ละวัตถุในภาพ โดยดูจาก Label ที่แตกต่างกัน

การทำการประมวลผลรูปภาพต้องทำการศึกษาพื้นฐานของรูปภาพก่อนว่ามีการสร้างอย่างไร ซึ่งในรูปภาพจะมีองค์ประกอบหนึ่งที่สามารถนำมาประมวลผลได้ คือ “พิกเซล(Pixel)” พิกเซลจะมีลักษณะที่เป็นจุดสีที่เล็กลงๆ ที่มีอยู่มากมายอยู่บนรูปภาพ ดังนั้นการประมวลผลภาพต้องนำพิกเซลแต่ละรูปภาพมาทำการประมวลผล โดยแต่ละพิกเซลของรูปภาพจะประกอบด้วยรายละเอียดสีต่างๆ มารวมกันเป็นพิกเซล ซึ่งเรียกกลุ่มของสีที่ชื่อว่า “Color Space” ซึ่งรูปแบบการบอกสีนั้นมีอยู่หลายรูปแบบ

## 2.8 กลุ่มสีที่ใช้แสดงผล

การเกิดภาพต่างๆบนจอคอมพิวเตอร์ จะเกิดจากการเรียงตัวของจุดสีที่เล็กลงๆ สีต่างๆ ที่เราเรียกว่า “พิกเซล” ซึ่ง “พิกเซล” เป็นตัวกำหนดความคมชัดของภาพในจอคอมพิวเตอร์ ยิ่งเรากำหนดให้ภาพแต่ละภาพมีจำนวนพิกเซลมาก ภาพก็จะมีรายละเอียดมากขึ้นตามไปด้วย แต่จะทำให้ไฟล์ของภาพมีขนาดใหญ่ขึ้นเนื่องจากภาพนั้นมีความละเอียดที่เพิ่มขึ้น

พิกเซล (Pixel มาจากคำว่า “Picture Element”) เป็นหน่วยพื้นฐานของสีที่สามารถโปรแกรมได้บนจอแสดงผลหรือบนภาพ ขนาดทางกายภาพของพิกเซลจะขึ้นอยู่กับข้อกำหนดความละเอียดของจอภาพ ถ้าทำการแสดงผลให้มีความละเอียดสูงสุด ขนาดทางกายภาพของพิกเซลก็จะเท่ากับขนาดทางกายภาพของจุด ( Dot Pitch หรือ Dot Size ) บนจอภาพ แต่ถ้าตั้งความละเอียดไว้ต่ำกว่าค่าสูงสุด พิกเซลก็จะมีขนาดใหญ่กว่าขนาดทางกายภาพของจุดบนจอภาพ แสดงว่าแต่ละพิกเซลจะใช้น้ำหนักกว่า 1 จุดในการแสดงผล

ในปัจจุบันกลุ่มสีที่แสดงผลมีมากมายหลายแบบ เช่น กลุ่มสีที่แสดงผล RGB เป็นกลุ่มสีที่แสดงผลที่แสดงลักษณะของสี 3 สี คือ สีแดง สีเขียว และสีน้ำเงิน ส่วนกลุ่มสีที่แสดงผล HSV เป็นกลุ่มสีที่แสดงผลที่บอกลักษณะของภาพโดยใช้ ระดับสี ความเข้มของสี ความสว่างของสี เป็นต้น

### 2.8.1 คุณสมบัติของกลุ่มสีที่แสดงผล

โดยทั่วไปแล้วภาพจะอยู่ในรูปแบบของกลุ่มสีที่แสดงผล RGB ซึ่งแทนสีทั้งหมดในกลุ่มสีที่แสดงผลด้วย เวกเตอร์ 3 มิติ เราจึงใช้กลุ่มสีที่แสดงผลนี้เป็นจุดเริ่มต้นของลักษณะเฉพาะของภาพได้โดยใช้ลักษณะของสี (Color Feature) จากนั้นค่อยทำการลดระดับของสีให้เหมาะสมต่อการทำฮิสโตแกรม คุณสมบัติดังกล่าวคือ

### 2.8.1.1. Uniformity

Uniformity เป็นคุณสมบัติที่ใช้ในการเปรียบเทียบความใกล้เคียงของเมตริกสีโดยกลุ่มสีที่แสดงจะมีคุณสมบัตินี้จะต้องคำนวณได้ง่าย อย่างเช่น การหาความใกล้เคียงของเมตริกสีโดยใช้ฟังก์ชันที่ไม่ขึ้นกับตำแหน่งในกลุ่มสีนั้น การทำการแปลงสีจะเป็นสีที่ทำให้กลุ่มสีมีคุณสมบัตินี้

### 2.8.1.2 Completeness

Completeness เป็นคุณสมบัติสำคัญที่กลุ่มสีที่แสดงผลจะต้องมีสีที่เราสามารถแยกแยะได้อย่างชัดเจน สามารถทำได้โดยการทำการลดระดับสี

### 2.8.1.3 Compactness

Compactness เป็นคุณสมบัติที่สีในกลุ่มสีที่แสดงผลจะต้องแตกต่างจากสีอื่นๆ ในกลุ่มสีที่แสดงผลเดียวกัน คุณสมบัตินี้จะเป็นตัวกำหนดมิติของภาพเพื่อให้แน่ใจว่า เมื่อผ่านการเปรียบเทียบจากกลุ่มสีที่แสดงผล RGB แล้วจะไม่มีสี 2 สีที่เหมือนกัน การลดระดับสีจะทำให้เกิดคุณสมบัตินี้

### 2.8.1.4 Naturalness

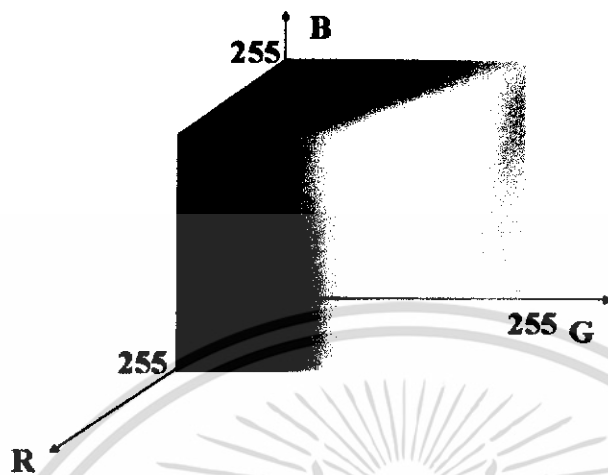
Naturalness เป็นคุณสมบัติของสีที่ผู้ใช้สามารถเข้าใจได้ง่าย ทำให้ผู้ใช้สามารถทำการดึงข้อมูลจากฐานข้อมูลโดยใช้สีเป็นดัชนีได้ ซึ่งสีดังกล่าวจะถูกแยกแยะออกเป็น 3 ส่วน คือ ความสว่าง (Brightness), ระดับสี (Hue), ความเข้มของสี (Saturation)

- ความสว่าง คือ คุณสมบัติที่มีผลกระทบต่อสีในด้านของแสง เช่น การ เปลี่ยนแปลงระหว่างความสว่างถึงความมืด (Bright to Dim)
- ระดับสี คือ คุณสมบัติที่มีผลกระทบต่อสีที่จะทำให้สีเหมือนกับ สีแดง สีเหลือง สีเขียว สีนํ้าเงิน ฯลฯ
- ความเข้มของสี คือ คุณสมบัติที่มีผลกระทบต่อสีหนึ่งๆ ของระดับสี โดยพิจารณาถึงปริมาณของสี (Color Light) โดยไม่สนใจถึงความสว่าง

## 2.8.2 กลุ่มสีแสดงผลแบบ RGB

ภาพสีที่อยู่ในรูปแบบของ RGB จะเป็นรูปแบบพื้นฐานของภาพที่ใช้แสดงบนจอคอมพิวเตอร์ โดยแต่ละ พิกเซลของภาพจะประกอบด้วยองค์ประกอบของสี 3 ค่า คือสีแดง,สีเขียว,สีนํ้าเงิน โดยแสดงองค์ประกอบสีแต่ละ n บิต ดังนั้นจะสามารถแสดงเฉดสีได้องค์ประกอบสีละ  $2^n$  เฉดสี ที่แตกต่างกัน รูปที่ 2.1 แสดงความสัมพันธ์ของสีแดง สีเขียว และสีนํ้าเงิน ใน 1 พิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.1 แสดงความสัมพันธ์ของสีในกลุ่มสีแบบ RGB ใน 1 Pixel

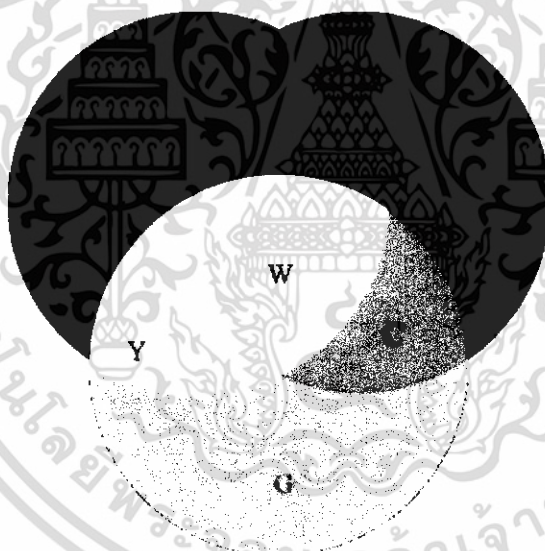
ในระบบแกน R จะแสดงปริมาณของสีแดงที่มีในพิกเซลนั้น แกน G จะแสดงปริมาณของสีเขียวที่มีในพิกเซลนั้น และแกน B จะแสดงปริมาณของสีน้ำเงินที่มีในพิกเซลนั้น เนื่องจากเก็บสีแยกเป็นคณค่าของสีกันทำให้เมื่อมีการคำนวณต้องคำนวณแยกทีละค่าของสี

การผสมสีบนคอมพิวเตอร์นั้นอาศัยระดับสีหลัก 3 สีคือ สีแดง สีเขียว และ น้ำเงิน หรือที่เราเรียกว่า “RGB Colors” (Red-Green-Blue) นั่นเอง ความเหมือนจริงของสีคอมพิวเตอร์นั้น ขึ้นอยู่กับในหนึ่งจุดพิกเซลของการแสดงผลนั้นใช้ระดับของสี (Color depths) ที่มีค่าเท่าไร เช่นถ้าสี RGB มีระดับของสีเป็น 8 planes นั่นคือ เราใช้ 8 บิตเก็บข้อมูลหนึ่งสี หมายความว่า เฉพาะระดับสีหลัก เช่น สีแดง สีเขียวก็จะเป็นความเป็นแดง อยู่ถึง  $2^8 = 256$  ระดับ พุดง่าย ๆ ก็คือว่า สมมติเราจะผสมสี หนึ่งสี จาก แดง-เขียว-น้ำเงิน (RGB) เราจะใช้สีแดง ก็ส่วนจาก 0 ถึง 255 ส่วน ใช้เขียว ก็ส่วนจาก 0 ถึง 255 และเช่นเดียวกัน สีน้ำเงินก็ส่วน จาก 0 ถึง 255 สีที่เกิดขึ้นก็จะเกิดจากการผสมของสีทั้งสาม ในอัตราส่วนต่างๆ กัน เช่น สีเหลืองธรรมชาติ เกิดจากการผสมสี แดง 255 ส่วน สีเขียว 255 ส่วน และสีน้ำเงิน 0 ส่วน

จากข้างต้น ระดับของสีแดง มีถึง 256 ระดับ สีเขียว 256 ระดับ สีน้ำเงิน 256 ระดับ ดังนั้น RGB ทั้งหมดใช้ 24 บิต (8+8+8) ในการแสดงสี RGB ของหนึ่งจุดพิกเซลซึ่งสามารถแสดงสีได้มากถึง  $256 \times 256 \times 256 = 16.7$  ล้านสี ตารางที่ 2.1 แสดงถึงจำนวนสีที่เป็นไปได้ ซึ่งขึ้นกับจำนวนบิตที่ใช้กำหนดระดับของสี และรูปที่ 2.2 แสดงการผสมสีของแม่สีในระบบ RGB

จำนวนบิตที่ใช้เก็บสีต่อหนึ่งจุด	Color Mode Name	จำนวนสีที่แสดงได้
1	Black and White	2
4	16-Color(EGA)	16
8	Pseudo Color	256
16	Hi-Color	65,536
24	True-Color	16,777,216

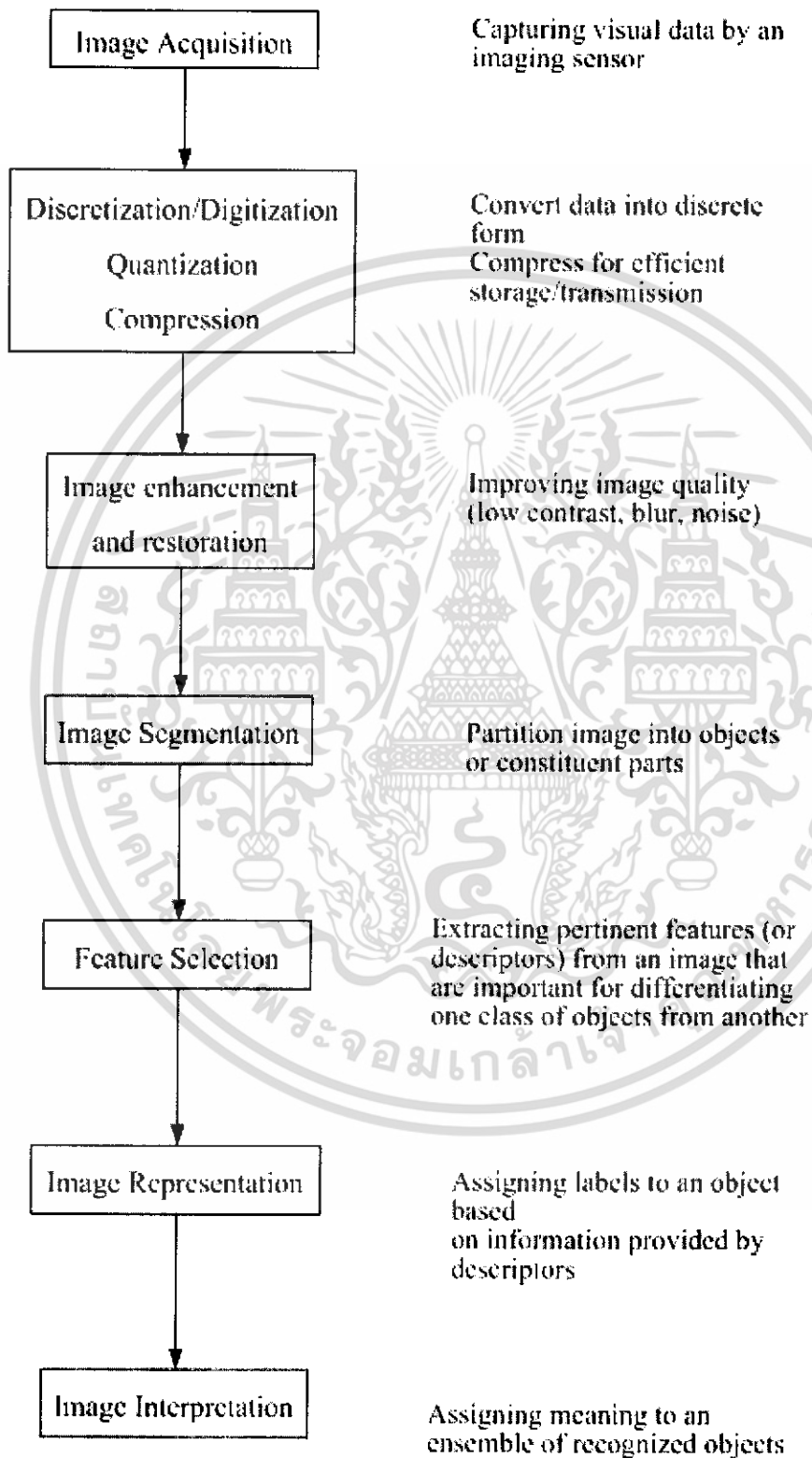
ตารางที่ 2.1 แสดงการกำหนดระดับสี



รูปที่ 2.2 แสดงการผสมสีในรูปแบบ RGB

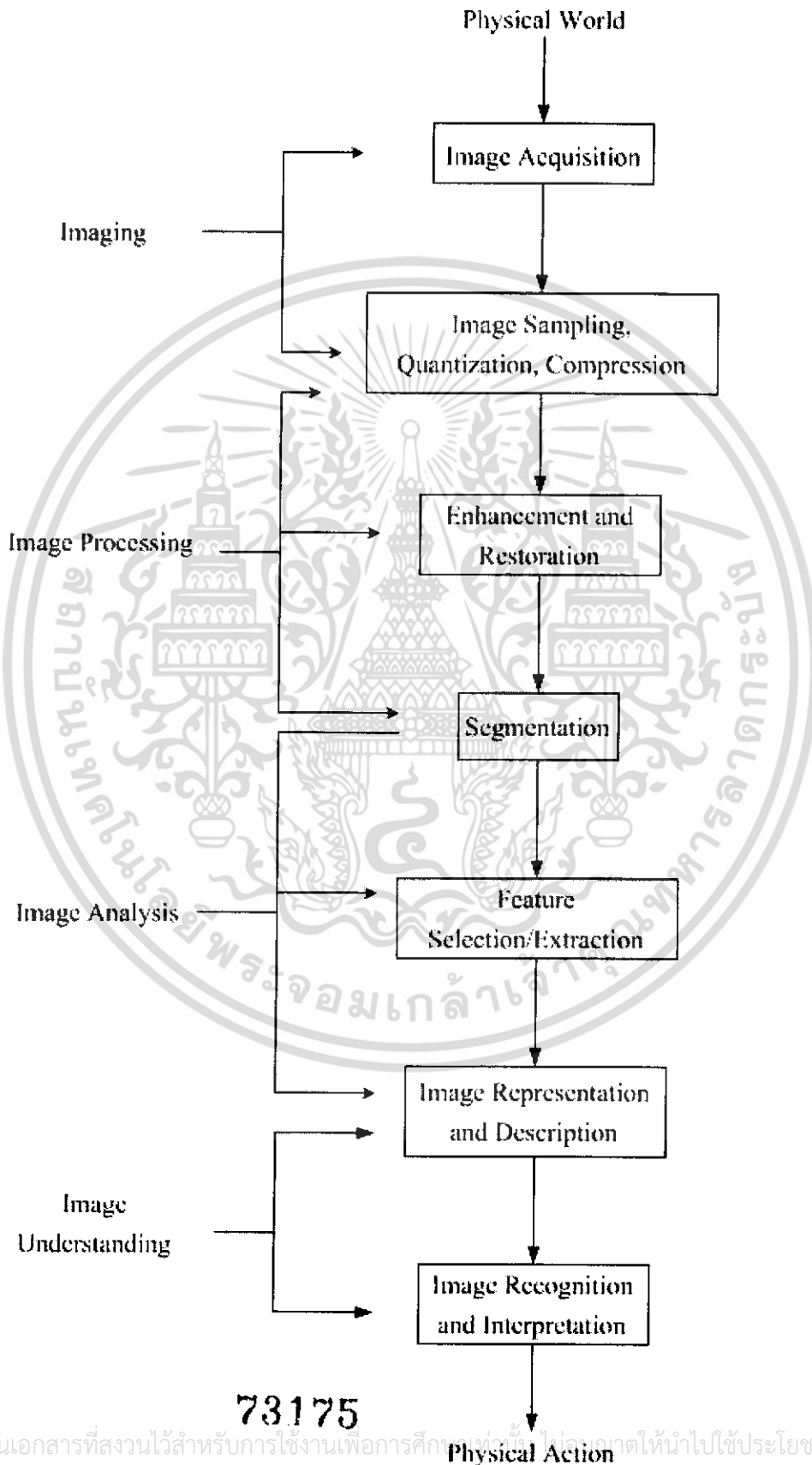
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ขั้นตอนในการประมวลผลทางภาพ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Important steps



73175

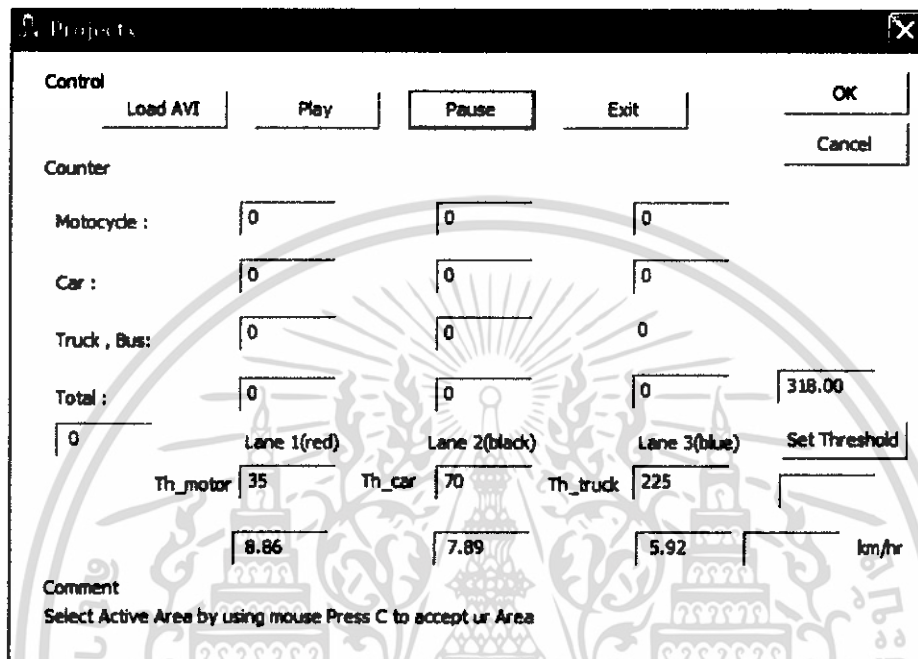
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่สามารถให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 3

### หลักการทำงานและการออกแบบ

#### 3.1 หลักการทำงานของโปรแกรม

##### หน้าต่างของโปรแกรมหลัก



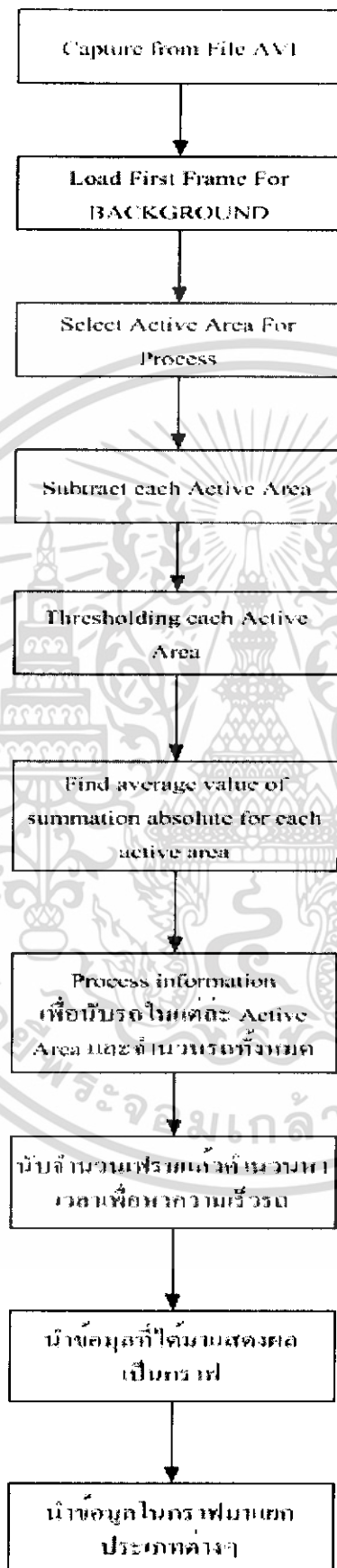
รูปที่ 3.1

- 3.1.1 **Load AVI** เป็นปุ่มที่ใช้สำหรับ โหลด ไฟล์ภาพนามสกุล avi เข้ามาในหน่วยความจำ
- 3.1.2 เมื่อคลิกปุ่ม Load AVI จะ ได้ภาพวิดีโอเฟรมแรกขึ้นมาเพื่อให้เราเลือก Active area
- 3.1.3 **Play** ปุ่มนี้ใช้สำหรับเริ่มประมวลผล เมื่อคลิกปุ่มนี้ ภาพวิดีโอจะเริ่มเล่น และการประมวลผลต่างๆก็จะเริ่มขึ้น
- 3.1.4 **Pause** ปุ่มนี้ใช้สำหรับหยุดภาพวิดีโอ หรือหยุดการประมวลผลชั่วคราว ถ้าต้องการเริ่มประมวลผลต่อให้คลิกปุ่ม Play เพื่อเริ่มประมวลผลใหม่อีกครั้ง
- 3.1.5 **Set Threshold** ปุ่มนี้ใช้สำหรับตั้งค่า Threshold สำหรับแยกประเภทยานพาหนะ
- 3.1.6 **Exit** ปุ่มนี้ใช้สำหรับต้องการเลิกหรือจบการทำงานของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2 การออกแบบระบบ

#### 3.2.1 โครงสร้างโดยรวมของโปรแกรม



รูปที่ 3.3 บล็อกไดอะแกรมแสดง  
โครงสร้างของโปรแกรม

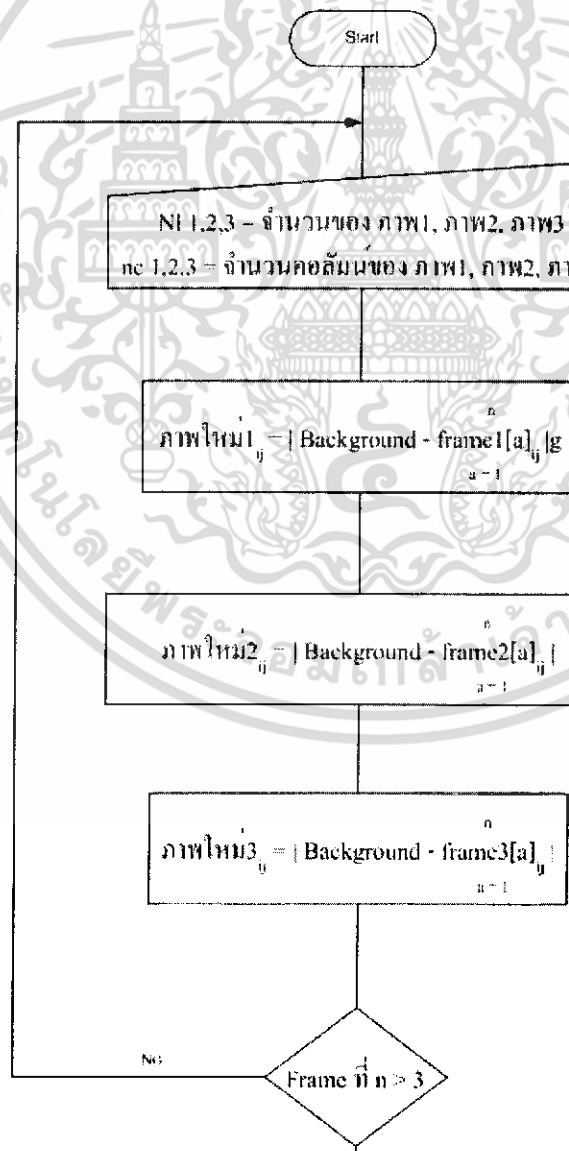
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 Capture From File AVI : เป็นการ โหลดไฟล์ภาพวิดีโอ avi เข้ามาไว้ในหน่วยความจำเพื่อ รอกการประมวลผลต่อไป

3.2.3 Load First Frame For Background : เป็นการ โหลดภาพเฟรมแรกของไฟล์วิดีโอเก็บเอาไว้ เป็นภาพ background

3.2.4 Select Active Area : เราจะทำการกำหนดขอบเขตพื้นที่ที่เราจะนำมาประมวลผล โดยการ สร้างกรอบสี่เหลี่ยมขึ้นมาในแต่ละเลนของถนนเพราะว่าถ้าประมวลผลภาพวิดีโอทั้งภาพจะเป็นการใช้ ทรัพยากรของเครื่องที่มากจนบางครั้งอาจทำให้เครื่องเกิดการ hang ได้ ซึ่งภาพใหม่ที่ได้ในแต่ละเลน จะใช้เป็นภาพ background ในนามสกุล .bmp

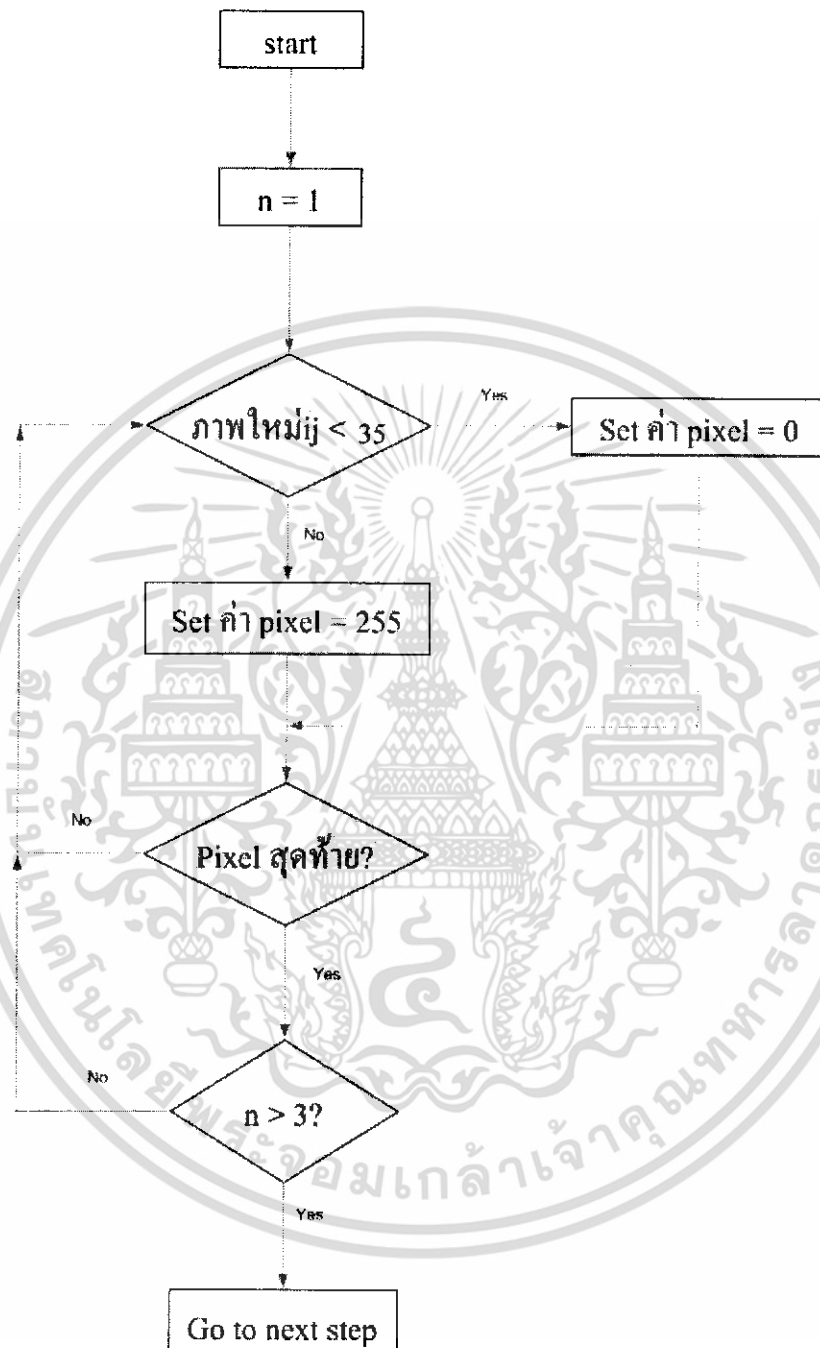
3.2.5 Subtract each Active Area : เป็นการนำภาพในแต่ละเลนที่เราได้ทำการเลือกไว้มาลบกับ ภาพ background ของแต่ละเลนซึ่งเป็นการลบกันแบบจุดต่อจุดของ pixel แล้วนำค่าที่ได้มาใส่ absolute แล้วนำไปสร้างเป็นภาพใหม่ ซึ่งเป็นภาพผลลัพธ์ที่ได้จากการลบกันของภาพ background กับ ภาพในแต่ละ เฟรมที่เรา queue มาจากไฟล์วิดีโอ



รูปที่ 3.4 บล็อกโคอะกแกรมของ Subtract each Active Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

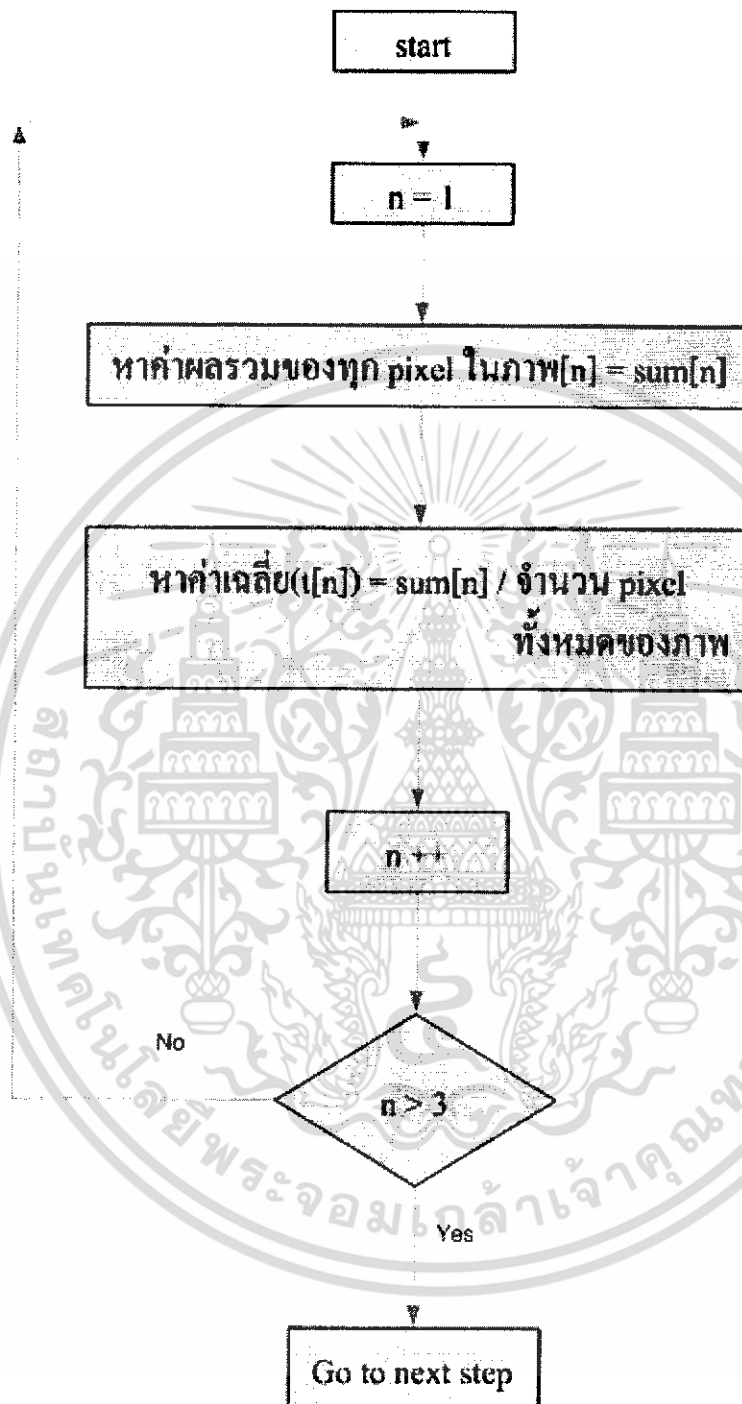
3.2.6 Thresholding each Active Area : เป็นการทำให้ Threshold ในแต่ละ active area ที่เรานำไปใช้ เพื่อให้ได้ค่าที่ได้จากการลบมีความแน่นอนขึ้น(เป็นการกำจัด noise ที่จะเกิดจากการลบภาพ)



รูปที่ 3.5 บล็อกไดอะแกรมของ Thresholding each Active Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

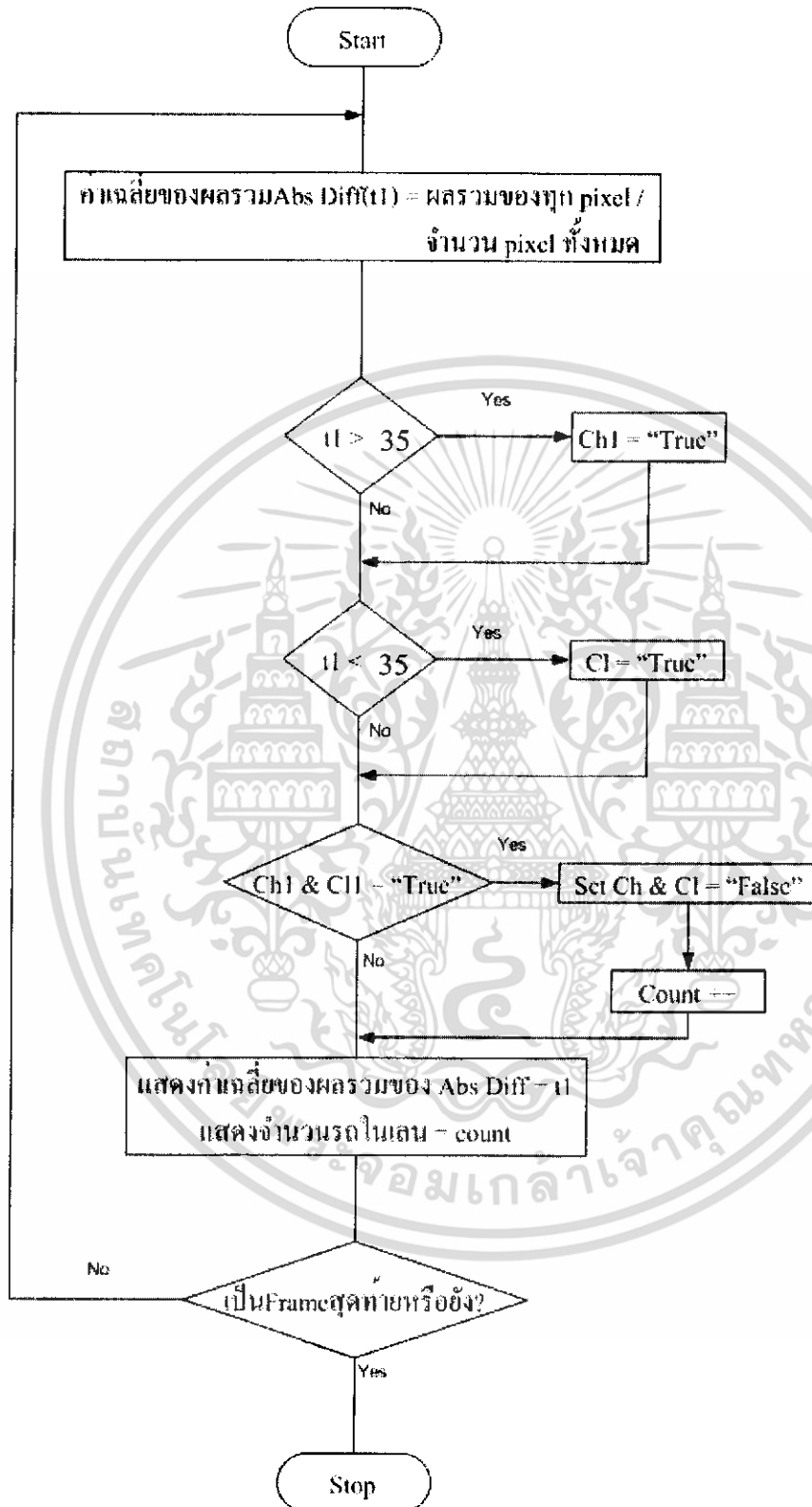
3.2.7 Find Average value of Summation Absolute for each Active Area : ทำการหาค่าเฉลี่ยของผลรวมที่ได้จากการลบภาพในแต่ละเฟรม



รูปที่ 3.6 บล็อกไดอะแกรมของ Find Average value of Summation Absolute for each Active Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.8 Process Information เพื่อนับรถในแต่ละ Active Area และจำนวนรถทั้งหมด : เป็นขั้นตอนที่นำค่าเฉลี่ยของผลรวมที่ได้จากการลบภาพมาวิเคราะห์ว่าค่าที่ได้มากแค่ไหนเราถึงจะเชื่อว่ามียาน

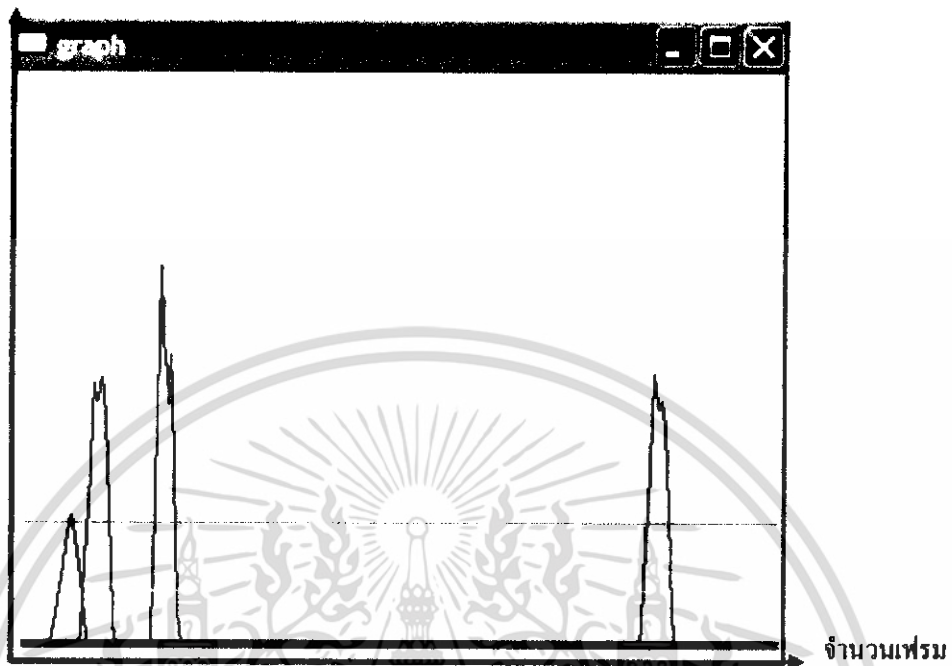


รูปที่ 3.7 บล็อกไดอะแกรมของ Process Information

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.9 Plot Graph : เป็นการนำค่าเฉลี่ยของผลรวมที่ได้จากการลบภาพในแต่ละเฟรมมาพลอตกราฟเพื่อแสดงให้เห็นภาพการเพิ่มขึ้นและลดลงของค่าเฉลี่ย

ค่า Threshold



รูปที่ 3.8 กราฟแสดงผลของโปรแกรม

3.2.10 วัดความเร็วรถ : เป็นการคำนวณความเร็วของรถที่วิ่งผ่าน Active Area ซึ่งเราจะทำการ Detect รถขณะเริ่มเข้ามาที่ขอบบน Active Area แล้วทำการบันทึกเฟรมนั้นว่าเป็นเฟรมที่เท่าไร และบันทึกเฟรมอีกทีเมื่อน้ำรถวิ่งมาแตะที่ขอบล่างของ Active Area จากนั้นนำเฟรมทั้งสองมาลบกัน จะได้จำนวนผลต่างของเฟรม จากการประมวลผลของโปรแกรมนี้อาจทำการ Queue ภาพมาในอัตรา 25 เฟรมต่อวินาที(25 fps) จึงนำมา 25 มาหารกับจำนวนผลต่างของเฟรมที่ได้ ก็จะได้เวลาที่รถใช้ในการเคลื่อนที่จากขอบบน ถึงขอบล่างของ Active Area มา ซึ่งการทดลองในครั้งนี้เราได้ทำการไปวัดระยะทางของถนนจริงจากเส้นปะสองอันได้ระยะทาง 5 เมตร ซึ่ง Active Area นี้ต้องกำหนดให้มีขนาดเท่ากับเส้นปะสองอันด้วยและจากสูตรการหาความเร็ว

$$V = S / T$$

V : ความเร็ว (เมตรต่อวินาที)

S : ระยะทาง(เมตร) ที่หน้ารถเคลื่อนที่จากขอบบน ถึงขอบล่างของActive Area

T : เวลา(วินาที) ที่หน้ารถเคลื่อนที่จากขอบบน ถึงขอบล่างของActive Area

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

### ผลการทดลอง

#### 4 ผลการทดลองของโครงการ

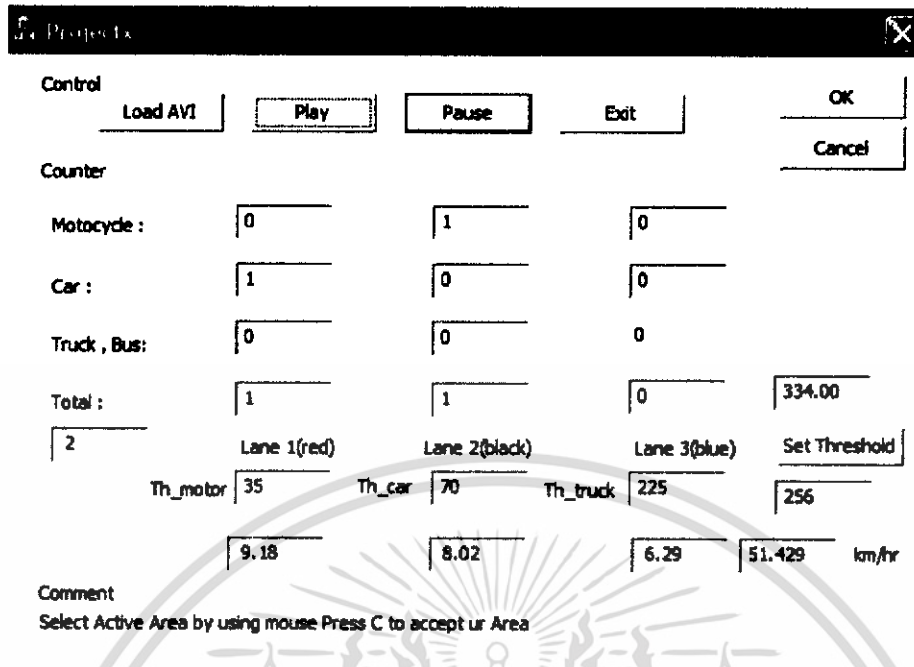
ในบทนี้ก็จะกล่าวถึงผลการทดลองของการนับจำนวนวัตถุ โดยแบ่งเป็น 7 หัวข้อคือ

- 4.1 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนซ้าย โดยรถ CAR
- 4.2 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนกลาง โดยรถ PICK UP
- 4.3 การนับแยกจำนวนยานพาหนะในช่องถนนทางเลนขวา โดยรถ MOTOCYCLE
- 4.4 การนับแยกจำนวนยานพาหนะในช่องถนนจำนวน 2 เลนพร้อมกัน โดยรถ CAR
- 4.5 การนับแยกจำนวนยานพาหนะในช่องถนนจำนวน 3 เลนพร้อมกัน โดยรถ CAR,VAN  
วิ่งในช่องทางซ้ายกับกลาง และ MOTOCYCLE ในช่องทางขวา
- 4.6 การนับแยกจำนวนยานพาหนะของรถบรรทุก (TRUCK)
- 4.7 ผลการทดลองที่เกิด Error ขึ้น
- 4.8 การทดลองวัดความเร็วของยานพาหนะในช่องทางเลนกลาง

#### 4.1 ผลการทดลอง การนับจำนวนยานพาหนะในช่องถนนทางเลนซ้าย โดยรถ CAR



รูปที่ 4.1 ภาพขณะที่ยานพาหนะวิ่งผ่านช่องถนนทางเลนซ้าย



รูปที่ 4.2 ภาพแสดงผลของโปรแกรม

### รูปภาพแสดงการประมวลผลโดยการทำ Threshold

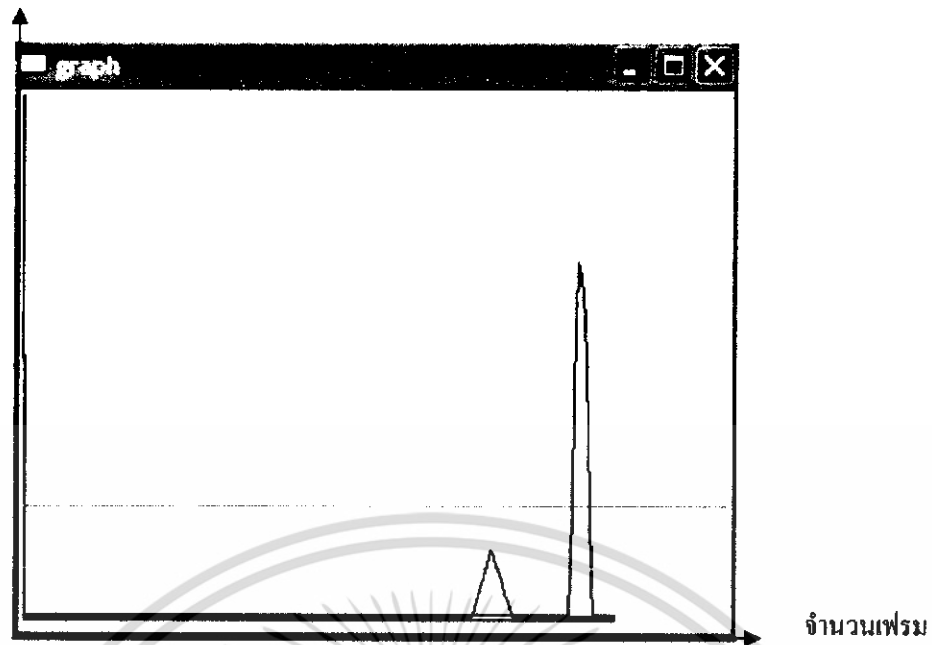


(a)

(b)

รูปที่ 4.3 (a) รูปภาพขณะที่ยานพาหนะเคลื่อนที่ผ่านช่องเลนซ้าย  
(b) รูปภาพที่ผ่านการลบและทำเทรชโซลแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.4 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

#### 4.2 ผลการทดลอง การนับจำนวนยานพาหนะในช่องถนนทางเลนกลาง โดยรถ PICK UP



รูปที่ 4.5 ภาพขณะที่ยานพาหนะวิ่งผ่านช่องถนนทางเลนกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Properties

Control

Load AVI   Play   Pause   Exit   OK

Cancel

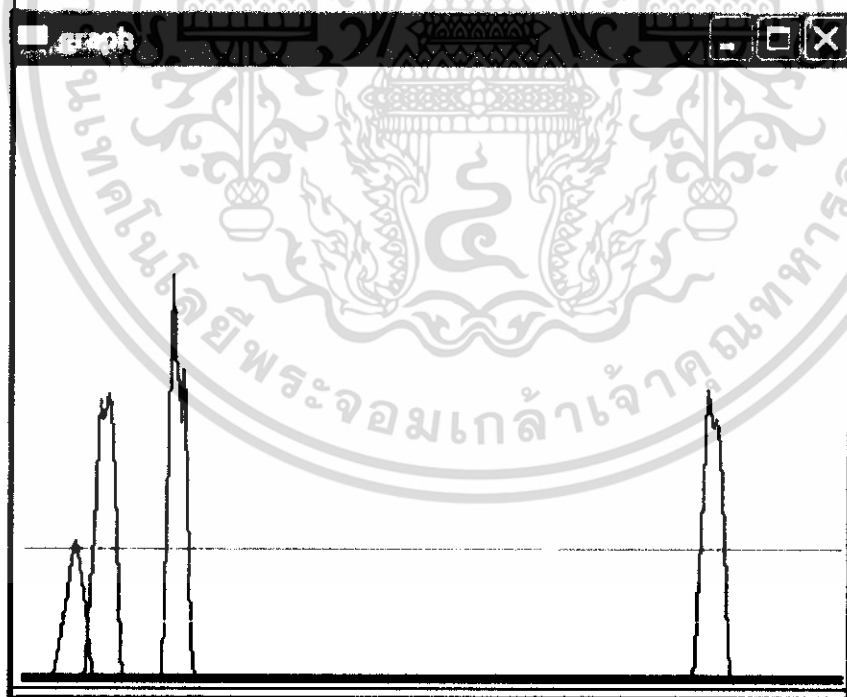
Counter

Motocycle :	0	2	1				
Car :	2	2	1				
Truck , Bus :	0	0	0				
Total :	2	4	2	3161.00			
8	Lane 1(red)	Lane 2(black)	Lane 3(blue)	Set Threshold			
Th_motor	35	Th_car	70	Th_truck	225		
	9.18		8.02		6.29	63.529	km/hr

Comment  
Select Active Area by using mouse Press C to accept ur Area

รูปที่ 4.6 ภาพภาพแสดงผลของโปรแกรม

ค่า Threshold



จำนวนเฟรม

รูปที่ 4.7 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 ผลการทดลอง การนับจำนวนยานพาหนะในช่องถนนทางเลนขวา โดยรถ MOTO CYCLE



รูปที่ 4.8 ภาพขณะที่ยานพาหนะวิ่งผ่านช่องถนนทางเลนขวา

Control: Load AVI, Play, Pause, Exit, OK, Cancel

Counter

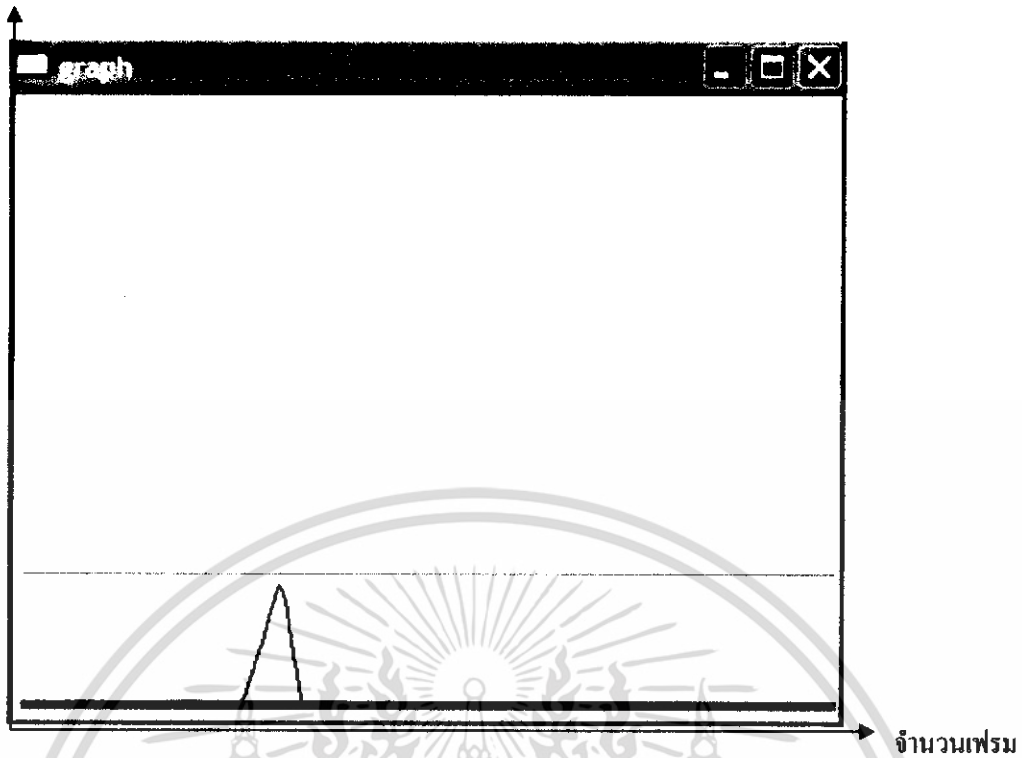
Motocycle :	0	2	1			
Car :	1	0	0			
Truck , Bus:	0	0	0			
Total :	1	2	1	559.00		
4	Lane 1(red)	Lane 2(black)	Lane 3(blue)	Set Threshold		
Th_motor	35	Th_car	70	Th_truck	225	371
	8.41		8.12		5.78	56.842 km/hr

Comment  
Select Active Area by using mouse Press C to accept ur Area

รูปที่ 4.9 ภาพแสดงผลของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า Threshold



รูปที่ 4.10 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

#### 4.4 ผลการทดลอง การนับจำนวนยานพาหนะในช่องถนนที่มีวัตถุวิ่งเข้ามาพร้อมกัน 2 เลน



รูปที่ 4.11 ภาพขณะที่ยานพาหนะวิ่งผ่านช่องถนน 2 ช่องทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5 Projects

Control

Load AVI Play Pause Exit OK Cancel

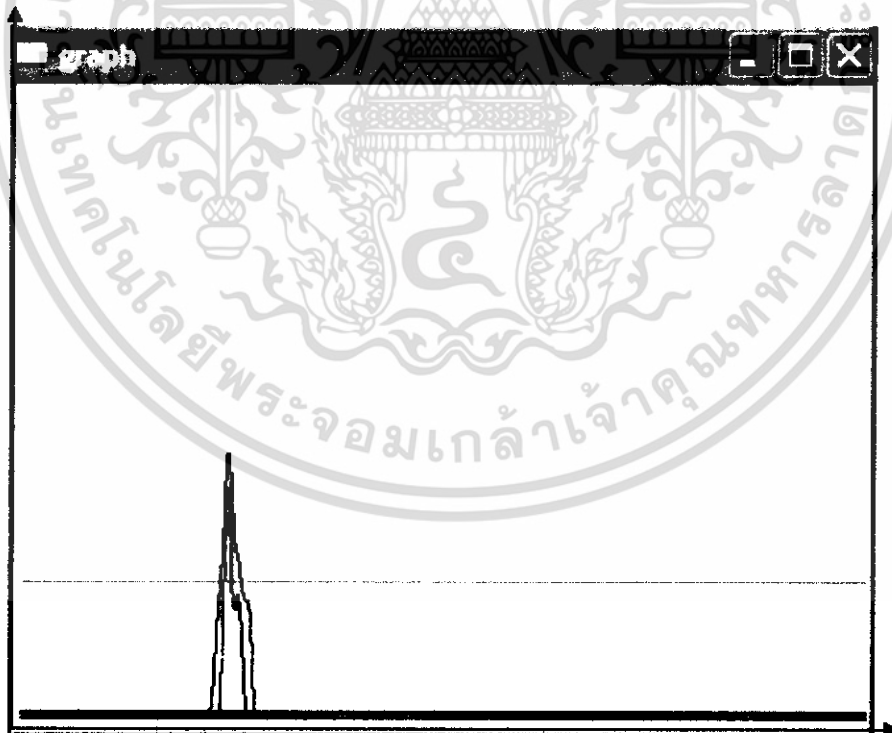
Counter

Motocycle :	0	7	3		
Car :	24	21	9		
Truck , Bus:	0	1	0		
Total :	24	29	12	14552.00	
65	Lane 1(red)	Lane 2(black)	Lane 3(blue)	Set Threshold	
Th_motor	35	Th_car	70	Th_truck	225
	9.18	8.10	6.29	54.000 km/hr	

Comment  
Select Active Area by using mouse Press C to accept ur Area

รูปที่ 4.12 ภาพแสดงผลของโปรแกรม

ค่า Threshold



จำนวนเฟรม

รูปที่ 4.13 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5 การนับจำนวนยานพาหนะในช่องถนนจำนวน 3 เลนพร้อมกัน โดยรถ CAR,VAN วิ่งในช่องทางซ้ายกึ่งกลาง และ MOTOCYCLE ในช่องทางขวา



รูปที่ 4.14 ภาพขณะที่ยานพาหนะวิ่งผ่านช่องถนน 3 lanes

Control: Load AVI, Play, Pause, Exit, OK, Cancel

Counter

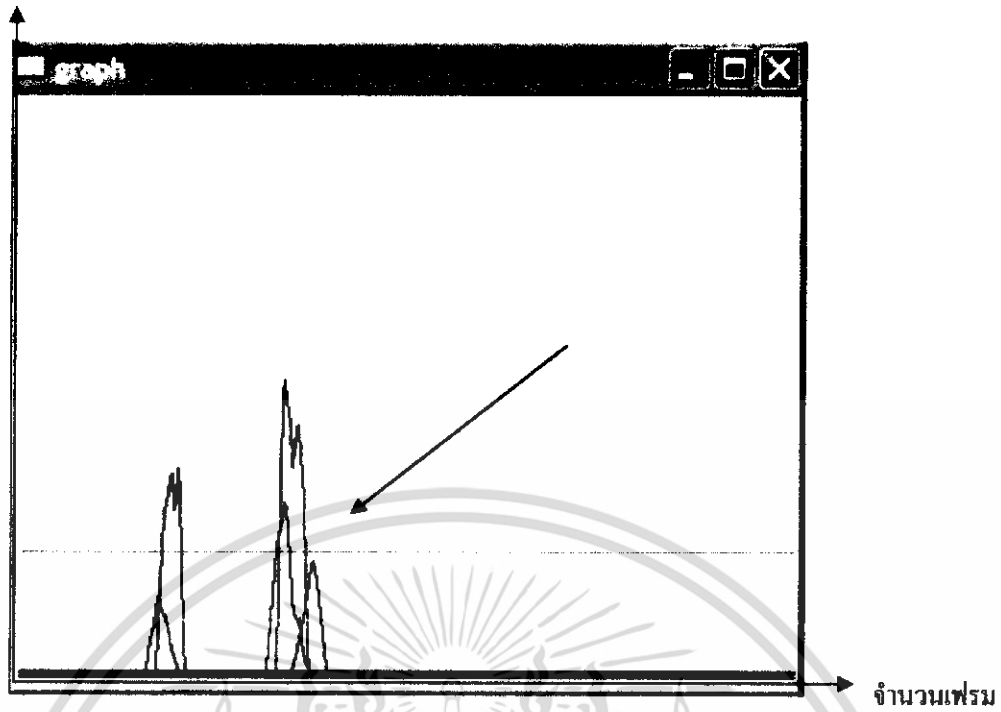
Motocycle :	0	3	3		
Car :	6	6	2		
Truck , Bus:	0	1	0		
Total :	6	10	5	7787.00	
21	Lane 1(red)	Lane 2(black)	Lane 3(blue)	Set Threshold	
Th_motor	35	Th_car	70	Th_truck	225
	9.18		8.02		6.29
				54.000	km/hr

Comment  
Select Active Area by using mouse Press C to accept ur Area

รูปที่ 4.15 ภาพแสดงผลของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่า Threshold



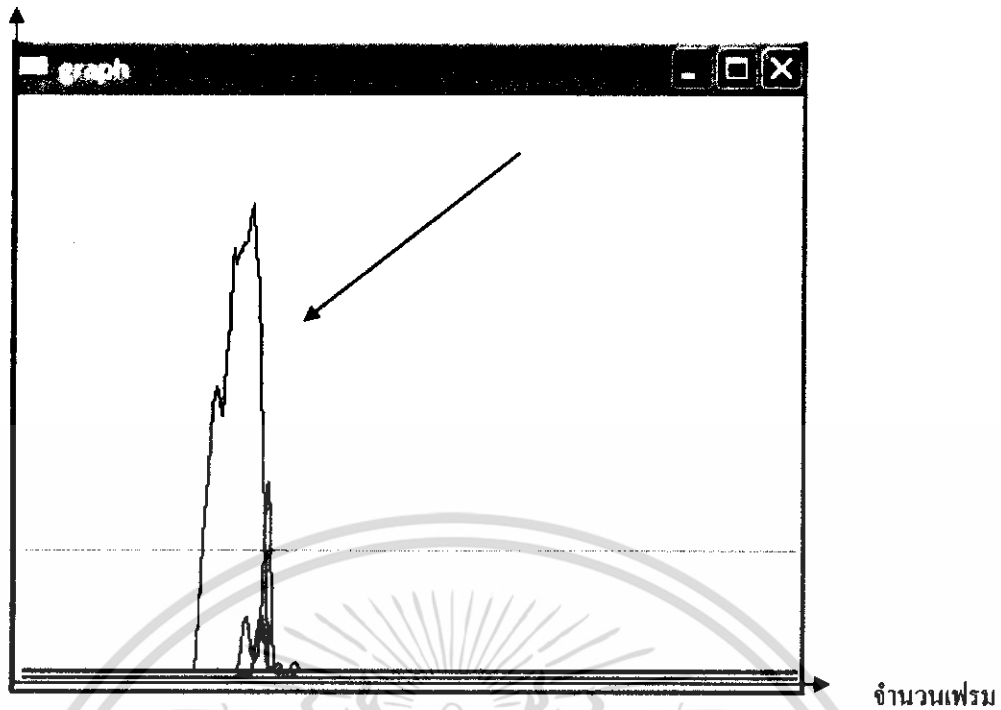
รูปที่ 4.16 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

#### 4.6 การนับจำนวนยานพาหนะของรถบรรทุก (TRUCK)



รูปที่ 4.17 การนับจำนวนยานพาหนะของรถบรรทุก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.18 กราฟแสดงผลขณะมียานพาหนะวิ่งเข้าสู่ ACTIVE AREA

Control						
Load AVI	Play	Pause	Exit	OK		
Counter						
Motocycle :	1	2	2			
Car :	3	4	2			
Truck , Bus:	0	1	0			
Total :	4	7	4	1813.00		
15	Lane 1(red)	Lane 2(black)	Lane 3(blue)	Set Threshold		
Th_motor	35	Th_car	70	Th_truck	225	1694
	9.52		8.57		5.92	29.189 km/hr
Comment						
Select Active Area by using mouse Press C to accept ur Area						

รูปที่ 4.19 ภาพแสดงผลของโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อโปรแกรมทำงานสิ้นสุดจะได้ผลดังรูป

The screenshot shows a software window titled 'Projects' with the following data:

Control				
Load AVI	Play	Pause	Exit	OK
				Cancel
Counter				
Motocycle :	1	8	6	
Car :	24	21	6	
Truck , Bus:	0	1	0	
Total :	25	30	12	6050.00
67	Lane 1(red)		Lane 2(black)	Lane 3(blue)
Th_motor	35	Th_car	70	Th_truck
	7.97	10.74	6.14	61.364 km/hr
Set Threshold				
5771				
Comment				
Select Active Area by using mouse Press C to accept ur Area				

รูปที่ 4.20 ภาพที่แสดงผลขณะทำการนับวัตถุเสร็จสิ้น

จากการทดลอง ได้ทำการตั้งค่าเฉลี่ยของผลรวมของAbsDiff(รูป Background - รูปที่วัตถุเคลื่อนที่เข้ามาใน Active Area) ไว้ที่ ค่า 35 ถ้าหากว่ามียานพาหนะวิ่งผ่านเข้ามาดังรูปที่ 4.14 คือ วัตถุวิ่งเข้ามาพร้อมกัน 2 ช่องทาง จะเห็นว่าช่องทางตรงกลางและด้านขวา ค่าของผลรวมเฉลี่ยที่นำรูปภาพมาลบกันจะมีค่า เป็น 78.17 กับ 40.19 ตามลำดับ ซึ่งค่าเฉลี่ยที่นำรูปภาพมาลบกันมีค่าเกิน 35 แล้วลดลงต่ำกว่า 12 ก็จะทำให้การ Count วัตถุ ส่วนทางเลนซ้าย จะเห็นว่าวัตถุกำลังเคลื่อนที่ออกนอก Active Area ค่าเฉลี่ยที่นำรูปภาพมาลบกันก็จะมีค่าลดลง

ส่วนผลการทดลองที่ 4.1 – 4.6 ก็ใช้หลักการเดียวกับที่ได้กล่าวไว้ข้างต้น

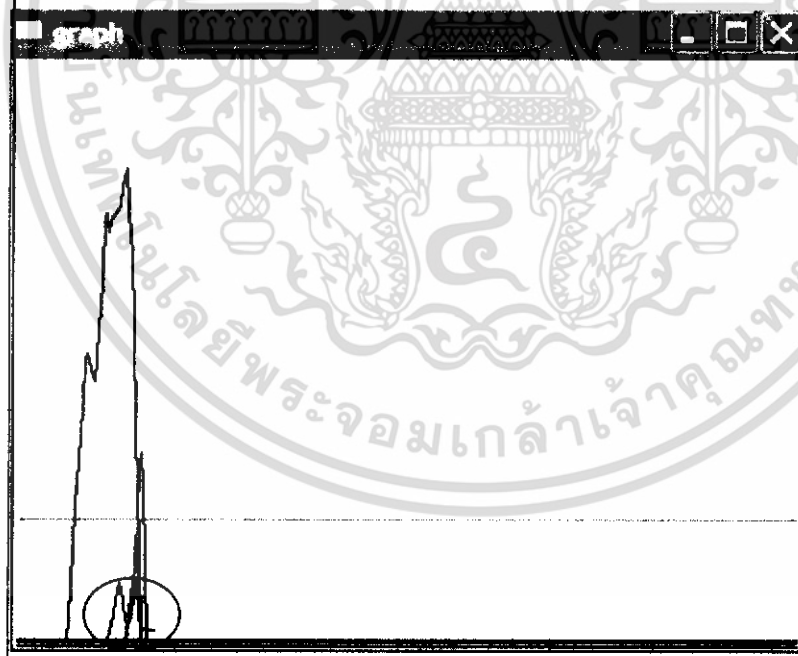
#### 4.7 ผลการทดลองที่เกิด error ขึ้น

จากรูป 4.20 จะเห็นว่าสามารถนับจำนวนวัตถุที่ผ่านเข้ามาใน Active Area ได้ 67 จำนวน ซึ่งในความเป็นจริงแล้วต้องจับวัตถุได้ 69 จำนวน ซึ่งข้อผิดพลาดที่เกิดขึ้นมานี้เกิดจากสาเหตุ ภาพวิดีโอที่ถ่ายเป็นภาพ Perspective ทำให้การตีกรอบ Active Area กับ เลนของถนนไม่ขนานกัน



รูปที่ 4.21 ภาพ Perspective

ค่า Threshold



จำนวนเฟรม

รูปที่ 4.22 กราฟแสดงให้เห็นข้อคิดในการนับจำนวนยานพาหนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 4.22 จะเห็นว่าวัตถุวิ่งเข้ามา 1 จำนวน แต่ผ่าน Active Area ที่ได้ทำการสร้างไว้ถึง 3 ช่องทาง และจากรูปที่ 4.22 ค่าเฉลี่ยที่นำรูปมาลบกัน จะเห็นว่าเกินค่า 35 อยู่ 2 ช่องทางคือ lane ซ้าย กับ lane กลาง ดังนั้นวัตถุ 1 จำนวนจะถูกการ count ไป 2 จำนวน ส่วน lane ขวา จะมีค่าเฉลี่ยอยู่ที่ 27.04 จึงไม่เกิดการ count เกิดขึ้น ซึ่งจากการเฝ้าสังเกตจะขอสรุปข้อผิดพลาดไว้ดังนี้

- ภาพที่นำมาทดลองเป็นภาพ perspective
- พาหนะเกิดวิ่งคร่อม Lane ขึ้น
- พาหนะวิ่งมาติดๆ กัน ค่าเฉลี่ยที่เกิดขึ้นยังเกิน ค่า 35 อยู่

#### 4.8 การหา % error ของโปรแกรม

##### ตารางที่ 1 ตารางแสดงผลของการทดลองหลังโปรแกรมทำงานสิ้นสุดลง

	L	M	R
<b>Motorcycle</b>	<b>1</b>	<b>8</b>	<b>6</b>
<b>Car</b>	<b>24</b>	<b>21</b>	<b>6</b>
<b>Truck</b>	<b>0</b>	<b>1</b>	<b>0</b>
<b>Total</b>	<b>25</b>	<b>30</b>	<b>12</b>

##### ตารางที่ 2 ตารางแสดงผลของยานพาหนะที่แท้จริงซึ่งไม่ได้ใช้โปรแกรมในการนับ

	L	M	R
<b>Motorcycle</b>	<b>0</b>	<b>7</b>	<b>12</b>
<b>Car</b>	<b>24</b>	<b>21</b>	<b>3</b>
<b>Truck</b>	<b>0</b>	<b>2</b>	<b>0</b>
<b>Total</b>	<b>24</b>	<b>30</b>	<b>15</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากตารางที่ 1 และ 2 จะเห็นว่าโปรแกรมมีการทำการประมวลที่เกิด Error ขึ้น ซึ่งข้อผิดพลาดนี้สามารถคำนวณเป็น % Error ได้ดังนี้

**ตารางที่ 3 ตารางแสดงผล % Error ของโปรแกรม**

	L	M	R
Motorcycle	100%	12.5%	50%
Car	0%	0%	50%
Truck	0%	100%	0%
Total	4.16%	0%	20%

**4.9 การทดสอบวัดความเร็วของยานพาหนะในช่องทางเลนกลาง**



รูปที่ 4.23 ภาพยานพาหนะวิ่งเข้ามาในเลนกลางซึ่งสามารถวัดค่าความเร็วได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

The screenshot shows a software window titled "Project" with the following elements:

- Control:** Buttons for "Load AVI", "Play", "Pause", "Exit", "OK", and "Cancel".
- Counter:** A table of counts for three lanes (red, black, blue).
 

	Lane 1(red)	Lane 2(black)	Lane 3(blue)
Motocycle :	0	6	0
Car :	16	12	9
Truck , Bus:	0	1	0
Total :	16	19	9
	44		
- Thresholds (Th):**
  - Th\_motor: 35
  - Th\_car: 70
  - Th\_truck: 225
- Speed Measurement:** A value of 67.500 km/hr is displayed and circled in red.
- Comment:** "Select Active Area by using mouse Press C to accept ur Area"

รูปที่ 4.24 ภาพแสดงผลของ โปรแกรมซึ่งจะแสดงผลของความเร็วรถขณะผ่านเลนกลาง

### ขั้นตอนการวัดความเร็ว

1. ทำการหาระยะทางจริงของพื้นถนน
2. สร้างเส้นขอบบนและขอบล่างเพื่อไว้ตรวจสอบการวิ่งเข้า-ออก ของยานพาหนะ
3. เมื่อมียานพาหนะวิ่งเข้ามาในขอบบนก็จะทำการบันทึกเฟรมต้นขณะนั้นไว้เมื่อหน้ารถยานพาหนะวิ่งออกไปถึงขอบล่างก็จะทำการบันทึกเฟรมท้ายไว้
4. นำเฟรมท้าย ลบ กับเฟรมต้น จะ ได้ค่าเฟรมทั้งหมดที่ยานพาหนะวิ่งผ่าน จากนั้นนำ 25 เฟรมมาหาร เพราะ โปรแกรมนี้ทำการ process ในอัตรา 25 frame/s
5. เข้าสูตร  $v = s/t$  ก็จะ ได้ความเร็วออกมาในหน่วย m/s จากนั้นนำ 18/5 มาคูณเพื่อให้หน่วยเป็น km/h

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลและวิจารณ์

#### 5.1 สรุปและวิจารณ์ผลการทดลอง

จากการทดลองที่ได้ออกแบบนั้นเป็นการสร้าง โปรแกรมเพื่อทำการนับจำนวนและแยกประเภทของยานพาหนะ ที่เกิดจากการใช้ทฤษฎีของ การแยกวัตถุจากภาพ(Segmentation), เทคนิคการทำเทรชโฮล (Thresholding Technique) ซึ่งจากการทดลองโปรแกรมสามารถนับจำนวนยานพาหนะได้ 67 จำนวน แต่ ในความเป็นจริงแล้วต้องนับจำนวนยานพาหนะได้ 69 จำนวน ข้อผิดพลาดที่เกิดขึ้นนี้มาจากสาเหตุ

- ภาพเกิด Perspective,
- ยานพาหนะวิ่งคร่อมLaneขณะผ่านบน Active Area,
- ยานพาหนะวิ่งติคๆ กันเข้ามาใน Active Area
- แสงของระบบสิ่งแวดล้อม

ซึ่งจากการทดลองจะได้ค่า Motorcycle = 15 คัน , Car = 51 คัน , Truck = 1 คัน แต่ในความเป็นจริงแล้วต้องได้ค่า Motorcycle = 19 คัน , Car = 48 คัน , Truck = 2 คัน

#### 5.2 ข้อดีและข้อเสีย

จากการดำเนินงานทำให้ทราบถึงข้อดีของโปรแกรม คือ สามารถทำการนับจำนวนยานพาหนะได้, รู้ว่าแต่ละLane มีจำนวนยานพาหนะวิ่งเข้ามาเป็นจำนวนเท่าไร, และเมื่อมีจำนวนยานพาหนะวิ่งเข้ามาใน Active Area ก็จะแสดงให้เห็นรูปค่าเฉลี่ยของผลรวมของ Absolute different, สามารถแยกประเภทของยานพาหนะได้, สามารถ Set Threshold ได้, สามารถวัดความเร็วของยานพาหนะได้

ส่วนข้อเสียของโปรแกรมคือ โปรแกรมไม่ยืดหยุ่นพอ ถ้ามีจำนวนยานพาหนะที่วิ่งคร่อม Lane เข้ามาก็จะเกิดการนับผิดพลาดขึ้น และ ไม่สามารถนับจำนวนยานพาหนะที่วิ่งติคๆ กันเข้ามาได้ ถ้าตั้งค่า Set Threshold ไม่ยืดหยุ่นพอก็จะทำให้การทำงานของโปรแกรมเกิดข้อผิดพลาดมากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 5.3 การนำไปประยุกต์การใช้งาน

สามารถนำโปรแกรมไปใช้จัดการกับปัญหาทางด้านจราจรที่ติดขัดได้ ซึ่งสามารถนับจำนวนยานพาหนะบนท้องถนน และทราบความหนาแน่นของยานพาหนะบนท้องถนนได้ เพื่อให้จราจรสามารถระบายจำนวนยานพาหนะบนท้องถนนได้รวดเร็วขึ้น

### 5.4 แนวทางการพัฒนา

- ทำให้โปรแกรมสามารถนับจำนวนและแยกประเภทรถได้ถูกต้องมากขึ้น
- พัฒนาโปรแกรมให้สามารถใช้งานจริงได้ทุกสภาวะ
- พัฒนา Active area ให้ยืดหยุ่นคือ ให้สามารถให้วาดรูปเป็นสี่เหลี่ยมต่างๆ ให้ครอบคลุมพื้นถนนได้
- พัฒนาให้สามารถแก้ปัญหาบนทางที่วิ่งคร่อมเลนส์
- พัฒนาให้สามารถแก้ปัญหาเรื่องเงา เพื่อให้การนับจำนวนยานพาหนะแม่นยำขึ้น

## บรรณานุกรม

- [1] “OpenCV Reference Manuals”. <http://groups.yahoo.com/group/OpenCV>
- [2] ยุทธนา ลีลาศวัฒนกุล. “คู่มือการเขียน โปรแกรมวิน โดว์ขั้นสูงด้วย Visual C++.NET Episode one”. กรุงเทพฯ: หจก.ไทยเจริญการพิมพ์, 2546.
- [3] ยุทธนา ลีลาศวัฒนกุล. “คู่มือการเขียน โปรแกรมและใช้งาน Visual C++.NET ฉบับสมบูรณ์”. กรุงเทพฯ: บริษัท คำนสุทธการพิมพ์ จำกัด, 2546.
- [4] R. C. Gonzalez and R. E. Woods. ”Digital Image Processing,Prentice Hall” New Jersey



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้