สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

โปรแกรมฝึกสอนคาราเต้-โด

Karate-Do Training Program

ยุทธนา กล่ำพระบุตร
วิสุทธิ์ ศุภชัยโกศล

เลขหมู่.....
เลขทะเบียน 119274
วัน,เดือน,ปี - 6 S.A. 2554

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

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2553

สาขาวิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โปรแกรมฝึกสอนคาราเต้-โด

KARATE-DO TRAINING PROGRAM

ผู้จัดทำ

1. นายยุทธนา กล่ำพระบุตร รหัสนักศึกษา 50011252
2. นายวิสุทธิ์ ศุภชัยโกศล รหัสนักศึกษา 50011492

_____อาจารย์ที่ปรึกษา

(รศ. ดร.เกียรติกูล เจียรนัยธนะกิจ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรมฝึกสอนคาราเต้ – โด

นายยุทธนา กล่ำพระบุตร 50011252
นายวิสุทธิ์ ศุภชัยโกศล 50011492
รศ. ดร.เกียรติกูล เจียรนัยธนะกิจ อาจารย์ที่ปรึกษา
ปีการศึกษา 2553

บทคัดย่อ

ปัจจุบันได้มีการประยุกต์ใช้เทคโนโลยีกับการออกกำลังกายเป็นจำนวนมาก ไม่ว่าจะเป็นเครื่องออกกำลังกายต่าง ๆ ในฟิตเนส หรือเกมต่าง ๆ ในรูปแบบของการออกกำลังกายก็ตาม เทคโนโลยีพวกนี้สามารถทำให้ประชาชนหลาย ๆ กลุ่มที่ไม่มีเวลา หรือไม่สามารถออกไปออกกำลังภายนอกได้ โครงงานนี้ได้นำเทคโนโลยีที่มีความเกี่ยวข้องกับการจับการเคลื่อนไหวของมนุษย์มาประยุกต์ใช้กับโปรแกรมภายในคอมพิวเตอร์ ซึ่งวิธีนี้เป็นวิธีทำให้ผู้เล่นสามารถเคลื่อนไหวและออกท่าทางต่าง ๆ ให้สอดคล้องกับภาพวิดีโอตัวอย่างที่เปิดให้ดู และสามารถที่จะเรียนรู้ศิลปะป้องกันตัวคาราเต้ – โด ได้ภายในเวลาเดียวกัน ขอบเขตของโครงงานฉบับนี้จะทำการศึกษาและพัฒนาโปรแกรมการเคลื่อนไหวของศิลปะป้องกันตัวคาราเต้ – โด โดยจะมีวิธีการตรวจจับการเคลื่อนไหวของผู้เล่นผ่านกล้องเวบแคมทั้งหมด และส่งข้อมูลไปยังคอมพิวเตอร์เพื่อประมวลผลผ่านกระบวนการทาง Image Processing และใช้ไลบรารีของภาษาซีพลัส พลัส ทำการคำนวณหาตำแหน่งการเคลื่อนไหวของผู้เล่น แล้วนำไปประมวลผลเพื่อหาความถูกต้องของท่าทางเมื่อเทียบกับภาพวีดีโอตัวอย่าง ซึ่งโปรแกรมจะมีการสอนตั้งแต่การขยับตัว เทคนิคการออกแรงต่างและท่าทางที่ใช้แข่งขันจริงของท่ารำคาราเต้ – โด ซึ่งจะทำให้ผู้เล่นนั้นมีความรู้สึกเหมือนได้อยู่ในห้องซ้อมจริง และได้เรียนรู้กับกีฬาชนิดนี้จริง ๆ อีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Karate-Do Training Program

Mr. Yuttana	Klamprabud	50011252
Mr. Wisut	Supachaikosol	50011492
Assoc.Prof.Dr.Kietikul	Jearanaitanakij	Advisor

Academic Year 2010

Abstract

Nowadays, a lot of technologies have been applied to exercise for many purposes, e.g., exercise and fitness improvements, and games in various different forms of exercise. Technology can help people who have no time to exercise outside. This project has the relevant technology to capture the movement of human by applying the computer program. This method allows players to move and gesticulate in accordance with various video samples. Then they are able to learn martial arts Karate-Do at the same time. Scope of this project is to study and develop the movements of martial arts Karate-Do. It detects the movement of players through a web cameras. All cameras send data to a computer for image processing. We use C + + library to calculate positions of movement of players and process to determine the correct posture of Karate-Do. Comparing with video examples, the player will be taught from the sample move which shows the real gesture of Karate – Do. This makes players feel the actual practice and learn about this sport really well even in the indoor exercise.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การทำปริญญานิพนธ์โปรแกรมฝึกสอนคาราเต้-โดนี้สามารถไปได้ด้วยดี ทั้งนี้ได้รับความช่วยเหลือ ให้คำปรึกษาทั้งทางด้านวิชาการและการปฏิบัติการ ตลอดจนช่วยเหลือสนับสนุนทางด้านต่างๆ จาก รศ.ดร. เกียรติกูล เจียรนัยธนะกิจ ทางคณะผู้จัดทำขอขอบคุณเป็นอย่างยิ่ง

ขอขอบคุณ สาขาวิชาวิศวกรรมคอมพิวเตอร์ ที่ได้ให้เงินทุนสนับสนุน ตลอดจนเอื้อเฟื้อสถานที่สำหรับศึกษาและค้นหาข้อมูล

ขอขอบคุณ ชุมนุมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ (C-Square) ที่ได้เอื้อเฟื้อสถานที่ตลอดจนให้คำปรึกษาต่างๆ

ขอขอบคุณครอบครัว รวมทั้งเพื่อนๆ น้องๆ ทุกคนที่ให้การสนับสนุน และให้กำลังใจในการทำปริญญานิพนธ์ครั้งนี้ ความดีของปริญญานิพนธ์โปรแกรมฝึกสอนคาราเต้-โดฉบับนี้ขอมอบแด่ทุกท่านที่เกี่ยวข้อง

ยุทธนา กล่ำพระบุตร
วิสุทธิ์ ศุภชัยโกศล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I
บทคัดย่อภาษาอังกฤษ.....II
กิตติกรรมประกาศ.....III
สารบัญ.....IV
สารบัญภาพ.....VI

บทที่ 1 บทนำ..... 1
1.1 ความเป็นมา..... 1
1.2 ความมุ่งหมายและวัตถุประสงค์..... 2
1.3 ประโยชน์ที่คาดว่าจะได้รับ..... 2
1.4 ขอบเขตของโครงงาน..... 2
1.5 ขั้นตอนการดำเนินงาน..... 3
1.6 ข้อจำกัดโปรแกรม..... 3

บทที่ 2 ทฤษฎีพื้นฐานที่ใช้ในโครงงาน..... 4
2.1 คาราเต้-โด..... 4
2.2 ทฤษฎีการประมวลผลภาพ..... 8
2.3 ทฤษฎีการตรวจจับการเคลื่อนไหว..... 13
2.4 OpenCV..... 15

บทที่ 3 การออกแบบและพัฒนา..... 17
3.1 ความต้องการของระบบ..... 17
3.2 ภาพรวมของระบบ..... 17
3.3 อัลกอริทึมและการออกแบบ..... 19

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 4 การทดลอง..... 27

4.1 การทดลองติดต่อ OpenCV เข้ากับ Microsoft Visual Studio 2008..... 27

4.2 การทดลองรับภาพเคลื่อนไหวจากกล้อง..... 28

4.3 การทดลองบันทึกวิดีโอจากกล้อง..... 30

4.4 การทดลองรับภาพจากกล้องพร้อมกันสามตัว..... 33

4.5 การทดลองจับการเคลื่อนไหวของภาพ..... 35

4.6 การทดลองทำมาร์กเกอร์..... 42

บทที่ 5 สรุป..... 43

5.1 สรุป..... 43

5.2 ปัญหาและอุปสรรค..... 43

5.3 แนวทางการแก้ไข..... 43

5.4 แนวทางการพัฒนาต่อ..... 44

บรรณานุกรม..... 45

ภาคผนวก ก คู่มือการติดตั้ง..... 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป **หน้า**

2.1 คาราเต้ (วิถีแห่งการใช้มือ) 4
2.2 คาราเต้-โด..... 5
2.3 การแข่งขันคาราเต้..... 6
2.4 การแข่งขันท่ารำ..... 6
2.5 อวัยวะต่าง ๆ ที่ใช้เป็นอาวุธในคาราเต้..... 7
2.6 การวิเคราะห์จุดภาพโดยวิธีกราดภาพ..... 9
2.7 รูปแบบ RGB..... 10
2.8 รูปแบบ HIS..... 11
2.9 ตัวอย่างของโปรแกรมที่พัฒนาโดย OpenCV..... 15
3.1 รูปแบบของระบบโดยรวม..... 18
3.2 หน้าต่างของโปรแกรม..... 18
3.3 ผังการทำงานของระบบ..... 19
3.4 หน้าการเช็คมาร์คเกอร์สองจุดที่มีสีเดียวกัน..... 21
3.5 หน้าแรกของโปรแกรม..... 22
3.6 หน้าต่าง Training Mode..... 23
3.7 หน้าต่าง Record Mode..... 24
3.8 หน้าต่าง Record Mode..... 24
3.7 หน้าการบันทึกท่ารำของผู้เล่น 25
4.1 หน้าต่าง Systems Properties..... 26
4.2 PATH ของ OpenCV\bin..... 27
4.3 หน้าต่าง option ของ Microsoft Visual Studio 2008..... 27
4.4 ภาพที่จับได้จากโปรแกรมทดลองรับภาพเคลื่อนไหว..... 30
4.5 ไฟล์ .avi ที่บันทึกได้จากการทดลองบันทึกวิดีโอ..... 32
4.6 ภาพจากกล้องที่ได้จากการทดลองรับภาพจากกล้องพร้อมกันสามตัว..... 35
4.7 ทดลองโปรแกรมจับการเคลื่อนไหวโดยการตรวจจับทิศทาง..... 41
4.8 ตำแหน่งการติดมาร์คเกอร์..... 42
ก.1 หน้าการเริ่มต้นการลง OpenCV..... 46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูป **หน้า**

ก.2 หน้าการยอมรับข้อตกลงของ OpenCV..... 47
ก.3 หน้าการแอด Add OpenCV..... 47
ก.4 หน้าการเลือกพื้นที่ ๆ ลง OpenCV..... 48
ก.5 หน้าความต้องการ Create shortcuts..... 48
ก.6 หน้าการเลือก Components..... 49
ก.7 หน้าการเริ่มต้นลงโปรแกรม..... 49
ก.8 หน้าการยอมรับข้อตกลง..... 50
ก.9 หน้าการเลือกพื้นที่ลงโปรแกรม..... 50
ก.10 หน้าการเลือกโปรแกรม..... 51
ก.11 หน้าการเลือก Create a desktop icon..... 51
ก.12 หน้าการลงโปรแกรม..... 52

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมา

ปัจจุบันคอมพิวเตอร์ได้เข้ามามีส่วนร่วมกับชีวิตประจำของผู้คนบนโลกมากขึ้นทุกวัน ๆ และผู้คนส่วนมากก็จะหมกมุ่นอยู่กับคอมพิวเตอร์ จนทำให้หลาย ๆ คนไม่มีเวลาในการออกกลังกายซึ่งเป็นสิ่งที่สำคัญของมนุษย์เรามาก ซึ่งปัจจุบันผู้คนส่วนใหญ่ประสบปัญหาร่างกายไม่แข็งแรง และเจ็บไข้ได้ป่วยเป็นจำนวนมาก ซึ่งเป็นผลมาจากการที่ผู้คนเหล่านั้นใช้ชีวิตอยู่กับคอมพิวเตอร์มากเกินไป สำหรับประเทศไทย สำนักงานสถิติ-แห่งชาติ ได้รายงานผลการสำรวจครั้งล่าสุดในปี 2550 ในกลุ่มประชาชนที่มีอายุตั้งแต่ 11 ปีขึ้นไป ซึ่งมีทั้งหมด 55 ล้านคน พบว่าในจำนวนนี้ออกกำลังกายร้อยละ 30 หรือประมาณ 16 ล้านกว่าคน เพิ่มขึ้นจากปี 2547 เล็กน้อย โดย 1 ใน 3 ออกกำลังกายสัปดาห์ละ 3-5 วัน ครั้งละ 21-30 นาที ส่วนใหญ่เกือบครึ่งเล่นกีฬา รองลงมาคือ เดิน วิ่ง เต้นแอโรบิก ฟิตเนต รำไม้พลอง โยคะ ไทเก็ก จี้กง ผู้ออกกำลังกายประมาณครึ่งหนึ่ง สำเร็จการศึกษาไม่เกินประถม ศึกษา และต่ำสุดคือระดับปริญญาตรี มีร้อยละ 14 ซึ่งทำให้เห็นว่ายิ่งมีการศึกษามากก็จะทำให้คนเหล่านั้นมองเห็นความสำคัญของการออกกำลังกายน้องลงไป

ปัจจุบันมีหน่วยงานหลาย ๆ หน่วยงานมองเห็นความสำคัญของการออกกำลังกาย ทั้งหน่วยงานของรัฐ และเอกชน แต่เนื่องด้วยปัญหาหลายอย่าง ทั้งในเรื่องของสถานที่ออกกำลังกาย เวลาการทำงานของคนในปัจจุบัน และสภาพแวดล้อมต่าง ๆ ทำให้ผู้คนส่วนใหญ่มีมีแรงจูงใจในการออกกำลังกาย ทำให้บุคคลเหล่านี้มองข้ามการออกำลังกายไป ซึ่งจากการสำรวจในปี 2550 พบประชาชนอายุ 11 ปีขึ้นไป ป่วย และเข้ารักษาตัวที่แผนกผู้ป่วยนอกเพียง 9 ล้านกว่าคน หรือป่วยเพียงร้อยละ 17 ในจำนวนนี้เป็นผู้ที่ไม่ได้ออกกำลังกายเกือบร้อยละ 70 ส่วนกลุ่มผู้ป่วยที่ต้องพักรักษาตัวในโรงพยาบาลมีจำนวน 3 ล้านกว่าคน ในจำนวนนี้เป็นผู้ที่ไม่ออกกำลังกายร้อยละ 74 เป็นข้อพิสูจน์ที่ชัดเจนว่า การออกกำลังกายทำให้สามารถลดการป่วยลงได้

จากปัญหาดังกล่าวทีมพัฒนาจึงมองเห็นความสำคัญของการออกกำลังการ และปัญหาต่าง ๆ ที่ทำให้ประชาชนส่วนใหญ่ไม่ออกกำลังกาย ซึ่งการออกกำลังกายสามารถทำได้หลากหลายวิธี เช่น การเดิน วิ่ง เต้น ตลอดจนกีฬาประเภทต่อสู้ ซึ่งคาราเต้เป็นกีฬาที่สามารถใช้ร่างกายได้ทุกส่วน ตั้งแต่การกำหนดลมหายใจจนถึงการออกแรงของทุกส่วนของร่างกาย ซึ่งจะทำให้ผู้เล่นออกกำลังกายได้ทุกส่วนของร่างกาย สามารถเล่นได้ทั้งเด็กและผู้ใหญ่ และใช้พื้นที่เพียงเล็กน้อยในการเล่น

ดังนั้นคาราเต้-โด จึงเป็นกีฬาอีกชนิดหนึ่งที่น่าสนใจ มีความท้าทายความสามารถของผู้เล่น และกีฬานี้เป็นกีฬาที่หลาย ๆ คนอาจจะยังไม่รู้จัก ซึ่งการเรียนคาราเต้นั้น จะต้องทำการเรียนตามโรงยิมต่าง ๆ ที่มีการจัดสอนซึ่งมีอยู่ไม่มากในประเทศไทย ทำให้หลาย ๆ คนไม่รู้จักคาราเต้-โด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางทีมงานพัฒนาจึงมองเห็นความสำคัญของปัญหาดังกล่าว จึงได้จัดทำการพัฒนาแอปพลิเคชั่นนี้เพื่อเป็นทางเลือกของการออกกำลังของประชาชนทั่วไป และเป็นการประชาสัมพันธ์กีฬาชนิดนี้ให้กับประชาชนที่ยังไม่รู้จัก ได้ศึกษาศิลปะป้องกันตัวชนิดนี้

1.2 ความมุ่งหมายและวัตถุประสงค์

1) เพื่อศึกษาทฤษฎีการประมวลผลภาพ (Image Processing) ในรูปแบบต่างต่าง ๆ รวมทั้งการหาตำแหน่งของความแตกต่างของพิกเซลที่ได้จาการเปรียบเทียบภาพจากการเคลื่อนไหว
2) เพื่อศึกษาการจับภาพการเคลื่อนไหวของกล้องเว็บแคมทั้ง 3 ตัว
3) เพื่อพัฒนาแอปพลิเคชั่นโปรแกรมให้มีการตอบสนองกับการเคลื่อนไหวผู้เล่นให้สมจริงมากที่สุด
4) เพื่อพัฒนาแอปพลิเคชั่นโปรแกรมที่ส่งเสริมการออกกำลังกาย
5) เพื่อเป็นการประชาสัมพันธ์กีฬาคาราเต้โด ให้เป็นที่รู้จักมากขึ้น

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1) ผู้พัฒนาสามารถพัฒนาแอปพลิเคชั่นที่ตอบสนองการเคลื่อนไหวของผู้เล่นได้
2) ผู้พัฒนาสามารถนำความรู้ที่ใช้ในการพัฒนาแอปพลิเคชั่นนี้ไปใช้ต่อไปในชีวิตประจำวันต่อไปได้
3) สามารถทำให้ผู้ใช้แอปพลิเคชั่นนี้มีความรู้สึกเหมือนกับการได้ฝึกซ้อมคาราเต้โดได้จริง ๆ
4) ผู้ใช้แอปพลิเคชั่นมีความพอใจ และใช้ประโยชน์ได้สูงสุดกับแอปพลิเคชั่นนี้
5) สามารถทำให้ประชาชนทั่วไปรู้จัก และเข้าใจเกี่ยวกับคาราเต้โดมากขึ้น

1.4 ขอบเขตของโครงงาน

สำหรับการพัฒนาแอปพลิเคชั่นนี้มุ่งเน้นที่การจับภาพการเคลื่อนไหวของผู้เล่นจากกล้องเว็บแคม แล้วทำการประมวลผลด้วยกระบวนกานทางอิมเมจโปรเซสซึ่ง ซึ่งในที่นี้ทีมงานพัฒนาแอปพลิเคชั่นโปรแกรม ได้ออกแบบแอปพลิเคชั่นเพื่อให้ผู้เล่นสามารถเคลื่อนไหวได้ทุกส่วนของร่างกายตั้งแต่ปลายเท้าจนถึงหัวศีรษะ ซึ่งแอปพลิเคชั่นจะค่อนข้างเสมือนจริงเพราะว่าผู้เล่นจะได้ศึกษาวีดีโอตัวอย่างที่เปิดให้ดู เมื่อผู้เล่นสามารถเล่นได้ในส่วนเริ่มต้น ก็จะมีการพัฒนาให้มีความยากเพิ่มขึ้นต่อไปเรื่อย ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับแอปพลิเคชั่นนี้เมื่อสิ้นสุดการพัฒนาแล้วจะสามารถเล่นได้ทั้งผู้ที่ไม่เคยรู้จักกับกีฬาคาราเต้โดมาก่อน หรือผู้ที่เคยได้ศึกษากีฬาคาราเต้-โดมาบ้างแล้ว สามารถเล่นได้ทุกเพศทุกวัย และเหมาะกับบุคคลที่รักในศิลปะป้องกันตัว และบุคคลที่ไม่มีเวลาที่ไปออกกำลังกายข้างนอกได้

1.5 ขั้นตอนการดำเนินงาน

1) กำหนดจุดประสงค์ในการดำเนินงาน
2) ศึกษาวางแผนในการดำเนินงาน
3) แบ่งหน้าที่ให้สมาชิกภายในทีมงานพัฒนาแอปพลิเคชั่น
4) สมาชิกแต่ละคนศึกษาหาความรู้ของงาน และทำงานที่รับผิดชอบ
5) สมาชิกทุกคนประชุม เพื่อบอกถึงปัญหาของแต่ละคนที่เกิดขึ้น และช่วยกันหาทางแก้ปัญหานั้น ๆ และเพื่อบอกถึงความคืบหน้าของงานที่รับผิดชอบ และบอกถึงงานชิ้นใหม่ที่ต้องรับผิดชอบ

1.6 ข้อจำกัดโปรแกรม

โปรแกรมฝึกซ้อมคาราเต้โดเป็นโปแกรมที่เน้นการสอนท่ารำของคาราเต้โด ภายในตัวแอปพลิเคชั่นนั้นจะสอนตั้งแต่ขั้นพื้นฐานของคาราเต้โด ตั้งแต่วิธีการออกแรง วิธีการออกอาวุธต่างๆ วิธีการปัด และพัฒนามาเป็นการท่ารำตั้งแต่ท่าเริ่มแรกของคาราเต้โดสายโกจูริว จนถึงท่าสูงสุดของคาราเต้โดสายโกจูริว ซึ่งภายในแอปพลิเคชั่นจะมีทั้งภาพวิดีโอสอน ซึ่งจะทำให้ผู้เล่นแอปพลิเคชั่นนี้เข้าใจ และสามารถที่จะฝึกซ้อมคาราเต้โดเองได้ โดยที่ไม่ต้องเสียเวลาออกมาฝึกซ้อมตามสถานที่ฝึกซ้อมต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
ทฤษฎีพื้นฐานที่ใช้ในโครงงาน

2.1 คาราเต้–โด

2.1.1 ประวัติความเป็นมาของคาราเต้–โด

คาราเต้ (「空手」, karate, – คะระเตะ ?) หรือ คาราเต้-โด (「空手道」, karatedo, คะระเตะโด , วิถีมือเปล่า) เป็นศิลปะการต่อสู้ถือกำเนิดที่โอกินาวา ประเทศญี่ปุ่น เป็นการผสมผสานระหว่างการต่อสู้ของชาวโอกินาวาและชาวจีน คาราเต้ได้เผยแพร่เข้าสู่ญี่ปุ่นในปี พ.ศ. 2464 (ค.ศ. 1921) เมื่อชาวโอกินาวาอพยพเข้าสู่ประเทศญี่ปุ่น

คาราเต้มักถูกเข้าใจผิดว่า เป็นการต่อสู้ด้วยการฟันอิฐ แต่ที่จริงแล้ว คือการต่อสู้ด้วยการใช้อวัยวะต่างๆ ในร่างกาย เช่น กำปั้น เท้า สันมือ นิ้ว ศอก เป็นต้น แต่เมื่อถูกดัดแปลงเป็นกีฬาแล้วเหลือเพียงมือและเท้า

空手道

KARA TE DO
Empty Hand Way

รูป 2.1 คาราเต้ (วิถีแห่งการใช้มือ)

คาราเต้ แปลว่า วิถีแห่งการใช้มือ (ร่างกาย) ต่อสู้โดยปราศจากอาวุธ วิถีแห่งคาราเต้เป็นวิธีการดึงพลังจากทั้งร่างมารวมให้เป็นหนึ่งในการต่อสู้โจมตี ซึ่งความรุนแรงของการโจมตีนั้นมีคำกล่าวถึงว่า "อิคเคน ฮิซัทสึ"(一拳必殺) หรือ "พิชิตในหมัดเดียว" สิ่งที่สำคัญของคาราเต้คือการต่อสู้กับตนเอง เช่นการฝึกยั้งแรงการโจมตี โดยใช้ในการหยุดโจมตีเมื่อสัมผัสร่างกายคู่ต่อสู้แม้เพียงเล็กน้อย เพื่อให้เกิดความรู้สึกเจ็บไม่มากและป้องกันการบาดเจ็บ ซึ่งเป็นการฝึกการกำหนดความรุนแรงของการโจมตี เมื่อผู้ฝึกสามารถยั้งแรงได้ เขาก็จะเพิ่มความรุนแรงในการโจมตีได้จนถึงขีดความสามารถเช่นเดียวกัน

คำว่า โด แปลว่า วิถีทาง ลู่ทาง ศาสตร์ อีกทั้งยังหมายถึงปรัชญาเต๋าอีกด้วย โด เป็นคำต่อท้ายที่ใช้สำหรับศิลปะหลายชนิด ให้ความหมายว่า นอกจากจะศิลปะเหล่านั้นจะเป็นทักษะแล้วยังต้องมีพื้นฐานของจิตวิญญาณอยู่ด้วย สำหรับในความหมายที่เกี่ยวข้องกับศิลปะการต่อสู้ อาจจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แปลได้ว่า "วิถีแห่ง..." เช่น ใน ไอคิโด ยูโด เคนโด ดังนั้น "คาราเต้-โด" จึงหมายถึง "วิถีแห่งมือเปล่า" "โด" อาจมองได้ 2 แบบ คือ แบบปรัชญา และแบบกีฬา เมื่อมีการบรรจุเป็นกีฬาต่อสู้จึงใช้คำว่าคาราเต้-โดเช่นเดียวกัน

คาราเต้ แต่เดิมไม่มีคำว่าโด เช่นกัน แต่ก่อนจะเรียกว่า คาราเต้จิทสุ หรือว่า คาราเต้ แต่เริ่มใช้คำว่า "โด" เมื่อมีการจัดการแข่งขันชิงชนะเลิศ ซึ่งต้องรวมนักคาราเต้จากทั้ง 4 สำนักใหญ่เข้าไว้จึงต้องบัญญัติกฎการแข่งขันใหม่ ลดทอนการจู่โจมที่อันตราย และสามารถแข่งขันกันได้อย่างเต็มที่และเป็นกลางที่สุด คำว่า "โด" ในคาราเต้จึงเกิดขึ้น และมีความหมายว่า วิถีทางการต่อสู้ในรูปแบบของคาราเต้ ซึ่งคำว่าคาราเต้-โด โดยมากจะใช้ในการแข่งขัน

รูป 2.2 คาราเต้-โด

2.1.2 กฎกติกาคาราเต้-โดเบื้องต้นและการนับคะแนน

การแข่งขันคาราเต้-โดจะแบ่งออกเป็น 2 ฝั่ง คือ ฝั่งแดง และฝั่งน้ำเงิน และจะแบ่งออกเป็น 2 ประเภท คือ ต่อสู้ (kumite) ท่ารำ (kata) ซึ่งการต่อสู้จะนับแต้มจากการชกและการเตะ บริเวณศีรษะ หน้า คอ ท้อง อก หลัง และข้างลำตัว และจากการทำให้คู่ต่อสู้เสียการทรงตัวและทำแต้มซ้ำจากการชกหรือเตะ แต่คาราเต้-โดได้ลดความรุนแรงจากการจู่โจมทำแต้ม เพื่อไม่ให้เกิดอันตรายในการแข่งขัน

ในการแข่งขันจริงจะมีกรรมการกลางสนามหนึ่งคนเพื่อเป็นคนตัดสินและให้คะแนน และจะมีกรรมการยกธงให้คะแนนอีก 3 คนอยู่ข้างสนาม ซึ่งการตัดสินใจให้คะแนนจะเป็นหน้าที่ของกรรมการกลาง หลักการให้แต้มจะมีด้วยกันทั้งสิ้น 6 อย่างคือ

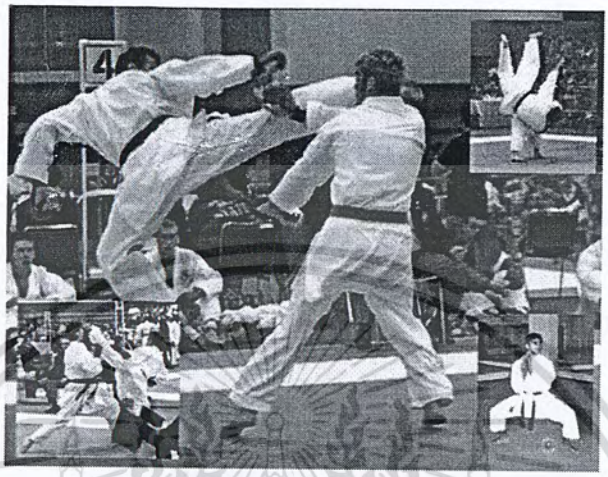
1) Good form : การจู่โจมด้วยท่าทางที่ดี
2) Sporting Attitude : ทัศนคติท่าทางที่ดี
3) Vigorous Application : การใช้พลังและความเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4) Awareness (Zanshin) : การระวังการจู่โจมกลับ

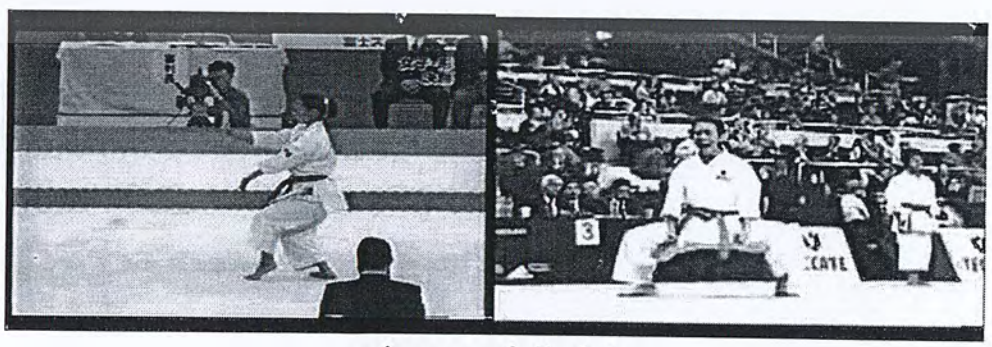
5) Good Timing : จังหวะการจู่โจม

6) Correct Distance : ระยะของการจู่โจม



รูป 2.3 การแข่งขันคาราเต้

ส่วนการแข่งขันท่ารำ (kata) ท่ารำของคาราเต้จะมีอยู่หลายสำนัก อย่างเช่น สำนักโกจูริว ชิโตริว โชโตกัน เป็นต้น แต่สำหรับในแอปพลิเคชั่นนี้จะใช้ท่ารำของสำนักโกจูริว ซึ่งท่ารำของสำนักโกจูริวจะค่อนข้างยากเพราะเบสิกของโกจูริวนั้นยากกว่าสำนักอื่น การแข่งขันท่ารำเป็นการแข่งขันที่ต้องใช้สายตาตัดสิน ซึ่งกรรมการตัดสินจะดูจากเบสิกของท่ารำ แรงที่แสดงออกมา จังหวะ ท่าทางและความสวยงามของท่ารำ ผู้ที่จะรำท่ารำคาราเต้ให้สวยงามได้นั้น ต้องใช้เวลาในการฝึกซ้อมที่ค่อนข้างนาน เพราะว่าท่ารำทุกท่าของคาราเต้นั้นมีความหมายในการใช้ต่อสู้หรือป้องกันตัวจริง ผู้ที่ฝึกซ้อมต้องเรียนรู้ทั้งเบสิค และความหมายของท่ารำแต่ละท่าด้วยว่ามีความหมายเช่นไร



รูป 2.4 การแข่งขันท่ารำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในการแข่งขันท่ารำจริง ๆ นั้น จะใช้กรรมการทั้งหมด 5 คนในการตัดสิน โดยกรรมการทุกคนจะมีธงสีแดง และสีน้ำเงิน คนละหนึ่งคู่ เมื่อผู้แข่งขันรำเสร็จก็จะยกธงเพื่อ บอกว่าฝ่ายไหนเป็นฝ่ายชนะ และเมื่อยกธงเสร็จก็จะนับผลรวมว่าธงสีใดมากกว่ากัน ฝ่ายนั้นก็จะเป็นฝ่ายชนะไป

2.1.3 อวัยวะต่างๆที่ใช้เป็นอาวุธในคาราเต้

คาราเต้ เป็นศิลปะป้องกันตัวด้วยมือเปล่า อวัยวะต่าง ๆ ของร่างกายจำเป็นต้องมีความแข็งแรงในทุกส่วน เพราะหลาย ๆ ส่วนในร่างกายล้วนเป็นทั้งอาวุธที่อันตราย และจุดที่อ่อนที่สุดเช่นเดียวกัน คาราเต้จะเน้นการโจมตีที่อันตราย และใช้ความเร็วในการโจมตี อย่างที่กล่าวไว้แล้วว่าคาราเต้ถูกฝึกมาเพื่อต่อสู้กับคู่ต่อสู้ที่มากกว่าและมีอาวุธ ทำให้คาราเต้ต้องอาศัยความเร็วในการต่อสู้ และโจมตีเฉพาะจุดที่อันตราย จนมีผู้กล่าวว่าการโจมตีของคาราเต้เป็นการโจมตีที่ครั้งเดียวจะต้องทำให้คู่ต่อสู้เสียชีวิตทันที อวัยวะต่าง ๆ ที่ใช้เป็นอาวุธในคาราเต้นั้นมีหลายส่วน

รูป 2.5 อวัยวะต่าง ๆ ที่ใช้เป็นอาวุธในคาราเต้

จากการศึกษาศิลปะป้องกันตัวคาราเต้ ทราบว่าหลักการออกแรงต่าง ๆ ทั้งการชก เตะ ปัด และอาวุธต่าง ๆ นั้น แรงที่ใช้จริง ๆ ไม่ได้มาจากแขนหรือขาเพียงอย่างเดียว แต่จะอาศัยการออกแรงจากท้อง ซึ่งเป็นจุดหลักของการระเบิดแรง หรือการออกแรง และอาศัยการบิดสะโพกหรือการบิดเอวในการออกอาวุธต่าง ๆ ซึ่งถ้าผู้เล่นออกแรงผิดส่วนก็จะเปรียบเสมือนการที่ผู้เล่นออกแรงโดยเสียเปล่า ซึ่งการออกแรงจากหน้าท้องและการบิดสะโพก หรือเอวนั้นเป็นจุดที่สำคัญที่สุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ของการออกอาวุธหนึ่งครั้ง และนอกจากการออกแรงจากหน้าท้องและการบิดสะโพก จุดสำคัญของการออกอาวุธอีกอย่างหนึ่งนั้นก็คือการหายใจนั้นเอง การชก หรือการปัดหนึ่งครั้งเราจะ ต้องหายใจออก เพื่อให้แรงนั้นออกมาได้เต็มที่ ผู้เล่นจะต้องกำหนดลมหายใจให้ได้ การหายมีผลต่อการออกแรงอย่างมาก ซึ่งการหายใจเข้า ออก ให้ลมหายใจสั้นหรือยาวนั้นมีผลกับการออกอาวุธและการป้องกันตัวทั้งสิ้น อย่างเช่น การชกออกไปเราจะต้องหายใจออก และอาศัยการกำหนดลมหายใจที่สั้น เพื่อที่จะให้หมัดนั้นออกไปด้วยความเร็วตามที่ต้องการ ซึ่งถ้าหากเรากำหนดหายใจออกแบบยาว หมัดที่ออกไปก็จะมีความเร็วช้ากว่าหมัดที่ออกไปด้วยลมหายใจสั้น เป็นต้น นี้เป็นเพียงส่วนหนึ่งของศิลปะป้องกันตัวคาราเต้โด ซึ่งเป็นศิลปะป้องกันตัวที่น่าสนใจไม่แพ้กับศิลปะป้องกันตัวประเภทอื่น ๆ เช่นกัน

2.2 ทฤษฎีการประมวลผลภาพ

เมื่อโปรแกรมได้รับภาพจากเว็บแคม จะทำการตรวจสอบการเคลื่อนไหวโดยการใช้เทคนิคของการประมวลผลภาพเข้ามาช่วยในการทำงาน กล่าวคือจะทำการหามาร์คเกอร์ตามตัวของผู้เล่นจากภาพที่ได้รับเข้ามา ซึ่งจะเป็นลักษณะของจุดสี การทำงานคือ เมื่อได้รับภาพเข้ามาจากเว็บแคมแล้วจะทำการประมวลผลตำแหน่งของมาร์คเกอร์ โดยมาร์คเกอร์นั้นประมาณถึงอวัยวะแต่ละจุดว่าอยู่ประมาณตำแหน่งไหนแล้วนำไปทำการเปรียบเทียบต่อ

การประมวลผลภาพ (Image Processing) คือ การนำภาพมาประมวลผลหรือคิดคำนวณด้วยเครื่องคอมพิวเตอร์ เพื่อให้ได้ข้อมูลในเชิงคุณภาพและปริมาณ ซึ่งเป็นการประยุกต์ใช้งานการประมวลผลสัญญาณบนสัญญาณ 2 มิติ เช่น ภาพนิ่ง ภาพวีดีทัศน์ ซึ่งจะเป็นชุดของภาพนิ่ง เรียกว่าเฟรม (frame) หลายๆ ภาพต่อกันไปตามเวลา นอกจากนี้ยังรวมถึงสัญญาณ 2 มิติอื่นๆ ที่ไม่ใช่ภาพด้วย

การประมวลผลภาพมีขั้นตอนที่สำคัญๆ คือ การทำให้ภาพมีความคมชัดมากยิ่งขึ้น การกำจัดสัญญาณรบกวนออกจากภาพ การแบ่งส่วนวัตถุที่เราสนใจออกมาจากภาพ เพื่อนำภาพวัตถุที่ได้ไปวิเคราะห์ข้อมูลเชิงปริมาณ เช่น ขนาด รูปร่าง และทิศทางการเคลื่อนไหวของวัตถุในภาพ ซึ่งข้อมูลเชิงปริมาณสามารถนำไปวิเคราะห์และสร้างเป็นระบบ เพื่อใช้ประโยชน์ในงานด้านต่างๆ

เมื่อหลายสิบปีมาแล้ว การประมวลผลภาพนั้นจะอยู่ในรูปของการประมวลผลสัญญาณอนาล็อก (Analog) โดยใช้อุปกรณ์ปรับแต่งแสง (Optics) แต่เนื่องจากอุปกรณ์คอมพิวเตอร์ในปัจจุบัน มีราคาถูกลง และมีความเร็วในการประมวลผลสูง การประมวลผลภาพดิจิตอล (Digital Image Processing) จึงได้รับความนิยมมากกว่า ทฤษฎีการประมวลผลภาพนำมาใช้ในการหาตำแหน่ง อ้างอิงของมาร์คเกอร์ (Marker) จากภาพที่ได้จากการบันทึกด้วยกล้องวีดีโอ เพื่อนำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่ง อ้างอิงของมาร์คเกอร์ ไปคำนวณหาตำแหน่งการเคลื่อนไหวของตัวละครต่อไป โดยทฤษฎีที่เกี่ยวข้องกับการประมวลผลภาพมีดังต่อไปนี้

2.2.1 จุดภาพ (Pixels)

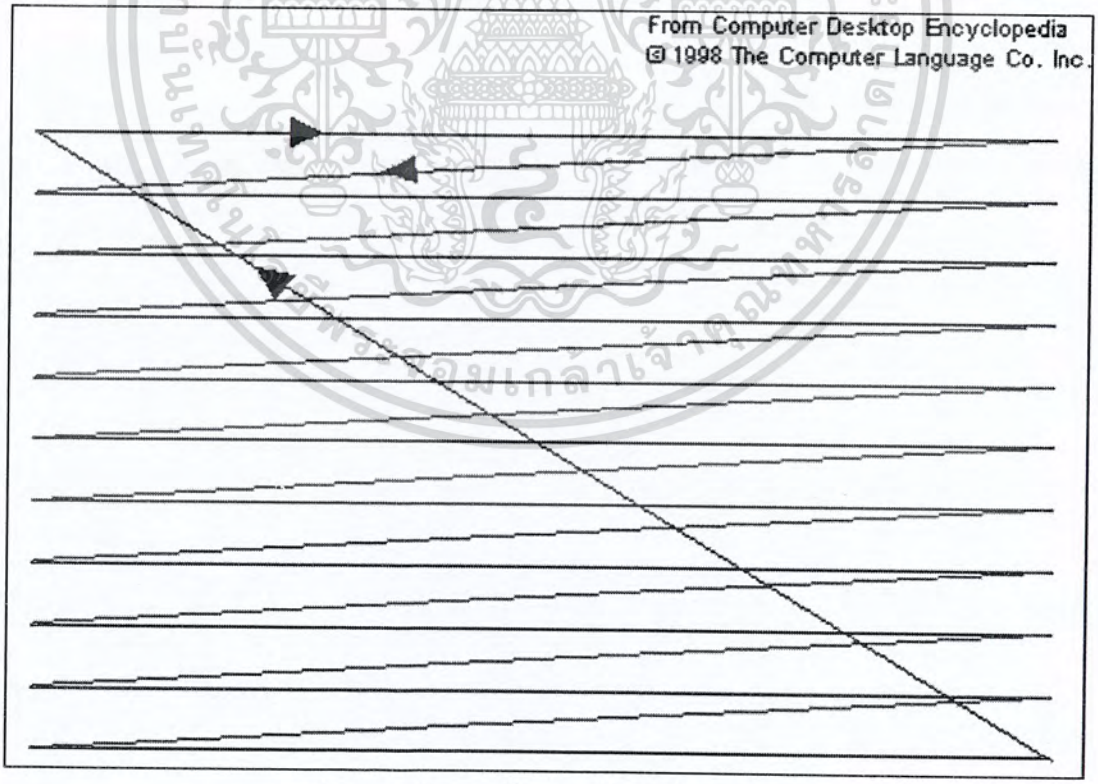
ภาพที่ได้จากการบันทึกด้วยกล้องวิดีโอเป็นภาพกราฟิกแบบราสเตอร์ (Raster Graphics) หรือ ภาพบิทแมป (Bitmap) โดยภาพเกิดจากจุดภาพ (pixels) หลายๆจุดประกอบรวมกันเป็นภาพ โดยจุดภาพแต่ละจุดจะเก็บค่าสี โดยค่าสีเหล่านี้จะทำให้เกิดรายละเอียดในภาพเป็นสีต่างๆ

2.2.2 ระบบอ้างอิงพิกัดของจุดภาพ (Pixel Coordinate System)

เป็นระบบพิกัดที่ใช้อ้างอิงตำแหน่งของจุดพิกเซลในภาพ ซึ่งมีจุดเริ่มต้นอยู่ที่มุมซ้ายบนของภาพ และจุดสิ้นสุดอยู่ที่มุมขวาล่างของภาพ โดยการอ้างอิงตำแหน่งจุดพิกเซลในภาพจะเป็นคู่ลำดับ (x , y) โดย x เป็นตำแหน่งพิกเซลในแนวแกนนอน y เป็นตำแหนงอ้างอิงในแนวแกนตั้ง

2.2.3 การวิเคราะห์จุดภาพโดยวิธีกราดภาพ (Raster Scan)

การวิเคราะห์รายละเอียดในภาพเป็นการวิเคราะห์ภาพในระดับพิกเซลโดยใช้หลักการการกราดภาพโดยวิเคราะห์จุดพิกเซลในภาพจากซ้ายไปขวา จนถึงจุดสิ้นสุดในแนวนอน หลังจากนั้นจะลงไปหนึ่งจุดแล้วทำการไล่จุดภาพจากซ้ายไปขวาจนกว่าจะครบทั้งภาพ



รูป 2.6 การวิเคราะห์จุดภาพโดยวิธีกราดภาพ

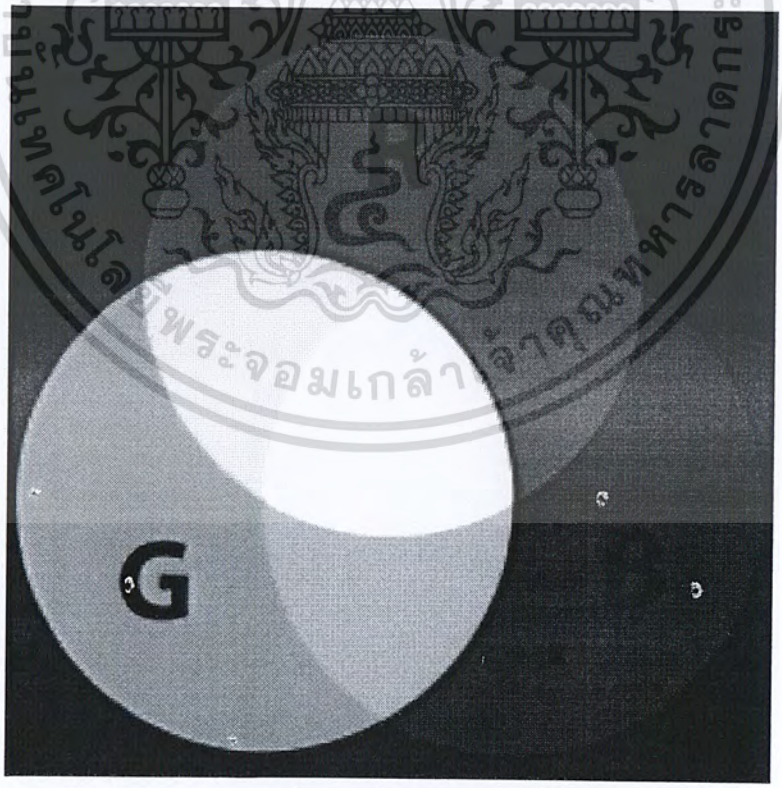
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.4 รูปแบบสี (Color Model)

สี คือ การรับรู้ความถี่หรือความยาวคลื่นของแสง โดยสีเกิดจากแสงที่มากระทบวัตถุแล้วสะท้อนเข้าตาเราทำให้เห็นเป็นสีต่างๆ การที่เรามองเห็นวัตถุเป็นสีใดๆ ได้ เพราะวัตถุนั้นดูดแสงสีอื่นและสะท้อนแต่สีของมันเอง เช่น วัตถุสีแดง เมื่อมีแสงส่องกระทบก็จะ ดูดทุกสีและสะท้อนแต่สีแดงทำให้เรามองเห็นเป็นสีแดง การคัดแยกมาร์คเกอร์ออกจากภาพสามารถใช้คุณสมบัติของสีโดยมาร์คเกอร์จะมีสีที่แตกต่างไปจากพื้นหลังของภาพเพื่อให้การหาตำแหน่งของมาร์คเกอร์จากภาพสามารถอ้างอิงได้จากตำแหน่งพิกเซลซึ่งเก็บค่าสีที่เป็นสีของมาร์คเกอร์ในภาพการประมวลผลภาพสีนั้นต้องทราบรูปแบบของสีที่ใช้ในคอมพิวเตอร์ ซึ่งจะอธิบายรูปแบบที่สำคัญ 2 แบบ ได้แก่

2.2.5 รูปแบบ RGB

เป็นระบบสีพื้นฐานของคอมพิวเตอร์ที่ใช้ในการแสดงผล โดยจุดพิกเซล 1 จุดจะประกอบด้วยค่าสี 3 ค่า คือ แดง (R) เขียว (G) และน้ำเงิน (B) การผสมสีทั้งสามนี้ด้วยค่าต่างๆ กันจะก่อให้เกิดสีที่แตกต่างกัน คอมพิวเตอร์จะเก็บค่าสีนี้แยกกันโดยใช้ขนาดข้อมูล 1 ไบต์ต่อ 1 สี ทำให้ค่าของสีแต่ละสีนั้นมีได้ 256 ระดับ ($0 \leq R \leq 255$), ($0 \leq G \leq 255$), ($0 \leq B \leq 255$) และผสมได้สีทั้งหมด 16 ล้านสี

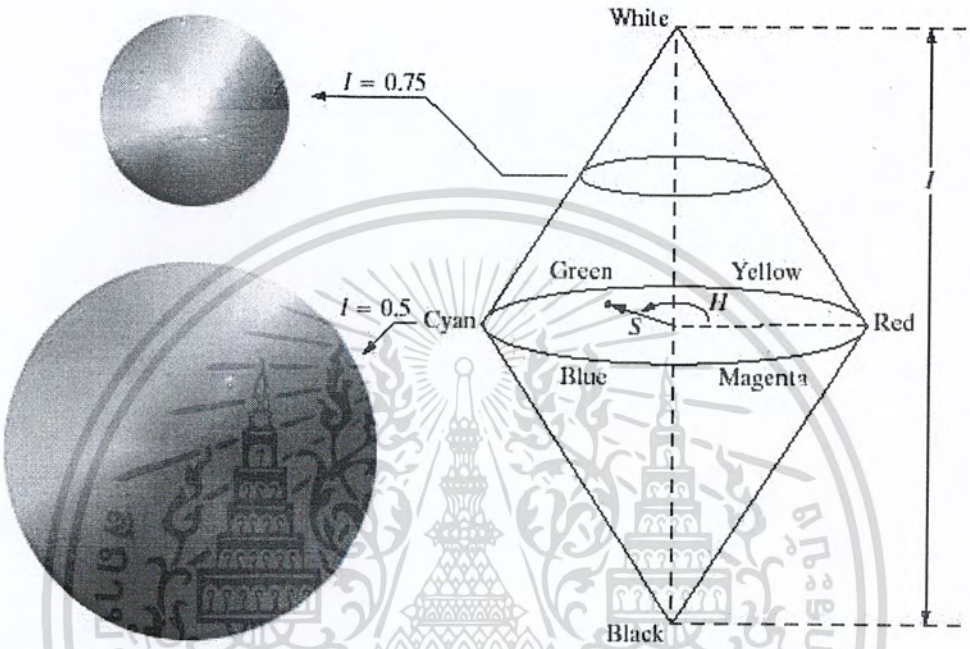


รูป 2.7 รูปแบบ RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.6 รูปแบบ HSI

ในระบบสี HSI (Hue, Saturation, Intensity) นั้นสีที่เรามองเห็นจะประกอบไปด้วยประเภทของสี (Hue), ความอิ่มตัวของสี (Saturation) และ ความเข้มของสี (Intensity) ดังรูป



รูป 2.8 รูปแบบ HIS

จากรูป สามารถอธิบายค่าแต่ละค่าในระบบสีHSIได้ดังนี้

1) ประเภทของสี (Hue) คือ เฉดสี ซึ่งจากรูปที่ 2-4นั้นค่า Hue คือมุมของกรวยโดยที่ค่าของ Hue จะมีค่าระหว่าง 0-359 ซึ่งจะไล่ลำดับการเปลี่ยนแปลงจากสีแดง (Hue = 0) ไปยังสีเขียว (Hue = 120) และ จากสีเขียวไปยังสีน้ำเงิน (Hue = 240) และ จากสีน้ำเงินไปยังสีแดง
2) ความอิ่มตัวของสี (Saturation) คือค่าที่ระบุว่าสีนั้นอยู่ห่างจากแกนกลางของกรวยมากเท่าใด โดยที่ ถ้าค่า Saturation มีค่ามากขึ้นก็จะทำให้สีที่ได้ใกล้เคียงกับเฉดสีที่แท้จริงมากขึ้น
3) ความเข้มของสี (Intensity) คือค่าที่ใช้ระบุว่าสีนั้นมีความเข้มมากเท่าใด ถ้าค่า Intensity มีค่ามากขึ้นก็จะทำให้สีที่ได้มีส่วนประกอบของสีขาวมากขึ้น ในทางกลับกันถ้าค่านี้มี ค่าน้อยลงก็จะทำให้สีที่ได้มีส่วนประกอบของสีดำมากขึ้นเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในระบบสี HSI นี้จะสังเกตได้ว่าถ้าความเข้มของสีที่เรามองเห็นมีการเปลี่ยนแปลงไป เช่นเปลี่ยนจากสีแดงที่มีค่า Hue = 0, Saturation = 1, และ Intensity = 0.5 เป็น สีแดงที่มี ความสว่างมากขึ้น (ค่า Intensity สูงขึ้น) การเปลี่ยนแปลงนี้จะไม่มีผลกับค่า Hue ทำ ให้ระบบสี HSI นี้เหมาะสำหรับการวิเคราะห์รูปภาพที่มีการเปลี่ยนแปลงความเข้มของ แสงมากการระบบสี RGB

2.2.7 พื้นฐานของการติดตาม (Track)

การติดตามจุดอ้างอิงของมาร์คเกอร์จากภาพเพื่อติดตามการเปลี่ยนแปลงตำแหน่งของจุดมาร์คเกอร์ในภาพใช้วิธีการสร้างพื้นที่จำลอง (Area Tracking) โดยพื้นที่จำลองจะมีขนาดเป็นรูปสี่เหลี่ยมจัตุรัสและจุดศูนย์กลางอยู่ที่จุดอ้างอิงของมาร์คเกอร์แต่ละจุด และมีขนาดของพื้นที่ประมาณ 30*30 พิกเซล พื้นที่จำลองจะทำการติดตามจุดมาร์คเกอร์จากเฟรมภาพถัดไปของกล้องตัวเดิม ซึ่งหากมีการเคลื่อนไหวของจุดมาร์คเกอร์ในเฟรมภาพถัดไปไม่มากนักหรือผู้แสดงเคลื่อนไหวอย่างช้าๆ จะทำให้สามารถพบจุดมาร์คเกอร์ของเฟรมภาพถัดไปในบริเวณของพื้นที่จำลอง จากนั้นก็จะทำการย้ายตำแหน่งจุดศูนย์กลางของพื้นที่จำลองไปยังตำแหน่งที่พบจุดมาร์คเกอร์ในเฟรมถัดไปจนครบทุกเฟรมภาพการเคลื่อนไหว

เมื่อต้องทำงานกับไฟล์วิดีโอซึ่งมีความแตกต่างจากไฟล์ภาพนิ่ง เรามักจะเลือกจับภาพเฉพาะวัตถุใดวัตถุหนึ่งหรือจับภาพตามวัตถุเคลื่อนที่ที่เราต้องการให้มันอยู่ภายในกรอบของภาพ ในการติดตามนั้นต้องใช้ความเข้าใจในเรื่องการเคลื่อนที่ของวัตถุซึ่งแบ่งการทำงานออกเป็นสองส่วนประกอบหลักๆ นั่นคือการจำแนกลักษณะและการทำโมเดล

2.2.8 การจำแนกลักษณะ

การจำแนกลักษณะมีความสำคัญในการค้นหาวัตถุที่เราสนใจจากภาพหนึ่งเฟรมในแต่ละเฟรมที่ต่อเนื่องกัน (Subsequent Frame) ของไฟล์วิดีโอ เทคนิคในเรื่องของโมเมนต์ (Moment) หรือ ฮิสโตรแกรม (Histograms) ของสีจะช่วยให้สามารถจำแนกวัตถุที่ต้องการได้ สิ่งที่จะทำการติดตามที่ยังจำแนกลักษณะไม่ได้จะเป็นปัญหาเกี่ยวข้อง การติดตามวัตถุโดยปราศจากการจำแนกลักษณะของวัตถุจะใช้ก็ต่อเมื่อต้องการจะพิจารณาการเคลื่อนไหวของวัตถุต่างๆ ว่าวัตถุใดที่มีการเคลื่อนไหวซึ่งน่าสนใจต่อการติดตาม เทคนิคสำหรับการติดตามวัตถุโดยปราศจากการจำแนกลักษณะมักจะเกี่ยวข้องกับการติดตามวัตถุที่เมื่อสังเกตด้วยตาดูแล้วน่าจะเป็นวัตถุที่มีนัยสำคัญกว่าวัตถุอื่นๆ หากใช้โปรแกรม OpenCV จะมีสองวิธีการสำหรับจัดการในส่วนนี้ ได้แก่เทคนิค ลูแคสคาเนด (Lucas-Kanade) และ ฮอร์นสชังค์ (Horn-Schunck) ซึ่งจะเป็นสิ่งที่ถูกอ้างถึงในเรื่อง “การติดตามการเคลื่อนไหวเฉพาะที่” (Sparse) หรือ “การติดตามการเคลื่อนไหวทั้งภาพ” (Dense) ของออพติคัลโฟลว์ (Optical Flow) ซึ่งจะขยายความการติดตามแบบออพติคัลโฟลว์ในลำดับถัดไป

ออพติคัลโฟลว์ (Optical Flow)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Optical Flow หรือ Optic Flow คือ รูปแบบของการเคลื่อนที่ของวัตถุ ที่พื้นผิวและบริเวณขอบของฉากรับภาพ (Visual Scene) ที่เกิดจากความสัมพันธ์ของการเคลื่อนไหวระหว่างผู้สังเกต (ตาหรือกล้องถ่ายภาพ) และพื้นหลัง เทคนิค Optical Flow ใช้ในการตรวจจับการเคลื่อนไหว การแบ่งส่วนวัตถุ เวลาในการชน (Time-to-Collision) และเน้นการคำนวณส่วนที่ขยายออก การชดเชยการเข้ารหัสของการเคลื่อนไหว และการวัดส่วนต่างของสเตอริโอ (Stereo) ใช้การเคลื่อนไหวของวัตถุบริเวณพื้นผิวและขอบของวัตถุ บ่อยครั้งที่อาจต้องการจะทำความเข้าใจระหว่างสองเฟรม (หรือช่วงเฟรมต่อเนื่อง) โดยปราศจากความรู้อื่นๆ ที่สำคัญเกี่ยวกับเฟรมเหล่านั้น โดยปกติแล้ว การเคลื่อนที่นั้นคือสิ่งที่บ่งชี้ว่าสิ่งที่เราสนใจกำลังเคลื่อนที่ไป

ความเร็วและแต่ละพิกเซลในเฟรมนั้นสามารถเชื่อมโยงกันได้เช่นเดียวกับการเปลี่ยนแปลงซึ่งอธิบายให้เห็นว่าระยะของพิกเซลได้เปลี่ยนไประหว่างเฟรมก่อนหน้ากับเฟรมปัจจุบัน โครงสร้างนั้นมักจะต้องอ้างถึงเรื่อง ออพติคัลโฟลว์ของทั้งภาพ (Dense Optical Flow) ซึ่งเกี่ยวข้องในเรื่องความเร็วและพิกเซลในรูปภาพใดรูปภาพหนึ่ง วิธีของฮอร์นสชังก์ (Horn Schunk) พยายามที่จะคำนวณในเรื่องสนามความเร็ว สิ่งหนึ่งที่ดูจะมุ่งตรงสู่วิธีการนี้คือความพยายามอย่างง่ายๆ เพื่อที่จะจับคู่กรอบรอบๆ แต่ละพิกเซลจากหนึ่งเฟรมไปยังอีกเฟรมซึ่งถูกอิมพลีเมนต์ค่า ด้วยเช่นกัน และเป็นที่รู้จักกันในชื่อเรียกที่ชื่อว่า บล็อกแมทชิง (Block Matching) ทั้งสองรูทีนนี้จะถูกกล่าวถึงในส่วนของเทคนิคการติดตามการเคลื่อนไหวทั้งภาพ (Dense Tracking Techniques)

ในทางปฏิบัติ การคำนวณออพติคัลโฟลว์ของทั้งภาพ (Dense Optical Flow) นั้นไม่ใช่เรื่องง่าย ลองพิจารณาจากการเคลื่อนที่ของแผ่นกระดาษสีขาว หลายๆ พิกเซลที่เหลือในเฟรมที่แล้วและจะทิ้งร่องรอยสีขาวในเฟรมถัดไป เพียงแค่แนวขอบกระดาษนั้นที่อาจเปลี่ยนแปลงไป ผลก็คือวิธีการติดตามเฉพาะที่นั้นมีการใส่จุดซึ่งง่ายต่อการติดตามมากกว่า เช่นเดียวกันการที่จะติดตามจุดหลายๆ จุดนั้นมีความยากยิ่งกว่า นอกจากนั้นแล้วการที่มันมีราคาของการคำนวณที่แพงยิ่งทำให้เห็นภาพของการทำออพติคัลโฟลว์ของทั้งภาพได้ชัดเจนยิ่งขึ้น สิ่งนี้จึงนำไปสู่ทางเลือกที่แตกต่างนั่นก็คือการทำออพติคัลโฟลว์เฉพาะที่ (Sparse Optical Flow) อัลกอริทึมนี้มีความน่าเชื่อถือตรงการกำหนดคุณลักษณะล่วงหน้าของจุดที่จะถูกติดตาม ถ้าจุดเหล่านี้มีลักษณะบางอย่างเป็นที่เหมาะสม

2.3 ทฤษฎีการตรวจจับการเคลื่อนไหว

Motion capture หมายถึง การอธิบายขั้นตอนการบันทึกการเคลื่อนไหวไม่ว่าจะเป็นวัตถุ สิ่งของหรือสิ่งมีชีวิต แล้วแปลความหมายของการเคลื่อนไหวให้อยู่ในรูปของโครงสร้างดิจิตอล การตรวจจับความเคลื่อนไหวของมนุษย์ถูกประดิษฐ์ครั้งแรกในประเทศสกอตแลนด์ โดยนำไปใช้ในด้านการทหาร บันเทิง กีฬา สถาปัตย์และการประยุกต์ทางการแพทย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การบันทึกการเคลื่อนไหว คือ การสร้างลักษณะท่าทางการเคลื่อนไหวที่สมจริงในรูปแบบสามมิติ โดยการตรวจจับหรือการบันทึกผลการเคลื่อนไหวโดยตรงจากคน สัตว์หรือวัตถุต่างๆ ซึ่งสามารถแสดงผลการเคลื่อนไหวที่ได้แบบเวลาจริง (Real-time) หรือบันทึกข้อมูลการเคลื่อนไหวเพื่อนำมาวิเคราะห์ในภายหลัง ซึ่งแตกต่างจากการเคลื่อนไหวอื่นๆ ที่ถูกสร้างด้วยมือ การตรวจจับความเคลื่อนไหวเป็นเทคนิคที่นิยมใช้เนื่องจากมีความเป็นธรรมชาติและสมจริงมากที่สุด ในการตรวจจับการเคลื่อนไหวของสิ่งของนั้น เป็นการจำลองการเคลื่อนไหวของวัตถุ เช่น การเคลื่อนไหวของแขนกล หรือ การเคลื่อนไหวของลูกบอลที่ตกลงสู่พื้น ซึ่งมีการเคลื่อนไหวที่แน่นอน มีจุดหมุนไม่มาก แต่การตรวจจับการเคลื่อนไหวของสิ่งมีชีวิตนั้นทำได้ยากกว่าเพราะสิ่งมีชีวิตมีการเคลื่อนไหวที่มีอิสระสูง

2.3.1 ประเภทของการตรวจจับความเคลื่อนไหว

1) การตรวจจับความเคลื่อนไหวโดยใช้แม่เหล็ก (Magnetic Motion Capture)
2) การตรวจจับความเคลื่อนไหวโดยใช้ชุดเชิงกล (Mechanical Motion Capture)
3) การตรวจจับความเคลื่อนไหวโดยใช้กล้องวีดีโอ (Optical Motion Capture) แบ่งได้เป็น แบบไม่มีมาร์คเกอร์ (Markerless), มาร์คเกอร์แบบสะท้อนแสงอินฟราเรด (Reflective Marker or Passive Marker) และมาร์คเกอร์แบบหลอด LED (Active Marker or Pulsed-LED)

2.3.2 การตรวจจับความเคลื่อนไหวโดยใช้กล้องวีดีโอ (Optical Motion Capture)

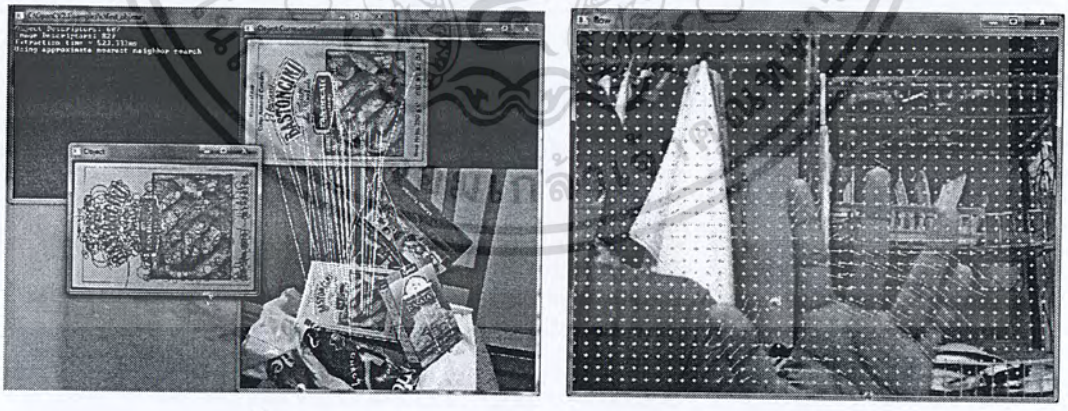
ใช้การบันทึกภาพผู้แสดงขณะแสดงท่าทางต่างๆ โดยที่ผู้แสดงจะมีมาร์คเกอร์ตัวสะท้อนหรือตัวส่งลำแสงติดอยู่ตามจุดสำคัญๆ ของร่างกาย หรือวัตถุใดๆที่ต้องการบันทึกการเคลื่อนไหว ระบบจะตรวจจับมาร์คเกอร์จากหลายๆ กล้องรวมกัน เพื่อนำไปใช้ในการคำนวณตำแหน่งใน 3 มิติต่อไป ระบบการตรวจจับการเคลื่อนไหวโดยใช้แสงมีข้อจำกัด คือ สภาพแวดล้อมที่ใช้ในการตรวจจับการเคลื่อนไหวจะต้องอยู่ในบริเวณที่ห่างจากแสงสะท้อนรบกวนอื่น ๆ มีการสลับมาร์คเกอร์และการบดบังของมาร์คเกอร์ ระบบการตรวจจับการเคลื่อนไหวโดยใช้กล้องเป็นระบบที่ใช้กล้องที่มีความละเอียดสูงเพื่อใช้ในการติดตามการเคลื่อนไหวของเครื่องหมายแบบสะท้อนกลับ (Reflective Marker) หรือตัวส่งลำแสงไดโอด (Pulsed-LED) โดยการนำมาร์คเกอร์ไปติดตามส่วนต่างๆ ของข้อต่อบนร่างกาย โดยทั่วไปแล้วการบันทึกการเคลื่อนไหวส่วนใบหน้าจะใช้กล้องประมาณ 1 ถึง 2 ตัว ขณะที่การบันทึกการเคลื่อนไหวของทุกส่วนของร่างกายจะใช้กล้องประมาณ 3 ถึง 16 ตัวหรือมากกว่า ระบบการตรวจจับการเคลื่อนไหวโดยใช้กล้องวีดีโอไม่ใช่ระบบที่แสงผลแบบ Real-time เนื่องจากข้อมูลที่ได้จากการการบันทึกจะถูกนำมาผ่านกระบวนการเพื่อประมวลผลก่อน โดยจุดศูนย์กลางของมาร์คเกอร์ที่ได้จากภาพถ่ายจะถูกนำมาเปรียบเทียบภาพที่ได้จากกล้องตัวอื่นๆ เพื่อคำนวณหาตำแหน่งในระบบ 3 มิติ แบบเฟรมต่อเฟรม ปัญหาที่เกิดขึ้นจะพบระหว่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กระบวนการตรวจหาตำแหน่งของมาร์คเกอร์และการสลับสับเปลี่ยนของมาร์คเกอร์ เมื่อได้ตำแหน่งของมาร์คเกอร์ที่ถูกต้องแล้วจึงนำค่าที่ได้มาคำนวณหาการเคลื่อนไหวที่แท้จริงของโครงกระดูกอีกครั้ง การเคลื่อนไหวของโครงกระดูกนี้สามารถนำไปใช้กับการเคลื่อนไหวของตัวละครแอนิเมชันต่างๆ ได้และนำไปสร้างเป็นการเคลื่อนไหวของผิวหนังบนตัวละครอีกครั้ง ซึ่งเป็นขั้นตอนสุดท้ายของการสร้างภาพการเคลื่อนไหวของตัวละคร ทำให้ตัวละครนี้มีการเคลื่อนไหวที่สมจริงและเป็นธรรมชาติ

2.4 OpenCV

OpenCV เป็นไลบรารี่สำหรับใช้ในการประมวลผลภาพ (Image Processing) ซึ่งเป็นไลบรารี่โอเพนซอร์ส (Open Source) สามารถดาวน์โหลดใช้งานได้ไม่เสียค่าใช้จ่าย ไลบรารี่ต่างๆของ OpenCV ได้พัฒนาขึ้นด้วยบริษัทอินเทล (Intel) จุดเด่นในด้านความสามารถของไลบรารี่ OpenCV คือสามารถประมวลผลภาพดิจิตอลได้ทั้งภาพนิ่ง และภาพเคลื่อนไหวเช่นภาพจากกล้องวีดีโอ หรือไฟล์วีดีโอ เป็นต้น โดยไม่ยึดติดทางด้านฮาร์ดแวร์ ทำให้ OpenCVสามารถพัฒนาโปรแกรมได้หลากหลายภาษา รวมถึงมีฟังก์ชันสำเร็จรูปสำหรับจัดการข้อมูลภาพและการประมวลผลภาพพื้นฐานเช่น การหาขอบภาพ การกรองข้อมูลภาพ โดยฟังก์ชันต่างๆของOpenCV จะสามารถเรียกใช้งานได้จะต้องมีการเรียก ไฟล์ส่วนหัว (Header file) และลิงค์ (Link) ไลบรารี่ต่างๆ รวมถึง DLL (Dynamic Link Library)



ก) ข)

รูป 2.9 ตัวอย่างของโปรแกรมที่พัฒนาโดย OpenCV

ก) ตรวจความตรงกันของสิ่งของ

ข) ตรวจหาความเคลื่อนไหว

เหตุผลที่เลือกใช้ OpenCV

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สนับสนุนการใช้งานกับ Visual Studio 2008 และระบบปฏิบัติการแบบ 64 บิต

เป็น library ที่นำไปใช้ได้ฟรี

มี source code และ ตัวอย่างให้ สามารถนำโค้ดที่มี ใน OpenCV ไปประยุกต์ใช้ได้

มีแหล่งข้อมูลให้ศึกษามาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

บทที่ 3
การออกแบบและพัฒนา

3.1 ความต้องการของระบบ

การใช้งานโปรแกรมที่จะพัฒนานี้ จะต้องใช้เว็บแคมเป็นอุปกรณ์ในการรับภาพ การเคลื่อนไหวของผู้เล่น และส่งการทำงานไปประมวลผลในโปรแกรมที่ถูกติดตั้งในคอมพิวเตอร์เพื่อตรวจสอบการเคลื่อนไหวของผู้เล่นจากนั้นส่งผลลัพธ์ ที่ได้จากการทำงานออกทางหน้าจอ โดยจะออกมาในลักษณะของผลการปฏิบัติ ว่าผ่านการฝึกในท่านั้นๆหรือไม่

การใช้งานโปรแกรมนี้จะต้องมีการรับภาพการเคลื่อนไหวของผู้เล่นจากเว็บแคมเพียงตัวเดียว การเตรียมสถานที่ให้มีความเหมาะสมกับการใช้งาน และจะต้องมีการจัดการชุดและมาร์คเกอร์ที่เหมาะสมสำหรับผู้เล่น เพื่อให้การประมวลผลภาพทำได้อย่างมีประสิทธิภาพ กล่าวคือสีของชุดที่ผู้เล่นสวมใส่ขณะใช้งานจะต้องมีความแตกต่างกับมาร์คเกอร์อย่างชัดเจน รวมทั้งมาร์คเกอร์แต่ละจุดต้องมีความแตกต่างกัน ถ้าหากสีของชุดผู้เล่นมีความใกล้เคียงกับสีของมาร์คเกอร์หรือมาร์คเกอร์ที่ใกล้เคียงกันมีสีใกล้เคียงกัน จะส่งผลให้ประสิทธิภาพของการประมวลผลภาพสำหรับการตรวจสอบการเคลื่อนไหวลดลง

3.2 ภาพรวมของระบบ

การทำงานของโปรแกรมเริ่มต้นด้วยการแสดงตัวอย่างของท่าคาราเต้-โด แล้วรับภาพของผู้เล่นด้วยเว็บแคมเพื่อตรวจสอบถึงลักษณะการเคลื่อนไหว จากนั้นทำการตรวจสอบกับตัวอย่างต้นแบบแล้วแสดงผลว่าท่าของผู้เล่นที่แสดงออกมาถูกต้องหรือไม่ ในระหว่างการเล่นจะมีการกำหนดลักษณะความยากง่ายเป็นลำดับ ผู้เล่นจะต้องผ่านระดับง่ายก่อนเพื่อปลดล็อกระดับที่ยากขึ้น และผู้เล่นสามารถใส่ท่าคาราเต้-โดเพิ่มเข้าไปในโปรแกรมเองได้

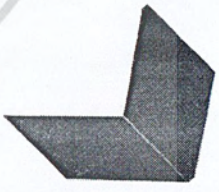
119274

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูป 3.1 รูปแบบของระบบโดยรวม

ภาพตัวอย่างจากโปรแกรม

ผู้เล่นแสดงท่าทางตามตัวอย่าง



ก)

ข)

รูป 3.2 หน้าต่างของโปรแกรม

ก) แสดงตัวอย่าง

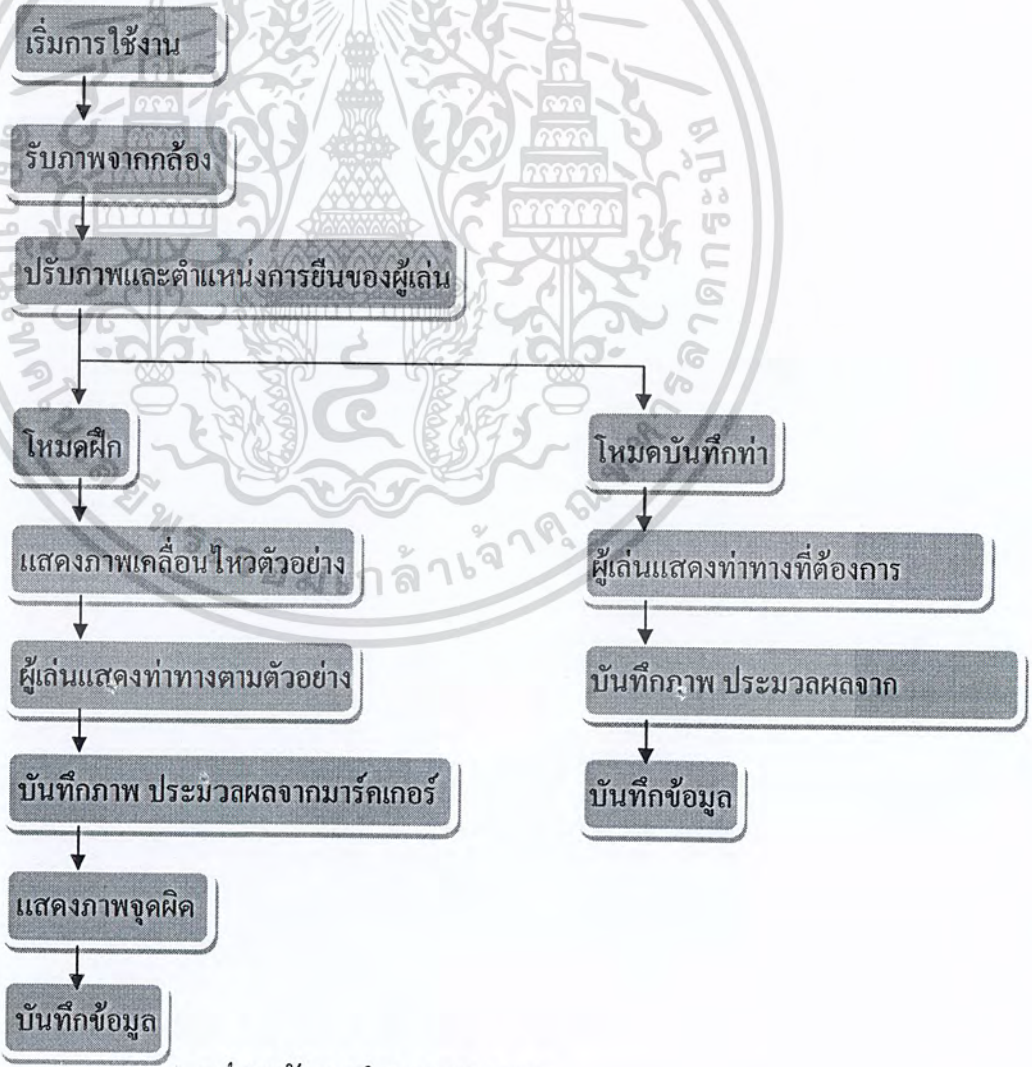
ข) ผู้เล่นแสดงท่าคาราเต้ตามตัวอย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 อัลกอริทึมและการออกแบบ

โปรแกรมฝึกสอนคาราเต้โดจัดทำขึ้นเพื่อพัฒนาแอปพลิเคชั่นที่สามารถตอบสนองการเคลื่อนไหวของมนุษย์ได้ โดยโปรแกรมจะตรวจสอบการแสดงท่าทางของผู้เล่นผ่านกล้องเว็บแคม โดยที่กล้องเว็บแคมจะเป็นตัวรับภาพการเคลื่อนไหวของร่างกาย แล้วนำเข้าไปประมวลผลผ่านหลักการของอิมเมจโปรเซสซิ่ง(Image Processing) และไลบรารี่ที่ชื่อว่าโอเพ่น(OpenCV) โดยโปรแกรมจะตรวจสอบความถูกต้องของท่าทาง โดยเปรียบเทียบกับวีดีโอตัวอย่างที่เราเปิดให้ดูก่อนผู้เล่นจะแสดงท่าทาง ซึ่งจะใช้ตัวมาร์คเกอร์ที่ติดบริเวณต่างๆ ของร่างกายเป็นตัวเปรียบเทียบระหว่างท่าทางที่ผู้เล่น กับท่าทางในวีดีโอต้นแบบที่เปิดให้ดูก่อนทำการแสดงท่าทางตาม

ในส่วนของวีดีโอต้นแบบที่ใช้ในเป็นตัวอย่างในโปรแกรม ตัวละครต้นแบบซึ่งเป็นบุคคลจริงนั้น ทางทีมงานพัฒนาได้ออกแบบไว้ว่าจะใช้ตัวละครซึ่งเป็นนักกีฬาคาราเต้โด ที่ได้ศึกษาคาราเต้โดมานานมากแล้ว ซึ่งสามารถมั่นใจได้ว่าท่าทางที่ออกมานั้นถูกต้อง



รูป 3.3 ผังการทำงานของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 การออกแบบวิธีการตรวจจับท่าทาง

วิธีที่ใช้ในการตรวจจับท่าทางของผู้เล่นนั้น ได้ออกแบบไว้ดังนี้

1) ตรวจจับมุมองศาของมาร์คเกอร์ โดยประมาณว่าลากเส้นผ่านจุดมาร์คเกอร์แต่ละจุด แล้ววัดองศา จากนั้นนำมาเปรียบเทียบกับต้นแบบ โดยคิดค่าที่อาจเกิดการคลาดเคลื่อนขึ้นเล็กน้อย
2) ใช้สีในการช่วยระบุตำแหน่งของร่างกาย โดยติดไว้ที่ตำแหน่งศีรษะ ศอกและมือของแขนทั้งสองข้าง และเข่าและข้อเท้าของขาทั้งสองข้าง

3.3.2 การออกแบบชุดข้อมูลสำหรับบันทึก

ออกแบบชุดข้อมูลสำหรับบันทึกข้อมูลของ Marker โดยมีลักษณะ คือ

ลำดับฉาก ใช้ข้อมูล 2 ตัวเลข

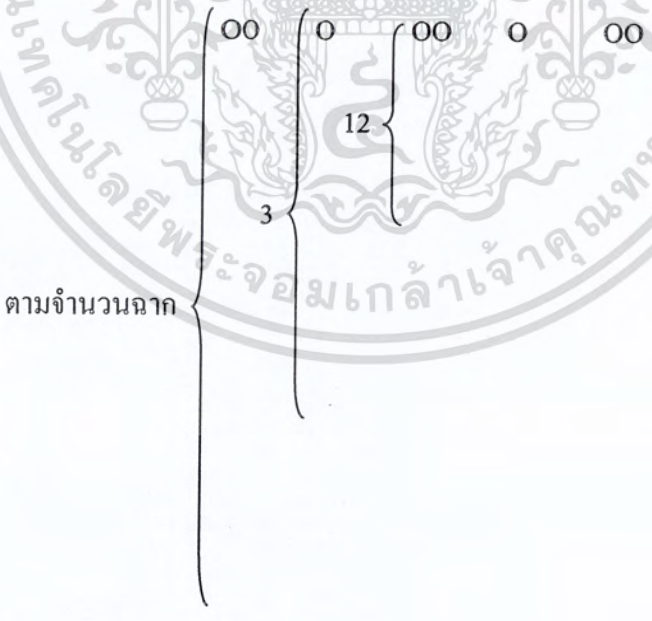
มุมกล้อง ใช้ข้อมูล 1 ตัวเลข

หมายเลข Marker ใช้ข้อมูล 2 ตัวเลข

การปรากฏของ Marker ใช้ข้อมูล 1 ตัวเลข

ความชันของ Marker ทั้งสอง ใช้ข้อมูล 2 ตัวเลข

สามารถเขียนเป็นชุดข้อมูลตัวอย่างได้ดังนี้



การเช็ค Marker สองจุดที่มีสีเดียวกัน

หาจุดแรกและจุดสุดท้ายที่ตรวจเช็คได้ของสีนั้น

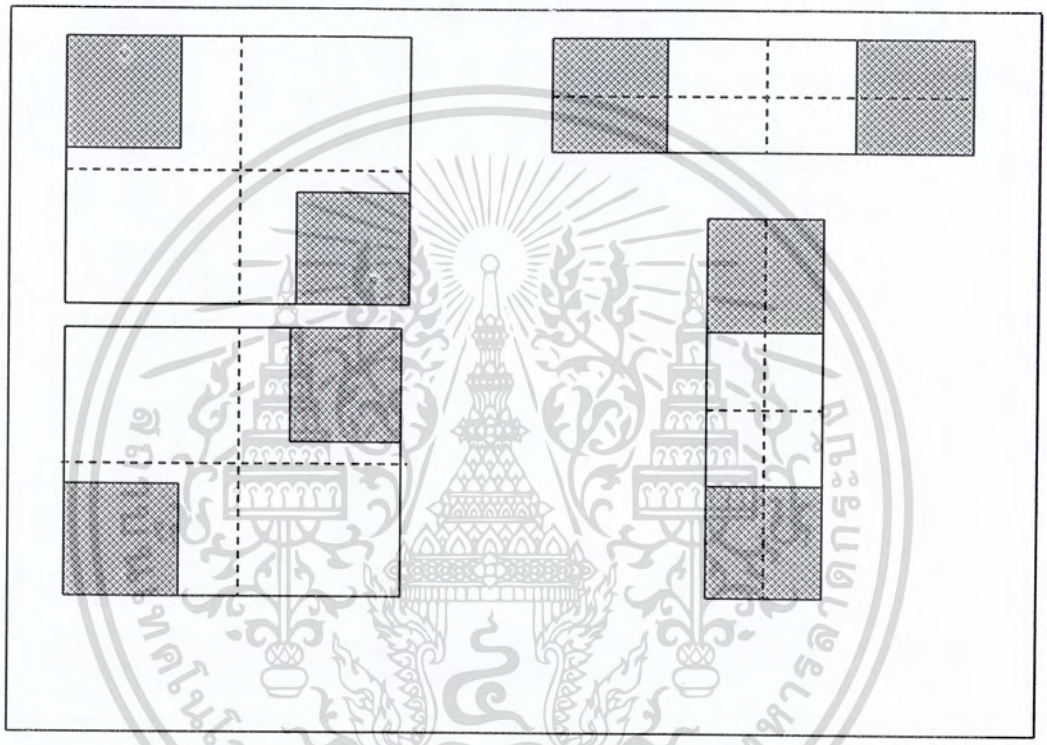
หาจุดกลางของทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หาแกนด้านซ้ายขวาที่ใกล้ที่สุดที่พบสี

หาแกนด้านบนล่างที่ใกล้ที่สุดที่พบสี

นำตำแหน่งมาคำนวณหาจุดกลางของ Marker ทั้งสองจุด โดยหากข้อ 3) ตำแหน่งที่ได้เป็นตำแหน่งเดียวกับแกน ให้ถือว่า Marker ทั้งสองมีจุดกลางแกน x เป็นจุดเดียวกัน และหากข้อ 4) ตำแหน่งที่ได้เป็นตำแหน่งเดียวกับแกน ให้ถือว่า Marker ทั้งสองมีจุดกลางแกน y เป็นจุดเดียวกัน และหากว่าข้อ 3) และ 4) เป็นจุดเดียวกันแล้ว ให้ถือว่า Marker นั้นมีจุดเดียว



รูป 3.4 การเช็ค Marker สองจุดที่มีสีเดียวกัน

3.3.4 หลักการตรวจสอบความถูกต้องของตำแหน่งมาร์คเกอร์

หลักการตรวจสอบความถูกต้องที่ใช้หลักๆ คือ ใช้ความชันของจุดที่ใกล้เคียงกัน โดยมีการกำหนดจุดที่ตรวจสอบดังนี้

ตำแหน่งมาร์คเกอร์บริเวณหัวไหล่ขวาตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อศอกขวา

ตำแหน่งมาร์คเกอร์บริเวณข้อศอกขวาตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อมือขวา

ตำแหน่งมาร์คเกอร์บริเวณหัวไหล่ซ้ายตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อศอกซ้าย

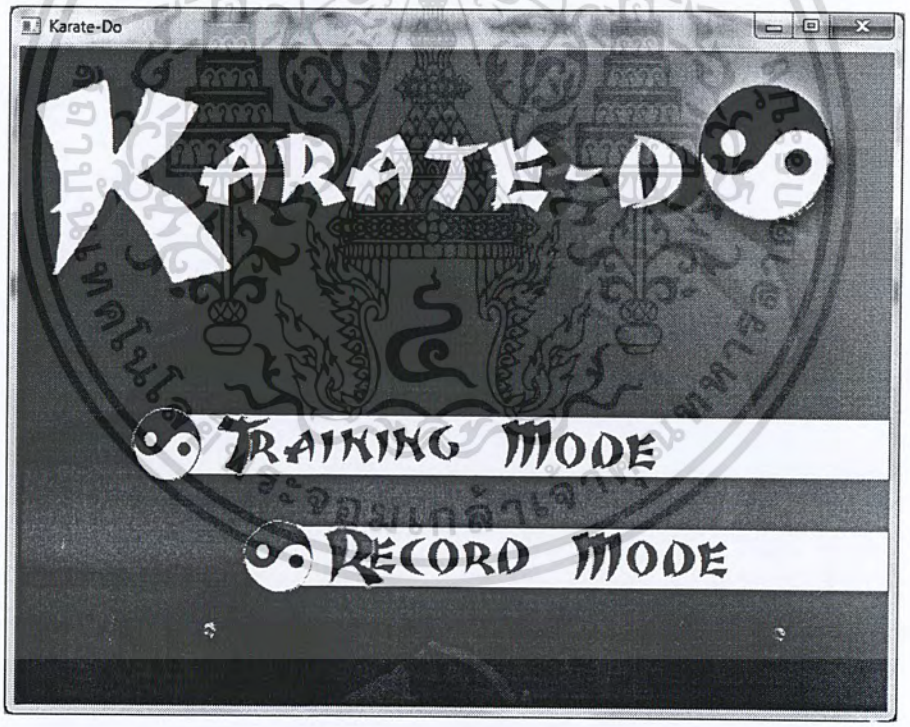
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งมาร์คเกอร์บริเวณข้อศอกซ้ายตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อมือซ้าย
ตำแหน่งมาร์คเกอร์บริเวณหัวเข่าขวาตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อเท้าขวา
ตำแหน่งมาร์คเกอร์บริเวณข้อเท้าขวาตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณข้อเท้าซ้าย
ตำแหน่งมาร์คเกอร์บริเวณข้อเท้าซ้ายตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณหัวเข่าซ้าย
ตำแหน่งมาร์คเกอร์บริเวณหัวเข่าซ้ายตรวจสอบกับตำแหน่งมาร์คเกอร์บริเวณหัวเข่าขวา

หลักการตรวจสอบความถูกต้องที่ใช้ต่อมาคือ การซ้านทับกันของมาร์คเกอร์ คือเมื่อมีการเคลื่อนไหวอวัยวะส่วนที่มีมาร์คเกอร์ติดไว้ผ่านอวัยวะที่มีมาร์คเกอร์ติดไว้ส่วนอื่น มาร์คเกอร์ที่ติดกับอวัยวะด้านหลังจะถูกบดบัง ซึ่งทำให้กล้องมองไม่เห็นมาร์คเกอร์ตัวที่ถูกบังนั้น ทำให้สามารถใช้ในการตรวจสอบได้

3.3.5 วิธีการใช้โปรแกรม

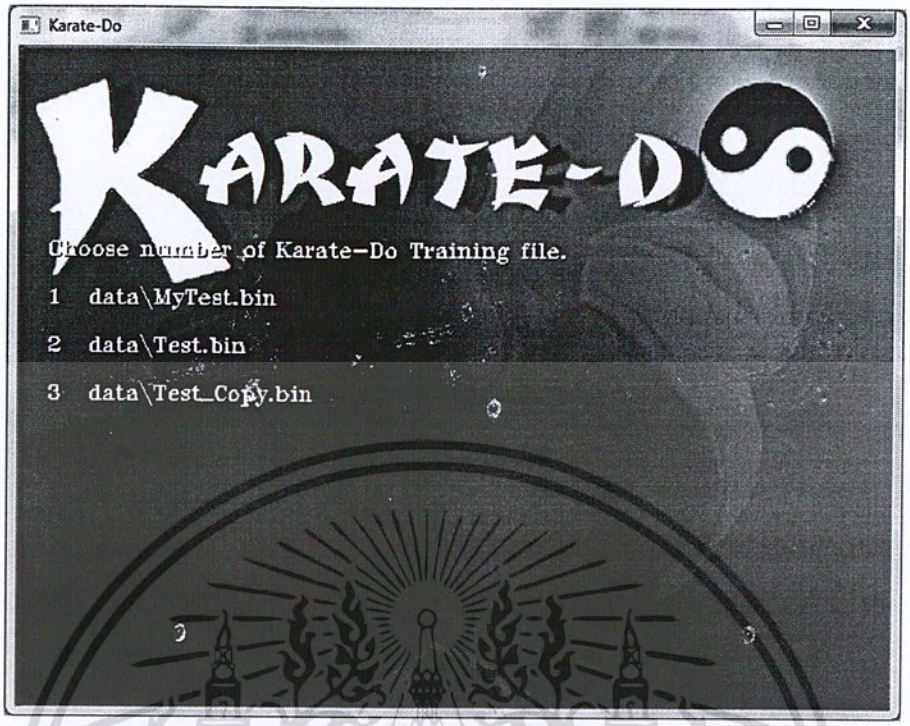
1) เริ่มจากการดับเบิ้ลคลิกที่ไอคอนของโปรแกรมเพื่อเข้าสู้โปรแกรม เมื่อเข้าสู่โปรแกรมจะเจอกับหน้าต่างดังรูป



รูป 3.5 หน้าแรกของตัวโปรแกรม

2) เมื่อเข้าตัวโปรแกรมจะมีตัวเลือกให้ 2 ตัวเลือก โดยที่ตัวเลือกแรกก็คือ Trining Mode มีไว้เพื่อเข้าไปใช้งานตัวโปรแกรม เราจะเข้าสู่หน้าของ Training Mode ได้โดยการกดที่ตัวอักษร “t”

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

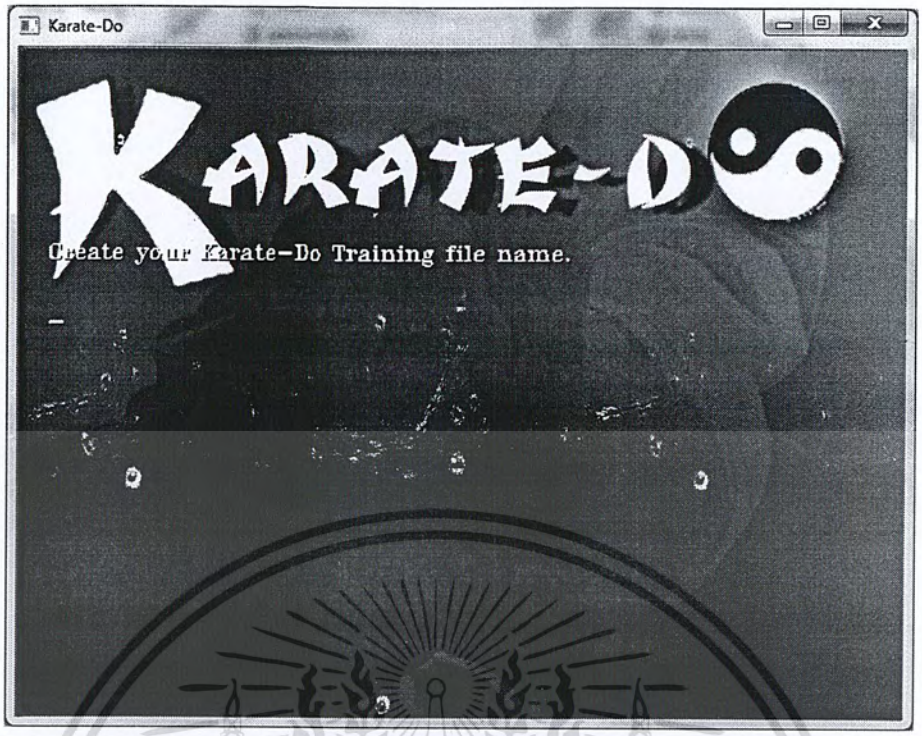


รูป 3.6 หน้า Training Mode

3) เลือกตัวเลขหน้าข้อความ เพื่อเลือกชื่อท่ารำที่เราต้องการศึกษาก่อนที่จะใช้รำจริงตามที่ต้องการ

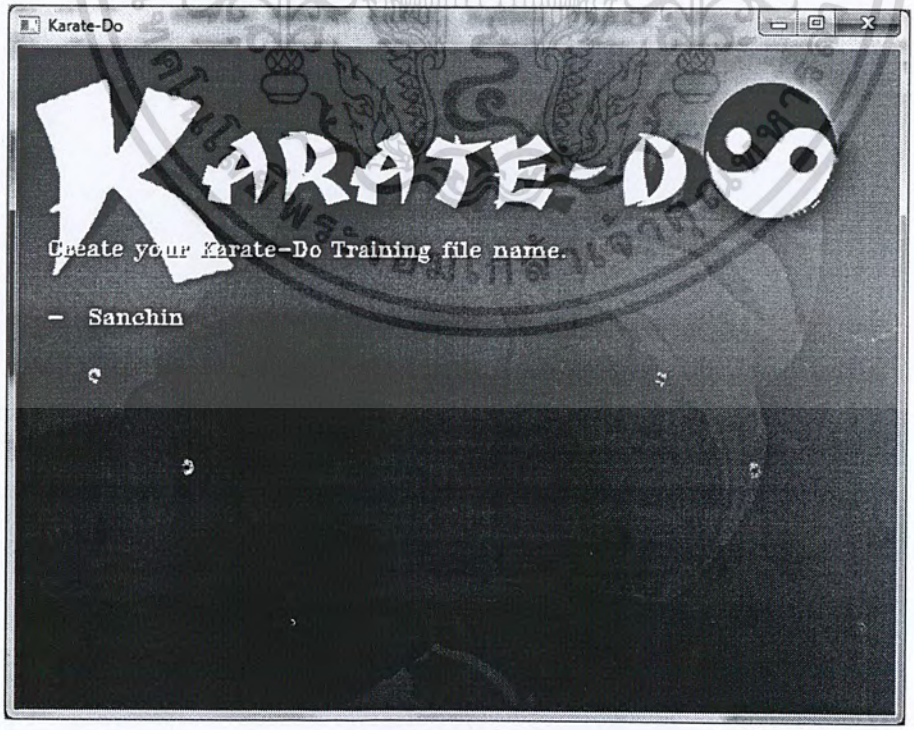
4) หน้า Record Mode เป็นหน้าต่างที่เราใช้บันทึกท่ารำ ที่เราได้รำไป เมื่อเราต้องการจะเข้าสู้หน้า Record Mode เราสามารถเข้าได้โดยการกด “r” เมื่อเข้าสู่ Record Mode ก็จะพบกับหน้าต่างดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 3.7 หน้า Record Mode

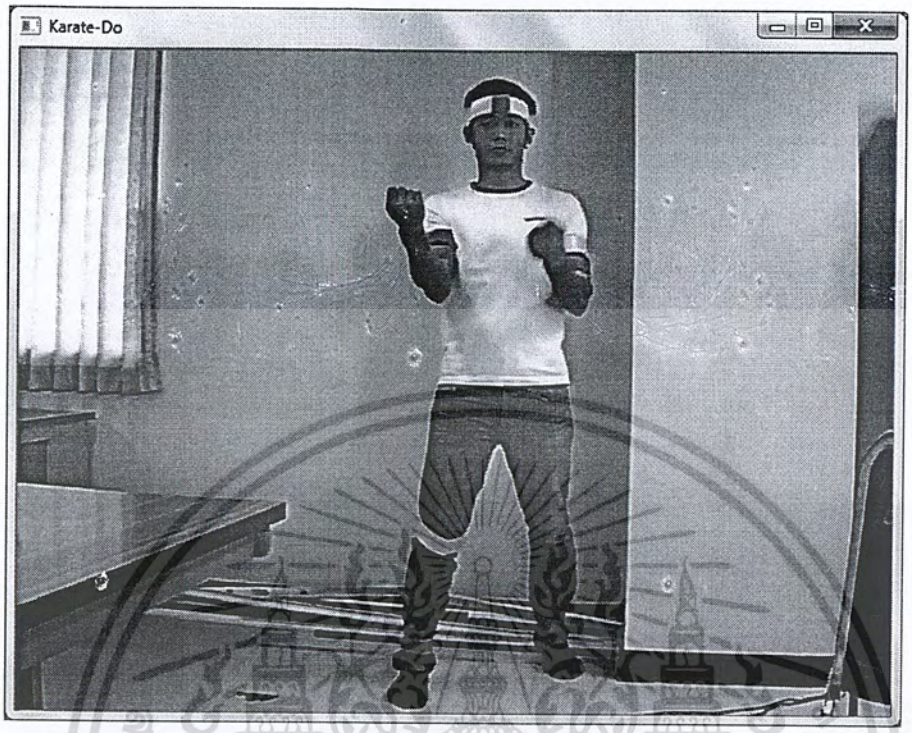
5) เมื่อเข้ามาสู้หน้าต่างนี้ โปรแกรมจะสั่งให้เราใส่ชื่อท่ารำที่เราต้องการรำ



รูป 3.8 หน้า Record Mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6) เมื่อใส่ชื่อท่าเรียบร้อยแล้ว ต่อไปจะเป็นการบันทึกท่าทางที่เราทำ



รูป 3.9 การบันทึกท่ารำของผู้เล่น

ลักษณะของการใช้ตัวโปรแกรมนี้ ผู้ที่ใช้โปรแกรมนี้จะต้องติดชุดมาร์คเกอร์ไว้ให้ เพื่อความสะดวกสบายในการเช็คความถูกผิดของท่าทางต่าง ๆ นอกจากมาร์คเกอร์แล้วยังต้องมีกล้องเวปแคมเพื่อใช้จับภาพขณะที่ผู้ใช้กำลังแสดงท่าทางต่าง ๆ ออกมา พื้นที่ ๆ ใช้ในการรำท่ารำจะต้องมีฉากหลังที่เป็นสีขาว ห้ามมีสีอื่นที่นอกเหนือจากสีขาว เพราะถ้ามีสีอื่น ๆ อยู่ในฉากหลังกล้องจับสีต่าง ๆ ที่อยู่ฉากหลังว่าเป็นมาร์คเกอร์ตัวหนึ่ง ซึ่งจะทำให้การประมวลคลาดเคลื่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

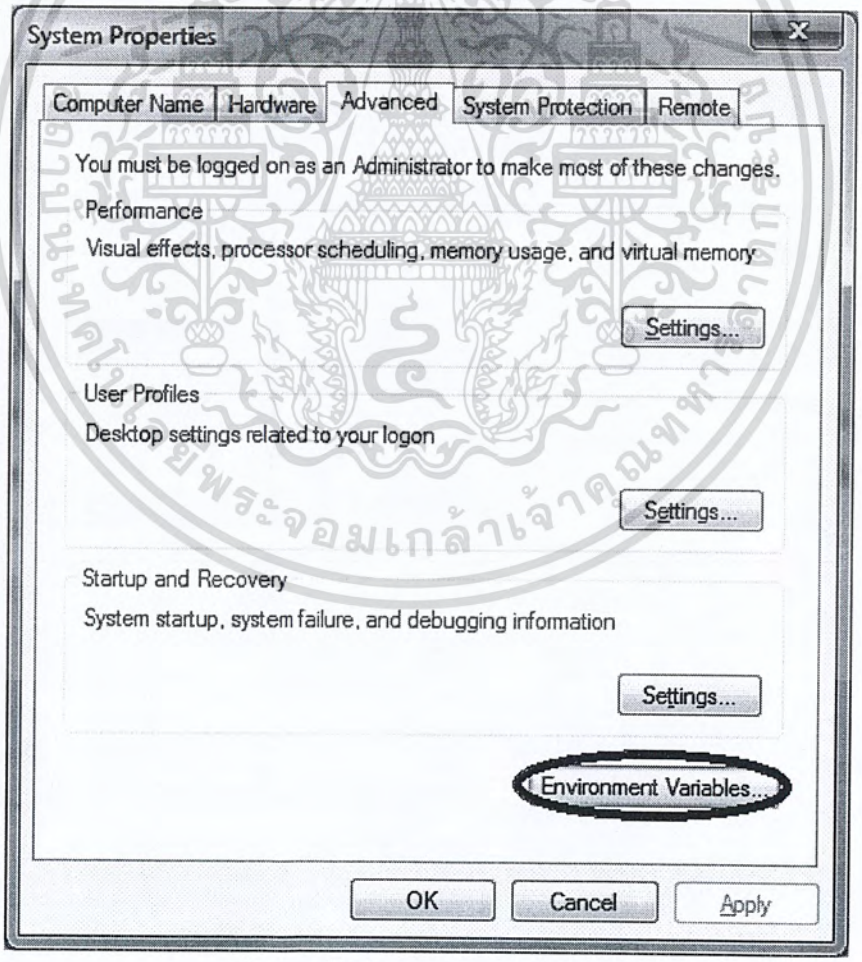
การทดลอง

4.1 การทดลองติดต่อ OpenCV เข้ากับ Microsoft Visual Studio 2008

การทดลองนี้เป็นการทดลองติดต่อ OpenCV เข้ากับ Microsoft Visual Studio 2008 เนื่องจากทางทีมผู้พัฒนาตัดสินใจใช้ OpenCV และ Microsoft Visual Studio 2008 ในการพัฒนาโปรเจค ซึ่งวิธีการติดต่อนั้น จะขอแบ่งเป็นหัวข้อหลักๆ ดังนี้

4.1.1 ติดตั้ง OpenCV

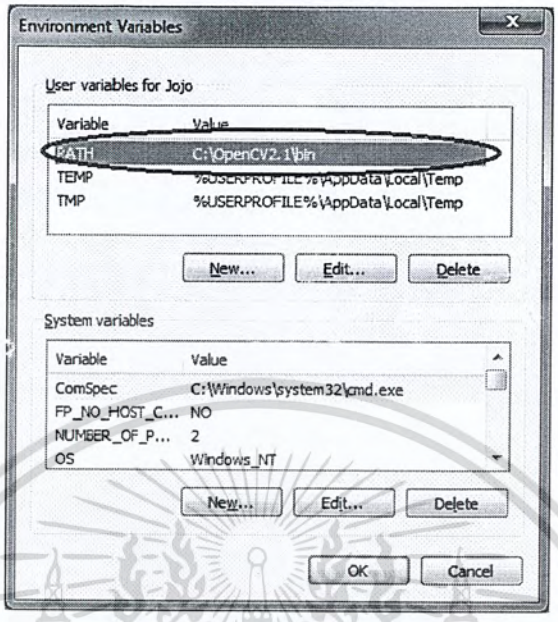
1) ติดตั้ง OpenCV โดยทางทีมผู้พัฒนาได้ใช้ตัว "OpenCV-2.1.0-win32-vs2008.exe"
2) ติดตั้ง PATH โดยเข้าไปที่ Control Panel> System >Advanced System Setting จะมีหน้าต่าง Systems Properties ขึ้นมา เข้าไปที่ Environment Variables



รูป 4.1 หน้าต่าง Systems Properties

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

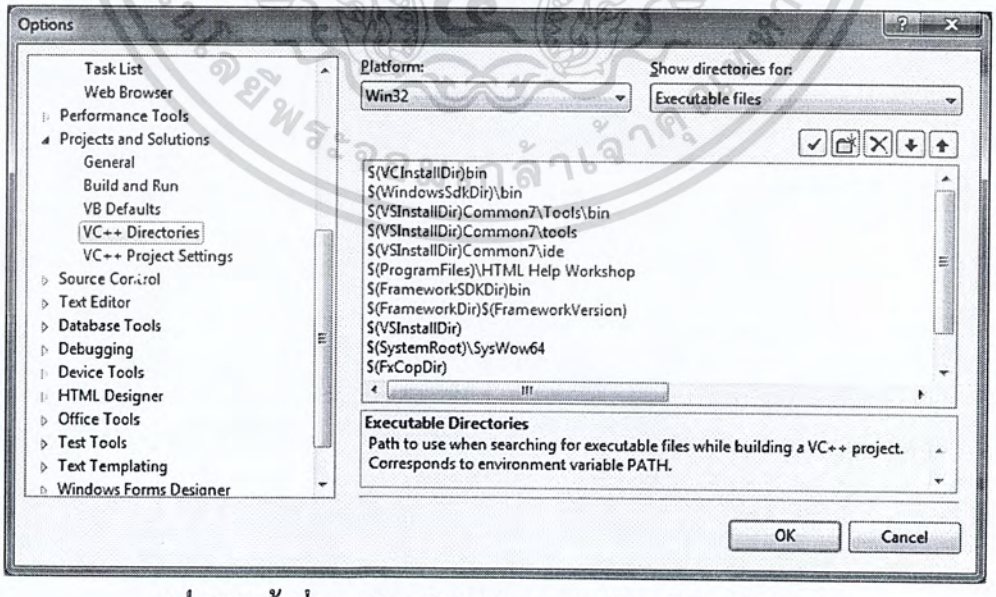
3) เช็คว่ามี PATH ของ OpenCV\bin หรือไม่ ถ้าไม่มีให้ Edit เพิ่มเข้าไป



รูป 4.2 PATH ของ OpenCV\bin

4.1.2 การตั้งค่า Microsoft Visual Studio 2008

1) เปิดโปรแกรม Microsoft Visual Studio 2008 ขึ้นมาแล้วเข้าไปที่ Tools > Options > Projects and Solutions > VC++ Directories



รูป 4.3 หน้าต่าง option ของ Microsoft Visual Studio 2008

2) เลือก "Show directories for: Include files" แล้วเพิ่ม C:\OpenCV2.1\include\opencv ลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เลือก "Show directories for: Library files" แล้วเพิ่ม C:\OpenCV2.1\lib ลงไป

4) เลือก "Show directories for: Source files" แล้วเพิ่ม C:\OpenCV2.1\src\ml, C:\OpenCV2.1\src\highgui, C:\OpenCV2.1\src\cxcore, C:\OpenCV2.1\src\cvaux\vs, C:\OpenCV2.1\src\cvaux และ C:\OpenCV2.1\src\cv ลงไป

4.1.3 การตั้งค่าโปรเจค

1) หลังจากสร้างโปรเจคแล้ว เปิด Project Properties ขึ้นมา

2) ไปที่ Configuration Properties > C/C++ > General ในส่วนของ Additional Include Directories ให้ใส่ C:\OpenCV2.1\include\opencv ลงไป

3) ไปที่ Configuration Properties > Linker > Input ในส่วนของ Additional Dependencies
ให้ใส่ C:\OpenCV2.1\lib\cv210.lib, C:\OpenCV2.1\lib\cxcore210.lib, C:\OpenCV2.1\lib\highgui210.lib, C:\OpenCV2.1\lib\cvaux210.lib, C:\OpenCV2.1\lib\opencv_ffmpeg210.lib และ C:\OpenCV2.1\lib\ml210.lib ลงไป

4.2 การทดลองรับภาพเคลื่อนไหวจากกล้อง

เนื่องการโปรแกรมฝึกสอนคาราเต้-โดนี้มีการรับภาพจากผู้เล่นเพื่อทำการประมวลผล จึงต้องทำการทดลองเพื่อรับภาพจากกล้อง ซึ่งคำสั่งสำคัญที่ใช้ในการรับกล้องคือ cvCaptureFromCAM( ) ซึ่งเป็นคำสั่งในการรับภาพจากกล้อง

โปรแกรม 4.1 โปรแกรมทดลองรับภาพเคลื่อนไหว

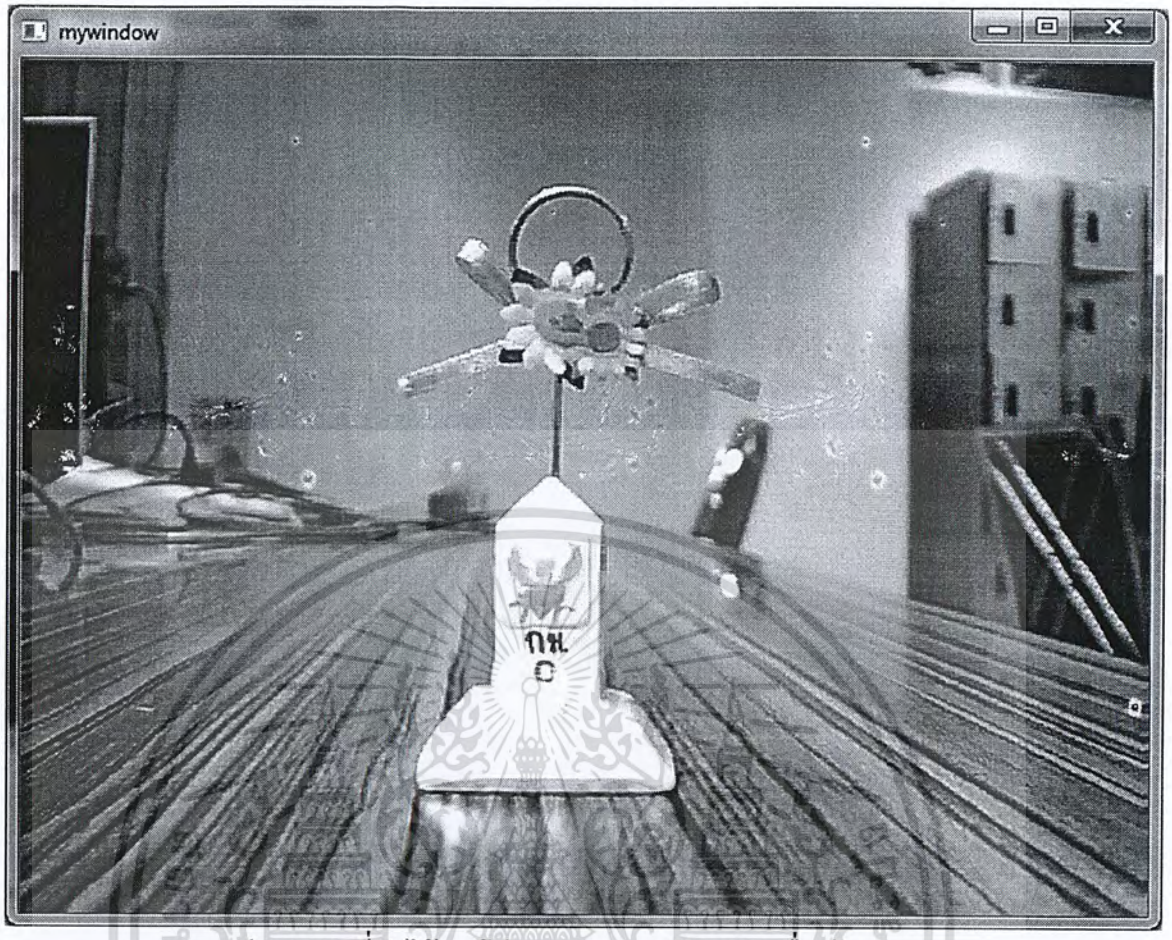
```
#include "stdafx.h"
#include <cv.h>
#include <cxcore.h>
#include <highgui.h>
#include <stdio.h>
void Record();
// A Simple Camera Capture
int main() {

CvCapture* capture = cvCaptureFromCAM(0);


if( !capture) {
    fprintf( stderr, "ERROR: capture is NULL \n" );
    getchar();
    return -1;
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.1 โปรแกรมทดลองรับภาพเคลื่อนไหว(ต่อ)

```
  // Create a window in which the captured images will be
presented
  cvNamedWindow( "mywindow", CV_WINDOW_AUTOSIZE );

// Grab Master frame for find size
 cvGrabFrame(capture);

 // Retrieve Master frame for find size
 IplImage* m_frame = cvRetrieveFrame(capture);

 // Set size of frame for AVI file
 CvSize  sizeT  =  cvSize(int(m_frame->width),int(m_frame-
>height));


 // Open Save DialogBox


 // If null path
 char* strFull = "beebrain.avi";
);


 while( 1 ) {
    // Get one frame
 cvGrabFrame(capture);
 m_frame = cvRetrieveFrame(capture);
 char* name = "mywindow";
 cvShowImage( name, m_frame );


    // Do not release the frame!
    //If ESC key pressed,
    //remove higher bits using AND operator
    if( (cvWaitKey(10) & 255) == 27 ) break;
  }


  // Release the capture device housekeeping
  cvReleaseCapture( &capture );
  cvDestroyWindow( "mywindow" );
  return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.4 ภาพที่จับได้จากโปรแกรมทดลองรับภาพเคลื่อนไหว

จากการทดลองโปรแกรมทดลองรับภาพเคลื่อนไหวนั้น ได้ผลว่า โปรแกรมสามารถแสดงภาพเคลื่อนไหวที่ถ่ายจากกล้องได้

4.3 การทดลองบันทึกวิดีโอจากกล้อง

ในโปรแกรมฝึกสอนคาราเต้-โดนั้น จะมีส่วนที่แสดงเป็นตัวอย่างให้กับผู้เล่น และส่วนที่แสดงท่าทางผู้เล่นหลังจากแสดงที่จำเป็นต้องใช้การบันทึกวิดีโอจากกล้อง ซึ่งมีคำสั่งที่สำคัญเช่น

cvCreateVideoWriter()	สำหรับสร้างไฟล์ .avi
cvWriteToAVI()	สำหรับการบันทึกวิดีโอในไฟล์ที่สร้างไว้
cvReleaseVideoWriter(&t_AVIWriter)	สำหรับจบการบันทึกวิดีโอ

โปรแกรม 4.2 โปรแกรมทดลองรับบันทึกวิดีโอจากกล้อง

```
#include "stdafx.h"
#include <cv.h>
#include <cxcore.h>
#include <highgui.h>
#include <stdio.h>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

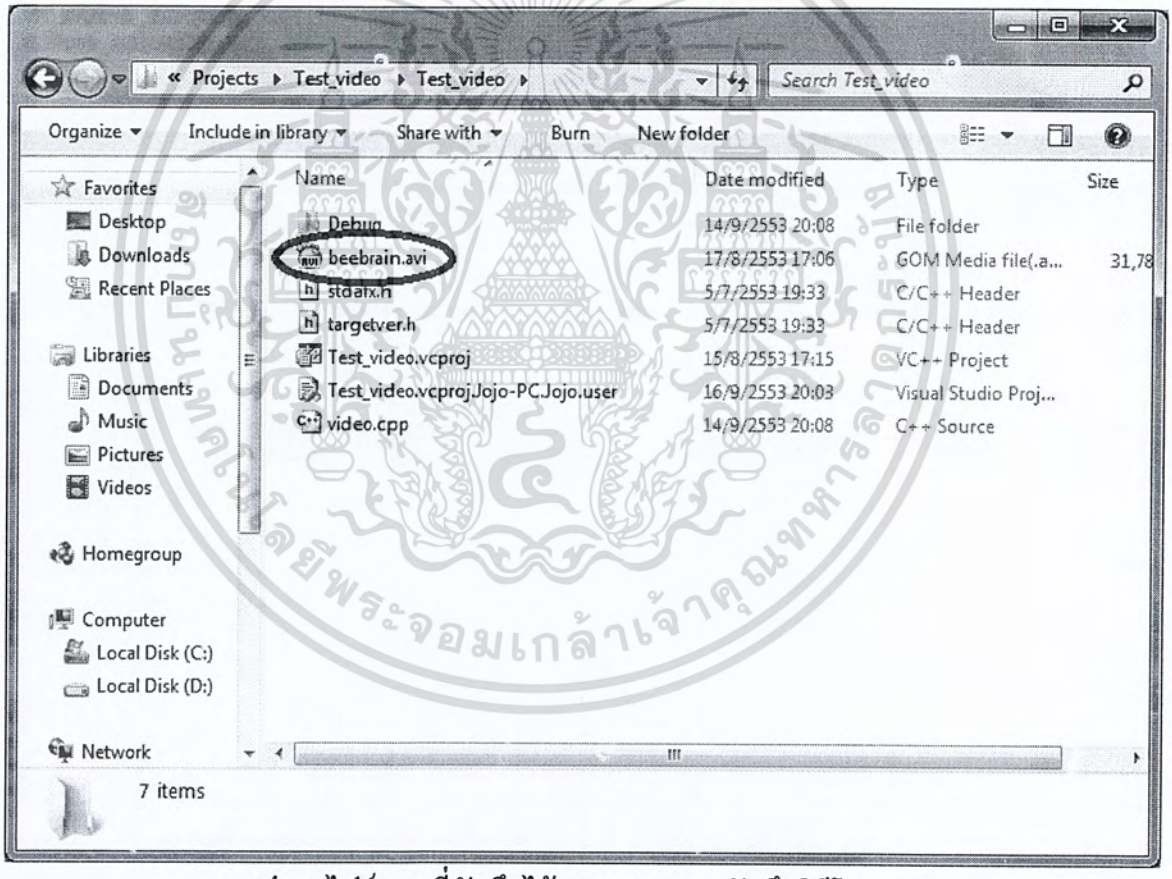
โปรแกรม 4.2 โปรแกรมทดลองรับบันทึกวิดีโอจากกล้อง(ต่อ)

```
void Record();
// A Simple Camera Capture

int main() {

  CvCapture* capture = cvCaptureFromCAM(0);

  if( !capture) {
    fprintf( stderr, "ERROR: capture is NULL \n" );
    getchar();
    return -1;
  }

  // Create a window in which the captured images will be
presented
  cvNamedWindow( "mywindow", CV_WINDOW_AUTOSIZE );
 // Grab Master frame for find size
 cvGrabFrame(capture);


 // Retrieve Master frame for find size
 IplImage* m_frame = cvRetrieveFrame(capture);

 // Set size of frame for AVI file
 CvSize  sizeT  =  cvSize(int(m_frame->width),int(m_frame-
>height));

 // Open Save DialogBox


 // If null path
 char* strFull = "beebrain.avi";

 // Create AVI file
  CvVideoWriter*t_AVIWriter=
cvCreateVideoWriter(strFull,-1,10, sizeT );

  while( 1 ) {
    // Get one frame
 cvGrabFrame(capture);
 m_frame = cvRetrieveFrame(capture);

 char* name = "mywindow";
  cvShowImage( name, m_frame );

  cvWriteToAVI(t_AVIWriter,m_frame);

 // Do not release the frame!
 //If ESC key pressed,
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.2 โปรแกรมทดลองรับบันทึกวิดีโอจากกล้อง(ต่อ)

```
    //remove higher bits using AND operator
    if( (cvWaitKey(10) & 255) == 27 ) break;
  }

 // Release the capture device housekeeping

  cvReleaseVideoWriter(&t_AVIWriter);

  cvReleaseCapture( &capture );
  cvDestroyWindow( "mywindow" );
  //cvDestroyWindow( "mys" );
  return 0;
}
```



รูป 4.5 ไฟล์ .avi ที่บันทึกได้จากการทดลองบันทึกวิดีโอ

จากการทดลองโปรแกรมบันทึกวิดีโอจากกล้องนั้น ได้ผลว่า โปรแกรมสามารถบันทึกวิดีโอจากกล้องได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.4 การทดลองรับภาพจากกล้องพร้อมกันสามตัว

โปรแกรมฝึกสอนคาราเต้-โดนั้นจำเป็นต้องใช้กล้องสามตัว ซึ่งจำเป็นต้องใช้คำสั่ง cvCaptureFromCAM( ) ซึ่งเป็นคำสั่งสำหรับจับภาพจากกล้อง โดยที่ตัวเลขในวงเล็บจะเป็นตัวกำหนดตัวกล้องที่ต้องการรับค่าเข้ามา หากค่าในวงเล็บเป็น -1 จะเป็นการหากล้องเองแบบอัตโนมัติ

โปรแกรม 4.3 โปรแกรมทดลองบันทึกวิดีโอจากกล้อง

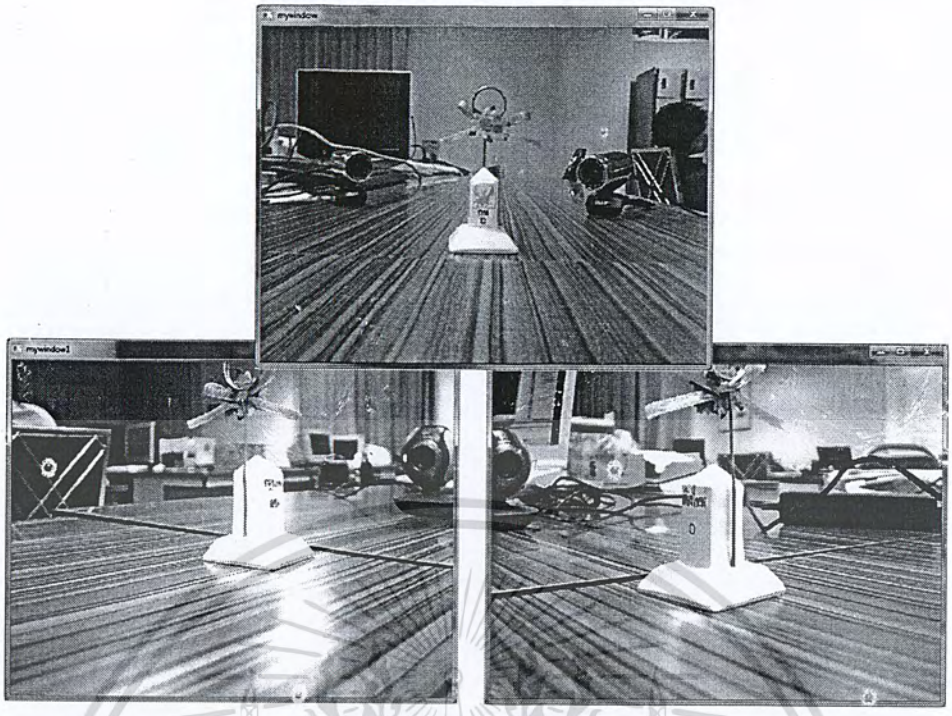
```
#include "stdafx.h"
#include <cv.h>
#include <cxcore.h>
#include <highgui.h>
#include <stdio.h>

void Record();
// A Simple Camera Capture


int main() {
  CvCapture* capture = cvCaptureFromCAM(0);
  CvCapture* capture1 = cvCaptureFromCAM(1);
  CvCapture* capture2 = cvCaptureFromCAM(2);


  if( !capture || !capture1 || !capture2) {
    fprintf( stderr, "ERROR: capture is NULL \n" );
    getchar();
    return -1;
  }


  // Create a window in which the captured images will be
presented
  cvNamedWindow( "mywindow", CV_WINDOW_AUTOSIZE );
  cvNamedWindow( "mywindow1", CV_WINDOW_AUTOSIZE );
  cvNamedWindow( "mywindow2", CV_WINDOW_AUTOSIZE );
 // Grab Master frame for find size



cvGrabFrame(capture);

 // Retrieve Master frame for find size
 IplImage* m_frame = cvRetrieveFrame(capture);
 IplImage* m_frame1 = cvRetrieveFrame(capture1);
 IplImage* m_frame2 = cvRetrieveFrame(capture2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.3 โปรแกรมทดลองบันทึกวิดีโอจากกล้อง(ต่อ)

```
// Set size of frame for AVI file
 CvSize  sizeT  =  cvSize(int(m_frame->width),int(m_frame-
>height));
 CvSize        sizeT1        =        cvSize(int(m_frame1-
>width),int(m_frame1->height));
CvSize sizeT2 = cvSize(int(m_frame2->width),int(m_frame2-
>height));


 // Open Save DialogBox
  while( 1 ) {
    // Get one frame
 cvGrabFrame(capture);
 m_frame = cvRetrieveFrame(capture);
 cvGrabFrame(capture1);
 m_frame1 = cvRetrieveFrame(capture1);
 cvGrabFrame(capture2);
 m_frame2 = cvRetrieveFrame(capture2);


 char* name = "mywindow";
 char* name1 = "mywindow1";
 char* name2 = "mywindow2";


  cvShowImage( name, m_frame );
  cvShowImage( name1, m_frame1 );
  cvShowImage( name2, m_frame2 );


 // Do not release the frame!
    //If ESC key pressed,
    //remove higher bits using AND operator
    if( (cvWaitKey(10) & 255) == 27 ) break;
  }
  // Release the capture device housekeeping
  cvReleaseVideoWriter(&t_AVIWriter);
  cvReleaseCapture( &capture );
  cvDestroyWindow( "mywindow" );
  cvReleaseCapture( &capture1 );
  cvDestroyWindow( "mywindow1" );
  cvReleaseCapture( &capture2 );
  cvDestroyWindow( "mywindow2" );
  return 0;
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูป 4.6 ภาพจากกล้องที่ได้จากการทดลองรับภาพจากกล้องพร้อมกันสามตัว

จากการทดลองรับภาพจากกล้องพร้อมกันสามตัว ได้ผลว่า โปรแกรมสามารถรับภาพจากกล้องพร้อมกันสามตัวได้

4.5 การทดลองจับการเคลื่อนไหวของภาพ

ทดลองโปรแกรมจับการเคลื่อนไหวโดยการตรวจจับทิศทาง เมื่อมีการเคลื่อนไหวไปทางซ้ายแล้วไปทางขวา หรือไปทางขวาแล้วไปทางซ้าย โปรแกรมจะแสดงคำว่า "Bye Bye" ออกมา

หลักการทำงานของโปรแกรมนี้คือ ตรวจจับการเคลื่อนไหวแต่ละจุดของภาพที่จับได้ และนำมาหาค่าเฉลี่ยได้เป็นทิศทางการเคลื่อนไหวโดยรวมของภาพทั้งหมด จากนั้นนำทิศทางที่ภาพเคลื่อนที่มาคิด โดยกำหนดว่าหากภาพเคลื่อนที่ไปที่มุมระหว่าง 135 องศา และ 225 องศา ถือว่าภาพเคลื่อนที่ไปทางซ้าย หากภาพเคลื่อนที่ไปที่มุมระหว่าง 315 องศา และ 360 องศา หรือระหว่าง 0 องศา และ 45 องศา ถือว่าภาพเคลื่อนที่ไปทางขวา

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ

```
#ifdef _CH_
#pragma package <opencv>

#endif
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
#define CV_NO_BACKWARD_COMPATIBILITY


#ifndef _EiC
// motion templates sample code
#include "cv.h"
#include "highgui.h"
#include <time.h>
#include <math.h>
#include <ctype.h>
#include <stdio.h>
#endif


// various tracking parameters (in seconds)
const double MHI_DURATION = 1;
const double MAX_TIME_DELTA = 0.5;
const double MIN_TIME_DELTA = 0.05;
// number of cyclic frame buffer used for motion
detection
// (should, probably, depend on FPS)
const int N = 4;


// ring image buffer
IplImage **buf = 0;
int last = 0;

// temporary images
IplImage *mhi = 0; // MHI
IplImage *orient = 0; // orientation
IplImage *mask = 0; // valid orientation mask
IplImage *segmask = 0; // motion segmentation map
CvMemStorage* storage = 0; // temporary storage
bool check, right = false, left = false;


// parameters:
//  img - input video frame
//  dst - resultant motion picture
//  args - optional parameters
void  update_mhi( IplImage* img, IplImage* dst, int
diff_threshold )
{
    double timestamp = (double)clock()/CLOCKS_PER_SEC; //
get current time in seconds
    CvSize size = cvSize(img->width,img->height); // get
current frame size
    int i, idx1 = last, idx2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
    IplImage* silh;
    CvSeq* seq;
    CvRect comp_rect;
    double count;
    double angle;
    CvPoint center, ck_center;
    double magnitude;
    CvScalar color;

     int num = 0;

    // allocate images at the beginning or
    // reallocate them if the frame size is changed
    if( !mhi || mhi->width != size.width || mhi->height
!= size.height ) {
        if( buf == 0 ) {
            buf = (IplImage**)malloc(N*sizeof(buf[0]));
            memset( buf, 0, N*sizeof(buf[0]));
        }

        for( i = 0; i < N; i++ ) {
            cvReleaseImage( &buf[i] );
            buf[i] = cvCreateImage( size, IPL_DEPTH_8U, 1
);
            cvZero( buf[i] );
        }
        cvReleaseImage( &mhi );
        cvReleaseImage( &orient );
        cvReleaseImage( &segmask );
        cvReleaseImage( &mask );

        mhi = cvCreateImage( size, IPL_DEPTH_32F, 1 );
        cvZero( mhi ); // clear MHI at the beginning
        orient = cvCreateImage( size, IPL_DEPTH_32F, 1 );
        segmask = cvCreateImage( size, IPL_DEPTH_32F, 1
);
        mask = cvCreateImage( size, IPL_DEPTH_8U, 1 );
    }

    cvCvtColor( img, buf[last], CV_BGR2GRAY );
     // convert frame to grayscale


    idx2 = (last + 1) % N; // index of (last - (N-1))th
frame
    last = idx2;
    silh = buf[idx2];
    cvAbsDiff( buf[idx1], buf[idx2], silh );
     // get difference between frames
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
    cvThreshold( silh, silh, diff_threshold, 1,
CV_THRESH_BINARY );
     // and threshold it
    cvUpdateMotionHistory( silh, mhi, timestamp,
MHI_DURATION );
     // update MHI

    // convert MHI to blue 8u image
    cvCvtScale( mhi, mask, 255./MHI_DURATION,
                (MHI_DURATION -
timestamp)*255./MHI_DURATION );
    cvZero( dst );
    cvMerge( mask, 0, 0, 0, dst );

    // calculate motion gradient orientation and valid
orientation mask
    cvCalcMotionGradient( mhi, mask, orient,
MAX_TIME_DELTA, MIN_TIME_DELTA, 3 );

    if( !storage )
        storage = cvCreateMemStorage(0);
    else
        cvClearMemStorage(storage);


    // segment motion: get sequence of motion components
    // segmask is marked motion components map. It is not
used further
    seq = cvSegmentMotion( mhi, segmask, storage,
timestamp, MAX_TIME_DELTA );


    // iterate through the motion components,
    // One more iteration (i == -1) corresponds to the
whole image
    //(global motion)
    for( i = -1; i < seq->total; i++ ) {

        if( i < 0 ) { // case of the whole image
            comp_rect = cvRect( 0, 0, size.width,
size.height );
            color = CV_RGB(255,255,255);
            magnitude = 100;
               check = true;
        }
        else { // i-th motion component
            comp_rect = ((CvConnectedComp*)cvGetSeqElem(
seq, i ))->rect;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
            if( comp_rect.width + comp_rect.height < 100
)
        // reject very small components
                continue;
            color = CV_RGB(255,0,0);
            magnitude = 30;
               check = false;
         }

         // select component ROI
         cvSetImageROI( silh, comp_rect );
         cvSetImageROI( mhi, comp_rect );
         cvSetImageROI( orient, comp_rect );
         cvSetImageROI( mask, comp_rect );

         // calculate orientation
         angle = cvCalcGlobalOrientation( orient, mask,
mhi, timestamp, MHI_DURATION);
         angle = 360.0 - angle;  // adjust for images with
top-left origin

         count = cvNorm( silh, 0, CV_L1, 0 );
        // calculate number of points within silhouette
      ROI

         cvResetImageROI( mhi );
         cvResetImageROI( orient );
         cvResetImageROI( mask );
         cvResetImageROI( silh );

         // check for the case of little motion
         if( count < comp_rect.width*comp_rect.height *
0.05 )
             continue;

         // draw a clock with arrow indicating the
direction

           center = cvPoint( (comp_rect.x +
comp_rect.width/2),
                             (comp_rect.y +
comp_rect.height/2) );

           if(check)
           {
                if(right&&left)
                {
                     printf("Bye Bye\n");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

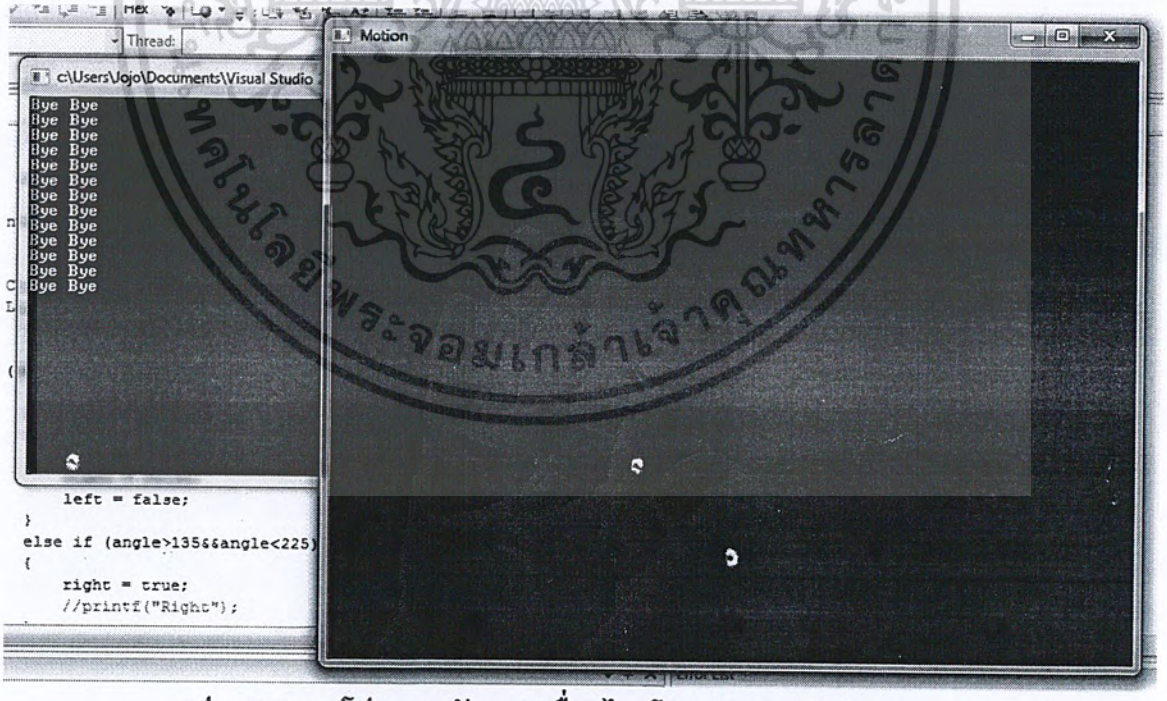
โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
                    right = false;
                    left = false;
                }
                else if (angle>135&&angle<225)
                {
                    right = true;
                }
                else                                                    if
((angle>0&&angle<45)||(angle>315&&angle<360))
                {
                    left = true;
                }
                else
                {
                    right = false;
                    left = false;
                }
            }
    }
}


int main(int argc, char** argv)
{
    IplImage* motion = 0;
    CvCapture* capture = 0;

    if( argc == 1 || (argc == 2 && strlen(argv[1]) == 1
&& isdigit(argv[1][0])))
        capture = cvCaptureFromCAM( argc == 2 ?
argv[1][0] - '0' : 0 );
    else if( argc == 2 )
        capture = cvCaptureFromFile( argv[1] );

    if( capture )
    {
        cvNamedWindow( "Motion", 1 );

        for(;;)
        {
            IplImage* image = cvQueryFrame( capture );
            if( !image )
                break;

            if( !motion )
            {
                motion = cvCreateImage( cvSize(image-
>width,image->height), 8, 3 );
                cvZero( motion );
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรแกรม 4.4 โปรแกรมทดลองจับการเคลื่อนไหวของภาพ(ต่อ)

```
                motion->origin = image->origin;
            }

            update_mhi( image, motion, 30 );
            cvShowImage( "Motion", motion );

            if( cvWaitKey(10) >= 0 )
                break;
        }
        cvReleaseCapture( &capture );
        cvDestroyWindow( "Motion" );
    }

    return 0;
}


#ifdef _EiC
main(1,"motempl.c");
#endif
```



รูป 4.7 ทดลองโปรแกรมจับการเคลื่อนไหวโดยการตรวจจับทิศทาง

จากการทดลองโปรแกรมจับการเคลื่อนไหวโดยการตรวจจับทิศทางนั้น ได้ผลว่า โปรแกรมสามารถตรวจจับการเคลื่อนไหวได้ แต่การตรวจจับนั้นอาจถูกการเคลื่อนไหวในทิศทางอื่น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ทำซ้ำ ดัดแปลง เผยแพร่ หรือจำหน่ายในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รบกวนได้ง่าย เนื่องจากการเคลื่อนมือไปทางซ้ายขวานั้น การเคลื่อนไหวโดยรวมอาจมีการเคลื่อนที่ขึ้นลงพร้อมกันไปด้วย

4.6 การทดลองทำมาร์คเกอร์

เนื่องจากแอปพลิเคชั่นที่ได้พัฒนาขึ้นมานั้นจำเป็นต้องใช้ตัวมาร์คเกอร์ในการใช้งานแอปพลิเคชั่น ผู้จัดทำจึงได้ทำมาร์คเกอร์ ซึ่งจากการทดลองเราสามารถสร้างมาร์คเกอร์ที่เหมาะสมกับการใช้งานได้ โดยใช้ไฟล์ภาพแล้วนำไปปริ้นสีออกมาจากร้านล้างรูปทั่วไป สาเหตุที่ต้องใช้ปริ้นจากร้านล้างรูปเพราะ สีที่นอกเหนือจากสีในคอมพิวเตอร์ เช่น สีของผ้า สีอะคีริคนั้นมาร์คเกอร์ไม่สามารถที่จะตรวจเช็คได้ ค่าสีที่ใช้จะต้องใกล้เคียงค่าสีในคอมพิวเตอร์จริง ๆ ซึ่งค่าสีต่าง ๆ มีค่าดังนี้ สีเขียว = [0 255 0] สีแดง = [255 0 0] สีน้ำเงิน = [0 0 255] สีม่วง = [255 0 255] สีฟ้า = [0 255 255] สีเหลือง = [255 255 0] ซึ่งค่าสีแต่ละค่าแทนค่าสีแดง สีเขียว สีน้ำเงิน (RGB) ตามลำดับ

การติดมาร์คเกอร์ตามร่างกายสามารถติดได้ตามนี้

รูป 4.8 ตำแหน่งการติดมาร์คเกอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุป

5.1 สรุป

จากการทดลองและพัฒนาแอปพลิเคชั่นนี้ ทีมผู้พัฒนาได้เปลี่ยนแนวคิดมาใช้กล้องเพียง 1 ตัว จากการใช้กล้องทั้งหมด 3 ตัว เพื่อความสะดวกในการเช็คและความสะดวกในการใช้พื้นที่ จากการพัฒนาโปรแกรมจนเสร็จสิ้น โปรแกรมสามารถที่จะบันทึกท่ารำของคาราเต้โด และสามารถเช็คความถูกต้องของท่าทางได้ แต่อาจจะเช็คได้ค่อนข้างยาก เพราะท่ารำของคาราเต้โดมีความละเอียดในการตรวจเช็คมาก และมีจุดหมุนของร่างกายมากจึงทำให้การเช็คค่อนข้างที่จะเช็คได้ยากพอสมควร

5.2 ปัญหาและอุปสรรค

ปัญหาและอุปสรรคในการพัฒนาแอปพลิเคชั่นนี้ ทางผู้พัฒนาได้แบ่งปัญหาและอุปสรรคออกเป็นดังนี้

1) แอปพลิเคชั่นนี้เป็นแอปพลิเคชั่นที่ต้องใช้พื้นที่มาก และพื้นที่ ๆ ใช้ต้องมีฉากหลังเป็นสีขาวหรือดำ
2) แอปพลิเคชั่นนี้มีอุปกรณ์ที่จะต้องใช้ค่อนข้างมาก เช่น กล้องเว็บแคม ชุดที่ต้องใช้ในการเล่น มาร์คเกอร์ เป็นต้น
3) สีของมาร์คเกอร์ต้องเป็นสีที่มีค่าของสีที่ค่อนข้างแน่นอน ซึ่งยากต่อการหาอุปกรณ์ที่มีสีตามสเปคที่ต้องการ
4) แสงสะท้อน และเงามีผลทำให้การตรวจเช็คสีของโปรแกรมผิดพลาด
5) ทีมพัฒนาที่มีความรูปเกี่ยวกับคาราเต้โดเกิดอุบัติเหตุกระดูดหัก จึงทำให้ไม่สามารถที่จะแสดงท่าทางที่ถูกต้องเพื่อเป็นตัวสอย่างในการเล่นแอปพลิเคชั่นนี้ได้
6) มาร์คเกอร์ที่ใช้ไม่สามารถยืดหยุ่นได้ จึงเกิดการชำรุดง่าย
7) ร่างกายของแต่ละคนมีขนาดไม่เท่ากัน จึงต้องสร้างมาร์คเกอร์ที่มีความสามารถในการปรับเปลี่ยนขนาดตามสัดส่วนของผู้เล่น

5.3 แนวทางการแก้ไข

1) ปรับเปลี่ยนจากการใช้กล้องทั้งหมด 3 ตัว มาเป็นกล้องเพียงแค่ 1 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) ศึกษาเกี่ยวกับการเกี่ยวกับการเขียนโปรแกรมเพื่อติดต่อกับกล้อง ภาพ และกราฟฟิคให้มากขึ้น
3) ทีมงานแก้ปัญหาเรื่องสีของมาร์คเกอร์โดยการสร้างสีของมาร์คเกอร์เป็นไฟล์ภาพ และทำการล้างเป็นรูปออกมา สาเหตุที่ต้องล้างเป็นรูปเพราะเพื่อความแข็งแรงของตัวมาร์คเกอร์ ซึ่งหากเป็นกระดาษธรรมดามาร์คเกอร์จะชำรุดได้ง่ายกว่าปกติ
4) ทีมงานใช้วิธีการโหลดวีดีโอจากอินเตอร์เนตมาเป็นเป็นวีดีโอตัวอย่างแทนการให้สมาชิกในทีมเป็นผู้แสดงเป็นตัวอย่าง
5) สร้างมาร์คเกอร์ให้มีความสามารถในการปรับเปลี่ยนขนาดได้ เพื่อที่จะรองรับกับบุคคลที่มีสัดส่วนที่แตกต่างกัน

5.4 แนวทางการพัฒนาต่อ

จากการพัฒนาแอปพลิเคชั่นนี้ขึ้นมาทางทีมงานผู้พัฒนาเห็นว่า แอปพลิเคชั่นนี้เป็นแอปพลิเคชั่นที่ต้องตรวจเช็คการเคลื่อนไหวของร่างกายของมนุษย์หลายจุด ซึ่งเป็นเรื่องยากในการตรวจเช็ค แต่แอปพลิเคชั่นนี้สามารถที่จะพัฒนาต่อไปในกีฬาอีกหลายชนิด เพื่อเป็นส่งเสริมการออกกำลังกาย การกระจายความรู้ทางด้านกีฬาออกไปสู่สถานที่ ๆ ไม่มีการส่งเสริมหรือสนับสนุนทางด้านกีฬา ซึ่งเป็นแอปพลิเคชั่นที่มีประโยชน์ต่อเยาวชนของประเทศชาติ ซึ่งปัจจุบันเยาชนไทยใช้ประโยชน์จากคอมพิวเตอร์ได้ค่อนข้างน้อย โปรแกรมนี้จะทำให้เกิดประโยชน์ต่อเยาชนของประเทศ และเมื่อเราสามารถพัฒนาแอปพลิเคชั่นที่เกี่ยวข้องกับกีฬาได้ ต่อไปในอนาคตเราก็จะสามารถที่จะพัฒนาต่อไปในส่วนอื่นที่นอกเหนือจากกีฬาได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

GaryBradski . OpenCV .

Online .Available : http://opencv.willowgarage.com/wiki/Welcome

Undermine . OpenCV .

Online .Available : http://imageopencv.blogspot.com/

Tutorial Stuffs . การใช้งาน OpenCV Library เบื้องต้นเพื่อการประมวลผลภาพจากกล้อง.

Online . Available : http://fibo.kmutt.ac.th/blog/humanoid

Matthew Hurajt Getting Started with OpenCV in Microsoft Visual Studio 2008 in Windows Vista.

Online . Available : http://uowteaminvincible.wordpress.com

Intel Corporation . **Open Source Computer Vision Library** .

U.S.A. December 8, 2000

ประวัติคาราเต้.

Online.Available: http://www.sakuradojo-th.com

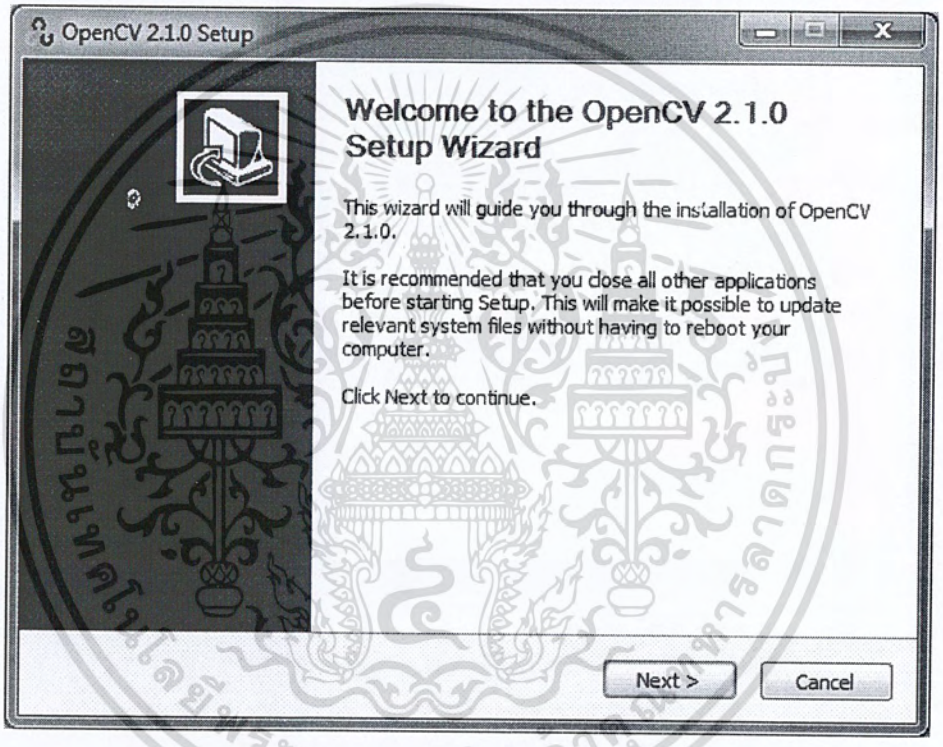
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
คู่มือการติดตั้ง

ก.1 คู่มือการติดตั้งโปรแกรมเพื่อใช้งานโปรแกรมฝึกสอนคาราเต้-โด

1) เริ่มต้นโดยการลง OpenCV 2.1.0 เลือกไฟล์ OpenCV-2.1.0-win32-vs2008.exe

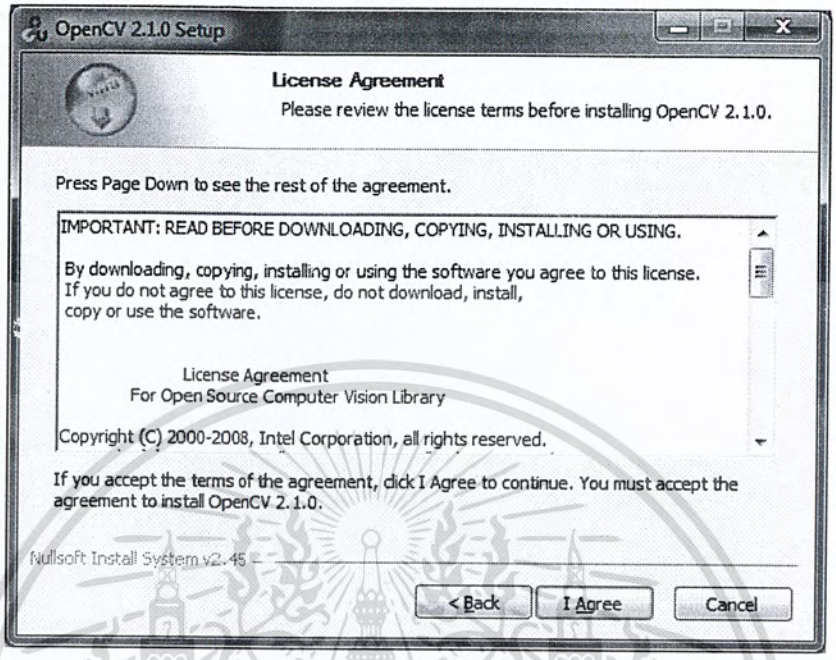
2) คลิก Next



รูป ก.1 หน้าการเริ่มต้นการลง OpenCV

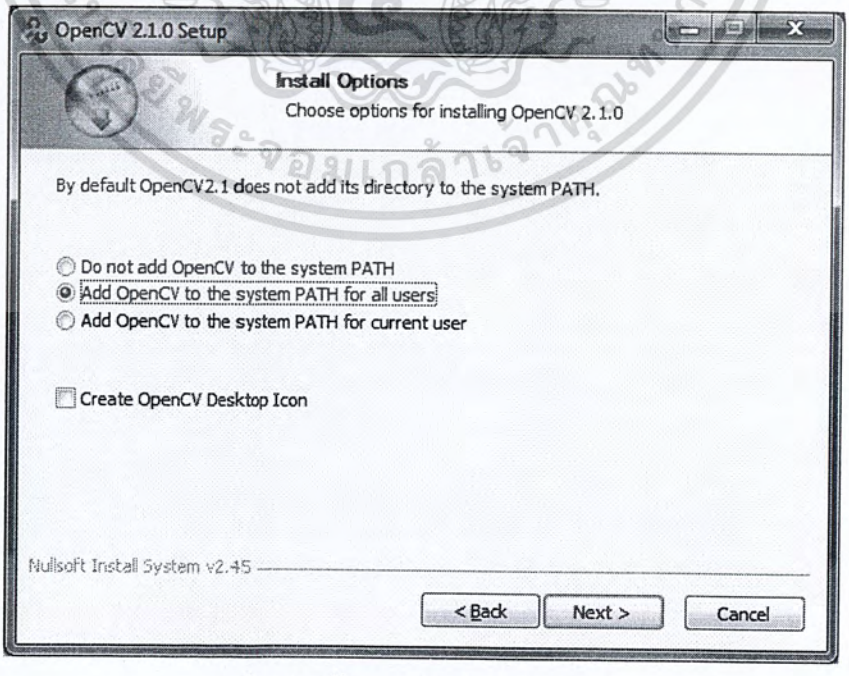
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) เลือก I Agree



รูป ก.2 หน้าการยอมรับข้อตกลงของ OpenCV

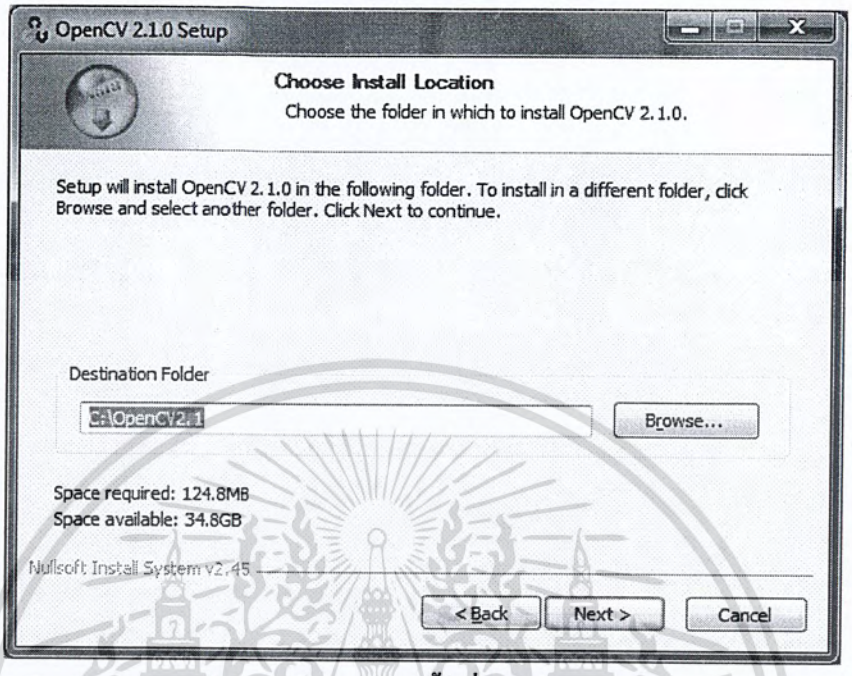
4) เลือก Add OpenCV to the system PATH for all users แล้วคลิก Next



รูป ก.3 หน้าการแอด Add OpenCV

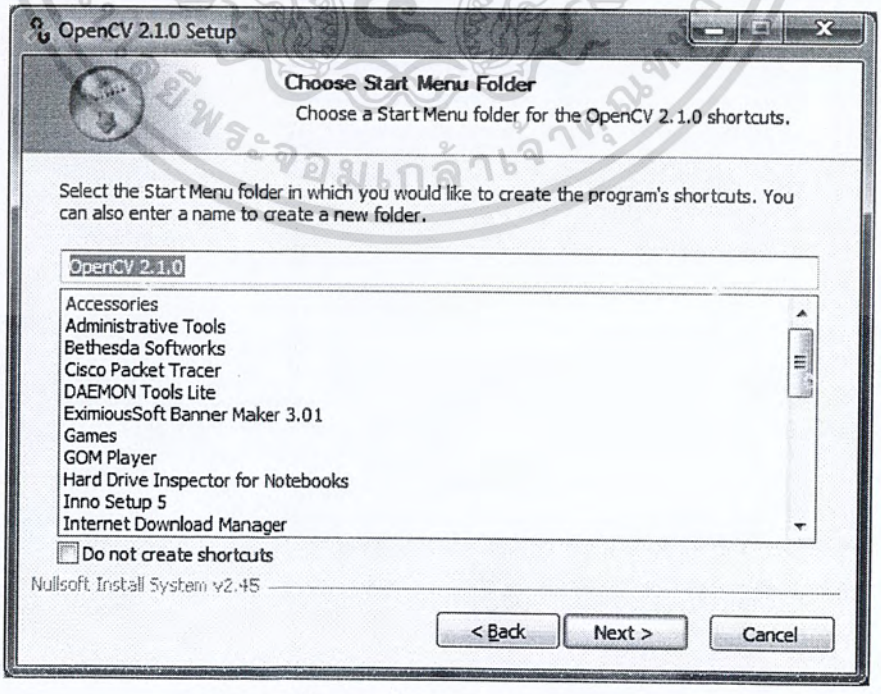
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) คลิก Next



รูป ก.4 หน้าการเลือกพื้นที่ ๆ ลง OpenCV

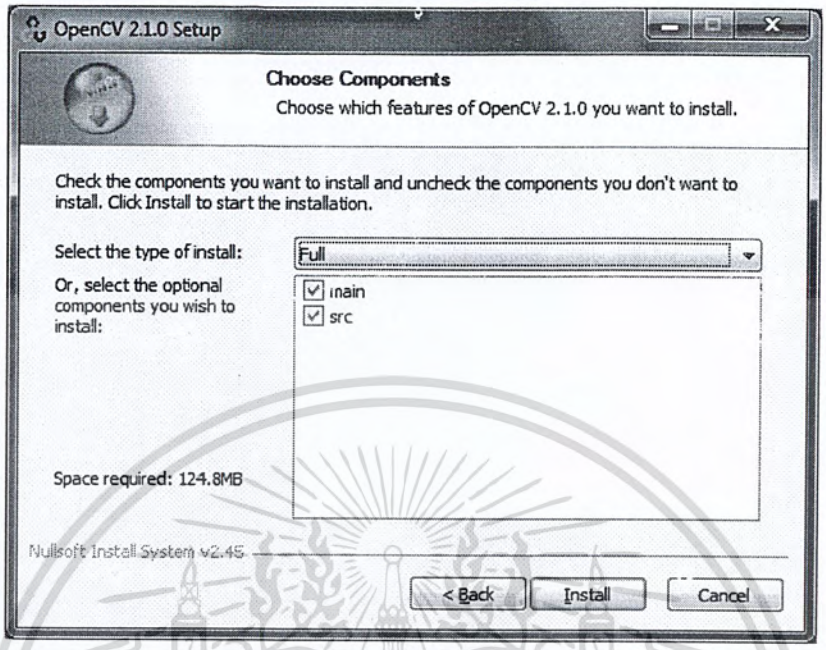
6) คลิก Next



รูป ก.5 หน้าความต้องการ Create shortcuts

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

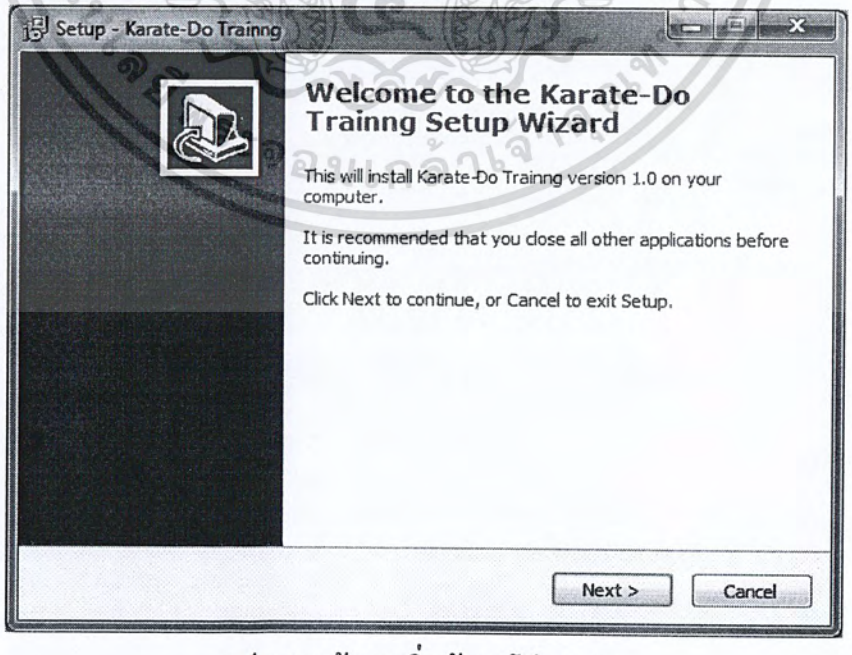
7) คลิก Install



รูป ก.6 หน้าการเลือก Components

8) หลังจากลง OpenCV 2.1.0 แล้ว ให้ลงตัวโปรแกรมต่อ โดยเลือกไฟล์ setup.exe

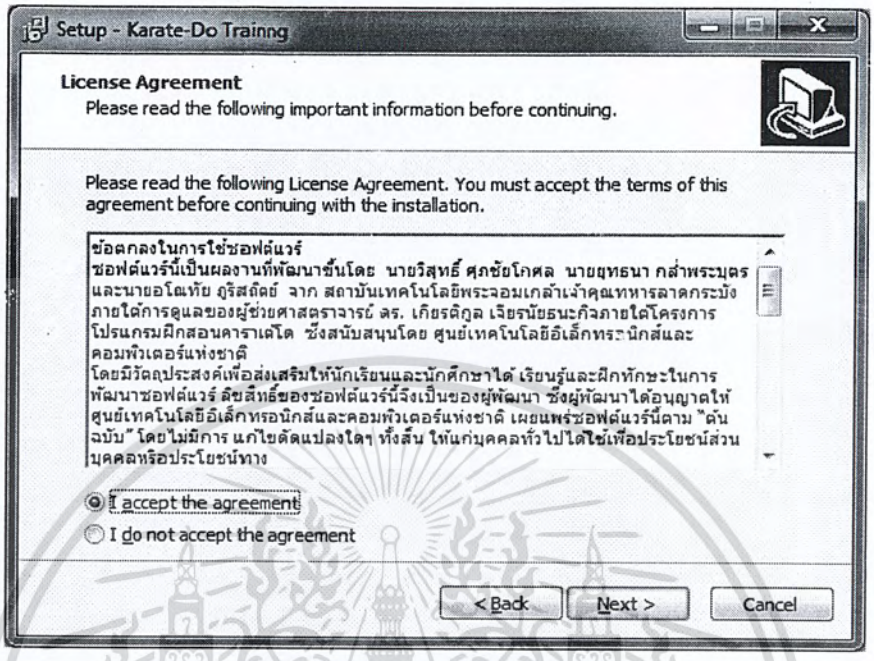
9) คลิก Next



รูป ก.7 หน้าการเริ่มต้นลงโปรแกรม

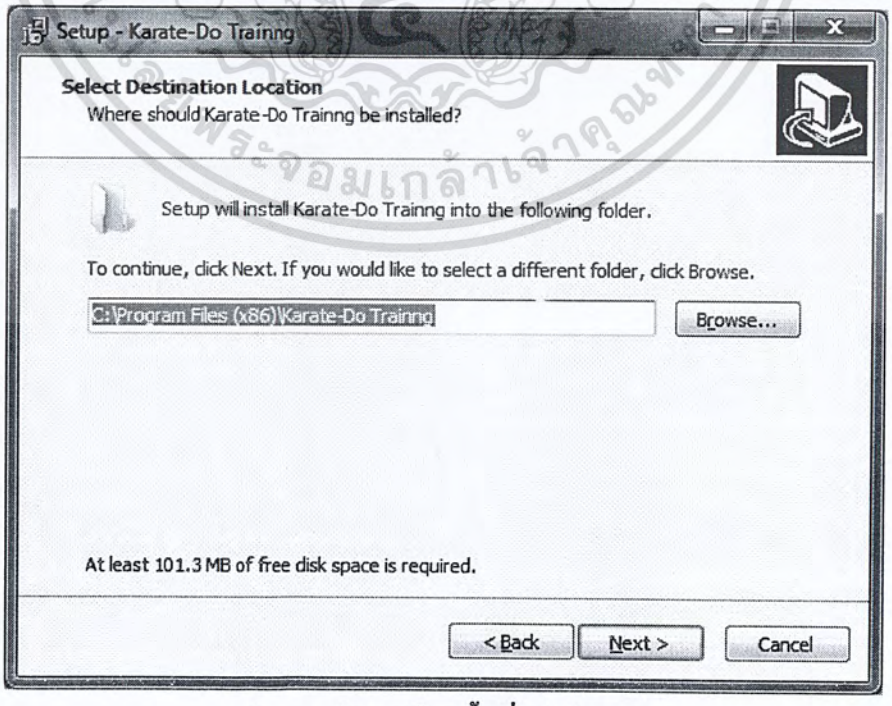
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10) เลือก I accept the agreement แล้วคลิก Next



รูป ก.8 หน้าการยอมรับข้อตกลง

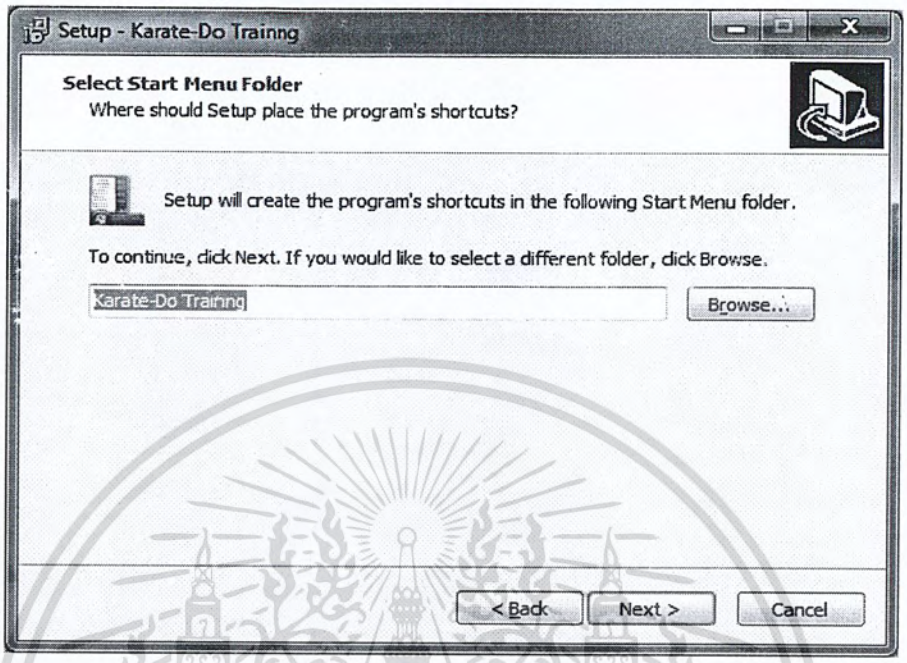
11) เลือกที่ลงโปรแกรม แล้วคลิก Next



รูป ก.9 หน้าการเลือกพื้นที่ลงโปรแกรม

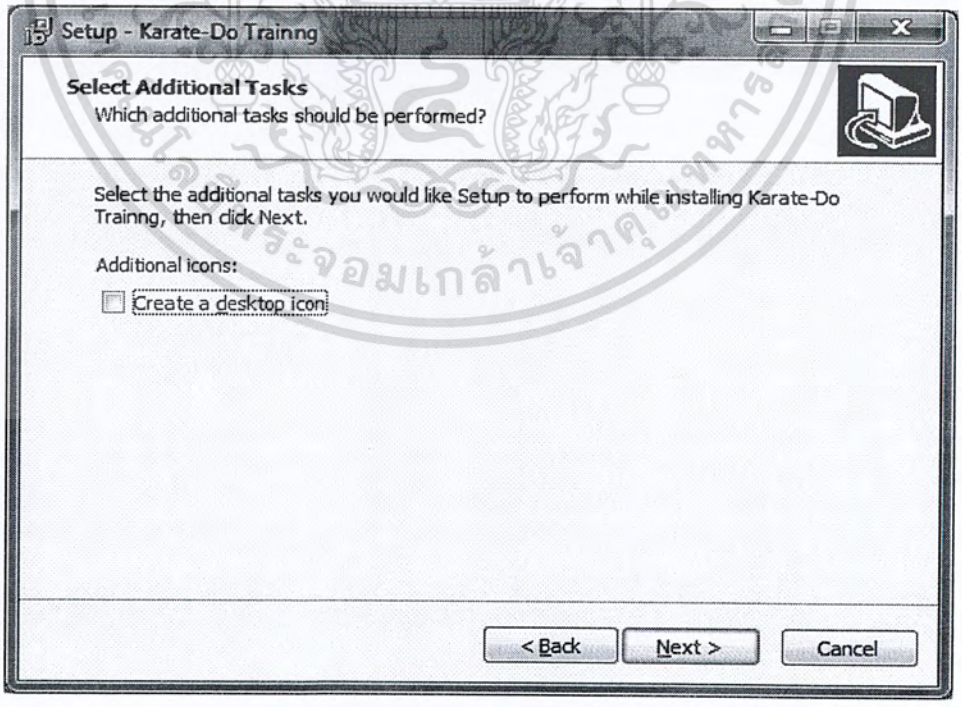
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12) ตั้งชื่อโฟลเดอร์ แล้วคลิก Next



รูป ก.10 หน้าการเลือกโปรแกรม

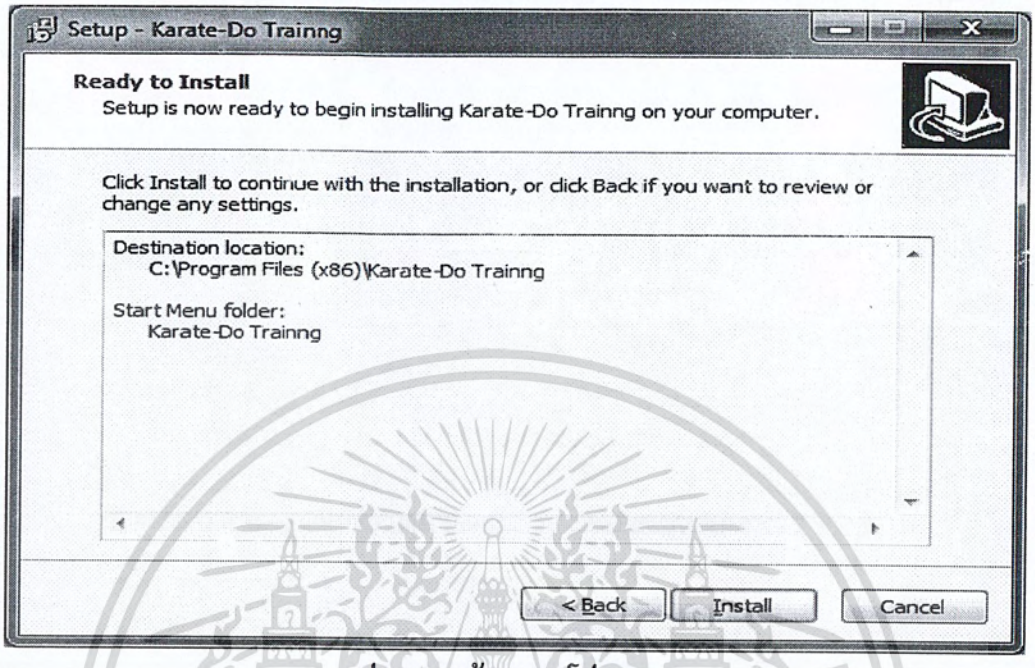
13) เลือกว่าจะ Create a desktop icon หรือไม่ แล้วคลิก Next



รูป ก.11 หน้าการเลือก Create a desktop icon

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

14) คลิก Install แล้วรอจนโปรแกรมลงเสร็จ



รูป ก.12 หน้าการลงโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้