สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

จอสัมผัสอินฟราเรด

INFRARED TOUCH SCREEN



ศิริวรรณ เบ็ญจวรรณ

อาทิตย์ ไชยพอ

เลขหมู่.....
ลงทะเบียน.....119234
วัน,เดือน,ปี.....- 6 S.A. 2554

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จอสัมผัสอินฟราเรด

INFRARED TOUCH SCREEN

ศิริวรรณ เบ็ญจวรรณ

อาทิตย์ ไชยพอ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมอิเล็กทรอนิกส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง จอสัมผัสอินฟราเรด

INFRARED TOUCH SCREEN

ผู้จัดทำ

1. นางสาวศิริวรรณ เบ็ญจวรรณ์ รหัสนักศึกษา 50011569
2. นายอาทิตย์ ไชยพอ รหัสนักศึกษา 50011932



.....อาจารย์ที่ปรึกษา

(อาจารย์สุรพันธุ์ เอื้อไพบูลย์)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จอสัมผัสอินฟราเรด

นางสาว ศิริวรรณ เบ็ญจวรรณ์ รหัส 50011569
นาย อาทิตย์ ไชยพอ รหัส 50011932
รศ.ดร.สุรพันธุ์ เอื้อไพบูลย์ อาจารย์ที่ปรึกษา
ปีการศึกษา 2553

บทคัดย่อ

โครงงานที่จัดทำในครั้งนี้เป็นการศึกษาเกี่ยวกับกระบวนการการประมวลผลภาพโดยการเชื่อมโยงระหว่างมนุษย์กับคอมพิวเตอร์ มีการโต้ตอบกันระหว่างคอมพิวเตอร์กับมนุษย์ ซึ่งจะมีอินฟราเรดเป็นเครื่องมือช่วยในการสั่งงานคอมพิวเตอร์ และจะมีกล้องเป็นตัวจับความเคลื่อนไหวของอินฟราเรดและจดจำตำแหน่งแล้วนำไปประมวลผลเพื่อให้เกิดการสั่งงาน โดยจอสัมผัสที่ใช้จะใช้เป็นการฉายภาพจากเครื่องโปรเจคเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INFRARED TOUCH SCREEN

Ms. Siriwan	Benjawan	ID. 50011569
Mr. Arthit	Chaiyapor	ID. 50011932
Assoc.Prof.Dr.Surapan	Airphaiboon	Advisor

Educational Year 2010

Abstract

Projects conducted in this study is an image processing to affect between a human and a computer, interact with the dialog between a human and a computer. The infrared is used to be a tool controls the computer. The camera webcam is used to be an image sensor. The camera webcam is detect a position and the movement for image processing. As a result in what we are touch on screened through a video projector.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

รายงานฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ทั้งนี้เพราะได้รับคำแนะนำและคำปรึกษาจาก รศ.ดร.สุรพันธุ์ เอื้อไพบูลย์ ซึ่งเป็นอาจารย์ที่ปรึกษา ในการทำรายงาน และโครงงานผู้จัดทำรู้สึกซาบซึ้งในความอนุเคราะห์จากท่านและขอกราบขอบพระคุณเป็นอย่างสูง

นอกจากนี้ขอขอบพระคุณพี่ๆ ที่ให้ความช่วยเหลือ ตลอดจนให้คำแนะนำต่างๆ จนทำให้รายงานฉบับนี้สำเร็จโดยสมบูรณ์ได้

จึงขอขอบพระคุณไว้ใน ณ ที่นี้

ศิริวรรณ เบ็ญจวรรณ์

อาทิตย์ ไชยพอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อ I

ABSTRACT II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูปภาพ VII

บทที่ 1 บทนำ 1

1.1 หลักการและเหตุผล 1

1.2 ขอบเขตของโครงงาน 1

1.3 วัตถุประสงค์ของโครงงาน 1

1.4 เป้าหมายของโครงงาน 1

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 2

2.1 ความรู้เกี่ยวกับข้อมูลภาพดิจิตอล 2

2.1.1 กระบวนการทำภาพดิจิตอล 2

2.1.2 พิกเซล (Pixel) 3

2.1.3 เกรย์สเกล 4

2.1.4 พื้นฐานและระบบสีโมเดล RGB 4

2.2 ทฤษฎีพื้นฐานของการประมวลผลภาพ 5

2.2.1 Histogram 5

2.2.2 Threshold 5

2.2.3 เทคนิคการตามรอยขอบภาพ (Contour following) 5

2.2.4 เทคนิคการแยกโดยใช้วิธีการ Contour with matrix 7

บทที่ 3 การสร้างจอสัมผัสอินฟราเรด 10

3.1 การจัดการตำแหน่งของกล้องเว็บแคม 10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2 รูปแบบการใช้งาน 11

3.2.1 การเลื่อนตำแหนงของอินฟราเรด 11

3.2.2 การแสดงผล 12

3.3 การออกแบบระบบ 12

3.3.1 Image Capture 13

3.3.2 Infrared Detect 13

3.3.3 Display 23

บทที่ 4 ผลการทดลอง 28

บทที่ 5 บทวิจารณ์และสรุปผล 31

บรรณานุกรม 32

ภาคผนวก 33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตาราง **หน้า**

3.1 โพรซีเยอร์ของ CONNECT 13
3.2 โพรซีเยอร์ของ FIND_OBJECT 16
3.3 โพรซีเยอร์ ของ FIND_TWO_OBJECT 21
3.4 โพรซีเยอร์ ของ CHECK ส่วนแสดงผล 24
3.5 โพรซีเยอร์ ของ CHECK ส่วนแสดงผลกรอบสี่เหลี่ยม 26



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ

รูปภาพ **หน้า**

2.1 กระบวนการทำภาพดิจิตอล 2

2.2 ลักษณะของตำแหน่งพิกเซล 3

2.3 โมเดล RGB 4

2.4 HISTOGRAM หลังจากปรับค่า THRESHOLD 5

2.5 การหาขอบด้วยวิธี CONTOUR FOLLOWING 5

2.6 FLOW CHART การหาขอบด้วยวิธี CONTOUR FOLLOWING 6

2.7 เมทริกซ์ขนาด 3*3 7

2.8 การหาขอบโดยวิธี CONTOUR WITH MATRIX 8

2.9 FLOW CHART การหาขอบโดยวิธี CONTOUR WITH MATRIX 9

3.1 ตำแหน่งการจัดวางกล้องเว็บแคม 11

3.2 แสดงการเลื่อนอินฟราเรดหาตำแหน่ง X,Y 11

3.3 FLOW CHART ภาพรวมของระบบ 12

3.4 FLOW CHART แสดงการ INFRARED DETECT 14

3.5 ภาพก่อนที่จะมีการ DETECT แสงอินฟราเรด 17

3.6 ภาพหลังจากการหาพิกัดของแสดงอินฟราเรด 17

3.7 FLOW CHART แสดงการ TWO POINTER INFRARED DETECT 18

3.8 แสดงการสแกนหาวัตถุสีขาวตามแนวแกน X 22

3.9 แสดงการสแกนหาวัตถุสีขาวตามแนวแกน Y 22

3.10 FLOW CHART ส่วนของ DISPLAY 23

3.11 ภาพแสดงปุ่มรูปภาพส่วนของรถทั้ง 8 ปุ่ม 25

3.12 แสดงการกดปุ่มรูปด้านข้างรถ 25

3.13 แสดงการกดปุ่มที่เป็นรูปด้านหน้ารถ 25

3.14 แสดงการสร้างกรอบสี่เหลี่ยม 26

3.15 แสดงการย่อกรอบสี่เหลี่ยม 27

3.16 แสดงการขยายกรอบสี่เหลี่ยมออกด้านข้าง 27

3.17 แสดงการขยายกรอบสี่เหลี่ยมขึ้นด้านบน 27

3.18 แสดงการขยายกรอบสี่เหลี่ยมให้มีขนาดใหญ่ขึ้น 27

4.1 แสดงภาพที่ยังไม่มีการกดปุ่ม 28

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูปภาพ (ต่อ)

รูปภาพ **หน้า**

4.2 แสดงภาพการกดภาพด้านข้างของรถ 28
4.3 แสดงภาพการกดภาพด้านหน้ารถ 28
4.4 แสดงภาพการกดภาพด้านหลังรถ 28
4.5 การกดภาพด้านในด้านคนขับ 29
4.6 แสดงภาพการกดภาพด้านใน 29
4.7 การกดภาพภายในหลังรถ 29
4.8 แสดงภาพการกดภาพล้อรถ 29
4.9 การกดภาพตัวเครื่องยนต์รถ 29
4.10 แสดงการย่อกรอบสี่เหลี่ยม 30
4.11 แสดงการขยายกรอบสี่เหลี่ยม 30
4.12 แสดงการขยายกรอบสี่เหลี่ยมขึ้นด้านบน 30
4.13 แสดงการขยายกรอบสี่เหลี่ยม 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่1
บทนำ

1.1 หลักการและเหตุผล

ในปัจจุบันเว็บแคมกำลังเป็นที่นิยมในการใช้สนทนากันผ่านอินเทอร์เน็ต จึงมีการนำจอสัมผัสอินฟราเรดมาใช้เพื่อช่วยอำนวยความสะดวกในการใช้งานมากขึ้น โดยการนำการเขียนโปรแกรมให้เชื่อมต่อกับกล้องเว็บแคม ดังนั้นเราจึงมีการประยุกต์ใช้โดยการนำกล้องเว็บแคมที่มีอยู่กับจอคอมพิวเตอร์ธรรมดาให้เป็นจอสัมผัสอินฟราเรด เพื่อให้เกิดประโยชน์สูงสุด สามารถนำไปใช้งานจริงได้ และยังเป็นการใช้งานเว็บแคมได้อย่างคุ้มค่าด้วย

1.2 ขอบเขตของโครงงาน

1. สามารถจำลองการใช้แสงสีขาว(อินฟราเรด)แทนการคลิกเมาส์ได้
2. สามารถหาตำแหน่งพิกัด(x,y) บนหน้าจอได้
3. สามารถใช้งานบนหน้าจอที่ไม่มีการเคลื่อนไหวได้

1.3 วัตถุประสงค์ของโครงงาน

1. เพื่อพัฒนาหน้าจอธรรมดาให้เป็นเสมือนหน้าจอสัมผัส
2. ศึกษาการเขียนโปรแกรมติดต่อกับเว็บแคม
3. เพื่อจำลองการใช้แสงอินฟราเรดแทนเมาส์ได้
4. สามารถนำคอมพิวเตอร์มาประยุกต์ใช้กับเว็บแคมให้เกิดประโยชน์สูงสุด

1.4 เป้าหมายของโครงงาน

เป้าหมายของโครงงาน เป้าหมายเพื่อที่จะประยุกต์ใช้ในการทำหน้าจอธรรมดาให้เป็นเสมือนหน้าจอสัมผัสอินฟราเรด โดยผู้ใช้สามารถจะใช้ตัวอินฟราเรดที่เป็นอุปกรณ์เสริมติดกับนิ้วมือไปสัมผัสยังตำแหน่งบนหน้าจอที่ต้องการจะเลื่อนเม้า ซึ่งจะใช้เว็บแคมหนึ่งตัว เพื่อจับการเคลื่อนไหวและตำแหน่งของแสงสีขาวของอินฟราเรดที่อยู่ปลายอุปกรณ์ที่มาสัมผัสลงบนหน้าจอ เพื่อใช้เป็นการแทนการทำงานของเมาส์ โดยจะจำลองสัญญาณเมาส์ส่งไปให้กับระบบปฏิบัติการแทนสัญญาณที่เกิดขึ้นจากอุปกรณ์จริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

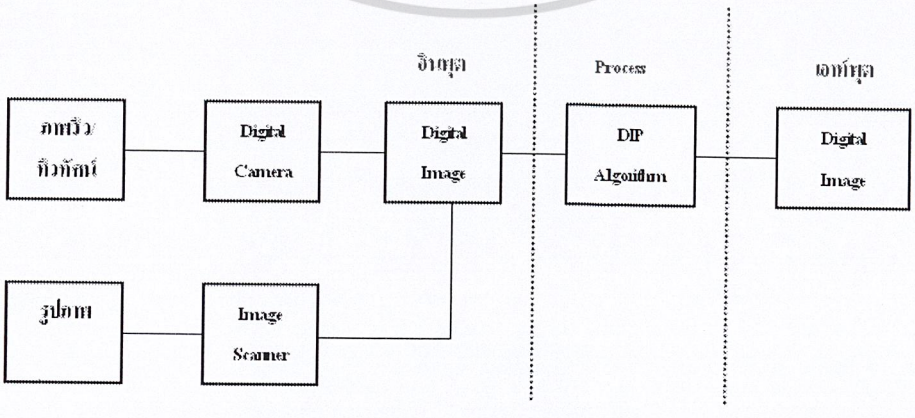
2.1 ความรู้เกี่ยวกับข้อมูลภาพดิจิตอล

ภาพดิจิตอล คือ ภาพ f (x,y) ซึ่งถูกทำให้เป็นค่าดีสครีตทั้งค่าพิกัดและค่าความสว่าง การแทนข้อมูลความสว่างด้วยเลขขนาด 8 บิท ค่าระดับเทาที่ "0" แทนความสว่างน้อยหรือสีดำ ค่าระดับสีเทาที่ "255" แทนความสว่างมากหรือสีขาว ภาพดิจิตอล M x N อาจถูกพิจารณาว่าเป็นเมทริกซ์ ที่มีขนาด M x N ซึ่งค่าอินเด็กซ์ของเมทริกซ์(แถวและคอลัมน์) ระบุจุดในภาพและค่าของเมทริกซ์ที่ตำแหน่งดังกล่าวแทนค่าระดับเทาที่จุดนั้น เราเรียกหน่วยที่เล็กที่สุดของภาพดิจิตอลว่า "พิกเซ็ล"

โดยปกติแล้วสายตาของบุคคลทั่วไปจะมองเห็นภาพทิวทัศน์ วิวต่างๆ เป็นลักษณะอนาล็อกซึ่งสามารถอธิบายได้ทางคณิตศาสตร์ ที่มีตัวแปรนับได้อย่างต่อเนื่อง แต่เครื่องคอมพิวเตอระใช้เลขฐานสองเป็นหลักในการคำนวณ เมื่อนำภาพนั้นมาแปลงเข้าสู่เครื่องคอมพิวเตอร์ ก็จะกลายเป็น ภาพดิจิตอล(Digital Image)

2.1.1 กระบวนการทำภาพดิจิตอล

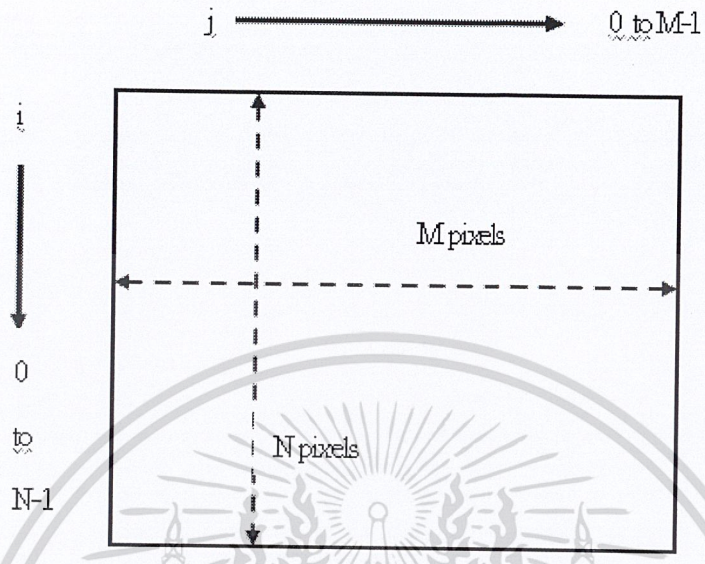
กระบวนการทำภาพดิจิตอล คือ การนำภาพดิจิตอลเข้ามาทำกระบวนการ โดยการใส่ฟังก์ชัน อัลกอริทึมต่างๆเข้าไป ก็จะได้เอาท์พุตเป็นภาพดิจิตอล ที่ตรงตามแนวความคิดของการทำกระบวนการภาพดิจิตอล เหตุผลที่ต้องการทำกระบวนการภาพดิจิตอล คือ เพื่อปรับปรุงภาพดิจิตอลให้มองเห็นได้ง่ายขึ้น



รูปที่ 2.1 กระบวนการทำภาพดิจิตอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 พิกเซล(Pixel)



รูปที่ 2.2 ลักษณะของตำแหน่งพิกเซล

ในภาพหนึ่งๆ เราสามารถอธิบายได้ด้วยเมทริกซ์ ของจุดพิกเซลขนาด N X M โดยใช้คู่อันดับ p(i , j) แทนค่าของจุดแต่ละจุด โดย i มีค่าตั้งแต่ 0 ถึง N-1 และ j มีค่าตั้งแต่ 0 ถึง M-1 และ p(i , j) นี้จะบ่งชี้ความความเข้มของแสงที่จุดนั้นๆ ของภาพ

ค่าที่กำกับแต่ละพิกเซลจะแสดงถึง ค่าเฉลี่ยของความเข้มแสงในภาพที่จุดพิกเซลนั้นแทนอยู่โดยค่าของพิกเซลนี้จะเขียนด้วย Pij ซึ่งมีค่าตั้งแต่ 0 ถึง

2.1.3 เกรย์สเกล

เกรย์สเกล หมายถึง ความแตกต่างของระดับความเข้มแสง โดยเกรย์สเกลหนึ่งๆ อาจแบ่งเป็น 13,20 และ9 ระดับ โดยระดับที่ว่านี้ก็คือ ระดับสีเท่า ในภาพหนึ่งๆ ถ้าต้องการแบ่งระดับความเข้มแสงหรือระดับสีเทาให้หลายๆ ค่านั้น จำเป็นอย่างยิ่งจะต้องเพิ่มจำนวนบิตที่แสดงค่าพิกเซล ตัวอย่าง เช่น ภาพที่มีระดับสีเทา 4 ระดับต้องแทนด้วยเลขฐานสอง จำนวน 2 บิต ถ้าต้องการภาพที่มีระดับสีเทา 16 ระดับต้องแทนด้วยเลขฐานสอง จำนวน 4 บิต และถ้าต้องการภาพที่มีระดับสีเทา 256ระดับต้องแทนด้วยเลขฐานสอง จำนวน 8 บิต เป็นต้น

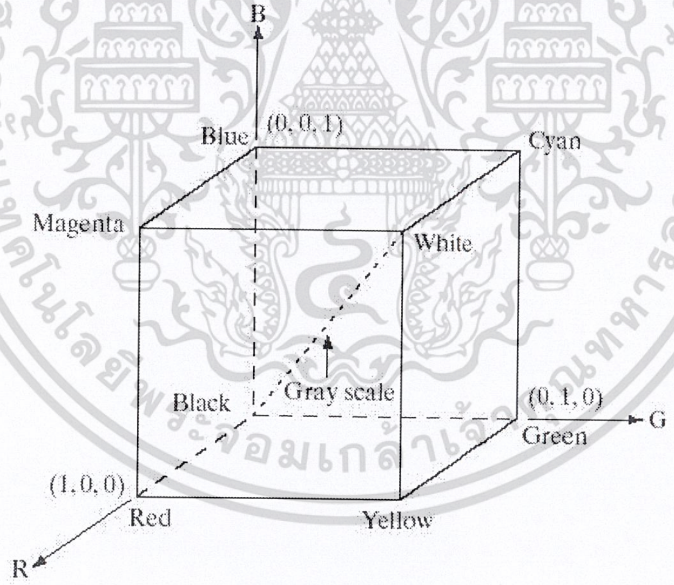
จำนวนระดับสีเทาที่ต้องการนี้ก็คือ ค่าเลขยกกำลัง2 นั้นเองซึ่งค่าต่ำสุดหรือ 0 จะแทนด้วยสีดำ หรือไม่มีความสว่างเลย และค่าที่มากก็คือค่าทีน้อยกว่าระดับสีเทาอยู่ 1 เช่น ค่า 15 ในระบบที่มีระดับสีเทา 16 จะมีสีขาวหรือสว่างมากที่สุด เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 พื้นฐานและระบบสีโมเดล RGB

ในโมเดลนี้ สีแต่ลีอยู่ในรูปของสีปฐมภูมิ(แดง,เขียว,น้ำเงิน)โมเดลนี้มีโครงสร้างอยู่ในระบบของพิกัดคาร์ทีเซียน(Cartesian coordinate)มีลักษณะเป็นลูกบาศก์ดังรูป2.3 ถ้าสีแดง,เขียว, น้ำเงิน จะอยู่ที่มุมทั้งสามบนเส้นทแยงมุมตรงข้ามกันและถ้าสีคราม,มาเจนตา,เหลือง จะอยู่ที่มุมทั้งสามในลักษณะเดียวกัน ส่วนสีดำอยู่ที่จุดกำเนิด สีขาวอยู่ที่มุมที่มีระยะทางไกลที่สุดจากแหล่งกำเนิด ค่าของระดับสีเทา(Gray scale)จะอยู่บนเส้นที่อยู่ระหว่างสีดำกับขาว ค่าสีคือจุดที่อยู่บนผิวหรือในลูกบาศก์ถูกกำหนดค่าโดยเวกเตอร์ที่ชี้ออกจากจุดกำเนิดเพื่อความสะดวกเราจะสมมุติให้ค่าสีถูกนอมอลไลซ์(Normalize)ทั้งสามสี โดยให้มีค่าอยู่ในช่วงศูนย์ถึงหนึ่งลูกบาศก์

ภาพในโมเดล RGB ประกอบด้วยภาพสามระนาบที่เป็นอิสระจากกันสำหรับแต่ละสีปฐมภูมิเมื่อป้อนเข้าไปในมอนิเตอร์ที่เป็นแบบ RGB ภาพทั้งสามสีจะรวมตัวกันที่จอภาพกลายเป็นภาพสีผสมดังนั้นการใช้โมเดลRGB ในการประมวลผลภาพนั้นจะสมเหตุสมผลเมื่อภาพถูกแยกออกโดยธรรมชาติให้อยู่ในเทอมของทั้งสามสี กล้องภาพสีส่วนใหญ่ที่ให้ภาพสีดิจิตอล จะอยู่ในรูปแบบโมเดลRGB ดังนั้นโมเดลนี้จึงเป็นโมเดลที่สำคัญมากในการประมวลผลภาพ



รูปที่ 2.3 โมเดล RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2 ทฤษฎีพื้นฐานของการประมวลผลภาพ

2.2.1 Histogram

Histogram คือ กราฟที่แสดงความสัมพันธ์ระหว่างจำนวน Pixel ของแต่ละระดับ Gray-level ของภาพ เช่น ภาพ 256 สี จะมีค่า Gray-level = 0 ถึง Gray-level =256

2.2.2 Threshold

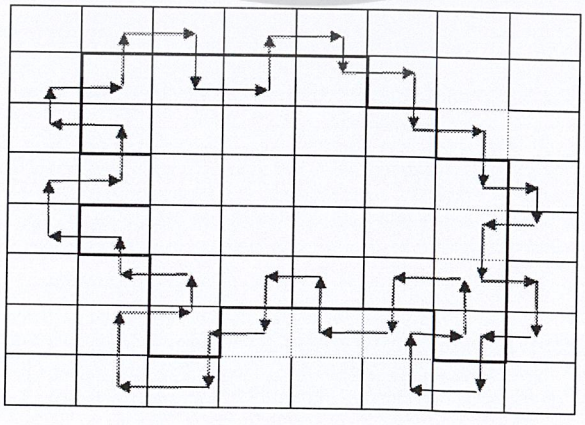
การปรับค่า Threshold นี้คือ ลักษณะการแยก(Split) ระดับ Gray-level ออกเป็น 2 ค่า คือค่าสูงสุด(Gray-level =256) และค่าต่ำสุด(Gray-level =0) ตามที่กำหนด โดยการเปรียบเทียบค่า Threshold ที่ตั้งไว้ว่ามีค่ามากกว่าหรือน้อยกว่า คือ ถ้าจุดภาพนั้นมีระดับ Gray-level มากกว่าค่า Threshold ก็เปลี่ยนให้จุดภาพนั้นมีระดับ Gray-level =256 แล้วถ้าจุดภาพนั้นมีระดับ Gray-level น้อยกว่า Threshold ก็เปลี่ยนให้จุดภาพนั้นมีระดับ Gray-level = 0



รูปที่ 2.4 Histogram หลังจากปรับค่าThreshold

2.2.3 เทคนิคการตามรอยขอบภาพ(Contour Following)

เทคนิคการตามรอยขอบภาพจะช่วยให้เราทำการการหาวัตถุได้ง่ายขึ้น โดยไม่ต้องคำนึงถึงรูปร่างของวัตถุ เมื่อพิจารณาขอบของวัตถุจะมีค่าสีที่มีความแตกต่างกันกับสีพื้นหลัง เช่น ภาพตัวอักขระสีดำที่วางอยู่บนพื้นสีขาว

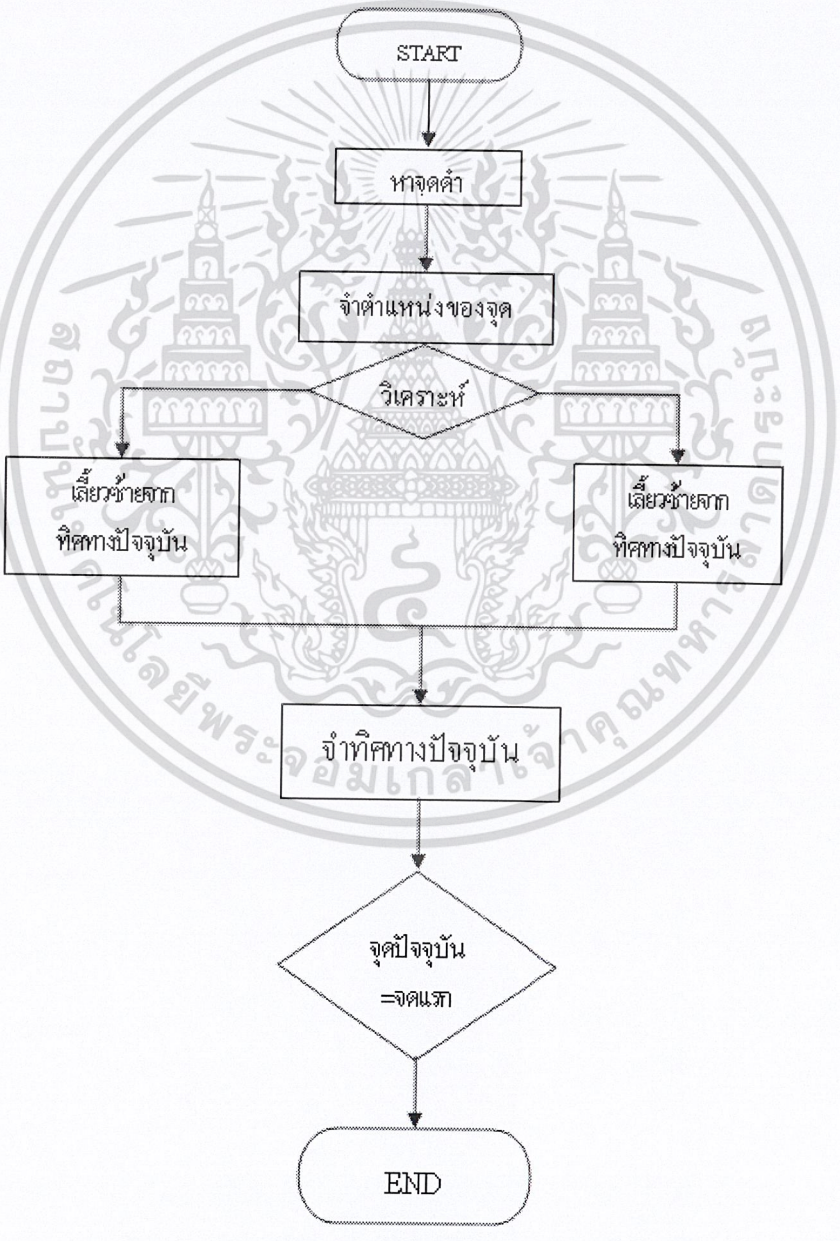


รูปที่ 2.5 การหาขอบด้วยวิธี Contour Following

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การทำงานของเทคนิคการตามรอยขอบภาพตามรูปที่ 2.5 เป็นการเดินไต่ไปตามขอบระหว่างส่วนที่เป็นรูปภาพกับส่วนที่เป็นพื้นหลัง โดยจะตรวจกวาดไปทุกๆจุดภาพ โดยจะเริ่มจากจุดมุมซ้ายบนของข้อมูลภาพ ตรวจกวาดไปในทิศทางจากซ้ายไปขวา และเลื่อนจากบนลงล่างโดยมีเงื่อนไขดังนี้

1. ถ้าจุดที่อยู่ปัจจุบันมีข้อมูลเป็นสีดำ ให้เลี้ยวซ้ายจากทิศทางปัจจุบัน
2. ถ้าจุดที่อยู่ปัจจุบันมีข้อมูลเป็นสีขาว ให้เลี้ยวขวาจากทิศทางปัจจุบัน
3. การเคลื่อนที่จะสิ้นสุด เมื่อจุดที่อยู่ปัจจุบันเป็นจุดเดียวกันกับจุดเริ่มต้น



รูปที่2.6 Flow Chart การหาขอบด้วยวิธี Contour Following

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อธิบาย Flow chart วิธีการ Contour Following จากรูปที่ 2.6

1. หาจุดดำ เป็นการหาจุดดำจุดแรมที่พบ โดยการสแกนตามแนวนอน

2. จำตำแหน่งของจุด เป็นการจำตำแหน่งของจุดดำที่พบ

3. วิเคราะห์จุด มี 2 กรณี

- ขาว เลี้ยวขวาจากทิศทางปัจจุบัน
- ดำ เลี้ยวซ้ายจากทิศทางปัจจุบัน

4. จำทิศทางปัจจุบัน จดจำทิศทางและตำแหน่งปัจจุบัน

5. จุดปัจจุบัน = จุดแรก เป็นการเปรียบเทียบตำแหน่งปัจจุบันกับจุดเริ่มต้น

- ใช้ จบการทำ Contour Following
- ไม่ใช้

ข้อดีของการทำวิธี Contour Following ก็คือ ไม่ต้องสนใจว่ารูปร่างของวัตถุจะมีลักษณะเป็นอย่างไรเมื่อเราวิ่งไปตามขอบของภาพเราก็จะได้ขอบของภาพออกมา

ข้อเสียของการทำวิธีการนี้ก็คือ ถ้าวัตถุที่มีส่วนติดกันอยู่จะไม่สามารถใช้วิธีในการแยกวัตถุออกจากกันได้

2.2.4 เทคนิกการแยกโดยใช้วิธีการ contour with matrix

เทคนิคนี้คล้ายกับการทำ Contour Following แต่จะต่างกันตรงที่เทคนิคนี้จะมีการดึงเอาข้อมูลภาพที่อยู่ในขอบเขตของการทำcontourไปใช้ด้วย ทำให้เราได้วัตถุทั้งอันไปใช้งานและการใช้งานไม่เพียง

แต่ดูว่าข้อมูลตรงจุดนั้นเป็นสีดำ หรือสีขาว แต่จะใช้ เมทริกซ์ช่วยตรวจสอบแทน เมทริกซ์ที่ใช้มีขนาด3*3 ดังรูปที่2.8

A1	A2	A3
A4	A5	A6
A7	A8	A9

*รูปที่2.7 เมทริกซ์ ขนาด 3*3*

แนวคิดของการทำ contour with matrix ดังรูปที่ 2.8 มีขั้นตอนการทำดังต่อไปนี้

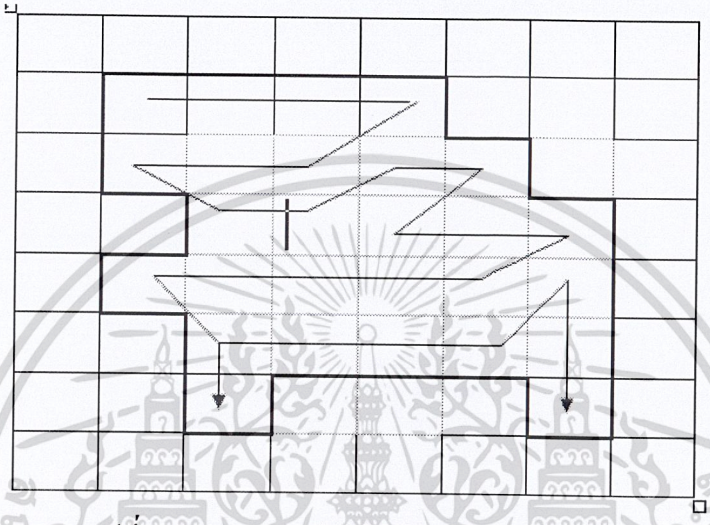
1. ให้กำหนดจุดดำที่พบเป็นจุดกลางของเมทริกซ์ (a5) คัดลอกจุดนี้ลงไปในบัฟเฟอร์แล้วลบจุดกลางนั้นออกจากรูปภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. หาจุดกลางถัดไป โดยตรวจสอบจากข้อมูลรอบๆจุดกลางปัจจุบัน(a5) ซึ่งเป็นจุดสีดำ คือ การตรวจสอบจะเริ่มจากด้านบนซ้ายก่อน คือ ตำแหน่ง a1,a2,a3,a4,a5,a6,a7,a8และa9 ตามลำดับ โดยกำหนดจุดดำที่พบจุดแรกให้เป็นจุดกึ่งกลางเมทริกซ์

3. เมื่อรอบๆจุดกลางนั้นไม่มีจุดดำแล้ว จะทำการย้อนกลับไปยังจุดกลางตัวก่อน

4. การทำงานจะเป็นเช่นนี้ไปเรื่อยๆจนกระทั่งจะไม่มีจุดดำรอบจุดกลางทุกตัว



รูปที่ 2.8 การหาขอบโดยวิธี Contour with matrix

ตัวอย่างการทำงานด้วยวิธีนี้แสดงดังรูปที่ 2.8 เส้นทางการวิ่งจะเป็นไปตามลูกศรที่แสดงในรูปและสามารถใช้ FlOW CHART จากรูปที่ 2.9 อธิบายได้ดังนี้

1. หาจุดดำ เป็นการหาจุดดำจุดแรกที่พบโดยการสแกนตามแนวนอน

2. จำตำแหน่งของจุด จำตำแหน่งของจุดดำที่พบ

3. copy pixels จะทำการคัดลอกจุดที่พบไว้ในบัฟเฟอร์ และทำการลบจุดที่พบออกจากรูปภาพ

4. จุดดำรอบจุดปัจจุบันมีจุดสีดำล้อมรอบหรือไม่

- มีจำตำแหน่งของจุด
- ไม่มี จุดที่ไม่ได้วิเคราะห์

5. จุดที่ไม่ได้วิเคราะห์ ตรวจว่ามีจุดที่ยังไม่ได้ทำการวิเคราะห์หรือไม่

- มี copy pixels
- ไม่มี contour with matrix

ข้อดีของวิธี contour with matrix ก็คือ สามารถดึงเอาส่วนที่เป็นเนื้อวัตถุออกมาได้ทั้งหมดเป็นประโยชน์เมื่อต้องการลบวัตถุนั้นๆออกจากภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อเสียของวิธีนี้ก็คือ ต้องเอาข้อมูลทุกพิกเซลของวัตถุออกมา ทำให้ต้องใช้เวลานานกับการคัดลอก และลบข้อมูล วิธีการนี้ทำงานแบบ recursive ถ้ามีการ backtracking มากเกินไปอาจทำให้ stack overflow ได้



รูปที่ 2.9 Flow Chart การหาขอบโดยวิธี Contour with matrix

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การสร้างจอสัมผัสอินฟราเรด

การสร้างจอสัมผัสอินฟราเรด จะใช้เวบแคม 1 ตัว ในการจับการเคลื่อนที่ของอินฟราเรด โดยจะวางไว้ตรงข้ามกับจอโปรเจคเตอร์ ซึ่งจะวางห่างประมาณ 2 เมตร อุปกรณ์ที่ใช้ในการสร้างจอสัมผัสอินฟราเรด มีดังนี้

1. อินฟราเรด
2. กล้องเวบแคม
3. แผ่นกรองอินฟราเรด หรือ ฟิล์มถ่ายรูปที่ไม่ใช้แล้ว

ในการสร้างจอสัมผัสอินฟราเรดนี้ จะใช้โปรแกรม Visaul Basic ในการติดต่อกับเวบแคม และตรวจจับอินฟราเรด ซึ่งโปรแกรม Visaul Basic นี้จะมีฟังก์ชันที่ช่วยในการประมวลผลภาพเพื่อให้ง่ายต่อการใช้งานด้วย

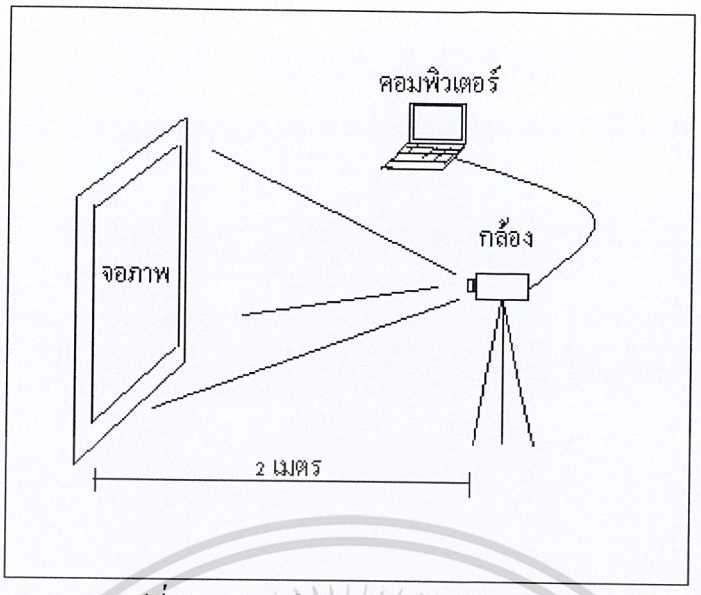
คุณสมบัติของกล้อง

- Resolution 2.0 M Pixels Up To 5.0M
- Frame rate 30 Fps
- USB 1.1/2.0 Interface
- Info Noise rate 48dB
- High qualith glass lens
- Focus range 30mm-infinitive
- Compatibility WIN2000 -- ME -- XP – VISTA

3.1 การจัดวางตำแหน่งของกล้องเวบแคม

การจัดวางตำแหน่งของกล้องเวบแคมควรจะให้อยู่ตรงกลางของจอภาพ เพื่อให้การตรวจการเคลื่อนไหวของอินฟราเรดมีความแม่นยำมากยิ่งขึ้น และให้ง่ายต่อการสั่งงาน ดังรูปที่ 3.1 ซึ่งในการจัดวางกล้องเวบแคม ไม่ควรมีสิ่งใดมากีดขวางหน้ากล้องซึ่งจะทำให้การสั่งงานด้วยอินฟราเรดไม่เกิดผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 ตำแหน่งการจัดวางกล้องนวบแคม

3.2 รูปแบบการใช้งาน

ในการที่จะสั่งงาน การเลื่อนตำแหน่ง การแสดงผล จะมีรูปแบบการทำงานดังนี้

3.2.1 การเลื่อนตำแหน่งของอินฟราเรด

ในการเลื่อนตำแหน่งของอินฟราเรดนั้น เราทำการกดอินฟราเรดค้างไว้ จากนั้นก็เลื่อนอินฟราเรดไปเรื่อยๆ เพื่อหาตำแหน่งของพิกัด X,Y โดยให้บอกพิกัดจุดตรงกลางของวัตถุ



รูปที่ 3.2 แสดงการเลื่อนอินฟราเรดหาตำแหน่ง X,Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2.2 การแสดงผล

ในการแสดงผล เมื่อเราได้พิกัด X,Y แล้ว จากการเลื่อนหาตำแหน่งจาก 3.2.1 จากนั้นนำมากำหนดปุ่มโดยแต่ละปุ่มจะมีพิกัด X,Y คนละพิกัด โดยแต่ละปุ่มกำหนดให้มีการแสดงผลเมื่อปรากฏอินฟราเรดขึ้นที่พิกัดนั้น เมื่อเอาอินฟราเรดไปไว้ที่ตำแหน่งพิกัดนั้นๆ จากนั้นพอกดสวิทซ์จะเหมือนลักษณะการคลิกเมาส์ ซึ่งจะปรากฏรูปภาพขึ้นมา

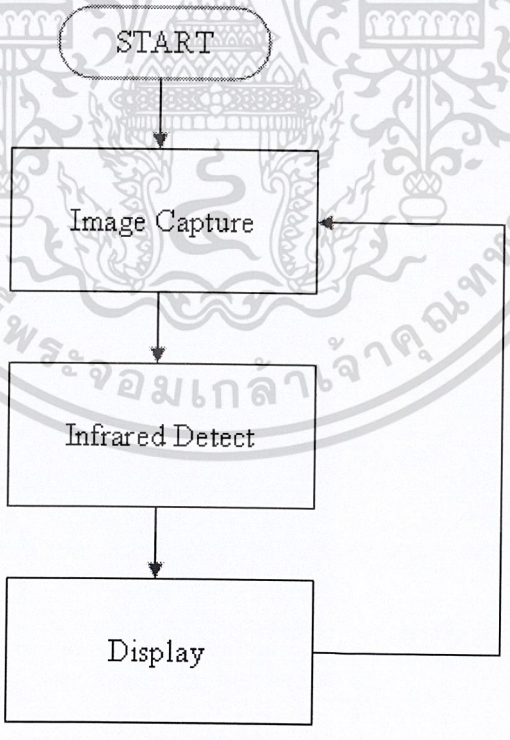
3.3 การออกแบบระบบ

ระบบจะแบ่งออกเป็น 3 ส่วนใหญ่ๆ ดังรูปที่ 3.3 คือ

1. Image Capture เป็นส่วนที่ทำการดึงภาพมาจากเวบแคมและเก็บไว้ในคลิปบอร์ด

2. Infrared Detect เป็นส่วนที่นำภาพจากคลิปบอร์ดมาประมวลผล โดยให้กล้องจับแสงอินฟราเรด หาตำแหน่งพิกัดของแสงอินฟราเรด

3. Display เป็นส่วนที่ทำการแสดงผล โดยนำค่าพิกัดที่ได้จากการหาพิกัดในขั้นตอนที่ 2 มาและกำหนดการแสดงผลขึ้นทางหน้าจอ



รูปที่ 3.3 Flow Chart ภาพรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 Image Capture

เป็นส่วนที่ทำการดึงภาพจากเวบแคมโดยจะใช้การเขียนโปรแกรมติดต่อกล้องโดยใช้โปรแกรม Visaul Basic 6 ในการติดต่อดังนี้

```
Private Sub cmdcon_Click()
mCapHwnd = capCreateCaptureWindow("WebcamCapture", 0, 0, 0, _ 320, 240, Me.hWnd,
0)
DoEvents
SendMessage mCapHwnd, CONNECT, 0, 0
Picture3.Width = 320 * Screen.TwipsPerPixelX
Picture3.Height = 240 * Screen.TwipsPerPixelY
picColor.AutoSize = True
picColor.AutoRedraw = True
shpLED0(0).Visible = False
Timer1.Enabled = True
End Sub
```

ตารางที่ 3.1 โพรซีเยอร์ ของ Connect

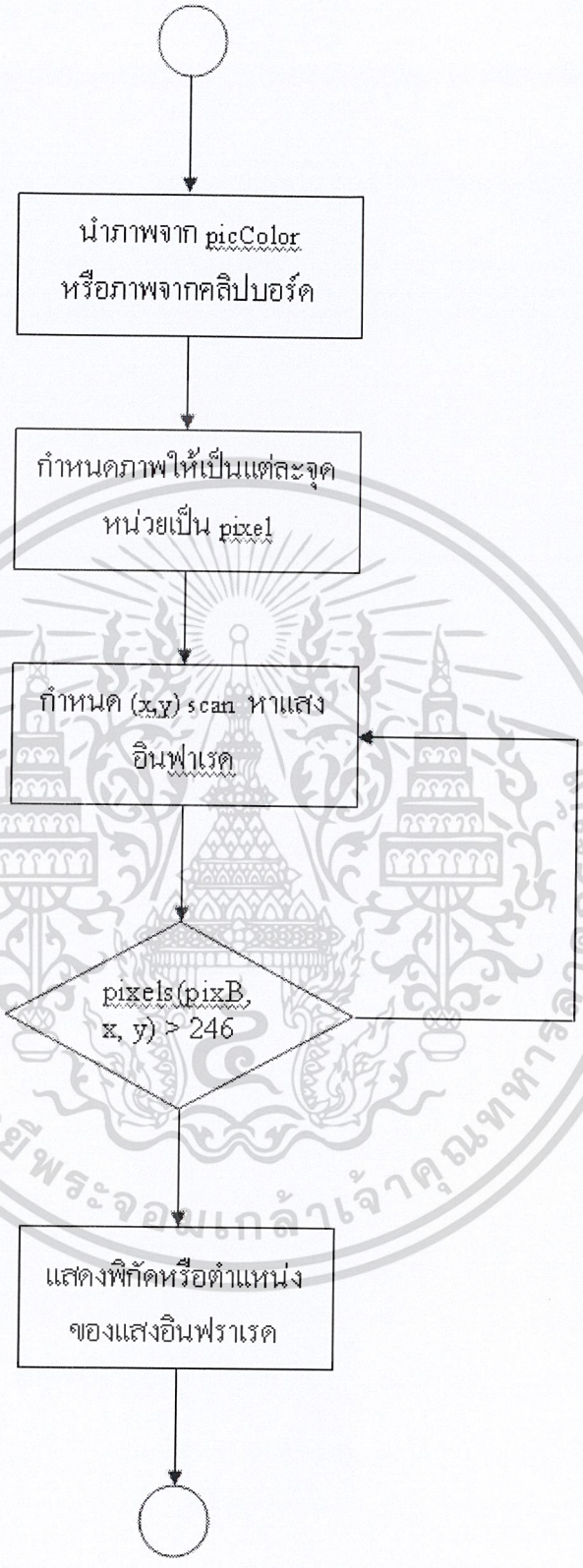
โพรซีเยอร์ cmdCon_Click ไว้กดติดต่อกับกล้องเวบแคม เมื่อเรากดปุ่ม Connect จะได้ mCapHwnd เป็นภาพขนาด 320*240 pixel ซึ่งภาพจะถูกแสดงขึ้นที่ pictureBox ชื่อว่า Picture3 โดยรูปจะถูกแปลงให้เป็น pixel โดยให้ขนาดความกว้าง เป็น pixel X และความสูงของภาพ แสดงเป็น pixel Y ซึ่งภาพจะถูกเก็บไว้ในคลิปบอร์ด และนำภาพจากคลิปบอร์ดไปแสดงที่ picColor ซึ่งภาพก็จะมีขนาดเท่ากันซึ่งเราจะใช้ภาพจาก picColor ไปทำการประมวลผล

3.3.2 Infrared Detect

<u>ตอนที่ 1 Single touch</u>

เป็นส่วนที่นำภาพจากคลิปบอร์ดมาประมวลผล โดยให้กล้องจับแสงอินฟราเรด หาตำแหน่งพิกัดของแสงอินฟราเรด เป็นส่วนของการหาวัตถุที่มีค่า pixel R,G,B > 246 เนื่องจากแสงอินฟราเรดมีค่าของ pixels(R,G,B)=252 โดยมีการหาวัตถุตามรูปที่ 3.4 และตารางที่ 3.2 ดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 Flow Chart แสดงการ Infrared Detect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Find_Object(ByVal picColor As PictureBox)
    minx = picColor.Width
    miny = picColor.Height
    maxx = 0
    maxy = 0
     ' Prepare the bitmap description.
    With bitmap_info.bmiHeader
       .biSize = 40
       .biWidth = picColor.ScaleWidth
       ' Use negative height to scan top-down.
       .biHeight = -picColor.ScaleHeight
       .biPlanes = 1
       .biBitCount = 32
       .biCompression = BI_RGB
       bytes_per_scanLine = ((((.biWidth * .biBitCount) + 31) \ 32) * 4)
       pad_per_scanLine = bytes_per_scanLine - (((.biWidth * _
                          .biBitCount)    + 7) \ 8)
       .biSizeImage = bytes_per_scanLine * Abs(.biHeight)
    End With
    ' Load the bitmap's data.
    ReDim pixels(1 To 4, 1 To picColor.ScaleWidth, 1 To _ picColor.ScaleHeight)
    GetDIBits picColor.hdc, picColor.Image, _
       0, picColor.ScaleHeight, pixels(1, 1, 1), _
       bitmap_info, DIB_RGB_COLORS
    ' Modify the pixels.
    For y = 1 To picColor.ScaleHeight
       For x = 1 To picColor.ScaleWidth
            If (pixels(pixB, x, y) > 246) And (pixels(pixG, x, y) > 246) _
               And (pixels(pixR, x, y) > 246) Then
              pixels(pixR, x, y) = 0
             pixels(pixG, x, y) = 0
```

1

2

3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            pixels(pixB, x, y) = 0
            If minx > x Then minx = x
            If maxx < x Then maxx = x
            If miny > y Then miny = y
            If maxy < y Then maxy = y
          End If
        Next x
      Next y
            N = (minx + maxx) / 2
          O = (miny + maxy) / 2
        shpLED0(0).Visible = True
        shpLED0(0).BackColor = vbBlue
        shpLED0(0).Width = 8
        shpLED0(0).Height = 8
        shpLED0(0).Top = O - 4
        shpLED0(0).Left = N - 4
        Label1.Caption = "Center is (" & (N - 4) & "," & (O - 4) & ")"
        U = N - 4
        V = O - 4
        check
    End Sub
```

3

4

ตารางที่ 3.2 โพรซีเยอร์ ของ Find_Object

โพรซีเยอร์ของ Find_Object เป็นส่วนที่นำภาพจากคลิปบอร์ดนำมาประมวลผลและแสดงผลออกไปที่ picColor ซึ่ง ประกอบด้วย 5 ส่วน ดังนี้

หมายเลข 1 เป็นส่วนกำหนดให้ค่าสูงสุดและค่าต่ำสุดของภาพคือส่วนไหนซึ่งมีหน่วยเป็น pixel

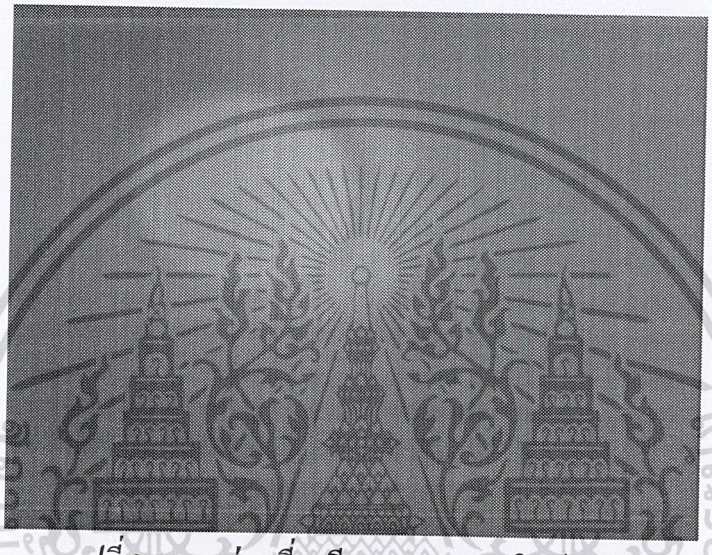
หมายเลข 2 เป็นส่วนการนำข้อมูลของภาพโดยภาพที่ได้จากกล้องเป็นภาพ Bitmap ซึ่งนำภาพมาประมวลผลโดยการเลื่อนวัตถุหาจุดโดยให้สแกนจากบนลงล่างโดยให้ pixel จากบนลงล่างมีค่าเป็นลบตามความสูงของภาพ โดยสแกนเป็นจุดและเป็น Byte เพื่อสแกนหาแสงของอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

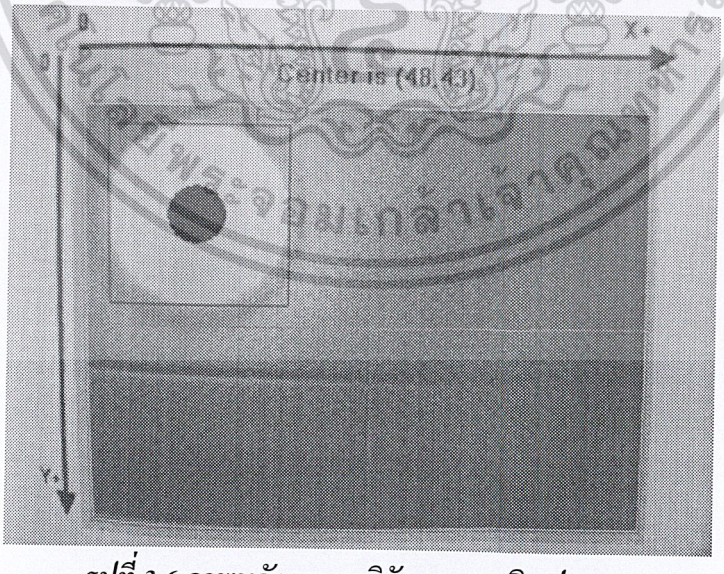
สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

หมายเลข 3 เป็นส่วนของการสแกนหาแสงอินฟราเรดโดยให้ pixel Y เป็นความสูงของภาพ และให้ pixel X เป็นความกว้างของภาพ โดยให้สแกนไปเป็นแนวนอน และค่อยเลื่อน pixel Y ลงมาทีละจุดสแกนหาแสงที่มีค่า pixel (R,G,B)>246

หมายเลข 4 เป็นส่วนที่ให้แสดงพิกัดกึ่งกลางของจุด ซึ่งเราใช้จุดในการแสดงพิกัดของแสงอินฟราเรดเพื่อนำจุดมาเป็นตัวกำหนดและบอกตำแหน่งพิกัดของแสงอินฟราเรด จากนั้นก็นำพิกัดไปแสดงผลโดยสั่งงานให้แสดงผลต่างๆ



รูปที่ 3.5 ภาพก่อนที่จะมีการ detect แสงอินฟราเรด

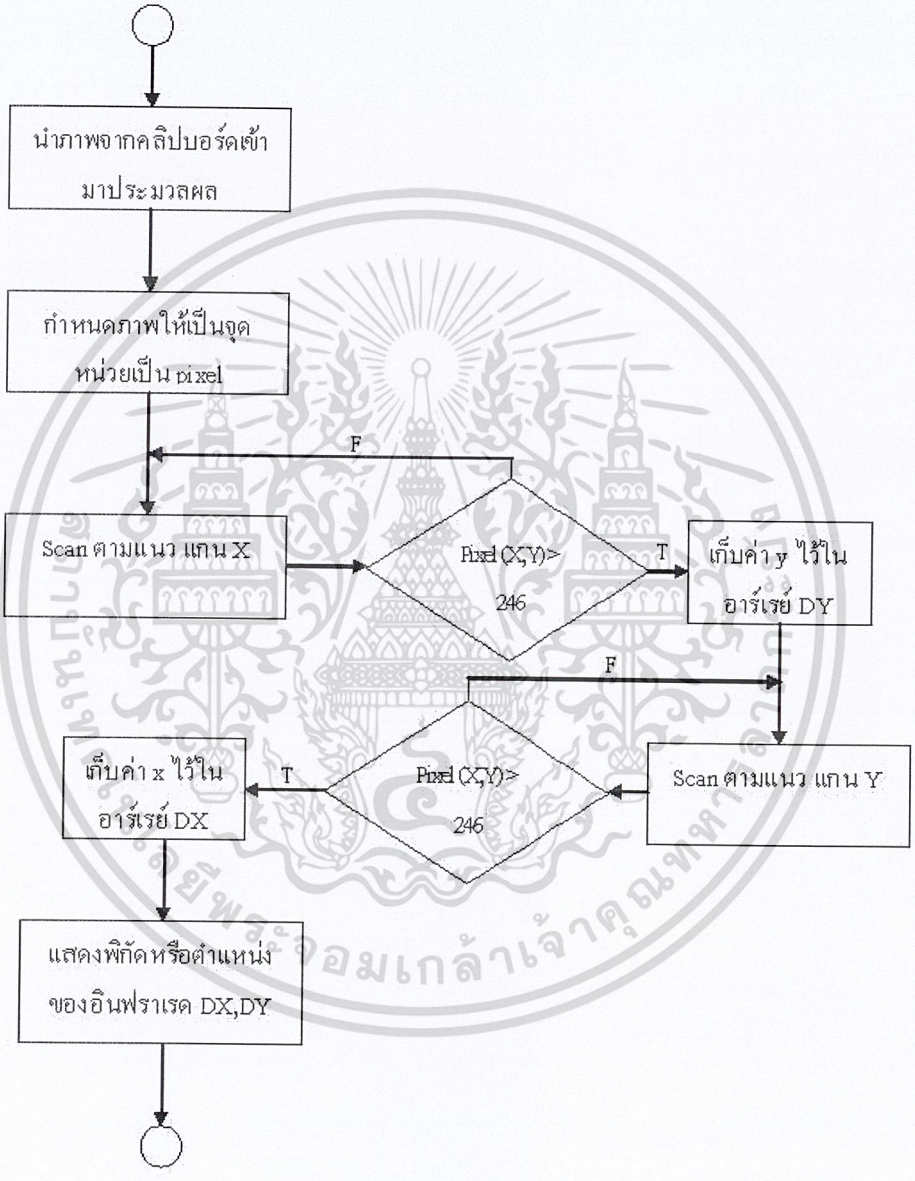


รูปที่ 3.6 ภาพหลังจากหาพิกัดของแสงอินฟราเรด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
119234

ตอนที่ 2 Two pointer detect

เป็นส่วนที่นำภาพจากคลิปบอร์ดมาทำการประมวลผลภาพ โดยกำหนดให้หาวัตถุโดยวัตถุที่มีค่าพิกัด (pixel(x,y)>246) ซึ่งในที่นี้จะให้แสดงพิกัด (pixel(x,y)) 2 จุดเนื่องจากต้องการตรวจจับภาพที่มีมากกว่า 1 อัน โดยมีวิธีการดังนี้



รูปที่ 3.7 Flow Chart แสดงการ Two pointer Infrared Detect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Find_Two_Object(ByVal picColor As PictureBox)
Dim bitmap_info As BITMAPINFO
Dim pixels() As Byte
Dim bytes_per_scanLine As Integer
Dim pad_per_scanLine As Integer
Dim x As Integer
Dim y As Integer
Dim ave_color As Byte
    ' Prepare the bitmap description.
   With bitmap_info.bmiHeader
      .biSize = 40
      .biWidth = picColor.ScaleWidth
      ' Use negative height to scan top-down.
      .biHeight = -picColor.ScaleHeight
      .biPlanes = 1
      .biBitCount = 32
      .biCompression = BI_RGB
      bytes_per_scanLine = ((((.biWidth * .biBitCount) + 31) \ 32) * 4)
      pad_per_scanLine = bytes_per_scanLine - (((.biWidth * .biBitCount) + 7) \ 8)
      .biSizeImage = bytes_per_scanLine * Abs(.biHeight)
   End With
   ' Load the bitmap's data.
   ReDim pixels(1 To 4, 1 To picColor.ScaleWidth, 1 To picColor.ScaleHeight)
   GetDIBits picColor.hdc, picColor.Image, _
      0, picColor.ScaleHeight, pixels(1, 1, 1), _
      bitmap_info, DIB_RGB_COLORS
   ' Modify the pixels.
   Dim i As Integer
Dim j As Integer
Dim object As Boolean
Dim DX(319) As Integer
Dim DY(239) As Integer
```

1

2

1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Qty = -1
Qty2 = -1
'Scan แนวแกน X
object = False
For x = 1 To picColor.ScaleWidth
  For y = 1 To picColor.ScaleHeight
    If (pixels(pixR, x, y) > 252) And (pixels(pixG, x, y) > 252) And _
      (pixels(pixB, x, y) > 252) Then
        DX(1) = x
        DY(1) = y
         object = True
        Qty = Qty + 1
      If Qty = 0 Then
        DX(2) = x
        DY(2) = y
      End If
    End If
  Next y
Next x
'Scan แนวแกน Y
object = False
For y = 1 To picColor.ScaleHeight
  For x = 1 To picColor.ScaleWidth
    If (pixels(pixR, x, y) > 252) And (pixels(pixG, x, y) > 252) And _
      (pixels(pixB, x, y) > 252) Then
        DX(3) = x
        DY(3) = y
         object = True
        Qty = Qty + 1
      If Qty = 0 Then
        DX(4) = x
        DY(4) = y
```

3

4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  End If
      End If
   Next x
  Next y
  Label1.Caption = "center( " & DX(1) & "),( " & DY(1) & "),( " & DX(2) & "), _
( " & DY(2) & ")"
    N = (DX(1) + 45)
    O = (DY(1) + 50)
    U = (DX(2) + 45)
    V = (DY(2) + 50)
check
End Sub
```

5

ตารางที่ 3.3 โพรซีเยอร์ ของ Find_Two_Object

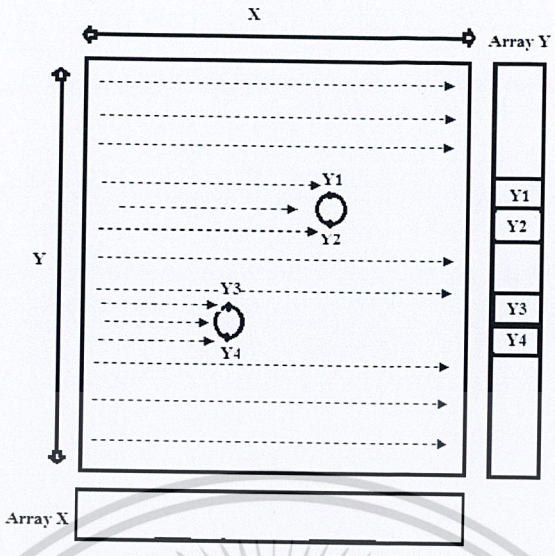
โพรซีเยอร์ของ Find_Two_Object เป็นส่วนที่นำภาพจากคลิปบอร์ดนำมาประมวลผลโดยการหาวัตถุที่มีแสงสีขาวดังนี้

หมายเลข 1 เป็นส่วนกำหนดให้ค่าสูงสุดและค่าต่ำสุดของภาพคือส่วนไหนซึ่งมีหน่วยเป็น pixel

หมายเลข 2 เป็นส่วนการนำข้อมูลของภาพโดยภาพที่ได้จากกล้องเป็นภาพ Bitmap ซึ่งนำภาพมาประมวลผลโดยการเลื่อนวัตถุหาจุดโดยให้สแกนจากบนลงล่างโดยให้ pixel จากบนลงล่างมีค่าเป็นลบตามความสูงของภาพ โดยสแกนเป็นจุดและเป็น Byte เพื่อสแกนหาแสงของอินฟราเรด

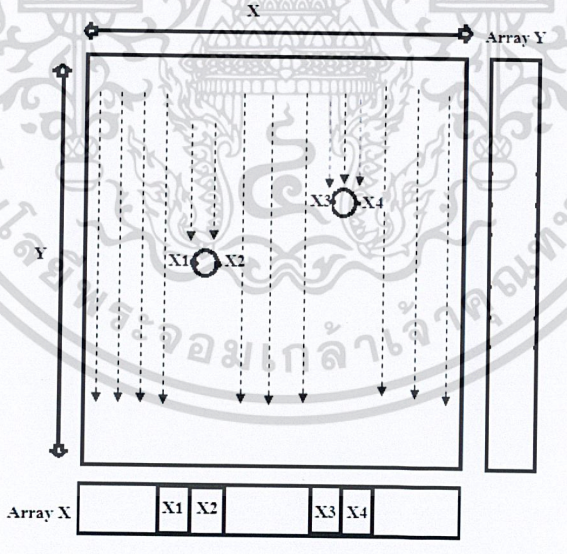
หมายเลข 3 สแกนหาวัตถุที่มีสีขาวตามแนวแกน X โดยจะได้ค่าพิกัด Y ดังรูปที่ 3.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 แสดงการสแกนหาวัตถุสีขาวตามแนวแกน X

หมายเลข 4 สแกนหาวัตถุที่มีสีขาวตามแนวแกน Y โดยจะได้ค่าพิกัด X ดังรูปที่ 3.9



รูปที่ 3.9 แสดงการสแกนหาวัตถุสีขาวตามแนวแกน Y

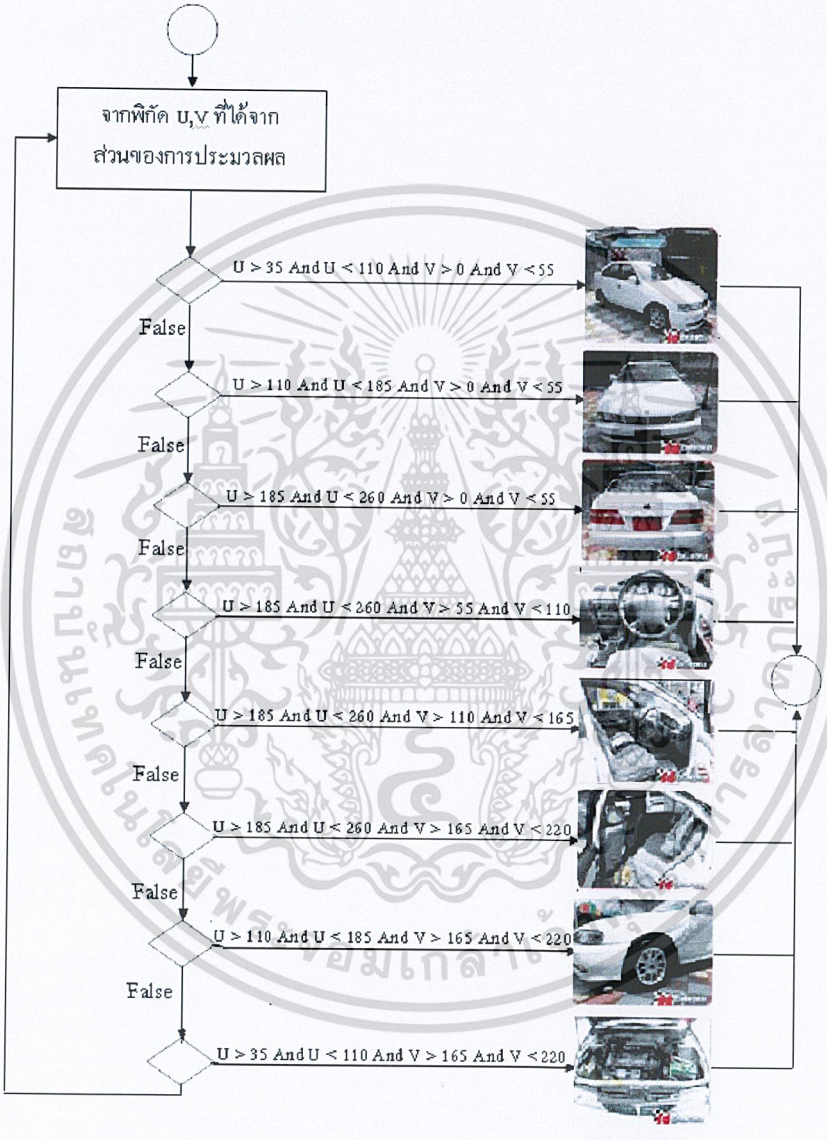
หมายเลข 5 ส่วนของการนำพิกัด DX,DY มาแสดงผลค่าพิกัดและนำพิกัดไปเขียนโปรแกรมควบคุมการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 Display

<u>ตอนที่ 1 Single touch</u>

เป็นส่วนที่ทำการแสดงผล โดยนำค่าพิกัดที่ได้จากการหาพิกัดในขั้นตอนที่ 2 มาและกำหนดการแสดงผลขึ้นทางหน้าจอ มีการทำงานแสดงดังรูปที่ 3.8



รูปที่ 3.10 Flow Chart ส่วนของ Display

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนของโปรแกรมแสดงดังตารางที่ 3.4 ดังนี้

```
Private Sub check()
If U > 35 And U < 110 And V > 0 And V < 55 Then
picZoom.Picture = pic1.Image
End If
If U > 110 And U < 185 And V > 0 And V < 55 Then
picZoom.Picture = pic2.Image
End If
If U > 185 And U < 260 And V > 0 And V < 55 Then
picZoom.Picture = pic3.Image
End If
If U > 185 And U < 260 And V > 55 And V < 110 Then
picZoom.Picture = pic4.Image
End If
If U > 185 And U < 260 And V > 110 And V < 165 Then
picZoom.Picture = pic5.Image
End If
If U > 185 And U < 260 And V > 165 And V < 220 Then
picZoom.Picture = pic6.Image
End If
```

ตารางที่ 3.4 โพรซีเยอร์ ของ check ส่วนแสดงผล

ในส่วนของโพรซีเยอร์ของ check () เนื่องจากเราได้พิกัด U,V ของตำแหน่งของแสงอินฟราเรด เมื่อนำอินฟราเรดไปไว้ที่ตำแหน่งที่กำหนดไว้ 8 ตำแหน่งนั้นจะเกิดการสั่งงานเหมือนการคลิกเมาส์นั่นเอง ซึ่งภาพจะโชว์ขึ้นตามรูปภาพของรถที่กำหนดไว้แต่ละจุดดังรูปที่ 3.11-3.13

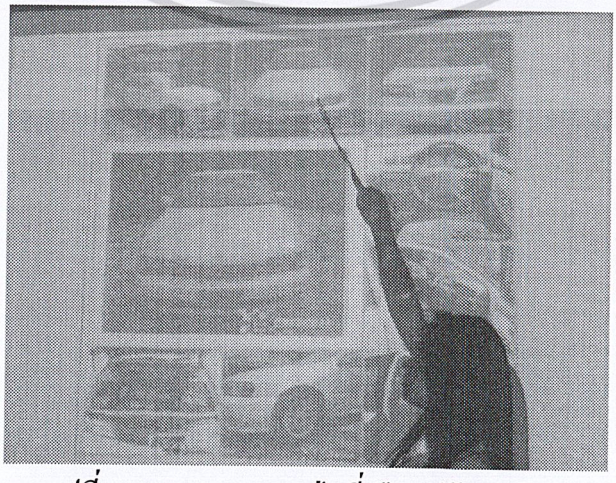
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 ภาพแสดงปุ่มรูปภาพส่วนของรถทั้ง 8 ปุ่ม



รูปที่ 3.12 แสดงการกดปุ่มรูปด้านข้างรถ

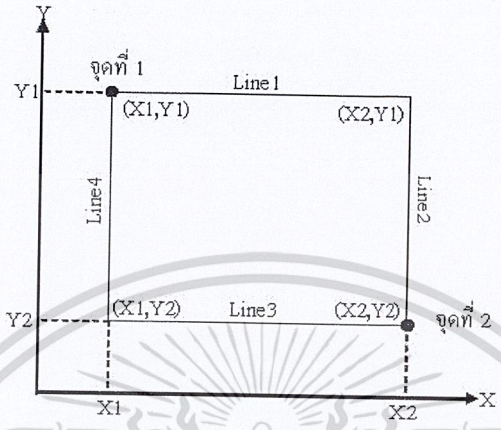


รูปที่ 3.13 แสดงการกดปุ่มที่เป็นรูปด้านหน้ารถ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 2 Two Pointer Detect

เป็นการสร้างรูปสี่เหลี่ยมโดยใช้จุดพิกัดสองจุดที่ทำการหาโดยการสร้างเส้นสี่เส้นและให้เส้นสี่เส้นเชื่อมต่อกันซึ่งมีลักษณะการสร้างดังรูป 3.14



รูปที่ 3.14 แสดงการสร้างกรอบสี่เหลี่ยม

จากรูปที่ 3.14 ทำการกำหนดโดยเมื่อรู้ค่าพิกัดของจุดสองจุดจะทำการสร้างกรอบสี่เหลี่ยมโดยนำจุดต้นลบจุดปลาย

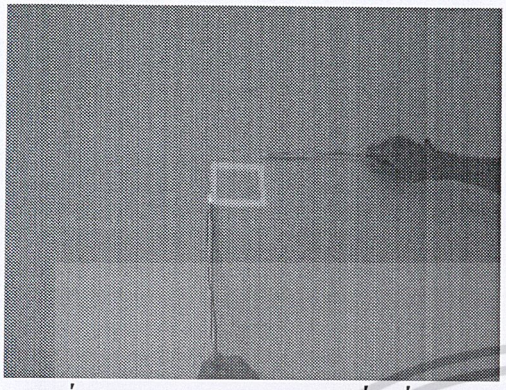
1. กำหนด Line1=(X1,Y1)-(X2,Y1), Line2=(X2,Y1)-(X2,Y2), Line3=(X2,Y2)-(X1,Y2), Line4=(X1,Y2)-(X1,Y1)
2. ทำการเขียนโปรแกรม

```
Private Sub check()
Line1.X1 = (N * 30),Line1.Y1 = (O * 30)
Line1.X2 = (U * 30),Line1.Y2 = (O * 30)
Line2.X1 = (U * 30),Line2.Y1 = (O * 30)
Line2.X2 = (U * 30),Line2.Y2 = (V * 30)
Line3.X1 = (U * 30),Line3.Y1 = (V * 30)
Line3.X2 = (N * 30),Line3.Y2 = (V * 30)
Line4.X1 = (N * 30),Line4.Y1 = (V * 30)
Line4.X2 = (N * 30),Line4.Y2 = (O * 30)
End Sub
```

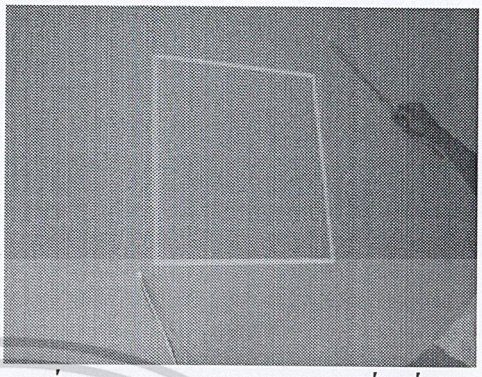
ตารางที่ 3.5 โพรซีเยอร์ ของ check ส่วนแสดงผลกรอบสี่เหลี่ยม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

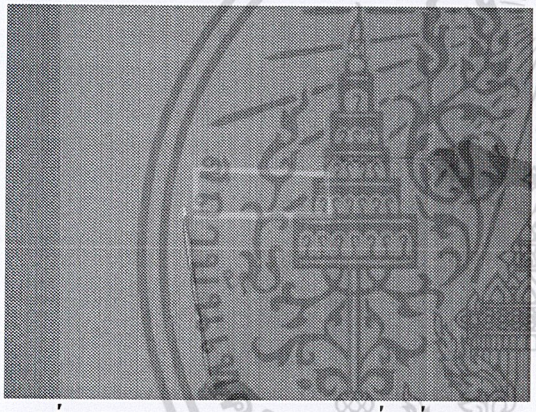
เมื่อทำการเลื่อนจุดอินฟราเรดกรอบสี่เหลี่ยมก็จะมีการเลื่อนตามซึ่งจะย่อ-ขยายขนาดตามจุดสองจุดดังนี้



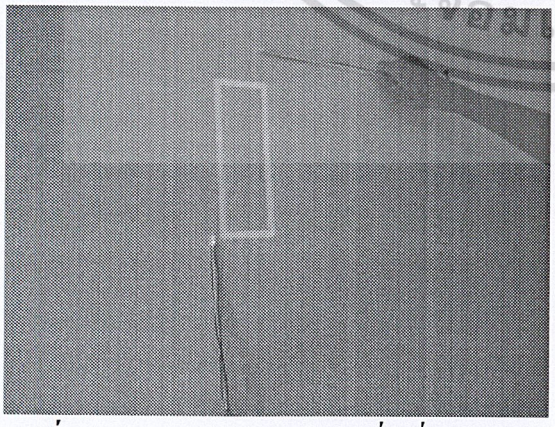
รูปที่ 3.15 แสดงการย่อกรอบสี่เหลี่ยม



รูปที่ 3.18 แสดงการขยายกรอบสี่เหลี่ยม ให้มีขนาดใหญ่ขึ้น



รูปที่ 3.16 แสดงการขยายกรอบสี่เหลี่ยม ออกด้านข้าง



รูปที่ 3.17 แสดงการขยายกรอบสี่เหลี่ยม ขึ้นด้านบน

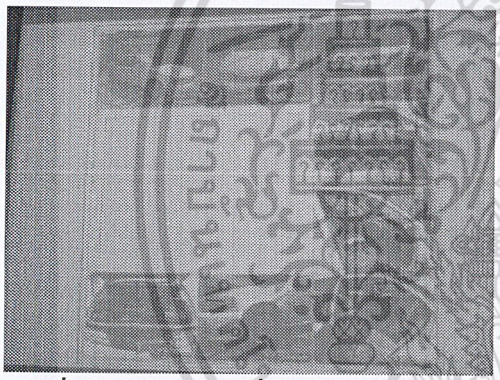
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

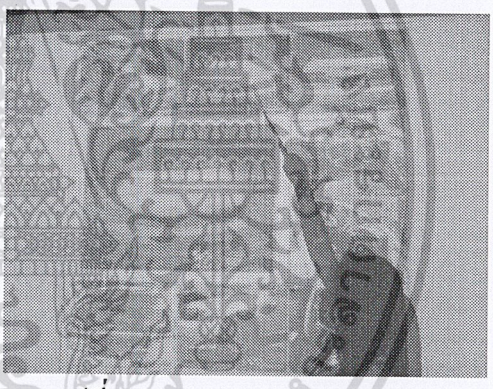
ผลการทดลอง

ตอนที่ 1 Single Touch

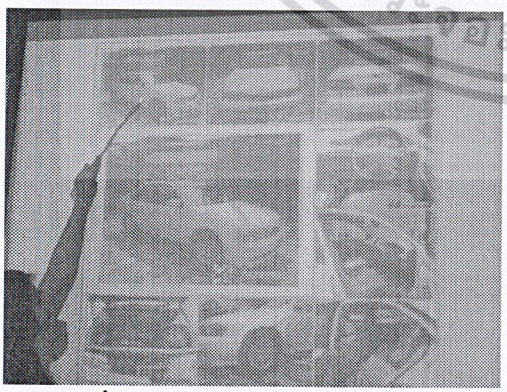
เมื่อเราทำการกำหนดค่าพิกัด(x,y)ของวัตถุที่เราต้องการแล้ว เราทำการกดอินฟราเรดหรือแสงสีขาวที่ตำแหน่งของรูปภาพที่แสดงส่วนต่างๆ ของรถ ที่ทำการกำหนดช่วงของ Pixcelsไว้แล้วบนหน้าจอก็จะทำการแสดงภาพใหญ่ที่บริเวณตรงกลางที่สร้างไว้ แต่ถ้าเราทำการกดอินฟราเรดหรือเลื่อนอินฟราเรดไม่ตรงกับกรอบสี่เหลี่ยมที่กำหนด Pixcels ไว้แล้ว ก็จะไม่มีการเปลี่ยนแปลงเกิดขึ้นบนหน้าจอ โดยที่หน้าจอจะทำการแสดงรูปภาพก่อนหน้านี้ค้างไว้จนกว่าจะมีแสงอินฟราเรดมากระทำที่กรอบสี่เหลี่ยมที่กำหนดค่าPixcelsไว้จึงจะมีการเปลี่ยนแปลงเกิดขึ้น โดยการทดลองจะทำการออกแบบเป็น 8 ปุ่ม ซึ่งผลการทดลองแสดงได้ดังรูป ต่อไปนี้



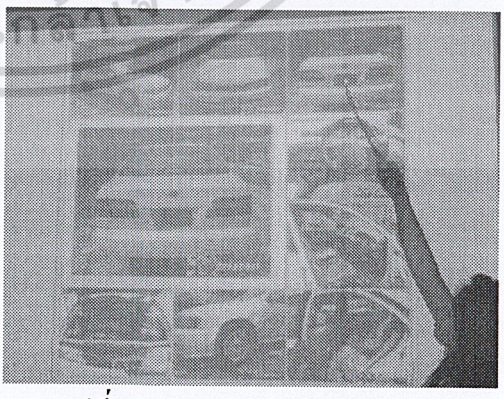
รูปที่ 4.1 แสดงภาพที่ยังไม่มีการกดปุ่ม



รูปที่ 4.3 แสดงภาพการกดภาพด้านหน้ารถ

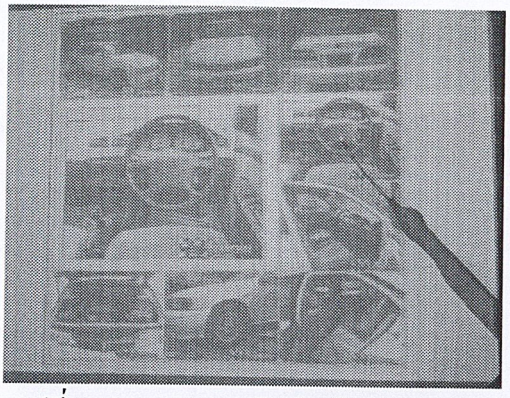


รูปที่ 4.2 แสดงภาพการกดภาพด้านข้างของรถ

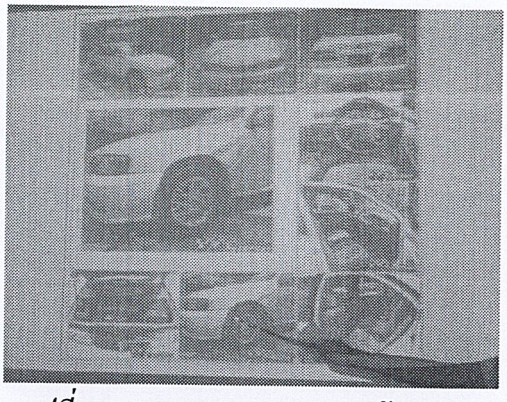


รูปที่ 4.4 แสดงภาพการกดภาพด้านหลังรถ

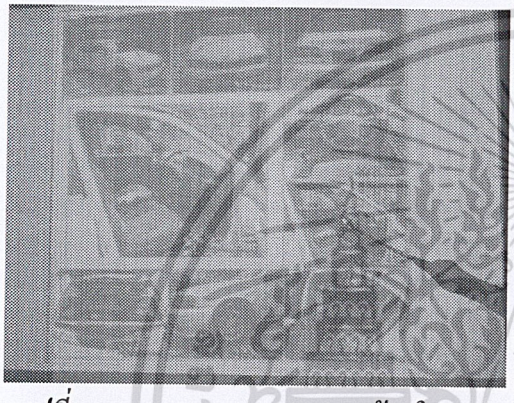
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



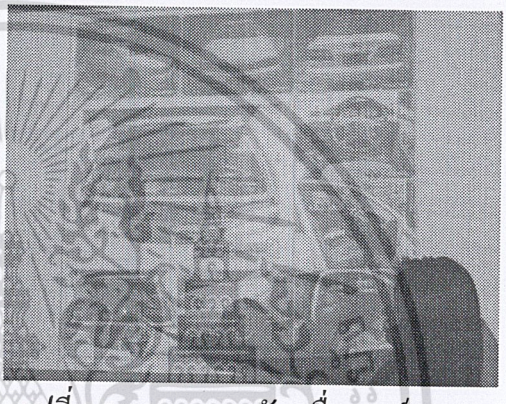
รูปที่ 4.5 การกดภาพด้านในด้านคนขับ



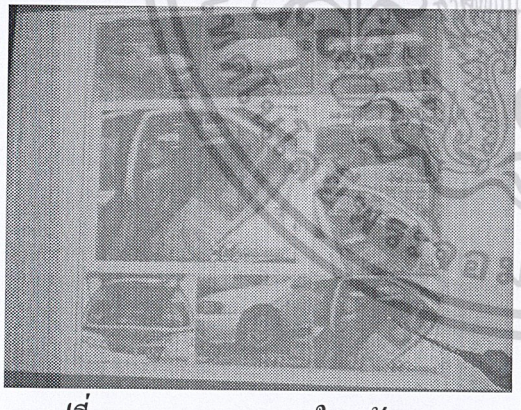
รูปที่ 4.8 แสดงภาพการกดภาพล้อรถ



รูปที่ 4.6 แสดงภาพการกดภาพด้านใน



รูปที่ 4.9 การกดภาพตัวเครื่องยนต์รถ

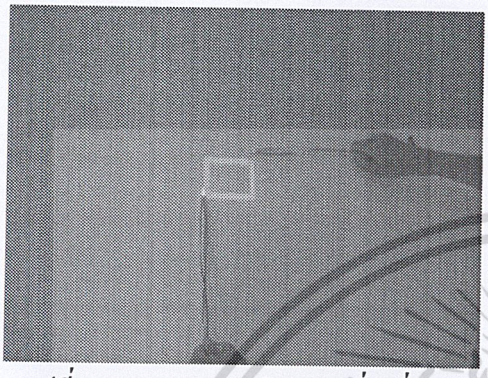


รูปที่ 4.7 การกดภาพภายในหลังรถ

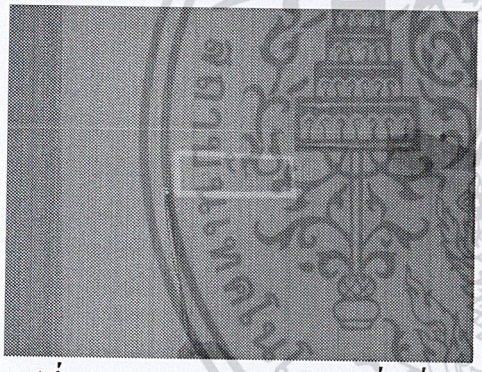
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตอนที่ 2 Two Pointer Detect

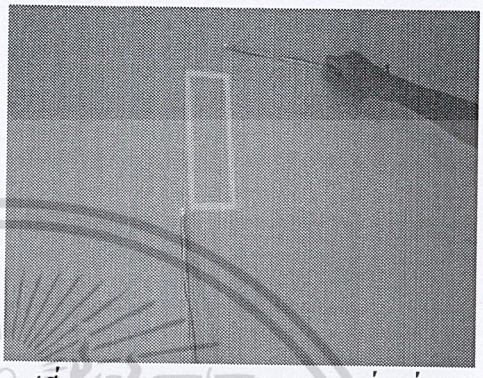
เมื่อเราทำการค้นหาวัตถุที่มีสีขาวได้พิกัดมาสองจุดโดยจะได้ pixel (x1,y1) เป็นจุดที่หนึ่ง และได้พิกัด pixel (x2,y2) เป็นจุดที่ 2 ทำการโปรแกรมให้มีการเชื่อมจุดสองนำมาสร้างเป็นกรอบสี่เหลี่ยม จากนั้นเมื่อเราทำการเลื่อนจุดสองจุดจะได้ดังนี้



รูปที่ 4.10 แสดงการย่อกรอบสี่เหลี่ยม



รูปที่ 4.11 แสดงการขยายกรอบสี่เหลี่ยมออกด้านข้าง



รูปที่ 4.12 แสดงการขยายกรอบสี่เหลี่ยมขึ้นด้านบน



รูปที่ 4.13 แสดงการขยายกรอบสี่เหลี่ยมให้มีขนาดใหญ่ขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทวิจารณ์และสรุปผล

เราสามารถใช้เว็บแคมมาช่วยให้หน้าจอธรรมดาเป็นเสมือนหน้าจอสัมผัสอินฟราเรด และยังสามารถใช้ประโยชน์อย่างอื่นได้อีกด้วย โครงงานนี้มีข้อดี คือ ในขณะนี้เว็บแคมเป็นอุปกรณ์ที่ได้รับความนิยมสำหรับการสนทนาการผ่านอินเตอร์เน็ต ทำให้หาง่ายและมีราคาถูกกว่าจอสัมผัสที่มีใช้ในปัจจุบันสามารถนำมาประยุกต์ใช้ให้เกิดประโยชน์สูงสุดอีกด้วย

แต่ว่าการใช้เว็บแคมก็ยังมีปัญหาในข้อจำกัดของตัวอุปกรณ์เองนั่นก็คือ เว็บแคมบางรุ่นมีเฟรมเรตที่ต่ำเกินไป จึงเกิดความช้าในการจับอินฟราเรดหรือแสงสีขาว และปัญหาอีกอย่างหนึ่งก็คือเราต้องทำการตั้งค่าพิกัดบนหน้าจอให้ตรงกับที่โปรแกรมเขียนไว้ทุกครั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ชูชาติ ปิณฑวิรุจน์. 2550. **การประมวลผลภาพ ดิจิตอลด้วย Matlab** กรุงเทพฯ: สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ชูชาติ ปิณฑวิรุจน์. 2550. **การประมวลผลภาพ ดิจิตอลด้วย C++**. กรุงเทพฯ: สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

สิทธิโชค ยอดระยับ. 2550. **การเขียนโปรแกรม Digital Image Processing**. กรุงเทพฯ: สสท.

เอกพันธ์ ซังแก้ว. 2551. **EX-DESINGZ.** [online]. Available : http://www.ex-designz.net/apidetail.asp?api_id=167

Rafael C. Gonzalez, Richard E. Woods. **Digital Image Processing.** 1992, P1-P28

Hitzuja. 2009. **Thaigaming board.**[online]. Available : http://www.thaigaming.com/downloads-file-10471.htm

Drakenza, Alex Stephen. 2009. **Board Thai.** [online]. Available : http://forums.devx.com/showthread.php?t=144143

Isty Ahmad. 2009. **Cursor Position.** [online]. Available : http://www.webtropy.com/articles/art...f=SetCursorPos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เกี่ยวกับภาคผนวก

ก.1 Code Single Infrared Detect

```
Option Explicit
Dim RoddersMotion As New ClassMotion
Public N As Integer
Public O As Integer
Public U As Integer
Public V As Integer
Public minx As Integer
Public miny As Integer
Public maxx As Integer
Public maxy As Integer

Private Declare Function SendMessage Lib "user32" Alias "SendMessageA" _
(ByVal hWnd As Long, ByVal wMsg As Long, ByVal wParam As Long, _
lParam As Any) As Long

Private Declare Function capCreateCaptureWindow Lib "avicap32.dll" Alias _
"capCreateCaptureWindowA" (ByVal lpszWindowName As String, ByVal _
dwStyle As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, _
ByVal nHeight As Long, ByVal hwndParent As Long, ByVal nID As Long) As Long
Private mCapHwnd As Long

Private Const CONNECT As Long = 1034
Private Const DISCONNECT As Long = 1035
Private Const GET_FRAME As Long = 1084
Private Const COPY As Long = 1054
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Const VIDEOFORMAT As Long = 1065
Private Const VIDEOSOURCE As Long = 1066
Private Const VIDEODISPLAY As Long = 1067
Private Const VIDEOCOMPRESSION As Long = 1070
Private Const PREVIEW As Long = 1074

Private Declare Function GetObject Lib "gdi32" Alias "GetObjectA" _
(ByVal hObject As Long, ByVal nCount As Long, lpObject As Any) As Long

Private Declare Function GetDIBits Lib "gdi32" (ByVal aHDC As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans As _
Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long

Private Declare Function SetDIBits Lib "gdi32" (ByVal hdc As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans _
As Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long
Private Type BITMAP '14 bytes
    bmType As Long
    bmWidth As Long
    bmHeight As Long
    bmWidthBytes As Long
    bmPlanes As Integer
    bmBitsPixel As Integer
    bmBits As Long
End Type
Private Type BITMAPINFOHEADER '40 bytes
    biSize As Long
    biWidth As Long
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
biHeight As Long
    biPlanes As Integer
    biBitCount As Integer
    biCompression As Long
    biSizeImage As Long
    biXPelsPerMeter As Long
    biYPelsPerMeter As Long
    biClrUsed As Long
    biClrImportant As Long
End Type
Private Type RGBQUAD
    rgbBlue As Byte
    rgbGreen As Byte
    rgbRed As Byte
    rgbReserved As Byte
End Type
Private Type BITMAPINFO
    bmiHeader As BITMAPINFOHEADER
    bmiColors As RGBQUAD
End Type
Private Const DIB_RGB_COLORS = 0&
Private Const BI_RGB = 0&
Private Const pixR As Integer = 3
Private Const pixG As Integer = 2
Private Const pixB As Integer = 1
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Find_Object(ByVal picColor As PictureBox)
Dim bitmap_info As BITMAPINFO
Dim pixels() As Byte
Dim bytes_per_scanLine As Integer
Dim pad_per_scanLine As Integer
Dim x As Integer
Dim y As Integer
Dim ave_color As Byte
   minx = picColor.Width
   miny = picColor.Height
   maxx = 0
   maxy = 0
    ' Prepare the bitmap description.
   With bitmap_info.bmiHeader
      .biSize = 40
      .biWidth = picColor.ScaleWidth
      ' Use negative height to scan top-down.
      .biHeight = -picColor.ScaleHeight
      .biPlanes = 1
      .biBitCount = 32
      .biCompression = BI_RGB
      bytes_per_scanLine = ((((.biWidth * .biBitCount) + 31) \ 32) * 4)
      pad_per_scanLine = bytes_per_scanLine - (((.biWidth * .biBitCount) + 7) \ 8)
      .biSizeImage = bytes_per_scanLine * Abs(.biHeight)
   End With
   ' Load the bitmap's data.
   ReDim pixels(1 To 4, 1 To picColor.ScaleWidth, 1 To picColor.ScaleHeight)
   GetDIBits picColor.hdc, picColor.Image, _
      0, picColor.ScaleHeight, pixels(1, 1, 1), _
      bitmap_info, DIB_RGB_COLORS
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' Modify the pixels.
For y = 1 To picColor.ScaleHeight
   For x = 1 To picColor.ScaleWidth
        If (pixels(pixB, x, y) > 246) And (pixels(pixG, x, y) > 246) And _
       (pixels(pixR, x, y) > 246) Then
        pixels(pixR, x, y) = 0
        pixels(pixG, x, y) = 0
        pixels(pixB, x, y) = 0
        If minx > x Then minx = x
        If maxx < x Then maxx = x
        If miny > y Then miny = y
        If maxy < y Then maxy = y
       Else
       End If
   Next x
Next y

   N = (minx + maxx) / 2
   O = (miny + maxy) / 2
   shpLED0(0).Visible = True
   shpLED0(0).BackColor = vbBlue
   shpLED0(0).Width = 32
   shpLED0(0).Height = 32
   shpLED0(0).Top = O - 16
   shpLED0(0).Left = N - 16
   Label1.Caption = "Center is (" & (N - 16) & "," & (O - 16) & ")"
   U = N - 16
   V = O - 16
   check
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub cmdcon_Click()
    mCapHwnd = capCreateCaptureWindow("WebcamCapture", 0, 0, 0, 320, 240, _
    Me.hWnd, 0)
    DoEvents
    SendMessage mCapHwnd, CONNECT, 0, 0
    Picture3.Width = 320 * Screen.TwipsPerPixelX
    Picture3.Height = 240 * Screen.TwipsPerPixelY
    picColor.AutoSize = True
    picColor.AutoRedraw = True
    shpLED0(0).Visible = False
    Timer1.Enabled = True
End Sub
Private Sub cmdDiscon_Click()
 DoEvents
    SendMessage mCapHwnd, DISCONNECT, 0, 0
    Timer1.Enabled = False
End Sub
Private Sub Form_Load()
picFovea.Width = 1500
picFovea.Height = 1500
End Sub
Private Sub Timer1_Timer()
    SendMessage mCapHwnd, GET_FRAME, 0, 0
    SendMessage mCapHwnd, COPY, 0, 0
    Picture3.Picture = Clipboard.GetData
    Clipboard.Clear
    picColor.Picture = Picture3.Image
    picColor.Visible = True
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
' Convert to gray scale.
    Screen.MousePointer = vbHourglass
    DoEvents
    Find_Object picColor
    Screen.MousePointer = vbDefault
     Call RoddersMotion.showFovea(Picture3, picFovea)
End Sub
Private Sub check()
If U > 35 And U < 110 And V > 0 And V < 55 Then
picZoom.Picture = pic1.Image
End If
If U > 110 And U < 185 And V > 0 And V < 55 Then
picZoom.Picture = pic2.Image
End If
If U > 185 And U < 260 And V > 0 And V < 55 Then
picZoom.Picture = pic3.Image
End If
If U > 185 And U < 260 And V > 55 And V < 110 Then
picZoom.Picture = pic4.Image
End If
If U > 185 And U < 260 And V > 110 And V < 165 Then
picZoom.Picture = pic5.Image
End If
If U > 185 And U < 260 And V > 165 And V < 220 Then
picZoom.Picture = pic6.Image
End If
If U > 110 And U < 185 And V > 165 And V < 220 Then
picZoom.Picture = pic7.Image
End If
If U > 35 And U < 110 And V > 165 And V < 220 Then
picZoom.Picture = pic8.Image
End If
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก.2 Code Two Pointer Detect

```
Option Explicit
Public N As Integer
Public O As Integer
Public U As Integer
Public V As Integer
Public Qty, Qty2, Qty3, Qty4 As Integer

Private Declare Function getpixel Lib "gdi32" (ByVal hdc As Long, _
ByVal x As Long, ByVal y As Long) As Long
Private Declare Function SendMessage Lib "user32" Alias "SendMessageA" _
(ByVal hWnd As Long, ByVal wMsg As Long, ByVal wParam As Long, _
lParam As Any) As Long

Private Declare Function capCreateCaptureWindow Lib "avicap32.dll" Alias _
"capCreateCaptureWindowA" (ByVal lpszWindowName As String, ByVal _
dwStyle As Long, ByVal x As Long, ByVal y As Long, ByVal nWidth As Long, _
ByVal nHeight As Long, ByVal hwndParent As Long, ByVal nID As Long) As Long

Private mCapHwnd As Long
Private Declare Function LineTo Lib "gdi32" (ByVal hdc As Long, _
ByVal x As Long, ByVal y As Long) As Long

Private Const CONNECT As Long = 1034
Private Const DISCONNECT As Long = 1035
Private Const GET_FRAME As Long = 1084
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Const COPY As Long = 1054
Private Const VIDEOFORMAT As Long = 1065
Private Const VIDEOSOURCE As Long = 1066
Private Const VIDEODISPLAY As Long = 1067
Private Const VIDEOCOMPRESSION As Long = 1070
Private Const PREVIEW As Long = 1074

Private Declare Function GetObject Lib "gdi32" Alias "GetObjectA" _
(ByVal hObject As Long, ByVal nCount As Long, lpObject As Any) As Long

Private Declare Function GetDIBits Lib "gdi32" (ByVal aHDC As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans As _
Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long

Private Declare Function SetDIBits Lib "gdi32" (ByVal hdc As Long, _
ByVal hBitmap As Long, ByVal nStartScan As Long, ByVal nNumScans _
As Long, lpBits As Any, lpBI As BITMAPINFO, ByVal wUsage As Long) As Long
Private Type BITMAP '14 bytes
    bmType As Long
    bmWidth As Long
    bmHeight As Long
    bmWidthBytes As Long
    bmPlanes As Integer
    bmBitsPixel As Integer
    bmBits As Long
End Type
Private Type BITMAPINFOHEADER '40 bytes
    biSize As Long
    biWidth As Long
    biHeight As Long
    biPlanes As Integer
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    biBitCount As Integer
    biCompression As Long
    biSizeImage As Long
    biXPelsPerMeter As Long
    biYPelsPerMeter As Long
    biClrUsed As Long
    biClrImportant As Long
End Type
Private Type RGBQUAD
    rgbBlue As Byte
    rgbGreen As Byte
    rgbRed As Byte
    rgbReserved As Byte
End Type
Private Type BITMAPINFO
    bmiHeader As BITMAPINFOHEADER
    bmiColors As RGBQUAD
End Type
Private Const DIB_RGB_COLORS = 0&
Private Const BI_RGB = 0&
Private Const pixR As Integer = 3
Private Const pixG As Integer = 2
Private Const pixB As Integer = 1
Private Sub Find_Object(ByVal picColor As PictureBox)
Dim bitmap_info As BITMAPINFO
Dim pixels() As Byte
Dim bytes_per_scanLine As Integer
Dim pad_per_scanLine As Integer
Dim x As Integer
Dim y As Integer
Dim ave_color As Byte
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ' Prepare the bitmap description.
  With bitmap_info.bmiHeader
    .biSize = 40
    .biWidth = picColor.ScaleWidth
    ' Use negative height to scan top-down.
    .biHeight = -picColor.ScaleHeight
    .biPlanes = 1
    .biBitCount = 32
    .biCompression = BI_RGB
    bytes_per_scanLine = ((((.biWidth * .biBitCount) + 31) \ 32) * 4)
    pad_per_scanLine = bytes_per_scanLine - (((.biWidth * .biBitCount) + 7) \ 8)
    .biSizeImage = bytes_per_scanLine * Abs(.biHeight)
  End With
  ' Load the bitmap's data.
  ReDim pixels(1 To 4, 1 To picColor.ScaleWidth, 1 To picColor.ScaleHeight)
  GetDIBits picColor.hdc, picColor.Image, _
    0, picColor.ScaleHeight, pixels(1, 1, 1), _
    bitmap_info, DIB_RGB_COLORS
  ' Modify the pixels.
  Dim i As Integer
 Dim j As Integer
 Dim object As Boolean
 object = False
 For x = 1 To picColor.ScaleWidth
  For y = 1 To picColor.ScaleHeight
    If (pixels(pixR, x, y) > 252) And (pixels(pixG, x, y) > 252) And _
        (pixels(pixB, x, y) > 252) Then
      DX(1) = x
      DY(1) = y
      object = True
      Qty = Qty + 1
    If Qty = 0 Then
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            DX(2) = x
            DY(2) = y
          End If
        End If
      Next y
    Next x
    Label1.Caption = "center( " & DX(1) & "),( " & DY(1) & "),( " & DX(2) & "), _
        ( " & DY(2) & ")"
    N = (DX(1) + 45)
    O = (DY(1) + 50)
    U = (DX(2) + 45)
    V = (DY(2) + 50)
  check
  End Sub

  Private Sub cmdcon_Click()
    mCapHwnd = capCreateCaptureWindow("WebcamCapture", 0, 0, 0, 320, 240, _
    Me.hWnd, 0)
    DoEvents
    SendMessage mCapHwnd, CONNECT, 0, 0
    Picture3.Width = 320 * Screen.TwipsPerPixelX
    Picture3.Height = 240 * Screen.TwipsPerPixelY
    picColor.AutoSize = True
    picColor.AutoRedraw = True
    Timer1.Enabled = True
  End Sub

  Private Sub cmdDiscon_Click()
   DoEvents
    SendMessage mCapHwnd, DISCONNECT, 0, 0
    Timer1.Enabled = False
  End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Private Sub Timer1_Timer()
    SendMessage mCapHwnd, GET_FRAME, 0, 0
    SendMessage mCapHwnd, COPY, 0, 0
    Picture3.Picture = Clipboard.GetData
    Clipboard.Clear
    picColor.Picture = Picture3.Image
    picColor.Visible = True
    ' Convert to gray scale.
    Screen.MousePointer = vbHourglass
    DoEvents
    Find_Object picColor
    Screen.MousePointer = vbDefault
End Sub

Private Sub check()
Line1.X1 = (N * 30)
Line1.Y1 = (O * 30)
Line1.X2 = (U * 30)
Line1.Y2 = (O * 30)
Line2.X1 = (U * 30)
Line2.Y1 = (O * 30)
Line2.X2 = (U * 30)
Line2.Y2 = (V * 30)
Line3.X1 = (U * 30)
Line3.Y1 = (V * 30)
Line3.X2 = (N * 30)
Line3.Y2 = (V * 30)
Line4.X1 = (N * 30)
Line4.Y1 = (V * 30)
Line4.X2 = (N * 30)
Line4.Y2 = (O * 30)
End Sub
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้