สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมวิเคราะห์อาการหลับใน

DOZE-OFF WARNING SYSTEM

T117565

วริศรา กังสวัสดิ์
สุรภัค สุขแสง
สุรศักดิ์ เหล่าประภัสสร

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เลขหมู่.....
เลขทะเบียน 117565
วัน,เดือน,ปี - 5 ส.ค. 2554

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมวิเคราะห์อาการหลับใน

DOZE-OFF WARNING SYSTEM

ผู้จัดทำ

1. นางสาววริศรา กังสวัสดิ์ รหัสนักศึกษา 50011398
2. นายสุรภัค สุขแสง รหัสนักศึกษา 50011756
3. นายสุรศักดิ์ เหล่าประภัสสร รหัสนักศึกษา 50011763

_____ อาจารย์ที่ปรึกษา

(อาจารย์อำนาจ ขาวเน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมวิเคราะห์อาการหลับใน

นางสาววริศรา กังสวัสดิ์ 50011398
นายสุรภัค สุขแสง 50011756
นายสุรศักดิ์ เหล่าประภัสสร 50011763
อาจารย์อำนาจ ขาวเน อาจารย์ที่ปรึกษา
ปีการศึกษา 2553

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อศึกษา ออกแบบ และพัฒนาโปรแกรมวิเคราะห์อาการหลับใน (Doze-off Warning System) โดยนำโครงงานนี้มาประยุกต์ใช้งานสำหรับผู้ขับขี่รถยนต์ เพื่อสามารถลดอุบัติเหตุทางถนนจากการพัฒนาระบบวิเคราะห์อาการหลับในจากดวงตา และจากการเปลี่ยนเลนถนน

ระบบวิเคราะห์อาการหลับในแบ่งออกเป็น 2 ระบบ คือ

1) ระบบวิเคราะห์อาการหลับในจากดวงตา จะมีกล้องเว็บแคมติดบริเวณด้านหน้าของผู้ขับขี่รถยนต์ เพื่อรับภาพใบหน้าของผู้ขับเข้ามา แล้วใช้หลักการประมวลผลภาพดิจิทัลมาวิเคราะห์หาบริเวณดวงตาของผู้ขับจากใบหน้าทั้งหมด เพื่อนำตาดำนั้นมาวิเคราะห์ตามหลักทางชีววิทยาว่าผู้ขับรถยนต์กำลังหลับในอยู่หรือไม่ หากกำลังหลับในอยู่ระบบจะแจ้งเตือนด้วยเสียงให้ผู้ขับได้สติขึ้นมา
2) ระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนน จะมีกล้องเว็บแคมติดบริเวณข้างตัวรถยนต์ด้านซ้ายและขวา เพื่อตรวจจับเส้นพื้นถนน หากมีการเปลี่ยนเลนที่ไม่เหมาะสมระบบจะแจ้งเตือนด้วยเสียงเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Doze-off Warning System

Ms. Varisara	Kangsawad	50011398
Mr. Surapak	Suksang	50011756
Mr. Surasak	Laoprapassorn	50011763
Mr. Amnach	Khawne	Advisor

Academic Year 2010

ABSTRACT

This project mainly focuses to study ,design and develop Doze-off Warning System so that applies for a driver and can reduce accident on the road. This project develop Doze-off Warning System by eye detection and switching lane of the road. The system divided into two parts:

1) Dozing - off Warning System has a webcam installed in front of the driver to detect the driver's face. The program will then analyze facial image, detect the eyes of the driver and analyze by the biology theory. This system can verify that the driver asleep or wake up (off - dozing). If the driver is sleeping while driving. Voice system will alert and encourage motorists to wake up.
2) Doze-off Warning System can check the driver that can sustain the car in a road lane. For detection of road lane, Doze-off Warning System was installed webcam camera for detecting road lane lines. If a sudden lane change, the program will alert the driver with voice immediately as well.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

โครงงานโปรแกรมวิเคราะห์อาการหลับใน (Doze-off Warning System) ซึ่งคณะผู้จัดทำได้ศึกษา ออกแบบ และพัฒนาโปรแกรม เพื่อนำมาประยุกต์ใช้งานสำหรับผู้ขับขี่รถยนต์ สามารถลดอุบัติเหตุทางถนน คณะผู้จัดทำได้รับความแนะนำช่วยเหลือดูแลตลอดการศึกษาเป็นอย่างดีจาก อาจารย์อำนาจ ขาวเน อาจารย์ที่ปรึกษา รวมทั้งอาจารย์วัจนพงศ์ เกษมศิริ ที่ให้ความช่วยเหลือแนะนำการทำเอกสารที่เกี่ยวกับการทำรายงานแก่คณะผู้จัดทำโครงงาน จึงขอบกราบขอบพระคุณท่าน มา ณ ที่นี้ด้วย

วริศรา กังสวัสดิ์
สุรภัค สุขแสง
สุรศักดิ์ เหล่าประภัสสร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญภาพ.....V

บทที่ 1 บทนำ.....1

1.1 ความเป็นมา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์.....1

1.3 ประโยชน์ที่คาดว่าจะได้รับ.....1

1.4 ขอบเขตของโครงงาน.....2

บทที่ 2 การประมวลผลภาพเบื้องต้น.....3

2.1 ความรู้เบื้องต้นเกี่ยวกับการประมวลผลภาพดิจิทัล.....3

2.2 ระบบสี (Color Model).....5

2.3 ฮิสโตแกรม (Histogram).....8

2.4 การประมวลผลภาพกับรูปร่างและโครงร่างของภาพ.....9

บทที่ 3 การปรับปรุงภาพและการหาขอบภาพ.....14

3.1 Hough transform.....14

3.2 Canny Edge detection Alogorithm.....22

3.3 การหาขอบภาพโดยวิธีโซเบล.....24

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 4 เครื่องมือที่ใช้ในการพัฒนา.....26

4.1 การทำงานของ OpenCV.....26

4.2 Region-of-interest (ROI).....26

4.3 Haar wavelet.....27

4.4 Haar like-feature.....28

4.5 Cascade Classifiers.....31

บทที่ 5 การออกแบบและการพัฒนา.....33

5.1 การทำงานโดยรวมของโปรแกรมวิเคราะห์อาการหลับใน.....33

5.2 การทำงานของระบบวิเคราะห์อาการหลับในจากดวงตา.....34

5.3 ขั้นตอนการสร้างไฟล์ XML สำหรับ Haar like-feature.....35

5.4 การตรวจหาดวงตาจากใบหน้าด้วย Haar like-feature.....38

5.5 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform.....38

5.6 การวิเคราะห์อาการหลับในจากการลืมตาและหลับตา.....39

5.7 การทำงานของระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลน.....39

5.8 อุปกรณ์และระบบ.....40

บทที่ 6 ผลการทดลอง.....41

6.1 การหาดวงตาจากใบหน้า.....41

6.2 การเปลี่ยนระบบสีของดวงตาจาก RGB เป็น Grayscale.....41

6.3 การทำ Thresholding ภาพดวงตา.....42

6.4 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform (Circle).....42

6.5 การตรวจหาลูกตาดำจากดวงตาด้วยวิธีอื่นๆ.....45

6.6 การปรับปรุงภาพเลนด้วย Morphological Image Processing (Erosion).....46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

6.7 การหาขอบภาพเลนด้วย Edge Detection (Canny).....46

6.8 การตรวจหาเลนถนนด้วย Hough Transform (Line).....47

6.9 การตรวจหาเลนถนนด้วยวิธีอื่นๆ.....47

บทที่ 7 สรุป.....49

7.1 สรุป.....49

7.2 ปัญหาและอุปสรรค.....49

7.3 แนวทางการแก้ไขและการพัฒนาต่อ.....49

บรรณานุกรม.....51

ภาคผนวก ก.....52

ภาคผนวก ข.....57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป หน้า

2.1 การแปลงภาพแอนะล็อกให้เป็นภาพดิจิทัล.....3
2.2 ตัวอย่างภาพแบบ Binary หรือ ภาพขาว-ดำ.....4
2.3 ตัวอย่างภาพแบบ Grayscale.....4
2.4 ตัวอย่างภาพแบบ RGB.....5
2.5 ตัวอย่างภาพแบบ Indexed.....5
2.6 ระดับสีของ Grayscale ตามขนาดข้อมูลที่เก็บค่าสี.....6
2.7 โครงสร้างระบบสี RGB 6
2.8 ความแตกต่างของสีแสง (RGB) และสีวัตถุ (CMYK).....7
2.9 วงล้อสีแบบ RGB.....7
2.10 โมเดลสี HSV ในรูปแบบโคน (Cone) 8
2.11 ตัวอย่างภาพที่เก็บในแต่ละ channel ในระบบสี RGB และ HSV.....8
2.12 กราฟฮิสโตแกรม.....9
2.13 ข้อมูลภาพ.....9
2.14 ข้อมูลภาพสำหรับการทำโอเปอเรชั่น.....10
2.15 A ยูเนียน B 10
2.16 A อินเตอร์เซคชั่น B 10
2.17 แสดงค่าแทนค่าของภาพในแต่ละพิกเซล.....10
2.18 ข้อมูลภาพและเทมเพลต.....11
2.19 ข้อมูลแถวแรก.....11
2.20 ผลการยูเนียนในตำแหน่งพิกเซลเท่ากับ 1.....11
2.21 ผลการยูเนียนในตำแหน่งพิกเซลเท่ากับ 2 12
2.22 ผลการยูเนียนทั้งภาพ.....12
2.23 ข้อมูลภาพและเทมเพลต.....12
2.24 ผลการยูเนียนทั้งภาพ.....13
2.25 ผลการยูเนียนทั้งภาพเมื่อเปลี่ยนเทมเพลต.....13
3.1 การแก้ปัญหาที่เป็นไปได้.....15

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป หน้า

3.2 คำอธิบายตัวแปรของ เส้นตรง (straight line)……15
3.3 ตัวอย่างภาพสี่เหลี่ยมสองรูป……16
3.4 ตัวอย่างภาพสี่เหลี่ยมสองรูปผ่าน Canny edge detector……17
3.5 ตัวอย่างภาพที่ผ่านการทำ Hough transform……17
3.6 ตัวอย่างภาพที่ผ่านการ Histogram equalizing……17
3.7 ตัวอย่างภาพที่ผ่านการ Mapping……18
3.8 ตัวอย่างภาพที่ salt และ pepper noise……18
3.9 ผลที่ได้จากตัวอย่างภาพที่มีsalt และ pepper noise……19
3.10 ตัวอย่างภาพที่ผ่าน De-Houghing……19
3.11 ตัวอย่างภาพที่ผ่าน Hough transform ……19
3.12 ตัวอย่างภาพ Hough representation……19
3.13 ตัวอย่างภาพ De-Houghed (ใช้ relative threshold 40%)……19
3.14 ตัวอย่างภาพเมืองที่อาคารเป็นอุปสรรคในหมอก……20
3.15 ตัวอย่างภาพขอบเมืองที่อาคารเป็นอุปสรรคในหมอก……20
3.16 Histogram equalized accumulator space ของตัวอย่างภาพต้นฉบับ……20
3.17 ตัวอย่างภาพ De-Houghed (relative threshold เป็น 70%)……21
3.18 ตัวอย่างภาพ De-Houghed (relative threshold เป็น50%)……21
3.19 ตัวอย่างภาพถนน……21
3.20 ตัวอย่างภาพขอบของภาพถนน……21
3.21 ตัวอย่างภาพที่ผ่านเครื่องตรวจจับ Hough line……22
3.22 ขั้นตอนของ Canny edge detection……22
3.23 เทมเพลตขนาด 3x3 สองเทมเพลต……24
3.24 ตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล……24
4.1 การกำหนด Region of Interest……27
4.2 ลักษณะของ Sine Wave และ Wavelet……27
4.3 ลักษณะของ Haar Wavelet……28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป หน้า

4.4 แบบของ Features สำหรับการตรวจจับลักษณะแบบต่างๆ.....28
4.5 ตัวอย่างการใช้ Feature ตรวจจับลักษณะต่างๆ.....29
4.6 การคำนวณแบบ Integral image 29
4.7 การทำงานของ Adaboost.....30
4.8 ผลลัพธ์จากการทำกระบวนการ Adaboost.....30
4.9 ตัวอย่างการค้นหาใบหน้าด้วยอัลกอริทึม Haar-Like.....31
4.10 การทำงานของ Haar Cascade Classifier.....32
4.11 ลักษณะการทำงานของ Cascade Classifier.....32
5.1 การทำงาน โดยรวมของระบบ.....33
5.2 การทำงานของระบบวิเคราะห์อาการหลับในจากการวิเคราะห์ดวงตา.....34
5.3 ตัวอย่างภาพ Positive image.....35
5.4 ตัวอย่างภาพ Negative image.....35
5.5 ตัวอย่างการสร้างไฟล์ในการ Train ของภาพ Negative.....36
5.6 ตัวอย่างการสร้างไฟล์ในการ Train ของภาพ Positive.....37
5.7 ขั้นตอนการหาลูกตาดำจากดวงตาด้วย Hough Transform..... 38
5.8 ขั้นตอนการหาเลนถนนด้วย Hough Transform.....39
6.1 การหาดวงตาจากสิ่งแวดล้อม.....41
6.2 ผลลัพธ์ของการหาดวงตา.....41
6.3 การเปลี่ยนระบบสีของดวงตาจาก RGB เป็น Grayscale.....42
6.4 การทำ Thresholding ภาพดวงตา.....42
6.5 ชุดของตาในกรณีปกติ (ตอนกลางวัน).....42
6.6 ชุดของตาในกรณีปกติ (ตอนกลางคืน).....43
6.7 ชุดของตาที่ตรวจจับไม่พบลูกตาดำ (ตอนกลางวัน).....43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูป หน้า

6.8 ชุดของตาที่ตรวจจับไม่พบลูกตาดำ (ตอนกลางคืน).....44
6.9 ชุดของตาในกรณีผิดปกติ.....44
6.10 การตรวจหาลูกตาดำดวงตาด้วยวิธีวัดปริมาณ Pixel.....45
6.11 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform (Circle) แบบระบบสี HSV.....45
6.12 การปรับปรุงภาพเลนด้วย Morphological Image Processing (Erosion).....46
6.13 การหาขอบภาพเลนด้วย Edge Detection (Canny).....46
6.14 การตรวจหาเลนถนนในเวลากลางคืน.....47
6.15 การตรวจหาเลนถนนในเวลากลางวัน.....47
6.16 การตรวจหาเลนถนนด้วย Hough Transform (Line) มุมกล้องอื่นๆ.....48
6.17 การตรวจหาเลนถนนที่เลือกค่า Thresholding ให้ฟังก์ชั่นไม่เหมาะสม.....48
ก.1 การกำหนด Root Path ในการติดตั้ง OpenCV.....53
ก.2 การกำหนด Library files ในโปรแกรม Visual Studio C++ 2008.....54
ก.3 การกำหนด Include files ในโปรแกรม Visual Studio C++ 2008.....54
ก.4 การกำหนด Source files ในโปรแกรม Visual Studio C++ 2008.....55
ก.5 การเพิ่มไฟล์ Library ที่ใช้พัฒนาในโปรแกรม Visual Studio C++ 2008.....56
ข.1 หน้าต่างโปรแกรมหลัก.....57
ข.2 หน้าต่างโปรแกรมในเมนู Video Night.....58
ข.3 หน้าต่างโปรแกรมในเมนู Video Day.....58
ข.4 หน้าต่างโปรแกรมในเมนู Video Lane.....59
ข.5 หน้าต่างโปรแกรมในเมนู Realtime.....59
ข.6 หน้าต่างโปรแกรมในเมนู Realtime Eye.....60

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

เนื่องจากอุบัติเหตุทางรถยนต์ในปัจจุบันส่วนหนึ่งมีสาเหตุมาจากปัญหาการหลับในของผู้ขับขี่ ซึ่งยังไม่มีการแก้ไขปัญหาที่ชัดเจน นอกจากการรณรงค์ให้ผู้ขับขี่ที่มีอาการง่วงนอนจอดรถยนต์ตามที่พักรถหรือข้างทางเพื่อนอนพัก นอกจากนั้นแล้วผู้ขับขี่บางคนอาจจะฝืนตัวเองในการขับรถโดยไม่จอดพักและขับรถต่อไปด้วยอาการง่วงนอนซึ่งอาจทำให้เกิดอุบัติเหตุร้ายแรงได้ทั้งตัวผู้ขับขี่เองและผู้อื่น โปรแกรมวิเคราะห์อาการหลับในนี้จะเป็นตัวช่วยหนึ่งในการแก้ไขปัญหาดังกล่าว โดยที่จะส่งเสียงเตือนทันทีที่ผู้ขับขี่มีอาการหลับในเพื่อให้ผู้ขับขี่รู้สึกตัวและมีสติในการขับรถ รวมถึงเตือนอย่างต่อเนื่อง ซึ่งผู้ขับขี่อาจจะเลือกที่จะจอดรถในจุดที่ปลอดภัยเพื่อพักให้หายง่วงได้ ซึ่งจะช่วยลดปัญหาอุบัติเหตุจากการหลับใน

1.2 ความมุ่งหมายและวัตถุประสงค์

1) สร้างและพัฒนาโปรแกรมวิเคราะห์อาการหลับใน
2) สามารถเตือนผู้ขับขี่ที่หลับในได้อย่างทันท่วงทีและมีประสิทธิภาพ
3) สามารถนำไปประยุกต์ใช้ได้จริงในชีวิตประจำวัน
4) มีประสบการณ์ในการเขียนซอฟต์แวร์โอเพนซอร์ส

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1) โปรแกรมนี้จะเป็นส่วนสำคัญประการหนึ่งที่จะช่วยเตือนสติให้โอกาสการเกิดอุบัติเหตุลดลงได้
2) ได้รับความรู้ความสามารถจากการเขียนโปรแกรมเรื่อง การประมวลผลภาพดิจิทัล (Digital Image Processing)
3) ได้รับความรู้ความสามารถการใช้ภาษา C++ และไลบรารี OpenCV ในพัฒนาโปรแกรม
4) สามารถได้นำโปรแกรมไปประยุกต์ใช้จริงในอนาคต เนื่องจากอุตสาหกรรมยานยนต์มีจุดประสงค์ที่จะส่งเสริม และให้ความสำคัญต่อการพัฒนา ปรับปรุงคุณภาพด้านความปลอดภัยของผู้ขับขี่มากขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตของโครงการ

สร้างโปรแกรมวิเคราะห์อาการหลับใน เพื่อช่วยเตือนผู้ขับขี่ที่เกิดอาการหลับในด้วยเสียง ให้รู้สึกตัวอย่างทันท่วงที โดยใช้เทคนิคการตรวจหาลูกตาดำของผู้ขับขี่แล้วนำไปประมวลผลว่ามีโอกาสหลับในอยู่หรือไม่ตามหลักชีววิทยา เมื่อพบว่าผู้ขับมีอาการหลับในอยู่ก็จะแจ้งเตือนให้ผู้ขับโดยทันที นอกจากนั้นแล้วยังมีการตรวจสอบการเปลี่ยนเลนบนถนนของรถยนต์ หากมีการเปลี่ยนเลนที่เปลี่ยนไปเร็วหรือช้าเกินไป หรือแม้แต่การเคลื่อนย้ายเข้าใกล้เส้นถนนที่เป็นเส้นที่เป็นขอบของถนนที่เป็นเส้นทึบ ระบบก็จะแจ้งเตือนด้วยเสียงอีกเช่นกัน เพื่อให้ผู้ขับขี่รถยนต์ได้สติและควบคุมรถยนต์ให้กลับเข้าสู่เส้นทางอีกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
การประมวลผลภาพเบื้องต้น

2.1 ความรู้เบื้องต้นเกี่ยวกับการประมวลผลภาพดิจิทัล

การประมวลผลภาพดิจิทัลนั้นเกี่ยวข้องกับการแปลงข้อมูลรูปที่เป็นสัญญาณแอนะล็อกให้อยู่ในรูปของสัญญาณดิจิทัล เพื่อใช้ในการประมวลผลผ่านทางคอมพิวเตอร์ได้และยังนำมาใช้ในการลดปัญหาของภาพ เช่น ลดสัญญาณรบกวนภายในภาพเป็นต้น

ในการแปลงภาพให้เป็นสัญญาณดิจิทัลนั้น ระบบจะนำรูปที่รับเข้ามาจากอุปกรณ์รับภาพ เช่น กล้อง ไปคำนวณ โดยกระบวนการ Sampling และ Quantization และส่งข้อมูลออกมาในรูปแบบดิจิทัล คอมพิวเตอร์จะเก็บข้อมูลภาพลงหน่วยความจำ โดยการจองหน่วยความจำภายในเครื่องในรูปแบบของอาร์เรย์ โดยค่าในแต่ละช่องของอาร์เรย์แสดงถึงคุณสมบัติต่างๆของรูปที่จุดพิกเซลนั้นๆ และตำแหน่งของช่อง อาร์เรย์ก็เป็นตัวกำหนดตำแหน่งของจุดพิกเซลภายในภาพด้วย

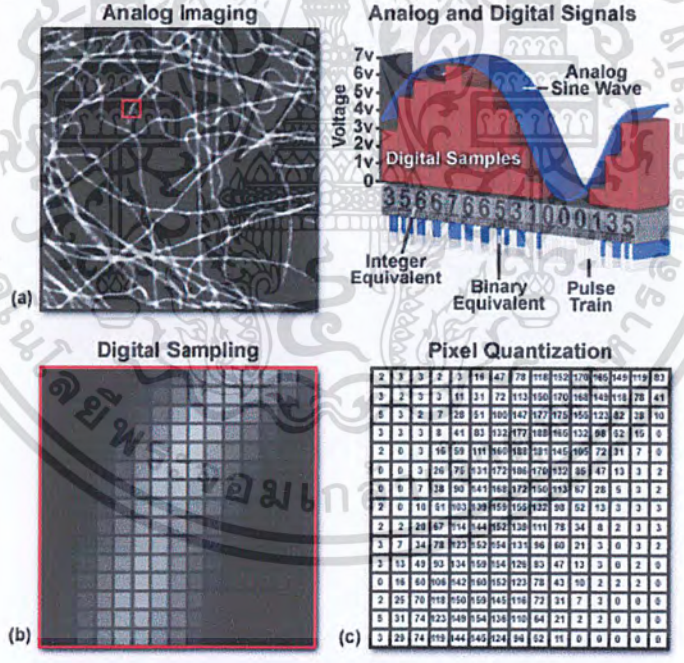


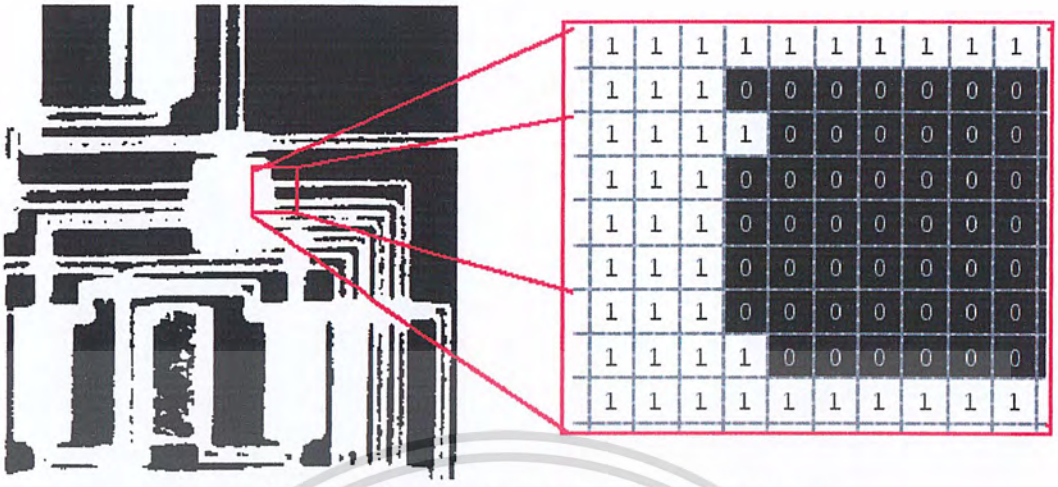
Figure 1

รูป 2.1 การแปลงภาพแอนะล็อกให้เป็นภาพดิจิทัล

ภาพดิจิทัลที่ได้จะมีรูปแบบการเก็บเป็นเมทริกซ์ ซึ่งจะมีการจัดเก็บภาพแต่ละชนิดต่างกัน ขึ้นอยู่กับระบบสีของภาพดังกล่าว โดยแบ่งชนิดของภาพได้ดังนี้

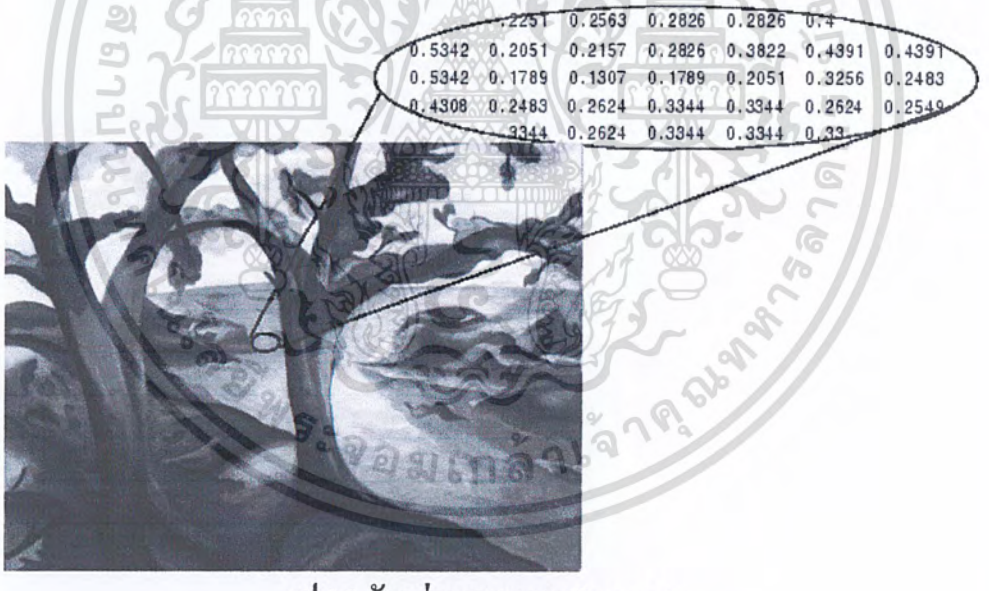
1) Binary image หรือ ภาพขาว-ดำ เป็นรูปที่ใช้เนื้อที่เพียง 1 บิต ต่อ พิกเซล โดยค่าสีจะมีแค่สองค่าคือ 0 หรือสีดำ และ 1 หรือสีขาว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.2 ตัวอย่างภาพแบบ Binary หรือ ภาพขาว-ดำ

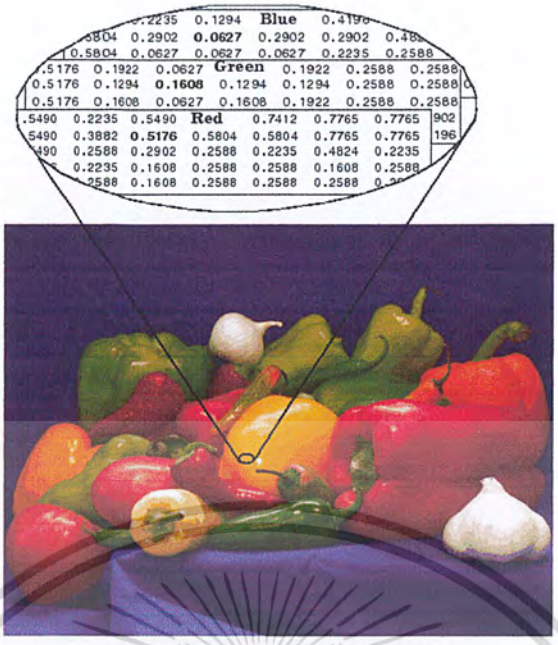
2) Grayscale Image เป็นรูปที่เก็บโดยใช้รูปแบบของอาร์เรย์ 2 มิติ โดยค่าที่เก็บจะมีค่าอยู่ในช่วงๆหนึ่ง ซึ่งระดับของสีขึ้นอยู่กับขนาดของบิตที่ใช้เก็บค่าสี



รูป 2.3 ตัวอย่างภาพแบบ Grayscale

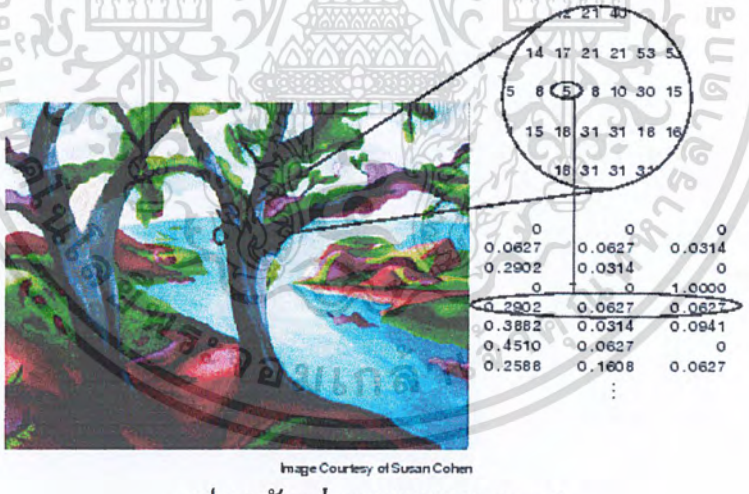
3) RGB Image หรือ Truecolor Image เป็นรูปที่เก็บโดยใช้อาร์เรย์ 3 มิติ ขนาด m x n x 3 โดยที่ m คือความยาว และ n คือความกว้างของภาพในหน่วยพิกเซล ส่วนมิติสุดท้ายนั้น ในแต่ละมิติจะเก็บค่าสีแยกกัน คือสีแดง(Red) สีเขียว(Green) และสีน้ำเงิน(Blue)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.4 ตัวอย่างภาพแบบ RGB

4) Indexed Image เป็นรูปที่มีรูปแบบการเก็บแบบ indexed คือ ภาพประเภทนี้จะเก็บค่าสีเป็น indexed และในแต่ละช่องอาร์เรย์ จะเก็บตำแหน่งของสีใน indexed นั้นๆไว้



รูป 2.5 ตัวอย่างภาพแบบ Indexed

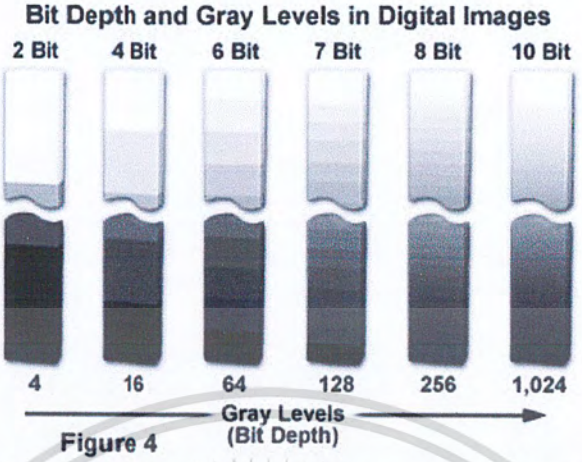
2. 2 ระบบสี (Color Model)

2.2.1 ระบบสีเทา (Grayscale)

ระบบสีเทา เป็นช่วงของเฉดสีเทา ซึ่งแตกต่างกับภาพขาว-ดำ ที่มีเพียง 2 สี คือขาว กับดำ สีใน ระดับสีเทานี้ แสดงถึงความเข้มของสี (Intensity) ในระดับต่างๆ โดยสีดำเป็นส่วนที่มีความเข้มของสีน้อยและสีขาวจะมีความเข้มของสีมาก โดยจำนวนระดับของสีขึ้นอยู่กับขนาดของบิตที่ใช้เก็บค่าสี แต่โดยทั่วไปแล้วจะเก็บข้อมูลสีประเภทนี้ด้วยข้อมูลขนาด 8 บิต หรือ 1 ไบต์ ซึ่งก็คือจะให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

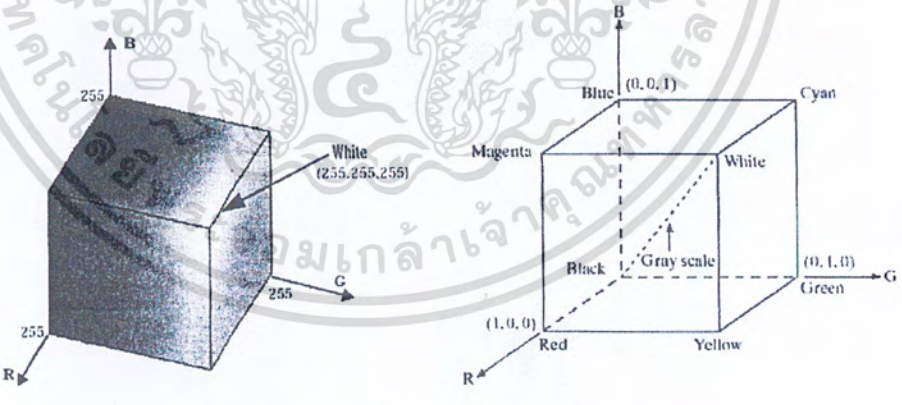
ความละเอียดของสีที่ 265 เฉดสี (0-255)



รูป 2.6 ระดับสีของ Grayscale ตามขนาดข้อมูลที่เก็บค่าสี

2.2.2 ระบบสี RGB

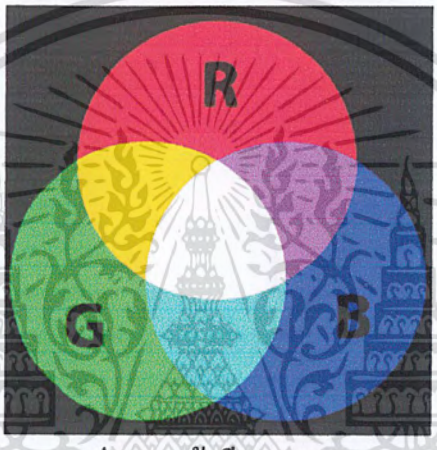
ระบบสี RGB เป็นระบบสีที่ประกอบด้วยแม่สีหลักสามสีคือ แดง (Red), เขียว (Green) และ น้ำเงิน (Blue) RGB นั้นเป็นระบบสีแสง และเป็นแบบ Additive คือ ถ้าไม่มีสีใดเลยจะมองเห็นเป็นสีดำ และในทางกลับกัน หากมีครบทุกสีจะมองเป็นสีขาว จะต่างกับระบบสีแบบ Subtractive หรือระบบสีแบบ CMYK ที่เป็นสีที่เกิดจากการสะท้อนหรือเรียกกันทั่วไปว่าสีวัตถุ



รูป 2.7 โครงสร้างระบบสี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.8 ความแตกต่างของ สีแสง (RGB) และ สีวัตถุ (CMYK)

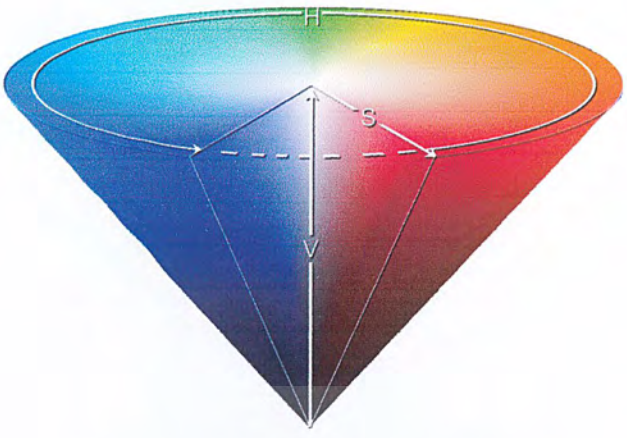


รูป 2.9 วงล้อสีแบบ RGB

2.2.3 ระบบสี HSV

ระบบสี HSV (Hue, Saturation, Value) หรือ HSB (Hue, Saturation, Brightness) เป็นระบบสีที่นิยมใช้กันในหมู่นักแต่งภาพ เนื่องจากเป็นระบบสีที่ใกล้เคียงกับความคิดของมนุษย์ได้ดีกว่าระบบสี RGB โดย Hue คือสีของภาพ, Saturation คือ ปริมาณความอิ่มตัวของสี ยิ่งมีค่านี้มาก ภาพจะมีสีสดยิ่งมีน้อย ภาพจะยิ่งมีสีน้อยลง จนในที่สุดจะกลายเป็นรูปที่ลักษณะแบบ Grayscale และ Value หรือ Brightness เป็นค่าที่แสดงถึงปริมาณความสว่างของภาพ หากมีค่ามากภาพจะยิ่งมีความสว่างมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 2.10 โมเดลสี HSV ในรูปแบบโคน (Cone)

รูป 2.11 ตัวอย่างภาพที่เก็บในแต่ละ channel ในระบบสี RGB และ HSV

2.3 ฮิสโตแกรม (Histogram)

ฮิสโตแกรม เป็นกราฟแสดงจำนวนพิกเซลที่ความสว่างต่างๆของภาพ โดยแกนนอนเป็นระดับความสว่างที่แบ่งระดับเป็น 256ระดับ (มักเรียกว่าระดับสีเทา หรือGray level) โดยมีค่าตั้งแต่ 0-255 โดยให้เป็นช่องๆเหมือนกับอาเรย์แบบ1มิติ แต่ละช่องมีความเข้มประจำช่องๆนั้นเอาไว้ ช่องแรกมีค่าความเข้มเป็น 0 ช่องต่อไปเป็น 1,2,3,4.... ไปเรื่อยๆจนถึง 255 ตามลำดับ เมื่อระดับสีเทามีค่าต่ำ (ด้านซ้ายมือ) หมายถึงมีความสว่างน้อย จะมองเห็นเป็นสีดำ ค่าระดับสีเทามาก (ด้านขวามือ) หมายถึงมีความสว่างมากจะมองเห็นเป็นสีขาว สำหรับแกนตั้งของกราฟจะแสดงจำนวนพิกเซลในแต่ละระดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.12 กราฟฮิสโตแกรม

2.4 การประมวลผลภาพกับรูปร่างและโครงร่างของภาพ(Morphological Image Processing)

เป็นการประมวลผลภาพโดยการเปลี่ยนแปลงลักษณะรูปร่างหรือโครงสร้างของภาพ โอเปอเรชั่นพื้นฐานโดยทั่วไปได้แก่การ Dilation Erosion และ Skeleton โดยการ Dilation คือการขยายภาพโดยมีสัดส่วนเท่ากันทั่วทั้งภาพ (Uniform) การ Erosion คือการย่อภาพ ส่วนการทำ Skeleton เป็นการหาโครงสร้างหลักของวัตถุ นอกจากโอเปอเรชั่นพื้นฐานดังที่ได้กล่าวข้างต้นแล้วยังมีโอเปอเรชั่นอื่นๆได้แก่การ Opening และ Closing เป็นต้น ในโครงงานนี้เราจะใช้เพียงแค่ Dilation และ Erosion เท่านั้น

2.4.1 เทคนิคของการ Hit และ Miss

โอเปอเรชั่นพื้นฐานสำหรับการกระทำกับรูปร่างหรือโครงสร้างของภาพ ไม่ว่าจะเป็นการย่อหรือการขยายภาพ จำเป็นที่จะต้องมีการนำเอาเทคนิคการ Hit และ Miss มาใช้แนวคิดของวิธีการนี้คือการกำหนดให้มีเมตริก (Template) ที่มีขนาดเล็กเล็ก ๆ และเป็นจำนวนคี่ (โดยทั่วไปจะมีค่าเท่ากับ 3x3) ไปเปรียบเทียบกับข้อมูลภาพ โดยการเปรียบเทียบจะทำตลอดทั้งภาพตั้งแต่จุดเริ่มต้นภาพจนถึงจุดสุดท้ายของภาพ ถ้าข้อมูลของภาพมีลักษณะเหมือนกับเมตริกดังกล่าวเอาพุทที่ได้จะขึ้นอยู่กับพิกเซลที่เป็นศูนย์กลางของเมตริกซึ่งจะถูกกำหนดให้เป็นค่าตามต้องการ (1 หรือ 0) แต่ถ้าข้อมูลในเมตริกไม่เหมือนกับข้อมูลภาพข้อมูลเอาท์พุทที่ได้จะมีค่าตรงกันข้าม

2.4.2 โอเปอเรชั่นพื้นฐานสำหรับรูปร่างหรือโครงสร้างพื้นฐาน

พิจารณาข้อมูลภาพจะเป็นลักษณะดังนี้

1 * 1 * 1
* 1 * 1 *
1 * 1 * 1

รูป 2.13 ข้อมูลภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากข้อมูลภาพตามรูป 2.13 สามารถแทนด้วยเซตในทางคณิตศาสตร์ได้ดังนี้คือ {(0,0),(0,2), (0,4), (1,1), (1,3), (2,0), (2,2), (2,4)}

เนื่องจากเราสามารถแทนลักษณะภาพได้ดังรูป 2.13 ดังนั้นเราสามารถกำหนดให้มีข้อมูลภาพสำหรับการทำโอเปอเรชั่นได้ ดังนี้คือ

$$A = \begin{matrix} 1 & * & 1 & * & 1 \\ * & 1 & * & 1 & * \\ 1 & * & 1 & * & 1 \end{matrix} \qquad B = \begin{matrix} * & * & * & 1 & 1 \\ * & * & * & 1 & 1 \\ * & * & * & 1 & 1 \end{matrix}$$

รูป 2.14 ข้อมูลภาพสำหรับการทำโอเปอเรชั่น

$$\text{A ยูเนียน B} = \begin{matrix} 1 & * & 1 & 1 & 1 \\ * & 1 & * & 1 & 1 \\ 1 & * & 1 & 1 & 1 \end{matrix}$$

รูป 2.15 A ยูเนียน B

$$\text{A อินเตอร์เซคชั่น B} = \begin{matrix} * & * & * & * & 1 \\ * & * & * & 1 & * \\ * & * & * & * & 1 \end{matrix}$$

รูป 2.16 A อินเตอร์เซคชั่น B

ข้อมูลภาพตามรูป 2.17 แสดงถึงพิกเซลที่เราทราบ(ค่าเท่ากับ 1) และค่าที่เราไม่ทราบ (แสดงด้วย *) ชุดของข้อมูลภาพจะขยายออกไปทางด้านบน ล่าง ซ้าย ขวาแบบไม่จำกัดข้อมูลภาพ สามารถเขียนได้ดังนี้

$$\begin{matrix} * & * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * & * \\ * & * & * & 1 & * & 1 & * & 1 & * & * & * \\ * & * & * & * & 1 & * & 1 & * & * & * & * \\ * & * & * & 1 & * & 1 & * & 1 & * & * & * \\ * & * & * & * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * & * & * & * \\ * & * & * & * & * & * & * & * & * & * & * \end{matrix}$$

รูป 2.17 แสดงค่าแทนค่าของภาพในแต่ละพิกเซล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วงกลมที่ลอบรอบพิกเซล 1 ตามรูป 2.17 แสดงตำแหน่งเริ่มต้นของภาพ (Origin)

2.4.3 การขยายภาพ (Dilation)

การขยายภาพในที่นี้จะพิจารณาสำหรับข้อมูลภาพที่เป็นแบบไบนารีโดยการใช้เทคนิคการ Hit และ Miss การขยายภาพจะทำได้โดยกำหนดTemplate (ซึ่งสามารถสร้างได้จาก * และ 1 โดยมีจุดเริ่มต้นที่กำหนดโดยวงกลม) และนำ Template นี้สแกนไปบนข้อมูลภาพตามลำดับตลอดทั้งภาพซึ่งในขณะที่จุดเริ่ม (Origin) ของ Template ตรงกับตำแหน่งข้อมูลภาพที่พิกเซลมีค่าเท่ากับ 1 นั้นก็จะทำการยูเนียน Template นี้เข้ากับข้อมูลภาพดังตัวอย่าง

ข้อมูลภาพ

*	*	*	*	*	*	1	*	*	1	*
*	*	*	*	*	*	1	*	*	*	1
*	*	*	*	*	1	1	*	1	1	*
*	*	*	*	1	1	1	1	1	1	1
*	*	*	*	1	1	1	1	1	*	1
*	*	*	*	1	1	1	1	1	1	1
*	*	*	*	1	1	1	1	1	1	1

Template

1	*
1	1

รูป 2.18 ข้อมูลภาพและเทมเพลต

ข้อมูลแถวแรกของภาพเป็นดังนี้

*	*	*	*	*	*	1	*	*	1	*

รูป 2.19 ข้อมูลแถวแรก

เมื่อทำการยูเนียนกับ Template ณ. ตำแหน่งข้อมูลภาพที่พิกเซลเท่ากับ 1 ในแถวแรก

*	*	*	*	*	*	1	*	*	*	*
*	*	*	*	*	*	1	1	*	1	*

รูป 2.20 ผลการยูเนียนในตำแหน่งพิกเซลเท่ากับ 1

และเมื่อยูเนียนกับ Template เข้ากับพิกเซลที่มีค่าเท่ากับ 1 ณ. ตำแหน่งพิกเซลที่สองในแถวแรก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
* * * * * * 1 * * * *
* * * * * * 1 1 * 1 1
```

รูป 2.21 ผลการยูเนียนในตำแหน่งพิกเซลเท่ากับ 2

และเมื่อทำการยูเนียนทั้งภาพจะได้ภาพสุดท้ายดังนี้

```
* * * * * * 1 * * 1 * *
* * * * * * 1 1 * 1 1 *
* * * * * 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1 1
```

รูป 2.22 ผลการยูเนียนทั้งภาพ

2.4.4 การย่อภาพ (Erosion)

การย่อภาพเป็นลักษณะของการลบข้อมูลภาพบริเวณขอบของภาพ การย่อภาพสามารถทำได้มีลักษณะคล้ายกับการขยายภาพโดยการสร้าง Template ขึ้นแล้วนำ Template ไปสแกนตามข้อมูลภาพ

สำหรับทุกตำแหน่งที่เลื่อน Template ไปบนภาพก็จะมีการเปรียบเทียบกับข้อมูลภาพ ถ้าข้อมูลภาพมีค่าเหมือนกับ Template จะทำการกำหนดค่าข้อมูลภาพในตำแหน่งที่ตรงกับจุดเริ่มต้น (Origin)ของ Template ถูกกำหนดให้มีค่าเท่ากับ 1

ข้อมูลภาพ

```
* * * * * * 1 * * 1 *
* * * * * * 1 * * * 1
* * * * * 1 1 * 1 1 *
* * * * 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 * 1
* * * * 1 1 1 1 1 1 1
* * * * 1 1 1 1 1 1 1
```

Template

```
1 *
1 1
```

รูป 2.23 ข้อมูลภาพและเทมเพลต

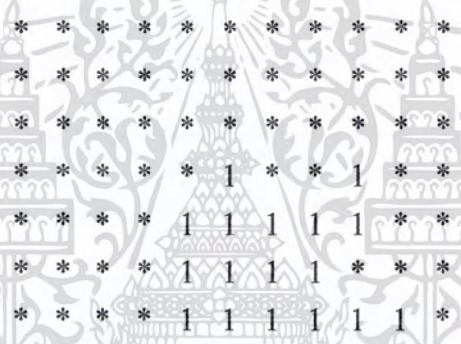
ผลที่ได้จะมีเพียง 3 ตำแหน่งเท่านั้นที่มีค่าเหมือนกับ Template

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 2.24 ผลการยูเนียนทั้งภาพ

ผลที่ได้ตามรูป 2.24 ข้อมูลภาพที่ผ่านการทำโอเปอเรชั่นกับ Template แล้วพบว่ามีข้อมูลของภาพเพียง 3 ตำแหน่งเท่านั้นที่เหมือนกับ Template ถ้ามีการเปลี่ยน Template เป็น

$\begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$ ผลที่ได้มีลักษณะดังนี้คือ



รูป 2.25 ผลการยูเนียนทั้งภาพเมื่อเปลี่ยนเทมเพลต

ผลที่ได้ตามรูปที่ 2.25 จะเห็นว่าจะเป็นการย่อขนาดของภาพแต่สามารถย่อขนาดได้น้อยกว่าเมื่อใช้ Template $\begin{matrix} 1 & * \\ 1 & 1 \end{matrix}$ ซึ่งได้ผลเป็นที่น่ายอมรับมากกว่าดังนั้นในการเลือก Template เป็นสิ่งที่สำคัญอย่างหนึ่งในการย่อและขยายภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การปรับปรุงภาพและการหาขอบภาพ

3.1 Hough transform

Hough transform เป็นเทคนิคที่สามารถนำมาใช้เพื่อแยกคุณลักษณะของรูปร่างเฉพาะภายในภาพ เพราะต้องการคุณสมบัติในรูปแบบพาราเมตริก Classical Hough transform ถูกใช้มากที่สุดในการตรวจหาของเส้นโค้งปกติ เช่น เส้นตรง, วงกลม, วงรีและอื่นๆ Hough transform ทั่วไปสามารถใช้ในapplication ซึ่งการอธิบายอย่างง่ายของการวิเคราะห์คุณสมบัติเป็นไปได้ยาก เนื่องจากความซับซ้อนในการคำนวณของขั้นตอน Hough algorithm ทั่วไป ดังนั้นในเอกสารนี้จะอธิบายแค่ Classical Hough transform

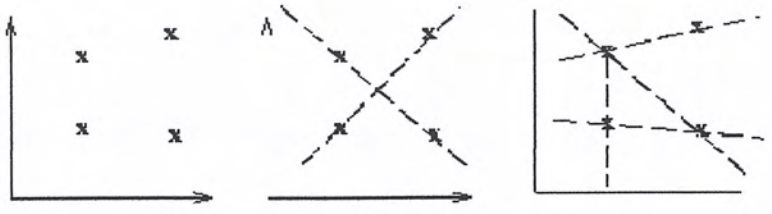
แม้จะมีข้อจำกัดของโดเมน Classical Hough transform (ซึ่งต่อจากนี้ไปโดยไม่มีคำนำหน้าคลาสสิก) ยังคงมีการใช้ application หลายๆส่วน มีขอบเขตของคุณสมบัติซึ่งสามารถอธิบายโดยเส้นโค้งปกติ ประโยชน์หลักของเทคนิค Hough transform คือมันเป็น tolerant ของช่องว่างในคุณลักษณะ boundary descriptions และค่อนข้างรับผลกระทบจากสัญญาณรบกวนภาพ

3.1.1 การทำงาน

เทคนิค Hough เป็นประโยชน์อย่างยิ่งสำหรับการใช้อธิบายคุณลักษณะทั่วไปที่กำหนดการวัดแบบ local ความคิดสร้างแรงจูงใจที่อยู่เบื้องหลัง Hough เทคนิคสำหรับ line detection ซึ่งแต่ละการวัด input (เช่น coordinate) แสดงถึงการมีส่วนร่วมในการแก้ปัญหาที่สอดคล้องกันทั่วไป (เช่น physical line ที่ให้ขึ้นไปยัง image point)

ตัวอย่างง่ายๆก็คือให้พิจารณาปัญหาของชุด line segments เพื่อการปรับตั้งค่าของ image points ที่ไม่ต่อเนื่อง (เช่น pixel location output จากเครื่องตรวจจับขอบ) รูป 3.1 แสดงการแก้ปัญหาที่เป็นไปได้ที่ของปัญหานี้ lack ของความรู้ priori เกี่ยวกับจำนวนของที่ต้องการ line segments (และความคลุมเครือเกี่ยวกับสิ่งที่ถือว่า line segments) ทำให้ปัญหานี้มีภายใต้ข้อจำกัด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ก) ข) ค)

รูป 3.1 การแก้ปัญหาที่เป็นไปได้

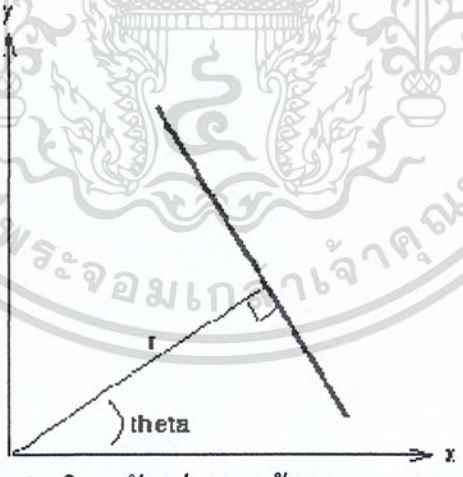
ก) Coordinate points

ข) และ ค) Possible straight line fittings

เราสามารถอธิบายส่วนของเส้นตรงในหลายรูปแบบ แต่สมการในการอธิบายการตั้งค่าของเส้นใช้ความคิดเชิง parametic หรือความคิดเชิง normal

$$x \cos \theta + y \sin \theta = r \qquad (3.1)$$

ซึ่ง r คือความยาวจากจุดเริ่มต้นกับ line นี้และ θ คือมุมของ r กับแกน X (ดูรูป 3.2) สำหรับจุดใดๆ (x,y) ใน line นี้, r และ θ เป็นค่าคงที่



รูป 3.2 คำอธิบายตัวแปรของ เส้นตรง (straight line)

ในบริบทของการวิเคราะห์ภาพ พิกัดของจุดของส่วนขอบ (เช่น (x_i,y_i)) ในภาพจึงเป็นที่รู้จักและเป็นค่าคงที่ในสมการเส้นพาราเมตริกในขณะที่ r และ θ จะไม่รู้จักตัวแปรที่เราแสวงหา ถ้าเราพล็อตค่า (r,θ) ที่เป็นไปได้กำหนดโดยแต่ละ (x_i,y_i) จุดในแผนที่พื้นที่ภาพ Cartesian ไปยังเส้นโค้ง ในพื้นที่ polar Hough parameter space

Point-to-curve transformation เป็น Hough transformation สำหรับเส้นตรง เมื่อดูในพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พารามิเตอร์ Hough จุดที่เป็นเส้นตรงเดียวกันในพื้นที่ภาพ Cartesian กลายเป็นเห็นได้ชัดอย่างง่ายดายตามที่พวกเขาให้ผลเส้นโค้งซึ่งตัดที่จุด (r,θ)

การแปลงถูกดำเนินการโดย quantizing พื้นที่พารามิเตอร์ Hough ไปเป็น finite intervals หรือ accumulator cells เมื่อ algorithm ทำงานแต่ละ (x_i,y_i) ถูกแปลงเป็นเส้นโค้ง (r,θ) และ accumulator cells ซึ่งอยู่ตามแนวเส้นโค้งนี้จะ incremented ส่งผลให้ยอดสะสมอาร์เรย์แสดงหลักฐานชัดเจนว่าเป็นเส้นตรงที่สอดคล้องกันอยู่ในภาพ

เราสามารถใช้ขั้นตอนเดียวกันเพื่อตรวจสอบคุณสมบัติอื่น ๆ ที่มีคำอธิบายเชิงวิเคราะห์ ตัวอย่างเช่นในกรณีของวงกลม สมการอิงตัวแปรคือ

$$(x - a)^2 + (y - b)^2 = r^2 \qquad (3.2)$$

ซึ่ง a และ b เป็นพิกัดของศูนย์กลางของวงกลมและ r เป็นรัศมี ในกรณีนี้ความซับซ้อนในการคำนวณของ algorithm จะเริ่มเพิ่มขึ้นเนื่องจากขณะนี้เรามีสามพิกัดในพื้นที่พารามิเตอร์และ 3-D accumulator (โดยทั่วไปแล้วการคำนวณและขนาดของอาร์เรย์เพิ่มขึ้นสะสมอย่าง polynomial กับจำนวนพารามิเตอร์ ดังนั้นเทคนิค Hough ขั้นพื้นฐานที่อธิบายที่นี่เป็นเพียงประโยชน์สำหรับเส้นโค้ง)

3.1.2 การใช้งาน

Hough transform สามารถนำมาใช้เพื่อระบุพารามิเตอร์ของเส้นโค้งที่เหมาะกับชุดของจุดที่กำหนดขอบ edge description นี้ถูกได้โดยทั่วไปจาก operator ตรวจสอบคุณสมบัติ เช่น Roberts Cross, Sobel หรือ Canny edge detector และอาจจะถูกรบกวน เช่นมันอาจจะมีขอบหลายส่วนที่เกี่ยวข้องกับรูปร่างทั้งหมด นอกจากนี้ผลผลิตของเครื่องตรวจจับขอบกำหนดคุณลักษณะเฉพาะที่อยู่ในภาพ การทำงานของ Hough transform คือการกำหนดคุณสมบัติทั้งสองสิ่งที่เป็น (เช่น เพื่อตรวจสอบคุณสมบัติที่จะมีการอิงคำอธิบายตัวแปร(หรืออื่นๆ)) และจำนวนของพวกมันที่อยู่ในภาพ

เพื่อที่จะแสดงให้เห็น Hough transform ในรายละเอียด เราเริ่มต้นด้วยภาพง่ายๆของรูปสี่เหลี่ยมสองรูป

รูป 3.3 ตัวอย่างภาพสี่เหลี่ยมสองรูป

Canny edge detector สามารถแสดงขอบเขตสำหรับส่วนนี้ ดังแสดงด้านล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

รูป 3.4 ตัวอย่างภาพสี่เหลี่ยมสองรูปผ่าน Canny edge detector

ที่นี่เราเห็นรอยต่อทั้งหมดในภาพ แต่ผลลัพธ์นี้ไม่มีอะไรเกี่ยวกับเอกลักษณ์(และปริมาณ)ของคุณลักษณะภายในรายละเอียดขอบเขตนี้ ในกรณีนี้เราสามารถใช้ Hough (line detection) transform เพื่อตรวจสอบแปดส่วนของเส้นตรงของภาพนี้และและระบุโครงสร้างทางเรขาคณิตที่แท้จริงของภาพด้วยวิธีนั้น

ถ้าเราใช้ขอบ/ จุดขอบเขตเหล่านี้ เป็นค่า input เพื่อเข้าสู่กระบวนการ Hough transform เส้นโค้งจะถูกสร้างขึ้นในพื้นที่พิกัดเชิงขั้ว (r,θ) สำหรับแต่ละจุดขอบในพื้นที่ Cartesian Accumulator array (เมื่อดูเป็นภาพความเข้ม) เหมือนดังภาพ

รูป 3.5 ตัวอย่างภาพที่ผ่านการทำ Hough transform

Histogram equalizing :: ภาพช่วยให้เราสามารถดูรูปแบบของข้อมูลที่มีอยู่ในค่าพิเซลความเข้มต่ำ ดังแสดงในภาพ

รูป 3.6 ตัวอย่างภาพที่ผ่านการ Histogram equalizing

ถึงแม้ว่า r และ θ เป็นพิกัดเชิงขั้วทางความคิด Accumulator space ถูกกำหนดอย่างมุมฉากด้วย θ เหมือน abscissa และ r เหมือน ordinate ซึ่ง Accumulator space ล้อมรอบที่ขอบตามแนวตั้งของภาพดังกล่าวว่าในความเป็นจริงมีเพียง 8 ยอดจริง

เส้นโค้งที่เกิดจากจุดบนเส้นตรงเดียวกันในการตัดภาพที่ไล่ระดับสีในยอด(r, θ) ในพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
117565

Hough transform จุดที่ตัดเหล่านี้แยกลักษณะส่วนเส้นตรงของภาพเดิม มีหลายวิธีที่ซึ่งหนึ่งนั้นใช้การสกัดจุดสว่างเหล่านี้หรือ local maxima จาก accumulator array ตัวอย่างเช่นวิธีการที่ง่ายเกี่ยวข้องกับการ thresholding และ thinning เพื่อแยกกลุ่มของจุดสว่างในรูป accumulator array ที่นี่เราใช้ relative threshold แยกสมการจุด (r, θ)ที่ไม่ซ้ำกันที่สอดคล้องกับแต่ละของขอบเส้นตรงในภาพต้นฉบับ(ในคำอื่นๆเราจะใช้เพียง local maxima ใน accumulator array ที่มีค่าเท่ากับหรือมากกว่าร้อยละคงที่บางส่วนของค่าสูงสุดทั้งหมด)

Mapping กลับจากพื้นที่ Hough transform (เช่น de – Houghing) ลงในพื้นที่ Cartesian ให้ผลคือชุดของ line descriptions ของภาพ โดยการซ้อนทับภาพนี้กับรูปกลับด้านของรูปเดิม เราสามารถยืนยันผลที่ Hough transform พบ 8 ด้านจริงของทั้งสองรูปสี่เหลี่ยมและจึงพบเรขาคณิต

รูป 3.7 ตัวอย่างภาพที่ผ่านการ Mapping

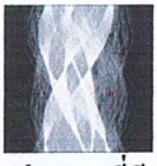
ความถูกต้องของการจัดตำแหน่งของตรวจพบและเส้นภาพต้นฉบับซึ่งไม่ชัดสมบูรณ์ในตัวอย่างง่ายๆนี้จะพิจารณา quantization ของ accumulator array (ทราบว่าหลายขอบรูปมีเส้นขอบที่ตรวจพบหลายๆเส้น การเกิดขึ้นจากการมียอด Hough-space หลายๆอันที่ใกล้เคียงกับค่า line พารามิเตอร์ที่คล้ายกันนี้ เทคนิคที่อยู่สำหรับการควบคุมผลกระทบนี้ แต่ไม่ได้ใช้ที่นี่เพื่อแสดงให้เห็นผลลัพธ์ของมาตรฐาน Hough transform)

lines ที่สร้างโดย Hough transform ถูกไม่จำกัดในความยาว ถ้าเราต้องการที่จะระบุส่วน line segments ที่เกิดขึ้นจริงซึ่งสร้างพารามิเตอร์การแปลง การวิเคราะห์ภาพต่อไปถูกต้องการเพื่อที่จะดูว่าบางส่วนของ lines ที่ยาวอย่าง ไม่จำกัดเหล่านี้จริงๆแล้วมีจุดบนพวกมัน เพื่อแสดงให้เห็นความแข็งแรงของเทคนิค Hough กับสิ่งรบกวน Canny edge description ได้รับความเสียหายจาก 1% salt และ pepper noise

รูป 3.8 ตัวอย่างภาพที่ salt และ pepper noise

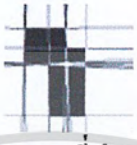
ก่อนที่จะทำ Hough transform มัน ผลที่ได้นำมาลงจุดในพื้นที่ Hough เป็นดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.9 ผลที่ได้จากตัวอย่างภาพที่มีsalt และ pepper noise

De-Houghing ผลลัพธ์ที่ได้ (และทับซ้อนบนตัวต้นฉบับ)



รูป 3.10 ตัวอย่างภาพที่ผ่าน De-Houghing

(ในกรณีข้างบนนี้, relative threshold เท่ากับ 40%)

ความไวของ Hough transform เพื่อให้ช่องว่างในคุณลักษณะขอบเขตสามารถตรวจสอบด้วยการเปลี่ยนภาพ

รูป 3.11 ตัวอย่างภาพที่ผ่าน Hough transform

ซึ่งได้รับการแก้ไขโดยใช้ paint program และ Hough representation เป็นดังนี้

รูป 3.12 ตัวอย่างภาพ Hough representation

และภาพ De-Houghed (ใช้ relative threshold 40%) ได้ดังนี้

รูป 3.13 ตัวอย่างภาพ De-Houghed (ใช้ relative threshold 40%)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในกรณีนี้เพราะ accumulator ไม่ได้รับเป็น entry จำนวนมากดังในตัวอย่างก่อนหน้านี้ เพียง 7 ยอดที่ถูกพบ แต่เหล่านี้จะเป็นเส้นโครงสร้างที่เกี่ยวข้องทั้งหมด

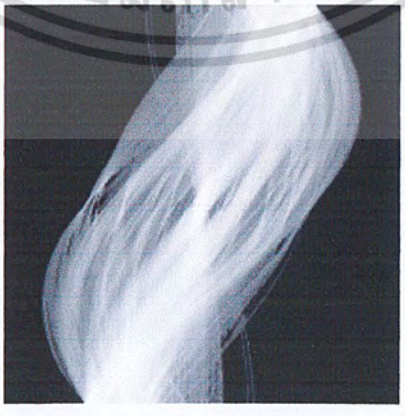
ตอนนี้เราจะแสดงตัวอย่างบางส่วนกับภาพธรรมชาติ ในกรณีแรกเรามีฉากเมืองที่อาคารเป็นอุปสรรคในหมอก

รูป 3.14 ตัวอย่างภาพเมืองที่อาคารเป็นอุปสรรคในหมอก

ถ้าเราต้องการหาขอบที่แท้จริงของอาคารซึ่ง edge detector (เช่น Canny) ไม่สามารถกู้คืนข้อมูลนี้ได้เป็นอย่างดี ดังแสดงในภาพ

รูป 3.15 ตัวอย่างภาพขอบเมืองที่อาคารเป็นอุปสรรคในหมอก

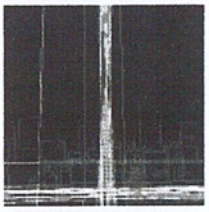
อย่างไรก็ตาม Hough transform สามารถตรวจสอบบางส่วนของเส้นตรงที่เป็นตัวแทนของขอบอาคารภายในพื้นที่กีดขวาง ส่วนแสดง Histogram equalized accumulator space ของรูปภาพต้นฉบับจะแสดงในภาพ



รูป 3.16 Histogram equalized accumulator space ของตัวอย่างภาพต้นฉบับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าเราตั้งค่า relative threshold เป็น 70% เราก็จะได้ ภาพ De-Houghed ดังต่อไปนี้



รูป 3.17 ตัวอย่างภาพ De-Houghed (relative threshold เป็น 70%)

มีเพียงไม่กี่ขอบที่มีการตรวจพบที่นี่และมีจำนวนมากที่ซ้ำที่มีหลายๆ lines หรือ edge fragments เกือบบนเส้นตรงเดียวกัน การประยุกต์ใช้ relative threshold ที่กว้างมากขึ้น นั่นคือ 50% ซึ่งให้ผลดังนี้

รูป 3.18 ตัวอย่างภาพ De-Houghed (relative threshold เป็น50%)

ผลที่ได้ lines ที่คาดหวัง ได้มากขึ้นแต่จำนวนของ lines ปลอมจำนวนมากที่เกิดขึ้นจาก หลายๆ edge fragments บนเส้นตรงเดียวกัน

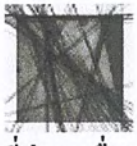
ตัวอย่างสุดท้ายของเรามาจากโปรแกรมตรวจจับระยะไกล ที่นี่เราต้องการจะตรวจจับถนนในภาพ

รูป 3.19 ตัวอย่างภาพถนน

รูป 3.20 ตัวอย่างภาพขอบของภาพถนน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อย่างไรก็ตามข้อมูลที่ถนนไม่สามารถใช้เป็น output ของเครื่องตรวจจับขอบเพียงอย่างเดียวได้

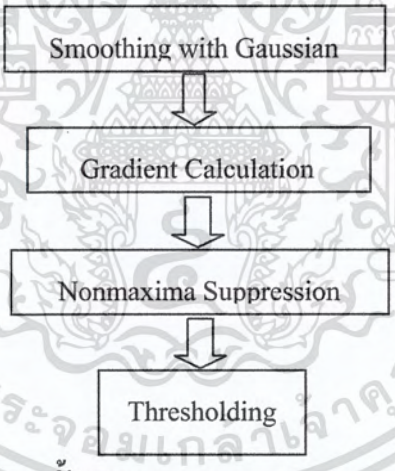


รูป 3.21 ตัวอย่างภาพที่ผ่านเครื่องตรวจจับ Hough line

ภาพนี้แสดงให้เห็นว่าเครื่องตรวจจับ Hough line สามารถกู้คืนข้อมูลบางอย่างนี้ได้ เนื่องจากความคมชัดในภาพต้นฉบับไม่ดีข้อจำกัดของการกำหนดของคุณสมบัติ (เช่นถนน) จะถูกระบุ

3.2 Canny Edge detection Algorithm

ขั้นตอนการหาขอบโดยวิธีของ Canny ประกอบด้วย 4 ขั้นตอน ดังรูป 3.22



รูป 3.22 ขั้นตอนของ Canny edge detection

การทำงานของ Canny edge detection นั้นเริ่มต้นจากการปรับภาพให้เรียบ (Smoothing) ด้วยตัวกรองเกาเซียน (Gaussian filter) เพื่อกำจัดสัญญาณรบกวน หลังจากนั้นคำนวณค่าขนาด (magnitude) และทิศทาง (orientation) ของ gradient โดยใช้การหาอนุพันธ์อันดับหนึ่ง ในถัดมาจึงใช้ nonmaxima suppression กับ gradient magnitude เพื่อทำให้ได้ขอบที่บางลงและในขั้นตอนสุดท้ายใช้ double thresholding algorithm เพื่อระบุพิกเซลที่เป็นขอบและช่วยเชื่อมต่อขอบ (Green, Bill 2002 ; ION528 - Image processing algorithms 2005 ; Canny Operator Links 2005 ; Rubino, Matthew 2005) โดยในแต่ขั้นตอนมีรายละเอียดดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.1 Smoothing

ในขั้นตอนแรกของการหาขอบโดยอัลกอริทึมนี้จะต้องกำจัดสัญญาณ รบกวนออกก่อน โดยใช้ Gaussian filter ซึ่งสามารถคำนวณได้จากการใช้กรอบ (mask) ขนาดเล็ก ขนาดของ Gaussian mask นี้หากมีขนาดกว้างจะมีผลทำให้ลดสัญญาณรบกวนได้มาก แต่ถ้ากว้างมากเกินไป จะมีผลทำให้ขอบย่อยๆ ที่เป็นส่วนรายละเอียดนั้นหายไป สำหรับการคำนวณหาภาพที่ได้จากการใช้ Gaussian filter เป็นดังสมการ

$$S[i,j] = G[i,j,\sigma]*I[i,j] \qquad (3.3)$$

กำหนดให้ $I[i,j]$ เป็นภาพที่ต้องการหาขอบ

$G[i,j,\sigma]$ เป็น Gaussian smoothing filter

σ เป็น spread of the Gussian (ควบคุมระดับของการ smoothing)

3.2.2 Gradient Calculation

ในขั้นแรกนำ smoothing image S[i, j] มาสร้าง x, y partial derivatives P[i, j] และ Q[i, j] ตามลำดับ ดังสมการ

$$P[i,j] \approx (S[i,j+1] - S[i,j] + S[i+1,j+1] - S[i+1,j]) / 2 \qquad (3.4)$$

$$Q[i,j] \approx (S[i,j] - S[i+1,j] + S[i,j+1] - S[i+1,j+1]) / 2 \qquad (3.5)$$

หลังจากนั้นนำค่า x , y partial derivatives มาคำนวณด้วยสูตรมาตรฐานสำหรับการแปลงรูปแบบจาก rectangular ไปเป็น polar (rectangular-to-polar conversion) เพื่อหาขนาดและทิศทางของ gradient ตามสมการ

$$M[i,j] = \sqrt{P[i,j]^2+Q[i,j]^2} \qquad (3.6)$$

$$\theta[i,j] = \arctan(Q[i,j],P[i,j]) \qquad (3.7)$$

จากสมการข้างต้นจะสามารถหาค่ามุม θ ออกมาได้เมื่อแทนค่าตัวแปรในฟังก์ชั่น arctan (x,y)

3.2.3 Nonmaxima Suppression

สำหรับการหาขอบโดย Canny method จุดที่ถือเป็นเส้นขอบได้นั้นต้องเป็นจุดที่ให้ค่าสูงสุดเฉพาะที่และเป็นทิศ ทางเดียวกับ gradient ด้วย ซึ่งด้วยวิธีดังกล่าวนี้ทำให้ได้ขอบที่บางเพียง 1 พิกเซล ภาพที่ได้หลังการทำ Nonmaxima Suppression จะให้ค่าเป็นศูนย์ในทุกจุดยกเว้นจุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่เป็น local maxima points ซึ่งจะยังคงค่าเดิมไว้

3.2.4 Thresholding

แม้ว่าภาพจะผ่านการ smoothing ในขั้นตอนแรกแล้วก็ตาม ภาพที่ได้อาจยังมีเส้นขอบที่ไม่ใช่ขอบที่แท้จริงปรากฏอยู่อันเนื่องจาก สัญญาณรบกวนหรือลักษณะของวัตถุในภาพเป็นพื้นผิวที่มีลวดลายหรือมีรายละเอียด ภายในมาก ดังนั้นเพื่อลดปัญหาดังกล่าวจึงได้มีการกำหนดค่า threshold ขึ้นมา 2 ค่า คือ high threshold (T1) และ low threshold (T2) โดยพิกเซลที่มีค่ามากกว่า T1 จะถูกปรับเป็น 1 (เป็นพิกเซลที่เป็นขอบ) แต่ถ้าน้อยกว่า T2 จะถูกปรับเป็น 0 ส่วนค่าที่อยู่ระหว่างค่า threshold ทั้งสอง การปรับเป็นค่า 0 หรือ 1 นั้นขึ้นอยู่กับพิกเซลที่อยู่รอบข้าง หากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ (ค่า >T1) มีค่ามากกว่า T2 แล้ว จะปรับค่าพิกเซลดังกล่าวให้มีค่าเป็น 1 และถือเป็นสมาชิกหนึ่งในภาพขอบด้วยเช่นกัน

3.3 การหาขอบภาพโดยวิธีโซเบล

การหาขอบภาพโดยวิธีโซเบล (Sobel Edge Detection) เป็นการหาขอบภาพโดยใช้เทมเพลตขนาด 3x3 สองเทมเพลต โดยเทมเพลตแรกจะใช้หาค่าความแตกต่างในแนวนอน (*Xdiff*) และค่าความแตกต่างในแนวตั้ง (*Ydiff*) ดังในรูป 3.23

$$X_{diff} = \begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \qquad Y_{diff} = \begin{matrix} 1 & 2 & 3 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{matrix}$$

รูป 3.23 เทมเพลตขนาด 3x3 สองเทมเพลต

ภาพ										$\lvert X_{diff}\rvert + \lvert Y_{diff}\rvert$								เทรสโฮลด์ด้วยค่า 12							
0	0	0	0	0	0	2	0	3	3																
0	0	0	1	0	0	0	2	4	2	4	6	4	10	14	12	14	4	0	0	0	0	1	1	1	0
0	0	2	0	2	4	3	3	2	3	6	8	10	20	16	12	6	0	0	0	0	1	1	1	0	0
0	0	1	3	3	4	3	3	3	3	4	10	14	10	2	4	2	4	0	0	1	0	0	0	0	0
0	1	0	4	3	3	2	4	3	2	2	12	12	2	2	4	6	8	0	1	1	0	0	0	0	0
0	0	1	2	3	3	4	4	4	3																

รูป 3.24 ตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.24 แสดงตัวอย่างการหาความแรงของขอบภาพด้วยวิธีโซเบล รูปสุดท้ายได้จาการทำ Threshold ด้วยค่า 12 แล้วปรับความเข้มแสงให้อยู่ในช่วง 0-1 เราสามารถสร้างเทมเพลตโซเบลที่มีขนาดใหญ่กว่า 3x3 เพื่อที่จะให้มีการครอบคลุมพื้นที่มากขึ้นได้

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
เครื่องมือที่ใช้ในการพัฒนา

4.1 การทำงานของ OpenCV

OpenCV เป็นไลบรารีสำหรับใช้งานเรื่องการประมวลผลภาพ (Image Processing) ซึ่งเป็นไลบรารี่ที่เป็นโอเพนซอร์ส (Open source) OpenCV ได้ถูกพัฒนาขึ้นโดยบริษัทอินเทล โดยมีจุดเด่นคือความสามารถในการประมวลผลภาพดิจิตอลได้ทั้งที่เป็นภาพนิ่งและภาพเคลื่อนไหว หรือไฟล์วีดีโอ ซึ่งไม่ยึดติดกับตัวฮาร์ดแวร์ นอกจากนั้นยังมีฟังก์ชันที่มีความสามารถทำภาพเบลอ หา threshold หา Histogram ของภาพได้ แต่ความสามารถโดยส่วนใหญ่แล้วจะทำการค้นหาขอบของภาพ การตรวจสอบการเคลื่อนไหว และการทำ Image Segmentation

นอกจากนี้ OpenCV สามารถจัดการข้อมูลแบบวีดีโอได้ด้วย เนื่องจาก OpenCV เป็นชุดคำสั่งที่ไม่ได้เป็นตัวโปรแกรม เมื่อต้องการเรียกใช้งานจึงต้องเขียนโปรแกรมเพื่อเรียกชุดคำสั่งเหล่านั้น ซึ่งภาษาที่นิยมเขียนคือภาษา C , C++ และภาษา Pyton ซึ่ง OpenCV จะประกอบด้วยสองส่วนคือ data structure ที่ใช้ในการเก็บข้อมูลต่างๆ อาทิ รูปภาพ เมตริกซ์และพิกัด สำหรับอีกส่วนคือ algorithm ซึ่งจะใช้ในการประมวลผลต่างๆ โดยเฉพาะการประมวลทางรูปภาพ สำหรับใน OpenCV จะประกอบด้วยไลบรารีอยู่4ส่วนได้แก่

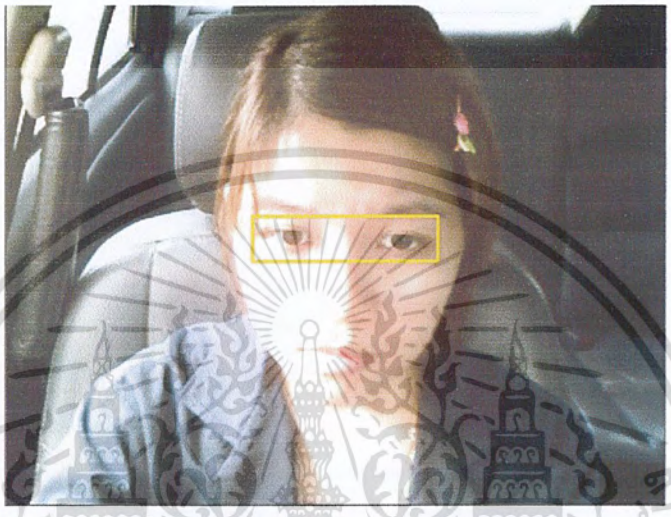
1) CXCORE เป็นฟังก์ชันเบื้องต้นที่ใช้จัดการเกี่ยวกับจุด ขนาด อาเรย์ (Array) หน่วยความจำ คำสั่งในการวาดภาพ การประกาศตัวแปรภาพ เป็นต้น ตัวอย่างคำสั่งในการประกาศรูปภาพคือ lplImage , CvMat , CvMatND
2) CV ใช้ในการประมวลผลและวิเคราะห์รูปภาพ ฟังก์ชันส่วนใหญ่จะทำงานกับพิกเซลที่เป็นอาเรย์ 2 มิติ หรือที่เราเรียกว่าภาพนั่นเอง เช่น การหาขอบหรือมุม การทำฮิสโตแกรม และการทำออพติคอลโฟลว์ (Optical Flow)เป็นต้น
3) Machine Learning เป็นไลบรารีที่รวมคลาสและฟังก์ชันทางสถิติ การแยกคลาส และการแบ่งกลุ่มข้อมูล อัลกอริทึมที่ใช้จะเขียนด้วยภาษา C++ แต่ละอัลกอริทึมจะมีคุณลักษณะเด่นแตกต่างกันไป แต่ทั้งหมดจะใช้คลาส CvStatModel ร่วมกัน
4) HighGUI เป็นไลบรารีที่ใช้ในการโหลดและบันทึกภาพ ติดต่อกับกล้องวีดีโอ การสร้างหน้าต่างเพื่อแสดงภาพและทำลายภาพ การเปลี่ยนขนาดและเคลื่อนย้ายหน้าต่าง รวมไปถึงการตรวจสอบเมาส์ (Mouse) และแป้นพิมพ์

4.2 Region-of-interest (ROI)

Region-of-interest (ROI) คือบริเวณที่เราสนใจ อาจจะเป็นบริเวณใดภายในภาพก็ได้ โดยการตี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

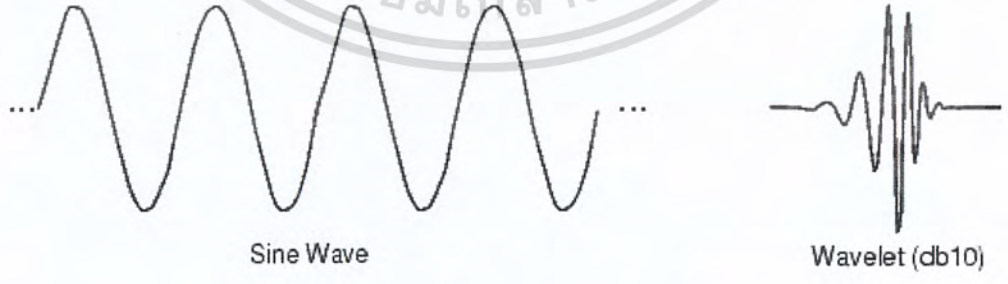
กรอบล้อมรอบบริเวณที่สนใจ ด้วยวงกลม กรอบสี่เหลี่ยม หรือกรอบรูปเหลี่ยมใดๆ เพื่อนำภาพเฉพาะส่วนดังกล่าวมาประมวลผล หรือเปลี่ยนแปลงภาพตามต้องการ โดยไม่มีผลกระทบกับส่วนอื่นๆ ซึ่งใน 1 ภาพ สามารถกำหนดได้หลายๆ ROI เมื่อกำหนดตำแหน่งต่างๆแล้ว จะสร้าง Mask ที่เป็น Binary Mask สำหรับใช้กำหนดขอบเขตที่จะมีการเปลี่ยนแปลงภายในรูปภาพนั้นๆ โดยให้ค่าส่วนที่สนใจเป็น 1 หรือสีขาว และให้ส่วนอื่นๆ เป็น 0 หรือสีดำ



รูป 4.1 การกำหนด Region of Interest

4.3 Haar wavelet

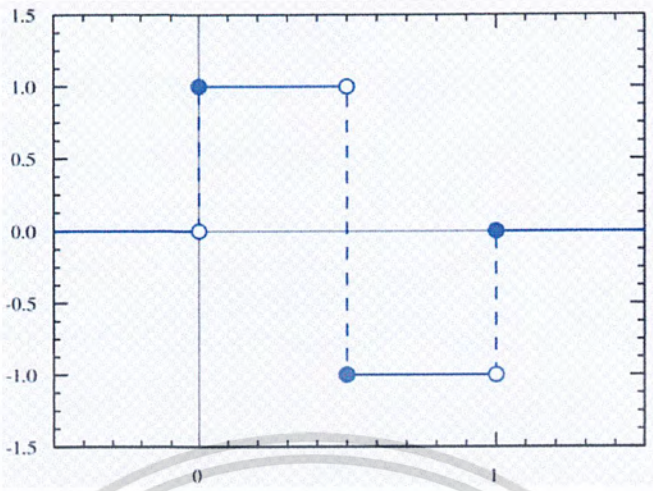
Wavelet Transform เป็นคณิตศาสตร์ สำหรับการวิเคราะห์ลักษณะของสัญญาณ เฉพาะช่วงเวลาและความถี่ที่สนใจได้ ลักษณะของคลื่นแบบนี้จะมีการเปลี่ยนแปลงอย่างต่อเนื่อง เป็น Asymmetric และมีค่าเฉลี่ยปฏิยาณุพันธุ์เข้าใกล้ศูนย์



รูป 4.2 ลักษณะของ Sine Wave และ Wavelet

Haar Wavelet เป็น wavelet พื้นฐาน มีลักษณะเป็น Discrete Function โดย wavelet แบบนี้มีข้อด้อยคือ ไม่มีความต่อเนื่องของฟังก์ชั่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

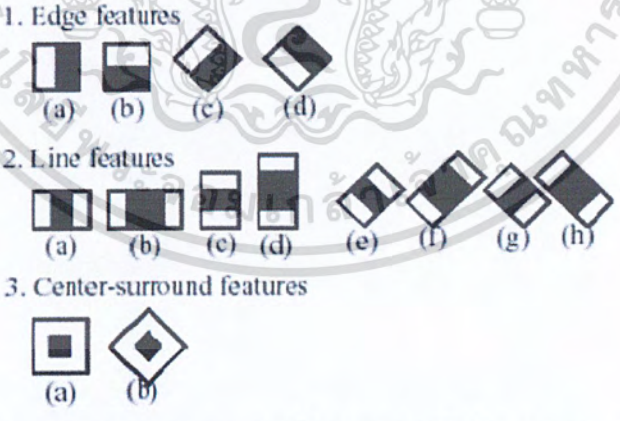


รูป 4.3 ลักษณะของ Haar Wavelet

4.4 Haar like-feature

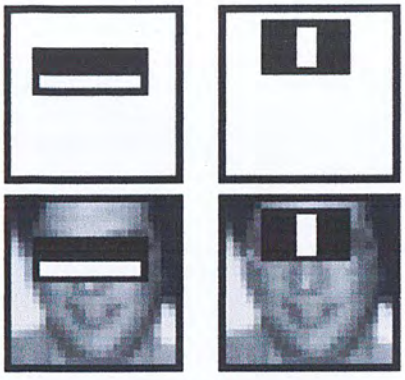
4.4.1 ทฤษฎีเบื้องต้น

Haar like-Features ตามวิธีของ Viola และ Jones เป็นวิธีการตรวจจับและตีความวัตถุภายในภาพ ด้วยหลักการของ Haar Wavelet สำหรับสร้างรูปที่เหลี่ยม (Feature) โดยที่ภาพนี้แสดงถึงผลต่างระหว่างพื้นที่ส่วนสีขาว และส่วนที่เป็นสีดำ ซึ่ง Feature สามารถเปลี่ยนแปลงขนาด และตำแหน่งได้ ใช้สำหรับการตรวจจับลักษณะบนภาพแบบต่างๆ เช่น เส้นตรง, วงกลม เป็นต้น



รูป 4.4 แบบของ Features สำหรับการตรวจจับลักษณะแบบต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.5 ตัวอย่างการใช้ Feature ตรวจจับลักษณะต่างๆ

การคำนวณค่าของ Feature นั้น ใช้หลักการคำนวณแบบ Integral image ซึ่ง Integral image คือผลรวมของค่าในทุกๆ พิกเซล ที่ตำแหน่ง (x, y) ใดๆ ซึ่งมีเวลาการทำงานเป็น O(1) ทำให้การคำนวณ Feature นั้นทำได้เร็วมาก

$$P(x,y) = \sum_{x' \le x, y' \le y} i(x', y')$$

รูป 4.6 การคำนวณแบบ Integral image

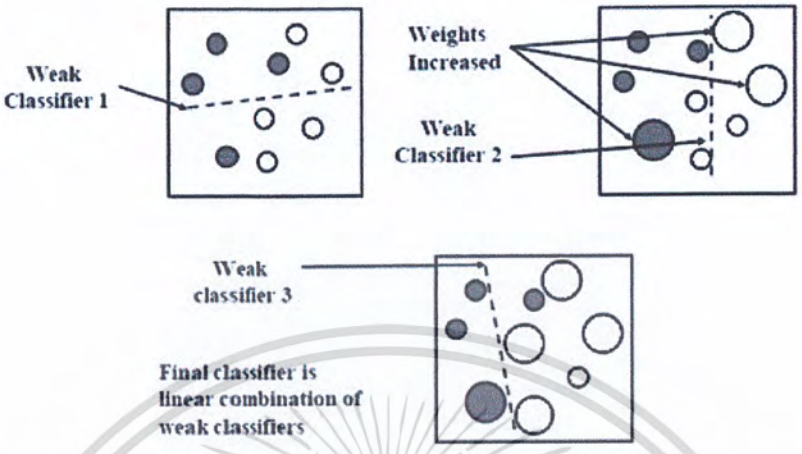
ในการทำ Haar like-Feature นั้น จำเป็นต้องมีภาพตัวอย่างจำนวนมาก ซึ่งใช้ในการคัดเลือกลักษณะของรูปที่ต้องการตรวจจับและตีความหมาย ซึ่งมีสองลักษณะคือ Positive Image หรือรูปที่มี Object นั้นๆประกอบอยู่ภายในภาพ และ Negative Image หรือภาพใดๆที่ไม่มี Object ที่เราต้องการอยู่ภายในภาพ

Haar like-Feature ใช้หลักการของ AdaBoost (Adaptive Boost) ซึ่งเป็นกระบวนการหา Feature ที่มีลักษณะใกล้เคียง และแตกต่างกับภาพนำเข้า สำหรับการจัดประเภทของภาพ โดยการถ่วงน้ำหนักให้ส่วนต่างๆภายในภาพ บนภาพ Positive และภาพ Negative เพื่อใช้หาลักษณะของ Object ที่ "ใช่" และ "ไม่ใช่" ในลักษณะต่างๆ มีกระบวนการดังนี้

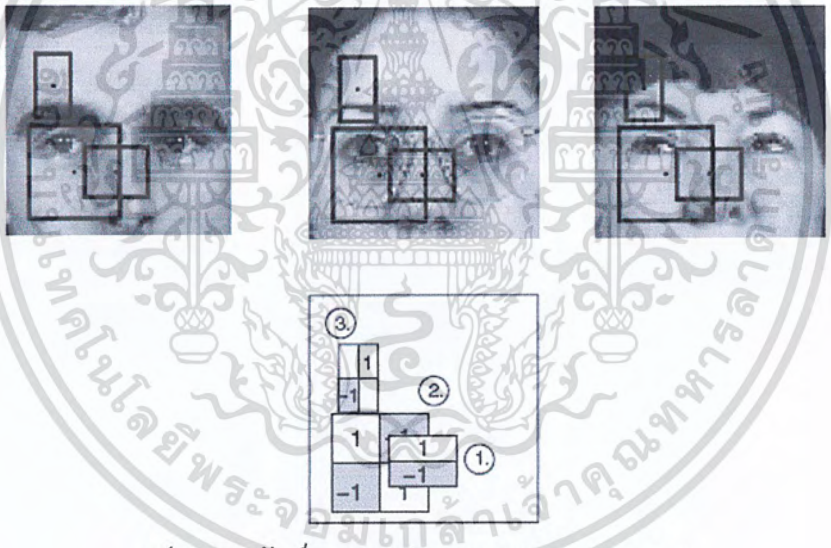
1) เริ่มแรกกำหนด ค่าน้ำหนักให้กับ Feature ที่วิ่งหาภายในภาพตัวอย่าง
2) หาบริเวณที่ประกอบด้วย ส่วนที่เราต้องการ
3) เพิ่มค่า น้ำหนักให้กับส่วนที่เหลือ เฉพาะลักษณะที่เราต้องการ ที่ยังไม่ได้ แบ่งลักษณะไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) ทำวนเช่นนี้ซ้ำไปเรื่อยๆ จนสุดท้าย นำบริเวณที่ได้ทั้งหมดมารวมกัน จะได้บริเวณของ Object ที่เราต้องการหา และลักษณะในส่วนต่างๆภายใน Object นั้น



รูป 4.7 การทำงานของ Adaboost



รูป 4.8 ผลลัพธ์จากการทำกระบวนการ Adaboost

4.4.2 การค้นหาใบหน้าด้วยกระบวนการ Haar-like(Face Detection by Haar-like Method)

Computer vision คือการเปลี่ยนแปลงอย่างหยาบๆ และรวดเร็วของพื้นที่ ดังนั้นจึงค่อนข้างจะเป็นเรื่องทางด้านเทคนิคมากกว่าทางด้านอัลกอริทึม ในเรื่องการทดลองที่ออกมา โดย face detector ได้เข้ามาพร้อม OpenCV ในการทดลองตามลำดับ แต่กระนั้นก็ตาม face detector ก็สามารถทำงานได้ดีเมื่อทำงานบนพื้นฐานทางด้านเทคนิค ด้วยเหตุนี้จึงมีการใช้กันอย่างแพร่หลาย ซึ่งในความเป็นจริงมีหลายหน่วยงานได้ใช้ face detector ของ OpenCV เพื่อประยุกต์ใช้ในการตรวจจับวัตถุที่ไม่มีความตายตัวมากนักเช่น ใบหน้า รถยนต์ จักรยาน หรือ ร่างกายมนุษย์ เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการทดสอบจากการตรวจพบเป็นพันๆครั้ง สำหรับภาพที่ผ่านการทดสอบและถูกเลือกแต่ละมุมมองของวัตถุ ซึ่งเทคนิคนี้ได้ถูกใช้ในการสร้างตัวตรวจจับที่เรียกว่า state-of-the-art แม้ว่ามีการตรวจสอบที่แตกต่างกันสำหรับการทดสอบสำหรับแต่ละมุมมองของวัตถุ ด้วยเหตุนี้แล้ว Haar Classifier จึงเป็นเครื่องมือที่มีประโยชน์ที่จะช่วยเหลือในการ recognize

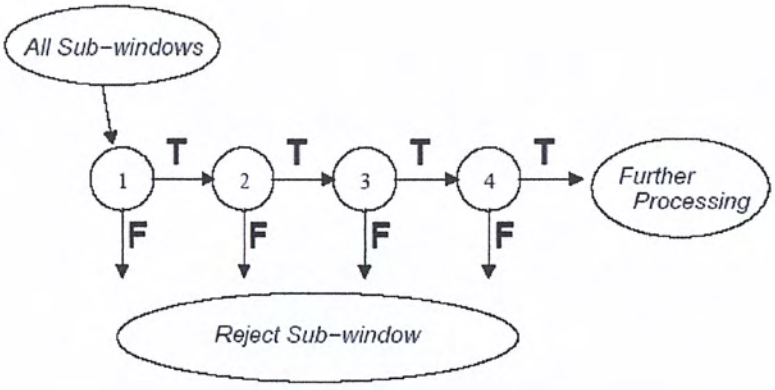
การพัฒนา face detector ใน OpenCV ได้มีขึ้น ซึ่งใช้คุณสมบัติของเส้นทแยงมุม โดย OpenCV เรียกการตรวจพบนี้ว่า Haar Classifier เพราว่ามันใช้คุณสมบัติของ Haar หรือ Haar-like wavelets ซึ่งประกอบด้วยการเพิ่มหรือลบพื้นที่สี่เหลี่ยมของภาพก่อนจะทำการกำหนดขีดเริ่มเปลี่ยน (threshoding) ภาพผลลัพธ์ ใน OpenCV จะมีไฟล์ที่บรรจุเซตของการทดสอบการจดจำวัตถุอยู่ แต่ตัวโค้ดเองก็อนุญาตให้มีการทดสอบและบันทึกค่าใหม่สำหรับโครงสร้างของวัตถุใหม่ด้วยเช่นกัน ซึ่งจากการทดสอบแล้วสรุปว่าสามารถทำงานได้ดีบนวัตถุใดๆก็ตามที่มีลักษณะของภาพหยาบๆไม่ยืดหยุ่น (ไม่เฉพาะแค่ใบหน้า)

รูป 4.9 ตัวอย่างการค้นหาใบหน้าด้วยอัลกอริทึม Haar-Like

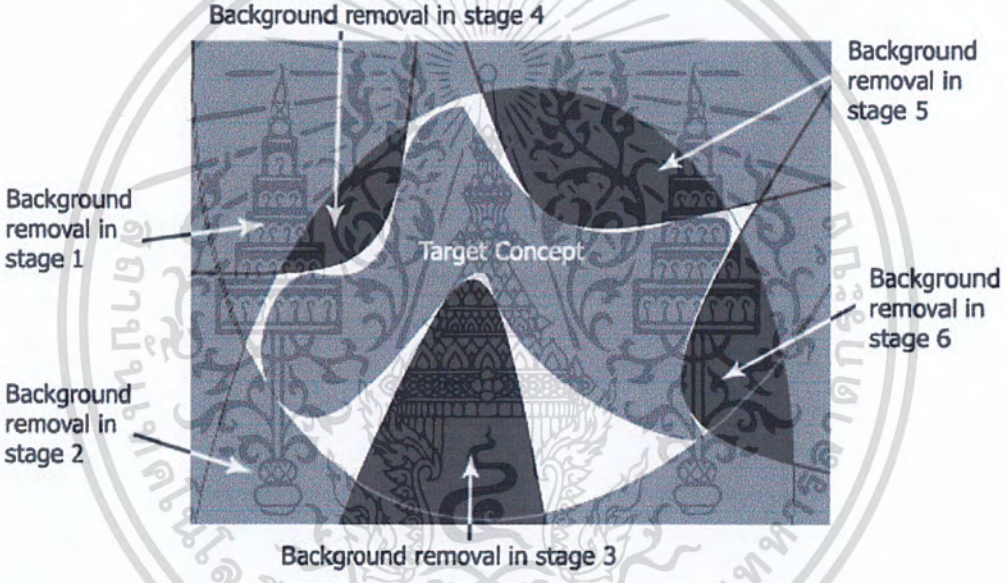
4.5 Cascade Classifiers

เป็นกระบวนการตีความหมายของภาพ โดยการแบ่งประเภทของภาพ ตามลักษณะภายในภาพ โดยเริ่มต้นจากการตัดส่วนของ Sub window ที่เป็น Negative ออกไปก่อน แล้วจากนั้นค่อยใช้ ส่วนที่เป็น Positive เคลื่อนที่วนภายในภาพ หากไม่พบพบเจอลักษณะที่ตรงกัน ก็จะเปลี่ยนลักษณะการตรวจจับภายใน Sub window หากพบลักษณะที่ตรงกัน ก็จะเปลี่ยนลักษณะในการตรวจจับ ทำเช่นนี้จนครบ จะได้รูปที่สามารถบอกได้ว่าภาพดังกล่าวเป็นภาพอะไรจากลักษณะต่างๆภายในภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.10 การทำงานของ Haar Cascade Classifier



รูป 4.11 ลักษณะการทำงานของ Cascade Classifier

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
การออกแบบและการพัฒนา

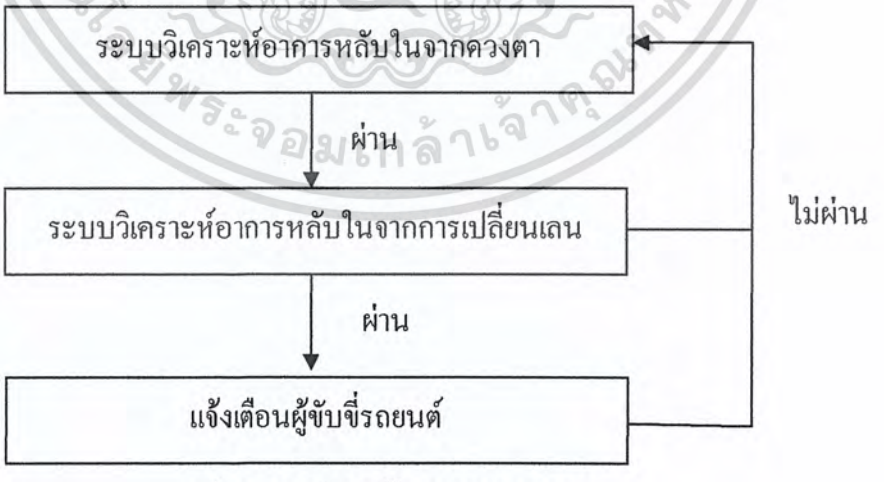
5.1 การทำงานโดยรวมของโปรแกรมวิเคราะห์อาการหลับใน

โปรแกรมวิเคราะห์อาการหลับในแบ่งออกเป็น 2 ระบบ คือ ระบบวิเคราะห์อาการหลับในจากดวงตา และระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนน

ระบบวิเคราะห์อาการหลับในจากดวงตา จะมีกล้องเว็บแคมติดบริเวณด้านหน้าของผู้ขับขี่รถยนต์ เพื่อรับภาพใบหน้าของผู้ขับเข้ามา แล้วใช้หลักการประมวลผลภาพดิจิทัลมาวิเคราะห์หาบริเวณดวงตาของผู้ขับจากใบหน้าทั้งหมด เพื่อนำตาดำนั้นมาวิเคราะห์ตามหลักทางชีววิทยาว่าผู้ขับรถยนต์กำลังหลับในอยู่หรือไม่ หากกำลังหลับในอยู่ระบบจะแจ้งเตือนด้วยเสียงให้ผู้ขับได้สติขึ้นมา

ระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนน จะมีกล้องเว็บแคมติดบริเวณข้างตัวรถยนต์ด้านซ้ายและขวา เพื่อตรวจจับเส้นพื้นถนน หากมีการเปลี่ยนเลนที่ไม่เหมาะสม เช่นการเปลี่ยนเลนที่เร็วหรือช้าเกินไป ระบบจะแจ้งเตือนด้วยเสียงเช่นกัน

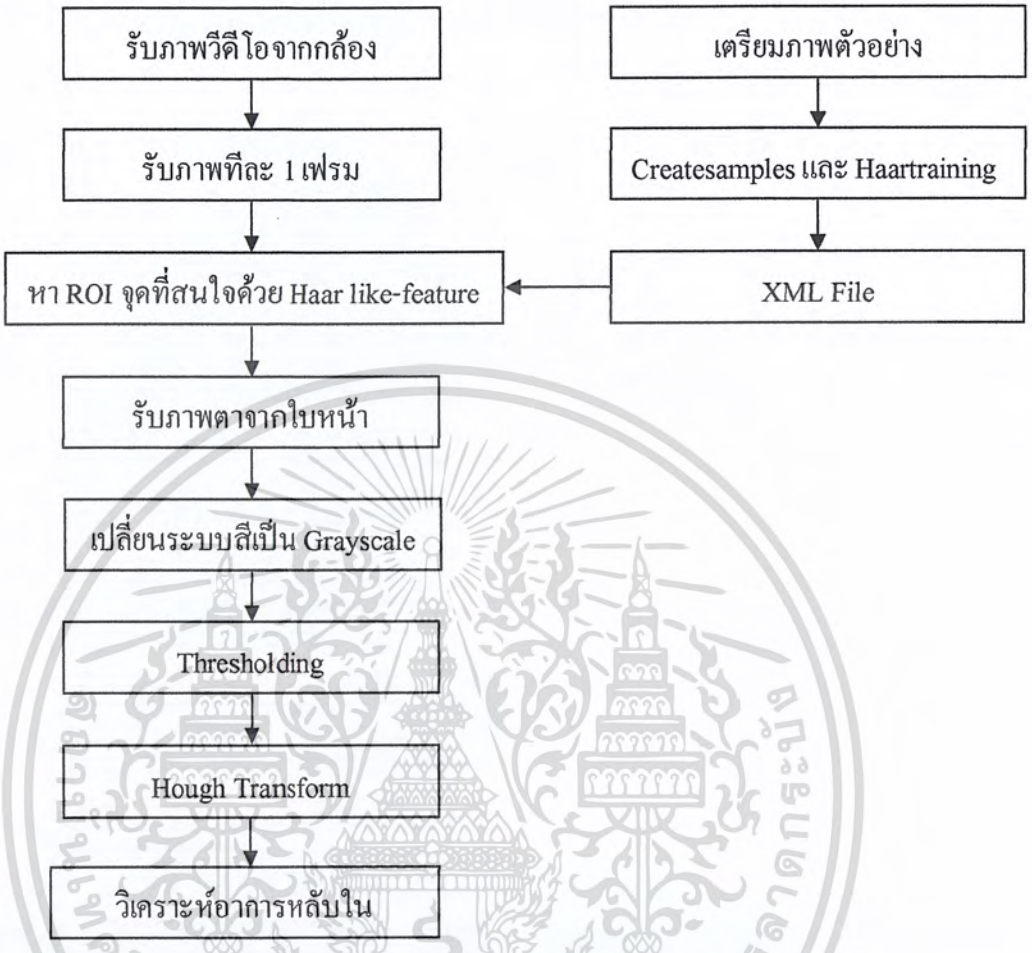
การทำงานร่วมกันของทั้งสองระบบ คือ ให้ระบบวิเคราะห์อาการหลับในจากดวงตามีความสำคัญสูงกว่าระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนน กล่าวคือ หากเมื่อใดที่ระบบวิเคราะห์อาการหลับในจากดวงตาตรวจพบว่าผู้ขับมีโอกาสหลับในอยู่ ระบบจะไปวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนนต่อไป และเมื่อระบบทั้งสองตัดสินใจว่าผู้ขับมีโอกาสหลับในอยู่ ก็จะแจ้งเตือนผู้ขับขี่รถยนต์ด้วยเสียงต่อไป



รูป 5.1 การทำงานโดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 การทำงานของระบบวิเคราะห์อาการหลับในจากดวงตา



รูป 5.2 การทำงานของระบบวิเคราะห์อาการหลับในจากการวิเคราะห์ดวงตา

การทำงานของระบบวิเคราะห์อาการหลับในจากการวิเคราะห์ดวงตา อาศัยการรับภาพหน้าของผู้ขับจากกล้องเว็บแคม ซึ่งจะได้ภาพวีดีโอที่มีใบหน้า ดวงตา และสิ่งแวดล้อมอยู่ปนกันบนภาพ ในแต่ละเฟรมภาพจะมีหาบริเวณที่สนใจของภาพ (Region of interest หรือ ROI) คือ ดวงตา โดยใช้หลักการ Haar like-feature ซึ่งหลักการนี้ต้องมีการเตรียมภาพตัวอย่างจำนวนมากทั้งภาพที่ต้องการและภาพที่ไม่ต้องการ มารัน opencv_createsamples.exe และ opencv_haartraining.exe ใน C:\OpenCV2.1\bin จะทำให้ได้ไฟล์ XML มาเพื่อใช้ร่วมกับฟังก์ชันของ Haar Classifier ต่อไป

จากนั้นเมื่อได้ดวงตาตามที่ต้องการแล้วนั้น ระบบจะนำดวงตามาเปลี่ยนระบบสีเป็นภาพระดับสีเทา (Grayscale) และทำระดับขีดเริ่ม (Threshold) หาลูกตาดำจากดวงตา เพื่อที่จะสามารถตรวจสอบรูปร่างของลูกตาดำว่าเป็นวงกลมหรือไม่ เพราะถ้าเป็นวงกลมแสดงว่าผู้ขับขี่กำลังลืมตาอยู่ แต่หากว่าเป็นรูปร่างอื่น เช่น เส้นโค้งหรือเส้นตรงแสดงว่าผู้ขับขี่กำลังหลับตา หลังจากนั้นวิเคราะห์ว่าผู้ขับมีการลืมตาหรือหลับตายาวนานเกิน 4 วินาทีหรือไม่หากเป็นอย่างนั้นจะแจ้งเตือนผู้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขับด้วยเสียง เนื่องจากโดยปกติแล้วผู้ขับขี่รถยนต์ที่จะต้องมีการกระพริบตา 16 ครั้งใน 1 นาที ขณะขับรถยนต์

5.3 ขั้นตอนการสร้างไฟล์ XML สำหรับ Haar like-feature

5.3.1 การเก็บและเตรียมภาพตัวอย่าง

ในขั้นตอนการหาดวงตาเราจะต้องเตรียมไฟล์ XML ที่เก็บลักษณะของดวงตาที่ต้องการไว้ ซึ่งสร้างมาจากภาพตัวอย่างทั้งแบบที่สนใจ (Positive) และภาพที่ไม่สนใจ (Negative)

5.3.1.1 รายละเอียดของภาพ Positive

ภาพที่สนใจ (Positive Image) เป็นภาพของวัตถุที่เราสนใจจะตรวจจับ โดยเก็บภาพในมุมมองต่างๆ และมีจำนวนหลากหลายบุคคล มุมมอง แสง เป็นต้น และควรใช้รูปที่มีจำนวนมากในระดับหนึ่ง เพื่อให้สามารถตรวจจับวัตถุที่ต้องการหาได้ง่าย ถูกต้อง และแม่นยำ

รูป 5.3 ตัวอย่างภาพ Positive image

5.3.1.2 รายละเอียดของภาพ Negative

ภาพที่ไม่สนใจ (Negative Image) เป็นภาพสิ่งแวดล้อมอื่นๆ โดยต้องไม่มีสิ่งที่เราสนใจจะตรวจจับอยู่ภายใน เช่น ภาพที่บุคคลกำลังภาพที่ไม่มีใบหน้าบุคคล เป็นต้น ภาพ Negative นี้จำเป็นต้องใช้ภาพตัวอย่างจำนวนมาก

รูป 5.4 ตัวอย่างภาพ Negative image

5.3.1.3 ขั้นตอนการเตรียมภาพ Positive และภาพ Negative

1) ถ่ายภาพ Negative ด้วยกล้องดิจิทัล หรืออาจหาภาพจากอินเทอร์เน็ต โดยเน้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพที่มีลักษณะหลากหลาย และ มีขนาดของภาพใกล้เคียงกับขนาดของวีดีโอ

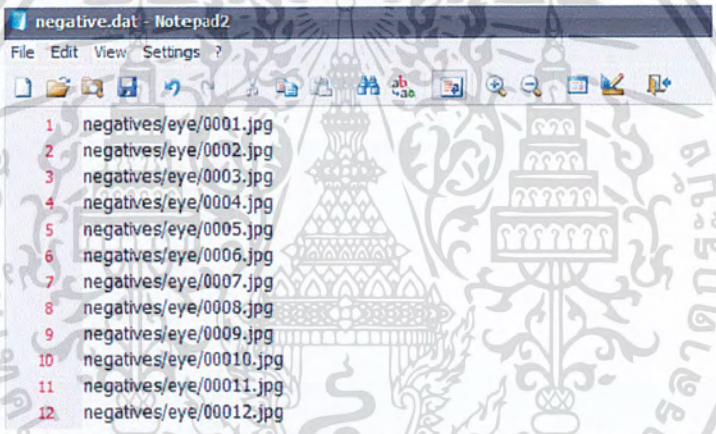
2) เก็บภาพ Positive จากกล้องวีดีโอ อาจใช้รูปบางส่วนจากกล้องดิจิทัลบ้าง แต่ควรมีอัตราส่วนของภาพจากกล้องวีดีโอมากกว่า เนื่องจากภาพมีความละเอียดที่ใกล้เคียงกัน

3) คัดภาพ Positive และภาพ Negative อีกครั้ง แปลงไฟล์ภาพให้เป็น .jpg ทั้งหมดใหญ่หรือเล็กกว่านี้ก็ได้ตามความเหมาะสม และตัดบางส่วนของรูปที่ไม่ต้องการทิ้งไปได้ สำหรับภาพ Negative

5.3.2 Training Data

5.3.2.1 ขั้นตอนการเตรียมข้อมูลก่อนการ Training

1) เก็บที่อยู่ของภาพ Negative ที่ใช้สำหรับ Training ไฟล์ .TXT หรือ .DAT สำหรับภาพ Negative



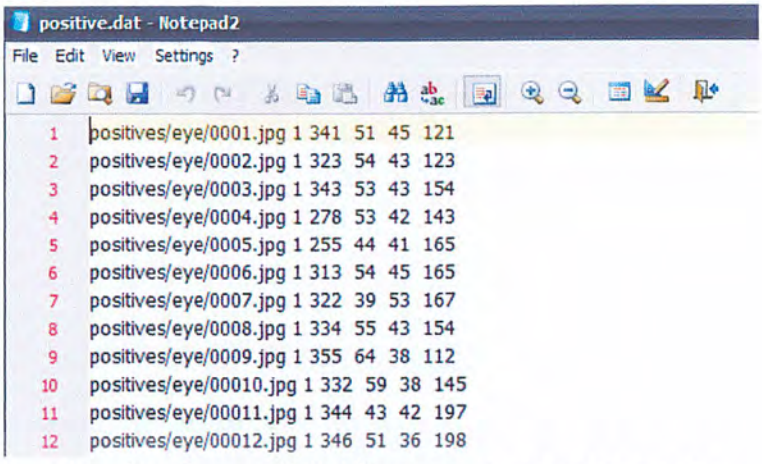
รูป 5.5 ตัวอย่างการสร้างไฟล์ในการ Train ของภาพ Negative

2) เก็บที่อยู่ของภาพ Positive โดยใส่ขนาดข้างหลัง เพื่อระบุตำแหน่งและขนาดของรูปที่ต้องการตรวจจับในภาพนั้นๆ โดยมีรูปแบบดังนี้

Directory_name/image_name.jpg [จำนวนภาพ] [จุดX] [จุดY] [ความกว้าง] [ความยาว]

หากในภาพมีรูปที่ต้องการตรวจจับหลายภาพ ให้ระบุตำแหน่ง และขนาด ของภาพ ต่อท้ายไปเรื่อยๆ ตามตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 5.6 ตัวอย่างการสร้างไฟล์ในการ Train ของภาพ Positive

ในการตีกรอบนี้ ผู้พัฒนาได้ใช้โปรแกรมช่วยในการตีกรอบภาพ เพื่อบอกตำแหน่ง และขนาดของภาพ โดยเก็บข้อมูลทั้งหมดในรูปแบบของไฟล์ .TXT หรือ .DAT เพื่อความสะดวกในการระบุตำแหน่งของภาพ

3) นำไฟล์ที่ได้จากข้อ 2 มาสร้างภาพ Vector โดยใช้โปรแกรมที่มาพร้อมกับ OpenCV ชื่อว่า createsamples.exe โดยใช้คำสั่งดังนี้

```
createsamples -info stop3.dat -vec stop3.vec -w 20 -h 20
```

โดยที่ : -info คือ ไฟล์ที่เก็บที่อยู่และตำแหน่งของภาพ Positive ที่ได้จากข้อ 2
-vec คือ ชื่อไฟล์ .VEC ที่ต้องการเก็บ
-w คือ ความกว้างของภาพ Vector
-h คือ ความสูงของภาพ Vector

ขนาดมาตรฐานในการสร้างภาพ Vector นั้นถูกกำหนดไว้ที่ 24x24 พิกเซล แต่ในโครงงานนี้เลือกใช้ขนาด 20x20 พิกเซล ตามที่ Naotoshi [9] และ Viola กับ Jones ได้กล่าวไว้ว่า ภาพขนาด 20x20 พิกเซลนั้น เป็นขนาดที่เหมาะสมที่สุดสำหรับการตรวจจับรูปที่มีหลายๆขนาด เราจะนำไฟล์ Vector ดังกล่าวไปใช้ในการ Training ต่อไป

5.3.2.2 การ Training

ในการ Training นี้ เราจะนำไฟล์ Vector ที่ได้จากข้อ 3 มา Train รวมกับภาพ Negative ที่เก็บที่อยู่ของภาพไว้ในในไฟล์ .TXT หรือ .DAT โดยใช้โปรแกรม Haar-training ที่มาพร้อมกับ OpenCV เพื่อสร้างไฟล์ XML ที่เก็บ Tree รูปแบบหนึ่ง ที่ใช้สำหรับการทำตรวจจับภาพ โดยมีรูปแบบตัวอย่างคำสั่งดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

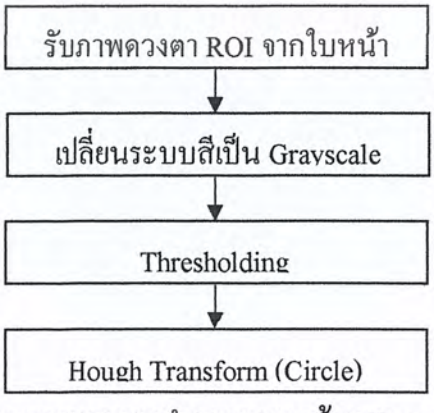
haartraining -data nopark_cascade -vec nopark.vec -bg negatives/train.dat -nstages 20 -npos 1000 -nneg 9650 -w 20 -h 20 -nonsym -mem 1024 -mode ALL

โดยที่: -data คือ ตำแหน่งของไฟล์ที่เก็บข้อมูลที่ได้จากการ Training
-vec คือ ชื่อตำแหน่งของไฟล์ .VEC ที่จะใช้ Train
-bg คือ ตำแหน่งของไฟล์ที่เก็บข้อมูลของภาพ Negative ไว้
-nstage คือจำนวน Stage ที่จำ Train
-npos คือ จำนวนของภาพ Positive
-neg คือจำนวนของภาพ Negative
-w คือ ความกว้างของภาพ Vector ของไฟล์ Vector ที่ใช้
-h คือ ความสูงของภาพ Vector ของไฟล์ Vector ที่ใช้
-nonsym ให้กำหนดไว้หากภาพ Positive ที่ Train มีลักษณะไม่สมมาตร ลักษณะสมมาตรไม่จำเป็นต้องใส่ค่านี้
-mem คือ ปริมาณ Memory ที่ใช้ในการ Train
-mode คือ โหมดในการ Train

5.4 การตรวจหาดวงตาจากใบหน้าด้วย Haar like-feature

ในขั้นตอนการหาดวงตาจากภาพใบหน้าทั้งหมดนั้นต้อง นำไฟล์ XML ที่ได้ จากกระบวนการ Training มาใช้ตรวจจับหาลูกตา ภายในภาพนำเข้า โดยใช้หลักการของ Sub window mask เคลื่อนที่หาลักษณะที่เหมือนกัน ตามหลักการทำ Haar Classifier ซึ่งเมื่อหาดวงตาไม่พบนั้น โปรแกรมจะเพิ่มขนาดของหน้าต่างขึ้นไปเรื่อยๆ จนกว่าจะพบหรือจนกว่าจะขยายไม่ได้อีกแล้ว

5.5 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform



รูป 5.7 ขั้นตอนการหาลูกตาดำจากดวงตาด้วย Hough Transform

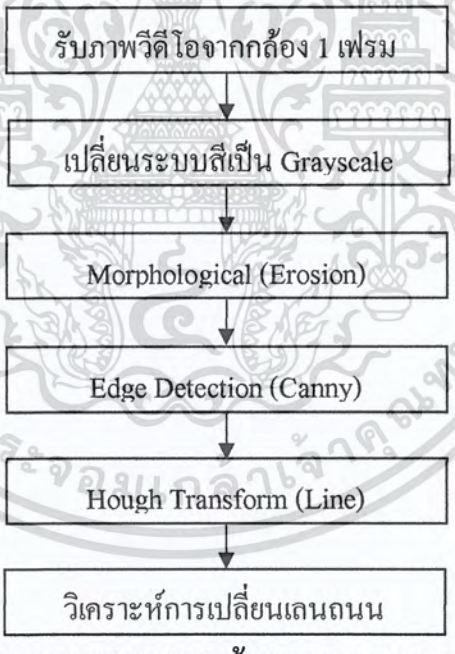
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่ได้บริเวณของดวงตาที่ตรวจจับได้แล้วนั้น (ROI) เราจะตัดเอาเฉพาะบริเวณดังกล่าวมาวิเคราะห์ต่อด้วยการเปลี่ยนภาพมาเป็นระบบสี Grayscale ก่อนที่จะทำ Thresholding เพื่อที่จะสามารถตรวจสอบรูปร่างของลูกตาดำว่าเป็นวงกลมหรือไม่ เพราะถ้าเป็นวงกลมแสดงว่าผู้ขับขี่กำลังลืมตาอยู่ แต่หากว่าเป็นรูปร่างอื่น เช่น เส้นโค้งหรือเส้นตรงแสดงว่าผู้ขับขี่กำลังหลับตา

5.6 การวิเคราะห์อาการหลับในจากการลืมตาและหลับตา

โปรแกรมจะมีการจับเวลาขณะที่ผู้ขับขี่ลืมตา (หาลูกตาดำพบรูปร่างเป็นวงกลม) และหลับตาอยู่ (หาลูกตาดำพบรูปร่างอื่นนอกเหนือจากวงกลม) หากระบบพบว่าผู้ขับขี่มีการหลับตาหรือลืมตายาวนานเกิน 4 วินาทีแสดงว่าผู้ขับกำลังเหม่อลอยและมีโอกาสหลับในอยู่ระบบจะแจ้งเตือนผู้ขับขี่ด้วยเสียง เนื่องจากโดยปกติแล้วผู้ขับขี่รถยนต์ที่จะต้องมีการกระพริบตา 16 ครั้งใน 1 นาที ขณะขับรถยนต์

5.7 การทำงานของระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลน



รูป 5.8 ขั้นตอนการหาเลนถนนด้วย Hough Transform

เมื่อรับภาพวีดีโอเลนถนนจากกล้องมา 1 เฟรม จะเปลี่ยนภาพมาเป็นระบบสี Grayscale หลังจากนั้นทำกระบวนการ Morphological Image Processing ด้วยวิธีการ Erosion เพื่อลด Noise บริเวณพื้นเลนถนนออกไป ทำกระบวนการ Edge Detection ด้วยวิธีกร Canny เพื่อที่จะหาขอบภาพออกมาให้ชัดเจนที่สุด แล้วเอาภาพขอบที่ได้ไปผ่านฟังก์ชั่น Hough Transform เพื่อหาเส้นจากภาพขอบที่มีอยู่แล้วนำไปวิเคราะห์ว่าผู้ขับขี่กำลังเปลี่ยนเลนอยู่หรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.8 อุปกรณ์และระบบ

5.8.1 อุปกรณ์ฮาร์ดแวร์

1) เครื่องคอมพิวเตอร์ใช้สำหรับพัฒนาโปรแกรม และประมวลผล

2) กล้องเว็บแคมสามารถถ่ายภาพเวลากลางคืน จำนวน 2 ตัว

5.8.2 ซอฟต์แวร์

1) ซอฟต์แวร์นี้ทำงานบนระบบปฏิบัติการWindow XP

2) Visual Studio C++ 2008 สำหรับพัฒนาโปรแกรม

3) NET framework 3.5 เป็นแพลตฟอร์มสำหรับการพัฒนาโปรแกรม

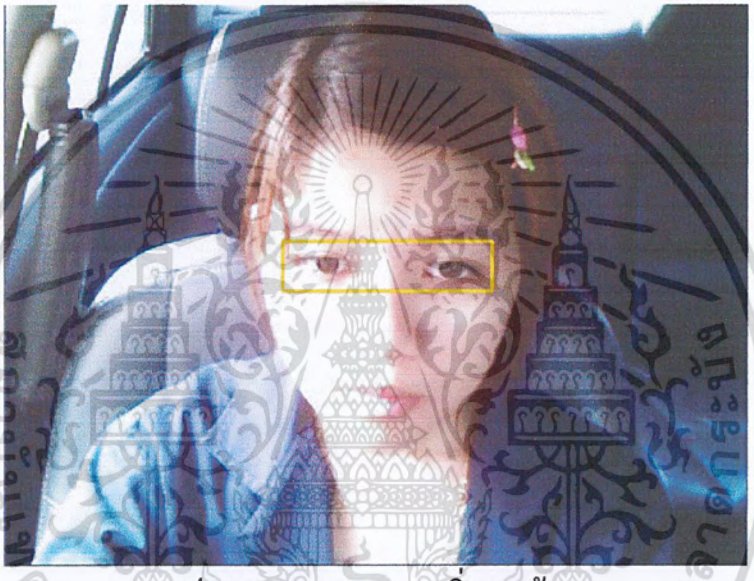
4) OpenCV 2.1 เป็นไลบรารี่โปรแกรมสำหรับการพัฒนาโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

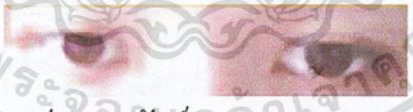
บทที่ 6
ผลการทดลอง

6.1 การหาดวงตาจากใบหน้า

การวิเคราะห์อาการหลับในจากดวงตา จะต้องหาดวงตาผู้ขับขี่รถยนต์จากใบหน้าและสิ่งแวดล้อมทั้งหมดออกมาให้ได้เป็นสิ่งแรก เพื่อภาพเฉพาะดวงตาไปวิเคราะห์อาการหลับในต่อไป ซึ่งการหาดวงตานั้นใช้วิธีการ Haar like-feature ในการหา ROI คือ ดวงตา



รูป 6.1 การหาดวงตาจากสิ่งแวดล้อม

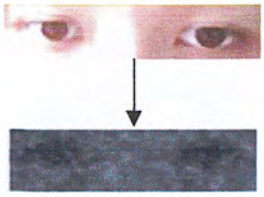


รูป 6.2 ผลลัพธ์ของการหาดวงตา

6.2 การเปลี่ยนระบบสีของดวงตาจาก RGB เป็น Grayscale

หลังจากที่หาดวงตาจากสิ่งแวดล้อมทั้งหมดได้แล้ว จะต้องนำดวงตาที่ตัดออกมาจากภาพทั้งหมด มาเปลี่ยนระบบสีเป็น Grayscale เพื่อเตรียมที่จะทำกระบวนการ Thresholding ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 6.3 การเปลี่ยนระบบสีของดวงตาจาก RGB เป็น Grayscale

6.3 การทำ Thresholding ภาพดวงตา

หลังจากที่ได้ดวงตาในระบบสี Grayscale แล้วนั้น จะต้องทำ Thresholding ให้ภาพที่มีหลากหลายระดับสีเหลือเพียงแต่ 2 ระดับ คือ ขาวและดำ นั่นคือการทำให้ภาพเปลี่ยนเป็นระบบสีแบบ Binary เพื่อที่จะให้ระดับสีในดวงตาที่สนใจเด่นชัดขึ้นมา

รูป 6.4 การทำ Thresholding ภาพดวงตา

6.4 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform (Circle)

กรณีที่ 1 ตรวจจับลูกตาดำพบในกรณีปกติ (ขับรถทั่วไป)

รูป 6.5 ชุดของตาในกรณีปกติ (ตอนกลางวัน)

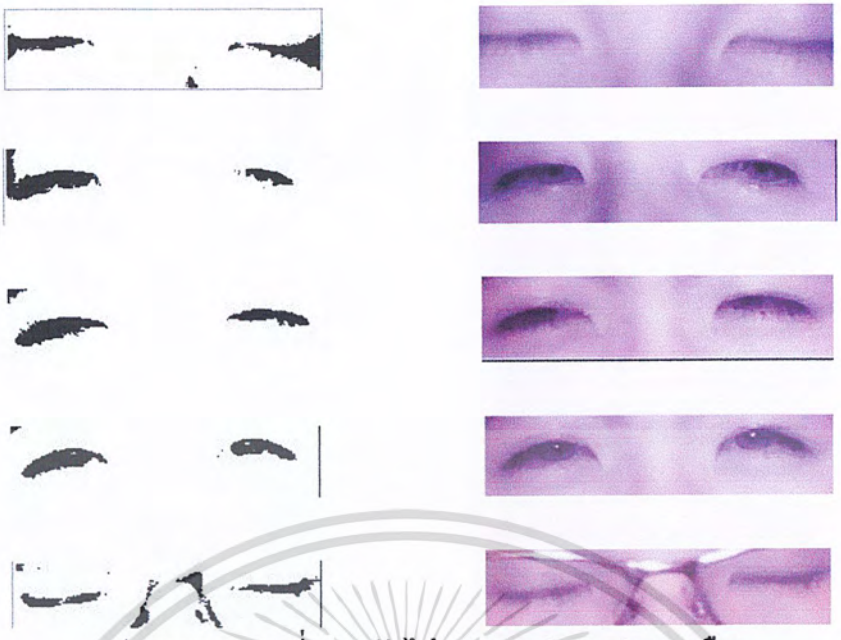
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 6.6 ชุดของตาในกรณีปกติ (ตอนกลางคืน)

กรณีที่ 2 ตรวจจับไม่พบลูกตาดำในกรณีหลับในหรือเหม่อลอย

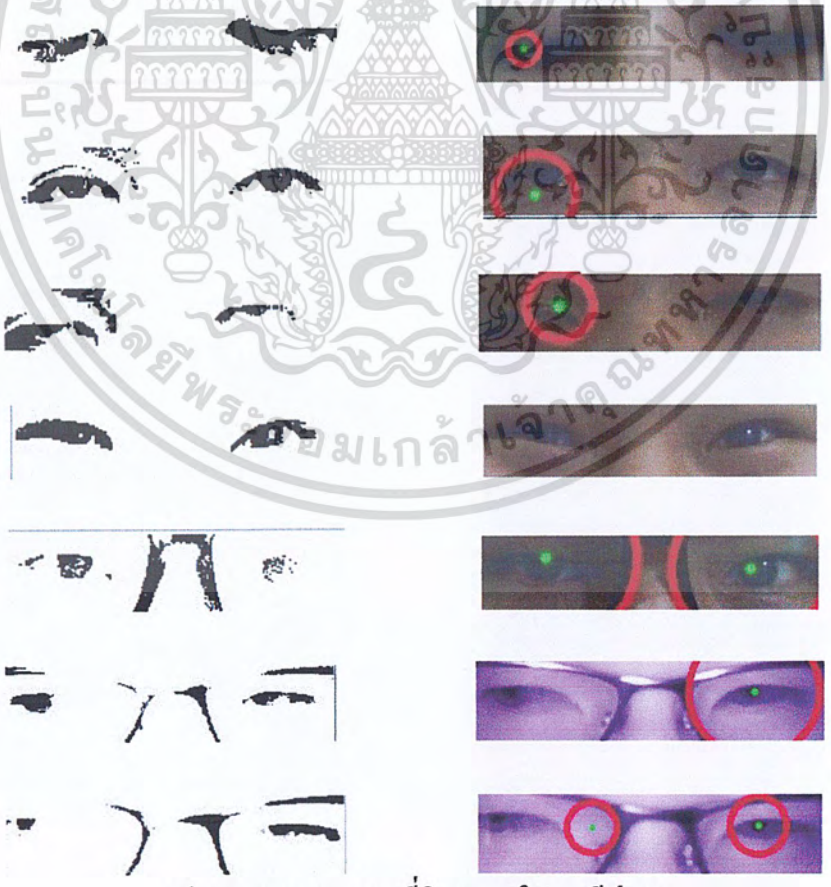
รูป 6.7 ชุดของตาที่ตรวจจับไม่พบลูกตาดำ (ตอนกลางวัน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 6.8 ชุดของตาที่ตรวจจับไม่พบลูกตาดำ (ตอนกลางคืน)

กรณีที่ 3 ผิดพลาดในกรณีต่างๆ



รูป 6.9 ชุดของดวงตาที่ผิดพลาดในกรณีต่างๆ

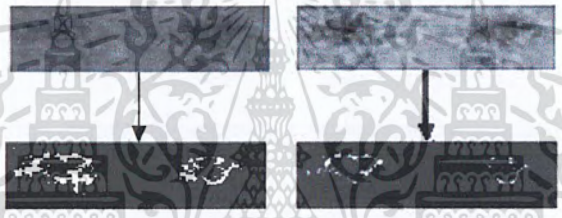
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การผิดพลาดในกรณีที่เหม่อลอยหรือหลับตาไปแล้วแต่ยังสามารถตรวจจับลูกตาดำได้นั้น เกิดจากการที่โปรแกรมจะทำการตรวจจับสิ่งที่เป็นวงกลมจากภาพของดวงตาที่ทำการ Thresholding ซึ่งในกรณีที่ผู้ขับขี่ใส่แว่นตาหรือผู้ขับขี่มีคิ้วที่หนาหรือคิ้วชนกัน ก็อาจให้เกิดภาพ Thresholding ที่เป็นวงกลมได้

6.5 การตรวจหาลูกตาดำจากดวงตาด้วยวิธีอื่นๆ

6.5.1 การตรวจหาลูกตาดำจากดวงตาด้วยวิธีวัดปริมาน Pixel

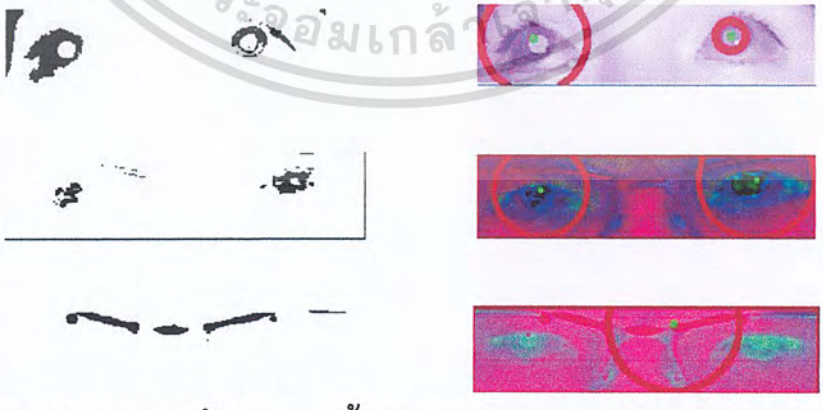
การตรวจหาลูกตาดำจากดวงตาด้วยการทำ Thresholding แล้ววัดปริมาน Pixel ที่มีอยู่นั้นไม่เหมาะสมกับการตรวจจับดวงตาผู้ขับขี่รถยนต์บนถนน เพราะการขับขี่รถยนต์ในแต่ละเวลาและสถานที่มีค่าแสงจากภายนอกที่ไม่เท่ากัน ทำให้ปริมาน Pixel ที่วัดได้ไม่คงที่ จึงไม่สามารถกำหนดได้ว่ามีมีปริมาณ Pixel ที่เท่าไหร่จึงจะเป็นลืมตาหรือหลับตา



รูป 6.10 การตรวจหาลูกตาดำดวงตาด้วยวิธีวัดปริมาน Pixel

จากรูป 6.10 จะเห็นได้ว่าเป็นตาของผู้ขับขี่คนเดิม แต่แสงจากภายนอกต่างกัน ทำให้ปริมาณ Pixel ของตาต่างกันไปด้วย

6.5.2 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform (Circle) แบบระบบสี HSV



รูป 6.11 การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform (Circle) แบบระบบสี HSV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การตรวจหาลูกตาดำจากดวงตาด้วย Hough Transform แบบระบบสี HSV จะทำให้ลูกตาดำกลมกลืนกับสีผิว เมื่อผ่านการทำ Thresholding แล้วจะทำให้รูปทรงของลูกตาดำมองเป็นรูปร่างได้ยากขึ้น

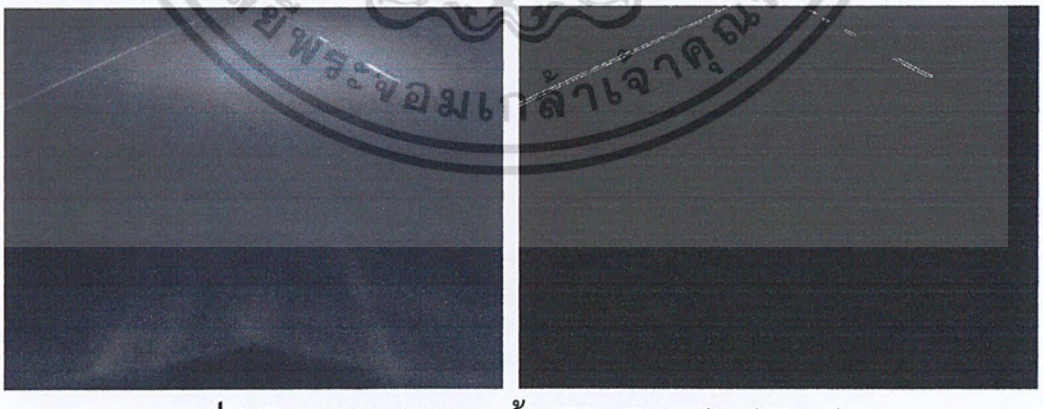
6.6 การปรับปรุงภาพเลนด้วย Morphological Image Processing (Erosion)

เนื่องจากบนเลนถนน อาจมีรอยหินสีขาวหรือคราบน้ำมัน ซึ่งเป็นสิ่งรบกวนปะปนอยู่ การทำ Erosion จะทำให้สิ่งรบกวนเหล่านี้มีขนาดเล็กลง

รูป 6.12 การปรับปรุงภาพเลนด้วย Morphological Image Processing (Erosion)

6.7 การหาขอบภาพเลนด้วย Edge Detection (Canny)

หลังจากนั้น นำภาพที่ผ่านกระบวนการ Erosion มาหาขอบภาพด้วย Edge Detection ซึ่งเลือกใช้ Canny Algorithm และกำหนดค่า Thresholding ที่เหมาะสมให้กับฟังก์ชั่น



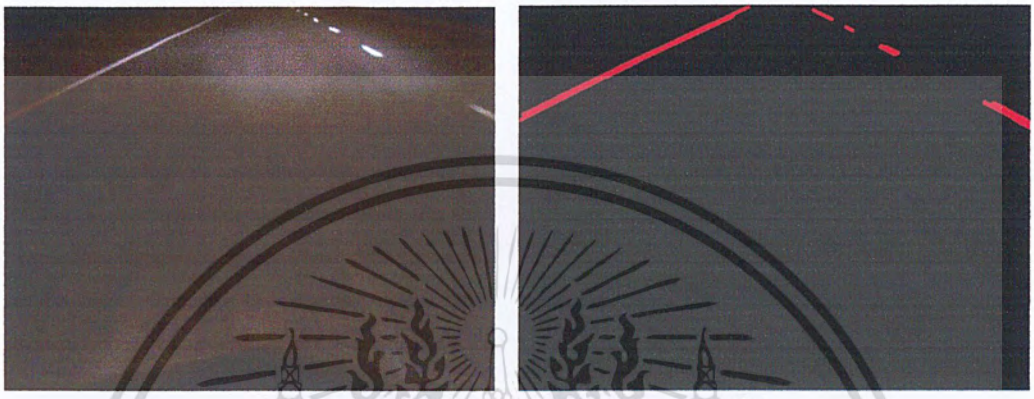
รูป 6.13 การหาขอบภาพเลนด้วย Edge Detection (Canny)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.8 การตรวจหาเลนถนนด้วย Hough Transform (Line)

การตรวจหาเส้นเลนถนนใช้กระบวนการ Hough Transform แบบ Line เพื่อที่จะหาเส้นถนนจากภาพแต่ละเฟรมที่ถ่ายได้

6.8.1 การตรวจหาเลนถนนในเวลากลางคืน



รูป 6.14 การตรวจหาเลนถนนในเวลากลางคืน

6.8.1 การตรวจหาเลนถนนในเวลากลางวัน

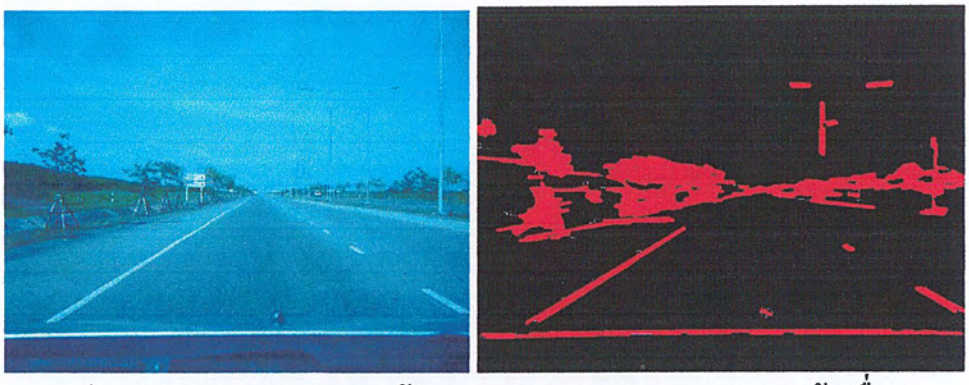
รูป 6.15 การตรวจหาเลนถนนในเวลากลางวัน

6.9 การตรวจหาเลนถนนด้วยวิธีอื่นๆ

6.9.1 การตรวจหาเลนถนนด้วย Hough Transform (Line) มุมกล้องอื่นๆ

หากเลือกใช้มุมกล้องที่เห็นสิ่งแวดล้อม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 6.16 การตรวจหาเลนถนนด้วย Hough Transform (Line) มุมกล้องอื่นๆ

6.9.2 การตรวจหาเลนถนนที่เลือกค่า Thresholding ให้ฟังก์ชั่นไม่เหมาะสม

รูป 6.17 การตรวจหาเลนถนนที่เลือกค่า Thresholding ให้ฟังก์ชั่นไม่เหมาะสม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
สรุป

7.1 สรุป

โปรแกรมวิเคราะห์อาการหลับในแบ่งออกเป็น 2 ระบบ คือ ระบบวิเคราะห์อาการหลับในจากดวงตา และระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนถนน โดยที่วิเคราะห์อาการหลับในจากดวงตาเป็นระบบหลักในการตรวจสอบอาการหลับในของผู้ขับขี่ ซึ่งยังคงมีข้อจำกัดอยู่หลายอย่าง จึงต้องใช้ระบบวิเคราะห์อาการหลับในจากการเปลี่ยนเลนมาช่วยในการตัดสินใจด้วย เพื่อเพิ่มประสิทธิภาพในการตรวจสอบให้แม่นยำมากยิ่งขึ้น ซึ่งในช่วงเวลากลางวันในส่วนจากดวงตาจะมีประสิทธิภาพมากกว่าทั้งในเรื่องของการตรวจจับดวงตาที่ง่ายและตรวจพบลูกตาดำได้ดีกว่า ส่วนจากเลนถนนนั้นในตอนกลางวันเนื่องจากแสงที่มีมากทำให้เกิดปัญหาของ noise ที่มากเช่น คราบน้ำมัน ก้อนดิน ฯลฯ แต่ในตอนกลางคืนส่วนของเลนถนนมีความชัดเจนและมองเห็น noise ได้ยาก

7.2 ปัญหาและอุปสรรค

ระบบวิเคราะห์อาการหลับในจากดวงตามีปัญหาและอุปสรรคดังนี้

1) ระบบนี้ไม่สามารถใช้ได้กับผู้ขับขี่ที่มีตาเล็กเกินไปรวมถึงผู้ที่ใส่แว่นกันแดดและแว่นสายตา
2) ผู้ขับขี่ที่มีคิ้วหนาหรือคิ้วชนกันอาจทำให้ระบบตรวจจับวงกลมจากคิ้วแทนที่จะเป็นลูกตาดำได้
3) หากสภาพแวดล้อมของผู้ขับขี่ไม่เหมาะสม อาจส่งผลต่อการตรวจจับและการวิเคราะห์อาการหลับในที่ผิดพลาดได้
4) ตำแหน่งการวางกล้องต้องวางในตำแหน่งที่เหมาะสม ซึ่งแต่ละผู้ขับขี่มีรูปร่างที่แตกต่างกัน

ระบบวิเคราะห์อาการหลับในจากเลนมีปัญหาและอุปสรรคดังนี้

1) แสงที่มากเกินไปในตอนกลางวันทำให้ระบบตรวจจับ noise ได้มากจนเกินไป เช่น คราบน้ำมัน เศษก้อนหิน เศษดิน ฯลฯ
2) ถนนบางเส้นทางมีเส้นถนนที่ไม่ชัดเจน ยากต่อการตรวจจับหาเลนถนน

7.3 แนวทางการแก้ไขและการพัฒนาต่อ

ระบบวิเคราะห์อาการหลับในจากดวงตามีแนวทางการแก้ไขและการพัฒนาต่อดังนี้

1) ควรพัฒนาระบบให้เหมาะสมกับผู้ขับขี่ที่มีดวงตาที่เล็กจนเกินไป
2) หาทางแก้ไขในกรณีที่ผู้ขับขี่ใส่แว่นตาในการขับรถและผู้ที่มีคิ้วหนามากเกินไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ควรพัฒนาระบบให้มีความยืดหยุ่นตามสภาพแวดล้อมของผู้ขับขี่ในกรณีต่างๆ เช่น อาจมีการปรับค่าต่างๆของโปรแกรมได้
4) ติดตั้งกล้องให้มีการปรับตำแหน่งของกล้องให้เหมาะสมกับผู้ขับขี่นั้นๆได้ทุกครั้งไป
5) พัฒนาอัลกอริธึมในการตรวจหาดวงตาในตอนกลางคืนให้มีความแม่นยำและมีประสิทธิภาพมากยิ่งขึ้น

ระบบวิเคราะห์อาการหลับในจากเลนมีแนวทางการแก้ไขและการพัฒนาต่อดังนี้

1) ควรพัฒนาระบบให้มีความยืดหยุ่นตามสภาพแวดล้อมของผู้ขับขี่ในกรณีต่างๆ
2) กำจัด noise บนถนนเวลากลางวันให้หมดไป โดยไม่ส่งผลต่อเลนถนน
3) พัฒนาอัลกอริธึมให้มีความสามารถและถูกต้องแม่นยำให้ดียิ่งขึ้น

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

Gary Bradski and Adrian Kaehler. 2008. **LearningOpenCV.** Sebastopol : OReilly.

ผศ.ดร.อรฉัตร จิตต์โสภักตร. 2552. **Digital Image Processing.** กรุงเทพฯ : บริษัท สงวนกิจ พริ้นท์ มีเดีย.

Murray W, John. 2550. A Sleep **Physiologist View of The Drowsy Driver.** Melbourne : Pergamon.

ฐิตวัฒน์ เตชจรัสชีวิน. **ระบบติดตามลักษณะเด่นบนใบหน้า โดยใช้กล้องเพียงหนึ่งตัว.** [Online]. Available : http://161.200.126.55/~achatcha/SeniorProject2552/Comp

นิพัฒน์ เอี่ยมสมบูรณ์. **OpenCV.** [Online]. Available : http://rabbitclan.com/blog/?cat=9

Adam Harvey. **OpenCV Face detection.** [Online]. Available : http://ahprojects.com/c/itp/thesis

Matthias Wimmer. **OpenCV Eye detection.** [Online]. Available : http://nashruddin.com/OpenCV_Eye_Detection

Gary Bradski. **Face Detection using OpenCV.** [Online]. Available : http://opencv.willowgarage.com/wiki/FaceDetection

Fisher, S. Perkins, A. Walker and E. Wolfart. **Hough Transform.** [Online]. Available : http://homepages.inf.ed.ac.uk/rbf/HIPR2/hough.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
การติดตั้งและวิธีใช้โปรแกรมที่ใช้พัฒนา

ก.1 การติดตั้งโปรแกรม Visual Studio C++ 2008

โปรแกรม Visual Studio นั้นเป็นโปรแกรมที่ใช้พัฒนาระบบในด้านของการประมวลผล ซึ่งโปรแกรมนี้ประกอบไปด้วยโปรแกรมย่อยอีกมากมาย ขึ้นกับภาษาที่ผู้ใช้ต้องการพัฒนา ในที่นี้จะแสดงเพียงวิธีการติดตั้งโปรแกรม Visual Studio C++ เท่านั้น ซึ่งมีขั้นตอนดังนี้

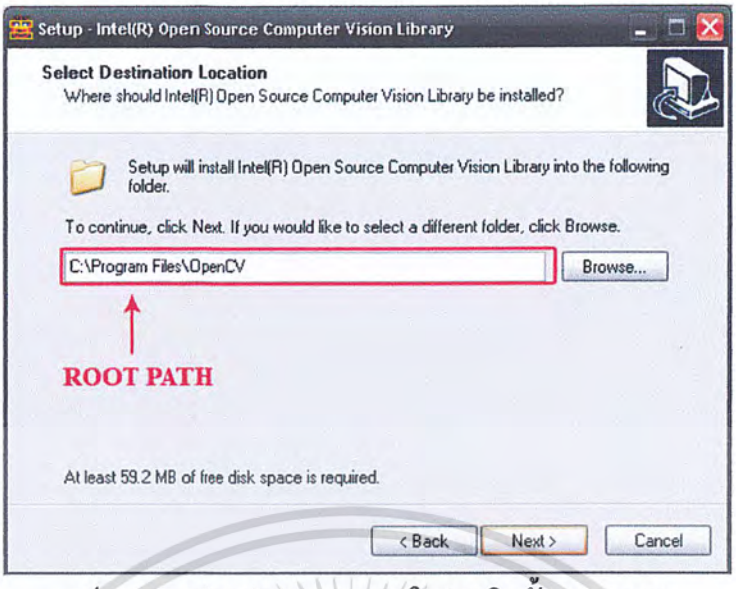
1) คลิกที่ Install Visual Studio 2008 เพื่อลงโปรแกรม เมื่อหน้าต่างโปรแกรมเปิดขึ้นรอสักครู่
2) เลือก I accept the term of the License Agreement ใส่ Serial Number แล้วกำหนดชื่อของผู้ลงโปรแกรม โดยค่ามาตรฐานจะมีค่าเดียวกับชื่อ User ของผู้ใช้ของระบบปฏิบัติการ Window
3) กำหนดไดเร็กทอรี่ที่ต้องการลงโปรแกรม และเลือกลงแบบ Custom เพื่อลงเฉพาะ Visual Studio C++ เท่านั้น (ผู้ใช้สามารถเลือกลงแบบอื่นได้ตามต้องการ)
4) เลือกโปรแกรมที่จะลงตามต้องการ หากต้องการลงเฉพาะโปรแกรม Visual Studio C++ เท่านั้น ให้เลือกเหมือนดังภาพ แล้วคลิก Install เพื่อลงโปรแกรม
5) เมื่อลงโปรแกรมเสร็จกด Finish เพื่อสิ้นสุดการลงโปรแกรม

ก.2 การติดตั้ง OpenCV 2.1

โปรแกรม OpenCV 2.1 เป็น Library ในภาษา C++ และ Phyton สำหรับการพัฒนาโปรแกรมที่เกี่ยวข้องกับ Image Processing และ Computer Vision โดยสามารถพัฒนาได้ทั้งในระบบปฏิบัติการวินโดว์ และ ระบบปฏิบัติการ Linux มีวิธีการติดตั้งดังนี้

1) คลิกที่ไอคอนเพื่อเริ่มติดตั้งโปรแกรม
2) เมื่อถึงหน้า License Agreement เลือก I accept the agreement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.1 การกำหนด Root Path ในการติดตั้ง OpenCV

3) โปรแกรมจะให้เลือกตำแหน่งที่จะ Install โดยค่ามาตรฐานจะถูกตั้งไว้ที่ C:/Program Files/ OpenCV/ ซึ่ง Path นี้ จะนำไปใช้เป็น Root Path ที่จะนำไปใช้ในส่วนของการใช้งาน OpenCV ร่วมกับโปรแกรม Visual Studio C++

4) ในขั้นตอนนี้โปรแกรมจะให้ผู้ใช้สามารถกำหนดชื่อของโฟลเดอร์ที่จะปรากฏใน Start Menu ได้ โดยค่ามาตรฐานของโปรแกรมกำหนดไว้เป็น OpenCV

5) เลือก Add<...>\OpenCV\bin to the system PATH เพื่อเพิ่ม PATH ของไฟล์.dll ของ OpenCV ลงไปใน System PATH ของระบบปฏิบัติการ

6) โปรแกรมจะแสดงหน้าต่างรายละเอียดที่ได้ตั้งไว้ในขั้นตอนที่ 3-5 ตรวจเช็ครายละเอียดแล้ว คลิก Install เพื่อลงโปรแกรม

7) เมื่อลงโปรแกรมเสร็จเรียบร้อยแล้ว ให้คลิก Finish เพื่อสิ้นสุดการลงโปรแกรม

ก.3 วิธีใช้งาน OpenCV ร่วมกับโปรแกรม Visual Studio C++

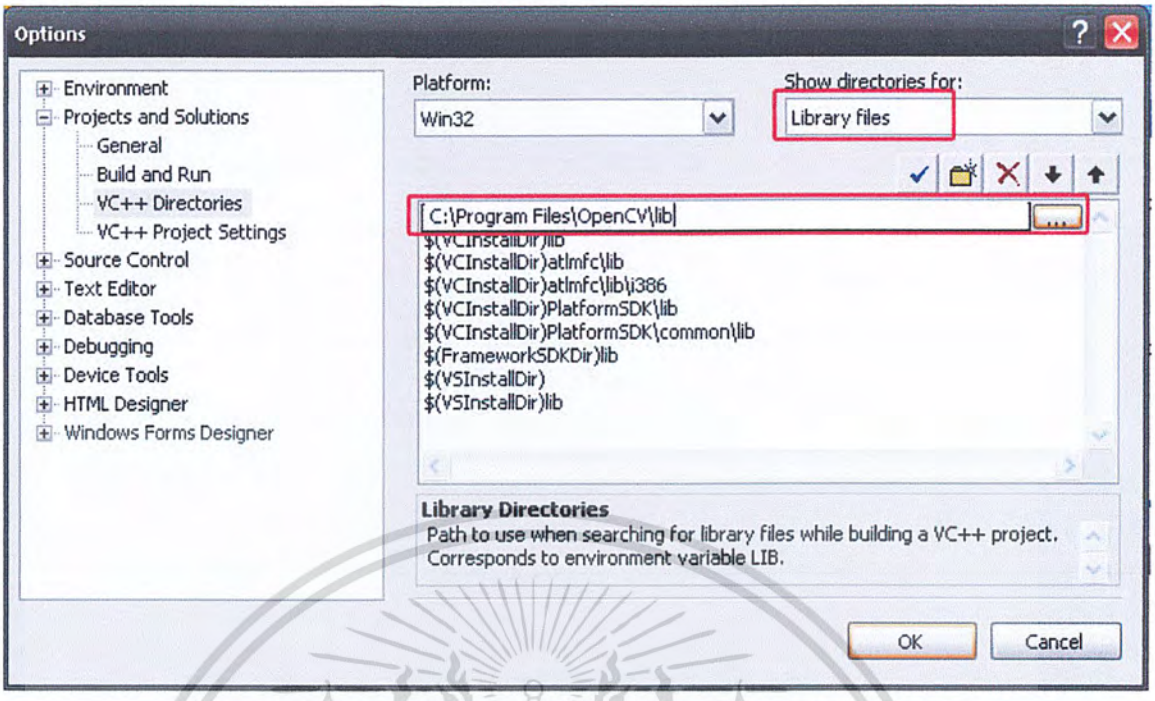
ในการนำ OpenCV มาใช้งานร่วมกับ Visual Studio C++ เพื่อใช้พัฒนาโปรแกรมนั้น เราจำเป็นต้องตั้งค่าให้กับโปรแกรม Visual Studio C++ ก่อน เพื่อระบุตำแหน่งของ Library ของ OpenCV ตำแหน่งของ ไฟล์ที่ต้องใช้ในโปรแกรม และ ตำแหน่งของ Source File ให้ตัวโปรแกรมทราบและสามารถดึงมาใช้ได้ โดยมาวิธีการตั้งค่าและการใช้งานดังนี้

1) เปิดโปรแกรม Visual Studio C++

2) ไปที่เมนู Tool > Options ในแถบเมนูข้างบน

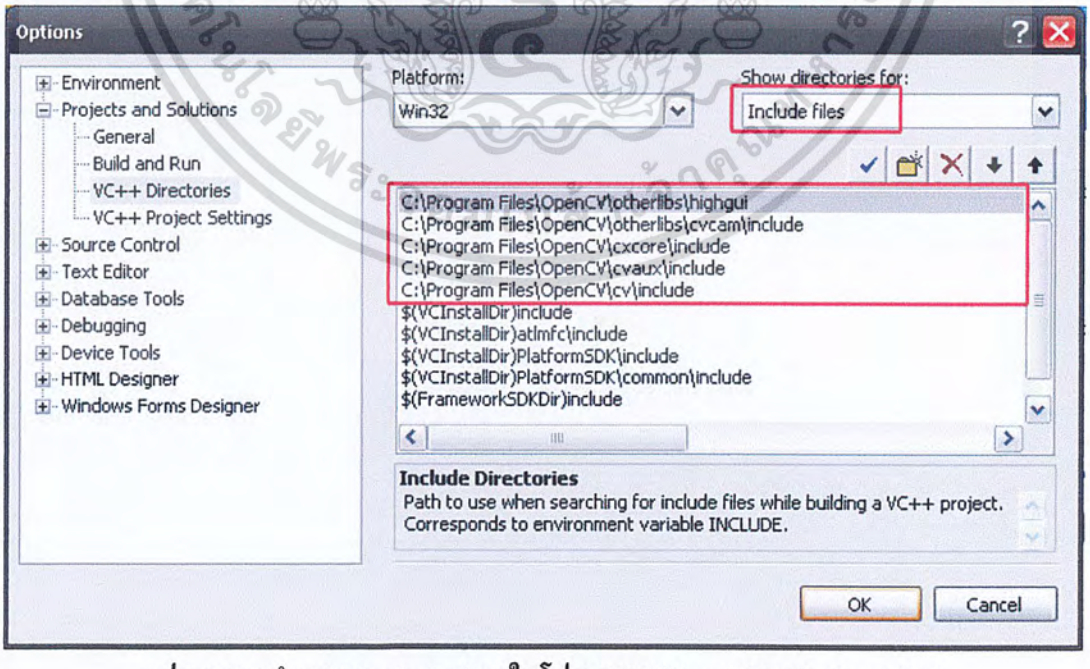
3) ที่หน้าต่าง Project and solutions ขยายเมนูย่อย แล้วเลือกที่ VC++ Directories

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.2 การกำหนด Library files ในโปรแกรม Visual Studio C++ 2008

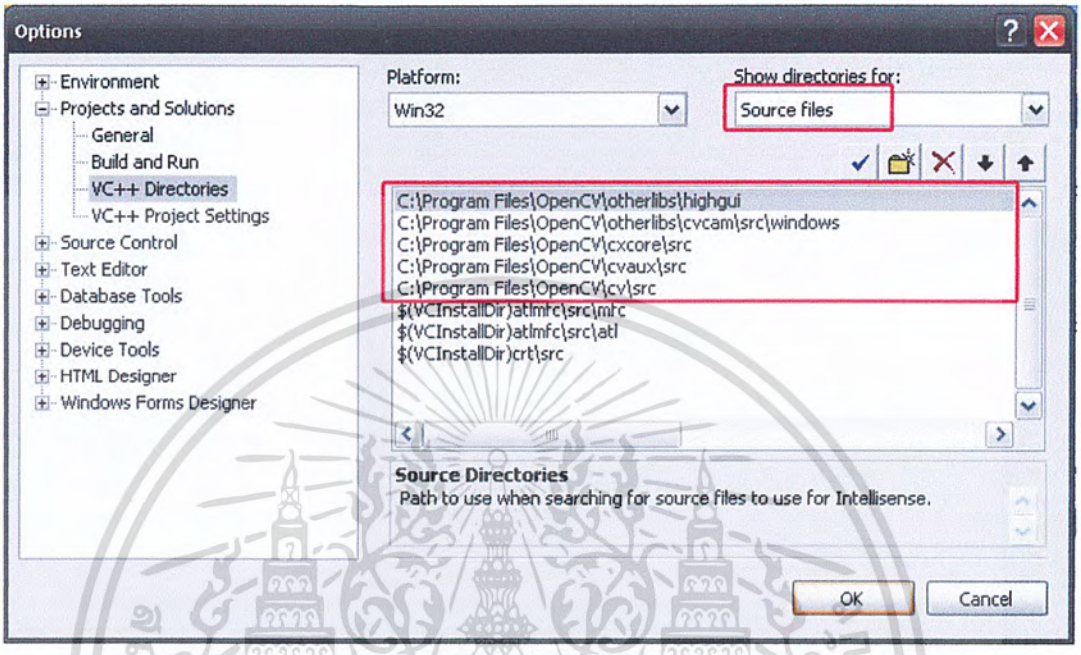
ที่ Show Derectories for: เลือก Library Files จากนั้นกดที่ไอคอน เพิ่ม Path [root path]/ lib โดยที่ [root path] คือ ไดเร็คทอรี่ที่เรา Install openCV ไว้ จากที่ได้กล่าวไปแล้วในหัวข้อการติดตั้งโปรแกรม OpenCV ซึ่งตามค่ามาตรฐานของโปรแกรมแล้ว [root path] จะเป็น C:/Program Files/ OpenCV/ ดังนั้นต้องเพิ่ม Path เป็น C:/Program Files/OpenCV/lib เป็นต้น



รูป ก.3 การกำหนด Include files ในโปรแกรม Visual Studio C++ 2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ Show Derectories for: เลือก Include Files จากนั้นกดที่ไอคอน เพื่อเพิ่ม Path [root path]/cv/include, [root path]/cvavx/include, [root path]/cxcore/include, [rootpath]/otherlibs/cvcam/include และ [rootpath]/otherlibs/highgui



รูป ก.4 การกำหนด Source files ในโปรแกรม Visual Studio C++ 2008

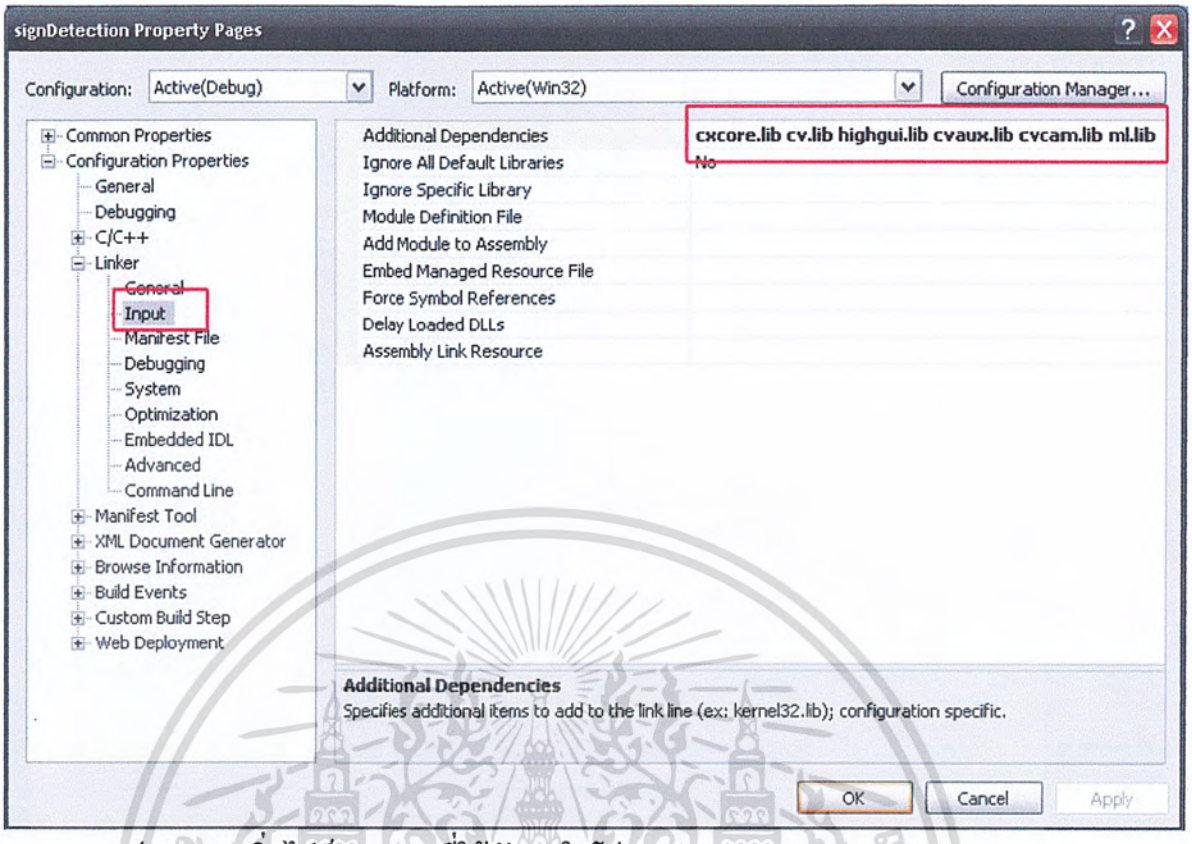
ที่ Show Derectories for: เลือก Source Files จากนั้นกดที่ไอคอน เพื่อเพิ่ม Path [root path]/cv/src, [root path]/cvavx/src, [root path]/cxcore/src, [root path]/otherlibs/cvcam/src/windows, [root path]/otherlibs/highgui

4) กด OK แล้วสร้าง Project ใหม่ หรือ โปรเจ็คที่มีอยู่แล้ว และเป็น C++ Project

5) เลือกเมนู Project > Properties ที่แถบเมนู

6) ที่หน้าต่าง Configuration Properties เลือกเมนูย่อย Linker แล้วเลือก Input

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ก.5 การเพิ่มไฟล์ Library ที่ใช้พัฒนาในโปรแกรม Visual Studio C++ 2008

7) ที่ Additional Dependencies ใส่ cxcore.lib cv.lib highgui.lib cvaux.lib cvcam.lib ml.lib เพื่อกำหนด Library ที่จะใช้ในโปรเจ็คนี้

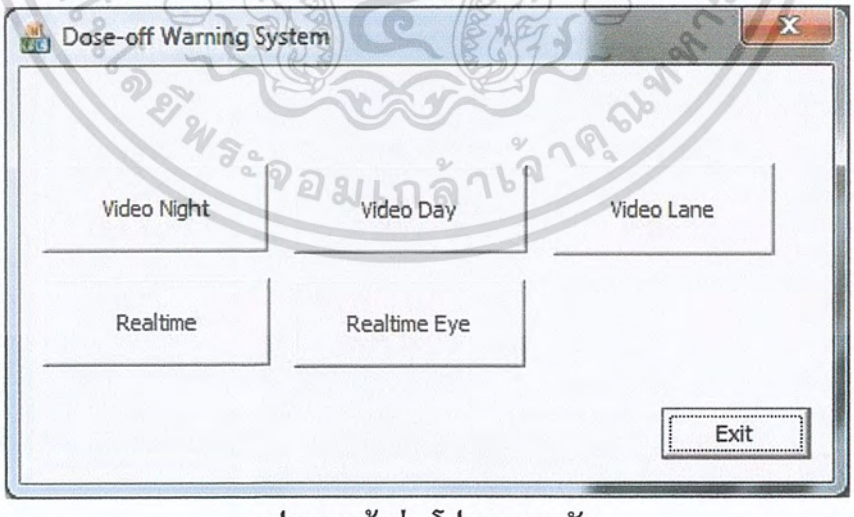
8) ทดสอบผลการติดตั้ง โดยลองรันโปรแกรมที่มีการเรียกใช้ OpenCV อยู่แล้ว หากไม่สามารถรันโปรแกรมได้ โดยมีการฟ้อง Error ไม่พบไฟล์ .dll ให้ผู้ใช้คัดลอกไฟล์ใน [root path]/opencv/bin เฉพาะไฟล์สกุล .dll ไปใส่ใน C:/windows/system

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข
วิธีการใช้โปรแกรมที่พัฒนาขึ้น

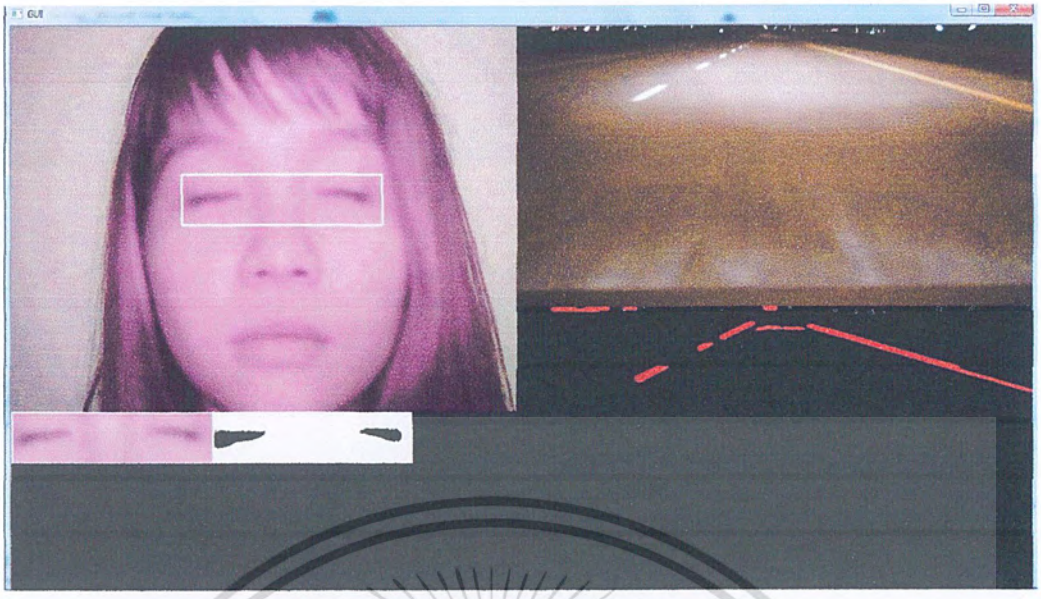
ข.1 วิธีการใช้โปรแกรม

1) เปิดโปรแกรม Dose-off Warning System
2) ปรับตั้งกล้องให้เหมาะสมต่อตัวผู้ขับขี่
3) หากเลือก เมนู Video Night จะเป็นการจำลองการทำงานของโปรแกรม โดยเป็นการเดินทางในเวลากลางคืน โดยการวิเคราะห์ไฟล์วีดีโอของดวงตาและไฟล์วีดีโอของเลนถนน
4) หากเลือก เมนู Video Day จะเป็นการจำลองการทำงานของโปรแกรม โดยเป็นการเดินทางในเวลากลางวัน โดยการวิเคราะห์ไฟล์วีดีโอของดวงตาและไฟล์วีดีโอของเลนถนน
5) หากเลือก เมนู Video Lane จะเป็นการจำลองการทำงานของโปรแกรม โดยเป็นการเดินทางโดยใช้การวิเคราะห์ไฟล์วีดีโอของเลนถนนเพียงอย่างเดียว
6) หากเลือก เมนู Realtime จะเป็นการใช้โปรแกรมจากกล้องทั้งดวงตาและเลนถนน
7) หากเลือก เมนู Realtime Eye จะเป็นการใช้โปรแกรมจากกล้องดวงตา
8) เมื่อโปรแกรมวิเคราะห์ว่ามีอาการหลับใน จะมีเสียงเตือนดังขึ้นมา



รูป ข.1 หน้าต่างโปรแกรมหลัก

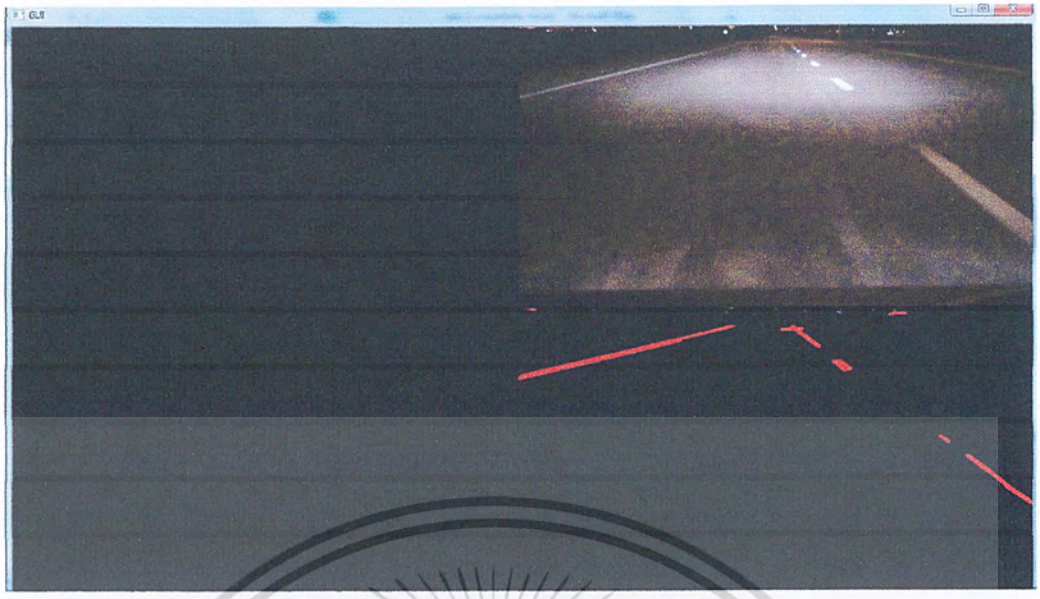
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



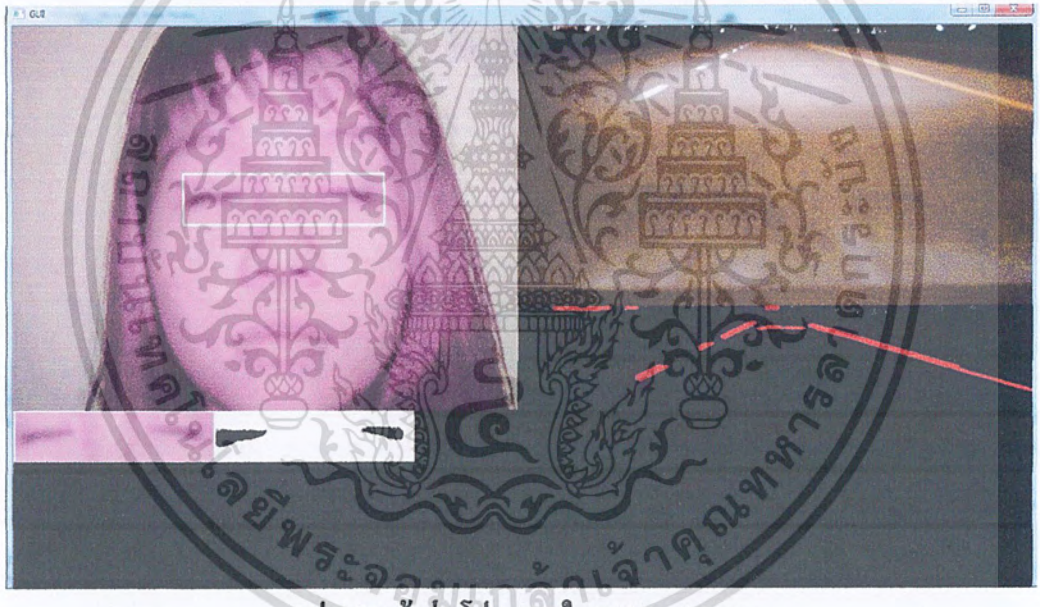
รูป ข.2 หน้าต่างโปรแกรมในเมนู Video Night

รูป ข.3 หน้าต่างโปรแกรมในเมนู Video Day

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

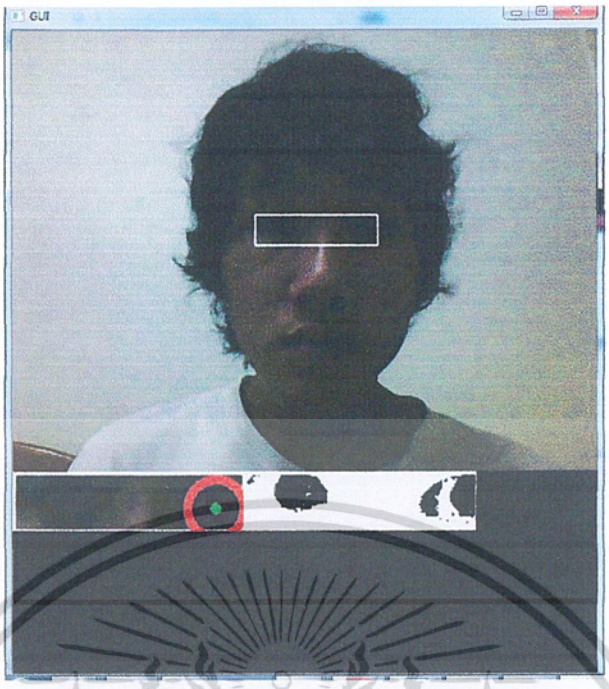


รูป ข.4 หน้าต่างโปรแกรมในเมนู Video Lane



รูป ข.5 หน้าต่างโปรแกรมในเมนู Realtime

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป ข.6 หน้าต่างโปรแกรมในเมนู Realtime Eye

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้