สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

การควบคุมการทำงานของโปรแกรม MS PowerPoint ด้วยการให้สัญญาณมือ

CONTROLLING MS POWERPOINT SLIDE TRANSITION USING HAND SINGNALS

T 1 0 3 0 1 6

ปฐมาภรณ์ หรัญลาภ
ปริญญา เทพโพธา
พงศ์พันธ์ สมบัติพานิช

เลขหมู่.....
เลขทะเบียน 103016
วัน,เดือน,ปี 24 ส.ค. 2552

b 12097251
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2551

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การควบคุมการทำงานของโปรแกรม MS PowerPoint ด้วยการให้สัญญาณมือ

CONTROLLING MS POWERPOINT SLIDE TRANSITION USING HAND SINGNALS

ผู้จัดทำ

1. นางสาวปฐมาภรณ์ หิรัญลาภ รหัสประจำตัว 48010493
2. นายปริญญา เมฆโหรา รหัสประจำตัว 48010521
3. นายพงศ์พัธน์ สมบัติพานิช รหัสประจำตัว 48010562

_____ อาจารย์ที่ปรึกษา

(ดร. ปกรณ์ วัฒนจตุรพร)

_____ อาจารย์ที่ปรึกษา

(อาจารย์จรรยักษ์ สิทธิกร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การควบคุมการทำงานของโปรแกรม MS PowerPoint ด้วยการให้สัญญาณมือ

นางสาวปฐมาภรณ์	หิรัญลาภ	48010493
นายปริญญา	เมฆโหรา	48010521
นายพงศ์พัธน์	สมบัติพานิช	48010562
ดร. ปกรณ์	วัฒนจตุรพร	อาจารย์ที่ปรึกษา
อาจารย์จิระศักดิ์	สิทธิกร	อาจารย์ที่ปรึกษาร่วม

ปีการศึกษา 2551

บทคัดย่อ

โครงงานนี้เป็นโครงงานที่มุ่งเน้นการพัฒนาโปรแกรมสำหรับควบคุมการทำงานของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ (Microsoft PowerPoint) ให้สามารถสั่งงานได้ผ่านทางกล้องเว็บแคมโดยการให้สัญญาณมือ ทำให้ผู้ใช้สามารถสั่งการในระหว่างการนำเสนอได้แทนการสั่งการผ่านคีย์บอร์ดโดยตรง โครงการนี้มีจุดประสงค์เพื่อพัฒนาเทคโนโลยีใหม่ ที่รองรับการทำงานของผู้ใช้ อีกทั้งยังอำนวยความสะดวกและเพิ่มลูกเล่นที่หลากหลายในการใช้งานโปรแกรมสำหรับนำเสนองาน โปรแกรมนี้จะทำงานร่วมกับ โปรแกรมไมโครซอฟท์เพาเวอร์พอยท์โดยจะรับภาพจากกล้องเว็บแคม แล้วนำมาผ่านกระบวนการประมวลผลภาพเพื่อหาภาพสัญญาณมือ จากนั้นจึงนำภาพสัญญาณมือไปวิเคราะห์ และตีความหมายของสัญญาณมือโดยวิธีการรู้จำภาพและส่งคำสั่งไปควบคุมโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ โปรแกรมสามารถออกคำสั่งควบคุมการเปลี่ยนสไลด์ได้ ประกอบด้วยคำสั่งเริ่มและจบสไลด์การนำเสนอ เปลี่ยนหน้าสไลด์ถัดไปและก่อนหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CONTROLLING MS POWERPOINT SLIDE TRANSITION USING HAND SINGNALS

Ms. Patamaporn	Hirunlap	48010493
Mr. Parinya	Mekhora	48010521
Mr. Pongpat	Sombatpanich	48010562
Dr. Pakorn	Watanachaturaporn	Advisor
Mr. Jirasak	Sittigorn	Co-Advisor

Academic Year 2008

ABSTRACT

This project aims to develop a program that has an ability to control slide transitions in the Microsoft PowerPoint program using hand signals thought a webcam. The program allows a user to control the slide transition during his/her presentation instead of controlling the transition from the keyboard. The purpose of this project is to develop a new technology which can provide convenient and useful presentation functions. This program works with the Microsoft PowerPoint program by retrieving images from a webcam and processing them via an image processing unit in order to detect the hand signals. Then, the hand signals are analyzed and classified by a recognition method. A command is, then, sent to the Microsoft PowerPoint program. This program can control the transition of the slides including a transition to the start, the end, the next and the previous page.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ผู้พัฒนาขอขอบพระคุณอาจารย์ที่ปรึกษาเป็นอย่างสูง ได้แก่ อาจารย์ปกรณ์ วัฒนจตุพร และอาจารย์จิรศักดิ์ สิทธิกร ซึ่งคอยให้คำปรึกษา และช่วยดูแลการทำงานของคณะผู้พัฒนามาโดยตลอด ทำให้งานคืบหน้าอย่างต่อเนื่อง

ผู้พัฒนาขอขอบพระคุณครอบครัวของผู้พัฒนา ที่คอยเป็นกำลังใจให้แก่คณะผู้พัฒนาในการทำงานให้สามารถลุล่วงไปได้ด้วยดี

ผู้พัฒนาขอขอบคุณเพื่อนๆที่สละเวลาในการมาช่วยทดสอบระบบให้แก่คณะผู้พัฒนา เพื่อให้ได้ทราบถึงประสิทธิภาพของระบบ และจุดบกพร่องของระบบเพื่อนำไปแก้ไขข้อผิดพลาดและจุดบกพร่องให้มีประสิทธิภาพยิ่งขึ้นต่อไป

ผู้พัฒนาขอขอบคุณเพื่อนร่วมห้องปฏิบัติการที่ทำงานร่วมกันมาโดยตลอด ทำให้การทำงานเป็นไปอย่างสนุกสนาน และช่วยเสริมสร้างบรรยากาศในห้องปฏิบัติการ

ผู้พัฒนาขอขอบคุณการแข่งขัน National Software Contest ที่คัดเลือกให้เราตกรอบแรก พร้อมทั้งให้คติประจำใจ คือ **"ขี่ช้างจับตั๊กแตน"**

นางสาวปฐมาภรณ์ หิรัญลาภ
นายปริญญา เมฆโหรา
นายพงศ์พัธน์ สมบัติพานิช

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VIII

สารบัญรูป IX

บทที่ 1 บทนำ 1

1.1 ความเป็นมาของปัญหา 1

1.2 วัตถุประสงค์ของโครงงาน 1

1.3 ประโยชน์ที่คาดว่าจะได้รับ 1

1.4 ขอบเขตของโครงงาน 2

1.5 ส่วนประกอบของรายงาน 2

บทที่ 2 ทฤษฎีที่เกี่ยวข้อง 3

2.1 ความรู้ข้อมูลภาพดิจิตอล 3

2.1.1 พิกเซล 3

2.1.2 รูปภาพขาวดำ 4

2.1.3 ภาพระดับสีเทา 4

2.1.4 รูปภาพสี 5

2.1.5 ระบบสีโมเดล HSL (HSL Color Model) 5

2.2 ไฟล์ข้อมูลภาพชนิดบิตแมป 7

2.2.1 รูปแบบของไฟล์ข้อมูลภาพชนิดบิตแมป 7

2.2.2 โครงสร้างของไฟล์ข้อมูลภาพชนิดบิตแมป 7

2.2.2.1 ข้อมูลเฮดเดอร์ (Header) 7

2.2.2.2 ข้อมูลจานสี (Palette) 8

2.2.2.3 ข้อมูลภาพ (Data) 8

2.2.3 การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป 8

2.2.3.1 แบบบีบอัดข้อมูล 8

2.2.3.2 แบบไม่ได้บีบอัดข้อมูล 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

2.2.4 เฮดเดอร์ไฟล์ข้อมูลชนิดบิตแมป 8
2.3 สัญญาณรบกวนแบบซอลท์และเปปเปอร์ (Salt and Pepper Noise) 9
2.4 การลดสัญญาณรบกวน (Noise) โดยใช้การกรองแบบค่ามัธยฐาน 10
2.5 การแปลงภาพสีเป็นภาพขาว-ดำด้วยวิธีเทรชโฮล 12
2.5.1 การหาค่าเทรชโฮลโดยการกำหนดล่วงหน้า 12
2.5.2 การหาค่าเทรชโฮลจากค่ากลาง 12
2.6 การหาจุดศูนย์กลางมวลของวัตถุ 13
2.7 การทำนอร์มัลไลเซชั่น 13
2.8 การลบกันของภาพพื้นหลัง 14
2.9 การหาขอบภาพ 15
2.10 การทำให้เส้นบาง 16
2.11 การรู้จำ 17
2.11.1 การรู้จำแบบเข้าคู่รูปแบบ (Template Matching) 17
2.11.2 การรู้จำแบบวิเคราะห์ทางโครงสร้าง (Structural Analysis) 17
2.12 การติดต่อกล้องโดยใช้เอวีไอแคป 32 18
2.12.1 ข้อดีของการใช้เอวีไอแคป 32 18
2.12.2 ข้อเสียของการใช้เอวีไอแคป 32 19
2.13 การควบคุมโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์โดยใช้ฟังก์ชัน keybd_event และ mouse_event 19
บทที่ 3 หลักการทำงานและโครงสร้างของโปรแกรม 20
3.1 หลักการทำงานโดยรวมของโปรแกรม 20
3.2 การเตรียมข้อมูลภาพก่อนทำการประมวลผล (Preprocessing) 22
3.2.1 การแปลงภาพสีเป็นภาพขาว-ดำด้วยวิธีเทรชโฮล 22
3.2.2 การกำจัดสัญญาณรบกวน 24
3.2.3 การตีกรอบวัตถุที่เราสนใจ 25
3.3 การทำนอร์มัลไลเซชั่น 26
3.4 การหาจุดศูนย์กลางมวลรวมของวัตถุ 26
3.5 การรู้จำภาพ 29

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

3.5.1 วิธีการเข้าคู่รูปแบบ 29

3.5.2 วิธีการวิเคราะห์ทางโครงสร้าง 30

3.6 การเขียนไลบรารีติดต่อเว็บแคม 33

3.6.1 Platform Invoke (P/Invoke) และ API Constants 33

3.6.1.1 Platform Invoke (P/Invoke) 33

3.6.1.2 เอพีไอคอนสแตนซ์ (API Constants) 35

3.6.2 3.6.2 การเขียนโค้ดเพื่อติดต่อกับเว็บแคม 35

3.6.2.1 เซตค่าแคปเจอร์วินโดว์ (Capture Window) 35

3.6.2.2 การกำหนดค่าต่างๆเพื่อติดต่อกับกล้อง 35

3.6.2.3 การสั่งรับภาพจากกล้องเว็บแคม 36

3.6.2.4 ยกเลิกการเชื่อมต่อเพื่อรับข้อมูลจากกล้อง 36

3.6.2.5 การจับภาพจากกล้อง 37

3.7 การเขียนไลบรารีติดต่อโปรแกรมไม่โครซอฟท์เพาเวอร์พอยท์ 38

3.7.1 การประกาศค่าคงที่ 38

3.7.2 ฟังก์ชันที่ใช้งาน 39

3.7.3 การทำงานของโปรแกรม 39

บทที่ 4 การทดลองและผลการทดลอง 43

4.1 การทดลองจับภาพสัญญาณมือโดยใช้ถุงมือสี 43

4.1.1 การทดลองส่วนของการรู้จำแบบเข้าคู่รูปแบบโดยใช้ถุงมือสี 43

4.1.2 การทดลองส่วนของการรู้จำแบบการวิเคราะห์ทางโครงสร้างโดยใช้ถุงมือสี 45

4.2 การทดลองจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสี 46

4.2.1 การทดลองส่วนของการรู้จำแบบเข้าคู่รูปแบบโดยไม่ใช้ถุงมือสี 47

4.2.2 การทดลองส่วนของการรู้จำแบบการวิเคราะห์ทางโครงสร้างโดยไม่ใช้ถุงมือสี 47

4.3 การทดลองหาค่าเปอร์เซ็นต์ความถูกต้องที่เหมาะสมสำหรับการเปรียบเทียบในวิธีการเข้าคู่รูปแบบ 49

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 5 บทสรุปและวิจารณ์ 52

5.1 สรุปการทำงาน 52

5.2 ปัญหาและอุปสรรค 53

5.3 ข้อเสนอแนะ 53

บรรณานุกรม 54

ภาคผนวก ก ขั้นตอนการติดตั้งโปรแกรม 55

ภาคผนวก ข คู่มือการใช้งาน 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 ข้อมูลต่างๆ ในส่วนของเฮดเดอร์ของไฟล์ข้อมูลชนิดบิตแมป 9

4.1 ผลการทดลองการรู้จำแบบเข้าคู่รูปแบบโดยใช้ถุงมือสี 44

4.2 ผลการทดลองการวิเคราะห์ทางโครงสร้างโดยใช้ถุงมือสี 45

4.3 ผลการทดลองการรู้จำแบบเข้าคู่รูปแบบโดยไม่ใช้ถุงมือสี 47

4.4 ผลการทดลองการวิเคราะห์ทางโครงสร้างโดยไม่ใช้ถุงมือสี 48

4.5 สรุปผลการทดลอง 49

4.6 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 60 % 49

4.7 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 70 % 50

4.8 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 80 % 50

4.9 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 90 % 50

4.10 ผลความถูกต้องคิดเป็นเปอร์เซ็นต์ ณ ที่ค่าเปอร์เซ็นต์ความถูกต้องค่าต่างๆ 51

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	ลักษณะของตำแหน่งพิกเซล	3
2.2	ลักษณะของรูปภาพขาว - ดำ	4
2.3	ระดับความเข้มแสงของภาพระดับสีเทา	4
2.4	ลักษณะของรูปภาพสี	5
2.5	แบบจำลองโมเดลสี HSL	6
2.6	ความสัมพันธ์ระหว่างค่าประกอบ Hue และ Saturation โดยที่ค่า Lightness เป็น 0.5	6
2.7	ความสัมพันธ์ระหว่างค่า Hue และ Lightness โดยที่ค่า Saturation เป็น 1	7
2.8	ภาพที่มีสัญญาณรบกวนแบบซอลท์และเปปเปอร์	9
2.9	ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 1	10
2.10	ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 2	10
2.11	ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 3	11
2.12	ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 4	11
2.13	การเปลี่ยนแปลงของภาพหลังจากการกรองแบบค่ามัธยฐาน	11
2.14	จุดศูนย์กลางมวลของวัตถุ	13
2.15	ตัวอย่างภาพพื้นหลัง	14
2.16	ตัวอย่างภาพปัจจุบัน	14
2.17	ตัวอย่างค่าความแตกต่างระหว่างภาพพื้นหลังและภาพปัจจุบัน	15
2.18	การทำไล่ตามขอบภาพ	16
2.19	รูปแบบของ B^1, B^2, B^3, B^4, B^5, B^6, B^7 และ B^8	17
2.20	Video Capture Messages	18
3.1	หลักการทำงานของโปรแกรมโดยรวม	21
3.2	ภาพที่ถ่ายได้จากกล้องเว็บแคม	23
3.3	ภาพที่ได้หลังจากการทำเทรชโฮลที่ใช้ค่าเทรชโฮลที่เหมาะสม	23
3.4	ภาพที่ได้หลังจากการทำเทรชโฮลที่ใช้ค่าเทรชโฮลที่สูงหรือต่ำเกินไป	24
3.5	วิธีการกำจัดสัญญาณรบกวนโดยใช้หน้าต่างขนาด 3 × 3	24
3.6	ภาพหลังกำจัดสัญญาณรบกวน	25
3.7	ภาพหลังจากการทำการตีกรอบ	25
3.8	จุดที่ทำให้คานอยู่ในลักษณะที่สมดุล	26

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ **หน้า**

3.9 แรงที่กระทำต่อมวล m 26

3.10 มวลที่ต่อกันตั้งแต่ m_1 ถึง m_n 27

3.11 การหาจุดศูนย์กลางมวลของ m1 27

3.12 ภาพรูปแบบที่กำหนดไว้ 29

3.13 ความสัมพันธ์ระหว่างตำแหน่งของจุดปลายนิ้วและจุดศูนย์กลางมวล 30

3.14 การวิเคราะห์สัญญาณมือเพื่อเริ่มการนำเสนอ 31

3.15 การวิเคราะห์สัญญาณมือเลื่อนสไลด์ไปหน้าถัดไป 31

3.16 การวิเคราะห์สัญญาณมือเพื่อสิ้นสุดการนำเสนอ 32

3.17 การวิเคราะห์สัญญาณมือเลื่อนสไลด์ไปหน้าก่อนหน้า 32

3.18 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน SendMessage 33

3.19 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน SetWindowPos 33

3.20 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน DestroyWindow 34

3.21 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน capGetDriverDescriptionA 34

3.22 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน capCreateCaptureWindowA 34

3.23 การประกาศเอพีไอคอนสแตนซ์ 35

3.24 การติดต่อกับ Handle Window 35

3.25 การกำหนดค่าต่างๆ เพื่อติดต่อกับกล้อง 36

3.26 การสั่งรับภาพเฟรมถัดไป 36

3.27 การยกเลิกการเชื่อมต่อข้อมูลจากกล้อง 36

3.28 การจับภาพจากกล้อง 37

3.29 ค่าคงที่ของคีย์บอร์ด 38

3.30 ค่าคงที่ของเมาส์ 38

3.31 ฟังก์ชั่นที่ใช้งานในการใช้งานวินโดว์แมสเสจ 39

3.32 ฟังก์ชั่น GetTaskWindows 40

3.33 ฟังก์ชั่น SearchTextForMatch 40

3.34 ฟังก์ชั่น SetHandleWindow 41

3.35 ฟังก์ชั่น PageUp 41

3.36 ฟังก์ชั่น PressLeftMouse 42

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ **หน้า**

4.1 ท่าทางของสัญญาณมือที่ใช้ในการทดลอง 44

4.2 แผนภูมิเปรียบเทียบการทดลองจับภาพสัญญาณมือโดยใช้ถุงมือสี 46

4.3 แผนภูมิเปรียบเทียบการทดลองจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสี 48

4.4 แผนภูมิเปรียบเทียบการทดลองใช้ค่าเปอร์เซ็นต์ความถูกต้องที่ค่าต่างๆ 51

ก.1 หน้าจอเริ่มต้นการติดตั้ง 55

ก.2 หน้าจอเลือกโฟลเดอร์สำหรับการติดตั้ง 55

ก.3 หน้าจอยืนยันการติดตั้ง 56

ก.4 หน้าจอสิ้นสุดการติดตั้ง 56

ข.1 ตัวอย่างโปรแกรม 57

ข.2 ตัวเลือกที่อยู่ในเมนู Processing Type 58

ข.3 ภาพปุ่ม START 58

ข.4 การเรียนรู้สี 59

ข.5 การกำหนดพื้นที่ที่สนใจ 60

ข.6 ภาพในโหมดปกติ 60

ข.7 ภาพในโหมดรูปภาพขาว - ดำ 61

ข.8 เมนู Detection Type 61

ข.9 ท่าทางของสัญญาณมือที่ใช้ในการออกคำสั่ง 62

ข.10 ผลการทำงานเมื่อออกคำสั่ง 63

ข.11 การประมวลผลรูปภาพขาว - ดำ 64

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันกล้องเว็บแคม (Webcam) ได้กลายเป็นอุปกรณ์ที่ใช้กันอย่างแพร่หลาย เพราะด้วยราคาที่ถูกลงอย่างมากเมื่อเทียบกับในอดีต และด้วยเทคโนโลยีทางด้านการสื่อสารที่ไร้พรมแดนทำให้การส่งข้อมูลภาพพร้อมเสียงนั้นเป็นไปด้วยความสะดวกรวดเร็ว ผู้ใช้ส่วนใหญ่จึงหันมาเล่นกล้องเว็บแคมกันมากขึ้น แต่ผู้ใช้โดยส่วนมากจะใช้ประโยชน์จากกล้องเว็บแคมเพียงแค่ในการสื่อสารเท่านั้น ซึ่งทางผู้จัดทำมีความเห็นว่ากล้องเว็บแคมยังสามารถใช้ประโยชน์ในด้านอื่นได้อีก อาทิเช่น นำมาควบคุมการเปลี่ยนสไลด์ของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ (Microsoft PowerPoint) โดยอาศัยการกระบวนการของการประมวลผลภาพ (Image Processing) ซึ่งเป็นวิธีการที่ยังไม่แพร่หลายมากนักในประเทศไทย จึงเป็นการดีที่จะได้ทำการศึกษาในด้านนี้เพื่อเป็นทางเลือกเสริมสำหรับการควบคุมการเปลี่ยนหน้าสไลด์ของไมโครซอฟท์เพาเวอร์พอยท์นอกเหนือจากการควบคุมด้วยเมาส์ (Mouse) หรือคีย์บอร์ด (Keyboard) ทำให้เกิดความก้าวหน้าทางเทคโนโลยี โดยการคิดค้นสิ่งใหม่ หรือทางเลือกใหม่ และสามารถนำมาประยุกต์ใช้กับในชีวิตประจำวันได้

1.2 วัตถุประสงค์ของโครงงาน

- เพื่อศึกษาหลักการและเทคนิคทางการประมวลผลภาพที่เกี่ยวข้องในการประมวลภาพสัญลักษณ์มือสำหรับการใช้ในโครงงาน
- เพื่อศึกษาการเขียนโปรแกรมติดต่อกับกล้องเว็บแคม
- เพื่อพัฒนาโปรแกรมสำหรับเพิ่มความสามารถของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- ได้รับความรู้ ความเข้าใจในด้านการประมวลผลภาพ และการเขียนโปรแกรมที่เกี่ยวข้องกับโครงงาน
- สามารถพัฒนาโปรแกรมที่เป็นประโยชน์และนำไปใช้จริงได้
- การนำเสนองานผ่านโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ โดยใช้โปรแกรมที่พัฒนาขึ้นสามารถทำได้ตามขอบเขตที่กำหนดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตของโครงงาน

- สามารถควบคุมคำสั่งพื้นฐานในโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ เช่น การเริ่มและจบการนำเสนอ การเปลี่ยนหน้าสไลด์
- ผู้นำเสนอสามารถเลือกสวมถุงมือสี หรือไม่ต้องสวมถุงมือในขณะใช้งานโปรแกรมก็ได้
- ระยะการใช้งานจะต้องอยู่ห่างจากกล้องไม่เกิน 1.5 เมตร
- ความเข้มของแสงในห้องจะต้องอยู่ในระดับที่เหมาะสม
- ฉากหลังจะต้องไม่มีวัตถุที่มีสีใกล้เคียงกับสีที่เราสนใจ
- กล้องเว็บแคมที่ใช้ต้องเป็นรุ่นที่สามารถปิดฟังก์ชันของการปรับความสว่างของภาพโดยอัตโนมัติได้

1.5 ส่วนประกอบของรายงาน

รายงานฉบับนี้ประกอบด้วยเนื้อหาจำนวน 5 บทดังต่อไปนี้

บทที่ 1 กล่าวถึงความเป็นมาของปัญหา วัตถุประสงค์ของโครงงาน ประโยชน์ที่คาดว่าจะได้รับ ขอบเขตของโครงงาน และส่วนประกอบของรายงานฉบับนี้

บทที่ 2 กล่าวถึงทฤษฎีต่างๆ ที่เกี่ยวข้องกับโครงงาน ซึ่งประกอบด้วยความรู้พื้นฐานเกี่ยวกับข้อมูลภาพดิจิตอล (Digital Image) ไฟล์ข้อมูลภาพชนิดบิตแมป (Bitmap) ประเภทของสัญญาณรบกวน (Noise) และวิธีการกำจัดสัญญาณรบกวน การแปลงภาพสีเป็นรูปภาพขาว - ดำ (Binary Image) ด้วยวิธีเทรชโฮล (Threshold) การหาจุดศูนย์กลางมวลรวม (Center of Mass) การทำนอร์มัลไลเซชั่น (Normalization) การหาขอบภาพ (Edge Detection) การทำการลบกันของภาพพื้นหลัง (Background Subtraction) การทำให้เส้นบาง (Thinning) การรู้จำภาพ (Recognition) การติดต่อกล้องโดยใช้เอวีไอแคป 32 (AVICAP 32) และการควบคุมโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์โดยใช้ฟังก์ชัน keybd_event และ mouse_event

บทที่ 3 กล่าวถึงโครงสร้าง และหลักการโดยรวมของโปรแกรม ตั้งแต่การเตรียมข้อมูลภาพก่อนนำมาประมวลผล (Preprocessing) การตีความหมายของภาพ และการสั่งงานไปยังโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์

บทที่ 4 กล่าวถึงการทดลอง และผลการทดลอง

บทที่ 5 กล่าวถึงสรุปผลการดำเนินงานทั้งหมดที่ได้ทำในภาคการศึกษานี้ ปัญหาที่พบในโครงงานและข้อเสนอแนะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

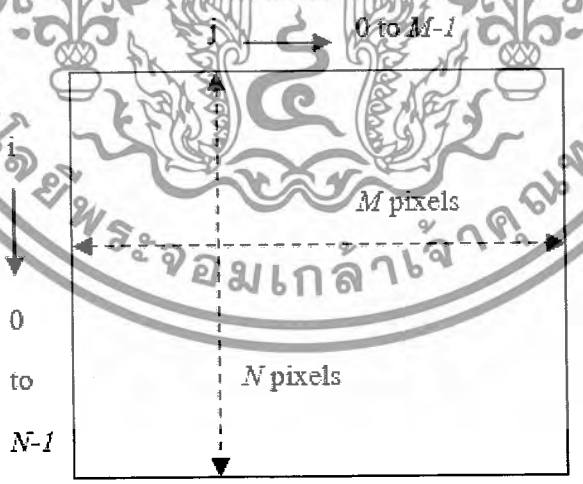
บทที่ 2
ทฤษฎีที่เกี่ยวข้อง

2.1 ความรู้ข้อมูลภาพดิจิตอล

โดยปกติแล้วสายตาของบุคคลทั่วไปจะมองเห็นภาพทิวทัศน์วิวต่างๆ เป็นลักษณะอนาล็อก (Analog) ซึ่งสามารถอธิบายได้ด้วยคณิตศาสตร์ที่มีตัวแปรแบบนับได้อย่างต่อเนื่อง แต่คอมพิวเตอร์จะใช้เลขฐานสองเป็นหลักในการคำนวณ เมื่อนำข้อมูลภาพมาทำการคำนวณด้วยคอมพิวเตอร์จึงต้องแปลงเป็นภาพดิจิตอล

ภาพดิจิตอลมีอยู่หลายชนิด เช่น รูปภาพขาว - ดำ, รูปภาพระดับสีเทา (Gray Level Image) และรูปภาพสี (Color Image) โดยรูปภาพแต่ละชนิดอาจได้มาด้วยวิธีการแตกต่างกัน เช่น อาจได้มาจากการถ่ายภาพของกล้องดิจิตอล การสแกนจากสแกนเนอร์ หรือได้จากการประมวลผล เป็นต้น รูปภาพดิจิตอลจะใช้เมตริกซ์ (Matrix) ในการเก็บข้อมูล โดยแต่ละช่องของเมตริกซ์นั้นจะเรียกว่าพิกเซล (Pixel) โดยรูปภาพแต่ละชนิดนั้นจะมีรูปแบบการเก็บข้อมูลที่แตกต่างกันไป เช่น รูปภาพขาว - ดำ และรูปภาพระดับเทา จะใช้เมตริกซ์ 2 มิติในการเก็บค่าสี ส่วนรูปภาพสีจะใช้เมตริกซ์ 2 มิติจำนวน 3 แผ่นในการเก็บค่าสี เป็นต้น

2.1.1 พิกเซล



รูปที่ 2.1 ลักษณะของตำแหน่งพิกเซล

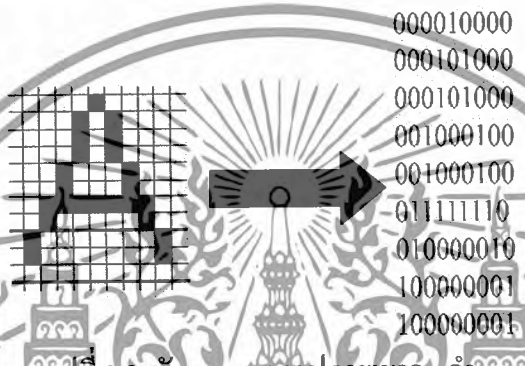
ในภาพหนึ่งๆ เราสามารถอธิบายได้ด้วยเมทริกซ์ของจุดพิกเซลขนาด $N \times M$ โดยใช้คู่ลำดับ p(i,j) แทนค่าของจุดแต่ละจุด โดย i มีค่าตั้งแต่ 0 ถึง N-1 และ j มีค่าตั้งแต่ 0 ถึง M-1 และ p(i,j) จะเป็นค่าบ่งชี้ความเข้มของแสงที่จุดนั้นๆ ของภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าที่กำกับแต่ละพิกเซลจะแสดงถึง ค่าเฉลี่ยของความเข้มแสงในภาพที่จุดพิกเซลนั้นแทนอยู่ โดยค่าของพิกเซลนี้จะเขียนแทนด้วย P_{ij} ซึ่งมีค่าตั้งแต่ 0 ถึง 1

2.1.2 รูปภาพขาว - ดำ

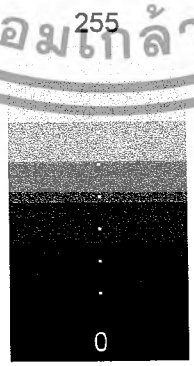
รูปภาพที่ประกอบด้วยสีขาวและสีดำเท่านั้น ซึ่งโดยส่วนใหญ่คอมพิวเตอร์จะสามารถวิเคราะห์ข้อมูลจากรูปภาพชนิดนี้ได้ง่ายกว่ารูปภาพชนิดอื่นๆ เนื่องจากรูปภาพขาว - ดำจะมีค่าข้อมูลเพียง 2 ค่าเท่านั้น โดยจะแทนสีดำด้วยค่า 0 และสีขาวด้วยค่า 1 หรือ แทนสีดำด้วยค่า 0 และสีขาวด้วยค่า 255



รูปที่ 2.2 ลักษณะของรูปภาพขาว - ดำ

2.1.3 รูปภาพระดับสีเทา

คือ รูปที่แต่ละพิกเซลจะเก็บข้อมูลของความเข้มแสงเป็นข้อมูล 8 บิต (Bit) ทำให้สามารถมีความเข้มแสงได้ถึง 256 ระดับสำหรับในแต่ละพิกเซล ซึ่งจะแทนด้วยเลขจำนวนเต็มตั้งแต่ 0 - 255 โดยจะแทนสีดำด้วยเลข 0 และแทนสีขาวด้วยเลข 255 ซึ่งค่าระหว่าง 0 - 255 จะเป็นความเข้มแสงในโทนสีเทา ดังแสดงในรูปที่ 2.3

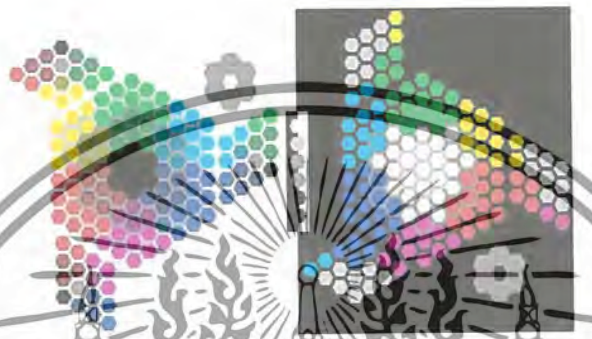


รูปที่ 2.3 ระดับความเข้มแสงของภาพระดับสีเทา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4 รูปภาพสี

รูปภาพสีจะมีอยู่หลายรูปแบบ เช่น รูปแบบ RGB (RGB Color Model) รูปแบบ HSI (HSI Color Model) หรือรูปแบบ CMY (CMY Color Model) เป็นต้น ซึ่งรูปภาพสีแต่ละรูปแบบจะมีความเหมาะสมกับลักษณะงานที่แตกต่างกัน เช่น รูปภาพสีแบบ RGB เหมาะสำหรับแสดงผลบนจอภาพคอมพิวเตอร์ หรือกล้องดิจิตอล ส่วนรูปภาพสีแบบ CMY เหมาะจะใช้ในงานที่เกี่ยวกับเครื่องพิมพ์ เป็นต้น



รูปที่ 2.4 ลักษณะของรูปภาพสี

2.1.5 ระบบโมเดลสี HSL (HSL Color Model)

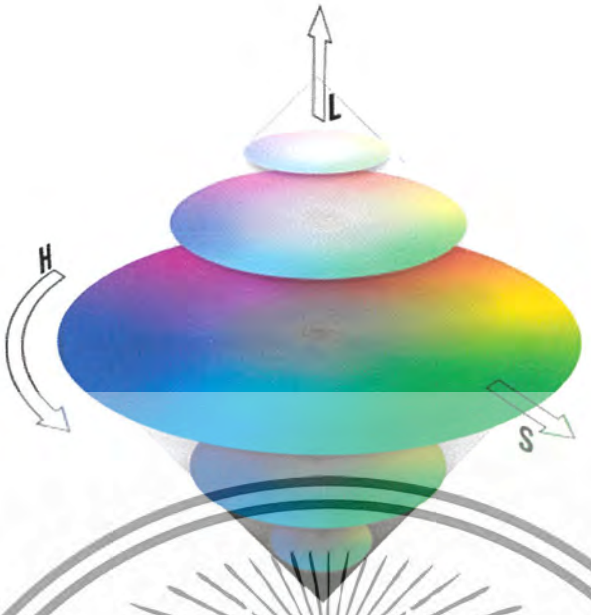
โมเดลสี HSL ที่ประกอบด้วยค่า Hue, Saturation และค่า Lightness โดยที่ค่า Hue จะเป็นตัวกำหนดค่าสี, Saturation กำหนดความบริสุทธิ์ของสี และ Lightness กำหนดความสว่างของสี

Hue ใช้ในการกำหนดสีของวัตถุ โดยค่า Hue จะมีค่าตั้งแต่ 0° ถึง 360° ซึ่ง 0° คือ สีแดง 120° คือ สีน้ำเงิน และ 240° คือ สีเขียว

Saturation ใช้ในการกำหนดความบริสุทธิ์ของสี ซึ่งช่วงความบริสุทธิ์ของสีจะถูกกำหนดเป็นค่าเปอร์เซ็นต์ โดย 100% คือ สีมีความเข้มมากที่สุด และ 0% คือ สีที่ไม่มีความเข้ม ซึ่งภาพขาว - ดำ จะมีค่า Saturation เท่ากับ 0%

Lightness ใช้ในการอ้างถึงปริมาณของสีขาว หรือสีดำที่อยู่ในภาพ โดยจะถูกกำหนดเป็นเปอร์เซ็นต์ 100% คือ ภาพสีขาวล้วน 0% คือ ภาพสีดำล้วน และ 50% คือ ภาพที่มีแต่ค่า Hue

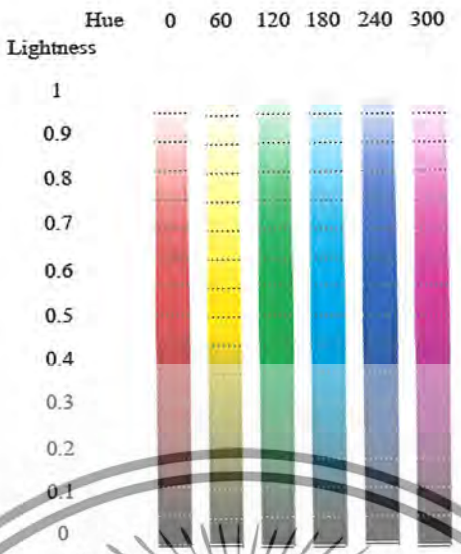
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5 แบบจำลองโมเดลสี HSL

รูปที่ 2.6 ความสัมพันธ์ระหว่างค่าประกอบ Hue และ Saturation โดยที่ค่า Lightness เป็น 0.5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.7 ความสัมพันธ์ระหว่างค่า Hue และ Lightness โดยที่ค่า Saturation เป็น 1

2.2 ไฟล์ข้อมูลภาพชนิดบิตแมป

2.2.1 รูปแบบของไฟล์ข้อมูลภาพชนิดบิตแมป

รูปแบบของไฟล์ข้อมูลชนิดบิตแมปเป็นฟอร์แมตของวินโดวส์บิตแมปซึ่งเป็นมาตรฐานสำหรับไฟล์กราฟิกบนวินโดวส์ โดยจะใช้ในการตัดต่อหรือสำเนาภาพต่างๆ ลงบนคลิปบอร์ด (Clipboard) เมื่อเวลาจัดเก็บไฟล์ที่มีสกุล BMP

2.2.2 โครงสร้างของไฟล์ของข้อมูลภาพชนิดบิตแมป

โครงสร้างของไฟล์ข้อมูลภาพชนิดบิตแมป จะประกอบด้วย 3 ส่วน คือ

2.2.2.1 ข้อมูลเฮดเดอร์ (Header)

ข้อมูลที่อยู่บริเวณส่วนหัวของไฟล์ ซึ่งประกอบไปด้วยข้อมูลที่บอกรายละเอียดต่างๆ ของภาพ เช่น ความกว้าง, ความยาวภาพ, จำนวนสี, จำนวนบิต และความละเอียด เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2.2 ข้อมูลจานสี (Palette)

ข้อมูลที่บอกถึงชุดของจานสีที่เกิดจากการผสมแม่สีทั้งสาม คือ สีแดง, สีเขียว และสีน้ำเงินมาผสมกันได้เป็นสีต่างๆ ตามจำนวนสีของภาพ เช่น รูปขนาด 4 บิต จะมี 16 ระดับสี, รูปขนาด 8 บิต จะมีขนาด 256 สี เป็นต้น ซึ่งถ้ามีจำนวนสีน้อยๆ ก็จะมีการเก็บค่าจานสีนี้ลงไฟล์ด้วย แต่ถ้ารูปประเภท 24 บิต จะไม่มีค่าจานสีแต่จะใช้วิธีเก็บค่าแม่สีทั้งสามลงไปเป็นข้อมูลแทน เพราะถ้าเก็บค่าจานสีที่มีถึง 16.7 ล้านสีลงไปด้วยจะเปลืองพื้นที่มาก ข้อแตกต่างที่สำคัญของภาพบิตแมปขนาดนี้ คือ ไฟล์ภาพบิตแมปจะเก็บจานสีชุดละ 4 ไบต์ (Byte) แต่ก็ใช้แค่ 3 ไบต์ เช่นกัน คือ แดง เขียว และน้ำเงิน

2.2.2.3 ข้อมูลภาพ (Data)

ข้อมูลสีของภาพแต่ละจุดที่มาประกอบกันเป็นรูปภาพ ซึ่งค่าที่เก็บนี้จะเป็นค่าที่ใช้ในการชี้ตารางจานสีว่ามีหมายเลขอะไร เช่น จุดแรกมีค่าเป็น 10 ก็ให้ไปเปิดตารางจานสี หมายเลข 10 โดยสมมุติว่าของแม่สีเป็น R = 0, G = 0 และ B = 100 ก็จะได้จุดนี้เป็นสีน้ำเงิน ซึ่งถ้าเป็นกรณีของรูป 24 บิต จะเป็นการอ่านข้อมูลขึ้นมา 3 ค่า เป็นค่าของแม่สี RGB แล้วนำไปผสมบนจอภาพแทน

2.2.3 การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป

การจัดเก็บไฟล์ข้อมูลชนิดบิตแมป มีการเก็บอยู่ 2 แบบ คือ

2.2.3.1 แบบบีบอัดข้อมูล

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

2.2.3.2 แบบไม่ได้บีบอัดข้อมูล

เป็นการเก็บข้อมูลจริงของสีของพิกเซล ซึ่งทำให้ขนาดของไฟล์ค่อนข้างใหญ่แต่จะทำการแสดงภาพได้เร็วกว่า เพราะไม่ต้องเสียเวลาในการคลายข้อมูล

2.2.4 เฮดเดอร์ไฟล์ข้อมูลชนิดบิตแมป

ในการอ่านและเขียนไฟล์รูปภาพบิตแมปนั้นจำเป็นต้องรู้ตำแหน่งของข้อมูลต่างๆทั้งภายในเฮดเดอร์ ดังตารางที่ 2.1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 ข้อมูลต่างๆ ในส่วนของเฮดเดอร์ของไฟล์ข้อมูลชนิดบิตแมป

Field Name	Size in Bytes	Description
bfType	2	เป็นตัวบอกว่าเป็น file BMP มีค่าเท่ากับ "BM"
bfSize	4	ขนาดของรูปภาพ
bfReserved1	2	Unused - must be zero
bfReserved2	2	Unused - must be zero
bfOffBits	4	ตำแหน่งเริ่มต้นการอ่าน
biSize	4	ขนาดของ header
biWidth	4	ความกว้างของรูป
biHeight	4	ความสูงของรูป
biPlanes	2	Must be 1
biBitCount	2	จำนวน bit ต่อ 1 ตำแหน่ง - 1, 2, 4, 8, 16, 24, หรือ 32
iCompression	4	รูปแบบการบีบอัด (0 = ไม่มีการบีบอัด)
biSizeImage	4	ขนาดของรูปที่มีการบีบอัด ถ้าไม่มีเป็น 0
biXPelsPerMeter	4	X Preferred resolution in pixels per meter
biYPelsPerMeter	4	Y Preferred resolution in pixels per meter
biClrUsed	4	Number Color Map entries that are actually used
biClrImportant	4	หมายเลขของสีที่สำคัญ

2.3 สัญญาณรบกวนแบบซอลท์และเปปเปอร์ (Salt and Pepper Noise)

ซอลท์ คือ สัญญาณรบกวนที่ทำให้ภาพเกิดพิกเซลสีขาวกระจายอยู่ทั่วทั้งภาพ

เปปเปอร์ คือ สัญญาณรบกวนที่ทำให้ภาพเกิดพิกเซลสีดำกระจายอยู่ทั่วทั้งภาพ

โดยสัญญาณรบกวนทั้งแบบซอลท์และเปปเปอร์สามารถถูกทำให้ลดลงได้ด้วยการใช้วิธีลดสัญญาณรบกวนโดยใช้การกรองแบบค่ามัธยฐาน (Median Filter)

รูปที่ 2.8 ภาพที่มีสัญญาณรบกวนแบบซอลท์และเปปเปอร์

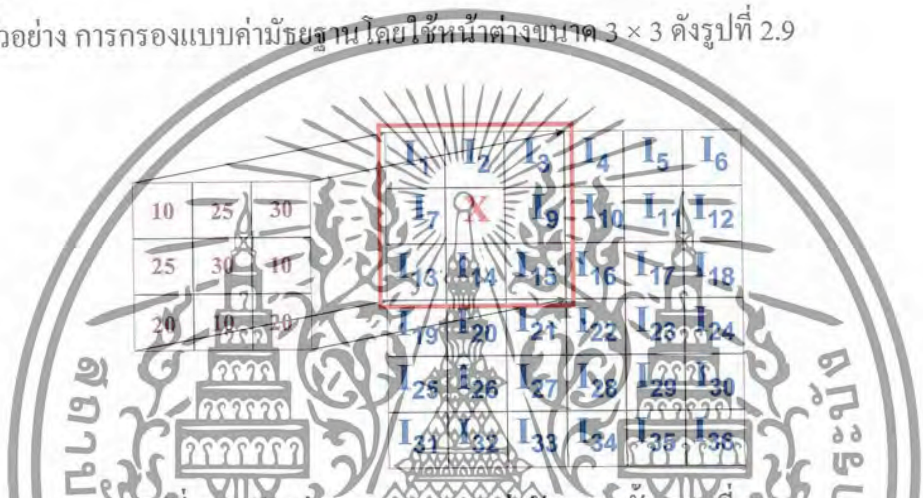
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 การลดสัญญาณรบกวน (Noise) โดยใช้การกรองแบบค่ามัธยฐาน

การกรองข้อมูลภาพวิธีนี้จะใช้หน้าต่าง (Window) ในการกำหนดขอบเขตของการพิจารณาเพื่อหาค่าความเข้มแสงของพิกเซลที่อยู่ตรงกึ่งกลางของหน้าต่าง โดยการนำค่าความเข้มแสงของพิกเซลที่อยู่ในหน้าต่างมาเรียงลำดับจากน้อยไปหามาก จากนั้นจะเลือกค่าที่อยู่ตรงกลางไปใช้ โดยที่หน้าต่างจะถูกเลื่อนไปยังตำแหน่งต่างๆ ในภาพจนครบทุกจุด วิธีการนี้จะต้องใช้การเรียงลำดับซึ่งเป็นกระบวนการที่ใช้เวลาในการคำนวณสูง แต่ข้อดีคือภาพจะไม่สูญเสียความคมชัด

โดยขนาดของหน้าต่างที่นิยมใช้กัน คือ 3 × 3, 5 × 5 หรือ 7 × 7 ถ้าขนาดของหน้าต่างมีขนาดใหญ่ขึ้นรายละเอียดของภาพก็จะลดลง

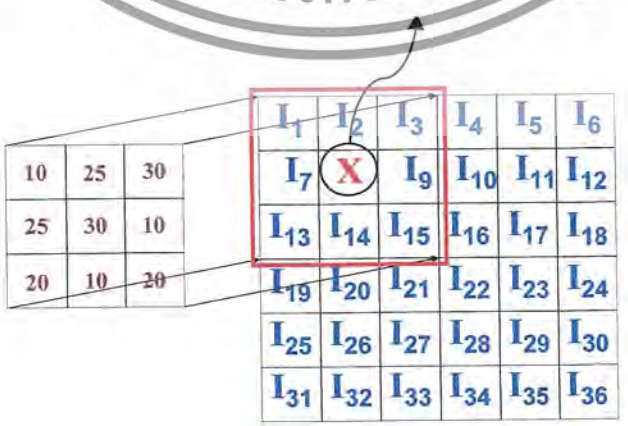
ตัวอย่าง การกรองแบบค่ามัธยฐานโดยใช้หน้าต่างขนาด 3 × 3 ดังรูปที่ 2.9



รูปที่ 2.9 ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 1

นำค่าความเข้มแสงมาเรียงลำดับรูปที่ 2.10 ได้ ดังนี้ 10, 10, 10, 20, 20, 25, 25, 30, 30 ค่าความเข้มแสงที่อยู่กึ่งกลาง คือ 20 เพราะฉะนั้นพิกเซลที่ถูกพิจารณาจะมีค่าความเข้มแสงใหม่เป็น 20 ดัง

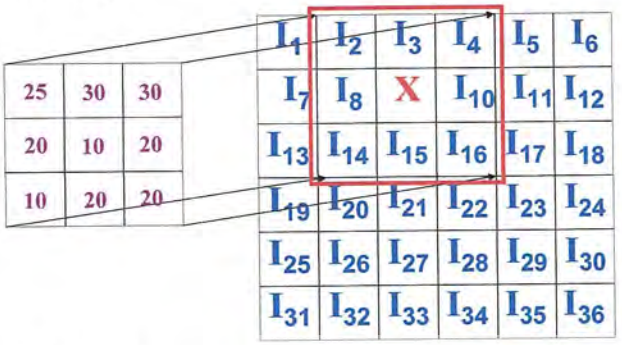
จะมีค่าความเข้มแสงเท่ากับ 20



รูปที่ 2.10 ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาในพิกเซลถัดไป



รูปที่ 2.11 ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 3

ทำจนครบทุกพิกเซล

รูปที่ 2.12 ตัวอย่างการกรองแบบค่ามัธยฐานขั้นตอนที่ 4

รูปที่ 2.13 การเปลี่ยนแปลงของภาพหลังจากการกรองแบบค่ามัธยฐาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5 การแปลงภาพสีเป็นรูปภาพขาว - ดำ ด้วยวิธีเทรชโฮล

ในการประมวลผลภาพที่ต้องการความเข้มพียง 2 ระดับ หรือ 1 บิต คือ 0 กับ 1 โดย 0 แทนจุดที่เป็นสีดำ และ 1 แทนจุดที่เป็นสีขาว แต่ภาพที่รับเข้ามานั้นมีความเข้มหลายระดับ เช่น ภาพ 16 ล้านสี จึงต้องแปลงข้อมูลภาพให้เป็นรูปภาพขาว - ดำ

การทำรูปภาพขาว - ดำทำได้โดยใช้เทคนิคเทรชโฮล คือ การพิจารณาจุดพิกเซลในภาพว่าจุดใดควรจะเป็นจุดที่มีสีขาว (ให้มีค่าเท่ากับ 1) หรือจุดใดควรจะเป็นสีดำ (ให้มีค่าเท่ากับ 0) โดยจะทำการเปรียบเทียบค่าของแต่ละพิกเซล pixel (f(x,y)) กับค่าคงที่เรียกว่า ค่าเทรชโฮล (Threshold Value) โดยการทำเทรชโฮลให้ได้ภาพที่ดี และคมชัดต้องเกิดจากการเลือกค่าเทรชโฮลที่ถูกต้องและเหมาะสม ถ้าเลือกค่าเทรชโฮลที่ไม่เหมาะสม เช่น ค่าเทรชโฮลที่มากเกินไป หรือน้อยเกินไป ภาพที่ได้จะขาดความคมชัดหรืออาจจะทำให้รายละเอียดของภาพขาดหายไป หรือภาพที่ได้อาจจะมืดเกินไป หรือสว่างเกินไป หรืออาจจะเป็นภาพที่มีสิ่งรบกวนเกิดขึ้นทำให้ภาพผลลัพธ์ที่ได้ไม่ชัดเจนเท่าที่ควร ดังนั้นปัญหาในการสร้างรูปภาพขาว - ดำ คือ ทำอย่างไรจึงจะคำนวณหาค่าเทรชโฮลที่เหมาะสมกับภาพแต่ละภาพที่จะนำมาสร้างเป็นรูปภาพขาว - ดำ ซึ่งค่าเทรชโฮลสามารถคำนวณได้หลายวิธี แต่ละวิธีจะเหมาะกับการทำงานที่แตกต่างกันไป เช่น การหาค่าเทรชโฮลโดยการกำหนดล่วงหน้า (Pre - Assigned Threshold Value) การหาค่าเทรชโฮลจากค่ากลาง (Mid - Range Threshold Value)

2.5.1 การหาค่าเทรชโฮลโดยการกำหนดล่วงหน้า

การหาค่าเทรชโฮลโดยการกำหนดล่วงหน้าเป็นวิธีที่ง่ายที่สุดเป็นการคำนวณค่าด้วยการกำหนดค่าเองของผู้ใช้งาน ซึ่งการกำหนดค่านี้จะขึ้นอยู่กับประสบการณ์ของผู้ใช้ โดยการเลือกค่าคงที่ที่เรียกว่าค่าเทรชโฮลนั้น ค่าที่เลือกมานี้จะเป็นค่าที่อยู่ระหว่างค่าสูงสุด และค่าต่ำสุดของข้อมูลอินพุท (Input Data) แต่ละพิเซลของภาพ เช่น ภาพข้อมูลอินพุทมีระดับสีเทา 256 ระดับจะมีค่าทั้งหมด 0 - 255 เมื่อเลือกค่าเทรชโฮลแล้วก็สามารถนำค่าเทรชโฮลเป็นตัวกำหนดในการสร้างรูปภาพขาว - ดำได้

2.5.2 การหาค่าเทรชโฮลจากค่ากลาง

การหาค่าเทรชโฮลโดยพิจารณาจากค่ากลางเป็นการหาค่าเทรชโฮลที่แตกต่างจากการหาค่าเทรชโฮลโดยการกำหนดล่วงหน้า สำหรับวิธีนี้จะเป็นการคำนวณโดยอัตโนมัติโดยไม่ต้องให้ผู้ใช้งานเป็นผู้กำหนด ซึ่งการหาค่าเทรชโฮลวิธีนี้อาศัยหลักการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลางหรือค่าเฉลี่ย (Mean) มาประยุกต์ใช้ ค่าเทรชโฮลที่คำนวณได้จากค่ากึ่งกลางที่อยู่ระหว่างค่าที่มีความเข้มสูงสุด (Maximum Level) และระดับความเข้มต่ำสุด (Minimum Level) ของข้อมูลอินพุทเมื่อทำการคำนวณค่าเทรชโฮลได้แล้วก็จะสามารถนำค่าเทรชโฮลเป็นตัวกำหนดในการสร้างรูปภาพขาว - ดำได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 การหาจุดศูนย์กลางมวลของวัตถุ

จุดศูนย์กลางมวลเป็นจุดที่เสมือนเป็นที่รวมมวลของวัตถุทั้งก้อนนั้น โดยที่จุดศูนย์กลางมวลอาจอยู่นอกเนื้อวัตถุนั้นได้ เช่น รูปโดนัท

โดยปกติวัตถุบางชนิดมีมวลภายในหนาแน่นไม่เท่ากันตลอดทั้งเนื้อสาร จุดศูนย์กลางมวลจึงเป็นเสมือนที่เป็นจุดรวมมวลของวัตถุทั้งก้อน ถ้าหากใช้แรงกระทำต่อวัตถุโดยให้แนวตรงผ่านจุดศูนย์กลางมวลจะทำให้วัตถุนั้นไม่หมุน แต่ถ้าหากแรงที่กระทำไม่ผ่านจุดศูนย์กลางมวลวัตถุจะหมุนในทิศทางตามแรงนั้น

รูปที่ 2.14 จุดศูนย์กลางมวลของวัตถุ

2.7 การทำนอร์มัลไลเซชั่น

การทำนอร์มัลไลเซชั่น คือ กระบวนการที่เปลี่ยนช่วงของค่าความหนาแน่นของพิกเซลที่อยู่ในภาพ ให้อยู่ในลักษณะที่สะดวกต่อการนำไปใช้

ตัวอย่างเช่น ต้องการเปลี่ยนภาพที่มีช่วงของความหนาแน่นเท่ากับ 50 ถึง 180 เป็น 0 ถึง 255 โดยเริ่มแรก จะทำการลบค่าความหนาแน่นของแต่ละพิกเซลออก 50 และกำหนดช่วงใหม่เท่ากับ 0 ถึง 130 จากนั้นคูณค่าความหนาแน่นของแต่ละพิกเซลด้วยค่า 255/130 เพื่อทำให้อยู่ในช่วง 0 ถึง 255

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8 การลบกันของภาพพื้นหลัง

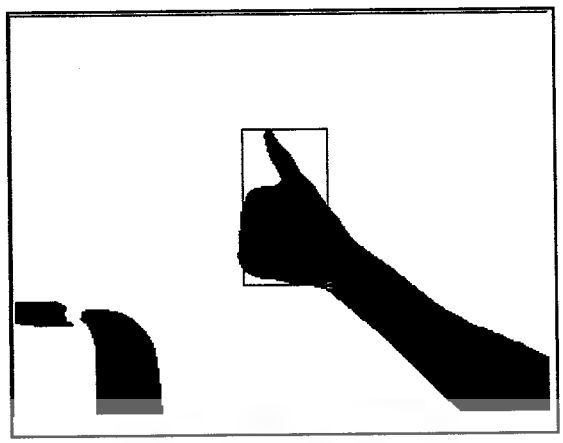
เป็นเทคนิคในการตรวจจับวัตถุในภาพ โดยมีหลักการ คือ นำภาพปัจจุบันและภาพพื้นหลัง (Background) มาลบกันเพื่อหาค่าความแตกต่างกัน โดยค่าความแตกต่างที่ได้แสดงถึงวัตถุที่เข้ามาในภาพ แต่อาจมีปัญหาเกิดขึ้นจากการที่ฉากหลังของภาพปัจจุบันไม่นิ่งทำให้เกิดสัญญาณรบกวนขึ้น จึงต้องกำหนดช่วงค่าความแตกต่างที่ยอมรับได้ไว้เพื่อลดความผิดพลาดที่อาจเกิดขึ้น โดยผลลัพธ์ที่ได้ คือ รูปภาพขาว - ดำของวัตถุที่เข้ามาในกล้อง

วิธีนี้สามารถใช้ได้ดีกับภาพที่มีฉากหลังคงที่ ในกรณีฉากหลังมีการเปลี่ยนแปลงอยู่บ้างสามารถแก้ไขได้ โดยการอัพเดท (Update) ภาพฉากหลังเป็นช่วงๆ เพื่อให้ภาพฉากหลังใกล้เคียงกับฉากหลังของภาพปัจจุบันมากที่สุด

รูปที่ 2.15 ตัวอย่างภาพพื้นหลัง

รูปที่ 2.16 ตัวอย่างภาพปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.17 ตัวอย่างค่าความแตกต่างระหว่างภาพพื้นหลังและภาพปัจจุบัน

2.9 การหาขอบภาพ

การหาขอบภาพเป็นการหาเส้นรอบวัตถุที่อยู่ในภาพ เมื่อทราบเส้นรอบวัตถุจะสามารถคำนวณหาพื้นที่ (ขนาด) หรือรู้จำนิดของวัตถุนั้นได้ การหาขอบภาพที่ถูกต้องนั้นจะขึ้นอยู่กับความชัดเจนสมบูรณ์ของภาพที่ใช้

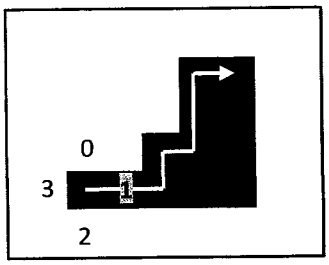
เทคนิคการหาขอบภาพสามารถทำได้หลายวิธีแต่ในที่นี้จะใช้วิธีการไล่ตามขอบภาพ (Edge Following) ซึ่งเป็นวิธีที่ง่ายที่สุด และสามารถใช้กับรูปภาพขาว - ดำได้ง่าย โดยหลักการ คือ หากทราบจุดใดจุดหนึ่งบนขอบภาพจะสามารถหาจุดข้างเคียงที่เป็นขอบภาพ และสามารถวนไปตามขอบภาพจนกลับมายังจุดเริ่มต้นได้ การตามขอบภาพมีวิธีดังต่อไปนี้

1) สมมุติให้จุด (x, y) เป็นจุดใดจุดหนึ่งบนขอบภาพ
2) ตั้งค่าแฟลก (Flag) ให้จุด (x, y) ว่าเคยผ่านมาแล้ว
3) หาค่าพิกเซลของจุดสี่จุดที่อยู่รอบจุด (x, y) โดยกำหนดให้มีทิศทางดังต่อไปนี้

0
3 x 1
2

4) กำหนดว่าจะเลือกตามขอบภาพในทิศทางไหนคือตามเข็มนาฬิกาหรือทวนเข็มนาฬิกา
5) ในกรณีที่เลือกตามเข็มนาฬิกาให้พิจารณาจุดในทิศทาง d – 1 เป็นอันดับแรกแล้วจึงไล่พิจารณาพิกเซลในทิศทางตามเข็มนาฬิกาต่อมา คือ ทิศทาง d และ d + 1 ตามลำดับ โดยจุดที่มีค่าพิกเซลสีดำเป็นจุดแรกคือจุดที่เป็นขอบภาพ ให้จุดนั้นเป็นจุดที่จะพิจารณาถัดไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.18 การทำไล่ตามขอบภาพ

6) ในกรณีที่เลือกตามขอบภาพในทิศทางทวนเข็มนาฬิกาให้พิจารณาจากทิศทาง d + 1 ทิศทาง d และทิศทาง d – 1 ตามลำดับ

7) ทำซ้ำข้อ 5 หรือ 6 ขึ้นอยู่กับว่าตามขอบภาพในทิศทางไหน

8) หยุดเมื่อวนขอบกลับมายังจุดเดิม หรือใช้เวลาในการตามขอบภาพนานเกินไป

วิธีการตามขอบภาพแบบสี่ทิศทางสามารถใช้ได้กับรูปภาพขาว - ดำเท่านั้น ในกรณีที่ต้องการใช้กับภาพระดับสีเทา สามารถทำได้โดยเพิ่มเป็นแปดทิศทาง และเลือกจุดที่มีความเข้มสูงสุดในแปดทิศทางนี้ หรืออาจเลือกใช้วิธีอื่น เช่น วิธีของโซเบล (Sobel Edge Detection) เป็นต้น

2.10 การทำให้เส้นบาง

การทำให้เส้นบางเป็นการลดขนาดความหนาของรูปให้ลงเหลือเส้นเดียว เมื่อนำมาประยุกต์กับภาพสัญญาณมือจะทำให้ง่ายต่อการวิเคราะห์โครงสร้างของมือ โดยเราสามารถเขียนให้อยู่ในรูปของการเปลี่ยนรูปฮิทมิส (Hit or Miss Transform) ดังสมการต่อไปนี้

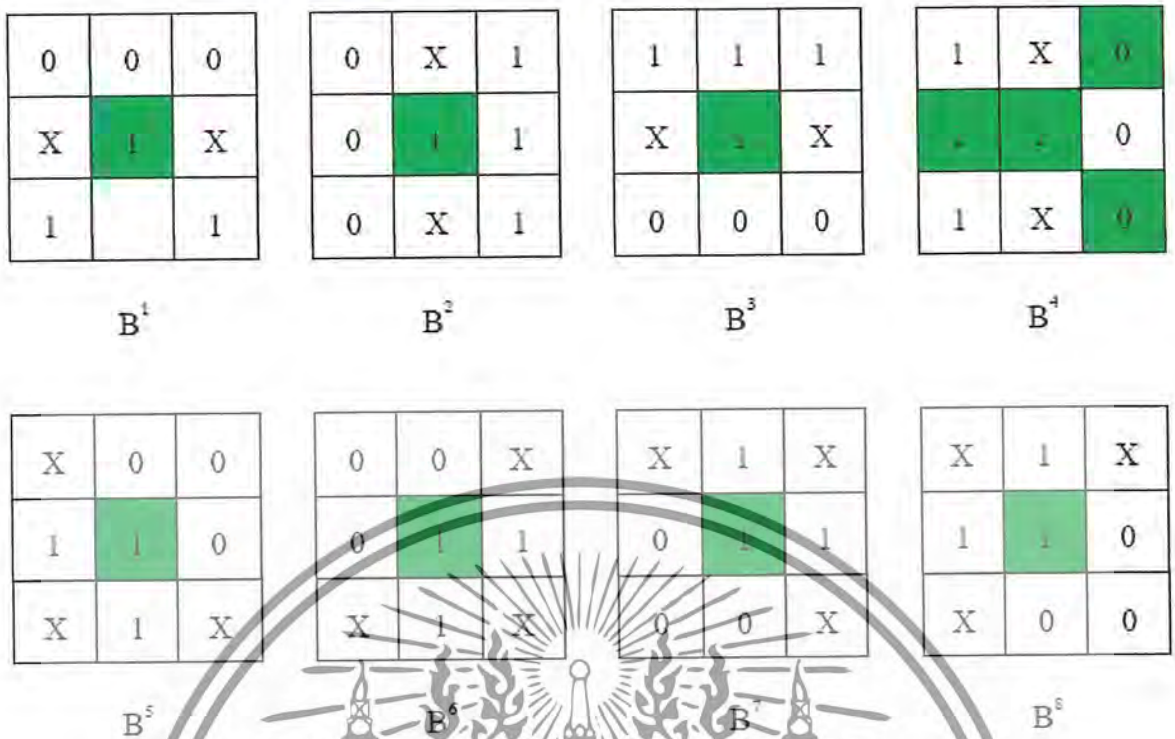
$$A \otimes B = A - \text{HitMiss}(A, B) \tag{2.1}$$

การนำไปใช้จริงของการทำให้เส้นบางของ A ในวิธีนี้ใช้ B ในรูปของลำดับของหน่วยโครงสร้างดังนี้ $\{B\} = \{B^1, B^2, B^3, \ldots, B^n\}$ ดังรูปที่ 2.15 ดังนั้นสมการการทำให้เส้นบางถูกเขียนใหม่ในรูปของ {B} ดังสมการต่อไปนี้

$$A \otimes B((\ldots((A \otimes B^1) \otimes B^2)\ldots) \otimes B^n) \tag{2.2}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 2.19 รูปแบบของ B^1, B^2, B^3, B^4, B^5, B^6, B^7 และ B^8

กระบวนการนี้ภาพ A จะถูกกระทำผ่าน B^1 จากนั้นผลที่ได้จะกระทำผ่าน B^2 ผลที่ได้จะถูกกระทำอย่างนี้ไปเป็นลำดับของ {B} จนกระทั่งถึง B^n กระบวนการทั้งหมดนี้จะถูกทำซ้ำจนกระทั่งผลที่ได้ไม่เกิดการเปลี่ยนแปลง

2.11 การรู้จำ

การรู้จำภาพเป็นส่วนหนึ่งของการรู้จำแบบรูป (Pattern Recognition) ที่มีการศึกษาอย่างแพร่หลาย คำว่าแบบรูป (Pattern) สามารถแปลความได้หลายแบบ ในที่นี้แบบรูปหมายถึงรูปร่างของมือในลักษณะต่างๆ ในการรู้จำภาพจะต้องรู้จำแบบรูปของแต่ละภาพเพื่อแยกแยะภาพที่ต่างกันออกจากกัน แบบรูปที่ดีต้องบอกถึงลักษณะเด่นของภาพ ซึ่งอาจได้จากการรูปร่างของมือ หรือได้จากคำนวณ เช่น การใช้จุดศูนย์กลางมวลของภาพ หรือตำแหน่งของปลายนิ้วมือ

2.11.1 การรู้จำแบบเข้าคู่รูปแบบ (Template Matching)

เป็นเทคนิคที่ใช้ในการประมวลผลภาพดิจิตอลสำหรับการหาส่วนที่เหมือนกันระหว่างภาพที่รับเข้ามากับภาพที่เป็นต้นแบบ

2.11.2 การรู้จำแบบวิเคราะห์ทางโครงสร้าง (Structural Analysis)

การวิเคราะห์ทางโครงสร้าง คือ การวิเคราะห์องค์ประกอบต่างๆ ในภาพ โดยหาลักษณะสำคัญของวัตถุที่เราสนใจ เช่น มุม ส่วนโค้ง เป็นต้น แล้วนำข้อมูลของลักษณะสำคัญเหล่านี้มา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
103016

ประกอบกันเพื่อวิเคราะห์เพื่อหาความหมายของภาพ วิธีการนี้มีข้อดีที่มีความยืดหยุ่นหลากหลาย แต่ทั้งนี้อัตราความถูกต้องจะขึ้นอยู่กับการสร้างกฎ และวิเคราะห์กฎที่ใช้ในการหาลักษณะสำคัญ ซึ่งเป็นส่วนที่สำคัญที่สุดของวิธีการนี้

ในกรณีของภาพมือลักษณะสำคัญที่ใช้วิเคราะห์ คือ จุดปลายนิ้วมือ และจุดศูนย์กลางมวล เมื่อนำองค์ประกอบทั้งสองมาวิเคราะห์ร่วมกัน สามารถแยกแยะสัญญาณมือในแต่ละแบบได้

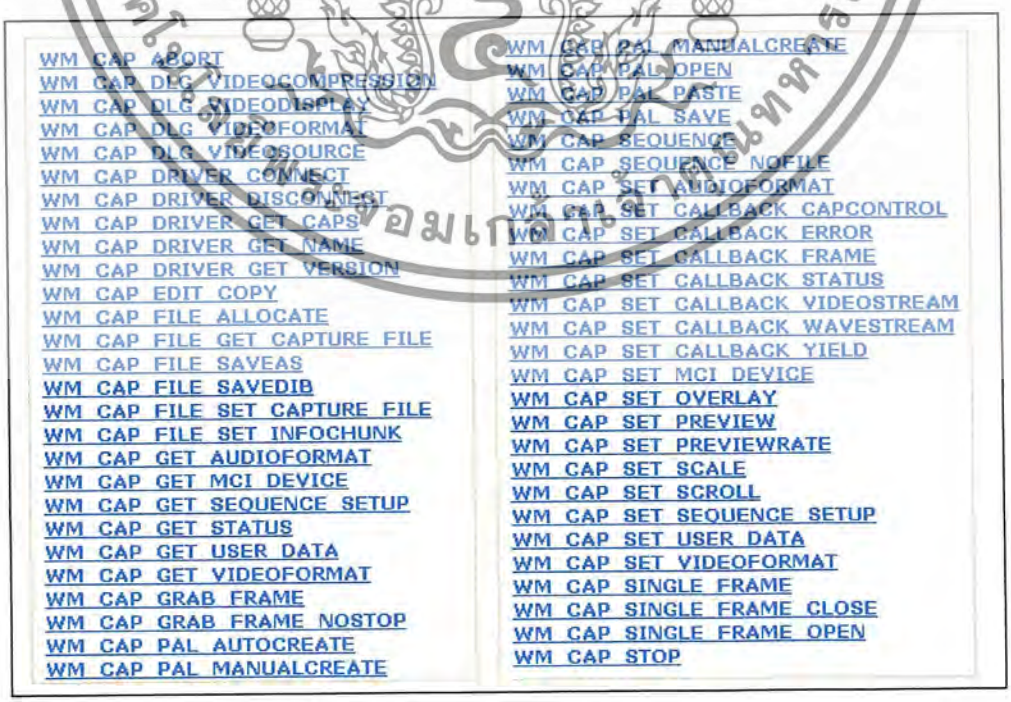
2.12 การติดต่อกล้องโดยใช้เอวีไอแคป 32

Audio Video Interleave หรือ AVI เป็นรูปแบบหนึ่งที่ใช้ในการบันทึกภาพมัลติมีเดียถูกคิดค้นขึ้นโดยบริษัทไมโครซอฟต์ในปี 1992 โดยเป็นส่วนหนึ่งของเทคโนโลยีที่ใช้สำหรับวิดีโอในระบบปฏิบัติการวินโดวส์

การใช้คลาสเอวีไอแคปวินโดวส์ (AVICap Windows Class) สามารถใช้งานได้บนระบบปฏิบัติการวินโดวส์ ซึ่งเราสามารถจับภาพวิดีโอได้อย่างง่ายดาย คลาสเอวีไอแคป (AVICap) จะอยู่ในไฟล์ avicap32.dll ซึ่งภายในประกอบด้วยส่วนต่อประสานโดยใช้แมสเสจ (Message - Based Interfaces) ที่ใช้ในการเข้าถึงอุปกรณ์สำหรับบันทึกวิดีโอและออดิโอ และมีความสามารถในการบันทึกวิดีโอสตรีมลงบนดิสก์

2.12.1 ข้อดีของการใช้เอวีไอแคป 32

สามารถจัดการกับคุณสมบัติที่หลากหลายในการบันทึกภาพ ดังภาพด้านล่างแสดงความสามารถในการทำงานของเอวีไอแคป 32



รูปที่ 2.20 Video Capture Messages

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.12.2 ข้อเสียของการใช้เอวีไอแคป 32

เอวีไอแคป 32 เป็นส่วนเชื่อมต่อในการเขียนแอพพลิเคชั่น (Application Programming Interface หรือ API) โดยเอวีไอแคป 32 นี้ไม่ได้ถูกพัฒนาภายใต้ดอทเน็ต (.NET) ดังนั้นเป็นหน้าที่ของนักพัฒนาโปรแกรมที่จำเป็นต้องใช้ Platform Invoke (P/Invoke) เพื่อทำการเรียกใช้ API เหล่านี้

P/Invoke เป็นกลไกของภาษาดอทเน็ตที่ใช้ในการเรียกฟังก์ชันที่ไม่ได้ถูกดูแลจัดการโดยภาษาดอทเน็ต (Unmanaged Functions) ใน DLL โดยเฉพาะอย่างยิ่งใช้ประโยชน์สำหรับการเรียกฟังก์ชันของวินโดวส์เอพีไอ (Windows API functions) ที่ไม่ได้ถูกเอนแคพซูเลทโดยคลาสของดอทเน็ตเฟรมเวิร์ก (.NET Framework) เช่นเดียวกับฟังก์ชันอื่นๆที่พัฒนาขึ้นโดยนักพัฒนาอิสระ (Third - Party Functions) อื่นๆที่อยู่ใน DLL

2.13 การควบคุมโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์โดยใช้ฟังก์ชัน keybd_event และ mouse_event

ฟังก์ชัน keybd_event และ mouse_event เป็นฟังก์ชันหนึ่งในไลบรารี user32.dll โดยหลักการทำงานจะส่งคำสั่งผ่านวินโดวส์แมสเสจ (Window Message) ออกไปเปรียบเหมือนการกดปุ่มบนแป้นคีย์บอร์ด (Keyboard) หรือเมาส์ (Mouse) แต่โดยปกติแล้วจะส่งคำสั่งไปยังโปรแกรมที่ทำงานอยู่หน้าสุด (Active Window) ดังนั้นโปรแกรมที่ต้องการให้ทำงานจะต้องเป็นโปรแกรมที่ทำงานอยู่หน้าสุดด้วย

ในการใช้ฟังก์ชัน keybd_event และ mouse_event จะต้องรู้รหัสของคีย์บอร์ด (Keyboard Character Codes) และเมาส์ (Mouse Code) เช่น การกดปุ่ม ESC จะส่งรหัสออกไปคือ 0x1B (เลขฐาน 16) เป็นต้น โดยสามารถหารหัสเหล่านี้ได้จาก http://msdn.microsoft.com/enus/library/ms645540(VS.85).aspx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
หลักการทำงานและโครงสร้างของโปรแกรม

3.1 หลักการทำงานโดยรวมของโปรแกรม

ระบบจะรับภาพเป็นเฟรม (Frame) จากกล้องเว็บแคมทุกๆ 0.1 วินาที และปรับขนาดภาพใหม่เป็น 320 × 280 พิกเซล ซึ่งเป็นขนาดที่เหมาะสมต่อการทำงานของโปรแกรม เพราะถ้าขนาดใหญ่เกินไปจะใช้เวลาในการประมวลผลนาน ทำให้ไม่สามารถประมวลผลแบบทันที (Real time) ได้ และถ้าขนาดเล็กเกินไปก็จะไม่สามารถตรวจจับรูปร่างมือได้ และหลังจากนั้นจะถูกนำไปประมวลผลในรูปของไฟล์ข้อมูลชนิดบิตแมป โดยใช้โมเดลสีแบบ RGB 24 บิต (RGB Color Model 24 Bit) จากนั้นจะเข้าสู่กระบวนการของการติดตามวัตถุ (Object Tracking) โดยจะแบ่งออกเป็น 2 กรณีด้วยกันดังนี้

กรณีที่ 1 ติดตามมือของผู้ใช้โดยอาศัยสีของถุงมือ ซึ่งจะทำการแปลงภาพสีที่ได้เป็นรูปภาพขาว - ดำโดยใช้ค่าเทรชโฮลที่ได้มาจากการเรียนรู้จากภาพที่รับเข้า

กรณีที่ 2 ติดตามมือของผู้ใช้โดยทำการลบกันของภาพพื้นหลัง ซึ่งพิกเซลที่มีความแตกต่างกันจะถูกเปลี่ยนเป็นสีดำ

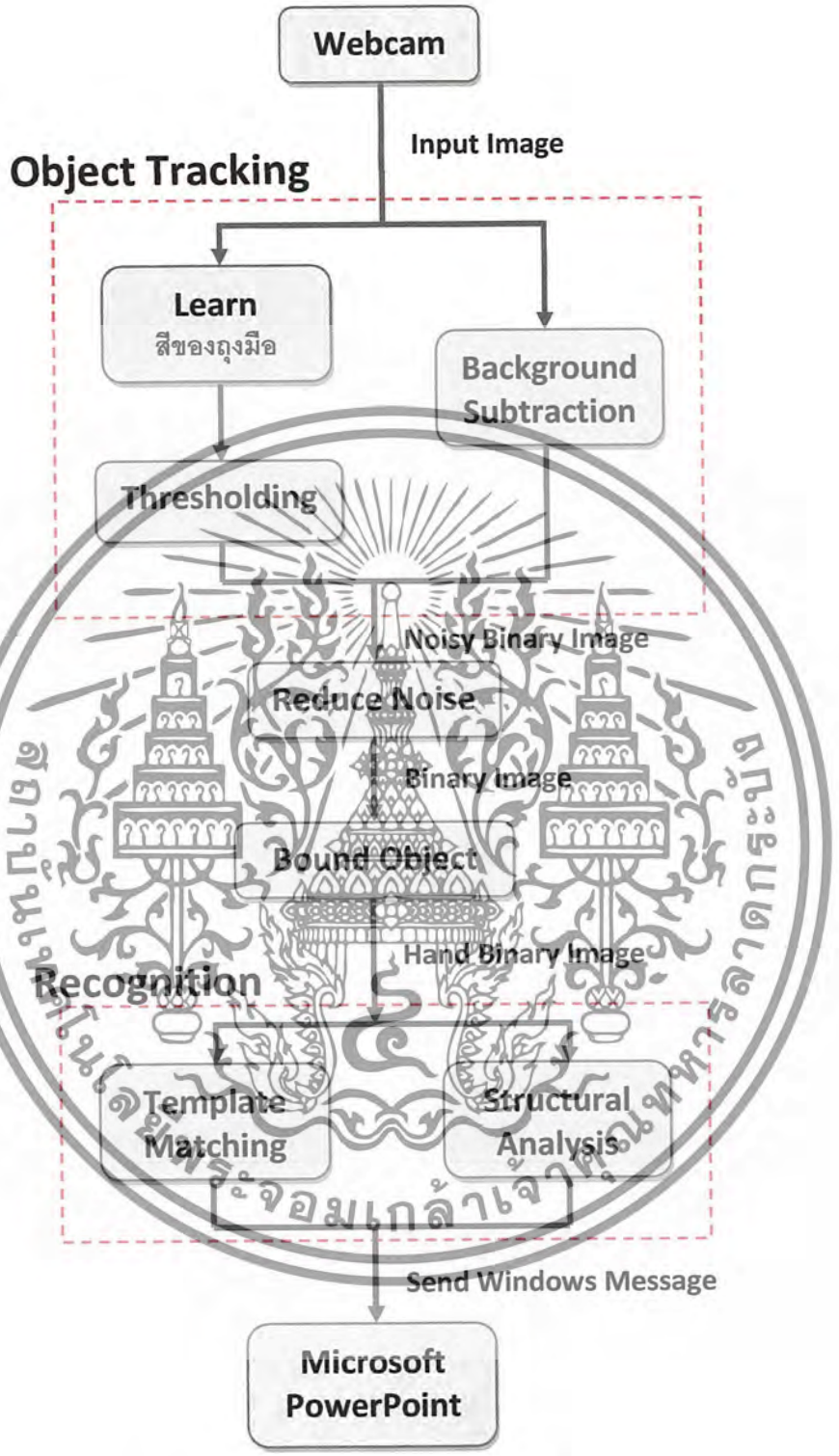
เมื่อเสร็จสิ้นขั้นตอนนี้จะได้รูปภาพขาว - ดำ โดยที่ส่วนของวัตถุที่เราต้องการจะเป็นสีดำ และในส่วนที่เป็นพื้นหลังจะเป็นสีขาว แต่ภาพที่ได้นั้นอาจจะยังมีสัญญาณรบกวนอยู่บ้างจึงต้องทำการลดสัญญาณรบกวนโดยการใช้การกรองแบบค่ามัธยฐาน โดยใช้หน้าต่างขนาด 7 × 7 ซึ่งจะได้ภาพที่ไม่มีสัญญาณรบกวน หรือยังมีอยู่บ้างเล็กน้อย จากนั้นทำการตีกรอบสี่เหลี่ยมเพื่อกำหนดขอบเขตของวัตถุที่เราสนใจ และนำวัตถุที่อยู่ในกรอบนั้นไปทำการรู้จำภาพเพื่อตีความหมายของภาพโดยการรู้จำที่ใช้ในที่นี้มีด้วยกัน 2 วิธีคือ

วิธีที่ 1 วิธีการเข้าคู่รูปแบบ

วิธีที่ 2 วิธีการวิเคราะห์ทางโครงสร้าง

จากนั้นจะทำการส่งคำสั่งผ่านวินโดวส์แมสเสจไปควบคุมการเปลี่ยนหน้าสไลด์ของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.1 หลักการทำงานของโปรแกรมโดยรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การเตรียมข้อมูลภาพก่อนทำการประมวลผล (Preprocessing)

3.2.1 การแปลงภาพสีเป็นรูปภาพขาว - ดำ ด้วยวิธีเทรชโฮล

เป็นการทำให้ข้อมูลภาพมีรูปแบบที่เหมาะสมในการประมวลผลจึงต้องทำการแปลงข้อมูลของภาพให้เป็นรูปภาพขาว - ดำ คือ มีระดับสีเพียง 2 ระดับได้แก่ สีดำที่มีค่าเป็น 0 และสีขาวที่มีค่าเป็น 1 วิธีการแปลงข้อมูลภาพให้เป็นรูปภาพขาว - ดำนั้นทำได้โดยการเลือกค่าเทรชโฮลที่เหมาะสม ในที่นี้ได้กำหนดค่าเทรชโฮลเป็นช่วง ซึ่งข้อมูลที่ไม่ได้อยู่ในช่วงของค่าเทรชโฮลจะถูกปรับเป็นสีขาว แต่ถ้าอยู่ในช่วงของค่าเทรชโฮลจะถูกปรับเป็นสีดำ ซึ่งการเลือกค่าเทรชโฮลให้เหมาะสมสามารถทำได้ 2 วิธี คือ

1) เลือกโดยการระบุลงไปในโปรแกรมก่อนทำงาน

ข้อดี

ไม่เสียเวลาในการคำนวณ

ข้อเสีย

ใช้ได้กับสีที่ต้องการเพียงสีเดียวเท่านั้น และยากต่อการเลือกค่าเทรชโฮลที่เหมาะสมเมื่อทำงานกับภาพหลายภาพ

2) ให้โปรแกรมคำนวณค่าเทรชโฮลที่เหมาะสมโดยดูจากภาพที่รับเข้ามา

ข้อดี

สามารถเรียนรู้สีได้ ทำให้สามารถสวมถุงมือสีใดก็ได้

ข้อเสีย

เพิ่มความยุ่งยากให้กับผู้ใช้ที่ต้องทำการเรียนรู้สีของถุงมือก่อนการใช้งานทุกครั้ง และสภาวะแสงในขณะที่ทำการเรียนรู้ต้องใกล้เคียงกับสภาวะแสงที่ใช้งานจริง

ในที่นี้ได้เลือกใช้แบบที่ 2 ในการหาค่าเทรชโฮล โดยจะใช้ คือ ค่า Hue, ค่า Saturation และค่า Lightness ของโมเดลสี HSL โดยค่าทั้งสามนั้นเป็นค่าเฉลี่ยที่ได้มาจากภาพที่รับเข้ามาเพื่อใช้ในการเรียนรู้สี ภาพที่เป็นรูปภาพขาว - ดำแล้วจะได้เป็นภาพของมือที่มีสีดำบนฉากหลังที่มีสีขาวอย่างชัดเจนเนื่องมาจากการใช้ค่าเทรชโฮลที่เหมาะสม แต่อาจจะมีบางกรณีที่ค่าเทรชโฮลสูงหรือต่ำเกินไป ทำให้เห็นรูปร่างของมือไม่ชัดเจน

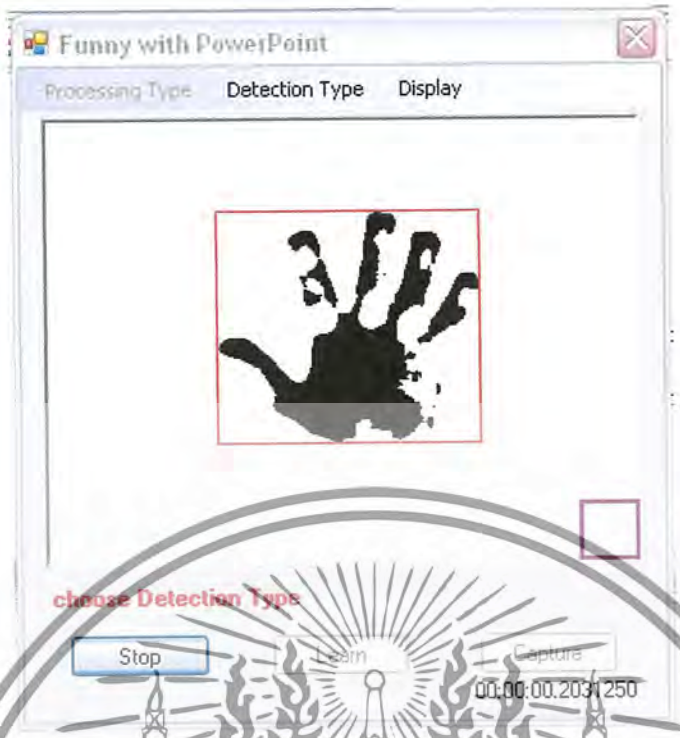
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.2 ภาพที่ถ่ายได้จากกล้องเว็บแคม

หลังจากผ่านการทำเทรชโฮลแล้วภาพที่ได้ออกมาจะเป็นรูปภาพขาว – ดำที่ยังมีสัญญาณรบกวนอยู่จึงต้องกำจัดออก โดยขั้นตอนการกำจัดสัญญาณรบกวนนั้นถือว่าเป็นขั้นตอนที่สำคัญมาก เพราะเป็นการทำให้การใช้งานมีความถูกต้องเพิ่มสูงขึ้น ซึ่งสัญญาณรบกวนที่เลือกที่จะกำจัด คือ สัญญาณรบกวนแบบซอลท์แอนด์เปปเปอร์

รูปที่ 3.3 ภาพที่ได้หลังจากการทำเทรชโฮลที่ใช้ค่าเทรชโฮลที่เหมาะสม

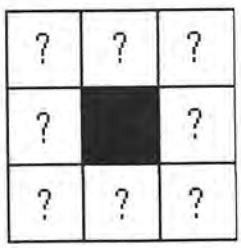
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 ภาพที่ได้หลังจากการทำเทรชโฮลที่ใช้ค่าเทรชโฮลที่สูงหรือต่ำเกินไป

3.2.2 การกำจัดสัญญาณรบกวน

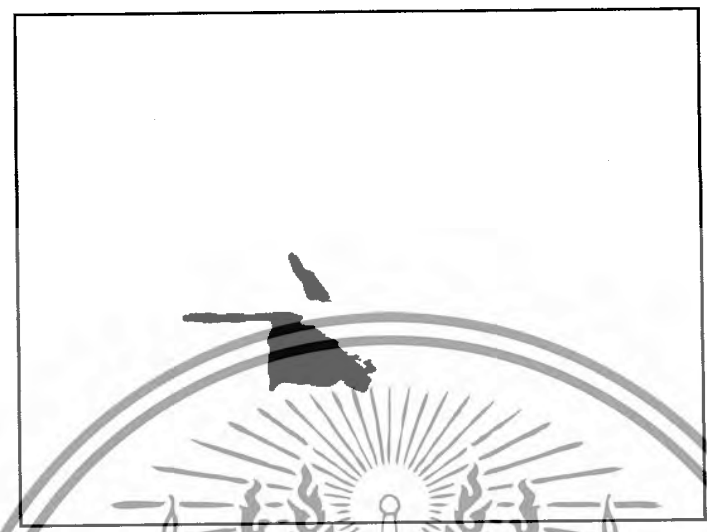
ในขั้นตอนการแปลงไฟล์รูปภาพเป็นรูปภาพขาว - ดำนั้นอาจมีสัญญาณรบกวนเกิดขึ้น และอาจทำให้การประมวลผลผิดพลาดได้จึงใช้วิธีลดสัญญาณรบกวนแบบค่ามัธยฐาน โดยทำการวนลูปไล่ไปตามพิกเซลแต่ละพิกเซล ซึ่งจะพิจารณาจุด 8 จุดล้อมรอบพิกเซลนั้นๆ โดยหากบริเวณโดยรอบมีค่าเป็น 1 มากกว่า 4 จุดขึ้นไปให้เปลี่ยนจุดตรงกลางเป็น 1 และเช่นเดียวกันถ้าหากบริเวณโดยรอบมีค่าเป็น 0 มากกว่า 4 จุดขึ้นไปก็ให้เปลี่ยนค่าจุดตรงกลางเป็น 0 จากรูปที่ 3.5 เป็นการยกตัวอย่างขนาดของหน้าต่างขนาด 3 × 3 แต่ในทางปฏิบัติจริงได้เลือกใช้ขนาด 7 × 7 เพราะสามารถกำจัดสัญญาณรบกวนได้มากกว่าแบบ 3 × 3 แต่ทั้งนี้ต้องดูความเหมาะสมด้วยว่า ต้องการรายละเอียดของภาพมากน้อยเพียงใด เพราะถ้าใช้ขนาดของหน้าต่างใหญ่ขึ้นรายละเอียดของภาพก็จะลดลง



รูปที่ 3.5 วิธีการกำจัดสัญญาณรบกวนโดยใช้หน้าต่างขนาด 3 × 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

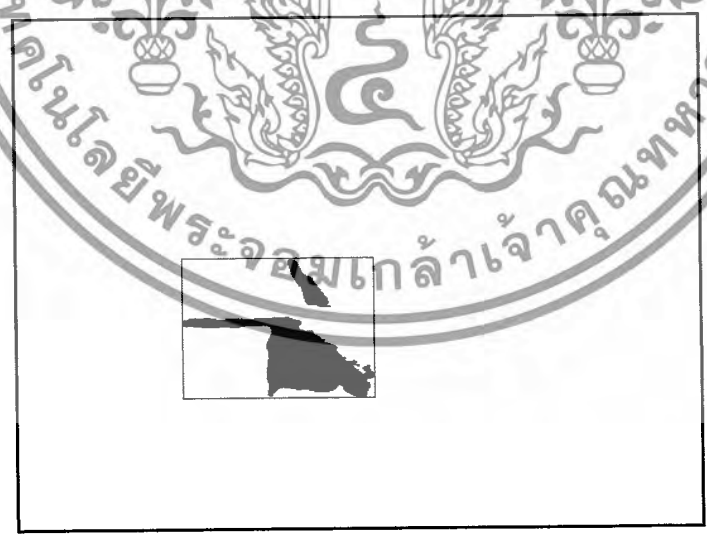
การใช้วิธีลดสัญญาณรบกวนแบบค่ามัธยฐานกำจัดสัญญาณรบกวนเป็นวิธีที่ให้ผลดีในระดับหนึ่ง



รูปที่ 3.6 ภาพหลังกำจัดสัญญาณรบกวน

3.2.3 การตีกรอบวัตถุที่เราสนใจ

จะใช้การวิธีการไล่ที่ขอบของภาพทีละด้านจนครบทั้ง 4 ด้าน เพื่อหาพิกเซลที่มีสีดำ ถ้าเจอพิกเซลที่มีสีดำ ก็จะเก็บตำแหน่ง X หรือ Y เพื่อนำไปใช้ในการตีกรอบของวัตถุ



รูปที่ 3.7 ภาพหลังจากการทำการตีกรอบ

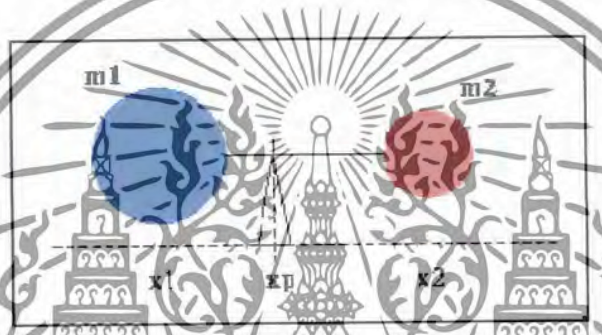
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การทำนอร์มัลไลเซชั่น

เป็นขั้นตอนที่ต้องทำก่อนการทำการรู้จำแบบการเข้าคู่รูปแบบเป็นการปรับขนาดของรูปภาพที่ได้จากการตีกรอบกับภาพที่ใช้เป็นรูปแบบ (Template) ให้มีขนาดเท่ากัน โดยได้มีการกำหนดขนาดความสูงของภาพให้มีขนาดคงที่ตามขนาดที่ได้กำหนดไว้ แต่ความกว้างยืดหยุ่นได้ตามอัตราส่วนของความสูง

3.4 การหาจุดศูนย์กลางมวลรวมของวัตถุ

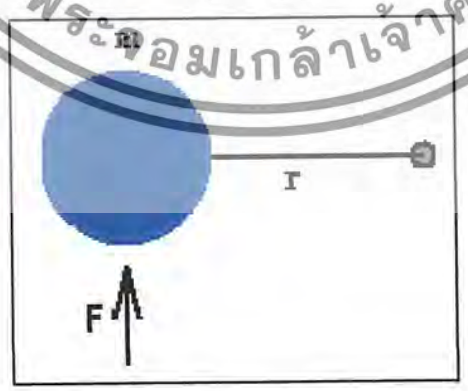
เป็นจุดที่เสมือนเป็นที่รวมมวลของวัตถุทั้งก้อนนั้น ในที่นี้ได้อาศัยการอ้างอิงจากการหาจุดที่สมดุลบนแกนหมุน



รูปที่ 3.8 จุดที่ทำให้การอยู่ในลักษณะที่สมดุล

จากรูป 3.8 จุดที่ทำให้ความสมดุล คือ จุด xp โดยที่จุดศูนย์กลางมวลนั้นสามารหาได้จากจุดที่ทำให้ทอร์ค (Torque) ของทั้งสองข้างมีขนาดเท่ากัน ซึ่งสมการของทอร์ค คือ

$$\text{Torque } (T) = Fr \tag{3.1}$$



รูปที่ 3.9 แรงที่กระทำต่อมวล m

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำให้ได้สมการ

$$m_1 g (x_1 - x_p) = m_2 g (x_2 - x_p) \quad (3.2)$$

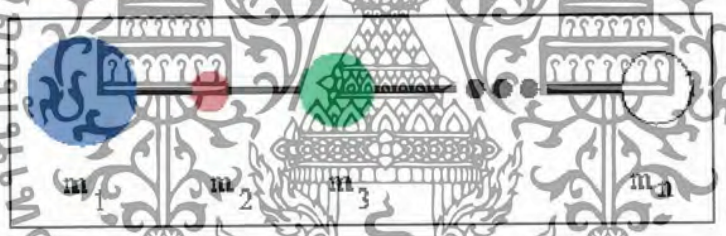
จัดรูปใหม่ได้เป็น

$$x_p (m_1 + m_2) = (x_1 m_1 + x_2 m_2) \quad (3.3)$$

แต่ถ้าอนุภาคเหล่านั้นอยู่บนระนาบ และมีจุดโคออดิเนต (Coordinate) เป็น (x1,y1), ... , (x_n, y_n) ดังนั้นจุดโคออดิเนตที่เป็นจุดศูนย์กลางมวล (x_cm , y_cm) สามารถหาได้จาก

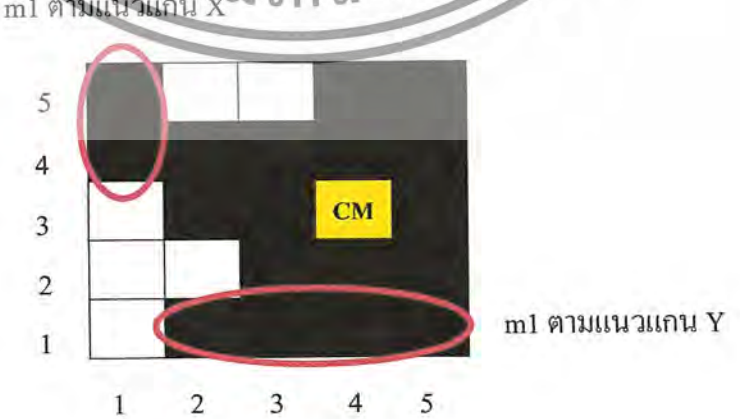
$$(m1 + \ldots + m_n) x_cm = m1\ x1 + \ldots + m_n x_n \quad (3.4)$$

$$(m1 + \ldots + m_n) y_cm = m1\ y1 + \ldots + m_n y_n \quad (3.5)$$



รูปที่ 3.10 มวลที่ต่อกันตั้งแต่ m_1 ถึง m_n

ซึ่งสามารถนำมาประยุกต์ใช้กับการหาจุดศูนย์กลางมวลของภาพได้ ดังนี้



รูปที่ 3.11 การหาจุดศูนย์กลางมวลของ m1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) หา x_cm

จากสมการ

$$(m1 + \ldots + m_n)\, x_cm = m1\, x1 + \ldots + m_n\, x_n \qquad (3.4)$$

จากรูปที่ 3.1 จะได้ m1 = 2, m2 = 3, m3 = 4, m4 = 5, m5 = 5 และ x1 = 1, x2 = 2, x3 = 3, x4 = 4, x5 = 5 นำไปแทนค่าในสมการที่ 3.4 เพื่อหา x_cm

$$(2 + 3 + 4 + 5 + 5)\, x_cm = (2 \times 1) + (3 \times 2) + (4 \times 3) + (5 \times 4) + (5 \times 5)$$

จะได้ $x_cm \approx 4$

2) หา y_cm

จากสมการ

$$(m1 + \ldots + m_n)\, y_cm = m1\, y1 + \ldots + m_n\, y_n \qquad (3.5)$$

จากรูปที่ 3.1 จะได้ m1 = 4, m2 = 3, m3 = 4, m4 = 5, m5 = 3 และ y1 = 1, y2 = 2, y3 = 3, y4 = 4, y5 = 5 นำไปแทนค่าในสมการที่ 3.5 เพื่อหา y_cm

$$(4 + 3 + 4 + 5 + 3)\, y_cm = (4 \times 1) + (3 \times 2) + (4 \times 3) + (5 \times 4) + (3 \times 5)$$

จะได้ $y_cm = 3$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 การรู้จำภาพ

เป็นการนำภาพที่รับเข้ามาจากกล้องเว็บแคมมาเปรียบเทียบกับภาพที่ใช้เป็นคู่แบบ โดยจะใช้การนับจุดพิกเซลที่เป็นสีดำ ถ้ามีจุดพิกเซลที่เป็นสีดำตรงกันมากกว่า 80 เปอร์เซ็นต์ของภาพต้นแบบ แสดงว่าภาพที่รับเข้ามานั้นผู้ใช้ได้ทำสัญลักษณ์มือที่ถูกต้อง และนำคำสั่งที่ได้นั้นไปควบคุมการเปลี่ยนหน้าสไลด์ของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ต่อไป

3.5.1 วิธีทางการเข้าคู่รูปแบบ

โดยวิธีนี้จะทำการกำหนดรูปแบบไว้ รูปแบบที่ใช้มีดังต่อไปนี้

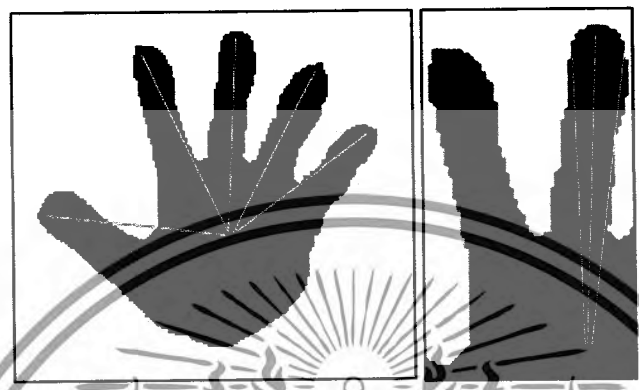
รูปที่ 3.12 ภาพรูปแบบที่กำหนดไว้

ซึ่งจะทำการนับจุดสีดำที่ตรงกันของภาพที่ได้จากการตีกรอบ และหลังจากการทำนอร์มัลไลเซชั่นเรียบร้อยแล้วกับภาพรูปแบบหลังจากการทำนอร์มัลไลเซชั่นเรียบร้อยแล้วเช่นกัน ถ้าจำนวนจุดสีดำที่นับได้มีปริมาณมากกว่าหรือเท่ากับ 80 % ของจุดสีดำที่มีอยู่ในภาพรูปแบบแสดงว่าภาพนั้นเป็นที่ตรงกับรูปแบบที่กำหนดไว้ และจะแปลงเป็นคำสั่งนำไปใช้การควบคุมสไลด์ของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 วิธีการวิเคราะห์ทางโครงสร้าง

วิธีวิเคราะห์ทางโครงสร้าง คือ การวิเคราะห์องค์ประกอบส่วนต่างๆ ของมือ โดยดึงเอาลักษณะสำคัญของมือมาวิเคราะห์ตีความหมายตามกฎที่วางไว้ ซึ่งอัตราความถูกต้องของวิธีนี้จะขึ้นอยู่กับการตั้งกฎและการวิเคราะห์กฎที่สร้างขึ้นเอง



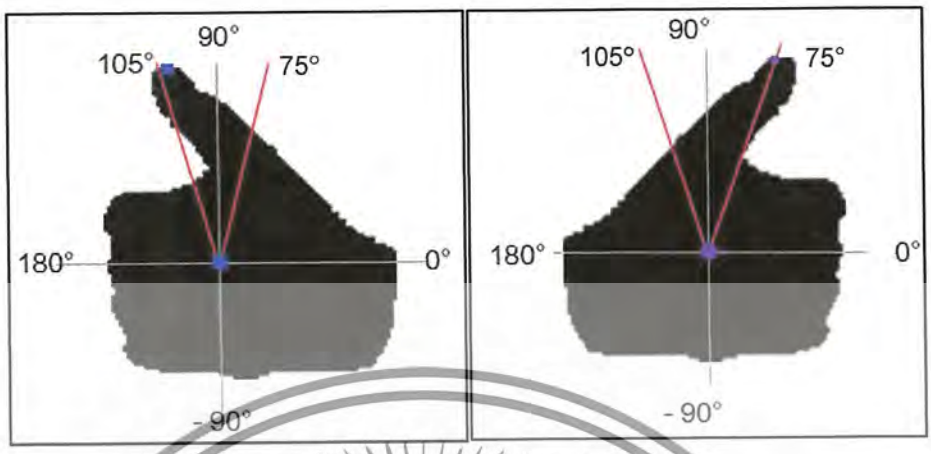
รูปที่ 3.13 ความสัมพันธ์ระหว่างตำแหน่งของจุดปลายนิ้วและจุดศูนย์กลางมวล

หลักการวิเคราะห์ทางโครงสร้างของมือที่ใช้ในโปรแกรมนี้จะอาศัยความสัมพันธ์ระหว่างตำแหน่งของปลายนิ้วมือและฝ่ามือในการจำแนกสัญญาณมือในลักษณะต่างๆ โดยในที่นี้จะเปรียบให้จุดศูนย์กลางมวลของภาพสัญญาณมือเป็นเหมือนตำแหน่งจุดศูนย์กลางของฝ่ามือ และสมมุติให้จุดปลายนิ้วมือคือตำแหน่งของขอบภาพที่มีระยะห่างจากจุดศูนย์กลางของฝ่ามือมากที่สุด ซึ่งสามารถหาได้โดยการไล่ขอบภาพแต่ละจุดด้วยการทำการหาขอบภาพ แล้วหาระยะห่างระหว่างจุดขอบภาพนั้นๆ กับจุดศูนย์กลางมวลเปรียบเทียบกับจุดขอบภาพข้างๆ ซึ่งจุดขอบภาพที่มีระยะห่างจากจุดจุดศูนย์กลางมวลมากที่สุด คือ จุดปลายนิ้ว

หลังจากได้ตำแหน่งของปลายนิ้วแต่ละจุดแล้วจะนำจุดเหล่านี้มาวิเคราะห์ว่าสัญญาณมือของภาพมีลักษณะอย่างไร โดยเปรียบเทียบจุดปลายนิ้วมือว่าทำองศากับจุดศูนย์กลางมวลในช่วงต่างๆ เท่าไร โดยเงื่อนไขที่ใช้ในการวิเคราะห์สัญญาณมือมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) กรณีที่พบปลายนิ้วมืออยู่ในช่วง 75° ถึง 105° องศา หมายถึงคำสั่งเริ่มการนำเสนอ



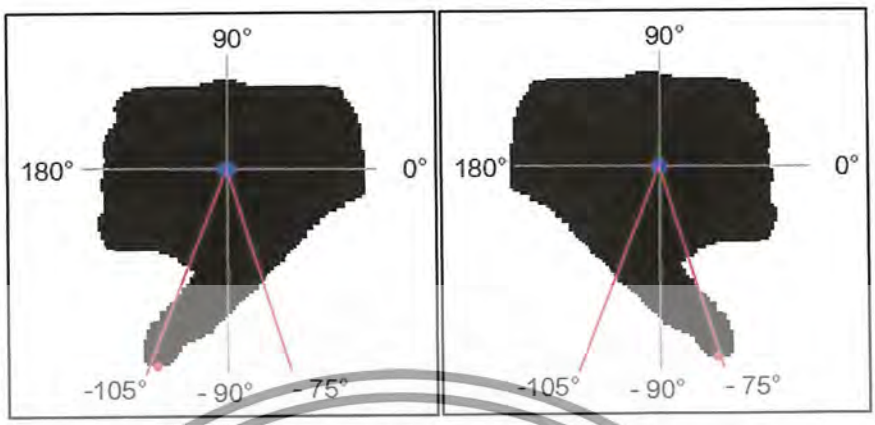
รูปที่ 3.14 การวิเคราะห์สัญญาณมือเพื่อเริ่มการนำเสนอ

2) กรณีที่พบปลายนิ้วมืออยู่ในช่วง 5° ถึง 20° องศา หมายถึงคำสั่งเลื่อนสไลด์ไปหน้าถัดไป

รูปที่ 3.15 การวิเคราะห์สัญญาณมือเลื่อนสไลด์ไปหน้าถัดไป

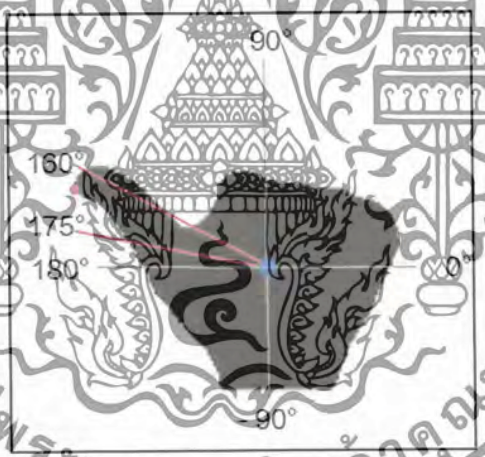
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) กรณีที่พบปลายนิ้วมืออยู่ในช่วง -75° ถึง -105° องศา หมายถึงคำสั่งสิ้นสุดการนำเสนอ



รูปที่ 3.16 การวิเคราะห์สัญญาณมือเพื่อสิ้นสุดการนำเสนอ

4) กรณีที่พบปลายนิ้วมืออยู่ในช่วง 160° ถึง 175° องศา หมายถึงคำสั่งเลื่อนสไลด์ไปก่อนหน้า



รูปที่ 3.17 การวิเคราะห์สัญญาณมือเลื่อนสไลด์ไปหน้าก่อนหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 การเขียนไลบรารีติดต่อเว็บแคม

ในการทดลองเขียนไลบรารีเพื่อใช้ในการติดต่อเว็บแคมจะใช้เทคโนโลยีของเอวีไอแคป 32 ซึ่งเป็นรูปแบบหนึ่งที่ใช้ในการบันทึกภาพมัลติมีเดียที่ถูกคิดค้นขึ้นโดยบริษัทไมโครซอฟต์โดยมีขั้นตอนในการเขียนดังต่อไปนี้

3.6.1 Platform Invoke (P/Invoke) และ API Constants

3.6.1.1 Platform Invoke (P/Invoke)

จากที่ได้กล่าวมาแล้วว่าไลบรารี avicap32.dll เป็นเอพีไอที่ไม่ได้ถูกขยายเป็น Managed Class จึงต้องใช้ PInvoke ซึ่งเป็นกลไกของภาษาดอทเน็ตที่ใช้ในการเรียก Unmanaged Functions ใน DLLs มีโค้ดดังต่อไปนี้

```
[DllImport("user32", EntryPoint = "SendMessageA")]
protected static extern int SendMessage(
    int hwnd,
    int wMsg,
    int wParam,
    [MarshalAs(UnmanagedType.AsAny)]object lParam);
```

รูปที่ 3.18 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน SendMessage

จากรูปที่ 3.18 เป็นการเรียกใช้ฟังก์ชัน SendMessage จากไลบรารี user32.dll โดยส่งอาร์กิวเมนต์ไป 4 ค่า มีชนิดเป็น int, int, int และ [MarshalAs(UnmanagedType.AsAny)]object ตามลำดับ

```
[DllImport("user32")]
protected static extern int SetWindowPos(
    int hwnd,
    int hWndInsertAfter,
    int x,
    int y,
    int cx,
    int cy,
    int wFlags);
```

รูปที่ 3.19 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน SetWindowPos

จากรูปที่ 3.19 เป็นการเรียกใช้ฟังก์ชัน SetWindowPos จากไลบรารี user32.dll โดยส่งอาร์กิวเมนต์ไป 7 ค่า มีชนิดเป็น int, int, int, int, int, int และ int ตามลำดับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
[DllImport("user32")]
protected static extern bool DestroyWindow(
    int hwnd);
```

รูปที่ 3.20 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน DestroyWindow

จากรูปที่ 3.20 เป็นการเรียกใช้ฟังก์ชัน DestroyWindow จากไลบรารี user32.dll โดยส่งอาร์กิวเมนต์ไป 1 ค่า มีชนิดเป็น int

```
[DllImport("avicap32.dll")]
protected static extern bool capGetDriverDescriptionA(
    short wDriverIndex,
    [MarshalAs(UnmanagedType.VBByRefStr)]ref String lpszName,
    int cbName,
    [MarshalAs(UnmanagedType.VBByRefStr)] ref String lpszVer,
    int cbVer);
```

รูปที่ 3.21 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน capGetDriverDescriptionA

จากรูปที่ 3.21 เป็นการเรียกใช้ฟังก์ชัน capGetDriverDescriptionA จากไลบรารี avicap32.dll โดยส่งอาร์กิวเมนต์ไป 5 ค่า มีชนิดเป็น short, [MarshalAs(UnmanagedType.VBBy RefStr)]ref String, int, [MarshalAs(UnmanagedType.VBByRefStr)]ref String และ int ตามลำดับ

```
[DllImport("avicap32.dll")]
protected static extern int capCreateCaptureWindowA(
    [MarshalAs(UnmanagedType.VBByRefStr)]ref string lpszWindowName,
    int dwStyle,
    int x,
    int y,
    int nWidth,
    int nHeight,
    int hWndParent,
    int nID);
```

รูปที่ 3.22 การใช้ P/Invoke เพื่อเตรียมการเรียกใช้ฟังก์ชัน capCreateCaptureWindowA

จากรูปที่ 3.22 เป็นการเรียกใช้ฟังก์ชัน capCreateCaptureWindowA จากไลบรารี avicap32.dll โดยส่งอาร์กิวเมนต์ไป 8 ค่า มีชนิดเป็น [MarshalAs(UnmanagedType.VBByRefStr)] ref String, int, int, int, int, int, int และ int ตามลำดับ

จะเห็นได้ว่าการใช้กลไก P/Invoke มีข้อจำกัด คือ การที่จะสามารถเรียกใช้ฟังก์ชันใดมาใช้ได้ เราต้องรู้ชื่อไฟล์ DLL ที่มีฟังก์ชันดังกล่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.1.2 เอพีไอคอนสแตนซ์ (API Constants)

การติดต่อกับเว็บแคมได้นอกจากต้องอาศัยฟังก์ชันจากไลบรารี avicap32.dll แล้ว ยังต้องอาศัยค่าคงที่ด้วยเช่น ค่าคงที่ของแมสเสจดังรูปที่ 3.23

```
private const int WM_CAP_START = 1024;
private const int WM_CAP_CONNECT = 1034;
private const int WM_CAP_DISCONNECT = 1035;
private const int WM_CAP_EDIT_COPY = 1054;
private const int WM_CAP_GET_FRAME = 1084;
private const int WM_CAP_SET_SCALE = 1077;
private const int WS_CHILD = 1073741824;
private const int WS_VISIBLE = 268435456;
```

รูปที่ 3.23 การประกาศเอพีไอคอนสแตนซ์

จากรูปที่ 3.23 จะเป็นการประกาศค่าคงที่ของเอพีไอคอนสแตนซ์ โดยค่าที่ประกาศเป็นค่าของแมสเสจที่ใช้ในการติดต่อกับเว็บแคม ซึ่งที่ค่านี้สามารถทราบได้จากไฟล์ Vfw.hในพาท C:\Program Files\Microsoft SDKs\Windows\v5.0\Include หรือในพาท C:\Program Files\ Microsoft SDKs\ Windows\v6.0A \Include (จะพบถ้าติดตั้งโปรแกรม Microsoft Visual Studio 2008)

3.6.2 การเขียนโค้ดเพื่อติดต่อกับเว็บแคม

3.6.2.1 เซตค่าแคปเจอร์วินโดว์ (Capture Window)

การที่จะติดต่อกับเว็บแคมได้เราต้องทราบ Handle ของคอนโทรล (Control) ที่ใช้ในการติดต่อ คือ Handle ของคอนโทรลนั้นๆ เอง ดังรูปที่ 3.24

```
handle = capCreateCaptureWindowA(ref deviceIndex, WS_VISIBLE | WS_CHILD, 0, 0,
    windowWidth, windowHeight, windowHandle, 0);
```

รูปที่ 3.24 การติดต่อกับ Handle Window

ซึ่งค่าที่รีเทิร์นออกมาจะเป็นค่าของ Handle ของคอนโทรล

3.6.2.2 การกำหนดค่าต่างๆ เพื่อติดต่อกับกล้อง

โค้ดภายในฟังก์ชัน Initialize นี้จะทำการกำหนดค่าต่างๆ ที่สำคัญเพื่อติดต่อกับกล้องเว็บแคมโดยได้รวมการเซตค่า Handle ของคอนโทรลกล้องเว็บแคม หลังจากนั้นก็จะส่งค่าต่างๆ ผ่านฟังก์ชัน SendMessage เพื่อไปกำหนดค่า และทำการเซตตำแหน่งที่จะแสดงโดยใช้ฟังก์ชัน SetWindowPos

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public void Initialize(int windowHandle, int windowHeight, int windowWidth)
{
    string deviceIndex = Convert.ToString(this.index);

    handle = capCreateCaptureWindowA(
        ref deviceIndex, WS_VISIBLE | WS_CHILD, 0, 0,
        windowWidth, windowHeight, windowHandle, 0);

    SendMessage(handle, WM_CAP_CONNECT, this.index, 0);
    SendMessage(handle, WM_CAP_SET_SCALE, 1, 0);

    SetWindowPos(handle, 1, 0, 0, windowWidth, windowHeight, 0);
}
```

รูปที่ 3.25 การกำหนดค่าต่างๆ เพื่อติดต่อกับกล้อง

3.6.2.3 การสั่งรับภาพจากกล้องเว็บแคม

โค้ดภายในฟังก์ชัน GetSingleFrame จะทำการสั่งให้รับภาพแฟรมถัดไปมาแสดงผลที่คอนโทรลใช้ส่งผ่านฟังก์ชัน SendMessage

```
public void GetSingleFrame()
{
    SendMessage(handle, WM_CAP_GET_FRAME, 0, 0);
    SendMessage(handle, WM_CAP_EDIT_COPY, 0, 0);
}
```

รูปที่ 3.26 การสั่งรับภาพแฟรมถัดไป

3.6.2.4 ยกเลิกการเชื่อมต่อเพื่อรับข้อมูลจากกล้อง

เมื่อมีการเรียกใช้ฟังก์ชันนี้จะทำการยกเลิกการรับข้อมูลจากกล้องที่เชื่อมต่ออยู่ โดยการเรียกใช้ WM_CAP_DRIVER_DISCONNECT ผ่านทางฟังก์ชัน SendMessage และมีการเรียกใช้ฟังก์ชัน DestroyWindow เพื่อทำการปิดวินโดวส์ในส่วนที่เราต้องการ

```
public void Stop()
{
    SendMessage(handle, WM_CAP_DISCONNECT, this.index, 0);

    DestroyWindow(handle);
}
```

รูปที่ 3.27 การยกเลิกการเชื่อมต่อข้อมูลจากกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6.2.5 การจับภาพจากกล้อง

เมื่อมีการเรียกใช้ฟังก์ชัน GetClipboardImage จะทำการจับที่ได้ภาพจากกล้องโดยการส่ง WM_CAP_EDIT_COPY ผ่านทางฟังก์ชัน SendMessage หลังจากเรียกใช้ฟังก์ชัน GetSingleFrame ซึ่งแมสเสจนี้จะทำการคัดลอกข้อมูลต่างๆ จากแต่ละเฟรมของวีดีโอ และสร้างความสัมพันธ์ระหว่างจานสีจากคลิปบอร์ด (Clipboard) จากนั้นจะสร้างออบเจ็คชื่อ dataObject เพื่อทำการเก็บข้อมูลจากคลิปบอร์ดแล้วทำการเปลี่ยนฟอร์แมตของภาพเป็นบิตแมป (Bitmap) ต่อมาจะเป็นการทำการทำนอร์มัลไลเซชั่น (Normalization) เพื่อปรับขนาดของภาพให้มีขนาดที่เหมาะสมต่อการนำไปใช้งาน และทำการกลับภาพให้ถูกต้องโดยใช้ RotateFlip ก่อนรีเทิร์นค่าออกไป

```
private Bitmap GetClipboardImage()
{
    Bitmap bitmap = null;
    Bitmap new_bmp = new Bitmap(pbImage.Width, pbImage.Height);

    IDataObject dataObject = Clipboard.GetDataObject();

    if (dataObject != null && dataObject.GetDataPresent(typeof(Bitmap)))
    {
        bitmap = (Bitmap)dataObject.GetData(typeof(Bitmap));

        using (Graphics graphics =
            Graphics.FromImage((Image)new_bmp))
        {
            if (bitmap != null)
                graphics.DrawImage(bitmap, 0, 0, pbImage.Width, pbImage.Height);
        }

        new_bmp.RotateFlip(RotateFlipType.Rotate180FlipY);
    }

    return new_bmp;
}
```

รูปที่ 3.28 การจับภาพจากกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 การเขียนไลบรารีติดต่อโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์

การสั่งงานเพื่อไปควบคุมไมโครซอฟท์เพาเวอร์พอยท์จะใช้หลักการการสั่งงานผ่านวินโดวส์แมสเสจ โดยใช้ฟังก์ชัน keybd_event และ mouse_event เป็นฟังก์ชันหลักในการใช้งาน ซึ่งจะเปรียบเสมือนเป็นการสั่งงานผ่านทางคีย์บอร์ดและเมาส์แต่ก่อนจะเรียกใช้ฟังก์ชัน keybd_event และ mouse_event จะต้องกำหนดค่าตัวแปรที่แทนรหัสของคีย์บอร์ด และเมาส์ และประกาศฟังก์ชันที่จำเป็นดังนี้

3.7.1 การประกาศค่าคงที่

ค่าคงที่ที่ใช้จะเป็นรหัสของคีย์บอร์ด (Keyboard Character Codes) และเมาส์ (Mouse Code)

```
public const int VK_UP = 0x26;          //UP ARROW key
public const int VK_RIGHT = 0x27;       //RIGHT ARROW key
public const int VK_LEFT = 0x25;        //LEFT ARROW key
public const int VK_DOWN = 0x28;        //DOWN ARROW key
public const int VK_ESCAPE = 0x1B;      //ESC key
public const int VK_F5 = 0x74;          //F5 key
public const int VK_HOME = 0x24;        //HOME key
public const int VK_END = 0x23;         //END key
public const int VK_RETURN = 0x0D;      //ENTER key
public const int VK_CONTROL = 0x11;     //CTRL key
public const int VK_A = 0x41;           //A key
public const int VK_P = 0x50;           //P key
public const int KEYEVENTF_EXTENDEDKEY = 0x1;
public const int KEYEVENTF_KEYUP = 0x02;
```

รูปที่ 3.29 ค่าคงที่ของคีย์บอร์ด

```
[Flags]
public enum MouseEventFlags
{
    LEFTDOWN = 0x00000002,
    LEFTUP = 0x00000004,
    MIDDLEDOWN = 0x00000020,
    MIDDLEUP = 0x00000040,
    MOVE = 0x00000001,
    ABSOLUTE = 0x00008000,
    RIGHTDOWN = 0x00000008,
    RIGHTUP = 0x00000010
}
```

รูปที่ 3.30 ค่าคงที่ของเมาส์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7.2 ฟังก์ชันที่ใช้งาน

ฟังก์ชันที่ใช้เป็นฟังก์ชันส่วนหนึ่งในไลบรารี user32.dll

```
[DllImport("user32.dll")]
public static extern int FindWindow(
    string lpClassName, // class name
    string lpWindowName // window name
);

[DllImport("user32.dll")]
public static extern int SendMessage(
    int hWnd, // handle to destination window
    uint Msg, // message
    int wParam, // first message parameter
    int lParam // second message parameter
);

[DllImport("User32.Dll")]
public static extern void GetWindowText(int h, StringBuilder s, int nMaxCount);

[DllImport("user32")]
public static extern int GetWindow(int hwnd, int wCmd);

[DllImport("user32")]
public static extern int IsWindowVisible(int hwnd);

[DllImport("user32")]
public static extern int GetDesktopWindow();

[DllImport("user32.dll")]
public static extern void keybd_event(byte bVk, byte bScan, uint dwFlags,
UIntPtr dwExtraInfo);

[DllImport("user32.dll")]
public static extern void mouse_event(uint dwFlags, uint dx, uint dy, uint dwData,
  int dwExtraInfo);
```

รูปที่ 3.31 ฟังก์ชันที่ใช้งานในการใช้งานเวิ้นโดวส์แมสเสจ

3.7.3 การทำงานของโปรแกรม

มีขั้นตอนดังต่อนี้

1) ทำการตรวจหาโปรแกรมที่เปิดใช้งานอยู่ในขณะนั้นแล้วดึงค่า Handle ของแต่ละโปรแกรมออกมาโดยใช้ฟังก์ชัน GetDesktopWindow และ GetWindow เมื่อได้ค่า Handle ของแต่ละโปรแกรมแล้วจะส่งค่า Handle ไปกับฟังก์ชัน GetWindowText ก็จะรีเทิร์นชื่อของโปรแกรมนั้นๆ (Title bar name) มาเก็บไว้ในตัวแปล windowTitle เมื่อสิ้นสุดการทำงานในฟังก์ชันนี้จะได้ชื่อของโปรแกรมที่ทำงานอยู่ในขณะนั้นทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
private void GetTaskWindows()
{
    // Get the desktopwindow handle
    int nDeshWndHandle = NativeWin32.GetDesktopWindow();
    // Get the first child window
    int nChildHandle = NativeWin32.GetWindow(nDeshWndHandle, NativeWin32.GW_CHILD);

    while (nChildHandle != 0)
    {
        //If the child window is this (SendKeys) application then ignore it.
        if (nChildHandle == this.Handle)
        {
            nChildHandle = NativeWin32.GetWindow(nChildHandle, NativeWin32.GW_HWNDNEXT);
        }
        // Get only visible windows
        if (NativeWin32.IsWindowVisible(nChildHandle) != 0)
        {
            StringBuilder sbTitle = new StringBuilder(1024);
            // Read the Title bar text on the windows to put in combobox
            NativeWin32.GetWindowText(nChildHandle, sbTitle, sbTitle.Capacity);
            String sWinTitle = sbTitle.ToString();
            {
                if (sWinTitle.Length > 0)
                {
                    windowTitle.Add(sWinTitle);

                    countWindowTitle++;
                }
            }
        }
        // Look for the next child.
        nChildHandle = NativeWin32.GetWindow(nChildHandle, NativeWin32.GW_HWNDNEXT);
    }
}
```

รูปที่ 3.32 ฟังก์ชัน GetTaskWindows

2) หลังจากได้ชื่อของโปรแกรมทุกโปรแกรมแล้วจะทำการค้นหาชื่อของโปรแกรมที่ต้องการโดยการกำหนดคำหรือข้อความซึ่งเป็นส่วนหนึ่งของชื่อโปรแกรมทั้งหมดที่เราต้องการ เช่น เรากำหนดคำว่า “PowerPoint Slide Show” จะเป็นส่วนหนึ่งของชื่อโปรแกรม “PowerPoint Slide Show - [Presentation1]” ที่ทำงานอยู่ในขณะนั้น และเก็บไว้ในตัวแปล handleWindow เพื่อใช้งานต่อไป

```
public void SearchTextForMatch(string text)
{
    string s = text;
    if (s.IndexOf("PowerPoint Slide Show") != -1)
    {
        handleWindow = text;
    }
}
```

รูปที่ 3.33 ฟังก์ชัน SearchTextForMatch

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) ในการใช้ฟังก์ชัน keybd_event และ mouse_event จะมีผลกับโปรแกรมที่ทำงานอยู่หน้าสุด (Active Window) การกำหนดให้โปรแกรมที่ต้องการทำงานอยู่หน้าสุดจะเรียกใช้ฟังก์ชัน SetHandleWindow โดย Handle ของโปรแกรมที่จะส่งเข้าไปยังฟังก์ชันนี้ได้จากการใช้ฟังก์ชัน SearchTextForMatch ที่ผ่านมา

```
public void SetHandleWindow(string stringHandle)
{
    int iHandle = NativeWin32.FindWindow(null, stringHandle);
    NativeWin32.SetForegroundWindow(iHandle);
}
```

รูปที่ 3.34 ฟังก์ชัน SetHandleWindow

4) การสั่งงานไปยังโปรแกรมที่ต้องการโดยใช้ฟังก์ชัน keybd_event และ mouse_event

```
public void PageUp(int Handle)
{
    this.Handle = Handle;
    InitializeTaskWindows();
    SetHandleWindow(handleWindow);
    NativeWin32.keybd_event(NativeWin32.VK_HOME, 1, 0, (UIntPtr)0);    // Home Press
    NativeWin32.keybd_event(NativeWin32.VK_HOME, 1, NativeWin32.KEYEVENTF_KEYUP, (UIntPtr)0); // Home Release
    handleWindow = null;
    windowTitle.Clear();
    countWindowTitle = 0;
}
```

รูปที่ 3.35 ฟังก์ชัน PageUp

จากรูปที่ 3.35 เป็นตัวอย่างของการส่งวินโดวส์เมสเสจที่ชื่อ VK_HOME ผ่านฟังก์ชัน keybd_event ผลของการเรียกใช้ฟังก์ชันนี้เปรียบเสมือนการกดปุ่ม Home บนแป้นคีย์บอร์ด (กดและปล่อยปุ่ม)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
public void PressLeftMouse(int Handle)
{
    this.Handle = Handle;
    InitializeTaskWindows();
    SetHandleWindow(handleWindow);
    NativeWin32.mouse_event((uint)NativeWin32.MouseEventFlags.LEFTDOWN, 0, 0, 0, 0);
    handleWindow = null;
    windowTitle.Clear();
    countWindowTitle = 0;
}
```

รูปที่ 3.36 ฟังก์ชัน PressLeftMouse

จากรูปที่ 3.36 เป็นตัวอย่างของการส่งวินโดวส์เมสเสจที่ชื่อ LEFTDOWN ผ่านฟังก์ชัน mouse_event ผลของการเรียกใช้ฟังก์ชันนี้เปรียบเสมือนการกดปุ่มเมาส์ด้านซ้ายค้างไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การทดลองและผลการทดลอง

4.1 การทดลองจับภาพสัญญาณมือโดยใช้ถุงมือสี

การทดลองจะประมวลผลภาพที่มีขนาดเท่ากับ 320 × 240 และใช้โมเดลสีแบบ RGB โดยภาพที่รับเข้ามาจากกล้องเว็บแคมจะเป็นภาพสี และผ่านการแปลงเป็นรูปภาพขาว - ดำ โดยใช้วิธีเทรชโฮลซึ่งค่าเทรชโฮลที่ใช้ได้มาจากการคำนวณหาค่าเฉลี่ยของค่า H, S และ L ที่ได้จากภาพที่รับเข้ามาในขั้นตอนของการเรียนรู้สี โดยภาพขาว - ดำที่นำมาใช้ในการทดลองควรมีลักษณะดังนี้

- ภาพควรอยู่ในสภาวะแสงที่มีลักษณะที่ใกล้เคียงกับสภาวะแสงของภาพที่ใช้ในการเรียนรู้สี
- ภาพไม่ควรมีสีที่ใกล้เคียงกับถุงมืออยู่ที่ฉากหลังของภาพ
- ภาพของวัตถุสีดำในภาพจะต้องไม่เล็กจนเกินไป โดยที่ผู้ใช้ควรอยู่ห่างจากกล้องเว็บแคมไม่เกิน 1.5 เมตร เพราะถ้าไกลกว่านี้ภาพที่ได้จะไม่เป็นรูปร่างมือ ทำให้ไม่สามารถทำการรู้จำได้

การทดลองหาสัญญาณมือโดย ไม่ใช้ถุงมือสีสามารถเลือกเทคนิคการรู้จำได้สองวิธีดังนี้

4.1.1 การทดลองส่วนของการรู้จำแบบเข้าคู่รูปแบบโดยใช้ถุงมือสี

การทดลองเข้าคู่รูปแบบจะนำภาพที่ผ่านการประมวลผลข้างต้นมาเปรียบเทียบกับรูปแบบของภาพที่กำหนด โดยลักษณะของรูปแบบที่ใช้ควรมีลักษณะดังนี้

- ควรมีลักษณะของมือที่ชัดเจน และควรจะเป็นสีดำทั่วทั้งภาพ
- ควรเป็นลักษณะของมือที่อยู่ในท่าที่ง่ายต่อการทำท่าทาง
- ภาพที่ใช้ในการแทนแต่ละคำสั่งไม่ควรมีลักษณะที่ใกล้เคียงกันมากนัก

ทำการทดลองโดยการให้ผู้ใช้ 10 คน ทำท่าตามที่ได้กำหนดไว้ โดยให้แต่ละคนทำท่า ท่าละ 10 ครั้ง โดยที่ผู้ใช้จะต้องทำท่าให้ตรงกับรูปแบบที่กำหนดอย่างน้อย 80% ท่าที่ใช้การทดลองมีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1. หรือ 2. แทนคำสั่งเริ่มต้นการนำเสนอ

3. หรือ 4. แทนคำสั่งสิ้นสุดการนำเสนอ

5. แทนคำสั่งไปที่สไลด์หน้าถัดไป

6. แทนคำสั่งไปที่สไลด์หน้าก่อนหน้า

รูปที่ 4.1 ท่าทางของสัญญาณมือที่ใช้ในการทดลอง

ตารางที่ 4.1 ผลการทดลองการรู้จำแบบเข้าคู่รูปแบบโดยใช้ถุงมือสี

1	10	8	10	10	8	9
2	10	7	7	8	6	9
3	6	10	8	7	8	8
4	9	9	6	8	9	7
5	9	8	9	8	8	10
6	7	7	8	9	10	8
7	9	8	7	8	7	7
8	8	9	7	7	6	9
9	7	10	9	6	10	10
10	10	7	10	7	9	8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

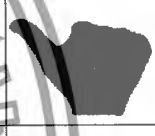
4.1.2 การทดลองส่วนของการรู้จำแบบการวิเคราะห์ทางโครงสร้างโดยใช้ถุงมือสี

การทดลองในส่วนนี้จะอาศัยองค์ประกอบส่วนต่างๆ ของมือ ประกอบด้วย ปลายนิ้วมือ และจุดศูนย์กลางมวลของภาพมือเข้ามาใช้วิเคราะห์ทางโครงสร้าง ภาพอินพุทที่รับเข้ามาควรมีลักษณะดังนี้

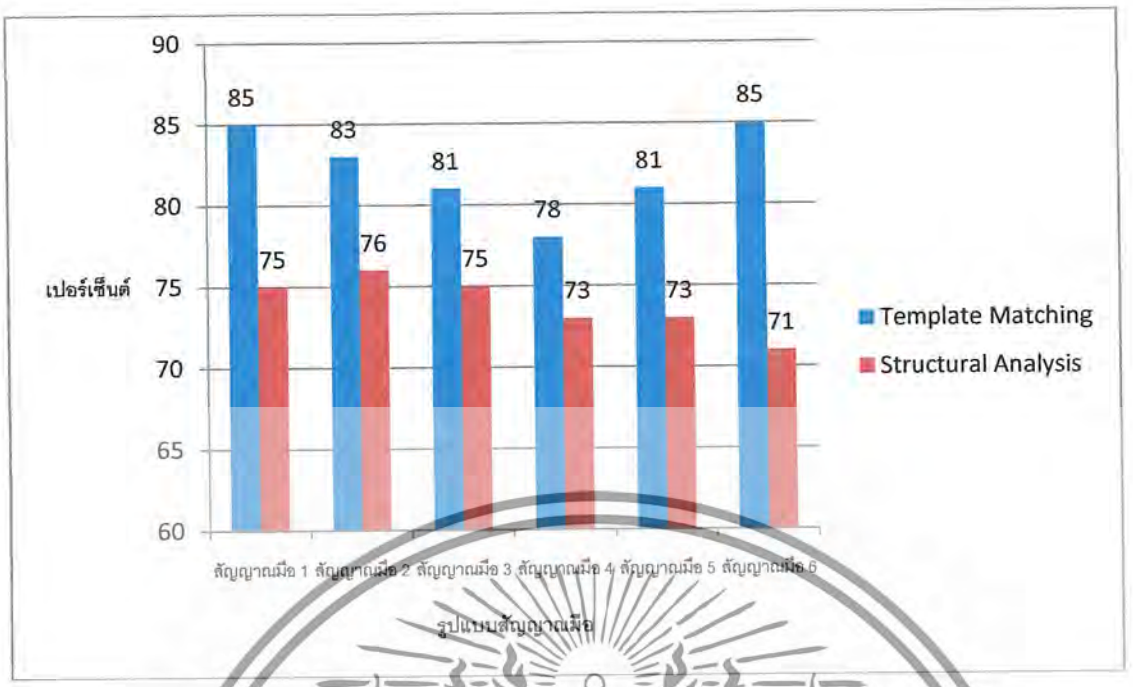
- ภาพที่ได้ควรมีขอบของมือที่ชัดเจน เนื่องจากในการหาจุดปลายนิ้วจะใช้เทคนิคไล่ขอบภาพ ซึ่งความถูกต้องของผลลัพธ์นั้นจะขึ้นอยู่กับความชัดเจนของขอบภาพ
- ในการออกคำสั่งแต่ละคำสั่ง ตำแหน่งของปลายนิ้วมือต้องอยู่ในช่วงองศาที่กำหนดไว้สำหรับแต่ละคำสั่ง
- ภาพที่ได้ควรเป็นสีดำทั่วทั้งภาพ เพื่อให้สามารถคำนวณจุด CM ของภาพได้ถูกต้อง

ทำการทดลองโดยการให้ผู้ใช้ 10 คน ทำท่าตามที่ได้กำหนดไว้ โดยให้แต่ละคนทำท่า ท่าละ 10 ครั้ง

ตารางที่ 4.2 ผลการทดลองการวิเคราะห์ทางโครงสร้างโดยใช้ถุงมือสี

						
1	10	9	6	8	6	5
2	7	8	7	7	9	8
3	5	7	7	8	7	6
4	8	7	8	9	5	7
5	6	5	9	7	8	7
6	9	10	8	6	9	8
7	5	8	6	7	8	8
8	7	6	7	8	5	7
9	9	8	9	8	7	8
10	9	8	8	6	9	7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 แผนภูมิเปรียบเทียบการทดลองจับภาพสัญญาณมือโดยใช้ถุงมือสี

จากการทดลองจะเห็นได้ว่าเทคนิคการรู้จำแบบจับคู่รูปแบบมีความแม่นยำกว่าการรู้จำแบบวิธีวิเคราะห์โครงสร้าง โดยความถูกต้องของการรู้จำแบบจับคู่รูปแบบจะแตกต่างกันไปในแต่ละคนขึ้นอยู่กับรูปแบบที่ใช้เปรียบเทียบ ส่วนการรู้จำแบบวิเคราะห์โครงสร้างจะตีความหมายสัญญาณมือที่ไม่มีความหมายผิดเป็นสัญญาณที่ถูกต้องอยู่บ่อยครั้ง

4.2 การทดลองจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสี

การทดลองจะหาความแตกต่างระหว่างภาพฉากหลังกับภาพเฟรมปัจจุบันจากกล้อง ผลลัพธ์ที่ได้คือภาพวัตถุที่เข้ามาในกล้องคือภาพสัญญาณมือ โดยภาพอินพุทที่นำมาประมวลผลของมีลักษณะดังนี้

- ฉากหลังของภาพจากกล้องควรคงที่ เพื่อให้เมื่อนำภาพฉากหลังและภาพเฟรมปัจจุบันมาหาความแตกต่างแล้วผลลัพธ์ที่ได้เหลือแต่ภาพสัญญาณมือเท่านั้น โดยไม่มีสัญญาณรบกวน
- ภาพสัญญาณมือที่ได้ควรให้มีส่วนของแขนหรือไหล่เข้ามาในกล้องให้น้อยที่สุด เพราะจะส่งผลต่อการจำกัดขอบเขตของมือเพื่อนำไปทำการรู้จำต่อไป
- สภาพแวดล้อมอื่นๆ เช่น แสง ระยะห่างระหว่างมือกับกล้องควรอยู่ในขอบเขตที่กำหนด

การทดลองหาสัญญาณมือโดยไม่ใช้ถุงมือสีสามารถเลือกเทคนิคการรู้จำได้สองวิธีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.1 การทดลองส่วนของการรู้จำแบบเข้าคู่รูปแบบโดยไม่ใช้ถุงมือสี

ทำการทดลองโดยการให้ผู้ใช้ 10 คน ทำท่าตามที่ได้กำหนดไว้ โดยให้แต่ละคนทำท่า ท่าละ 10 ครั้ง

ตารางที่ 4.3 ผลการทดลองการรู้จำแบบเข้าคู่รูปแบบโดยไม่ใช้ถุงมือสี

1	5	5	0	2	3	5
2	2	3	1	4	4	2
3	2	1	1	3	5	4
4	5	4	0	3	2	3
5	4	3	2	0	1	0
6	3	2	1	3	3	2
7	4	4	1	1	2	3
8	4	5	1	0	1	2
9	5	3	3	0	3	3
10	4	2	2	0	2	2

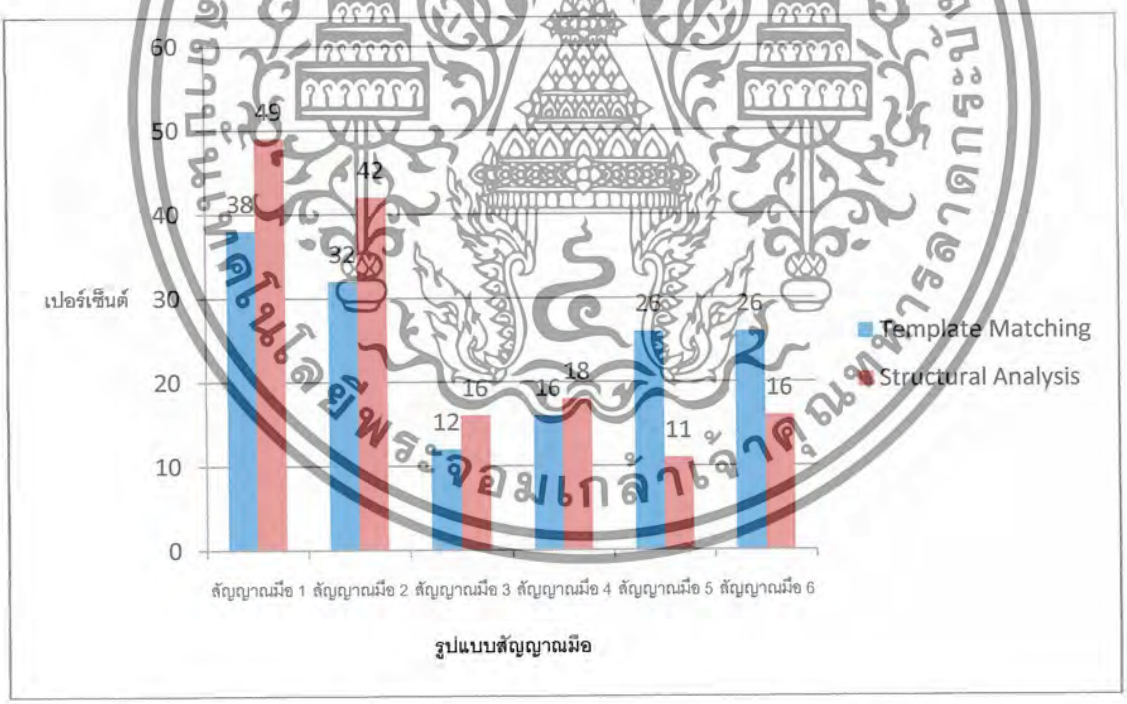
4.2.2 การทดลองส่วนของการรู้จำแบบวิเคราะห์ทางโครงสร้างโดยไม่ใช้ถุงมือสี

ตารางแสดงการทดลองแสดงจำนวนครั้งที่ผู้ใช้ทำท่าได้ถูกต้องจากการทำท่าทั้งหมด 10 ครั้ง ในแต่ละท่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.4 ผลการทดลองการวิเคราะห์ทางโครงสร้างโดยไม่ใช้ถุงมือสี

1	6	4	1	0	2	4
2	4	3	1	3	1	0
3	5	2	0	2	0	3
4	5	6	2	4	1	1
5	3	6	2	3	0	0
6	4	5	3	2	0	1
7	4	3	1	0	0	0
8	6	5	1	0	2	2
9	7	6	2	1	3	1
10	5	2	3	3	2	4



รูปที่ 4.3 แผนภูมิเปรียบเทียบการทดลองจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสี

จากแผนภูมิเมื่อเปรียบเทียบการจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสีกับแบบใช้ถุงมือสีจะเห็นได้ว่าการจับภาพสัญญาณมือโดยใช้ถุงมือสีมีค่าความถูกต้องสูงกว่า ทั้งนี้เนื่องจากการจับภาพสัญญาณมือโดยไม่ใช้ถุงมือสีจะใช้เทคนิคการลบกันของภาพพื้นหลังซึ่งภาพที่ได้ยังมีองค์ประกอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อื่นนอกจากมือ เช่น แขนและไหล่เข้ามาด้วย ทำให้สามารถตีความหมายของภาพได้ยากขึ้น สามารถสรุปผลการทำงานโดยรวมได้ดังนี้

ตารางที่ 4.5 สรุปผลการทดลอง

ผลการทดลอง	เปอร์เซ็นต์ที่รู้จำได้
การทดลองส่วนการรู้จำแบบเข้าคู่รูปแบบโดยใช้ถุงมือสี	82.16
การทดลองส่วนการรู้จำแบบวิเคราะห์โครงสร้างโดยใช้ถุงมือสี	73.5
การทดลองส่วนการรู้จำแบบเข้าคู่รูปแบบโดยไม่ใช้ถุงมือสี	25
การทดลองส่วนการรู้จำแบบวิเคราะห์โครงสร้างโดยไม่ใช้ถุงมือสี	25.33

4.3 การทดลองหาค่าเปอร์เซ็นต์ความถูกต้องที่เหมาะสมสำหรับการเปรียบเทียบในวิธีการเข้าคู่รูปแบบ

ทำการทดลองโดยการให้ผู้ใช้ 5 คน ทำท่าตามที่ได้กำหนดไว้ โดยให้แต่ละคนทำท่า ท่าละ 5 ครั้ง ณ ที่เปอร์เซ็นต์ความถูกต้องค่าต่างๆ

ตารางที่ 4.6 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 60 %

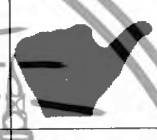
1	0	0	0	0	1	0
2	1	2	0	0	1	0
3	0	1	0	1	0	1
4	0	1	0	1	0	0
5	2	0	1	0	0	0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.7 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 70 %

1	0	1	1	0	2	0
2	1	1	1	1	1	0
3	0	1	0	2	0	3
4	2	1	2	2	1	1
5	1	1	0	0	0	0

ตารางที่ 4.8 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 80 %

						
1	5	4	4	3	3	4
2	4	5	4	4	4	5
3	5	4	5	3	5	3
4	5	5	3	4	4	5
5	3	5	4	3	5	4

ตารางที่ 4.9 ผลการทดลองที่ค่าเปอร์เซ็นต์ความถูกต้องเท่ากับ 90 %

1	3	4	1	0	3	4
2	4	3	1	3	4	4
3	2	3	0	2	3	3
4	3	3	3	2	4	5
5	3	5	2	3	5	4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.10 ผลความถูกต้องคิดเป็นเปอร์เซ็นต์ ณ ที่ค่าเปอร์เซ็นต์ความถูกต้องค่าต่างๆ

60%	12%	16%	4%	8%	8%	4%
70%	16%	20%	16%	20%	16%	16%
80%	88%	92%	80%	68%	84%	84%
90%	60%	72%	28%	40%	76%	80%

รูปที่ 4.4 แผนภูมิเปรียบเทียบการทดลองใช้ค่าเปอร์เซ็นต์ความถูกต้องที่ค่าต่างๆ

จากการทดลองจะเห็นได้ว่า ที่เปอร์เซ็นต์ความถูกต้องเท่ากับ 80 % โปรแกรมสามารถตีความหมายของภาพสัญญาณมือได้มากที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
บทสรุป และวิจารณ์

5.1 สรุปการทำงาน

การตีความหมายของภาพเพื่อส่งคำสั่งไปควบคุมการทำงานในส่วนของการนำเสนอของโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์ในกรณีที่ผู้ใช้สวมถุงมือซึ่งใช้การรู้จำ 2 แบบ ด้วยกัน คือ แบบเข้าคู่รูปแบบซึ่งมีความถูกต้องคิดเป็น 82.16 % และแบบวิเคราะห์โครงสร้างที่ให้ความถูกต้องคิดเป็น 73.5% ซึ่งมีปัจจัยหลายประการที่ทำให้การตีความหมายผิดพลาดได้ง่าย ตั้งแต่สภาวะของแสงที่มืดหรือสว่างเกินไปจนทำให้การแปลงภาพเป็นรูปภาพขาว - ดำแล้วได้ข้อมูลไม่ครบถ้วน เช่น บางส่วนของมือขาดหายไป เป็นต้น การตั้งกล้องที่ห่างเกินไปทำให้วัตถุที่ตรวจจับได้มีขนาดเล็กเกินไป หรือการตั้งกล้องในบริเวณที่ฉากหลังมีวัตถุสีเดียวกับสีของถุงมือ ทำให้มีสัญญาณรบกวนเกิดขึ้นภายในภาพ รวมทั้งส่วนของการทำการลบกันของภาพพื้นหลังที่ต้องปิดการทำงานฟังก์ชันการปรับความสว่างโดยอัตโนมัติของกล้องเว็บแคม เนื่องจากความสว่างที่ไม่คงที่ทำให้เกิดสัญญาณรบกวนที่ฉากหลังของภาพผลลัพธ์ที่ได้ และการตีกรอบของวัตถุในกรณีที่ไม่ใส่ถุงมือยังมีความผิดพลาดอยู่มากเนื่องจาก ภาพผลลัพธ์ที่ได้จากการทำการลบกันของภาพพื้นหลังจะได้ส่วนที่เป็นมือ แขน และไหล่ติดมาด้วย ทำให้การตีกรอบของวัตถุที่สนใจผิดพลาดส่งผลให้การตีความหมายผิดพลาดตามไปด้วย โดยสามารถคิดเป็นเปอร์เซ็นต์ความถูกต้องได้ 25 % สำหรับกรณีที่ใช้วิธีการเข้าคู่รูปแบบ และ 25.33 % สำหรับวิธีการวิเคราะห์โครงสร้าง

ปัจจัยต่างๆ เหล่านี้ล้วนมีผลให้ประสิทธิภาพการทำงานของระบบลดลง รวมทั้งวิธีที่ใช้ในการรู้จำก็ส่งผลต่อประสิทธิภาพของระบบเช่นกัน ซึ่งสามารถสรุปผลจากการใช้การรู้จำทั้ง 2 วิธี ได้ดังต่อไปนี้

1) แบบการเข้าคู่รูปแบบเป็นวิธีที่ต้องมีรูปแบบที่สร้างขึ้นมาสำหรับตีความหมายของสัญลักษณ์มือ ซึ่งเป็นวิธีการที่ค่อนข้างอ่อนไหวต่อขนาดมือของผู้ใช้ การเอียงของมือ ความสมบูรณ์ของมือ
2) แบบวิเคราะห์ทางโครงสร้างเป็นวิธีที่วิเคราะห์โครงสร้างของมือ เช่น ตำแหน่งของนิ้วมือที่อยู่ในภาพ ซึ่งให้ผลความถูกต้องมากกว่าวิธีแรก แต่ในบางกรณีอาจเกิดความซ้ำซ้อนของท่าได้ เช่น กรณีที่ทำท่ากำมือแล้วชูนิ้วโป้ง กับ กำมือแล้วชูนิ้วชี้ โดยที่ตำแหน่งของนิ้วเป็นตำแหน่งเดียวกันทำให้ออกคำสั่งได้เหมือนกัน

จากการทดลองในกรณีใส่ถุงมือวิธีการรู้จำแบบเข้าคู่รูปแบบมีความถูกต้องมากกว่า เนื่องจากภาพสัญญาณมือที่ใช้ยังไม่มีความซับซ้อนมากนัก และการคิดเปอร์เซ็นต์ความถูกต้องเพียงแค่ 80 % ทำให้ท่าสัญญาณมือบางท่าผู้ใช้ไม่ต้องทำมือในลักษณะที่เอียงเท่ากับรูปแบบ 100 % ก็

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถสั่งการการเปลี่ยนหน้าสไลด์ได้ ซึ่งในกรณีการรู้จำแบบโครงสร้างที่ต้องอาศัยความแม่นยำมากกว่าเนื่องจากได้มีการกำหนดตำแหน่งของนิ้วมือว่าต้องอยู่ระหว่างองศาใดถึงองศาใด ซึ่งการเอียงมือผิดไปเล็กน้อยทำให้ความหมายของผิดไป หรือไปซ้ำซ้อนกับความหมายอื่นได้ วิธีนี้จึงเหมาะกับสัญญาณมือที่ซับซ้อนที่ต้องอาศัยตำแหน่งของนิ้วหลายนิ้วเป็นตัวอ้างอิง แต่ถ้าสัญญาณมือไม่ซับซ้อนวิธีการเข้าคู่รูปแบบเป็นวิธีที่เหมาะในการรู้จำ

5.2 ปัญหาและอุปสรรค

1) สภาวะของแสงจะมีผลกระทบมาก คือ สภาวะแสงที่สว่างเกินไปหรือมืดเกินไปจะส่งต่อคุณภาพของภาพทำให้การแปลงเป็นรูปภาพขาว - ดำ ซึ่งได้ข้อมูลไม่ครบถ้วน ทำให้ได้สัญลักษณ์มือที่ไม่ชัดเจน
2) ถ้ามืออยู่ไกลจากกล้องเว็บแคมเกิน 1.5 เมตร รายละเอียดของรูปภาพขาว - ดำจะลดลง ซึ่งส่งผลต่อการตีความหมายของรูปทำให้ตีความหมายได้ไม่ถูกต้อง
3) ความถูกต้องของการตีความหมายของภาพโดยวิธีการรู้จำแบบเข้าคู่รูปแบบจะแตกต่างกันไปตามแต่ละบุคคล เนื่องจากลักษณะของมือแต่ละคนจะแตกต่าง เช่น มือของผู้ชายจะยาวกว่ามือของผู้หญิง เป็นต้น
4) การตีความหมายของภาพโดยวิธีการรู้จำแบบวิเคราะห์ทางโครงสร้างไม่สามารถแยกแยะความแตกต่างระหว่างมือบางแบบได้ เช่น กรณีที่ออกคำสั่งไปที่สไลด์หน้าแรกสุดซึ่งจะต้องทำสัญญาณมือในลักษณะกำมือและชูนิ้วโป้งขึ้นอาจซ้ำซ้อนกับการกำมือและชูนิ้วชี้แทนได้
5) การตีความหมายของภาพโดยไม่ใช้ถุงมือสีนั้นยังมีความผิดพลาดสูง เนื่องจากภาพที่ได้จากการทำการลบกันของภาพพื้นหลังจะได้ภาพส่วนอื่นนอกเหนือจากภาพสัญญาณมือเข้ามาด้วย เช่น แขนและไหล่ ทำให้ตีกรอบเฉพาะส่วนที่เป็นมือได้ยากขึ้น

5.3 ข้อเสนอแนะ

1) แสงและสิ่งแวดล้อมเป็นปัจจัยที่ส่งผลต่อประสิทธิภาพของระบบมาก ดังนั้นหากนำไปใช้ประยุกต์ในงานด้านต่างๆ จำเป็นต้องควบคุมสภาวะแสงให้เหมาะสม
2) ลองใช้ Neural Network ในการรู้จำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

สัจจะ จรัสรุ่งรวีวร.**Visual c# 2005 ฉบับสมบูรณ์,** DEV BOOK Infopress Developer Book.

Rafael C. Gonzalez, Richard E. Woods.Digital Image Processing Second Edition, Prentice – Hall,Inc.

Russ, John C. 2007.**The Image Processing handbook,** New York, CRC Press.

Pratt William K. 2007. **Digital Image Processing PIKS Scientific Inside,** d New jersey, John Wiley & Sons

Image Processing in C#, http://www.codersource.net/csharp_image_Processing.aspx

Image Processing for Dummies with C# and GDI+ Part 2 - Convolution Filters, http://www.codeproject.com/KB/GDIplus/csharpfilters.aspx?fid=3511&df=90&mpp=25&noise=3&sort=Position&view=Quick&fr=176&select=991004

C# Tutorial – Image Editing: Saving Cropping, and Resizing, http://blog.paranoidferret.com/?p=11

Edge Detection http://fivedots.coe.psu.ac.th/~montri/Teaching/240-373/Chapter8.pdf

HSL Colorspace http://www.chaospro.de/documentation/html/paletteeditor/colorspace_hsl.htm

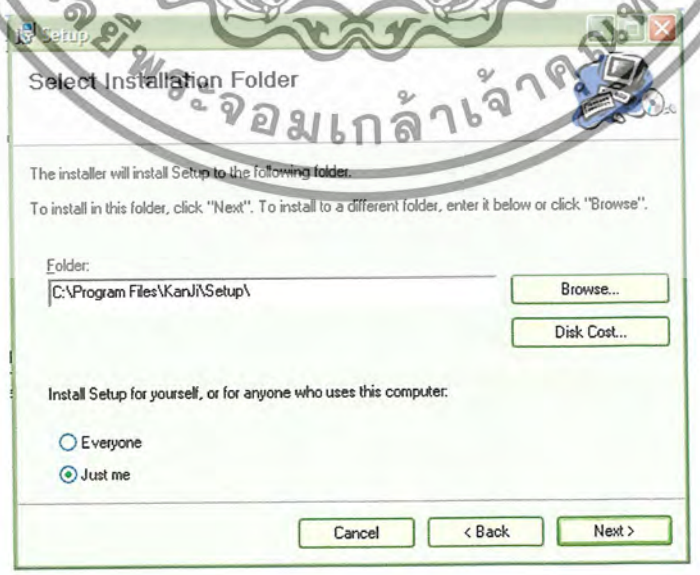
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
ขั้นตอนการติดตั้งโปรแกรม

1) กดปุ่ม Next เพื่อเริ่มขั้นตอนการติดตั้ง

รูปที่ ก.1 หน้าจอเริ่มต้นการติดตั้ง

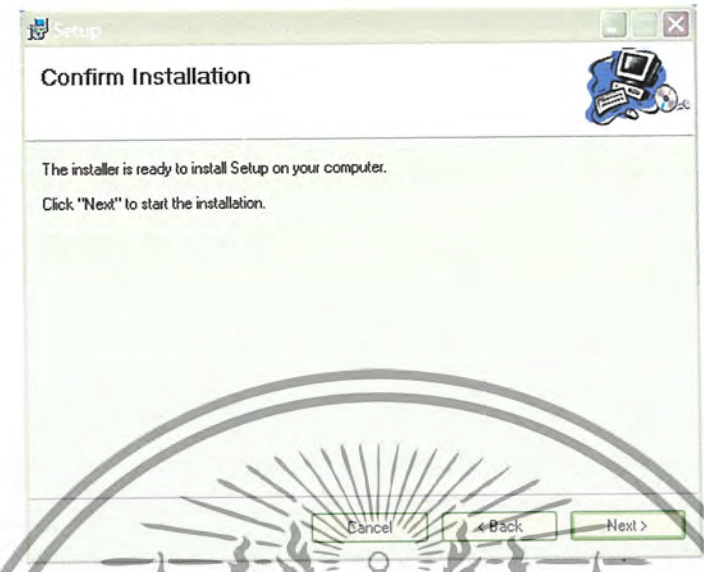
2) เลือกไดเรคทอรี (Directory) ที่ต้องการติดตั้งโปรแกรม สามารถเลือกดูพื้นที่หน่วยความที่ใช้ได้



รูปที่ ก.2 หน้าจอเลือกโฟลเดอร์สำหรับการติดตั้ง

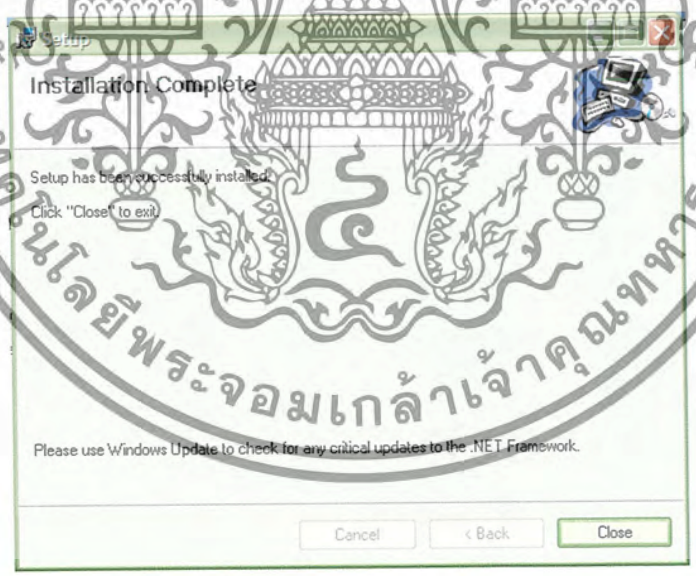
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) กดปุ่ม Next เพื่อยืนยันการติดตั้ง ระบบจะทำการติดตั้งโปรแกรม



รูปที่ ก.3 หน้าจอยืนยันการติดตั้ง

4) หลังการติดตั้งเสร็จสิ้น กดปุ่ม Close เพื่อสิ้นสุดการติดตั้ง



รูปที่ ก.4 หน้าจอสิ้นสุดการติดตั้ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข

คู่มือการใช้งาน

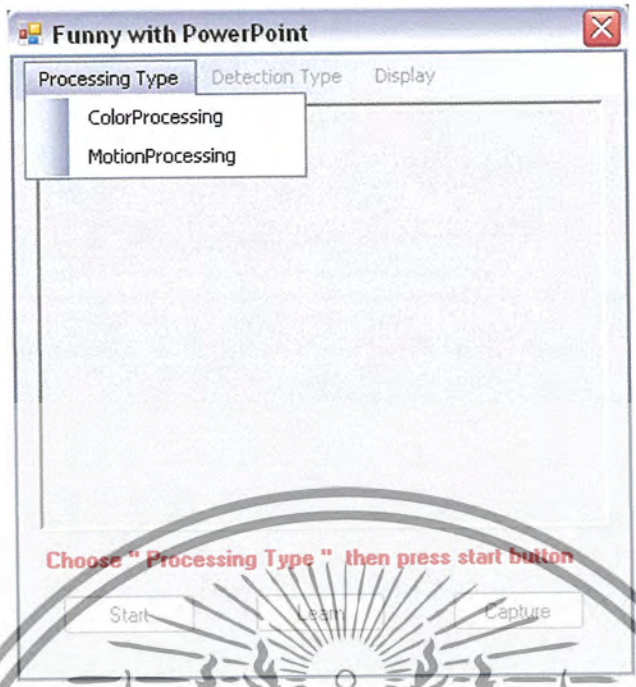
การใช้งาน Window Application

1) เปิดโปรแกรมไมโครซอฟท์เพาเวอร์พอยท์
2) รันโปรแกรม Funny with PowerPoint

รูปที่ ข.1 ตัวอย่างโปรแกรม

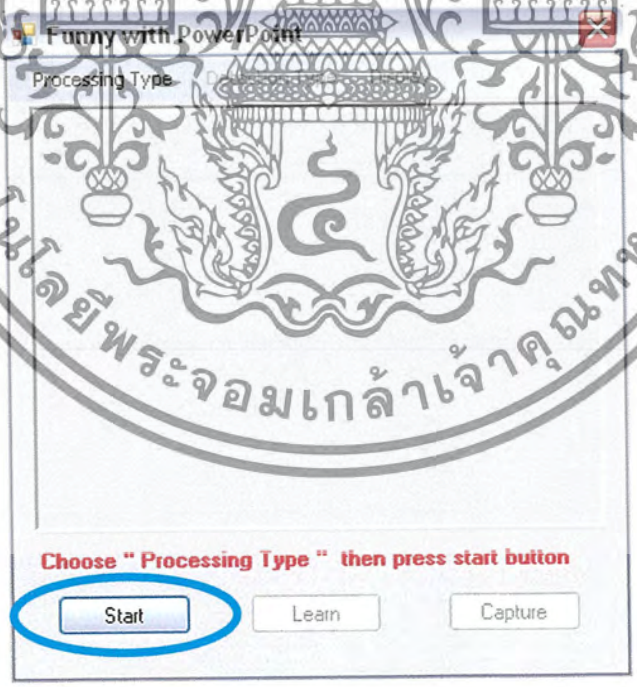
3) ทำการเลือกชนิดของการประมวลผลจากเมนู Processing Type มีให้เลือก 2 อย่าง คือ
   - ColorProcessing ซึ่งต้องใส่ถุงมือในการใช้งาน
   - MotionProcessing ไม่ต้องใส่ถุงมือในการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.2 ตัวเลือกที่อยู่ในเมนู Processing Type

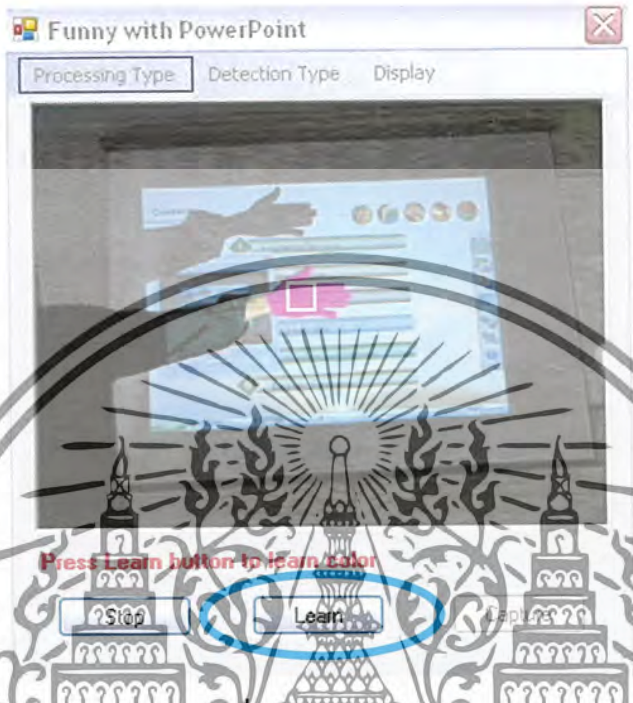
4) กดปุ่ม START และปรับตำแหน่งของมุมกล้องให้เหมาะสม



รูปที่ ข.3 ภาพปุ่ม START

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

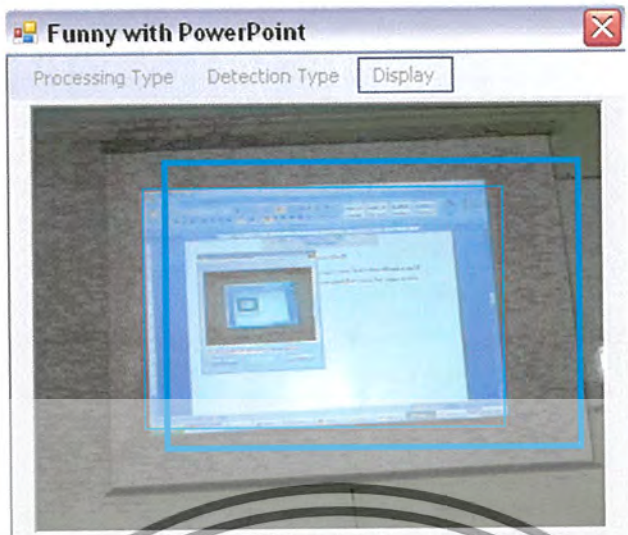
5) ถ้าเลือก ColorProcessing จะต้องทำการเรียนรู้สีของถุงมือก่อน โดยการคลิกเมาส์ซ้ายและลากจนครอบคลุมส่วนที่ต้องการเรียนรู้สีและปล่อยเมาส์ โดยจะต้องดูว่าสีที่อยู่ในกรอบนั้นจะต้องเป็นสีที่คล้ายกับสีตอนที่ใช้งานจริง จากนั้นกดปุ่ม Learn



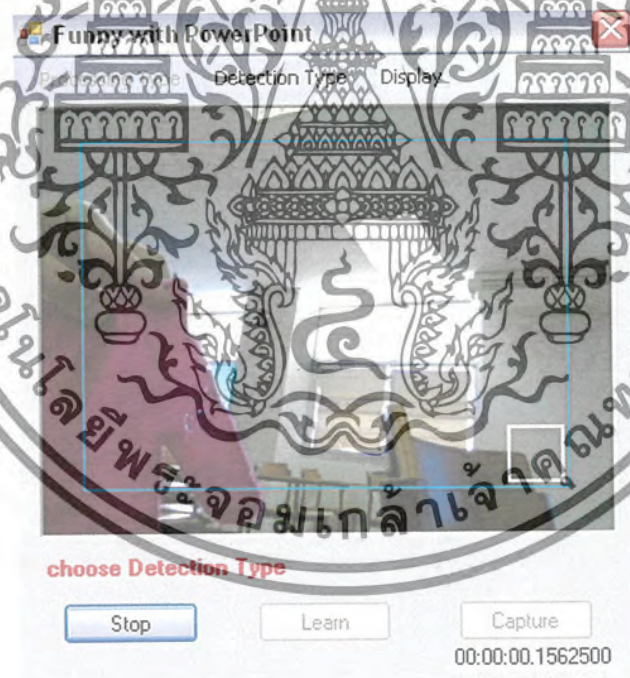
รูปที่ ง.4 การเรียนรู้สี

6) ทำการกำหนดพื้นที่ที่สนใจ (Area of interest) โดยการคลิกเมาส์ซ้ายและลากจนได้ขอบเขตที่กำหนดและปล่อยเมาส์ ถ้าไม่กำหนดจะถูกเซทเป็นค่า Default ซึ่งค่า Default จะเท่ากับขนาดของหน้าต่างที่เห็นในตัวโปรแกรม และกดปุ่ม Capture เพื่อนำภาพไปประมวลผล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

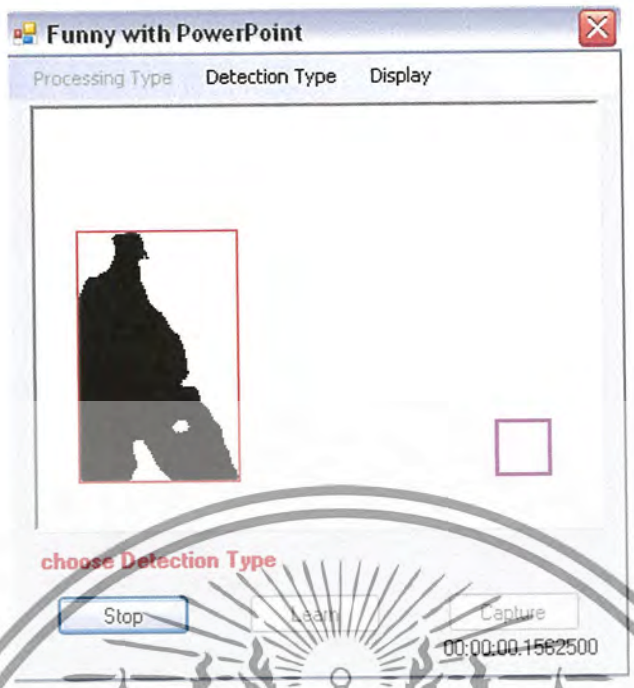


รูปที่ ข.5 การกำหนดพื้นที่ที่สนใจ



รูปที่ ข.6 ภาพในโหมดปกติ

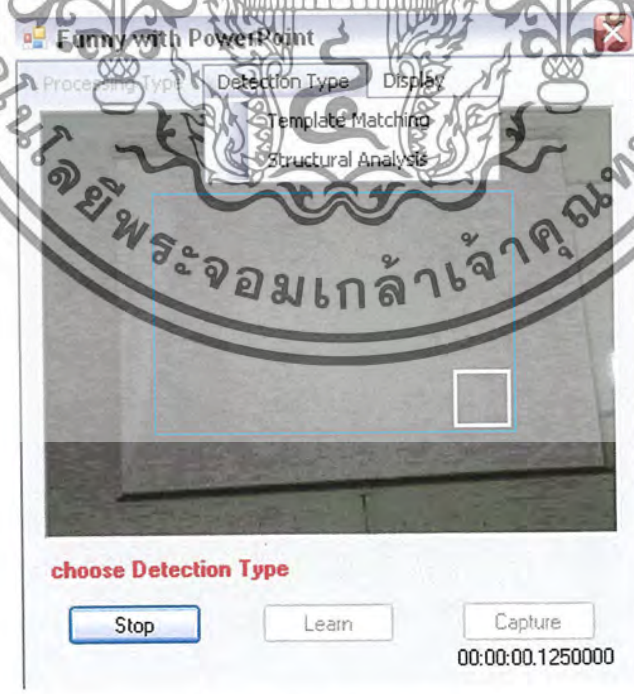
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ข.7 ภาพในโหมดรูปภาพขาว - ดำ

7) เลือกวิธีที่จะใช้ในการรู้จำจากเมนู Detection Type โดยมีให้เลือก 2 อย่าง คือ

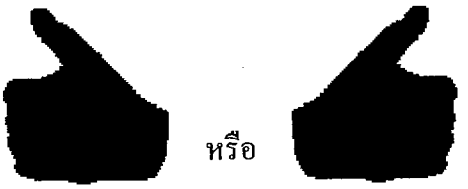
- Template Matching จะใช้วิธีการเข้าสู่รูปแบบในการรู้จำ
- Structural Analysis จะใช้วิธีการวิเคราะห์หาโครงสร้างในการรู้จำ



รูปที่ ข.8 เมนู Detection Type

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8) ทำมือในลักษณะที่กำหนดดังต่อไปนี้



หรือ แทนคำสั่งเริ่มต้นการนำเสนอ

หรือ แทนคำสั่งสิ้นสุดการนำเสนอ

แทนคำสั่งไปที่สไลด์หน้าถัดไป

แทนคำสั่งไปที่สไลด์หน้าก่อนหน้า

รูปที่ ข.9 ท่าทางของสัญญาณมือที่ใช้ในการออกคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9) ถ้าทำท่าได้ถูกต้องตามที่กำหนดไว้ จะมีเสียงเตือนดังขึ้น พร้อมทั้งจะมีรูปภาพปรากฏดังต่อไปนี้ ปรากฏที่มุมซ้ายล่างของหน้าจอโปรแกรม

รูปที่ ข.10 ผลการทำงานเมื่อออกคำสั่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10) สามารถเลือกเมนู Display ได้เมื่อต้องการดูการประมวลผลภาพแบบรูปภาพขาว - ดำ

รูปที่ ข.11 การประมวลผลรูปภาพขาว - ดำ

11) กดปุ่ม STOP เมื่อต้องการเลิกใช้งานโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้