ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

เกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส

INTERACTIVE WEBCAM GAME USING BODY MOTION DETECTOR: MYFITNESS

เมธาวี อยู่อ่อน

นวพร อุปริบุญ

เลขหมู่.....
เลขทะเบียน.....06605
วัน, เดือน, ปี.....28 ก.พ. 2555

b.18388125
i.....

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต
สาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ภาคเรียนที่ 2 ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส

INTERACTIVE WEBCAM GAME USING BODY MOTION DETECTOR: MYFITNESS

โดย

เมธาวี อยู่อ่อน

นวพร อุปริบุญ

อาจารย์ที่ปรึกษา

ดร. สุภวรรณ อันนันหนับ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต

สาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2553

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

INTERACTIVE WEBCAM GAME USING BODY MOTION DETECTOR: MYFITNESS

MAYTHAWEE U-ON

NAWAPORN UPARIBOON

A PROJECT SUBMITTED IN PARTIAL FULLFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/2010

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2011

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองวิทยานิพนธ์ ประจำปีการศึกษา 2553

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง เกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส

INTERACTIVE WEBCAM GAME USING BODY MOTION DETECTOR: MYFITNESS

ผู้จัดทำ

1. นายเมธาวี อยู่อ่อน รหัสนักศึกษา 50070089
2. นางสาวนวพร อุปริบุญ รหัสนักศึกษา 50070094

..... อาจารย์ที่ปรึกษา

(ดร. สุภวรรณ อันนันหนับ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อโครงงาน	เกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส		
นักศึกษา	นายเมธาวี	อยู่อ่อน	รหัสนักศึกษา 50070089
	นางสาวนวพร	อุปริบุญ	รหัสนักศึกษา 50070094
ปริญญา	วิทยาศาสตรบัณฑิต		
สาขาวิชา	เทคโนโลยีสารสนเทศ		
ปีการศึกษา	2553		
อาจารย์ที่ปรึกษา	ดร. สุภวรรณ	อันนันหนับ	

บทคัดย่อ

ปริญญานิพนธ์ฉบับนี้นำเสนอการพัฒนาเกมแอพพลิเคชั่น "เกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส" มุ่งเน้นการพัฒนาเกมแอพพลิเคชั่นที่มีรูปแบบการเล่นที่แตกต่างจากเดิม โดยการใช้การเคลื่อนไหวร่างกายของผู้เล่นในการเล่นและควบคุมเกมซึ่งทำการตรวจจับและวิเคราะห์การเคลื่อนไหวด้วยเทคนิคการประมวลผลภาพ ที่พัฒนาขึ้นด้วยภาษาซีชาร์ป (C#) โดยใช้เฟรมเวิร์ก XNA Game Studio Express (XNA GSE) ที่ใช้สำหรับพัฒนาเกมโดยเฉพาะ ร่วมกับไลบรารีอีเอ็มจียูซีวี (EmguCV) ที่มีหน้าที่ในด้านการประมวลผลภาพ และปรับปรุงรูปภาพแล้วนำผลที่ได้จากการวิเคราะห์มาประมวลผลในส่วนของตัวเกม เพื่อแสดงผลโต้ตอบกับผู้เล่นในรูปแบบของภาพสองมิติและสามมิติพร้อมทั้งมีการวัดผลสมรรถภาพทางกายของผู้เล่นใน 4 ด้านด้วยกัน ได้แก่ ความอดทนกล้ามเนื้อ ความยืดหยุ่น การประสานสัมพันธ์ และความคล่องแคล่ว ซึ่งเกมนี้สามารถช่วยผ่อนคลายความเครียด และบรรเทาความเมื่อยล้าของผู้เล่นจากการใช้เมาส์และคีย์บอร์ดในการเล่นเกมแบบเดิมได้ นอกจากนี้ ผู้เล่นจะได้รับอรรถรสในการเล่นเกมมากขึ้นเสมือนตนเองได้เล่นอยู่ในเกมนั้นจริง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Interactive Webcam Game Using Body Motion Detector : myFitness	
Student	Mr. Maythawee U-on	Student ID 50070089
	Miss Nawaporn Upariboon	Student ID 50070094
Degree	Bachelor of Science	
Program	Information Technology	
Year	2010	
Advisor	Dr. Supawan Annanab	

ABSTRACT

This thesis presents the development of game application "Interactive Webcam Game Using Body Motion Detector: myFitness". In this thesis, we focus on game development which its format plays differently. The game is played and controlled by player's body movement. The image processing technique can detect and analyze movement. Moreover, we implement in C# programming by using XNA Game Studio Express (XNA GSE) framework which it is suit for game development and using EmguCV library processes and enhances images. The results from image processing can be displayed on the screen and interacted with player by graphic 2D and 3D interfaces. From player's movement, the game evaluates and divides player's fitness in 4 categories such as muscular endurance, flexible, co-ordination and agility. The results show this game helps relieve stress and fatigue which is occurred by using mouse and keyboard. Furthermore, players are more active and realistic.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จไปได้ด้วยดี ด้วยความกรุณาของอาจารย์ที่ปรึกษาโครงงาน ดร. สุภวรรณ อันนันหนับ ที่ได้สละเวลาให้ความช่วยเหลือ คำแนะนำในการปรับปรุงแก้ไขปัญหาต่างๆที่เกิดขึ้น ตลอดจนให้ความรู้และประสบการณ์ที่ดีในการทำงาน และขอขอบพระคุณอย่างสูง

ขอกราบขอบพระคุณคณาจารย์คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกๆท่านที่ได้ประสิทธิ์ประสาทวิชาให้กับข้าพเจ้า คอยให้คำปรึกษาและชี้แนะแนวทางในการแก้ไขปัญหาต่างๆให้สำเร็จลุล่วงไปได้

ขอขอบคุณพี่ เพื่อน และน้องคณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ทุกคนที่ให้ความช่วยเหลือ คำแนะนำ และกำลังใจเสมอมา

ขอกราบขอบพระคุณ บิดา มารดา และครอบครัวของข้าพเจ้า ที่ให้กำลังใจ และให้การสนับสนุนในทุกๆเรื่อง ทำให้ข้าพเจ้าสามารถทำโครงงานฉบับนี้สำเร็จลุล่วงด้วยดี

คุณค่าและประโยชน์อันพึงมาจากโครงงานฉบับนี้ ข้าพเจ้าขอมอบแด่ผู้มีพระคุณทุกท่าน

เมธาวี อยู่อ่อน

นวพร อุปริบุญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญรูป VIII

สารบัญตาราง XII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของโครงการ 1

1.3 ขอบเขตของการพัฒนาระบบ 2

1.4 แผนการดำเนินงาน 2

1.5 ขั้นตอนของการศึกษา 3

1.6 ประโยชน์ที่คาดว่าจะได้รับ 3

บทที่ 2 ทฤษฎีการประมวลผลภาพ และการพัฒนาเกม 4

2.1 การประมวลผลภาพดิจิตอล (Digital Image Processing) 4

2.1.1 ภาพดิจิตอล (Digital Image) 4

2.1.2 แบบจำลองสี (Color Models) 7

2.1.3 การแยกส่วนที่สนใจออกจากสิ่งแวดล้อม (Image Subtraction) 11

2.1.4 การแปลงภาพสีให้เป็นภาพขาวดำ (Thresholding) 13

2.1.5 การปรับแต่งภาพด้วยมอร์โฟโลยี (Morphology) 15

2.2 XNA Game Studio Express (XNA GSE) 19

2.2.1 โครงสร้างการทำงานของ XNA Framework และฟังก์ชันที่สำคัญ 20

2.2.2 ข้อดีและข้อเสียของ XNA 22

2.3 การใช้ซอฟต์แวร์ไลบรารี EmguCV ในการประมวลผลภาพ 22

2.3.1 สถาปัตยกรรมของ EmguCV 22

2.3.2 การ Wrapping OpenCV ของ EmguCV 23

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

2.3.3 ข้อดีและข้อเสียของ EmguCV 25
2.4 ระบบฟิสิกส์ (Physics Engine) 25
2.5 การสร้างเอฟเฟกต์ด้วย Mercury Particle Engine 26
2.6 การใส่เสียงประกอบเกมด้วย XACT 27
2.6.1 XACT Features 27
2.7 DirectX 28
2.7.1 DirectX Runtime 28
2.7.2 DirectX SDK (DirectX Software Development Kit) 29
2.8 ระบบพิกัดสามมิติ 30
2.8.1 ทฤษฎีโมเดลสามมิติใน XNA 31
2.8.2 การแสดงผลโมเดลสามมิติ 32
2.9 3D Studio Max (3ds Max) 34
2.9.1 ขั้นตอนการสร้างงานสามมิติ 34

บทที่ 3 การวิเคราะห์และออกแบบระบบ 36
3.1 ความหมายและความสำคัญของสมรรถภาพทางกาย 36
3.1.1 สมรรถภาพทางกายที่สัมพันธ์กับสุขภาพ (Health – Related Physical Fitness) 36
3.1.2 สมรรถภาพทางกายที่สัมพันธ์กับทักษะ (Skill – Related PhysicalFitness) .. 37
3.1.3 ความสำคัญของสมรรถภาพทางกาย 38
3.2 วิธีการทดสอบสมรรถภาพ และการวิเคราะห์ความเป็นไปได้ของเกม 38
3.3 การออกแบบรูปแบบเกม (Game Design) 42
3.3.1 เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ 42
3.3.2 เกมทดสอบสมรรถภาพการประสานสัมพันธ์ 42
3.3.3 เกมทดสอบสมรรถภาพความยืดหยุ่น 43
3.3.4 เกมทดสอบสมรรถภาพความคล่องแคล่ว 43
3.3.5 ข้อจำกัดของเกม 43
3.4 ความต้องการของระบบ 43

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.4.1 ความต้องการที่เป็นฟังก์ชันการทำงาน (Functional Requirements) 43

3.4.2 ความต้องการที่ไม่ได้เป็นฟังก์ชันการทำงาน (Non-Functional Requirements) 44

3.5 การออกแบบระบบ 44

3.5.1 แผนภาพยูสเคส (Use Case Diagram) 44

3.5.2 รายละเอียดยูสเคส (Use Case Description) 46

3.5.3 แผนภาพกิจกรรม (Activity Diagram) 52

3.5.4 การออกแบบส่วนติดต่อกับผู้ใช้ 58

บทที่ 4 การพัฒนาระบบ 60

4.1 การพัฒนาส่วนประมวลผลภาพ (Image Processing) 60

4.1.1 การเขียนโปรแกรมเพื่อใช้งานซอฟต์แวร์ไลบรารี EmguCV 60

4.1.2 การทำการประมวลผลภาพ 61

4.2 การพัฒนาส่วนตัวเกม (Game Application) 66

4.2.1 การแปลงข้อมูลภาพจากกล้องเว็บแคมเพื่อแสดงผลใน XNA Game Studio Express 66

4.2.2 การแสดงผลกราฟฟิกสามมิติ 67

4.2.3 การตรวจสอบการชนของวัตถุกับผู้เล่น 68

4.2.4 การเขียนโปรแกรมเพื่อใช้งานระบบฟิสิกส์ FarseerPhysics Engine 69

4.2.5 การแปลงพิกัดระบบสองมิติเป็น สามมิติ 71

4.2.6 การแปลงพิกัดระบบสามมิติเป็นสองมิติ 72

4.2.7 การเขียนโปรแกรมเพื่อใช้งาน Particle Effect ด้วย Mercury Particle Engine 72

4.2.8 การเขียนโปรแกรมเพื่อจัดการไฟล์เสียงโดยใช้ XACT 77

4.2.9 รูปแบบเกม 79

4.3 การพัฒนาส่วนติดต่อกับผู้ใช้ (Interface) 82

4.3.1 ส่วนติดต่อกับผู้ใช้ที่ควบคุมผ่านเมาส์และคีย์บอร์ด 82

4.3.2 ส่วนติดต่อกับผู้ใช้ที่ควบคุมด้วยการสัมผัส 91

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 5 สรุปผลโครงงาน และข้อเสนอแนะ 96

5.1 สรุปผลโครงงาน 96

5.2 ปัญหาและข้อจำกัดในการทำงานของระบบ 97

5.3 ข้อเสนอแนะในการพัฒนาระบบต่อไป 98

บรรณานุกรม 99

ภาคผนวก 101

ภาคผนวก ก คู่มือการติดตั้งโปรแกรม 103

ภาคผนวก ข การวิเคราะห์ข้อมูลและออกแบบเกณฑ์สมรรถภาพ 109

ประวัติผู้เขียน 112

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 ภาพในความหมายของฟังก์ชันสองมิติ g(x, y) ของค่าความเข้มแสง 4
2.2 ภาพไบนารี่ (Binary Image) 5
2.3 ภาพระดับสีเทา (Gray Scale Image) 5
2.4 ภาพสี (Color Image) 6
2.5 ภาพแบบดัชนี (Index Image) 6
2.6 สี RGB 7
2.7 แบบจำลองสี RGB 8
2.8 แบบจำลองสี Grayscale 8
2.9 แบบจำลองสี HSI 9
2.10 สี HSI 10
2.11 อัลกอริธึมการหาความแตกต่างระหว่างเฟรม 11
2.12 การนำ Frame Differencing มาใช้งาน 12
2.13 ขั้นตอนการทำงานของการลบฉากหลัง 12
2.14 การทำงานของ Thresholding 14
2.15 ผลลัพธ์จากการกำหนดค่าเทรชโฮลที่ไม่เหมาะสม 14
2.16 การทำ Dilation 16
2.17 การนำ Dilation มาประยุกต์ใช้งาน 16
2.18 การทำ Erosion 17
2.19 การนำ Erosion มาประยุกต์ใช้งาน 18
2.20 การทำ Opening 18
2.21 การทำ Closing 19
2.22 ลำดับการทำงานของ XNA Framework 20
2.23 สถาปัตยกรรมของไลบรารี EmguCV 23
2.24 การใช้งาน Farseer Physics Engine 26
2.25 เอฟเฟกต์ต่างๆที่แสดงบนโปรแกรม Editor ของ Mercury Particle Engine 27
2.26 การทำงานของ DirectX 29
2.27 วัตถุที่เกิดจากการรวมจุด Vertex และ Polygon 30
2.28 ระบบพิกัดมือซ้าย และมือขวา 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

2.29 ภาพแบบ Perspective 31
2.30 ภาพแบบ Orthographic 32
2.31 พื้นที่การมองในโลกสามมิติ 33
2.32 โปรแกรม 3ds Max 2011 34
2.33 โมเดลสามมิติที่เสร็จสมบูรณ์ 35
3.1 การวัดความหนาของไขมันใต้ผิวหนัง 38
3.2 ดันพื้น 30 วินาที 39
3.3 นั่งงอตัวไปข้างหน้า 39
3.4 ดึงข้อ หรืองอแขนห้อยตัว 40
3.5 วิ่งเร็ว 40
3.6 วิ่งแตะสลับ 41
3.7 การตอบสนองต่อปุ่มไฟ 41
3.8 รูปแบบของการเล่นเกมผ่านกล้องเว็บแคม 42
3.9 แผนภาพยูสเคสของระบบ 45
3.10 แผนผังการทำงานของส่วนติดต่อกับผู้ใช้ (UI Flow) 59
4.1 การเพิ่มไลบรารี EmguCV ในโปรเจค 60
4.2 การสร้างแบบจำลองฉากหลัง 62
4.3 ผลลัพธ์ที่ได้จากการหาค่าสัมบูรณ์ของความแตกต่าง 62
4.4 การแยกส่วนที่สนใจออกจากฉากหลัง 63
4.5 การกำจัดสัญญาณรบกวน 64
4.6 ผลลัพธ์การหาความแตกต่างระหว่างเฟรม 65
4.7 แบบจำลองแสดงการเก็บข้อมูลสีของ XNA 66
4.8 ส่วนประกอบของวัตถุสามมิติ 68
4.9 การตรวจสอบการชน 69
4.10 การเพิ่มไลบรารีFarseer Physic Engine ในโปรเจค 70
4.11 การแปลงพิกัดระบบสองมิติเป็นสามมิติ 72
4.12 การเพิ่มไลบรารี Mercury Particle Engine ในโปรเจค 73
4.13 การเริ่มสร้างเอฟเฟกต์แบบ Basic Emitter 75

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

4.14 การกำหนดค่าต่างๆให้กับเอฟเฟกต์ 75
4.15 การเลือก Texture ให้เอฟเฟกต์ 76
4.16 การกำหนด Modifiers ให้เอฟเฟกต์ 76
4.17 ผลลัพธ์เอฟเฟกต์ที่สร้างขึ้นเอง 77
4.18 เริ่มการสร้างไฟล์เสียง 77
4.19 การลากไฟล์ Wave Bank ไปยัง Sound Bank 78
4.20 การเพิ่ม XACT ในโปรเจค 78
4.21 เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ 80
4.23 เกมทดสอบสมรรถภาพความยืดหยุ่น 81
4.24 เกมทดสอบสมรรถภาพความคล่องแคล่ว 82
4.25 การแยก Layer ของ Object 83
4.26 การเลือกสี 83
4.27 ฉากของเกมที่เสร็จสมบูรณ์ 84
4.28 หน้า Main Menu 85
4.29 หน้า How To Play 85
4.30 หน้า Leader Board 86
4.31 หน้า Credit 86
4.32 หน้า Select Profile 87
4.33 หน้า Create New Profile 87
4.34 หน้า BMI Detail 88
4.35 หน้า BMI Calculate 88
4.36 หน้า Profile Stat 89
4.37 หน้า Background Subtraction Config 89
4.38 หน้า Graph Result 90
4.39 หน้า Congratulated 90
4.40 การสร้างปุ่มสัมผัสในเกม 91
4.41 การกำหนด Material ID 92
4.42 Material Editor ในการทำงานแบบ Multi/Sub-Object 93

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่ **หน้า**

4.43 โมเดลที่ใส่ Material แบบ Multi/Sub-Object 93
4.44 หน้า Tutorial 94
4.45 หน้า Calibrate 94
4.46 หน้าเกม 95
4.47 ไอคอนของแต่ละเกม 95

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ **หน้า**

2.1 แสดงการเชื่อมโยงโครงสร้างข้อมูลระหว่าง EmguCV และ OpenCV 24
2.2 แสดงการเชื่อมโยงโครงสร้างข้อมูลระหว่าง .NET และ OpenCV 24
2.3 พารามิเตอร์ของเมธอด Matrix.CreateLookAt 32
2.4 พารามิเตอร์ของ Matrix.CreatePerspectiveFieldOfView 33
3.1 รายละเอียดยูสเคส Play Game 46
3.2 รายละเอียดยูสเคส Image Processing 47
3.3 รายละเอียดยูสเคส Create Profile 48
3.4 รายละเอียดยูสเคส Delete Profile 49
3.5 รายละเอียดยูสเคส View Data 50
3.6 รายละเอียดยูสเคส Query Frame 51
3.7 แสดงแผนภาพกิจกรรมการเล่นเกม 52
3.8 แสดงแผนภาพกิจกรรมการประมวลผลภาพ 53
3.9 แสดงแผนภาพกิจกรรมการสร้างข้อมูลผู้เล่น 54
3.10 แสดงแผนภาพกิจกรรมการลบข้อมูลผู้เล่น 55
3.11 แสดงแผนภาพกิจกรรมการดูข้อมูลสถิติการเล่นเกม 56
3.12 แสดงแผนภาพกิจกรรมการรับภาพจากกล้องเว็บแคม 57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

"เกม" (Game) เป็นซอฟต์แวร์คอมพิวเตอร์ชนิดหนึ่งที่ถูกสร้างขึ้นเพื่อความบันเทิงเป็นหลัก จากเดิมผู้เล่นต้องเล่นโดยใช้เครื่องเล่นเกม เช่น เกมบอย (Gameboy) หรือเพลย์สเตชั่น (Playstation) ซึ่งลักษณะของการเล่นเกมนี้ ผู้เล่นจะต้องเล่นและควบคุมเกมผ่านแผงบังคับที่มีลักษณะเป็นปุ่มต่างๆ แต่ในปัจจุบันเทคโนโลยีได้พัฒนาไปมาก รวมถึงคอมพิวเตอร์ได้เข้ามามีบทบาทในชีวิตประจำวันมากขึ้นทำให้การเล่นเกมในปัจจุบันไม่ได้ถูกจำกัดเพียงแค่บนเครื่องเล่นเกมเท่านั้น แต่ยังสามารถเล่นบนคอมพิวเตอร์ได้ ทำให้ผู้พัฒนาเกมได้หันมาพัฒนาเกมบนเครื่องคอมพิวเตอร์เพิ่มขึ้นอย่างต่อเนื่อง สังเกตได้จากรูปแบบของเกมที่มีความแปลกใหม่ ลักษณะเกมที่มีความหลากหลาย หรือรูปแบบการเล่น ที่ทำให้ผู้เล่นได้รับอรรถรสในการเล่นเกมมากขึ้นเช่น การใช้เทคนิคภาพสมจริงโดยใช้ภาพแอนิเมชั่น ที่จะทำให้ผู้เล่นรู้สึกเสมือนได้เล่นอยู่ในเกมนั้นจริงๆ โดยผู้เล่นเกมจะควบคุมผ่านเมาส์หรือคีย์บอร์ด

จากที่กล่าวมาข้างต้น ผู้จัดทำโครงการจึงมีความประสงค์ที่จะพัฒนาเกมแอพพลิเคชั่นที่ใช้การเคลื่อนไหวของผู้เล่นในการเล่นเกม แทนการใช้เครื่องเล่นเกมหรือเมาส์หรือคีย์บอร์ดอย่างเช่นที่ผ่านมาโดยกล้องเว็บแคม (Web Camera) เป็นอุปกรณ์ที่เพิ่มขึ้นมาเพื่อตรวจจับการเคลื่อนไหวของผู้เล่นทุกส่วนของร่างกาย และวิเคราะห์การเคลื่อนไหวร่างกายของผู้เล่น ออกมาเป็นผลเกมตอบสนองแก่ผู้เล่น พร้อมกันนี้จะมีการสร้างและพัฒนาซอฟต์แวร์ที่เน้นให้ผู้เล่นได้ความบันเทิงและการออกกำลังในเวลาเดียวกัน ซึ่งจะช่วยให้ผู้เล่นเกมได้รับความสนุกสนานสามารถคลายเครียดและบรรเทาความเมื่อยล้าจากการใช้เมาส์และคีย์บอร์ดในการเล่นเกมแบบเดิมๆได้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อศึกษาความรู้ด้านการประมวลผลภาพ การพัฒนาเกมแอพพลิเคชั่น และการสร้างคอมพิวเตอร์กราฟฟิก
2. เพื่อออกแบบและพัฒนาเกม ที่ผู้เล่นสามารถใช้การเคลื่อนไหวทุกส่วนของร่างกายในการเล่นเกมได้
3. เพื่อสร้างรูปแบบในการเล่นเกมที่แตกต่างไปจากเดิม
4. เพื่อให้ผู้เล่นรู้สึกเสมือนจริง และสนุกสนานในการเล่นเกม เนื่องจากใช้การควบคุมเกมผ่านการเคลื่อนไหวของตนเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.3 ขอบเขตของการพัฒนาระบบ

ระบบที่พัฒนาคือเกมแอพพลิเคชั่น ที่ให้ผู้เล่นใช้การเคลื่อนไหวของร่างกายในการควบคุมเกม เพื่อให้บรรลุเป้าหมายที่กำหนดไว้ในเกม โดยอินพุทของตัวเกมคือท่าทางการเคลื่อนไหวที่ได้จากส่วนประมวลผลภาพแทนการใช้เมาส์และคีย์บอร์ด ซึ่งมีคุณลักษณะดังนี้

1. ใช้กล้องเว็บแคมเป็นอุปกรณ์จับภาพร่างกายของผู้เล่น ซึ่งจะตั้งอยู่กับที่ และนำเข้าภาพการเคลื่อนไหวของผู้เล่นแบบ Real-Time
2. พัฒนาส่วนประมวลผลภาพ เพื่อทำการตรวจจับการเคลื่อนไหวของผู้เล่น ที่จะเน้นให้ผู้เล่นได้เคลื่อนไหวร่างกายมากขึ้นกว่าการนั่งเล่นเกมแบบปกติ
3. สภาพแวดล้อมขณะเล่นเกมในส่วนของฉากหลังจะต้องไม่มีการเคลื่อนไหวใดๆ อีกทั้งสภาพแสงภายในห้องจะต้องไม่มืดหรือสว่างเกินไป
4. ซอฟต์แวร์ที่พัฒนาจะครอบคลุมการบริหารส่วนต่างๆของร่างกาย
5. เป็นเกมสำหรับผู้เล่น 1 คน

1.4 แผนการดำเนินงาน

ภาคเรียนที่ 1

- ศึกษาทฤษฎีการประมวลผลภาพ
- ศึกษา XNA Framework
- ศึกษาไลบรารี EmguCV
- ศึกษาการสร้างกราฟฟิกสามมิติด้วย 3ds Max 2011
- ศึกษาการบริหารร่างกายเพื่อสุขภาพ
- ออกแบบส่วนประมวลผลภาพ และรูปแบบเกม

ภาคเรียนที่ 2

- พัฒนาส่วนประมวลผลภาพ
- พัฒนาตัวเกม
- ทดสอบ และแก้ไขข้อผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนของการศึกษา

โครงการนี้ได้แบ่งเนื้อหาออกเป็น 5 บทด้วยกัน ได้แก่

บทที่ 1 กล่าวถึงความเป็นมาและความสำคัญของปัญหา วัตถุประสงค์ของโครงการ ขอบเขตของการศึกษา ขั้นตอนการศึกษา และประโยชน์ที่คาดว่าจะได้รับ

บทที่ 2 กล่าวถึงทฤษฎีพื้นฐานที่เกี่ยวข้องกับโครงการ และการพัฒนาเกม ซึ่งประกอบด้วย การประมวลผลภาพดิจิตอล (Digital Image Processing) การพัฒนาเกมด้วย XNA Framework การใช้งานไลบรารี EmguCV และการสร้างกราฟฟิกสามมิติ

บทที่ 3 กล่าวถึงการวิเคราะห์และออกแบบระบบ ซึ่งประกอบด้วย รายละเอียดความต้องการของระบบ (Functional and Non-Functional Requirements) ยูสเคสไดอะแกรม (Use Case Diagram) รายละเอียดยูสเคส (Use Case Description) การออกแบบเกม (Game Design) รวมถึงการออกแบบส่วนติดต่อกับผู้ใช้ (User Interface Design)

บทที่ 4 กล่าวถึงรายละเอียดของการพัฒนาระบบ ประกอบด้วย การพัฒนาในส่วนการประมวลผลภาพ ส่วนตัวเกม และส่วนติดต่อกับผู้ใช้

บทที่ 5 กล่าวถึงผลสรุปของโครงการ และข้อเสนอแนะ

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้เรื่องการประมวลผลภาพ การพัฒนาเกมแอพพลิเคชั่น และการสร้างคอมพิวเตอร์กราฟฟิก ซึ่งสามารถนำไปประยุกต์ใช้กับงานด้านต่างๆได้ในอนาคต
2. ได้แนวความคิดในการพัฒนาเกมในรูปแบบใหม่ที่ยังไม่เคยพัฒนามาก่อน
3. ผู้เล่นได้รับความสนุกสนาน และเพลิดเพลินไปกับเกม ถือเป็นการผ่อนคลายความเครียดได้
4. การใช้ร่างกายของผู้เล่นในการควบคุมเกม จะช่วยให้ผู้เล่นได้เคลื่อนไหวร่างกายมากขึ้น เพื่อลดผลกระทบที่จะเกิดขึ้นต่อสุขภาพในอนาคต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

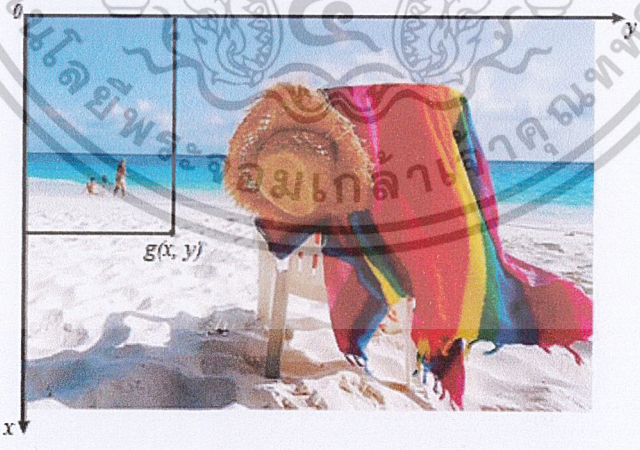
บทที่ 2
ทฤษฎีการประมวลผลภาพ และการพัฒนาเกม

2.1 การประมวลผลภาพดิจิตอล (Digital Image Processing)

การประมวลผลภาพดิจิตอลโดยทั่วไปแล้วหมายถึง การประมวลผลภาพดิจิตอลสองมิติโดยใช้เครื่องคอมพิวเตอร์ หรือหมายถึง การประมวลผลทางดิจิตอลของข้อมูลสองมิติใดๆ ซึ่งการประมวลผลภาพดิจิตอลนี้ถูกนำมาประยุกต์ใช้ในงานด้านต่างๆอย่างกว้างขวาง เช่น การวิเคราะห์ภาพถ่ายทางอากาศ การประมวลผลทางการแพทย์ การประมวลผลภาพถ่ายด้วยเรดาร์ งานด้านหุ่นยนต์ หรือการตรวจสอบอัตโนมัติของชิ้นส่วนทางอุตสาหกรรม เป็นต้น

2.1.1 ภาพดิจิตอล (Digital Image)

ภาพดิจิตอล (Digital Image) คือ รูปที่เก็บอยู่ในรูปแบบของดิจิตอล ที่แสดงผลภาพในลักษณะสองมิติในหน่วยที่เรียกว่า พิกเซล (Pixel) ซึ่งเป็นฟังก์ชันไม่ต่อเนื่อง g(x, y) ของค่าความเข้มแสง ที่เกิดจากการแปลงภาพด้วยวิธี Image Digitization มาจากฟังก์ชันต่อเนื่อง f(x, y) เพื่อให้สามารถทำการประมวลผลด้วยคอมพิวเตอร์ได้ โดยที่ x และ y คือพิกัดเชิงพื้นที่ (Spatial Coordinates) และค่าฟังก์ชัน g ณ ตำแหน่ง (x, y) ใดๆ คือความเข้มแสงของภาพ (Intensity) และเมื่อฟังก์ชัน g เป็นค่าจำกัด (Finite Value) ด้วยแล้ว จะเรียกรูปภาพนั้นว่า “ภาพดิจิตอล”



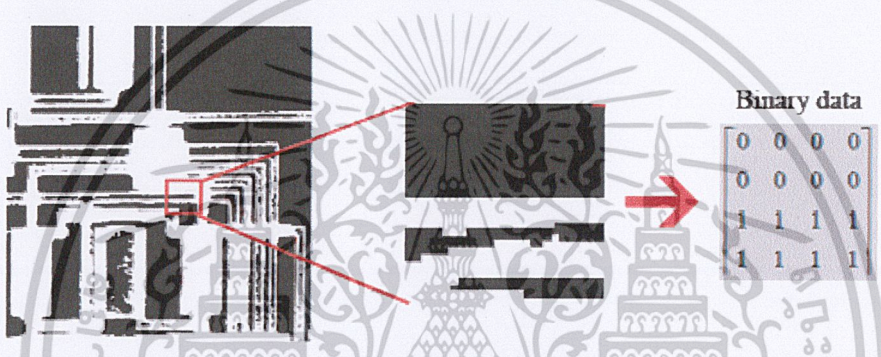
รูปที่ 2.1 ภาพในความหมายของฟังก์ชันสองมิติ g(x, y) ของค่าความเข้มแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพดิจิตอลสามารถสร้างได้ด้วยวิธีการต่างๆ เช่น การกราดภาพ (Scan) การถ่ายภาพด้วยกล้องดิจิตอล การสร้างภาพด้วยคอมพิวเตอร์ หรือรูปที่เกิดจากกระบวนการทางแสง ซึ่งเกิดจากพลังงานคลื่นแม่เหล็กไฟฟ้า (Electromagnetic Spectrum) หลายช่วงความถี่ โดยทั่วไปสามารถแบ่งประเภทของภาพดิจิตอลตามคุณสมบัติของระบบสีได้ ดังนี้

2.1.1.1 ภาพไบนารี่ (Binary Image)

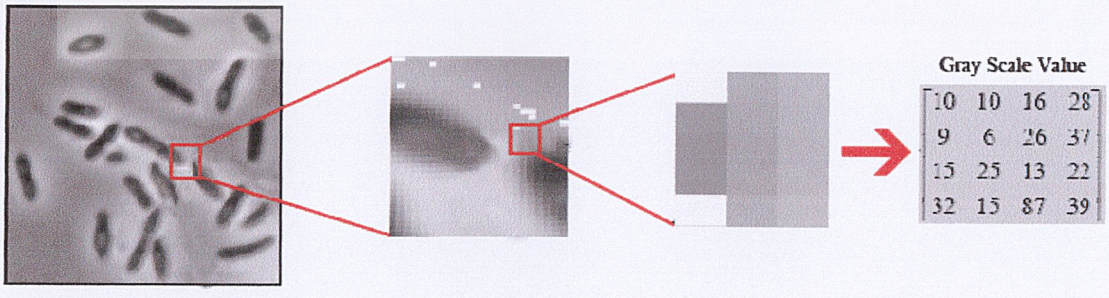
คือภาพดิจิตอลที่แต่ละพิกเซลจะแสดงด้วยค่าไบนารี่ 1 บิท ซึ่งประกอบด้วยค่า 1 แทนพิกเซลสีขาว และค่า 0 แทนพิกเซลสีดำ บางครั้งเรียกว่า ภาพสองระดับ (Bi-level) หรือภาพขาวดำ (Black and White) ภาพประเภทนี้เหมาะสำหรับรูปที่เกี่ยวกับตัวอักษร (Text) ภาพลายนิ้วมือ (Finger print) เป็นต้น



รูปที่ 2.2 ภาพไบนารี่ (Binary Image)

2.1.1.2 ภาพระดับสีเทา (Gray Scale Image or Intensity Image)

ลักษณะของภาพชนิดนี้ในแต่ละพิกเซลจะมีค่าความเข้มของแสงในระดับที่แตกต่างกันตั้งแต่สีขาวไปยังสีดำ บางครั้งเรียกว่า ภาพสีเดียว (Monochrome Image) โดยปกติภาพระดับสีเทาจะมีความละเอียด (Resolution) เท่ากับ 8 บิท ซึ่งระดับความเข้มของแสงสีดำเท่ากับ 0 และค่าความเข้มของแสงสีขาวเท่ากับ 255

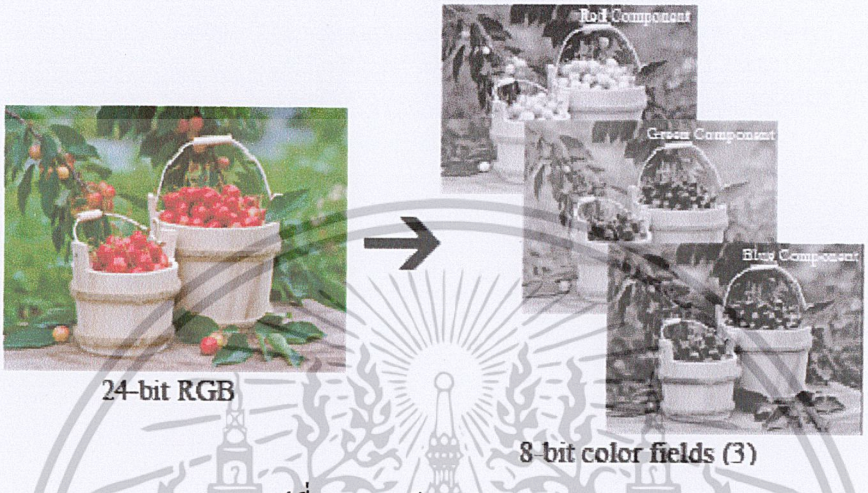


รูปที่ 2.3 ภาพระดับสีเทา (Gray Scale Image)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.3 ภาพสี (Color Imageor RGB Image)

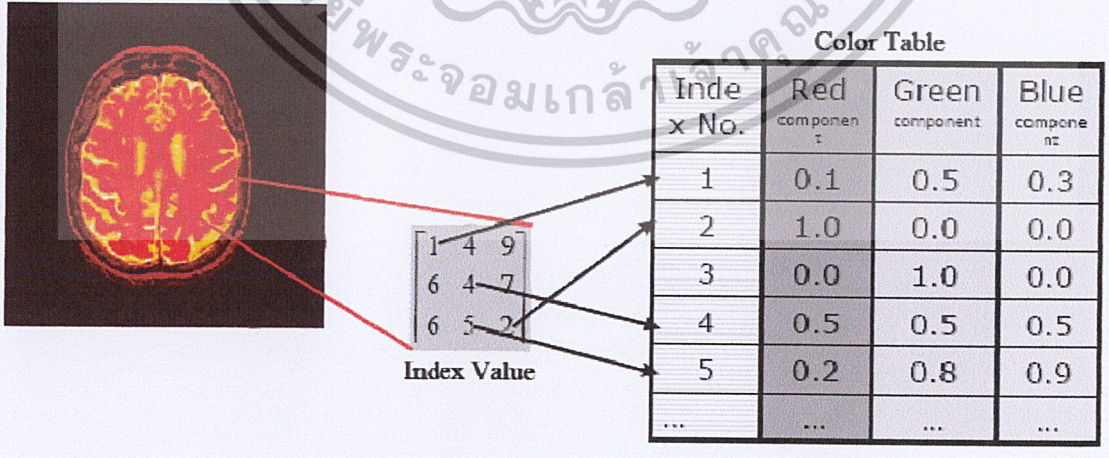
ภาพชนิดนี้แต่ละพิกเซลจะเก็บค่าระดับความเข้มของแต่ละแถบแสงของแม่สีหลัก 3 สีคือ สีแดง (Red) สีเขียว (Green) สีน้ำเงิน (Blue) มาผสมกันในอัตราส่วนที่ต่างกันเพื่อประกอบกันเป็นภาพหนึ่งภาพ



รูปที่ 2.4 ภาพสี (Color Image)

2.1.1.4 ภาพแบบดัชนี (Index Image)

ในแต่ละพิกเซลของภาพชนิดนี้จะเก็บค่าดัชนี (Index Number) ซึ่งเป็นเลขจำนวนเต็ม เพื่อนำมาเทียบกับตารางสี (Color Table) ที่แสดงค่าแสงสีแดง เขียว และน้ำเงิน ซึ่งค่าดัชนีนี้จะเป็นตัวชี้ให้เห็นว่าภาพในตำแหน่งพิกเซลนั้นๆมีค่าอัตราส่วนของแสง R G B ในอัตราส่วนเท่าไร



รูปที่ 2.5 ภาพแบบดัชนี (Index Image)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2 แบบจำลองสี (Color Models)

สีเป็นคุณสมบัติทางกายภาพอย่างหนึ่งของวัตถุ เกิดจากการรับรู้สัญญาณในรูปของคลื่นแสงที่ตกกระทบตามนุษย์ และส่งสัญญาณผ่านประสาทตาไปยังสมอง จากนั้นจึงแปลงค่าเป็นสีต่างๆ แบบจำลองสีเป็นวิธีกำหนดสีต่างๆให้เป็นมาตรฐาน ซึ่งแบบจำลองแต่ละแบบก็มีคุณสมบัติแตกต่างกันออกไป จึงเหมาะสำหรับการใช้งานที่แตกต่างกัน ในแบบจำลองสีนั้นจะใช้สีหลัก หรือแม่สีมาผสมกันเพื่อให้เกิดการแสดงค่าสีอื่นๆ โดยแม่สีหลักจะแตกต่างกันตามโมเดลสีแต่ละชนิด

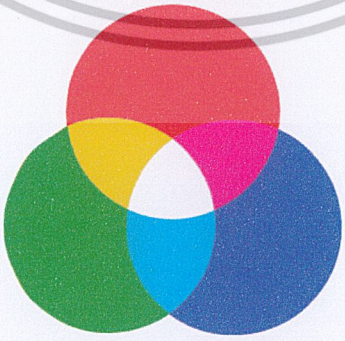
2.1.2.1 แบบจำลองสี RGB

แบบจำลองสี RGB เป็นระบบสีที่ใช้กันทั่วไป ประกอบด้วยแม่สีหลัก 3 สี ที่เกิดจากการรวมตัวกันของแสงแบบบวก (Additive Color) คือ สีแดง สีเขียว และสีน้ำเงิน ซึ่งแต่ละแม่สีจะมีขนาดเป็น 8 บิท ในความลึกของแต่ละพิกเซลมีขนาดเป็น 24 บิท แต่ละแม่สีจะมีค่าตั้งแต่ 0 ถึง 255 ดังนั้นจึงสามารถสร้างสีได้ถึง $256 \times 256 \times 256$ เท่ากับ 16.7 ล้านสี ระบบสี RGB มักถูกนำมาใช้ในงานด้านการแสดงผลข้อมูลบนจอภาพ รวมไปถึงการเก็บข้อมูลภาพในระบบคอมพิวเตอร์ ค่าสี RGB แต่ละค่าสามารถหาได้จากสมการ ดังนี้

$$r = \frac{R}{(R + B + G)} \tag{2.1}$$

$$g = \frac{G}{(R + B + G)} \tag{2.2}$$

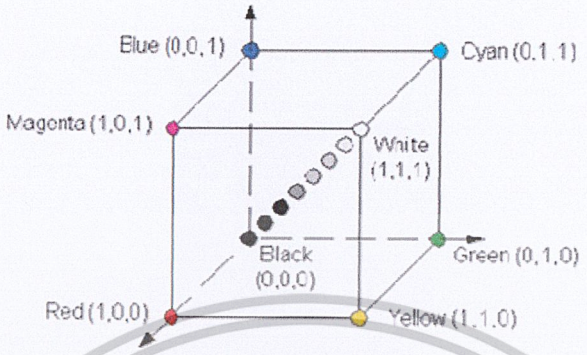
$$b = \frac{B}{(R + B + G)} \tag{2.3}$$



รูปที่ 2.6 สี RGB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบจำลองสี RGB สามารถนำมาแสดงในระบบพิกัดปริภูมิสี (Color Space) ในแบบลูกบาศก์ได้ดังรูปที่ 2.7



รูปที่ 2.7 แบบจำลองสี RGB

2.1.2.2 แบบจำลองสี Grayscale

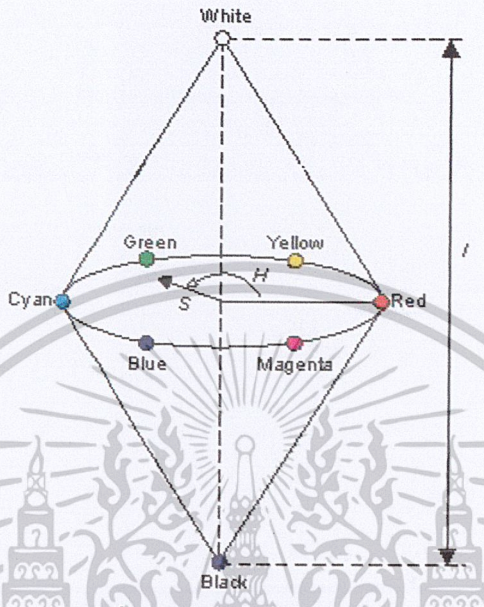
เป็นภาพโทนสีเทาที่เกิดจากการกำหนดค่าให้กับความเข้มแสงของแต่ละจุดภาพ ซึ่งเป็นระดับของสีเทาล้วนที่มีการเพิ่มระดับของโทนสีเข้มขึ้นเรื่อยๆ จากสีขาวจนถึงสีดำมีทั้งหมด 256 ระดับ โดยสีขาวจะถูกแทนด้วยเลข 255 และสีดำจะถูกแทนค่าด้วยเลข 0

รูปที่ 2.8 แบบจำลองสี Grayscale

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.3 แบบจำลองสี HIS

แบบจำลองสีนี้เป็นแบบจำลองสีที่สอดคล้องกับการรับรู้ และการมองเห็นค่าสีต่างๆของมนุษย์ ประกอบด้วยลักษณะของสี 3 ลักษณะ คือ



รูปที่ 2.9 แบบจำลองสี HSI

1) **H (Hue)** เป็นค่าที่บอกถึงความเป็นสีจริงซึ่งเกิดจากการนำแม่สีแดง สีเขียว และสีน้ำเงินมาผสมกัน ถูกวัดโดยตำแหน่งการแสดงสีบน Standard Color Wheel ซึ่งเป็นส่วนของมุมองศาที่ทำมุมกับรอบแกนแนวตั้ง มีช่วงตั้งแต่ 0 องศา จนถึง 360 องศา โดยมีระดับสีทีละ 60 องศา ซึ่งแม่สีหลัก H = 0 คือค่าสีแดง H = 120 คือค่าสีเขียว และ H = 240 คือค่าสีน้ำเงิน หมุนกันจนเป็นรูปหกเหลี่ยม
2) **S (Saturation)** คือ คุณสมบัติที่บ่งบอกถึงความเข้มข้นและความเจือจางของสี ที่ถูกทำให้เจือจางด้วยสีขาว ซึ่งมีค่าอยู่ระหว่าง 0 ถึง 1 ถ้าถูกวัดโดยตำแหน่งบน Standard Color Wheel แล้ว ค่า Saturation จะเพิ่มขึ้นจากจุดกึ่งกลางจนถึงเส้นขอบ โดยค่าที่เส้นขอบจะมีสีที่ชัดเจนและอิ่มตัวที่สุด
3) **I (Intensity)** คือ คุณสมบัติที่ใช้อธิบายความเข้ม หรือความสว่างของสี โดยระดับที่สว่างที่สุดของทุกสีคือสีขาว I = 1 และระดับที่มืดที่สุดของทุกสีคือสีดำ I = 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยแบบจำลองสี HSI มีความสัมพันธ์กับแบบจำลอง RGB ดังสมการต่อไปนี้

$$I = \frac{1}{3}(R+G+B) \tag{2.4}$$

$$S = 1 - \frac{3\min(R,G,B)}{R+G+B} \tag{2.5}$$

$$H = \cos^{-1}\left[\frac{\frac{1}{2}[(R-G)+(R-B)]}{\sqrt{(R-G)^2+(R-B)(G-B)}}\right] \tag{2.6}$$

รูปที่ 2.10 สี HSI

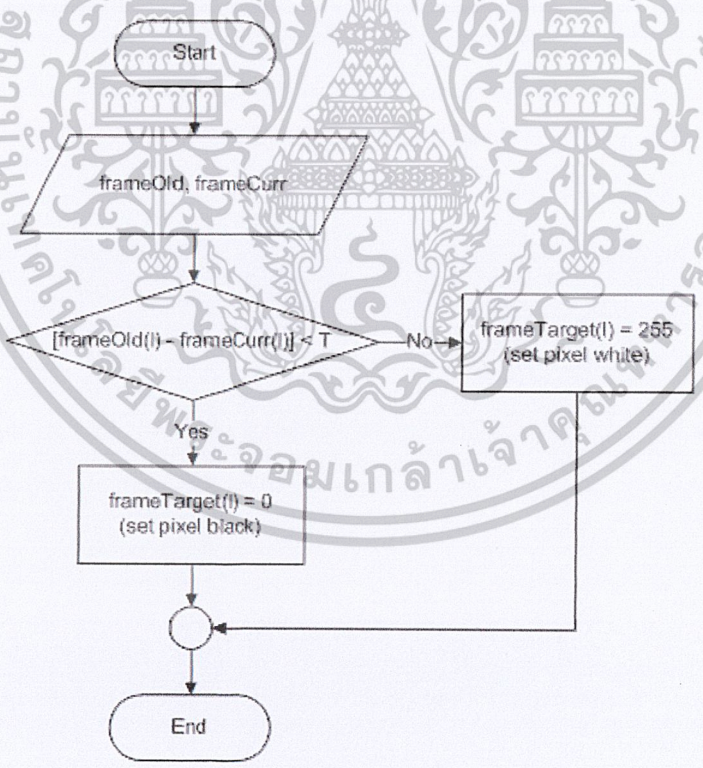
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.3 การแยกส่วนที่สนใจออกจากสิ่งแวดล้อม (Image Subtraction)

การแยกส่วนที่สนใจออกจากสิ่งแวดล้อมนั้นเป็นขั้นตอนแรกในการทำงานหลายๆอย่างที่เกี่ยวกับการประมวลผลภาพ ในการแยกวัตถุที่สนใจ (Target) ออกจากพื้นหลังหรือสิ่งแวดล้อม (Background) เพื่อนำไปใช้ในการคำนวณและการวิเคราะห์ในขั้นตอนต่อๆไป

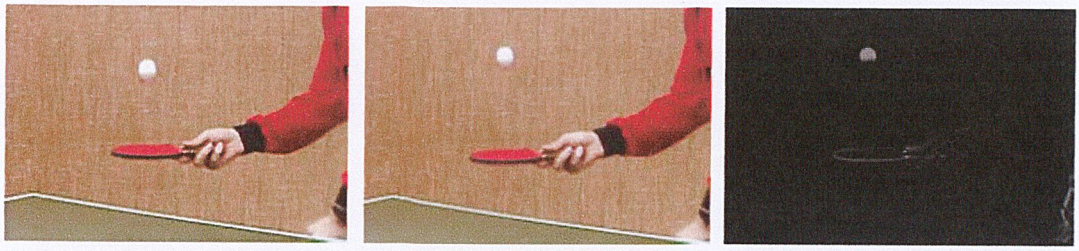
2.1.3.1 การหาความต่างระหว่างเฟรม (Frame Differencing)

Frame Differencing หรือการหาความต่างระหว่างเฟรมเป็นวิธีพื้นฐานที่ง่ายที่สุดในระเบียบวิธีการทำการลบฉากหลัง ที่จะทำการหักลบเฟรมหนึ่งกับอีกเฟรมหนึ่งซึ่งทุกส่วนที่แตกต่างกันจะเป็นวัตถุที่สนใจ กระบวนการนี้จะทำหน้าที่คอยตรวจจับวัตถุที่มีการเคลื่อนไหวเพื่อให้มองภาพได้ชัดเจนสมมติว่ามีภาพ 1 channel จำนวน 3 ภาพคือ frameOld, frameCurr และ frameTarget ซึ่ง frameOld จะเป็นภาพระดับสีเทาของเฟรมก่อนหน้า ขณะที่ frameCurr คือภาพระดับสีเทาของภาพปัจจุบัน และท้ายสุด frameTarget คือภาพผลลัพธ์ที่ได้จากการหาความต่างระหว่างเฟรม โดยมีการหาค่าสัมบูรณ์ของความแตกต่าง แล้วนำมาเปรียบเทียบกับ T คือค่า Threshold ที่กำหนดไว้ ซึ่งสามารถทำความเข้าใจจากแผนผังต่อไปนี้



รูปที่ 2.11 อัลกอริธึมการหาความแตกต่างระหว่างเฟรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

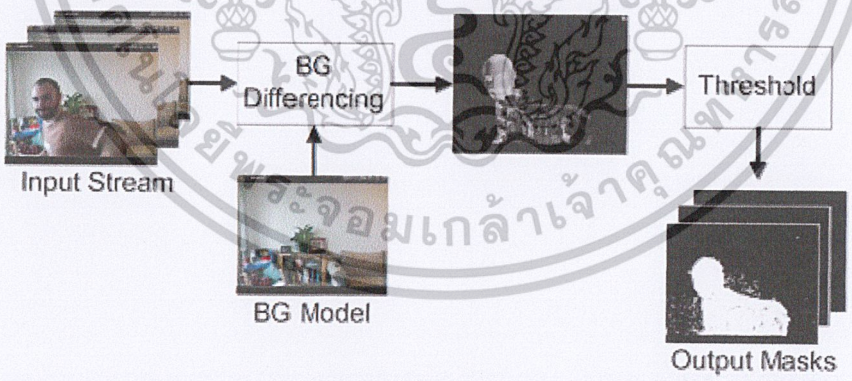


รูปที่ 2.12 การนำ Frame Differencing มาใช้งาน

2.1.3.2 การลบฉากหลัง (Background Subtraction)

การลบฉากหลังเป็นเทคนิคที่ง่าย และมีประสิทธิภาพในการแยกวัตถุที่สนใจออกจากฉากหลังของภาพ ซึ่งหากกระบวนการลบฉากหลังทำได้ไม่ดีแล้ว จะส่งผลต่อการวิเคราะห์และการตรวจจับการเคลื่อนไหวของผู้เล่นผิดตามไปด้วย

หลักการในการลบฉากหลังนั้นจะเริ่มจากการเก็บภาพฉากหลังหลายๆภาพจากกล้องวีดีโอเพื่อนำมาคำนวณสร้างแบบจำลองฉากหลัง (Background Model) หรือที่เรียกว่าเป็นการเรียนรู้ฉากหลัง (Learning Background) ซึ่งการเก็บภาพฉากหลังด้วยวิธีนี้จะมีความเหมาะสมในกรณีที่กล้องตั้งอยู่กับที่ และฉากหลังไม่มีการเคลื่อนไหว หรือเคลื่อนไหวน้อย หลังจากสร้างแบบจำลองฉากหลังแล้ว จะทำการนำแต่ละเฟรมของวีดีโอลบออกจากรูปแบบจำลองฉากหลัง เพื่อหาวัตถุเป้าหมายภายในภาพ ผลลัพธ์ที่ได้จะเป็นในลักษณะของโครงร่างของเป้าหมาย (Target's Silhouette) โดยโครงร่างนี้เองที่จะนำไปใช้ในการวิเคราะห์ คำนวณ และตรวจจับการเคลื่อนไหวต่อไป [1], [2]



รูปที่ 2.13 ขั้นตอนการทำงานของการลบฉากหลัง

จากรูปที่ 2.13 อธิบายได้ดังนี้ BG Model คือ การหาค่าเฉลี่ยแต่ละพิกเซลของรูปภาพจากการเรียนรู้ฉากหลังเพื่อนำมาสร้างเป็นแบบจำลองฉากหลัง หลังจากนั้นเมื่อมีวีดีโอเข้าสู่ระบบ (Input Stream) กระบวนการลบฉากหลัง (BG Differencing) จะทำการนำแบบจำลองฉากหลังที่สร้างไว้มาทำการหักลบกับอินพุทที่เข้ามา ผลที่ได้ก็จะเป็นรูปที่ฉากหลังถูกลบออกไปแล้ว แต่ยังไม่ดีเท่าที่ควร จึงเข้าสู่การแปลงภาพสีเป็นภาพขาวดำ เพื่อให้พร้อมนำไปใช้งานต่อ เมื่อได้ผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สุดท้าย (Output Masks) แล้ว จะสังเกตว่ารูปที่ได้ยังมีสัญญาณรบกวน (Noise) อยู่ ดังนั้นจึงจำเป็นต้องมีการปรับแต่งภาพอีกครั้ง ซึ่งกระบวนการต่างๆที่ได้กล่าวไว้ในที่นี้จะมีการอธิบายให้ชัดเจนขึ้นในหัวข้อถัดไป

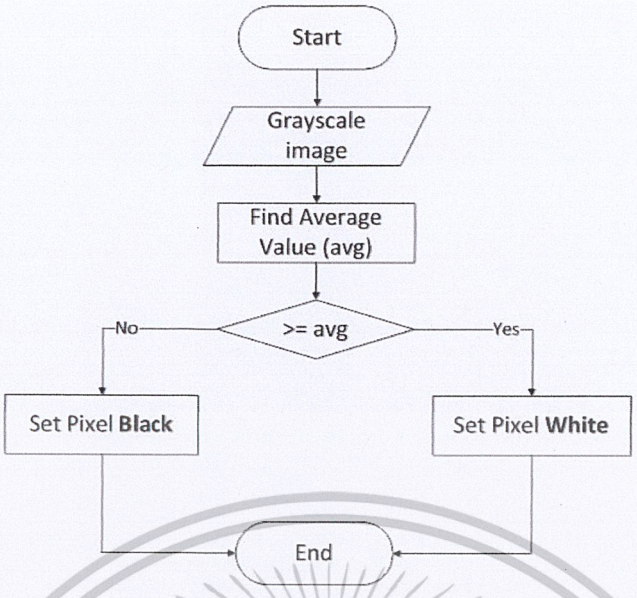
การแยกส่วนที่สนใจออกจากสิ่งแวดล้อมด้วยเทคนิคการลบฉากหลังนี้ มีข้อดีคือ เป็นวิธีการที่ง่าย ประมวลผลได้รวดเร็ว มีความแม่นยำถูกต้องสูง ง่ายต่อการคำนวณและออกแบบ รูปที่ได้จะมีความชัดเจน แต่มีข้อเสียคือ ปัญหาเรื่องของแสงจากสภาพแวดล้อม และเรื่องของการเคลื่อนไหวของสภาพแวดล้อมที่เป็นฉากหลัง ซึ่งข้อจำกัดในเรื่องนี้อาจทำให้การประมวลผลภาพผิดเพี้ยนไปได้ เพราะฉะนั้นจึงต้องมีการควบคุมสภาพแวดล้อมโดยรอบไม่ให้มีแสงมากหรือน้อยเกินไป หรือการเคลื่อนไหวของฉากหลังก็ต้องให้เกิดขึ้นน้อยที่สุด

2.1.4 การแปลงภาพสีให้เป็นภาพขาวดำ (Thresholding)

การแปลงภาพสีให้เป็นภาพขาวดำ หรือภาพไบนารี่นั้น หมายถึง การแปลงข้อมูลรูปที่มีความเข้มหลายระดับ (Multilevel Image) ให้เป็นรูปที่มีระดับความเข้มสองระดับ คือ 1 และ 0 ภายใน 1 พิกเซล โดยพิกเซลที่แทนด้วยค่า 1 คือสีขาว และพิกเซลที่แทนด้วยค่า 0 คือสีดำ ซึ่งการแปลงภาพไบนารี่นี้มีประโยชน์ในการแสดงผลภาพบนอุปกรณ์ที่มีความสามารถในการแสดงผลได้สองระดับ และเพื่อลดเนื้อที่การเก็บข้อมูลภาพ และที่สำคัญมีการนำไปประยุกต์ใช้ในการประมวลผลภาพขั้นต้น (Preprocessing) ด้วย

การแปลงภาพสีเป็นภาพขาวดำ จะพิจารณาจากพิกเซลในภาพว่าควรจะเป็นจุดขาวหรือจุดดำ โดยนำไปเปรียบเทียบกับค่าคงที่ค่าหนึ่งที่เรียกว่า ค่าเทรชโฮล (Threshold Value) ถ้าค่าของพิกเซลใดที่มีค่าน้อยกว่าค่าเทรชโฮลจะกำหนดให้เป็น 0 (จุดดำ) และถ้าค่าของพิกเซลใดมีค่ามากกว่า หรือเท่ากับค่าเทรชโฮลจะกำหนดให้เป็น 1 (จุดขาว)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.14 การทำงานของ Thresholding

สิ่งสำคัญในการแปลงภาพสีเป็นภาพขาวดำนั้นคือ การกำหนดค่าเทรชโฮล เนื่องจากค่าเทรชโฮลที่ไม่เหมาะสมจะส่งผลต่อรูปที่ได้ หมายความว่า ถ้าเลือกค่าเทรชโฮลมากไป รูปที่ได้จะมืดเกินไป รายละเอียดบางส่วนขาดหายไปหรือถ้าเลือกค่าเทรชโฮลน้อยไป รูปที่ได้จะสว่างเกินไป ขาดความคมชัด หรืออาจเกิดสัญญาณรบกวนได้

(ก) (ข)

รูปที่ 2.15 ผลลัพธ์จากการกำหนดค่าเทรชโฮลที่ไม่เหมาะสม

(ก) การกำหนดค่าเทรชโฮลมากเกินไป

(ข) การกำหนดค่าเทรชโฮลน้อยเกินไป

ดังนั้นจึงต้องมีวิธีในการหาค่าเทรชโฮลที่เหมาะสม ซึ่งแต่ละวิธีก็มีความเหมาะสมกับลักษณะการทำงานที่แตกต่างกันไป มีวิธีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.1 การหาค่าเทรชโฮลโดยการกำหนดค่าล่วงหน้า

วิธีนี้เป็นวิธีที่ง่ายที่สุด เพราะเป็นการกำหนดค่าเทรชโฮลจากผู้ใช้ ซึ่งการกำหนดค่านั้นขึ้นอยู่กับประสบการณ์ของผู้ใช้นั้นๆ โดยการเลือกค่าเทรชโฮลมาค่าหนึ่งที่จะต้องมีค่าอยู่ระหว่างค่าต่ำสุดและค่าสูงสุดของระดับความเข้มภาพอินพุท หมายความว่า ถ้าภาพอินพุทเป็นภาพระดับสีเทา ค่าเทรชโฮลที่เลือกได้จะมีค่าตั้งแต่ 0 ถึง 255 เพราะภาพระดับสีเทามีระดับความเข้ม 256 ระดับ

2.1.4.2 การหาค่าเทรชโฮลจากค่ากลาง

เป็นวิธีการคำนวณค่าเทรชโฮลที่อาศัยการคำนวณพื้นฐานทางสถิติในเรื่องของการหาค่ากลาง หรือค่าเฉลี่ย (Mean) มาประยุกต์ใช้ ค่าเทรชโฮลที่ได้จากการคำนวณด้วยวิธีนี้ จะเป็นค่าที่เหมาะสมสำหรับภาพนั้นๆ

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \qquad (2.7)$$

โดยที่ x_i คือ ค่าของแต่ละพิกเซล

ซึ่งวิธีการหาค่าเทรชโฮลในโครงการนี้ได้เลือกใช้วิธีการหาโดยอาศัยค่ากลาง เนื่องจากว่าเป็นวิธีที่ง่าย และเหมาะสมกับการใช้งานที่มีพื้นหลังนิ่ง แม้ว่าการหาค่าเทรชโฮลแบบกำหนดล่วงหน้าจะง่ายและสะดวกกว่า แต่ในเรื่องของการกำหนดค่าตายตัวเลยนั้นอาจไม่ใช่วิธีที่ดีนัก เพราะปัจจัยภายนอกจากสภาพแวดล้อม เช่น แสง อาจส่งผลให้การแปลงภาพผิดเพี้ยนไป และไม่ดีเท่าที่ควร ดังนั้นการใช้แบบหาค่าเฉลี่ยจึงเป็นวิธีที่เหมาะสมกว่า เพราะสามารถปรับเปลี่ยนค่าเทรชโฮลให้เหมาะสมกับสภาวะแวดล้อม ณ ตอนนั้นได้

2.1.5 การปรับแต่งภาพด้วยมอร์โฟโลยี (Morphology)

มอร์โฟโลยีเป็นการกำจัดสัญญาณรบกวนที่เกิดขึ้นออกไป ก่อนที่จะนำภาพไปประมวลผลต่อ โดยการพิจารณาจุดขาวบนพื้นหลังดำ หรือภาพไบนารี่ การทำงานพื้นฐานของมอร์โฟโลยีประกอบด้วย การขยาย (Dilation) และการย่อ (Erosion) ซึ่งเป็นกระบวนการพื้นฐานในการนำไปใช้กับกระบวนการเปิด (Opening) และกระบวนการปิด (Closing)

2.1.5.1 การขยาย (Dilation)

การขยายในที่นี้จะพิจารณาสำหรับข้อมูลรูปที่เป็นไบนารี่ ซึ่งเป็นการเพิ่มจุดสีขาวและลดจุดสีดำโดยใช้หน่วยโครงสร้าง (Element Structure) ทาบในตำแหน่งที่มีค่าเป็น 1 ถ้าค่าของพิกเซลรอบๆมีค่าเป็น 0 ให้ปรับค่าเป็น 1 และถ้าค่าของพิกเซลรอบๆมีค่าเป็น 1 ให้คงค่าเดิมไว้ สามารถนิยามในรูปของเซตได้ดังนี้

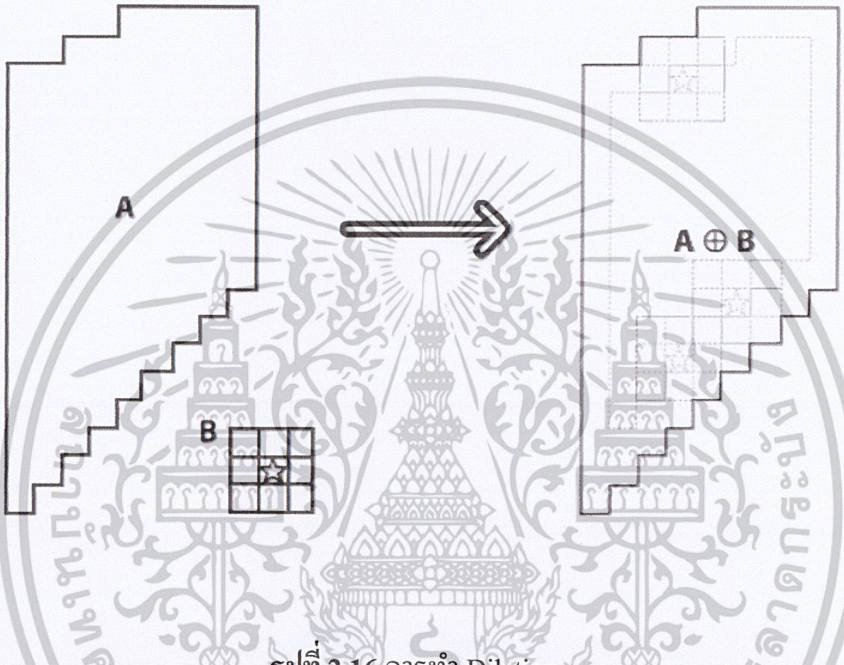
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$A \oplus B = \{x \mid [\hat{B}_x \cap A] \subseteq A\} \quad (2.8)$$

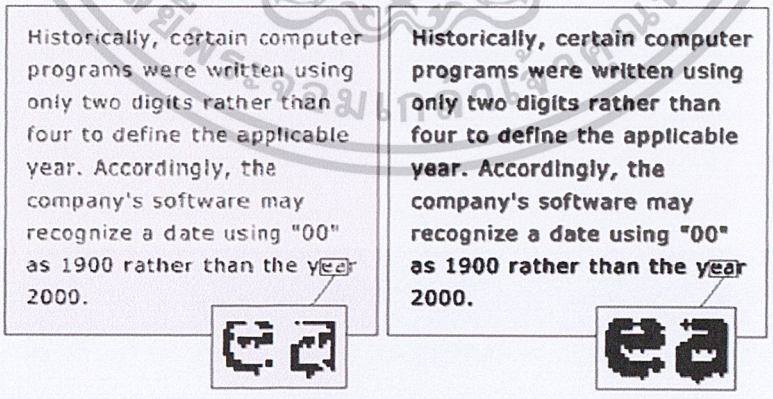
โดยที่ A คือ รูปภาพที่ทำการขยาย

B คือ หน่วยโครงสร้างในการทำการขยาย

นั่นคือ การขยายของ A ด้วย B คือเซตของระยะ x ที่เคลื่อนที่ทั้งหมด โดยที่ $\hat{B}$ และ A มีการซ้อนทับกัน (Overlap) อย่างน้อยหนึ่งสมาชิกที่ไม่เป็นศูนย์



รูปที่ 2.16 การทำ Dilation



รูปที่ 2.17 การนำ Dilation มาประยุกต์ใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

2.1.5.2 การย่อ (Erosion)

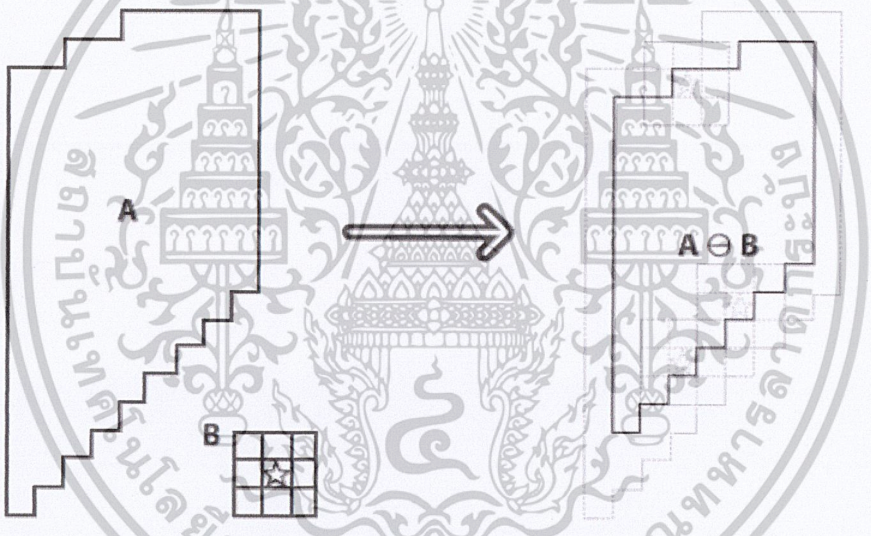
การย่อเป็นการลดจุดสีขาวและเพิ่มจุดสีดำ โดยใช้หน่วยโครงสร้างทาบในตำแหน่งที่มีค่าเป็น 1 ถ้าค่าของพิกเซลรอบๆมีค่าเป็น 0 ให้ปรับค่าเป็น 0 สามารถนิยามในรูปของเซตได้ดังนี้

$$A \ominus B = \{x \mid \hat{B}_x \subseteq A\} \tag{2.9}$$

โดยที่ A คือ รูปภาพที่ทำการย่อ

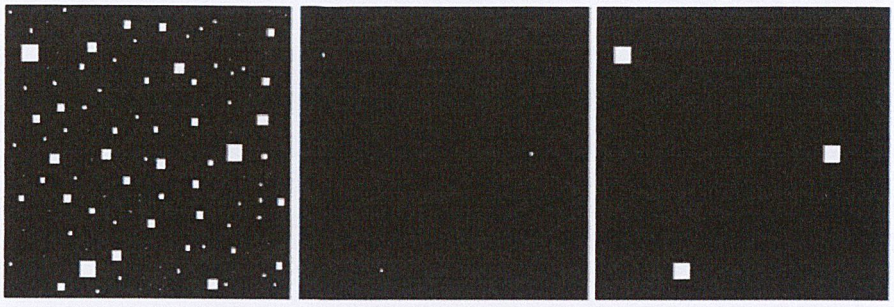
B คือ หน่วยโครงสร้างในการทำการขยาย

นั่นคือ การย่อของ A ด้วย B คือเซตของจุด x ทุกจุดที่ซึ่งเคลื่อนย้าย B ด้วยระยะ x อยู่ภายใน A



รูปที่ 2.18 การทำ Erosion

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
06605



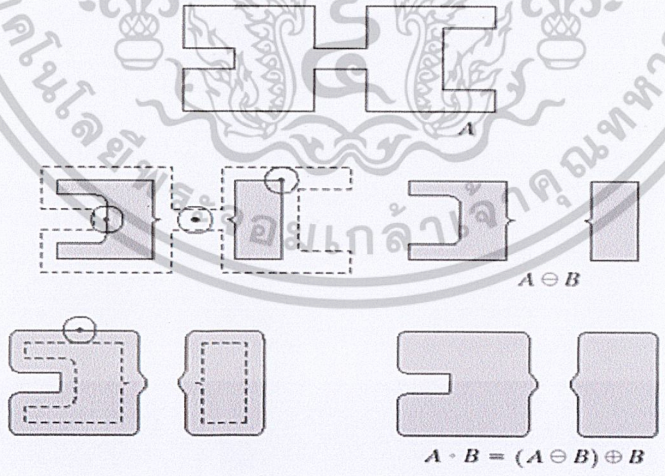
รูปที่ 2.19 การนำ Erosion มาประยุกต์ใช้งาน

2.1.5.3 การเปิด (Opening)

เป็นการทำงานร่วมกันระหว่างการย่อและการขยายตามลำดับ เพื่อทำให้เส้นโครงร่างของรูปมีความเรียบขึ้น หรือทำให้ช่องแคบที่ต่อภาพขาดจากกัน และกำจัดส่วนเกินบางๆออกไป หรือกล่าวได้ว่า กระบวนการเปิดคือการทำให้สัญญาณรบกวนขนาดเล็กในรูปภาพหายไปนั่นเอง สามารถนิยามในรูปของเซตได้ดังนี้

$$A \circ B = (A \ominus B) \oplus B \tag{2.10}$$

นั่นคือ การเปิดของเซต A โดย B ทำได้โดยการย่อ A ด้วย B แล้วตามด้วยการขยายผลลัพธ์ด้วย B



รูปที่ 2.20 การทำ Opening

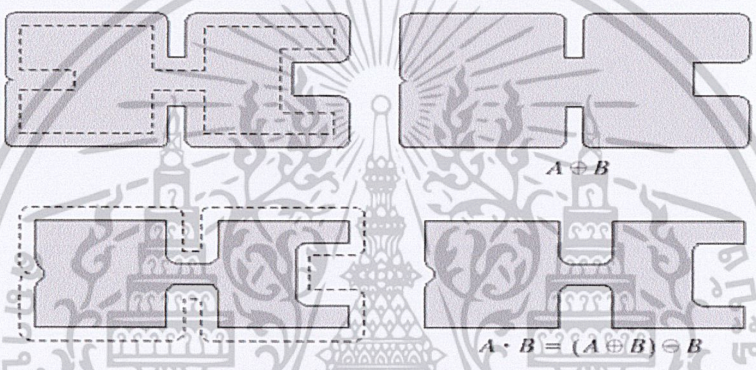
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.5.4 การปิด (Closing)

เป็นกระบวนการที่ทำงานตรงข้ามกับการเปิด คือการปิดจะเริ่มจากการขยายและการย่อตามลำดับ เพื่อทำให้ช่องว่าง หรือช่องว่างเล็กๆของบริเวณที่เชื่อมต่อกันหมดไป สามารถนิยามในรูปของเซตได้ดังนี้

$$A \bullet B = (A \oplus B) \ominus B \qquad (2.11)$$

นั่นคือ การปิดของเซต A โดย B ทำได้โดยการขยาย A ด้วย B แล้วตามด้วยการย่อผลลัพธ์ด้วย B



รูปที่ 2.21 การทำ Closing

2.2 XNA Game Studio Express (XNA GSE)

XNA GSE เป็นเครื่องมือในการพัฒนาเกมโดยเฉพาะ ถูกพัฒนาขึ้นโดยบริษัทไมโครซอฟต์ ซึ่ง XNA สามารถพัฒนาเกมที่รันบน Windows หรือ บนเครื่อง Xbox 360 ได้ โดยทางไมโครซอฟต์เปิดให้ผู้ที่สนใจสามารถดาวน์โหลดใช้ได้ฟรี เวอร์ชั่นล่าสุดคือ XNA Game Studio Express 4 Beta ที่มาพร้อมกับการทำงานบน Windows Phone 7 แต่เวอร์ชั่นที่ใช้ในโครงการนี้คือ XNA Game Studio Express 3.1 องค์ประกอบหลักของ XNA จะประกอบด้วย 2 ส่วน คือ XNA Game Studio Express เป็นตัว IDE (Integrated Development Environment) และเครื่องมือต่างๆที่ใช้ในการพัฒนาเกม อีกส่วนหนึ่งคือ XNA Framework เป็นไลบรารีในกลุ่มของ .NET ซึ่งต้องใช้ภาษา C# ในการพัฒนาเท่านั้น ซึ่งก่อนที่จะลง XNA GSE นั้นจะต้องเตรียมเครื่องคอมพิวเตอร์ดังนี้

ระบบปฏิบัติการ

- Microsoft Windows XP Service Pack 2
- Microsoft Windows Vista

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Microsoft Windows 7

ฮาร์ดแวร์

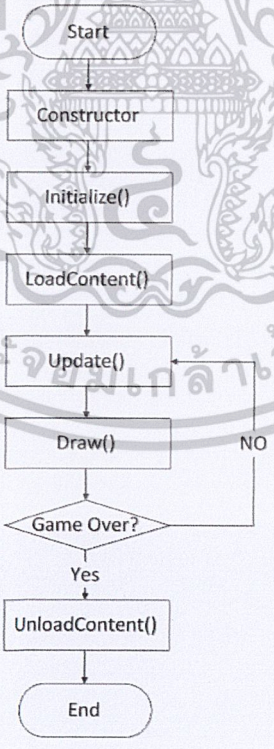
- การ์ดจอที่รองรับ DirectX 9.0 และ Shader Model 1.1 และ 2.0
- เครื่อง Xbox 360 ในกรณีที่ต้องการพัฒนาบนเครื่อง Xbox 360

ซอฟต์แวร์

- Microsoft .NET Framework 2.0
- Microsoft Visual C# 2005 Express Edition Service Pack 1
- Microsoft DirectX Runtime 9.0c

2.2.1 โครงสร้างการทำงานของ XNA Framework และฟังก์ชันที่สำคัญ

โครงสร้างของโปรแกรม [3] เมื่อเริ่มสร้างจะมีไฟล์สองไฟล์ที่มีนามสกุล .cs ซึ่งเป็นนามสกุลของภาษา C# ทั่วไป คือ ไฟล์ Program.cs ที่เป็นตัวสร้างออปเจ็คต์ของคลาส Game ที่อยู่ในไฟล์ Game.cs ซึ่งการเขียนเกมโดยใช้ XNA นี้ได้มีการสืบทอดเมธอดต่างๆจากคลาส Game โดยคลาสนี้ประกอบไปด้วยเมธอดต่างๆที่จำเป็นต้องนำมาใช้ เพื่อให้สามารถทำงานได้ตามต้องการ ซึ่งมีโครงสร้างลำดับการทำงาน และหน้าที่การทำงานดังนี้



รูปที่ 2.22 ลำดับการทำงานของ XNA Framework

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) **Constructor** เมื่อเริ่มต้นทำงานจะมีการเรียก Constructor ก่อน ภายในนี้จะทำการสร้างออปเจ็คต์อื่นๆ หรือกำหนดค่าเริ่มต้นของเกม เช่น โหมดการแสดงผล ขนาดหน้าจอ จำนวนบิทสีของหน้าจอที่ใช้ได้

```
public Game1()
{
        graphics = new GraphicsDeviceManager(this);
        graphics.PreferredBackBufferWidth = 1024;
        graphics.PreferredBackBufferHeight = 768;
        graphics.ApplyChanges();
        Content.RootDirectory = "Content";
}
```

จากโค้ด มีการกำหนดขนาดการแสดงผลเป็น 1024 × 768 พิกเซล

2) **Initialize()** หลังจาก Constructor ถูกเรียกทำงาน Initialize() จะถูกเรียกทำงานต่อ โดยในตอนนี้ XNA ได้ทำการเริ่มต้นเข้าสู่โหมดกราฟฟิกทำให้สามารถกำหนดค่าเริ่มต้นของการแสดงผล เช่น มุมกล้อง ตำแหน่งของการมอง เป็นต้น

3) **LoadContent()** ส่วนนี้จะถูกเรียกใช้งานทันทีเพื่อให้ทำการเขียนคำสั่งต่างๆในการโหลดกราฟฟิกต่างๆ เช่น ภาพ เสียงประกอบ หรือเนื้อหาอื่นๆเพื่อนำมาใช้ในเกม

```
protected override void LoadContent()
{
        spriteBatch = new SpriteBatch(GraphicsDevice);
        abox = Content.Load<Model>("Models\\box1");
        Font1 = Content.Load<SpriteFont>("Tahoma");
}
```

จากโค้ดมีการโหลดวัตถุสามมิติมาไว้ที่ตัวแปร abox และรูปแบบอักขระมาไว้ที่ Font1 เพื่อนำไปใช้ต่อไป

4) **Update() และ Draw()** สองส่วนนี้จะมีการทำงานแบบวนรอบไปเรื่อยๆจนกว่าจะออกจากเกมด้วยคำสั่ง Exit() นอกจากนั้นใน Update() สามารถที่จะอ่านค่าจากคีย์บอร์ดเมาส์ หรือ Joy Pad ของ Xbox ได้ หลังจากนั้นทำการปรับปรุงตำแหน่งของวัตถุหรือคำนวณสถานะต่างๆของตัวละคร และใน Draw() สามารถทำการวาดตัวละคร หรือวัตถุต่างๆได้ตามต้องการ

5) **UnloadContent()** จะถูกเรียกหลังจากออกจากการวนรอบของเกมด้วยคำสั่ง Exit() แล้ว โดยใน UnloadContent() สามารถเขียนคำสั่งเพื่อลบข้อมูลในหน่วยความจำที่ใช้ไปในเกม หรือปิดไฟล์ต่างๆที่เปิดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.2 ข้อดีและข้อเสียของ XNA

ข้อดี

- เป็น Free Engine
- สามารถสร้างเกมบนเครื่อง Xbox 360 ได้
- มีไลบรารีที่สามารถโหลดโมเดลสามมิติมาใช้งานได้โดยตรง
- รองรับการทำงานกับเครือข่าย (Network)
- มีการตรวจสอบ Collision แบบพิกเซล

ข้อเสีย

- มีหนังสือในประเทศไทยให้ศึกษาน้อย ส่วนมากเป็นหนังสือจากต่างประเทศ
- Bounding Box ตรวจสอบได้แต่รูปสี่เหลี่ยม

2.3 การใช้ซอฟต์แวร์ไลบรารี EmguCV ในการประมวลผลภาพ

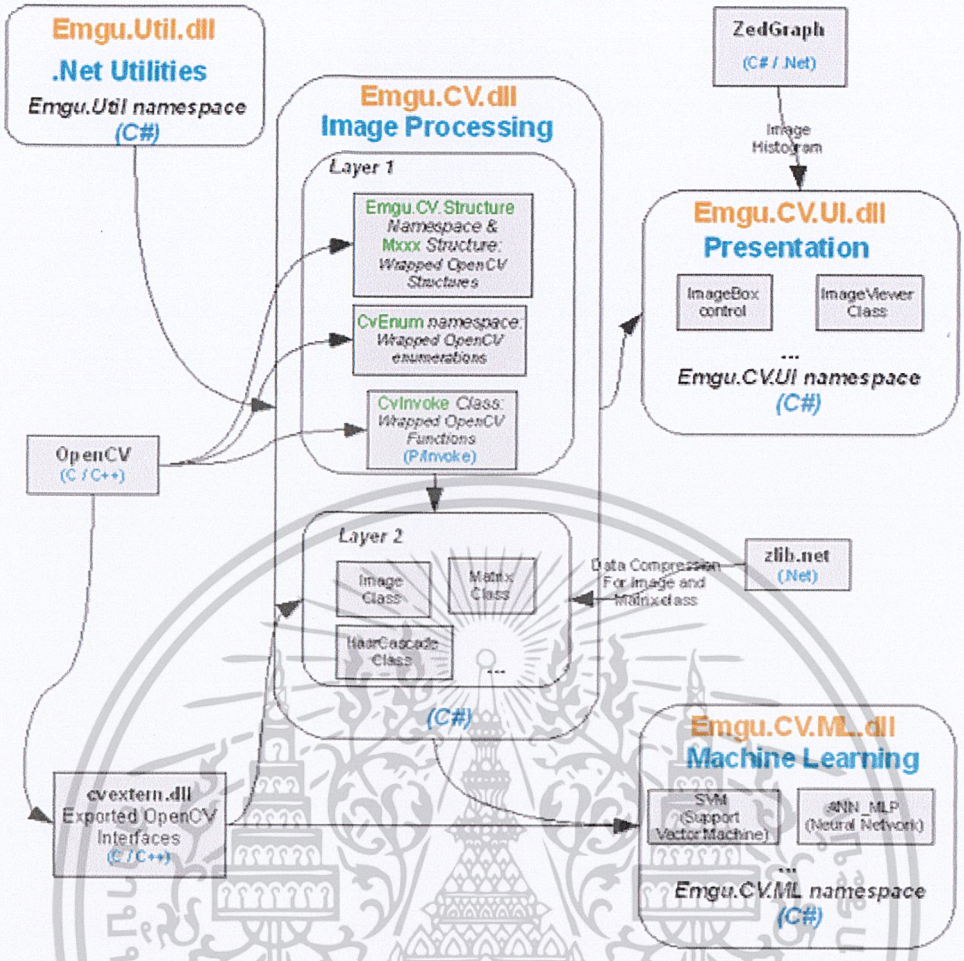
EmguCV [9] คือ .Net Wrapper ที่ช่วยให้สามารถเรียกใช้คำสั่งของ OpenCV (Open Source Computer Vision Library) ได้ ซึ่ง OpenCV เป็นไลบรารีที่จำเป็นสำหรับใช้ในการประมวลผลภาพ โดยสามารถเรียกใช้คำสั่งของ OpenCV ที่มีคำสั่งการทำงานสำหรับการจัดการข้อมูลรูปภาพ และการประมวลผลภาพพื้นฐานอยู่มากมาย เช่น การหาความต่างของสี การทำภาพไบนารี่การหาขอบภาพ ได้จากภาษาที่มีการทำงานร่วมกับ .NET เช่น C#, Visual Basic, Visual Basic C++, IronPython เป็นต้น นอกจากนี้ EmguCV ยังสามารถคอมไพล์และทำงานบนระบบปฏิบัติการ Linux และ Mac OSX ได้ EmguCV เป็นไลบรารีโอเพนซอร์ส (Open Source) ซึ่งสามารถดาวน์โหลดมาใช้งานได้ฟรี

2.3.1 สถาปัตยกรรมของ EmguCV

ลักษณะโครงสร้างของ EmguCV Wrapper สามารถแบ่งได้ออกเป็น 2 Layer คือ

1) Layer พื้นฐาน (Layer 1) ประกอบด้วย ฟังก์ชันการทำงาน โครงสร้าง และ Enumeration ที่เชื่อมโยงโดยตรงกับ OpenCV
2) Layer ขั้นที่ 2 (Layer 2) ประกอบด้วย คลาสที่มีประโยชน์จาก .NET

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.23 สถาปัตยกรรมของไลบรารี EmguCV

2.3.2 การ Wrapping OpenCV ของ EmguCV

2.3.2.1 Function Mapping - Emgu.CV.CvInvoke

CvInvoke Class เป็นคลาสที่ใช้เรียกฟังก์ชันการทำงานของ OpenCV ภายในภาษา .NET บางเมธอดในคลาสมีชื่อตรงกับฟังก์ชันใน OpenCV ดังตัวอย่าง

```
IntPtr image = CvInvoke.cvCreateImage(new System.Drawing.Size(400, 300),
CvEnum.IPL_DEPTH.IPL_DEPTH_8U, 1);
```

มีการทำงานที่เหมือนกันกับฟังก์ชันดังต่อไปนี้ในภาษา C

```
IplImage* image = cvCreateImage(cvSize(400, 300), IPL_DEPTH_8U, 1);
```

ทั้ง 2 คำสั่งเป็นคำสั่งสร้างภาพขนาด 400× 300 พิกเซล ขนาด 8 บิท ภาพระดับสีเทา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.2.2 Structure Mapping - Emgu.CV.Structure.Mxxx

โครงสร้างข้อมูลของ EmguCV จะมีการเชื่อมโยงกับโครงสร้างข้อมูลของ OpenCV โดยตรง แตกต่างกันตรงที่โครงสร้างข้อมูลของ EmguCV จะมี M (Managed Structure) อยู่ข้างหน้า

ตารางที่ 2.1 แสดงการเชื่อมโยงโครงสร้างข้อมูลระหว่าง EmguCV และ OpenCV

EmguCV Structure	OpenCV structure
Emgu.CV.Structure.MIplImage	IplImage
Emgu.CV.Structure.MCvMat	CvMat
...	...
Emgu.CV.Structure.M*xxxx*	*xxxx*

และมีโครงสร้างข้อมูลบางชนิดที่ EmguCV ได้ยืมโครงสร้างข้อมูลที่มีอยู่ของ .NET มาใช้แทนโครงสร้างข้อมูลของ OpenCV เลย

ตารางที่ 2.2 แสดงการเชื่อมโยงโครงสร้างข้อมูลระหว่าง .NET และ OpenCV

.NET Structure	OpenCV structure
System.Drawing.Point	CvPoint
System.Drawing.PointF	CvPoint2D32f
System.Drawing.Size	CvSize
System.Drawing.Rectangle	CvRect

2.3.2.3 Enumeration Mapping - Emgu.CV.CvEnum

CvEnum Namespace จะเชื่อมโยงโดยตรงกับ OpenCV Enumerations เช่น CvEnum.IPL_DEPTH.IPL_DEPTH_8U มีค่าเท่ากับ IPL_DEPTH_8U ใน OpenCV เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.3.3 ข้อดีและข้อเสียของ EmguCV

ข้อดี

- สามารถทำงานข้ามระบบปฏิบัติการได้
- ภาษาในการเขียนสามารถใช้ได้หลากหลาย
- มีการทำ Garbage Collection ให้อัตโนมัติ
- มีรูปแบบ (Syntax) ที่เรียบง่าย ทำให้เข้าใจความหมายได้ง่าย

ข้อเสีย

- ข้อมูลในการศึกษามีน้อย

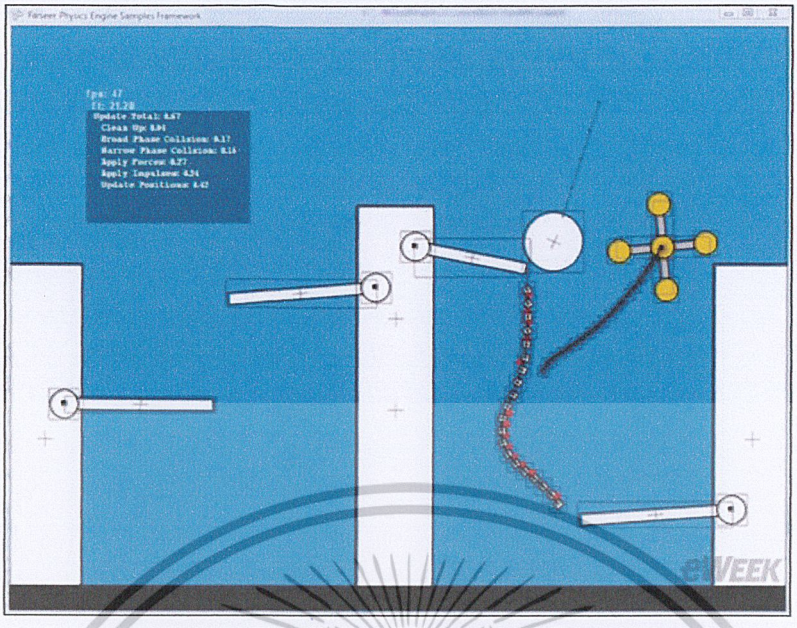
2.4 ระบบฟิสิกส์ (Physics Engine)

ระบบฟิสิกส์หรือ Physics Engine [12] นั้นเป็นซอฟต์แวร์คอมพิวเตอร์ที่จำลองระบบทางฟิสิกส์อย่างง่าย เช่น การเคลื่อนที่ของวัตถุแข็งแกร่ง (Rigid Body) การเคลื่อนที่ของวัตถุอ่อนตัว (Soft Body) และพลศาสตร์ของไหล (Fluid Dynamics) ที่ใช้ในส่วนของคอมพิวเตอร์กราฟฟิก วีดีโอเกม และภาพยนต์ ซึ่งโดยทั่วไปมักจะใช้กับวีดีโอเกมและทำงานแบบ Real time โดยทั่วไปนั้น Physics Engine มี 2 ประเภทด้วยกัน คือ Real-timePhysics Engine และ High-precision Physics Engine

High-precision Physics Engine เป็นการทำงานที่ต้องใช้การประมวลผลขั้นสูงที่มีความแม่นยำมากในการคำนวณทางฟิสิกส์ ซึ่งมักจะใช้โดยนักวิทยาศาสตร์ และใช้ในภาพยนต์แอนิเมชั่น ส่วน Real-time Physics Engine นั้นจะใช้ในวีดีโอเกมและงานอื่นๆที่มีการทำงานเชิงปฏิสัมพันธ์ ซึ่งจะใช้งานง่ายกว่าแบบแรก ในการประมวลผลนั้นจะมีการลดอัตราความถูกต้องในการคำนวณทางฟิสิกส์ลง เพื่อให้เหมาะสมสำหรับการเล่นเกมที่จำเป็นต้องแสดงผลลัพธ์แบบ Real time นั่นเอง

Real-time Physics Engine ที่เหมาะสมกับการพัฒนาเกมโดย Microsoft XNA นั้นก็คือ Farseer Physics Engine ซึ่งเป็น Physics Engine สำหรับการทำงานกับภาพสองมิติที่ใช้งานง่าย อีกทั้งยังสามารถรองรับได้หลายแพลทฟอร์ม เช่น Microsoft XNA, Silverlight, WPF และ Vanilla .NET เป็นต้น Farseer Physics Engine นั้นจะเน้นที่ความเรียบง่ายในการคำนวนค่าฟิสิกส์เกี่ยวกับเรื่อง แรงโน้มถ่วง (Gravity), แรง (Force), ทอร์ก (Torque), การชนกัน (Collisions), สปริง (Spring) และข้อต่อ (Joints) รวมทั้งคุณสมบัติอื่นๆที่มีประโยชน์ที่จะช่วยให้การพัฒนาเกมมีความสมจริงและสนุกสนานมากยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.24 การใช้งาน Farseer Physics Engine

2.5 การสร้างเอฟเฟกต์ด้วย Mercury Particle Engine

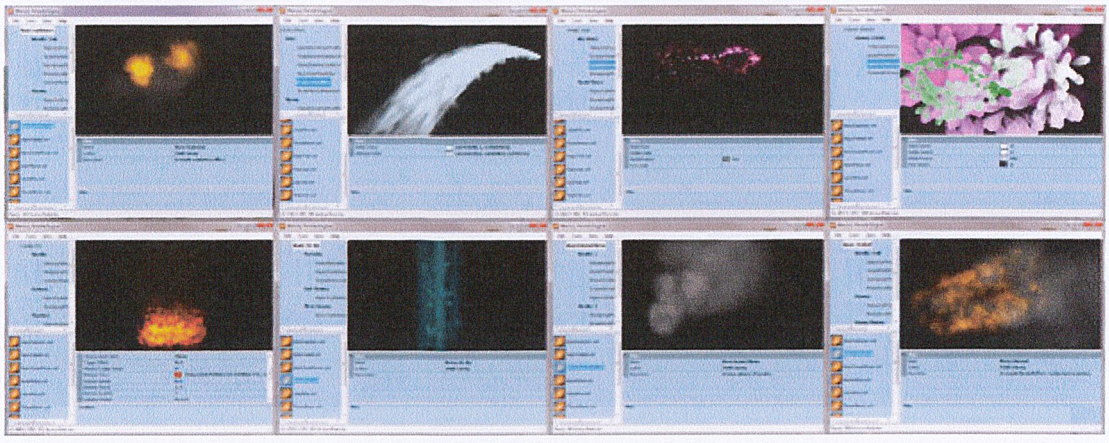
Mercury Particle Engine [11] เป็นไลบรารีตัวหนึ่งที่ช่วยให้นักพัฒนาเกมสามารถเพิ่มความสมบูรณ์ให้กับเกมด้วยการเพิ่มเอฟเฟกต์ต่างๆที่จะทำให้เกมดูมีความสมจริงมากยิ่งขึ้น Mercury Particle Engine ถูกพัฒนาขึ้นสำหรับใช้กับ Microsoft XNA Framework และสามารถทำงานร่วมกับ Windows และ Xbox 360 ได้เป็นอย่างดี

Mercury Particle เป็นองค์ประกอบของภาพกราฟฟิกชิ้นเล็กที่แสดงเกี่ยวกับเอฟเฟกต์ต่างๆในเกม เอฟเฟกต์ที่ Mercury Particle สามารถทำได้ มีดังนี้

1) Explosions เอฟเฟกต์จำพวกควัน คลื่นน้ำ เมฆ รวมถึงการระเบิดของวัตถุต่างๆ
2) Weather Effects เอฟเฟกต์ที่ใช้ในการแสดงเกี่ยวกับผลสภาพทางอากาศ เช่น ฝน หิมะ หรือ พายุหมุน เป็นต้น
3) Sparks & Flames เอฟเฟกต์ประกายไฟที่แสดงถึงการลุกไหม้ของวัตถุต่างๆ
4) Muzzle Flash เอฟเฟกต์เกี่ยวกับการยิงปืน
5) Water Effect เอฟเฟกต์ไอน้ำ หมอก หรือน้ำตก
6) Magic Spells เอฟเฟกต์จำพวกเวทมนต์ต่างๆ เช่น ลูกไฟหมุน พลังงาน หรือแสงระยิบระยับ (Glitter) เป็นต้น

นอกจากเอฟเฟกต์ที่มีให้นั้น ยังสามารถสร้างขึ้นเองได้อีกด้วยจากโปรแกรม Editor ของ Mercury Particle Engine

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.25 เอฟเฟกต์ต่างๆที่แสดงบนโปรแกรม Editor ของ Mercury Particle Engine

2.6 การใส่เสียงประกอบเกมด้วย XACT

XACT เป็นเครื่องมือสำหรับรวบรวมทรัพยากรเสียงลงในเกม [10] โดยทั่วไปแล้วนักออกแบบเสียงจะตัดต่อเสียงจากเสียงที่ขายในเชิงพาณิชย์ เพื่อนำมาทำเป็นเสียงหลักสำหรับเกมในรูปแบบของไฟล์ Waves หลังจากที่ได้ตัดต่อเป็นไฟล์ Waves แล้ว นักออกแบบจะใช้ XACT GUI หรือ Command Line ในการจัดการทรัพยากรเสียงต่างๆที่มีให้เป็นหมวดหมู่ และหลังจากนั้นนักพัฒนาเกมจะใช้ XACT API ในการรวบรวมเสียงทั้งหมดลงในเกมอีกที

XACT ประกอบด้วย 3 ส่วนหลัก ดังนี้

- XACT API

ช่วยในการรวบรวมของทรัพยากรเสียงเข้าไปในเกม สร้างวิธีการโหลด และการเล่นไฟล์เสียง รวมถึงการจัดการไฟล์เสียงเหล่านี้ด้วย โดยทั่วไปนักพัฒนาเกมจะเป็นผู้ใช้งานในส่วนนี้

- XACT GUI

เป็นเครื่องมือแบบกราฟฟิกสำหรับจัดทรัพยากรเสียงที่จะใช้ใน XACT API โดยทั่วไปนักออกแบบเสียงจะเป็นผู้ใช้งานในส่วนนี้

- XACT Command Line Tool

มีส่วนติดต่อผ่าน Command Line ซึ่งจะถูกรวมในขั้นตอน Build ของเกม

2.6.1 XACT Features

2.6.1.1 Features สำหรับนักออกแบบเสียง

- สามารถบันทึกไฟล์เสียงได้หลายประเภท เช่น WAV, AIFF, ADPCM, XMA, xWMA
- รองรับระบบเสียง Stereo และ Multichannel

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถจัดการไฟล์เสียงที่มีอยู่ได้
- สามารถทดสอบและตั้งค่าเสียงได้ขณะที่เกมกำลังทำงานอยู่

2.6.1.2 Features สำหรับนักพัฒนาเกม

- สามารถรวบรวมไฟล์ Wave, Sound และ Cue ที่ได้รับจากผู้ออกแบบเสียงได้โดยง่าย
- สนับสนุนการทำงานแบบ In-memory และ Streaming wave banks อย่างมีประสิทธิภาพ
- สามารถดาวน์โหลดมาใช้ได้โดยไม่เสียค่าใช้จ่าย
- สามารถเข้าถึงไฟล์เสียงได้ด้วย Cue name แทนที่จะต้องเข้าไปที่ไฟล์เสียงนั้นโดยตรง
- สามารถรู้เหตุการณ์ต่างๆที่เกิดกับเสียงได้

2.7 DirectX

DirectX เป็นชุดคำสั่ง (Application Programming Interface: API) ทางด้านมัลติมีเดีย ถูกพัฒนาขึ้นภายใต้เทคโนโลยี COM (Component Object Model) โดยบริษัทไมโครซอฟต์ ซึ่งทางบริษัทได้ติดตั้ง DirectX มาพร้อมกับระบบปฏิบัติการตั้งแต่ Windows 95 พร้อมเปิดให้ผู้ใช้ทั่วไปสามารถดาวน์โหลดเวอร์ชั่นใหม่ได้ฟรีจากเว็บไซต์ของไมโครซอฟต์ ปัจจุบันได้พัฒนามาถึงเวอร์ชั่น 10.0 ที่ติดตั้งมาพร้อมกับระบบปฏิบัติการ Windows Vista

DirectX สามารถติดต่อกับฮาร์ดแวร์ได้โดยตรง เช่น Graphic Card, Sound Card, 3D Accelerator โดยไม่ต้องผ่านระบบปฏิบัติการ และมีความสามารถในการวิเคราะห์ความสามารถของฮาร์ดแวร์เพื่อปรับแต่งค่าต่างๆให้มีความเหมาะสมกับการใช้งาน ทำให้โปรแกรมมัลติมีเดีย หรือเกมที่ติดต่อผ่าน DirectX ใช้ความสามารถด้านฮาร์ดแวร์ของระบบได้อย่างเต็มประสิทธิภาพ และยังสามารถทำงานบนเครื่องคอมพิวเตอร์ที่มีระบบฮาร์ดแวร์ที่แตกต่างกันได้ มีส่วนประกอบสำคัญ 2 ส่วนดังนี้

2.7.1 DirectX Runtime

เป็นส่วนประกอบของ DirectX ทางฝั่งผู้ใช้โปรแกรมหรือผู้เล่นเกม เพื่อให้โปรแกรมมัลติมีเดีย หรือเกมสามารถทำงานกับฮาร์ดแวร์ได้ โดยปกติแล้วจะถูกติดตั้งมาพร้อมกับระบบปฏิบัติการ Windows ถ้าเครื่องคอมพิวเตอร์เครื่องใดไม่มี DirectX Runtime ติดตั้งไว้ จะไม่สามารถเล่นเกมได้ หรือหากมีเวอร์ชั่นไม่สอดคล้องกับที่โปรแกรมต้องการ ก็จะไม่สามารถทำงานได้เช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7.2 DirectX SDK (DirectX Software Development Kit)

เป็นส่วนของผู้พัฒนาโปรแกรม โดยไมโครซอฟต์พัฒนา DirectX SDK สำหรับใช้ในการพัฒนาโปรแกรมหลายภาษา เช่น C/C++ C# และ Visual Basic .NET ซึ่งประกอบด้วยชุดคำสั่งต่างๆ ดังนี้

- Direct Graphics

ทำหน้าที่จัดการเกี่ยวกับการแสดงผลทางจอภาพ ประกอบด้วยฟังก์ชัน DirectDraw ที่สนับสนุนการสร้างภาพสองมิติ และ Direct3D (D3D) สำหรับการสร้างภาพสามมิติ

- Direct Audio

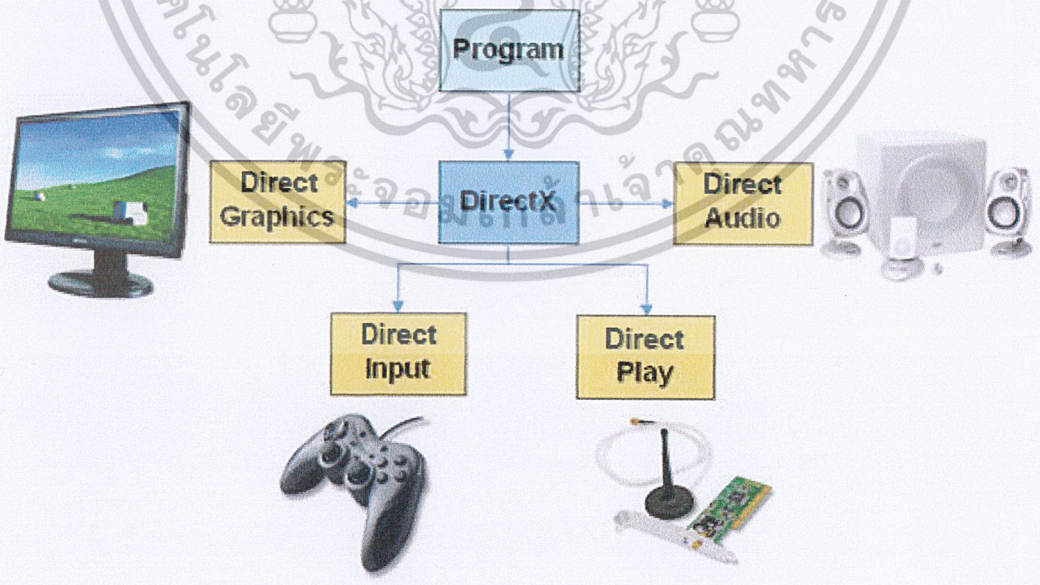
ทำหน้าที่จัดการการประมวลผลเกี่ยวกับเสียงต่างๆ

- Direct Input

ประกอบด้วยฟังก์ชันในการรับส่งข้อมูลจากผู้ใช้ผ่านอุปกรณ์ต่างๆ เช่น เมาส์ คีย์บอร์ด หรือก้านควบคุม (Joystick) ซึ่งใน DirectX 10.0 ได้มีการเพิ่มเติมการรองรับอุปกรณ์อินพุทนอกเหนือจาก Direct Input โดยเรียกว่า XInput ซึ่งมาจากอุปกรณ์ควบคุมตัวใหม่ของเครื่องเกม Xbox 360

- Direct Play

ประกอบด้วยฟังก์ชันที่ใช้ในการเชื่อมต่อเครือข่าย (Network)

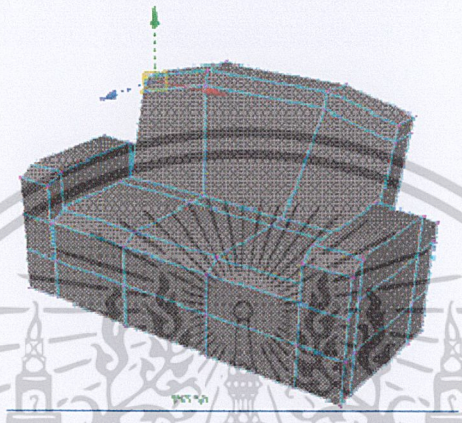


รูปที่ 2.26 การทำงานของ DirectX

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

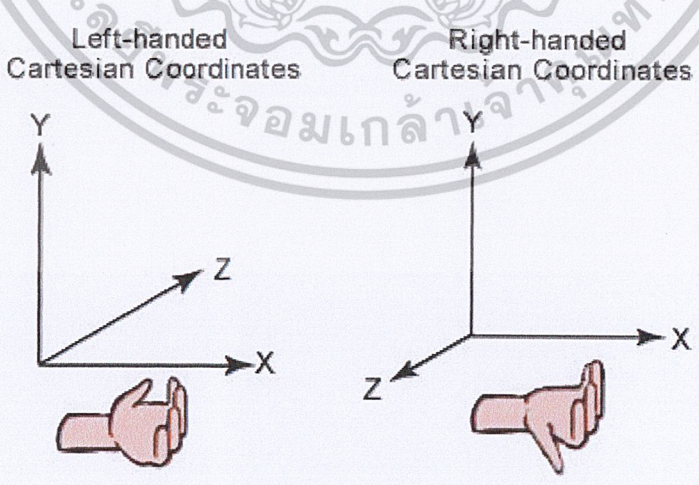
2.8 ระบบพิกัดสามมิติ

โลกสามมิติเกิดจากพิกัดสามค่ารวมกัน คือ X Y และ Z เรียกค่าพิกัดเหล่านี้ว่าจุด Vertex เมื่อนำจุด Vertex สามจุดมารวมกัน และใช้หลักการทางคณิตศาสตร์เรื่องเวกเตอร์และระนาบเข้ามาเกี่ยวข้อง จะทำให้เกิดรูปสามเหลี่ยมหรือมากกว่า ที่เรียกว่า Polygon และเมื่อนำ Polygon แต่ละอันมารวมกัน จะทำให้เกิดวัตถุสามมิติ



รูปที่ 2.27 วัตถุที่เกิดจากการรวมจุด Vertex และ Polygon

ค่าพิกัดในแต่ละแกนมีทั้งค่าบวกและค่าลบ ค่าที่เป็น 0 ของทุกแกนเรียกว่าจุดกำเนิด (Origin) ซึ่งอยู่ตรงจุดกึ่งกลางของโลกสามมิติ ส่วนระบบพิกัดสามารถแบ่งได้ 2 แบบคือ ระบบพิกัดมือซ้าย (Left-handed Coordinate System) และระบบพิกัดมือขวา (Right-handed Coordinate System) สำหรับใน XNA จะใช้ระบบพิกัดมือขวา (Right-handed Coordinate System) [3]



รูปที่ 2.28 ระบบพิกัดมือซ้าย และมือขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.8.1 ทฤษฎีโมเดลสามมิติใน XNA

- Vector

ใน XNA จะมี Vector2 Vector3 และ Vector4 ให้ใช้ ใน Vector2 นั้นจะมีส่วนประกอบคือ x กับ y สามารถใช้ในการทำงานเกี่ยวกับภาพสองมิติ และเมื่อใช้ Texture ใน Vector3 จะมีแกน z เพิ่มเข้ามาเพื่อใช้เกี่ยวกับภาพสามมิติ

- Effect

เพื่อให้การแสดงผลสามมิติมีความสมจริง ซึ่งใน XNA จะใช้ High Level Shading Language (HLSL) ในการควบคุม หรืออาจใช้คลาส BasicEffect ที่มีอยู่ใน XNA ก็ได้

- World Transformation

เป็นการแปลงระยะพิกัดของโมเดลสู่ระยะพิกัดจริงในรูปแบบเมทริกซ์ของ World Transformation คือ การย้าย (Translation) การหมุน (Rotation) และการย่อขยาย (Scaling)

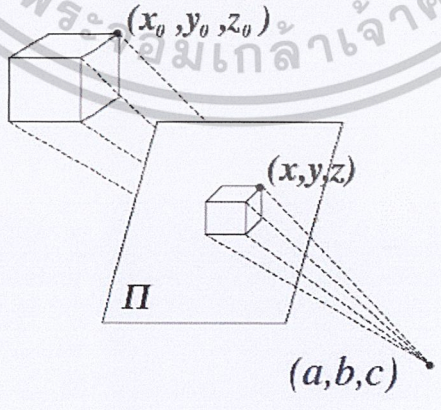
- View Transformation

การแปลงตำแหน่งของกล้อง ข้อมูลทิศทางต่างๆที่กล้องฉายไป หรือเทียบได้กับตาของมนุษย์ที่มองไปยังโลกสามมิติ

- Projection Transformation

เป็นการกำหนดรูปแบบ สัดส่วนความกว้างต่อความสูง ระยะใกล้สุด และระยะไกลสุดที่กล้องมองเห็น มีลักษณะการแสดงผล 2 แบบคือ

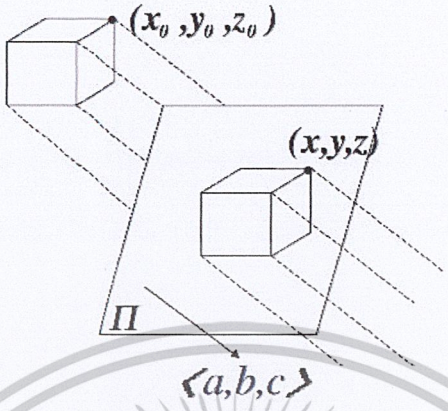
1) **Perspective Projection** เป็นการแสดงทัศนียภาพตามที่ตาเห็น คือวัตถุที่อยู่ไกลจะมีขนาดเล็ก ส่วนวัตถุที่อยู่ใกล้จะมีขนาดใหญ่ จึงทำให้เกิดความลึกของวัตถุเวลามอง



รูปที่ 2.29 ภาพแบบ Perspective

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) **Orthographic Projection** เป็นการแสดงทัศนียภาพขนาดเดียว คือไม่ว่าวัตถุจะอยู่ใกล้ไกล ขนาดของวัตถุจะเท่าเดิมเสมอ



รูปที่ 2.30 ภาพแบบ Orthographic

2.8.2 การแสดงผลโมเดลสามมิติ

การแสดงผลโมเดลสามมิติจะมีเรื่องของกล้อง (Camera) เข้ามาเกี่ยวข้องด้วย [5] คุณสมบัติของกล้องจะถูกเก็บอยู่ในรูปของออปเจ็คต์เมทริกซ์ ซึ่งกล้องใน XNA จะประกอบด้วยออปเจ็คต์เมทริกซ์ 2 อัน คือ View Matrix ที่จะเก็บข้อมูลตำแหน่งของกล้องใน World ทิศทางที่กล้องกำลังฉาย และรวมถึงการกำหนดตำแหน่งของกล้องอีกออปเจ็คต์หนึ่งคือ Projection Matrix ที่เก็บข้อมูลที่กำหนดคุณสมบัติของกล้องที่ขึ้นอยู่กับมุมของ View Matrix ว่าระยะเท่าใดที่กล้องจะยังสามารถมองเห็นได้ ซึ่งเมทริกซ์นี้จะแทนการแปลงจากโลกสามมิติไปเป็นระนาบสองมิติของหน้าจอ

View Matrix จะรับพารามิเตอร์ประเภท Vector3 สามตัว คือ ตำแหน่งของกล้อง (Position) ตำแหน่งของเป้าหมาย (Target) และเวกเตอร์ที่กำหนดความสูงของกล้อง (upVector) โดยสามารถเรียกใช้ผ่านเมธอด CreateLookAt ซึ่งอยู่ในคลาส Matrix ของ XNA อยู่แล้ว

```
Matrix viewMatrix = Matrix.CreateLookAt (position, target, upVector);
```

ตารางที่ 2.3 พารามิเตอร์ของเมธอด Matrix.CreateLookAt

Parameter	Type	Description
cameraPosition	Vector3	พิกัดของกล้อง
cameraTarget	Vector3	พิกัดของจุดที่กล้องกำลังฉาย
cameraUpVector	Vector3	เวกเตอร์ที่ชี้ในทิศบน

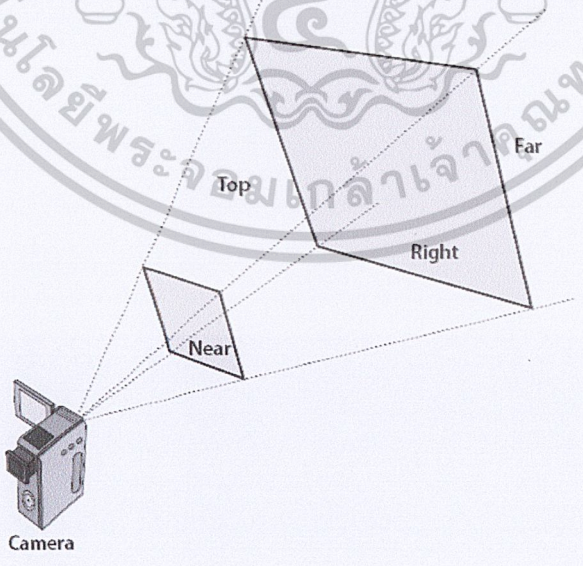
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การสร้าง Projection Matrix จะต้องรู้มุมของพื้นที่ในการมองเห็น (Field of view) ในหน่วยเรเดียน อัตราส่วนของหน้าจอ และระนาบระยะใกล้ไกล ซึ่งระนาบระยะใกล้ไกลจะเป็นตัวกำหนดระนาบที่จะวาดวัตถุระหว่างระนาบทั้งสอง ส่วนวัตถุอื่นๆที่อยู่นอกระนาบจะไม่ถูก Renders โดยการทำ Projection Matrix นั้น จะทำผ่านเมธอด Matrix.CreatePerspectiveFieldOfView ซึ่งอยู่ในคลาส Matrix เช่นกัน

```
projectionMatrix= Matrix.CreatePerspectiveFieldOfView(MathHelper.PiOver4,
                  GraphicsDevice.Viewport.AspectRatio, 1, 400);
```

ตารางที่ 2.4 พารามิเตอร์ของ Matrix.CreatePerspectiveFieldOfView

Parameter	Type	Description
fieldOfView	Float	มุมของกล้องในหน่วยเรเดียน โดยปกติจะมีค่า 45 องศา หรือ $\pi/4$
aspectRatio	Float	อัตราส่วนของกล้อง โดยปกติจะใช้เท่ากับความกว้างและความสูงของหน้าจอ
nearPlaneDistance	Float	ระยะใกล้สุดของกล้องที่สามารถมองเห็นวัตถุได้ ก่อนที่จะมองไม่เห็นวัตถุ
farPlaneDistance	Float	ระยะไกลสุดของกล้องที่สามารถมองเห็นวัตถุได้ ก่อนที่จะมองไม่เห็นวัตถุ



รูปที่ 2.31 พื้นที่การมองในโลกสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.9 3D Studio Max (3ds Max)

โปรแกรม 3ds Max ถูกพัฒนาขึ้นโดยบริษัท Autodesk, Inc. เป็นโปรแกรมสำหรับสร้างงานสามมิติที่ได้รับความนิยมอย่างมาก และได้ถูกพัฒนาโปรแกรมมาจนถึงเวอร์ชั่น 2011 แล้ว ซึ่งในปัจจุบันงานด้านสามมิติได้เข้ามามีบทบาทในงานด้านต่างๆมากขึ้น เช่น งานด้านภาพยนต์และสื่อโฆษณา งานด้านสถาปัตยกรรม งานด้านวิทยาศาสตร์ งานด้านการสร้างเว็บไซต์ งานด้านการพัฒนาเกม เป็นต้น

รูปที่ 2.32 โปรแกรม 3ds Max 2011

2.9.1 ขั้นตอนการสร้างงานสามมิติ

ในการสร้างงานสามมิติใดๆจะมีขั้นตอนในการทำงาน ซึ่งเป็นขั้นตอนในการทำงานของโปรแกรมสามมิติเท่านั้น ไม่ได้รวมถึงขั้นตอนในการตัดต่อหรืออื่นๆ มีทั้งหมด 5 ขั้นตอนหลัก ได้แก่

1) **Modeling** คือ การใช้เครื่องมือของโปรแกรมสร้างสิ่งที่ออกแบบไว้ให้เป็นรูปร่าง เมื่อทำเสร็จจะได้วัตถุคล้ายดินน้ำมัน เรียกว่า โมเดล (Model)

2) **Shading** คือ ขั้นตอนการนำโมเดลมาใส่รายละเอียดพื้นผิว (Material and Texture)

3) **Animation** คือ การใส่การเคลื่อนไหวให้กับโมเดล เช่น การวิ่ง การเดิน การกระโดด

4) **Light & Cam** คือ นำผลงานที่ได้มาจัดแสงเงา และมุมกล้อง ซึ่งถือเป็นขั้นตอนที่ต้องใช้เวลาในการศึกษานาน เพื่อให้ได้ผลงานที่สวยงาม

5) **Rendering** คือ ขั้นตอนสุดท้ายที่จะนำผลงานมาประมวลผลรวมกันให้ออกมาเป็นภาพใน Format ต่างๆเพื่อนำไปใช้ในงานที่ต้องการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.33 โมเดลสามมิติที่เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
การวิเคราะห์และออกแบบระบบ

มายฟิตเนส (myFitness) เป็นเกมที่ต้องการให้ผู้เล่นได้ใช้การเคลื่อนไหวของร่างกายในการควบคุมเกม ซึ่งถือเป็นการออกกำลังกายได้อีกทางหนึ่งโดยไม่ต้องออกจากบ้าน เพื่อให้การพัฒนารูปแบบเกมมีความสอดคล้องจึงมีการนำเรื่องของสมรรถภาพทางกายของมนุษย์เข้ามาประยุกต์ใช้ในเกม นอกจากจะทำให้เกมมีความน่าสนใจแล้ว ยังสามารถบอกได้ว่าผู้เล่นมีสมรรถภาพอยู่ในเกณฑ์ดีหรือไม่ดีได้อีกด้วย

3.1 ความหมายและความสำคัญของสมรรถภาพทางกาย

สมรรถภาพทางกาย (Physical Fitness) หมายถึง [13] สภาวะของร่างกายที่อยู่ในสภาพที่ดีที่จะช่วยให้บุคคลสามารถทำงานได้อย่างมีประสิทธิภาพ บุคคลที่มีสมรรถภาพทางกายที่ดีนั้นจะสามารถปฏิบัติภารกิจต่างๆในชีวิตประจำวัน การออกกำลังกาย การเล่นกีฬา รวมถึงการแก้ไขสถานการณ์ต่างๆได้เป็นอย่างดี

เมื่อพิจารณาถึงสมรรถภาพทางกายภาพตลอดชั่วชีวิตของคนเราพบว่า [15] คนเรานั้นจะมีสมรรถภาพทางกายดีขึ้นจากวัยเด็กเรื่อยมาจนถึงจุดสูงสุดในช่วงอายุ 25 – 30 ปี หลังจากนั้นสมรรถภาพทางกายและวุฒิภาวะจะเริ่มลดลงตามลำดับ

การมีสุขภาพดีเป็นรากฐานของการมีสมรรถภาพทางกายที่ดี ดังนั้นสมรรถภาพจึงเป็นตัวบ่งชี้ถึงความสามารถของร่างกายในการที่จะประกอบกิจกรรมต่างๆได้อย่างมีประสิทธิภาพมากน้อยเพียงใดโดยทั่วไปสมรรถภาพทางกายแบ่งออกเป็น 2 ชนิด คือสมรรถภาพทางกายที่สัมพันธ์กับสุขภาพ (Health – Related Physical Fitness) และสมรรถภาพทางกายที่สัมพันธ์กับทักษะ (Skill – Related PhysicalFitness)

3.1.1 สมรรถภาพทางกายที่สัมพันธ์กับสุขภาพ (Health – Related Physical Fitness)

หมายถึง ความสามารถของระบบต่างๆในร่างกาย ประกอบด้วย ความสามารถเชิงสรีรวิทยาด้านต่างๆ ที่ช่วยป้องกันบุคคลจากโรคที่มีสาเหตุจากภาวะการขาดการออกกำลังกาย ซึ่งนับเป็นปัจจัยที่สำคัญของการมีสุขภาพดี ความสามารถเหล่านี้สามารถปรับปรุงพัฒนาหรือคงสภาพได้โดยการออกกำลังกายอย่างสม่ำเสมอ อีกทั้งช่วยลดอัตราความเสี่ยงของการเกิดปัญหาทางด้านสุขภาพต่างๆด้วย โดยมีองค์ประกอบดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ความแข็งแรงของกล้ามเนื้อ (Muscle Strength) คือความสามารถของกล้ามเนื้อที่เกิดความตึงตัวเพื่อใช้แรงในการยกหรือดึงสิ่งของต่างๆ
2) ความอดทนของกล้ามเนื้อ (Muscle Endurance) คือความสามารถของกล้ามเนื้อที่ได้ประกอบกิจกรรมซ้ำซากได้เป็นระยะเวลานานอย่างมีประสิทธิภาพ
3) ความอดทนของระบบไหลเวียนโลหิตและระบบหายใจ (Cardio Respiratory Endurance) คือ ความสามารถของหัวใจ ปอด และหลอดเลือดในการที่จะลำเลียงออกซิเจน และสารอาหารไปยังเซลล์กล้ามเนื้อ ทำให้ร่างกายสามารถยืนยัดที่จะทำงานหรือออกกำลังกายที่ใช้กล้ามเนื้อมัดใหญ่เป็นระยะเวลานานได้
4) ความอ่อนตัว (Flexibility) คือความสามารถของข้อต่อต่างๆในการที่จะเคลื่อนไหวได้อย่างกว้างขวาง
5) องค์ประกอบของร่างกาย (Body Composition) คือ ดัชนีประมาณค่าที่ทำให้ทราบถึงเปอร์เซนต์ของน้ำหนักที่เป็นส่วนของไขมันที่มีอยู่ในร่างกาย โดยวัดออกมาเป็นเปอร์เซนต์ไขมัน (%Fat)

3.1.2 สมรรถภาพทางกายที่สัมพันธ์กับทักษะ (Skill – Related Physical Fitness)

หมายถึง ความสามารถของร่างกายที่ช่วยให้บุคคลสามารถประกอบกิจกรรมทางกายโดยเฉพาะอย่างยิ่งในการเล่นกีฬาได้ดี มีองค์ประกอบ 6 ด้าน ดังนี้

1) ความคล่องแคล่ว (Agility) คือความสามารถในการเปลี่ยนทิศทางการเคลื่อนที่ได้อย่างรวดเร็วและสามารถควบคุมได้
2) การทรงตัว (Balance) คือความสามารถในการรักษาความสมดุลร่างกายในขณะอยู่กับที่หรือเคลื่อนที่
3) การประสานสัมพันธ์ (Co-ordination) คือความสามารถในการเคลื่อนไหวร่างกายได้อย่างราบรื่น กลมกลืน และมีประสิทธิภาพซึ่งเป็นการทำงานประสานสอดคล้องกันระหว่าง ตา-มือ-เท้า
4) พลังกล้ามเนื้อ (Muscle Power) คือ ความสามารถของกล้ามเนื้อส่วนใดส่วนหนึ่ง หรือหลายๆส่วนของร่างกายในการหดตัวเพื่อทำงานด้วยความเร็วสูง หรืองานที่ได้เป็นผลรวมของความแข็งแรงและความเร็วที่ใช้ในระยะเวลาสั้นๆ เช่น การกระโดดไกล การทุ่มน้ำหนัก เป็นต้น
5) เวลาปฏิกิริยาตอบสนอง (Reaction) หมายถึงระยะเวลาที่ร่างกายใช้ในการตอบสนองต่อสิ่งเร้าต่างๆ
6) ความเร็ว (Speed) คือ ความสามารถในการเคลื่อนที่จากที่หนึ่งไปยังอีกที่หนึ่งอย่างรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.3 ความสำคัญของสมรรถภาพทางกาย

สมรรถภาพทางกายมีความสำคัญต่อการปรับปรุงคุณภาพชีวิตที่จะทำให้มนุษย์สามารถปฏิบัติงานได้อย่างมีประสิทธิภาพผลของการเป็นผู้ที่มีสมรรถภาพทางกายที่ดีคือลดอัตราการเสี่ยงต่อการเป็นโรคหัวใจเพิ่มพูนประสิทธิภาพของระบบต่างๆในร่างกายเช่นระบบหมุนเวียนโลหิตระบบการหายใจระบบการย่อยอาหาร ฯลฯ ทำให้รูปร่างและสัดส่วนของร่างกายดีขึ้นช่วยควบคุมไม่ให้น้ำหนักเกินหรือควบคุมไขมันในร่างกายช่วยลดความดันโลหิตสูงช่วยลดไขมันเลือดเพิ่มความคล่องตัวส่งผลให้เกิดประสิทธิภาพในการทำงานมากขึ้น

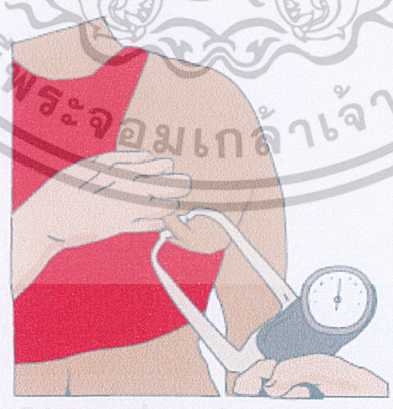
3.2 วิธีการทดสอบสมรรถภาพ และการวิเคราะห์ความเป็นไปได้ของเกม

สมรรถภาพทางกายแต่ละชนิดย่อมมีรูปแบบในการทดสอบที่แตกต่างกันไปตามความเหมาะสม ทำให้ต้องมีการศึกษาและวิเคราะห์ความเป็นไปได้ของเกมที่จะพัฒนาขึ้น เพื่อให้สอดคล้องกับรูปแบบการทดสอบเหล่านั้นให้มากที่สุด อีกทั้งยังต้องคำนึงถึงการตรวจจับของส่วนประมวลภาพด้วยว่าสามารถตรวจจับได้ถึงระดับใด รูปแบบที่ใช้ในการทดสอบสมรรถภาพ [14], [16] มีดังนี้

1. การวัดความหนาของไขมันใต้ผิวหนัง (Skinfold Thickness)

   ทดสอบสมรรถภาพด้าน องค์ประกอบของร่างกาย (Body Composition)

   วิธีการทดสอบ ใช้เครื่องมือวัดความหนาแน่นของไขมันใต้ผิวหนังหนีบไปที่เนื้อใต้ท่อนแขนหรือส่วนอื่นๆ



รูปที่ 3.1 การวัดความหนาของไขมันใต้ผิวหนัง

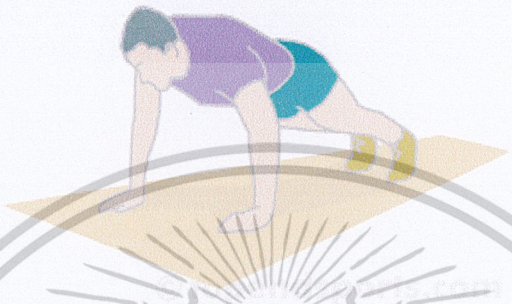
วิเคราะห์ความเป็นไปได้ มีความเป็นไปได้ยากที่จะทำเกมลักษณะนี้ออกมา เนื่องจากส่วนประมวลผลภาพไม่สามารถตรวจจับความหนาแน่นของไขมันได้ อีกทั้งการทดสอบนี้จำเป็นที่จะต้องมีเครื่องมือในการวัดผลด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ดันพื้น 30 วินาที (Push-Ups 30 Seconds)

ทดสอบสมรรถภาพด้าน ความแข็งแรงของกล้ามเนื้อและความอดทนของกล้ามเนื้อแขน รวมถึงกล้ามเนื้อส่วนบนของร่างกาย

วิธีการทดสอบ ให้ผู้ทดสอบนอนคว่ำ และเหยียดลำตัวไปข้างหน้า โดยยันฝ่ามือทั้งสองข้างไว้กับพื้น หลังจากนั้นให้ยุบข้อเพื่อดันพื้นลงไป แล้วยกแขนและลำตัวกลับขึ้นมาอยู่ในท่าเดิม



รูปที่ 3.2 ดันพื้น 30 วินาที

วิเคราะห์ความเป็นไปได้ เนื่องจากใช้ร่างกายส่วนบนในการทดสอบ เช่น ช่วงมือ หรือช่วงแขน จึงมีความเป็นไปได้ที่จะสามารถตรวจจับได้ด้วยการประมวลผลภาพ

3. นั่งงอตัวไปข้างหน้า (Sit and Reach)

ทดสอบสมรรถภาพด้าน ความยืดหยุ่น

วิธีการทดสอบ ให้ผู้ทดสอบนั่งเหยียดขาไปข้างหน้า แขนทั้งสองเหยียดตรงไปข้างหน้า จากนั้นค่อยๆก้มลำตัวลง และใช้ปลายนิ้วแตะไปข้างหน้าจนไม่สามารถก้มลำตัวลงไปได้อีก

รูปที่ 3.3 นั่งงอตัวไปข้างหน้า

วิเคราะห์ความเป็นไปได้ อาจใช้วิธีการลบฉากหลังในการแยกผู้เล่นออกจากสิ่งแวดล้อม เพื่อดูท่าทางการเคลื่อนไหวของผู้เล่นจากโครงร่างที่โปรแกรมวิเคราะห์ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. ดึงข้อ หรืองอแขนห้อยตัว (Pull-ups or Flexed-Arm Hang)

ทดสอบสมรรถภาพด้าน ความอดทนของกล้ามเนื้อส่วนบนของร่างกาย

วิธีการทดสอบ ให้ผู้ทดสอบจับราวเดี่ยวด้วยท่าคว่ำมือห่างกันประมาณ 1 ช่วงไหล่ และงอแขนเพื่อเกร็งข้อแขนไว้ เพื่อดึงตัวไว้ในท่าเดิมให้นานที่สุด โดยคางของผู้ทดสอบจะต้องอยู่เหนือราวขึ้นไป

รูปที่ 3.4 ดึงข้อ หรืองอแขนห้อยตัว

วิเคราะห์ความเป็นไปได้ น่าจะสามารถตรวจจับด้วยการประมวลภาพผลได้ แต่นำมาประยุกต์เป็นเกมได้ยากเพราะผู้ทดสอบต้องเกร็งกล้ามเนื้อเป็นเวลานาน และต้องมีอุปกรณ์ในการทดสอบ

5. วิ่งเร็ว (Sprint Test)

ทดสอบสมรรถภาพด้าน ความเร็ว

วิธีการทดสอบ ให้ผู้ทดสอบออกวิ่งเต็มที่จนถึงเส้นชัย

รูปที่ 3.5 วิ่งเร็ว

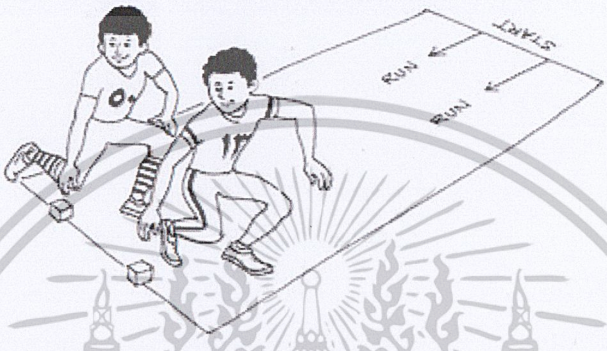
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วิเคราะห์ความเป็นไปได้ อาจจะใช้การประมวลผลภาพตรวจจับได้ แต่ต้องใช้บริเวณที่กว้างมาก และต้องมีการตรวจจับที่ชัดเจนและแม่นยำ

6. วิ่งแตะสลับ (Side Step Test)

ทดสอบสมรรถภาพด้าน ความคล่องแคล่ว

วิธีการทดสอบ ให้ผู้ทดสอบออกวิ่งไปหยิบของที่จุดหนึ่ง แล้ววิ่งกลับมาวางที่จุดออกวิ่ง



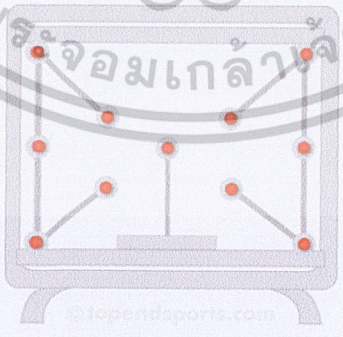
รูปที่ 3.6 วิ่งแตะสลับ

วิเคราะห์ความเป็นไปได้ มีความเป็นไปได้ที่ส่วนประมวลผลภาพจะสามารถตรวจจับได้ โดยสนใจเพียงจุดที่ต้องการให้วิ่งไปถึง 2 จุด

7. การตอบสนองต่อปุ่มไฟ (Reaction Light Board Test)

ทดสอบสมรรถภาพด้าน การประสานสัมพันธ์ และเวลาปฏิกิริยาตอบสนอง

วิธีการทดสอบ ให้ผู้ทดสอบแตะบนปุ่มที่มีไฟขึ้นอย่างรวดเร็ว



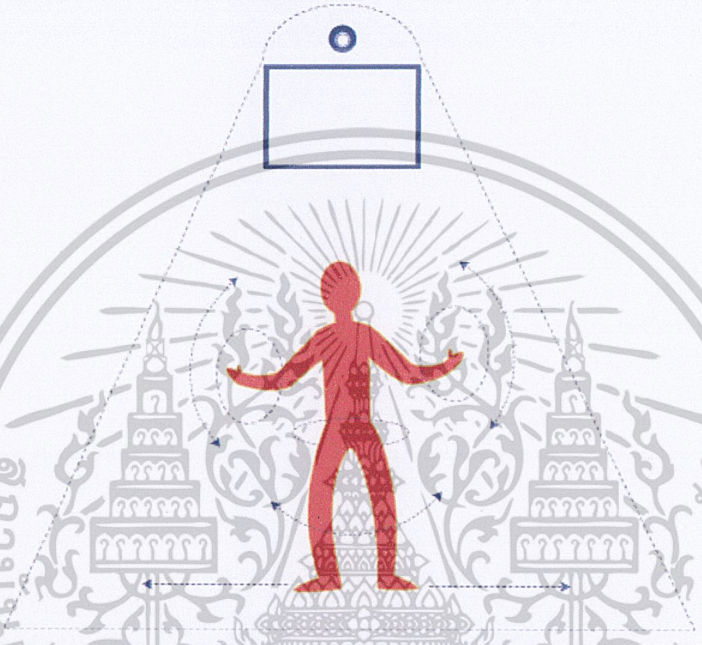
รูปที่ 3.7 การตอบสนองต่อปุ่มไฟ

วิเคราะห์ความเป็นไปได้ มีความเป็นไปได้ที่ส่วนประมวลผลภาพจะตรวจจับได้ โดยสนใจในสิ่งที่มีการเคลื่อนไหว แต่อาจจะไม่สามารถตรวจสอบได้ว่าสิ่งที่เคลื่อนไหวอยู่นั้นใช่สิ่งที่ต้องการหรือไม่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3 การออกแบบรูปแบบเกม (Game Design)

รูปแบบของเกมมายฟิตเนส (myFitness) จุดประสงค์หลักคือต้องการเน้นให้ผู้เล่นได้เคลื่อนไหวร่างกายในการเล่นและควบคุมเกม และสร้างทางเลือกในการเล่นเกมใหม่ๆ ที่นอกเหนือจากการใช้เมาส์และคีย์บอร์ด โดยการใช้กล้องเว็บแคมเป็นอุปกรณ์หลักในการรับข้อมูลอินพุทแทน ซึ่งถือเป็นการออกกำลังกายไปในตัวด้วย



รูปที่ 3.8 รูปแบบของการเล่นเกมผ่านกล้องเว็บแคม

หลังจากการวิเคราะห์ความเป็นไปได้ของแต่ละแบบทดสอบในหัวข้อที่ 3.2 แล้วนั้น มีแบบทดสอบที่สามารถนำมาประยุกต์ให้เข้าเกมและการตรวจจับของส่วนประมวลผลภาพได้ 4 ด้าน มีรายละเอียดดังนี้

3.3.1 เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ

จากวิธีทดสอบดันพื้น 30 วินาทีที่ใช้ท่อนแขนเป็นหลัก ดังนั้นจึงมีการออกแบบเกมให้มีลักษณะที่ต้องใช้กำลังแขนมากๆ และใช้ติดต่อกันเป็นเวลานาน โดยออกแบบให้เป็นเกมชกลูกบอลตามจำนวนครั้งที่กำหนด

3.3.2 เกมทดสอบสมรรถภาพการประสานสัมพันธ์

จากวิธีทดสอบการแตะปุ่มไฟ สังเกตว่าจะใช้ความสามารถเรื่องของดวงตาที่ต้องสัมพันธ์กับแขนหรือขาในการเคลื่อนไหวตอบสนอง ดังนั้นจึงออกแบบเกมให้มีสิ่งของพุ่งขึ้นมาอย่างรวดเร็ว แล้วให้ผู้เล่นตอบสนองต่อสิ่งของนั้นโดยการชก หรือเตะอย่างรวดเร็ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.3 เกมทดสอบสมรรถภาพความยืดหยุ่น

จากวิธีทดสอบการนั่งงอตัว หากออกแบบให้นั่งงอตัวธรรมดา ตัวเกมอาจจะไม่มีความน่าสนใจ หรือสนุก ดังนั้นจึงออกแบบให้มีช่องที่มีลักษณะการงอตัวในท่าทางต่างๆ โดยนำบางท่ามาจากโยคะหรือคิดขึ้นเอง แล้วให้ผู้เล่นทำท่าทางให้ผ่านช่องนั้นไปให้ได้

3.3.4 เกมทดสอบสมรรถภาพความคล่องแคล่ว

จากวิธีทดสอบวิ่งแตะสลับที่ใช้พื้นที่น้อยกว่าการวิ่งเร็ว อีกทั้งการทดสอบเน้นที่การวิ่งระยะสั้นและหมุนตัวกลับ จึงออกแบบให้เป็นเกมวิ่งหลบสิ่งของ

โดยออกแบบให้ทั้ง 4 เกมนี้เล่นต่อเนื่องกันจนจบ เกมละ 60 วินาที รวมทั้งหมด 4 นาที และสรุปผลออกมาในขั้นตอนสุดท้าย

3.3.5 ข้อจำกัดของเกม

ข้อจำกัดเรื่องลักษณะของการทดสอบที่อาจต้องใช้สถานที่ขนาดใหญ่หรือมีอุปกรณ์เฉพาะที่ใช้ในการทดสอบและการทดสอบบางอย่างส่วนประมวลผลภาพอาจตรวจจับได้ยาก จึงต้องมีการปรับเปลี่ยนรูปแบบของเกมเล็กน้อย เพื่อให้เป็นในลักษณะของเกม และสามารถตรวจจับการเคลื่อนไหวได้ ดังนั้นคะแนนสมรรถภาพที่ได้จึงอาจไม่ถูกต้องตามหลักการทดสอบสมรรถภาพ 100% เพราะระยะการจับภาพของกล้องเว็บแคมไม่ไกลมากนัก อีกทั้งการทำเกมในครั้งนี้ต้องการให้ผู้เล่นได้รับความสนุกสนาน ประกอบกับการได้ขยับเคลื่อนไหวร่างกายมากกว่า

3.4 ความต้องการของระบบ

3.4.1 ความต้องการที่เป็นฟังก์ชันการทำงาน (Functional Requirements)

1. สามารถรับภาพจากกล้องเว็บแคมได้
2. สามารถตรวจจับการเคลื่อนไหวของผู้เล่นได้
3. ส่วนการประมวลผลภาพจะต้องสามารถแยกสิ่งที่สนใจออกจากสิ่งแวดล้อมได้ และสามารถลดสัญญาณรบกวนให้น้อยลง
4. สามารถตอบสนองต่อการเคลื่อนไหวของผู้เล่นได้อย่างถูกต้อง
5. สามารถให้คะแนนสมรรถภาพทางร่างกายทั้ง 4 ด้าน ได้แก่ ความอดทนของกล้ามเนื้อ การประสานสัมพันธ์ ความยืดหยุ่น และความคล่องแคล่วของผู้เล่นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4.2 ความต้องการที่ไม่ได้เป็นฟังก์ชันการทำงาน (Non-Functional Requirements)

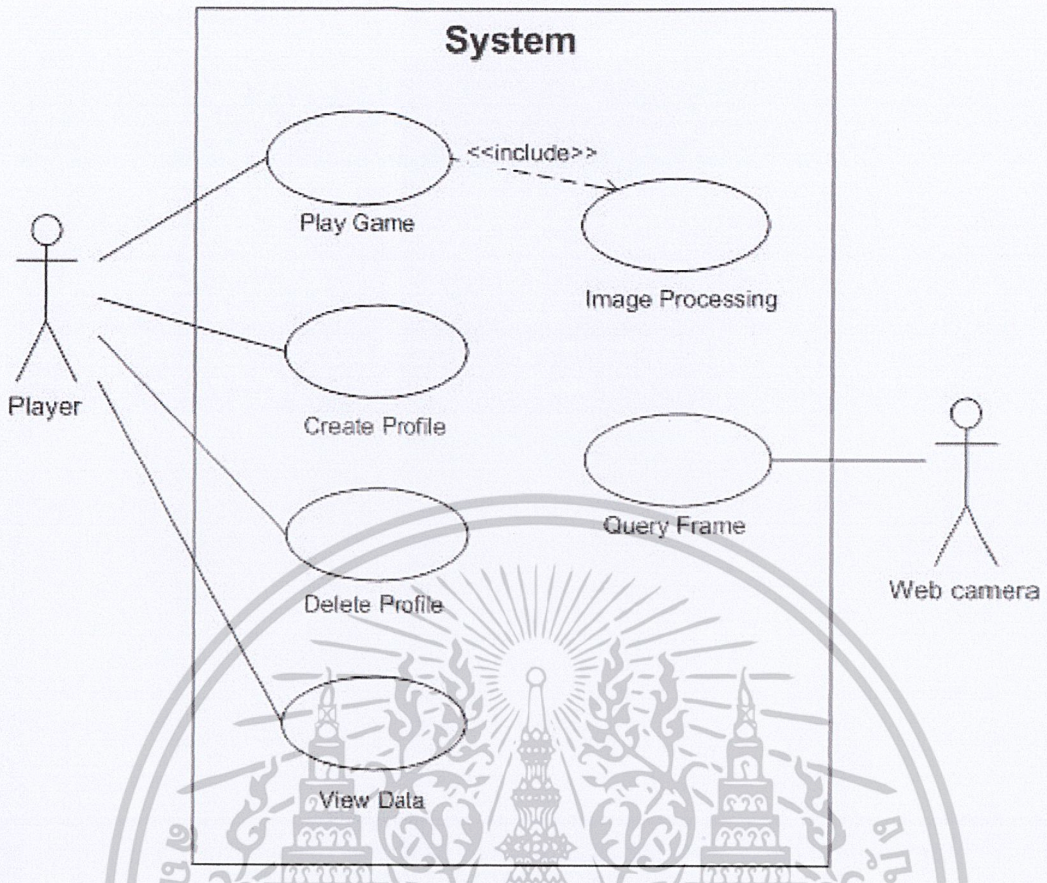
1. สามารถแสดงผลได้แบบ Real time และไม่กระตุก
2. สามารถแสดงผลภาพกราฟฟิกสองมิติและสามมิติได้
3. การแสดงผลคะแนนทางสมรรถภาพของผู้เล่น จะต้องแสดงเป็นกราฟเพื่อให้เข้าใจง่าย
4. สามารถสร้างชื่อผู้เล่นได้ 3 คน และสามารถลบเพื่อสร้างชื่อใหม่ได้
5. สามารถบันทึกข้อมูลประวัติคะแนนการเล่นของผู้เล่นแต่ละคนได้
6. เงื่อนไขความต้องการของคอมพิวเตอร์ที่ระบบสามารถใช้งานด้วยได้
   - ระบบปฏิบัติการ : Windows XP (SP2 or higher), Window Vista or 7 (32-bit)
   - หน่วยประมวลผล : CPU 1GHz
   - หน่วยความจำ : RAM 1024 MB or more
   - จอแสดงผล : 1024×768 Resolution or more
7. เงื่อนไขความต้องการขั้นต่ำของกล้องเว็บแคมที่ระบบสามารถใช้งานด้วยได้
   - Resolution 640×480 pixel
   - อัตราการแสดงผลภาพ 30 fps

3.5 การออกแบบระบบ

3.5.1 แผนภาพยูสเคส (Use Case Diagram)

เป็นแผนรูปที่แสดงภาพรวมการทำงานหลักของระบบ ทำให้ทราบว่าผู้ใช้ในแต่ละบทบาทนั้นเกี่ยวข้องกับส่วนงานใดของระบบ โดยแสดงได้ดังแผนรูปที่ 3.9

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.9 แผนภาพยูสเคสของระบบ

จากแผนภาพยูสเคสประกอบด้วย

1. ผู้กระทำกับระบบ (Actor) ได้แก่
   - ผู้เล่น (Player)
   - อุปกรณ์รับภาพ (Web Camera)
2. ฟังก์ชันการทำงานหลัก (Use Case) ได้แก่
   - การเล่นเกม (Play Game)
   - การประมวลผลภาพ (Image Processing)
   - การรับภาพ (Query Frame)
   - การสร้างข้อมูลผู้เล่น (Create Profile)
   - การลบข้อมูลผู้เล่น (Delete Profile)
   - การเรียกดูข้อมูลประวัติการเล่น (View Data)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.2 รายละเอียดยูสเคส (Use Case Description)

ตารางที่ 3.1 รายละเอียดยูสเคส Play Game

Use Case Name : Play Game	**ID**: UC-01
Primary Actor : Player	
Brief Description : ส่วนของการเล่นเกม	
Pre condition : เมื่อผู้เล่นมีข้อมูล Profile เรียบร้อยแล้ว	
Post condition : เมื่อผู้เล่นเล่นเกมครบ 1 รอบ โดย 1 รอบเกมมี 4 เกมย่อย	
Trigger Event : เมื่อผู้เล่นต้องการเล่นเกม	
Relationships : Association: - Include: Image Processing(UC-02) Extend: - Generalization: -	
Normal Flow of Events : 1. ผู้เล่นปรับกล้องเว็บแคม และระยะการเล่นตามที่เกมกำหนด 2. ผู้เล่นเลือก Profile ที่ต้องการ 3. เกมจะเริ่มพร้อมกับจับเวลา 1 นาทีต่อ 1 เกม 4. ส่วนประมวลผลภาพทำการตรวจจับการเคลื่อนไหว และตรวจสอบว่ามีการเคลื่อนไหวตามกำหนดหรือไม่ 5. ระบบแสดงคะแนนที่ผู้เล่นเล่นได้	
Alternate/Exceptional Flows : -	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 รายละเอียดยูสเคส Image Processing

Use Case Name : Image Processing	**ID:** UC-02
Primary Actor : -	
Brief Description : ประมวลผลภาพเพื่อแยกวัตถุที่สนใจออกจากพื้นหลัง และปรับปรุงภาพเพื่อลดสัญญาณรบกวน	
Pre condition : -	
Post condition : เมื่อผู้เล่นเล่นเกมครบ 1 รอบ โดย 1 รอบเกมมี 4 เกมย่อย	
Trigger Event : เมื่อผู้เล่นเคลื่อนไหวร่างกายผ่านกล้องเว็บแคม	
Relationships : Association: - Include: - Extend: - Generalization: -	
Normal Flow of Events : 1. แยกสิ่งที่สนใจออกจากสิ่งแวดล้อม 2. ทำการแปลงภาพเป็นภาพขาวดำด้วย Thresholding 3. ทำการปรับปรุงภาพด้วย Morphology 4. ตรวจจับการเคลื่อนไหวของผู้เล่น	
Alternate/Exceptional Flows : -	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 รายละเอียดยูสเคส Create Profile

Use Case Name : Create Profile	**ID:** UC-03
Primary Actor : Player	
Brief Description : การกรอกข้อมูลส่วนตัวของผู้เล่นเกี่ยวกับเพศ อายุ ส่วนสูง และน้ำหนักเพื่อใช้ในการคำนวณค่าดัชนีมวลกาย (BMI) และคะแนนสมรรถภาพเมื่อจบเกม	
Pre condition : ผู้เล่นต้องการสร้างข้อมูลผู้เล่น	
Post condition : ระบบสร้างไฟล์ข้อมูลผู้เล่น	
Trigger Event : เมื่อผู้เล่นยืนยันการสร้างข้อมูล	
Relationships : Association: - Include: - Extend: - Generalization: -	
Normal Flow of Events : 1. ผู้เล่นกรอกชื่อ 2. ผู้เล่นกรอกอายุ 3. ผู้เล่นเลือกเพศ 4. ผู้เล่นกรอกน้ำหนัก (กิโลกรัม) 5. ผู้เล่นกรอกส่วนสูง (เซนติเมตร) 6. ผู้เล่นถ่ายรูป Profile 7. ผู้เล่นยืนยันการสร้างข้อมูล 8. ระบบสร้างไฟล์ข้อมูลผู้เล่น	
Alternate/Exceptional Flows : A1ขั้นตอนที่ 1-6 กรอกข้อมูลไม่ครบหรือผิดพลาด: โปรแกรมแสดงเครื่องหมายกากบาทที่ช่องกรอกข้อมูลพร้อมคำแนะนำในการกรอกข้อมูล A2 ขั้นตอนที่ 6 เปลี่ยนรูป Profile ใหม่: ผู้เล่นกดปุ่ม Change Picture เพื่อถ่ายรูปใหม่ A3 ยกเลิกการสร้างข้อมูล: ระบบอนุญาตให้ผู้เล่นยกเลิกการสร้างข้อมูลได้ตลอดเวลา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.4 รายละเอียดยูสเคส Delete Profile

Use Case Name : Delete Profile	**ID:** UC-04
Primary Actor : Player	
Brief Description : ลบข้อมูลผู้เล่นออกจากหน้าเลือกผู้เล่น และลบไฟล์ที่เก็บข้อมูลของผู้เล่นในระบบ	
Pre condition : ผู้เล่นต้องการลบข้อมูลผู้เล่น	
Post condition : ข้อมูลถูกลบจากหน้าเลือกผู้เล่นและไฟล์ในระบบ	
Trigger Event : เมื่อผู้เล่นยืนยันการลบข้อมูล	
Relationships : Association: - Include: - Extend: - Generalization: -	
Normal Flow of Events : 1. ผู้เล่นเข้าสู่หน้าเลือกผู้เล่น 2. ผู้เล่นกดลบข้อมูลที่ต้องการ 3. ผู้เล่นยืนยันการลบข้อมูล 4. ระบบลบไฟล์ข้อมูลผู้เล่น	
Alternate/Exceptional Flows : A1 ขั้นตอนที่ 3 ยกเลิกการลบข้อมูล: ระบบกลับสู่หน้าเลือกผู้เล่น	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 รายละเอียดยูสเคส View Data

Use Case Name : View Data	**ID:** UC-05
Primary Actor : Player	
Brief Description : แสดงข้อมูลสถิติการเล่นเกมของผู้เล่นในรูปแบบกราฟ	
Pre condition : ผู้เล่นต้องการดูสถิติการเล่นเกม	
Post condition : ระบบแสดงข้อมูลการเล่นเกมในรูปแบบกราฟให้ผู้เล่นทราบ	
Trigger Event : เมื่อผู้เล่นกดปุ่มดูข้อมูลสถิติ	
Relationships : Association: - Include: - Extend: - Generalization: -	
Normal Flow of Events : 1. ผู้เล่นเลือกหัวข้อที่ต้องการทราบข้อมูล 2. ระบบดึงข้อมูลจากไฟล์สถิติที่เก็บไว้ 3. ระบบแสดงข้อมูลให้ผู้เล่นทราบ	
Alternate/Exceptional Flows : E1 ขั้นตอนที่ 2-3 ระบบตรวจสอบข้อมูลภายในไฟล์: หากไม่มีข้อมูลที่ผู้เล่นต้องการ ระบบจะไม่แสดงผลในส่วนของข้อมูลนั้น	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.6 รายละเอียดยูสเคส Query Frame

Use Case Name : Query Frame	**ID:** UC-06
Primary Actor : Web Camera	
Brief Description : ทำการรับภาพเข้ามาในระบบผ่านกล้องเว็บแคม	
Pre condition : -	
Post condition : ออกจากโปรแกรม	
Trigger Event : เมื่อโปรแกรมเริ่มทำงาน	
Relationships : Association: - Include: - Extend: - Generalization: -	
Normal Flow of Events : 1. กล้องเว็บแคมจับภาพ 2. ส่งภาพเข้าสู่ระบบ 3. แปลงภาพให้อยู่ในรูปแบบที่ใช้แสดงภาพบน XNA	
Alternate/Exceptional Flows : -	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.3 แผนภาพกิจกรรม (Activity Diagram)

เพื่อให้เกิดความเข้าใจและเห็นภาพขั้นตอนการทำงานของกิจกรรมต่างๆในยูสเคสได้ชัดเจนขึ้นจึงใช้แผนภาพกิจกรรมอธิบายลำดับการทำงาน ดังตารางที่ 3.7 – 3.12

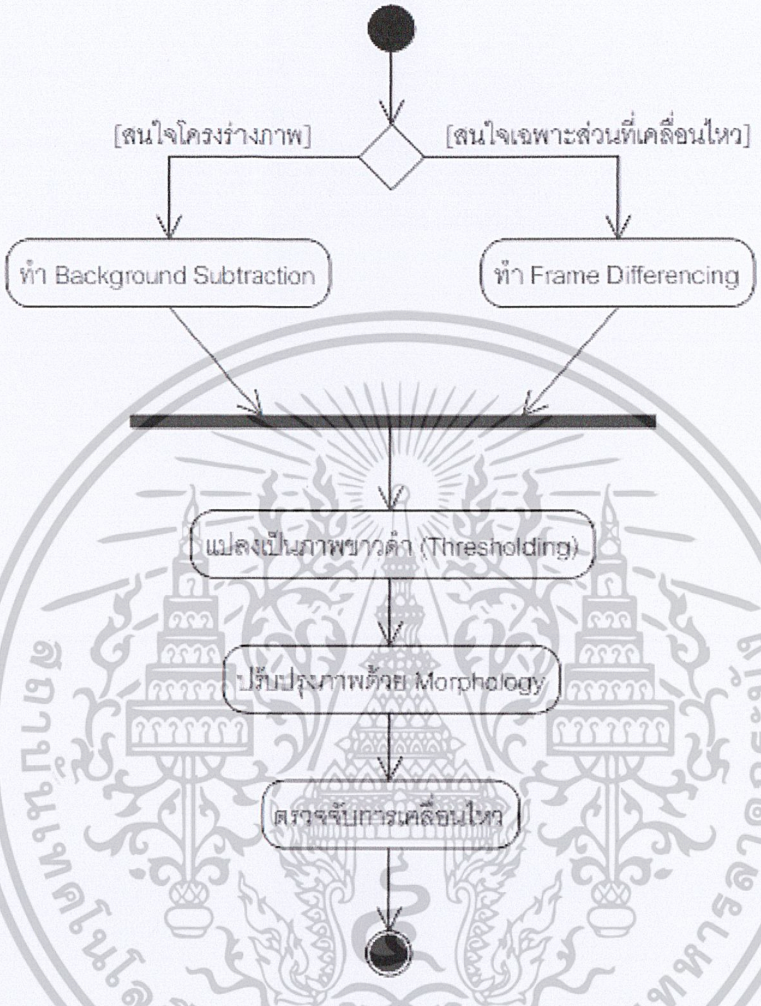
ตารางที่ 3.7 แสดงแผนภาพกิจกรรมการเล่นเกม

ชื่อแผนภาพกิจกรรม:กิจกรรมการเล่นเกม	**หมายเลข:** AC-01

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.8 แสดงแผนภาพกิจกรรมการประมวลผลภาพ

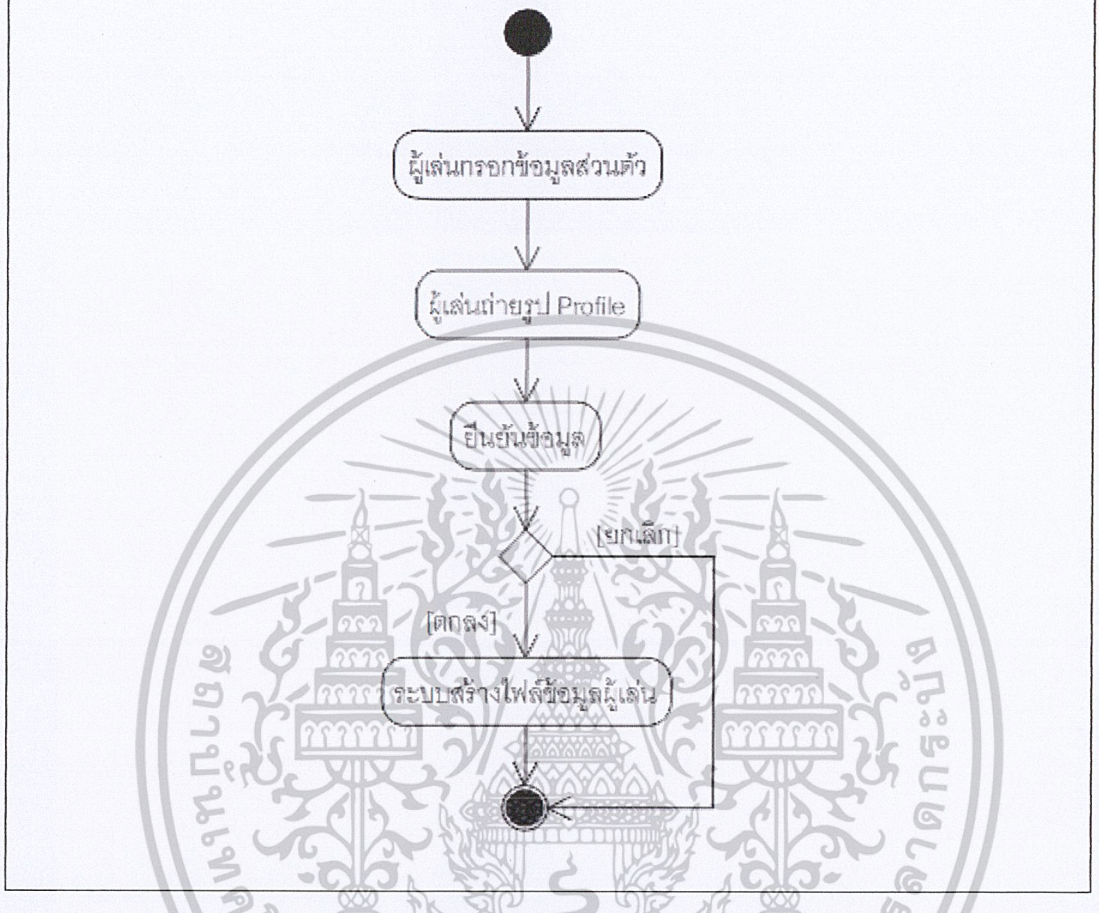
ชื่อแผนภาพกิจกรรม:กิจกรรมการประมวลผลภาพ	**หมายเลข:** AC-02



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.9 แสดงแผนภาพกิจกรรมการสร้างข้อมูลผู้เล่น

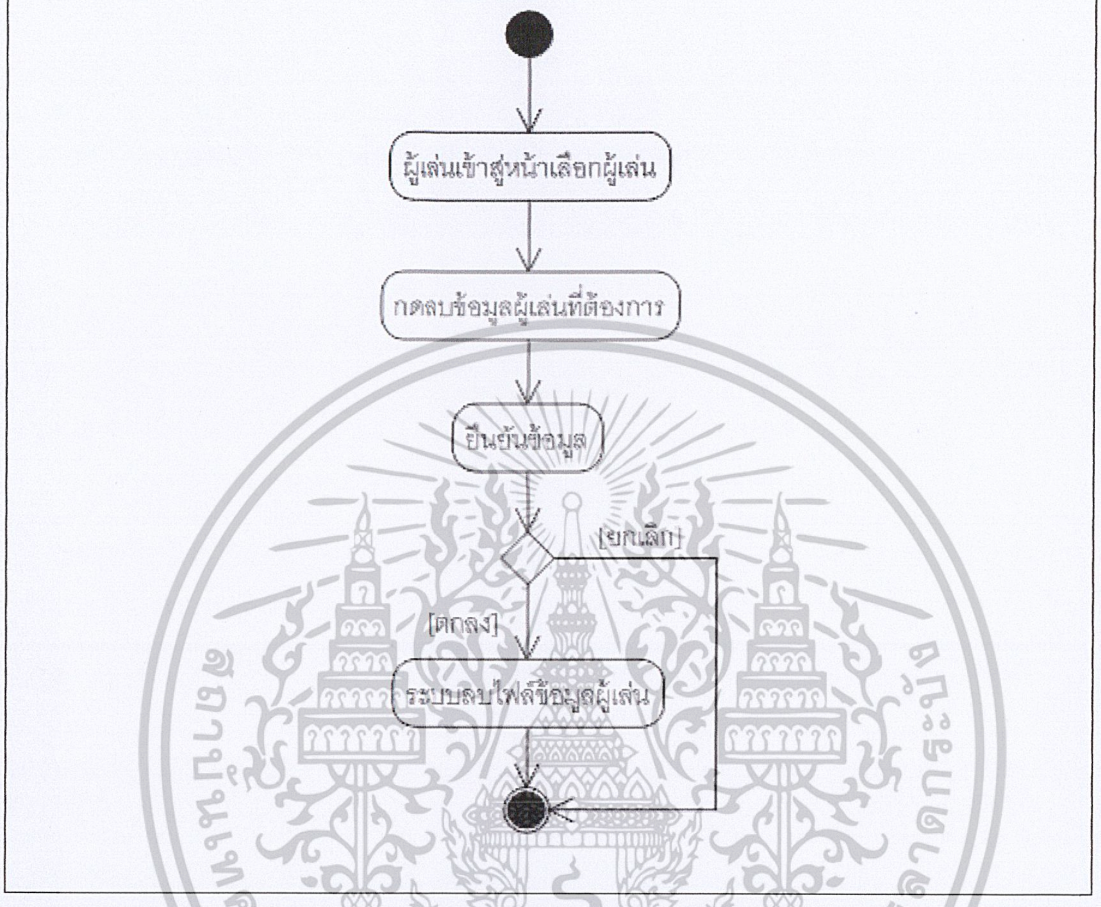
ชื่อแผนภาพกิจกรรม:กิจกรรมการสร้างข้อมูลผู้เล่น	**หมายเลข:** AC-03



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.10 แสดงแผนภาพกิจกรรมการลบข้อมูลผู้เล่น

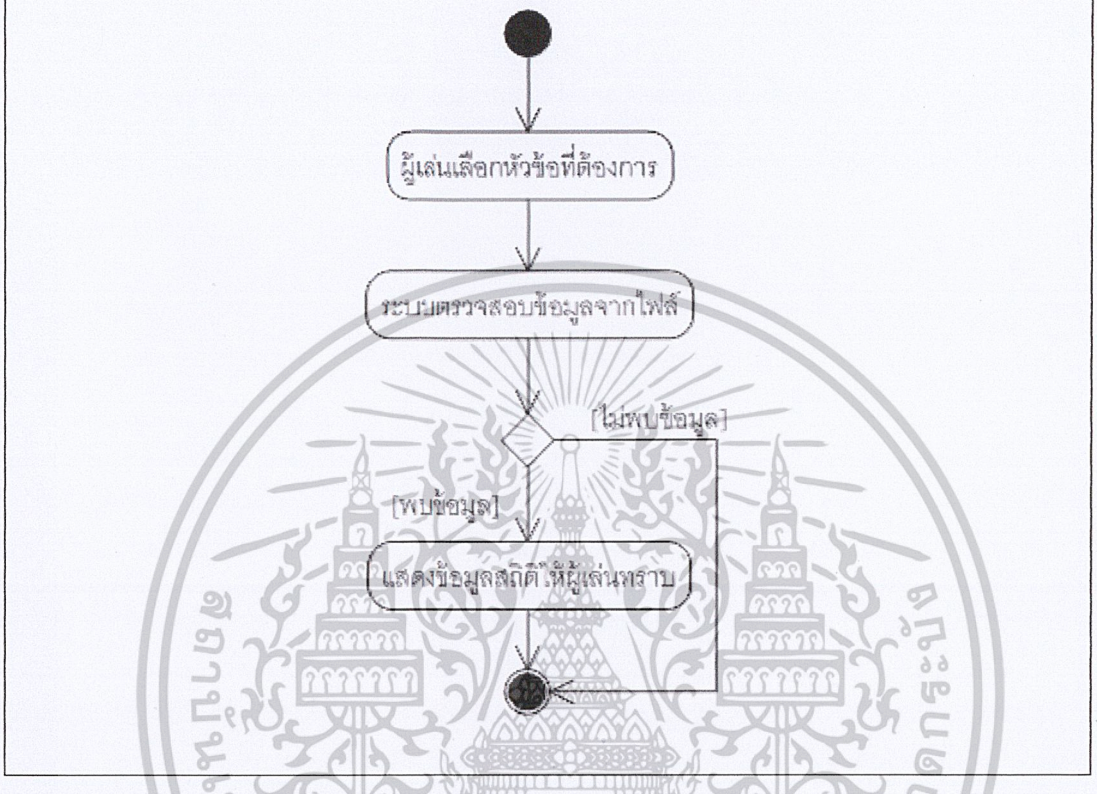
ชื่อแผนภาพกิจกรรม:กิจกรรมการลบข้อมูลผู้เล่น	**หมายเลข:** AC-04



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.11 แสดงแผนภาพกิจกรรมการดูข้อมูลสถิติการเล่นเกม

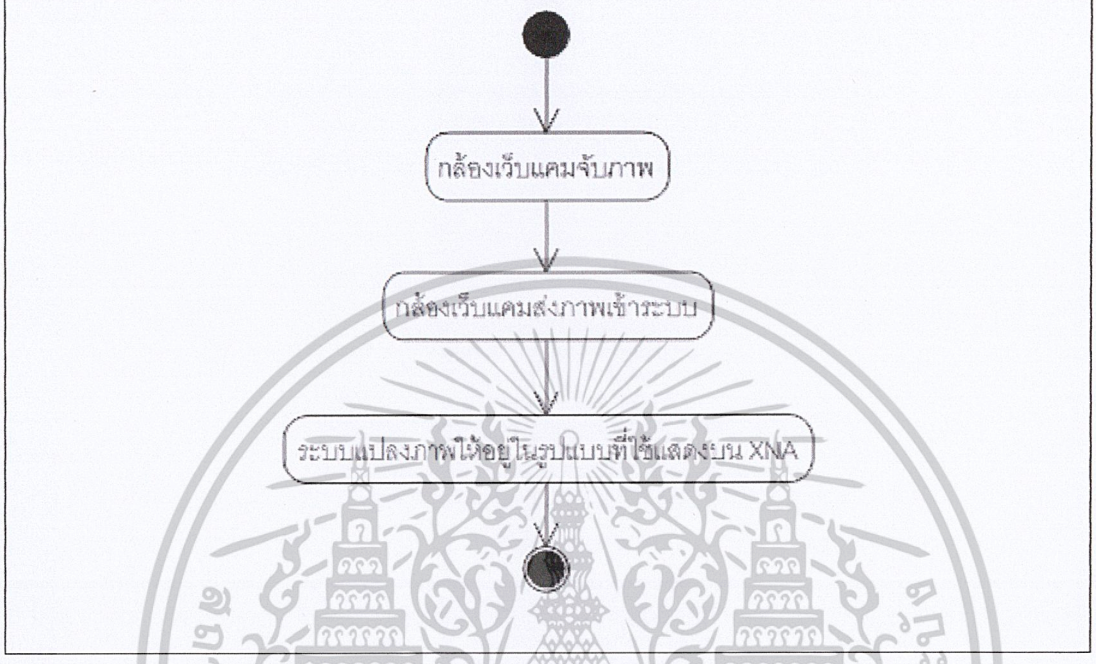
ชื่อแผนภาพกิจกรรม:กิจกรรมการดูข้อมูลสถิติการเล่นเกม	**หมายเลข:** AC-05



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.12 แสดงแผนภาพกิจกรรมการรับภาพจากกล้องเว็บแคม

ชื่อแผนภาพกิจกรรม:กิจกรรมการรับภาพจากกล้องเว็บแคม	**หมายเลข:** AC-06



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5.4 การออกแบบส่วนติดต่อกับผู้ใช้

ส่วนติดต่อกับผู้ใช้ (User Interface) เป็นส่วนที่ผู้เล่นใช้ติดต่อกับระบบในการทำงานต่างๆ เช่น การเลือกเมนู การกรอกข้อมูลส่วนตัว การแสดงผลลัพธ์เกม เป็นต้น ซึ่งถือเป็นส่วนสำคัญส่วนหนึ่งที่ต้องมีการออกแบบให้เหมาะสมกับการใช้งาน โดยมีการออกแบบที่เรียบง่าย เน้นสีสันสดใส เพื่อให้เหมาะกับเกม และมีข้อมูลที่ครบถ้วน

ภายในระบบจะแบ่งส่วนติดต่อกับผู้ใช้ออกเป็น 2 ส่วนหลักแยกตามลักษณะของการควบคุม ได้แก่

3.5.4.1 ส่วนที่ควบคุมด้วยเมาส์และคีย์บอร์ด

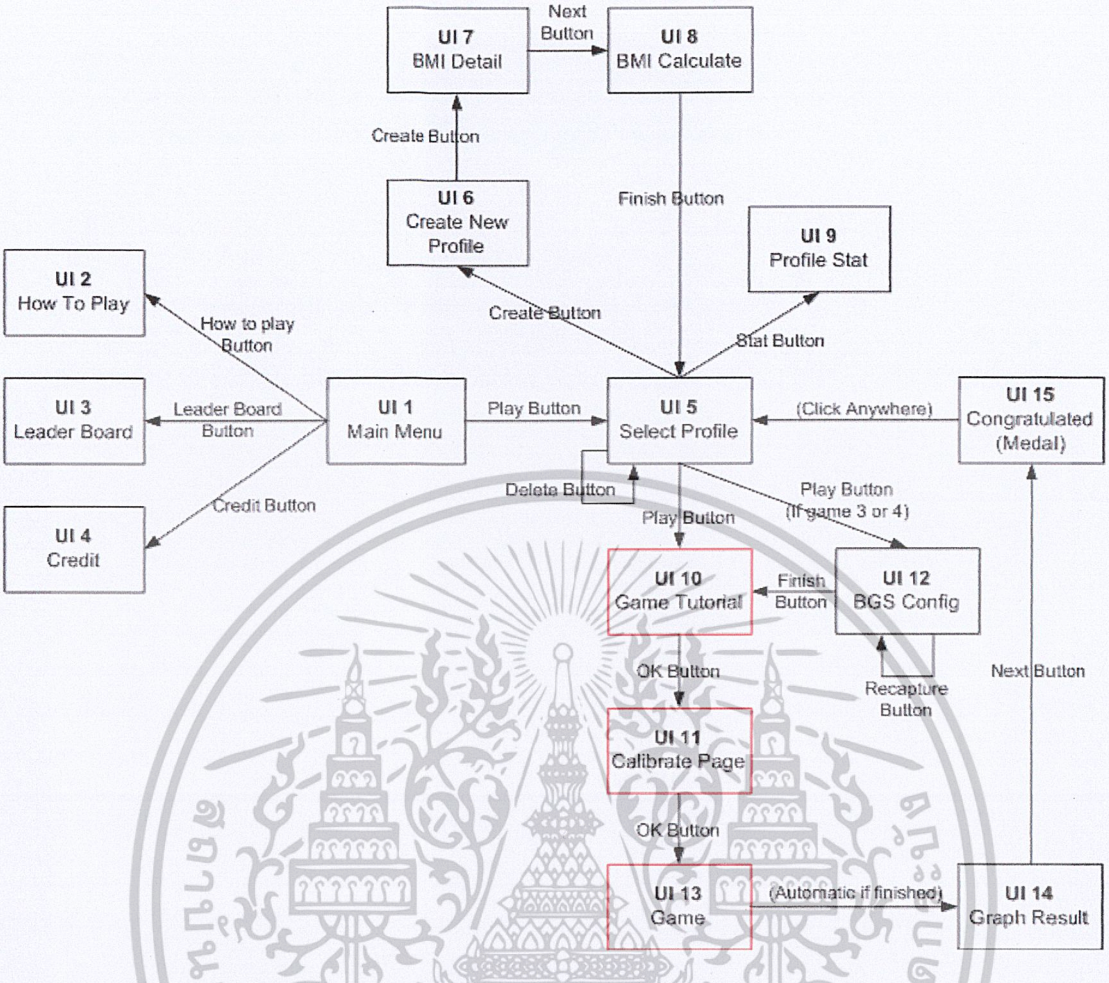
ในส่วนนี้จะเป็นส่วนที่เน้นเรื่องการแสดงข้อมูลต่างๆ เช่น การสร้างข้อมูลผู้เล่น การเลือกผู้เล่น การดูข้อมูลสถิติการเล่น การแสดงกราฟคะแนน เป็นต้น เพื่อให้การใช้งานของผู้เล่นเป็นไปอย่างสะดวก โดยจะมีสัญลักษณ์เมาส์บอกผู้เล่นไว้ที่ด้านขวาล่างของจอ

3.5.4.2 ส่วนที่ควบคุมด้วยการสัมผัส

เนื่องจากตัวเกมจะใช้หลักการประมวลผลภาพในการตรวจจับการเคลื่อนของผู้เล่นอยู่แล้ว ดังนั้นจึงออกแบบให้ตั้งแต่ส่วนที่เป็นวิธีการเล่น (Tutorial) เป็นการควบคุมด้วยการสัมผัส เพื่อให้เกิดความต่อเนื่องก่อนเข้าสู่ตัวเกม

สามารถแสดงลำดับการทำงานของส่วนติดต่อกับผู้ใช้ได้ดังแผนผังที่ 3.10 โดยส่วนที่เป็นกรอบสีดำ คือส่วนที่ควบคุมด้วยเมาส์และคีย์บอร์ด และส่วนที่เป็นกรอบสีแดง คือส่วนที่ควบคุมด้วยการสัมผัส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.10 แผนผังการทำงานของส่วนติดต่อกับผู้ใช้ (UI Flow)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การพัฒนาระบบ

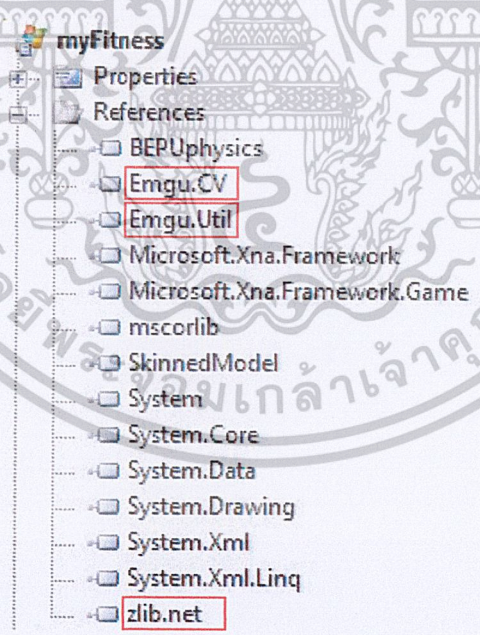
4.1 การพัฒนาส่วนประมวลผลภาพ (Image Processing)

ส่วนประมวลผลภาพถือเป็นส่วนที่มีความสำคัญมาก เพราะมีหน้าที่ในการรับภาพการเคลื่อนไหวของผู้เล่นจากกล้องเว็บแคม และนำรูปที่ได้มาประมวลผลด้วยเทคนิคการประมวลผลภาพแบบต่างๆ เพื่อให้สะดวกต่อการนำไปใช้วิเคราะห์ในขั้นตอนต่อๆไป

4.1.1 การเขียนโปรแกรมเพื่อใช้งานซอฟต์แวร์ไลบรารี EmguCV

ไลบรารี EmguCV เป็นไลบรารีที่ช่วยการทำงานเกี่ยวกับการประมวลผลภาพให้สะดวกยิ่งขึ้น รวมถึงอำนวยความสะดวกในการเขียนโปรแกรมเพื่อติดต่อกับอุปกรณ์รับภาพ เช่น กล้องเว็บแคม

1. นำ zlib.net.dll, Emgu.Util.dll และ Emgu.CV.dll จาก C:\Program Files\...\emgucv 2.1.0.793\bin เพิ่มเข้าไปในโฟลเดอร์ References ของโปรเจคที่สร้าง



รูปที่ 4.1 การเพิ่มไลบรารี EmguCV ในโปรเจค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. การเรียกใช้งาน Namespace ของ EmguCV เพื่อให้สามารถใช้งานไลบรารี EmguCV ได้

```
using Emgu.CV;
using Emgu.CV.Structure;
```

3. การเขียนโปรแกรมติดต่อกับกล้อง

```
Image<Bgr, byte> nextFrame;
capture = new Capture(0);
nextFrame = capture.QueryFrame();
```

4.1.2 การทำการประมวลผลภาพ

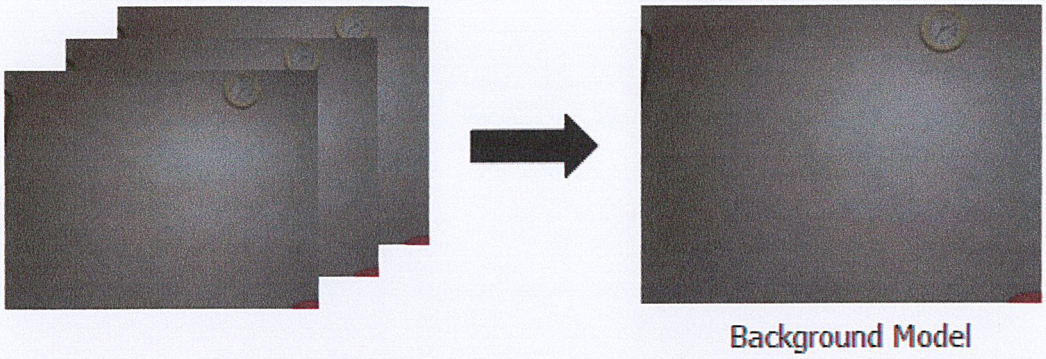
การตรวจจับและติดตามการเคลื่อนไหวของผู้เล่นจากรูปที่รับมาจากกล้องเว็บแคมโดยตรงเลยนั้นทำได้ยาก เพราะในรูปภาพที่รับเข้ามามีทั้งสิ่งที่เราสนใจและไม่สนใจปะปนกัน จึงต้องมีขั้นตอนในการแยกส่วนที่สนใจออกจากสิ่งที่ไม่สนใจเพื่อให้การวิเคราะห์ในขั้นตอนต่อไปสามารถทำได้ง่ายขึ้น ในโครงงานนี้ได้แบ่งลักษณะของสิ่งที่สนใจออกได้เป็น 2 รูปแบบ คือ โครงร่างร่างกายของผู้เล่น และส่วนที่มีการเคลื่อนไหว ซึ่งจะเลือกใช้วิธีใดนั้นจะพิจารณาจากลักษณะของเกมและการตรวจจับในเกมนั้นๆ

4.1.2.1 สนใจโครงร่างร่างกายของผู้เล่น (Silhouette)

การที่จะแยกส่วนของโครงร่างร่างกายของผู้เล่นจากภาพนั้น สามารถทำได้ด้วยวิธีการลบฉากหลัง โดยเริ่มต้นจากการสร้างแบบจำลองฉากหลังเป็นขั้นตอนแรกในกระบวนการลบฉากหลัง โดยนำเอารูปที่รับเข้ามา มาลบกับแบบจำลองฉากหลัง เพื่อให้เหลือแต่สิ่งที่สนใจ แล้วนำผลลัพธ์ที่ได้ไปใช้ในขั้นตอนต่อไป เพราะฉะนั้นจึงนับเป็นขั้นตอนหนึ่งที่มีความสำคัญมากเช่นเดียวกัน

การสร้างแบบจำลองฉากหลังคือการนำค่าสีในแต่ละจุดภาพมาเฉลี่ย เพื่อนำไปสร้างแบบจำลองฉากหลัง ในที่นี้ได้ใช้ภาพจำนวน 50 ภาพในการหาค่าเฉลี่ย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.2 การสร้างแบบจำลองฉากหลัง

หลังจากสร้างแบบจำลองฉากหลังแล้ว นำรูปที่รับเข้ามา มาทำการลบพื้นหลังเพื่อให้สิ่งที่สนใจเด่นชัดยิ่งขึ้น โดยขั้นตอนแรกคือการนำแบบจำลองฉากหลังและภาพปัจจุบันที่รับเข้ามาแปลงให้เป็นภาพระดับสีเทา ก่อนนำมาหาค่าสัมบูรณ์ของความแตกต่างระหว่างกันด้วยฟังก์ชัน cvAbsDiff ดังสมการ

$$dst_{i,j} = |I_{i,j} - BM_{i,j}| \quad (4.1)$$

โดยที่ $dst_{i,j}$ คือ พิกัดของจุดภาพหนึ่งของภาพผลลัพธ์

$I_{i,j}$ คือ พิกัดของจุดภาพหนึ่งของภาพปัจจุบันเฉดเทาที่รับเข้ามา

$BM_{i,j}$ คือ พิกัดของจุดภาพหนึ่งของภาพแบบจำลองฉากหลัง

รูปที่ 4.3 ผลลัพธ์ที่ได้จากการหาค่าสัมบูรณ์ของความแตกต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นำผลลัพธ์ที่ได้ไปมาแปลงให้เป็นภาพขาวดำด้วยฟังก์ชัน cvThreshold เพื่อให้ผลลัพธ์ที่ได้ชัดเจนมากยิ่งขึ้น โดยเลือก thresholdType เป็น CV_THRESH_BINARY สามารถอธิบายได้ตามสมการ

$$dst_{i,j}=\begin{cases}maxValue, & src_{i,j}>threshold \\ 0, & otherwise\end{cases} \qquad (4.2)$$

ถ้าค่าในพิกัดของภาพมากกว่าค่าเทรชโฮลให้พิกัดนั้นมีค่าเป็น maxValue ถ้ากำหนดให้ maxValue มีค่าเป็น 1 สิ่งที่สนใจจะเป็นสีขาว นอกจากนี้จะเป็นค่า 0

(ก) (ข)

รูปที่ 4.4 การแยกส่วนที่สนใจออกจากฉากหลัง

(ก) ภาพปัจจุบันที่นำเข้ามา

(ข) ผลลัพธ์ที่ได้จากการลบพื้นหลัง

จะสังเกตเห็นว่าภาพผลลัพธ์ที่ได้ยังมีสัญญาณรบกวนอยู่ซึ่งอาจจะส่งผลต่อการวิเคราะห์ประมวลผลมีโอกาสผิดพลาดได้ จึงต้องนำผลลัพธ์ที่ได้ไปเข้าสู่กระบวนการมอร์โฟโลยี ทั้งกระบวนการเปิด และกระบวนการปิด เพื่อกำจัดสัญญาณรบกวนให้น้อยลงโดยใช้ฟังก์ชัน cvMorphologyEx

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก) (ข)

รูปที่ 4.5 การกำจัดสัญญาณรบกวน

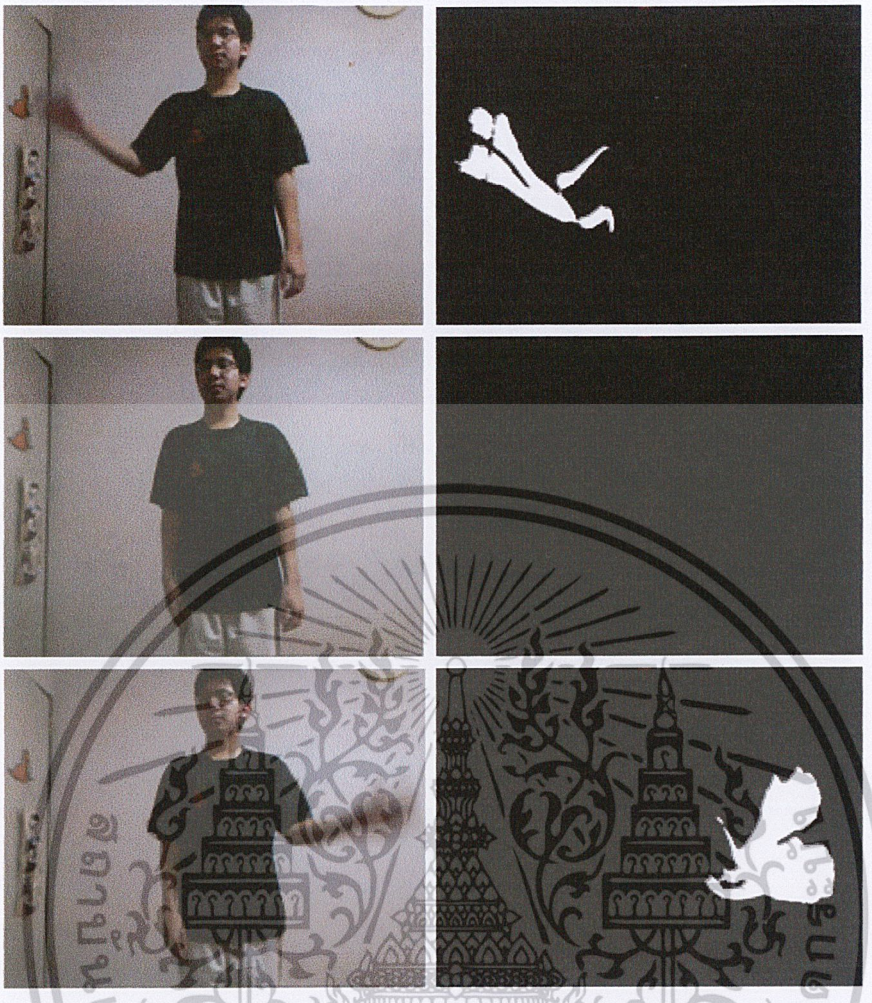
(ก) รูปที่ได้จากกระบวนการลบพื้นหลัง

(ข) ผลลัพธ์ที่ได้จากการกำจัดสัญญาณรบกวน

4.1.2.2 สนใจส่วนที่มีการเคลื่อนไหว

วิธีการหาความต่างระหว่างเฟรม เป็นวิธีหาความแตกต่างระหว่างเฟรมปัจจุบันกับเฟรมที่ผ่านมาก่อนหน้า ทำให้เห็นว่าพิกัดใดของรูปภาพที่มีการเปลี่ยนค่าไป นั่นคือมีการเคลื่อนไหวของผู้เล่น ณ บริเวณดังกล่าว การหาความต่างระหว่างเฟรมมีกระบวนการทำงานคล้ายกับวิธีการลบฉากหลัง ต่างกันตรงที่วิธีการลบฉากหลังต้องมีการทำแบบจำลองฉากหลังแล้วนำภาพปัจจุบันมาเปรียบเทียบ แต่วิธีการหาความต่างระหว่างเฟรมนั้นจะเปรียบเทียบเฉพาะเฟรมปัจจุบันกับเฟรมก่อนหน้าเท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.6 ผลลัพธ์การหาความแตกต่างระหว่างเฟรม

จากเทคนิคการประมวลผลภาพทั้งสองวิธีที่กล่าวมานั้น เมื่อพิจารณาถึงลักษณะของตัวเกมที่ได้ออกแบบไว้ สามารถสรุปได้ว่า เกมทดสอบความอดทนของกล้ามเนื้อ และเกมทดสอบการประสานสัมพันธ์ มีลักษณะของเกมที่เน้นการเคลื่อนไหวของร่างกายส่วนใดส่วนหนึ่งให้สัมผัสกับวัตถุ ดังนั้นจึงเหมาะที่จะใช้เทคนิคการประมวลผลภาพแบบหาค่าความต่างระหว่างเฟรม

ส่วนเกมทดสอบความยืดหยุ่น และเกมทดสอบความคล่องแคล่ว จะเน้นที่โครงร่างของผู้เล่นว่าได้ทำในท่าทางที่กำหนดหรือไม่ หรือหลบสิ่งของพ้นหรือไม่ ดังนั้นจึงเหมาะที่จะใช้วิธีการลบฉากหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2 การพัฒนาส่วนตัวเกม (Game Application)

4.2.1 การแปลงข้อมูลภาพจากกล้องเว็บแคมเพื่อแสดงผลใน XNA Game Studio Express

การแสดงผลภาพสองมิติ ใน XNA นั้น รูปแบบของภาพต้องเป็นข้อมูลประเภท Texture2D ก่อนจึงจะสามารถแสดงผลได้ แต่รูปที่รับเข้ามาจากกล้องเว็บแคมเป็นข้อมูลประเภท Image<Bgr, byte> ที่ไม่สามารถนำมาแสดงผลได้บน XNA ดังนั้นจึงจำเป็นต้องแปลงข้อมูลภาพจากเว็บแคมจาก Image<Bgr, byte> เป็น Texture2D โดยมีขั้นตอนดังนี้

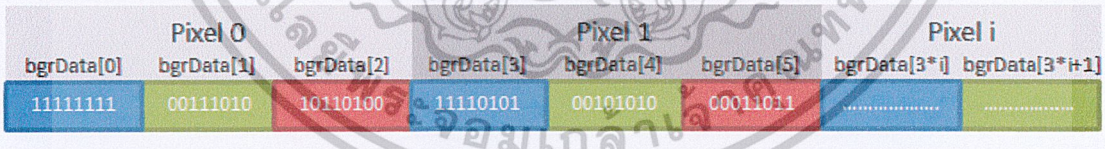
1. ประกาศตัวแปร 4 ตัว ได้แก่ bgrData เก็บข้อมูล byte ของภาพจากเว็บแคม, nextFrame เก็บข้อมูลของภาพจากเว็บแคม, colorData เก็บข้อมูลสีของรูปที่จะนำไปแสดงผล และ frame เก็บข้อมูลรูปที่จะแสดงผล

```
byte[] bgrData = new byte[1024 * 768];
Image<Bgr, byte> nextFrame;
Color[] colorData = new Color[1024 * 768];
Texture2D frame;
```

2. นำรูปที่ได้จากเว็บแคมจากคำสั่ง capture.QueryFrame() เก็บเข้าตัวแปร nextFrame และดึงเอาข้อมูล byte ของรูปภาพเก็บไว้ที่ bgrData

```
nextFrame = capture.QueryFrame();
bgrData = nextFrame.Bytes;
```

โดยแต่ละพิกเซลของรูปภาพจะเก็บข้อมูลสี 3 สี คือ น้ำเงิน เขียว และแดง



รูปที่ 4.7 แบบจำลองแสดงการเก็บข้อมูลสีของ XNA

3. นำข้อมูล byte ใน bgrData มาแทนเป็นข้อมูลประเภทสีใน colorData

```
for (int i = 0; i < colorData.Length; i++)
{
      colorData[i] = new
      Color(bgrData[3*i+2],bgrData[3*i+1],bgrData[3*i]);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. นำข้อมูลสีใน colorData มากำหนดสีให้กับ frame ที่เป็นข้อมูลประเภท Texture2D โดยใช้คำสั่ง SetData<Color>

```
frame.SetData<Color>(0, null, colorData, 0, colorData.Length,
SetDataOptions.None);
```

4.2.2 การแสดงผลกราฟฟิกสามมิติ

โมเดลกราฟฟิกสามมิติที่สามารถนำมาแสดงผลใน XNA ได้นั้นต้องมีนามสกุลต่างๆ เช่น .fbx, .x เป็นต้น ในการแสดงโมเดลกราฟฟิกสามมิติใน XNA มีพารามิเตอร์ที่สำคัญอยู่ 3 ตัวด้วยกัน คือ

1. World เก็บค่าการ Transform ทั้งหมดของโมเดล เช่น การปรับขนาด ตำแหน่งการวาง การหมุน เป็นต้น
2. View เก็บค่าตำแหน่งของกล้องภายในโลกสามมิติว่าอยู่ที่ตำแหน่งใด และหันมุมมองไปในทิศทางใด
3. Projection เก็บคุณสมบัติของกล้องว่ามีมุมกล้องเป็นอย่างไร อัตราส่วนของภาพ ระยะใกล้ไกล ที่มากที่สุดที่สามารถมองเห็นวัตถุบนโลกสามมิติคือเท่าใด เป็นต้น

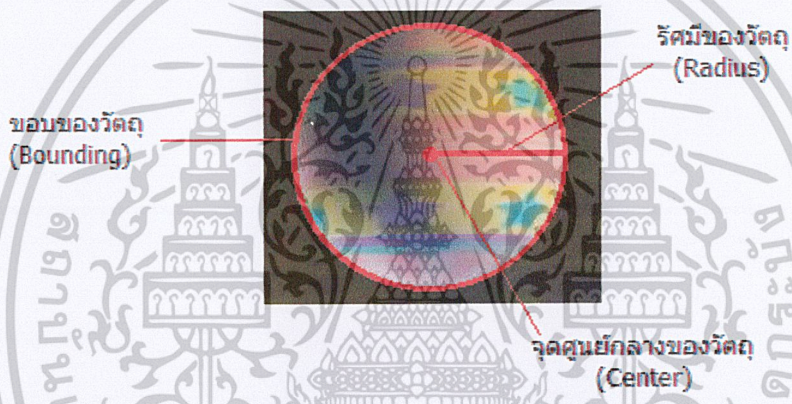
```
Model currentmodel;
...
currentmodel = Content.Load<Model>("Models\\dude");
...
foreach (ModelMesh mesh in currentmodel.Meshes)
{
     foreach (Effect effect in mesh.Effects)
     {
        Matrix world = Matrix.CreateScale(5.0f);
        Matrix view  = Matrix.CreateLookAt(cameraPosition,
                       Vector3.Zero, Vector3.Up);
        Matrix projection = Matrix.CreatePerspectiveFieldOfView(
                            MathHelper.ToRadians(45.0f),
                            aspectRatio, 1.0f, 10000.0f);
        // Setup camera
        effect.Parameters["Bones"].SetValue(bones);
        effect.Parameters["View"].SetValue(view);
        effect.Parameters["Projection"].SetValue(projection);
     }
  // Draw a model mesh
  mesh.Draw();
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.3 การตรวจสอบการชนของวัตถุกับผู้เล่น

การตรวจสอบการชนของวัตถุในเกมกับผู้เล่นไม่สามารถตรวจสอบได้โดยตรงเหมือนการตรวจสอบการชนของวัตถุผ่านในเกมทั่วๆไป เนื่องจากวัตถุภายในเกมเป็นสิ่งที่สมมติขึ้นอยู่ภายในเกม ส่วนผู้เล่นเคลื่อนไหวควบคุมอยู่ภายนอกเกม ทำให้การตรวจสอบการชนต้องใช้วิธีอื่น นั่นคือการนำภาพผลลัพธ์จากการประมวลผลที่ได้ มาวิเคราะห์และตรวจสอบการชน โดยมีขั้นตอนดังนี้

1. หาพิกัดจุดศูนย์กลางของวัตถุ และความยาวของรัศมีเพื่อหาบริเวณขอบของวัตถุ
   - สามารถหาพิกัดจุดศูนย์กลางของวัตถุได้จาก mesh.BoundingSphere.Center.X (พิกัด X) และ mesh.BoundingSphere.Center.Y (พิกัด Y)
   - หาความยาวรัศมีได้จาก mesh.BoundingSphere.Radius



รูปที่ 4.8 ส่วนประกอบของวัตถุสามมิติ

2. ตรวจสอบจำนวนพิกัดที่เป็นสีขาวภายในระหว่างบริเวณขอบของวัตถุกับจุดศูนย์กลาง

ถ้าในบริเวณดังกล่าวมีจำนวนพิกัดที่มีค่าเป็นสีขาว หรือในภาพผลลัพธ์บริเวณดังกล่าวมีสิ่งที่เราสนใจอยู่มากกว่าที่กำหนดไว้ ให้ถือว่าผู้เล่นได้ชนหรือสัมผัสกับวัตถุ โดยสามารถตรวจสอบพิกัดนั้นว่าอยู่ภายในรัศมีของวัตถุหรือไม่จากสูตรการหาระยะห่างระหว่างจุด 2 จุด

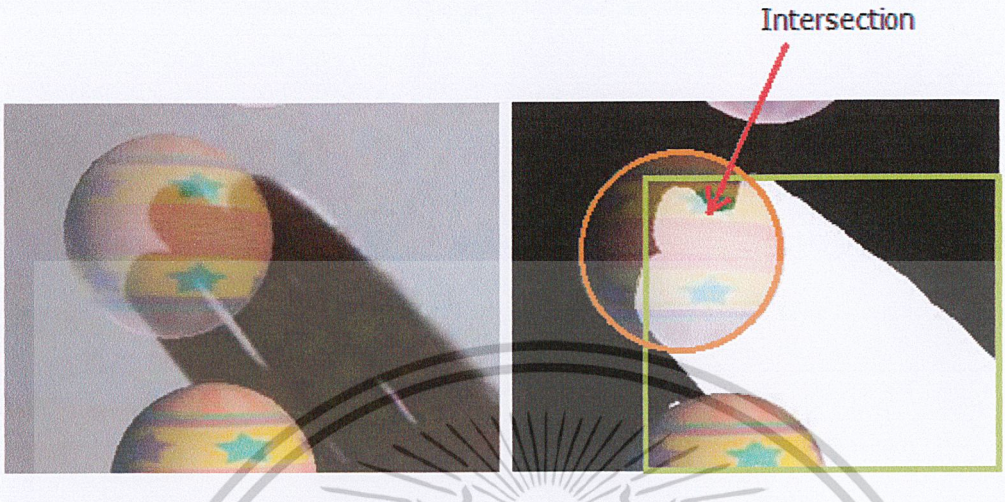
$$d=\sqrt{(x_2-x_1)^2+(y_2-y_1)^2} \qquad (4.3)$$

โดยที่ d คือ ระยะห่างระหว่างจุด

(x1,y1) และ (x2,y2) คือ พิกัดของจุดศูนย์กลางวัตถุและพิกัดที่สนใจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยขั้นตอนนี้จะตรวจสอบเฉพาะบริเวณวัตถุที่เราสนใจไม่จำเป็นต้องตรวจสอบทั้งภาพ เพื่อลดเวลาในการทำงานลง



(ก) (ข)

รูปที่ 4.9 การตรวจสอบการชน

(ก) รูปที่แสดงผลออกมาปกติ

(ข) รูปที่ใช้ในการวิเคราะห์

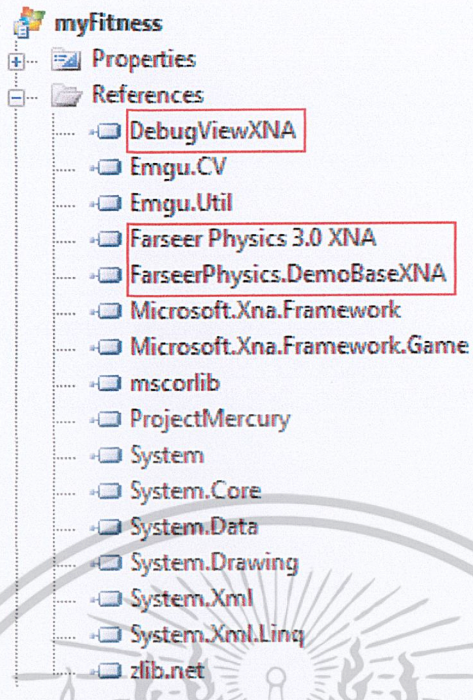
ในภาพ (ข) กรอบวงกลมสีส้ม คือบริเวณขอบของวัตถุสามมิติ กรอบสี่เหลี่ยมสีเขียว คือมือของผู้เล่น บริเวณที่ทั้ง 2 กรอบมีการทับซ้อนกัน (Intersection) คือบริเวณที่จะนำมานับพิกัดที่เป็นสีขาว

4.2.4 การเขียนโปรแกรมเพื่อใช้งานระบบฟิสิกส์ FarseerPhysics Engine

การนำระบบฟิสิกส์มาใช้ในตัวเกม ทำให้เกมมีความสมจริงมากขึ้น ผู้เล่นจะรู้สึกเหมือนตัวเองได้เป็นส่วนหนึ่งในเกม และทำให้เกมมีความน่าสนใจมากยิ่งขึ้นด้วย สำหรับโครงงานนี้ได้นำระบบฟิสิกส์ Farseer Physics Engine มาใช้กับโครงงาน เนื่องจากเป็นระบบฟิสิกส์ที่สามารถนำมาใช้ร่วมกับ XNA Game Studio Express ได้ ใช้งานง่าย และสามารถดาวน์โหลดมาใช้ได้โดยไม่เสียค่าใช้จ่าย

โดยก่อนการใช้งานต้องนำ DebugViewXNA.dll, Farseer Physics 3.0 XNA.dll และ FarseerPhysics.DemoBaseXNA.dll เพิ่มเข้าไปในโฟลเดอร์ References ของโปรเจคที่สร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.10 การเพิ่มไลบรารีFarseer Physic Engine ในโปรเจค

และเรียกใช้งาน Namespace ของ Farseer Physics Engine เพื่อให้สามารถใช้งานได้

```
using FarseerPhysics.Common;
using FarseerPhysics.Collision;
using FarseerPhysics.Collision.Shapes;
using FarseerPhysics.DemoBaseXNA;
using FarseerPhysics.DemoBaseXNA.DemoShare;
using FarseerPhysics.DemoBaseXNA.ScreenSystem;
using FarseerPhysics.Dynamics;
using FarseerPhysics.Factories;
using FarseerPhysics.DebugViewXNA;
```

ในระบบฟิสิกส์ของ Farseer Physics Engine ประกอบด้วยวัตถุ 4 ชนิด คือ

1) World โลกเสมือนของระบบฟิสิกส์ ควบคุมวัตถุทุกอย่างในโลกระบบฟิสิกส์ ด้วยคำสั่ง Step() และ ควบคุมให้วัตถุในระบบฟิสิกส์สอดคล้องและมีความเสถียร
2) Body เป็นวัตถุที่อยู่ใน World เป็นวัตถุพื้นฐานในระบบฟิสิกส์ที่มีผลต่อแรงต่างๆในระบบ เช่น แรงจากการชน หรือ แรงโน้มถ่วง
3) Shape เป็นวัตถุที่มีลักษณะเสมือนเป็นรูปทรงในโลก สองมิติ มวลและแรงเฉื่อยจะถูกคำนวณจาก Shape
4) Fixture เป็นวัตถุที่มี Shape ร่วมอยู่ด้วย จุดศูนย์กลางของ Shape จะเป็นตำแหน่งของ Fixture ด้วย เมื่อมีแรงกระทำกับ Shape แรงจะถูกคำนวณและส่งผลต่อ Fixture ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
World _world = new World(new Vector2(0, -20));

Fixture _ballFixture = FixtureFactory.CreateCircle
                (_world, 5, 1, new Vector2(-16, 15));
                _ballFixture.Body.BodyType = BodyType.Dynamic;
                _ballFixture.Body.Mass = 1;
                _ballFixture.Restitution = 0.3f;
                _ballFixture.Friction = 0.5f;
```

สร้าง _world เป็นวัตถุประเภท World โดยกำหนดให้มีแรงโน้มถ่วงทางทิศล่าง -20 หน่วย และสร้าง Fixture ชื่อ _ballFixtureมีรูปร่างเป็นทรงกลม ให้มีมวล 1 หน่วย แรงหลังการกระทบ 0.3 หน่วย แรงเสียดทาน 0.5 หน่วย ใน _world

```
_world.Step((float)gameTime.ElapsedGameTime.TotalMilliseconds *
0.001f);
```

ใน Update ของ XNA ใส่คำสั่ง Step() ไว้ด้วยเพื่อให้ระบบฟิสิกส์ทำงาน และมีการ Update ตำแหน่งตลอดเวลา

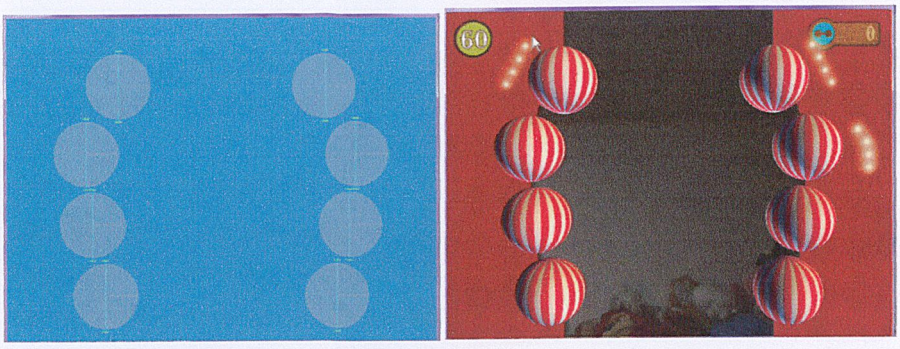
4.2.5 การแปลงพิกัดระบบสองมิติเป็นสามมิติ

เนื่องจากระบบฟิสิกส์ของ Farseer Physics Engine เหมาะสำหรับการใช้กับงานที่มีลักษณะเป็นสองมิติ แต่สำหรับโครงงานนี้ต้องนำระบบฟิสิกส์มาใช้ร่วมกับโมเดลสามมิติ ซึ่งอยู่บนระบบพิกัดสามมิติที่มีพิกัดแตกต่างกับระบบพิกัดสองมิติ จึงจำเป็นต้องแปลงพิกัดระบบสองมิติที่ได้จากระบบฟิสิกส์เป็นพิกัดระบบสามมิติ เพื่อให้พิกัดในการแสดงผลแก่โมเดลสามมิติซึ่งต้องมีการแปลงพิกัดก่อนการวาดโมเดล

```
Matrix[] transforms = new Matrix[m.get3DModel().Bones.Count];
m.get3DModel().CopyAbsoluteBoneTransformsTo(transforms);

effect.World = Matrix.CreateFromYawPitchRoll(m.getRotation().X,
m.getRotation().Y, m.getRotation().Z) *
                Matrix.CreateScale(m.getModelScale()) *
                Matrix.CreateTranslation(m.getModelPosition() * 82)
                * transforms[mesh.ParentBone.Index];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



(ก) (ข)

รูปที่ 4.11 การแปลงพิกัดระบบสองมิติเป็นสามมิติ

(ก) มุมมองวัตถุในโลกของฟิสิกส์ที่เป็นสองมิติ

(ข) มุมมองที่เห็นจริงในโลกของสามมิติ

4.2.6 การแปลงพิกัดระบบสามมิติเป็นสองมิติ

ในการเล่นของผู้เล่นบางเกมผู้เล่นจะต้องสัมผัสกับโมเดลสามมิติในการเล่น เพราะฉะนั้นระบบต้องรู้พิกัดของโมเดลที่เป็นระบบพิกัดสามมิติว่าตรงกับพิกัดผลลัพธ์จากการประมวลผลรูปรูปที่เป็นระบบพิกัดสองมิติหรือไม่ (ผลลัพธ์คือจุดภาพสีขาวที่ได้จากการหาความต่างระหว่างเฟรมหรือการลบฉากหลัง) โดยใช้คำสั่ง graphics.GraphicsDevice.Viewport.Project()

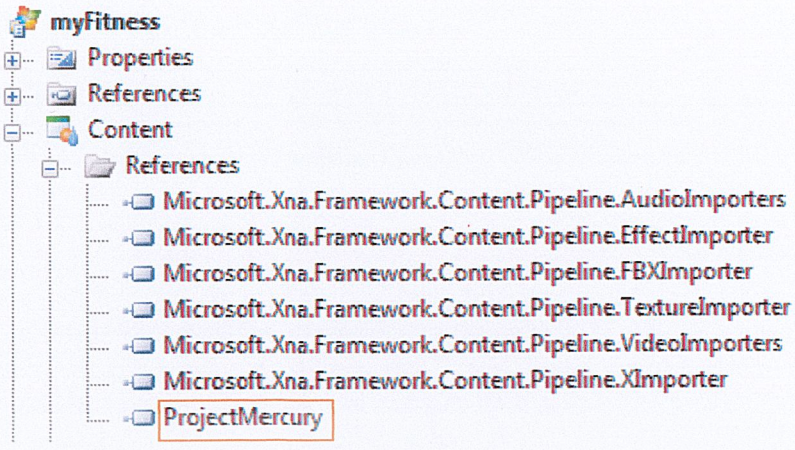
```
Vector3 coord3d2 = graphics.GraphicsDevice.Viewport.Project(
                    new Vector3(mesh.BoundingSphere.Center.X,
                    mesh.BoundingSphere.Center.Y,
                    mesh.BoundingSphere.Center.Z),
                    effect.Projection, effect.View, effect.World);
```

4.2.7 การเขียนโปรแกรมเพื่อใช้งาน Particle Effect ด้วย Mercury Particle Engine

เอฟเฟกต์เป็นสิ่งที่สำคัญสิ่งหนึ่งในการพัฒนาเกม เพราะว่าเป็นตัวช่วยให้เกมมีความสมจริง และเป็นสิ่งที่ช่วยดึงดูดใจผู้เล่นอีกด้วย โดยเอฟเฟกต์ที่ใช้ในการพัฒนาเกมบน XNA Game Studio Express จะเรียกว่า Particle Effect ในโครงงานนี้ได้นำ Mercury Particle Engine ซึ่งเป็นไลบรารีที่ช่วยในการสร้างและแสดงผลเอฟเฟกต์ใน XNA Game Studio Express ได้เป็นอย่างดี และสามารถดาวน์โหลดมาใช้งานได้โดยไม่เสียค่าใช้จ่าย

ก่อนการใช้งานต้องนำ ProjectMercury.dll เพิ่มเข้าไปในโฟลเดอร์ References ของโปรเจคที่สร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.12 การเพิ่มไลบรารี Mercury Particle Engine ในโปรเจค

และเรียกใช้งาน Namespace ของ Mercury Particle Engine เพื่อให้สามารถใช้งานได้

```
using ProjectMercury;
using ProjectMercury.Emitters;
using ProjectMercury.Modifiers;
using ProjectMercury.Renderers;
```

Mercury Particle Engine มีโครงสร้างที่สำคัญ 3 ชนิดคือ

1) Particle Effect เป็นโครงสร้างพื้นฐานของ Mercury เป็นส่วนที่จะเก็บ Particle Emitters ไว้ และเป็นส่วนติดต่อระหว่าง Mercury กับเกม
2) Particle Emitters เป็นส่วนงานหลักของทุกๆ Particle Effect มีหน้าที่อัพเดท และ สร้าง กำหนดคุณลักษณะของ Particle ใหม่ เช่น สี หรือทิศทาง
3) Modifiers จะถูกใช้งานเมื่อ Particle มีการใช้งานจะทำงานในส่วนของการ Fading การกำหนดแรงโน้มถ่วง เป็นต้น

Renderers เป็นส่วนที่ทำงานในด้านการแสดงผลของ Particle Effect

```
Renderer myRenderer;
ParticleEffect myEffect;
```

ประกาศตัวแปร myRenderer เป็นประเภท Renderer และ myEffect เป็นประเภท ParticleEffect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
protected override void LoadContent()
{
    ...
    myEffect = Content.Load<ParticleEffect>("Effect\\mouseEff");
    myEffect.LoadContent(Content); myEffect.Initialise();
    ...
    myRenderer.LoadContent(Content);
}
```

ใน Method LoadContent() โหลด Effect ที่ได้สร้างขึ้นจาก Mercury Particle Engine EffectEditor (.xml) จากใน Content ของโปรเจค

```
protected override void Update(GameTime gameTime)
{
    ...
    float SecondsPassed =
                    (float)gameTime.ElapsedGameTime.TotalSeconds;
    myEffect.Update(SecondsPassed);
    ...
}

protected override void Draw(GameTime gameTime)
{
    ...
    myRenderer.RenderEffect(myEffect);
    ...
}
```

ใน Method Update() ให้อัพเดทตำแหน่ง การเคลื่อนไหว ของ Particle Effect โดยใช้คำสั่ง myEffect.Update(); และใน Method Draw() ให้วาด Particle Effect ลงใน myRenderer

```
myEffect.Trigger(new Vector2(900, 230));
```

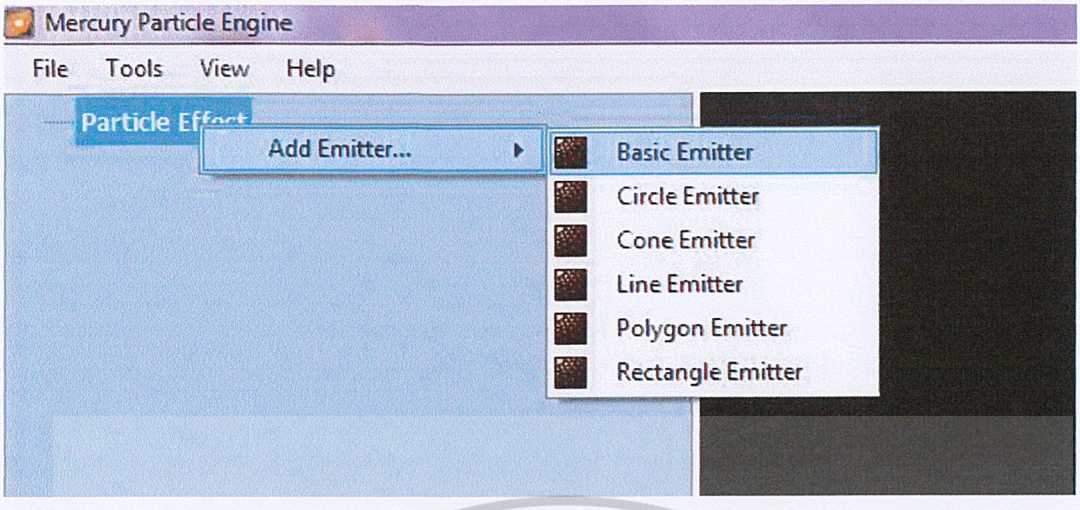
ใช้คำสั่ง Trigger ในการเล่น Particle Effect โดยระบุตำแหน่งที่ต้องการให้แสดงผลด้วย

4.2.7.1 การสร้าง Particle Effect ด้วย Mercury Particle Engine EffectEditor

ผู้ใช้งาน Mercury Particle Engine สามารถนำเอฟเฟกต์พื้นฐานไปใช้งานได้ อีกทั้งยังสามารถสร้างเอฟเฟกต์มาใช้งานได้เองตามต้องการ เพื่อให้สอดคล้องกับลักษณะของเกม โดยสามารถสร้างได้จากโปรแกรม Mercury Particle Engine EffectEditor สามารถสร้างได้ง่าย และมีรูปแบบที่หลากหลาย

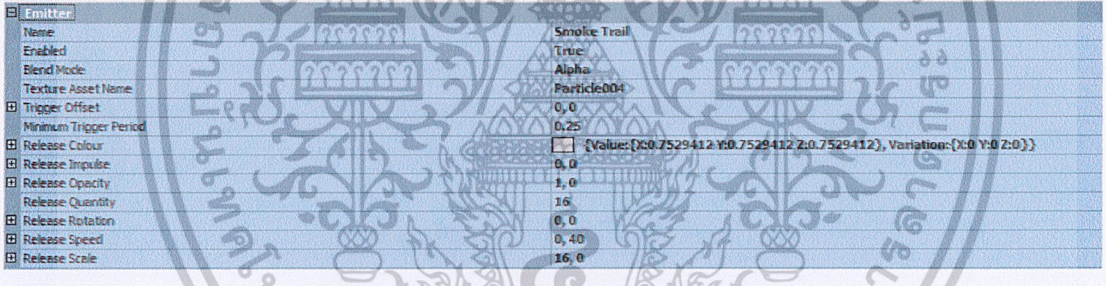
โดยสำหรับตัวอย่างนี้จะแสดงวิธีการสร้าง Particle Effect ควัน โดยเริ่มต้นจากการเปิดโปรแกรม Mercury Particle Engine EffectEditor คลิกขวาที่ Particle Effect เลือก Add Emitter... เลือก Basic Emitter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 การเริ่มสร้างเอฟเฟกต์แบบ Basic Emitter

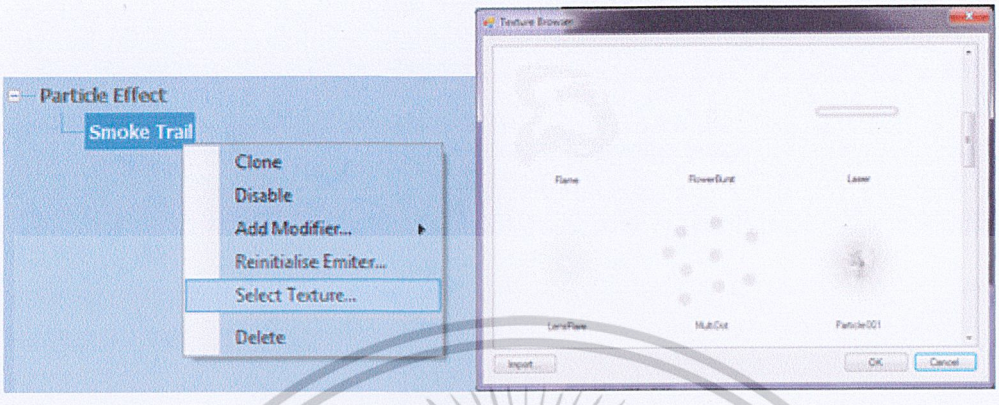
กำหนดค่าต่างๆให้กับ Particle Effect เพื่อให้แสดงผลตามที่ต้องการเช่น Minimum Trigger Period กำหนดช่วงเวลาที่จะเล่นเอฟเฟกต์ต่อรอบ Release Colour กำหนดสีของเอฟเฟกต์ หรือ Release Quantity กำหนดปริมาณของเอฟเฟกต์ที่จะเล่นต่อรอบ เป็นต้น



รูปที่ 4.14 การกำหนดค่าต่างๆให้กับเอฟเฟกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถเลือก Texture ให้กับ Particle Effect ได้เองโดยคลิกขวาที่ Particle Effect ที่สร้างแล้วเลือก Select Texture…



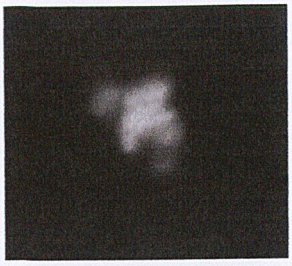
รูปที่ 4.15 การเลือก Texture ให้เอฟเฟกต์

สามารถกำหนด Modifiers เพิ่มเติมให้กับ Particle Effect โดยการคลิกขวาแล้วเลือก Add Modifiers…

รูปที่ 4.16 การกำหนด Modifiers ให้เอฟเฟกต์

- OpacityModifier เป็นตัวกำหนดระดับการมองเห็นของ Particle Effect ในตอนเริ่มและจบ
- ScaleModifier เป็นตัวกำหนดขนาดของ Particle Effect ในตอนเริ่มและจบ
- RotationModifier เป็นตัวกำหนดการหมุนของ Particle Effect

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

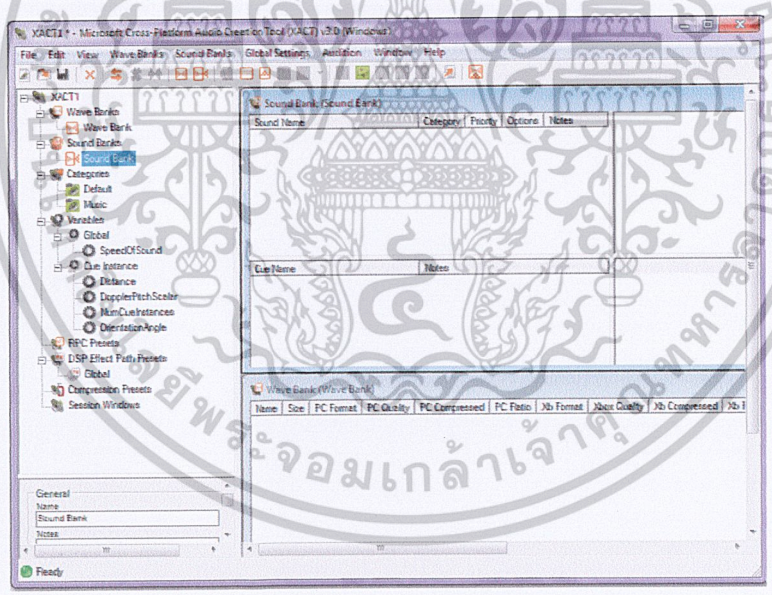


รูปที่ 4.17 ผลลัพธ์เอฟเฟกต์ที่สร้างขึ้นเอง

4.2.8 การเขียนโปรแกรมเพื่อจัดการไฟล์เสียงโดยใช้ XACT

การจะนำไฟล์เสียงมาใช้ต้องแปลงไฟล์เสียงให้อยู่ในรูปแบบ Wave (.WAV) ก่อนที่จะนำเข้าสู่ XACT เพื่อนำเข้ามาใช้ในเกมต่อไป โปรแกรม XACT นั้นจะติดตั้งมาพร้อมกับการติดตั้ง XNA Game Studio Express

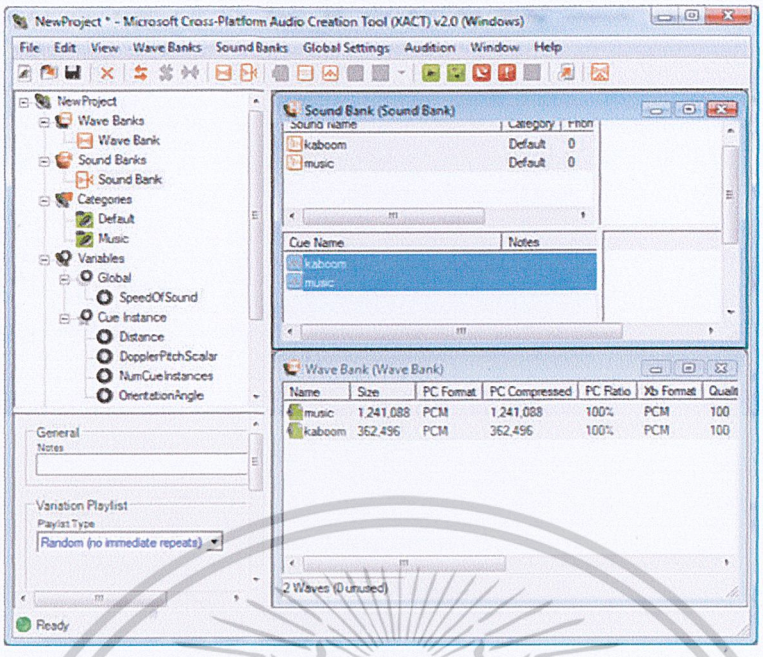
เมื่อเปิดโปรแกรมแล้วให้ทำการสร้างโปรเจคและกำหนดตำแหน่งของโปรเจค หลังจากนั้นให้สร้าง Wave Banks และ Sound Banks



รูปที่ 4.18 เริ่มการสร้างไฟล์เสียง

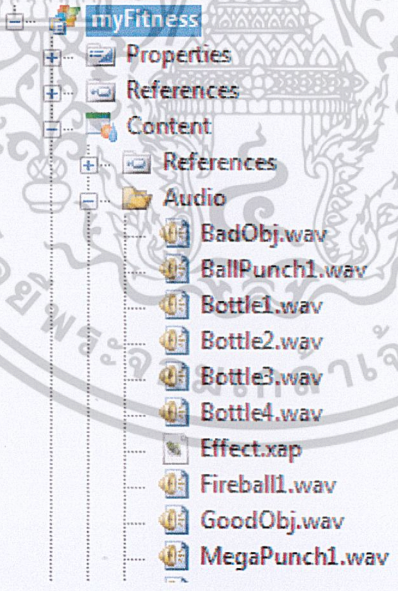
คลิกขวาที่ Wave Banks เลือก Insert Wave File(s) หลังจากนั้นลากไฟล์เสียงจากหน้าต่าง Wave Bank ไปยังหน้าต่าง Sound Bank บริเวณ Cue Name แล้ว Save Project

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.19 การลากไฟล์ Wave Bank ไปยัง Sound Bank

Add XACT Project และไฟล์เสียง เข้าสู่เกม



รูปที่ 4.20 การเพิ่ม XACT ในโปรเจค

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประกาศตัวแปรของ AudioEngine,SoundBank และ WaveBank หลังจากนั้นโหลดไฟเสียงด้วยคำสั่ง soundBank.GetCue() แล้ว เล่นเสียงด้วยคำสั่ง Play()

```
private AudioEngine engine;
private SoundBank soundBank;
private WaveBank waveBank;
...
engine = new AudioEngine("Content\\Audio\\Effect.xgs");
soundBank = new SoundBank(engine, "Content\\Audio\\Sound
Bank.xsb");
waveBank = new WaveBank(engine, "Content\\Audio\\Wave Bank.xwb");
...
protected override void Update(GameTime gameTime)
{
   ...
   engine.Update();
   ...
}

Cue bottleSound4 = soundBank.GetCue("Bottle3");
bottleSound4.Play();
```

4.2.9 รูปแบบเกม

เป็นการวัดสมรรถภาพของร่างกาย 4 ด้านประกอบด้วย ความอดทนของกล้ามเนื้อการประสานสัมพันธ์ความยืดหยุ่น และความคล่องแคล่ว สมรรถภาพแต่ละด้านจะทดสอบด้วยเกม 1 เกม แต่ละเกมจะมีเวลาเกมละ 1 นาที คะแนนเต็มเกมละ 1000 คะแนน รวม 4000 คะแนน แล้วนำคะแนนสุดท้ายมาประเมินผล

4.2.9.1 เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ (Muscular Endurance Game)

รูปแบบเกม จะมีลูกบอลอยู่ทั้ง 2 ฝั่งของผู้เล่นฝั่งละ 4 ลูก ผู้เล่นจะต้องใช้มือหรือขาในการชกหรือเตะลูกบอลที่มีไฟขึ้นอยู่จนกระทั่งไฟจะดับลงทุกดวง

เกมนี้ผู้เล่นจะได้ออกแรงที่แขนและขาเพื่อชกและเตะลูกบอล เป็นการทดสอบความอดทนกล้ามเนื้อที่ต้องออกแรงชกและเตะบอลเป็นเวลานาน

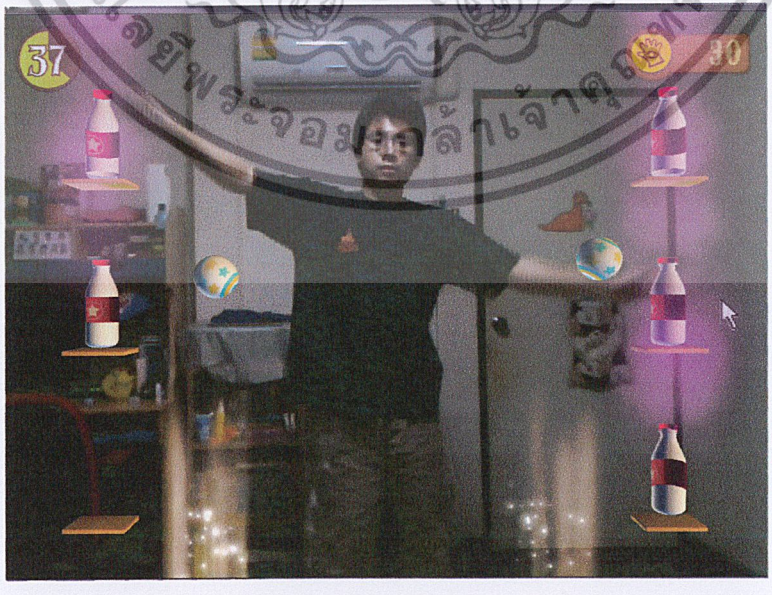
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.21 เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ

4.2.9.2 เกมทดสอบสมรรถภาพการประสานสัมพันธ์ (Co-ordination Game)

รูปแบบเกม จะมีขวดปรากฏขึ้นทั้ง 2 ฝั่งของผู้เล่น ฝั่งละไม่เกิน 3 ขวด ผู้เล่นจะต้องชกลูกบอลที่ลอยขึ้นมาเพื่อให้บอลโดนขวดจึงจะได้คะแนน

เกมนี้ผู้เล่นจะได้ทดสอบความสามารถในการมองเห็นและควบคุมการเคลื่อนไหวของมือให้สอดคล้องกับสิ่งที่เห็น คือ เมื่อเห็นลูกบอลลอยอยู่ในตำแหน่งที่ต้องการ ผู้เล่นจะต้องควบคุมมือชกลูกบอลในตำแหน่งนั้นอย่างถูกต้อง



รูปที่ 4.22เกมทดสอบสมรรถภาพการประสานสัมพันธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.2.9.3 เกมทดสอบสมรรถภาพความยืดหยุ่น (Flexibility Game)

รูปแบบเกม จะมีแผ่นกระดานที่เจาะช่องว่างเป็นรูปร่างต่างๆ ผู้เล่นต้องจัดระเบียบร่างกายให้สามารถผ่านช่องว่างของกระดานนั้น จึงจะได้คะแนน

เกมนี้ผู้เล่นจะต้องจัดระเบียบร่างกาย ให้ตรงกับรูปร่างที่เห็น ซึ่งจะเป็นรูปร่างในลักษณะที่ต้องการความยืดหยุ่น ถ้าผู้เล่นมีความยืดหยุ่นน้อยจะไม่สามารถผ่านไปได้

รูปที่ 4.23 เกมทดสอบสมรรถภาพความยืดหยุ่น

4.2.9.4 เกมทดสอบสมรรถภาพความคล่องแคล่ว (Agility Game)

รูปแบบเกม จะมีวัตถุลอยออกมาจากทุกทิศทางแบ่งออกเป็น วัตถุที่ต้องหลบ วัตถุที่ต้องเก็บ ซึ่งผู้เล่นต้องหลบวัตถุที่ต้องหลบ และต้องเก็บวัตถุที่ต้องเก็บจึงจะได้คะแนน

เกมนี้ผู้เล่นจะต้องมีความคล่องแคล่วในการหลบหลีกสิ่งของ ถ้าผู้เล่นมีความคล่องแคล่วจะสามารถหลบสิ่งของเหล่านั้นได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.24 เกมทดสอบสมรรถภาพความคล่องแคล่ว

4.3 การพัฒนาส่วนติดต่อกับผู้ใช้ (Interface)

จากที่ได้ออกแบบไว้ในบทที่ 3 (หัวข้อที่ 3.5.5 การออกแบบส่วนติดต่อกับผู้ใช้) ส่วนติดต่อกับผู้ใช้จะมี 2 แบบด้วยกันคือ ควบคุมผ่านเมาส์และคีย์บอร์ด และควบคุมด้วยการสัมผัส

4.3.1 ส่วนติดต่อกับผู้ใช้ที่ควบคุมผ่านเมาส์และคีย์บอร์ด

ส่วนนี้จะเน้นที่การแสดงข้อมูลต่างๆให้กับผู้ใช้ จึงทำเป็นภาพกราฟฟิกสองมิติเพื่อให้การแสดงผลมีความชัดเจน และสามารถควบคุมผ่านเมาส์และคีย์บอร์ดได้ง่ายขึ้นและจะใส่สัญลักษณ์รูปเมาส์ไว้ที่มุมขวาล่างของจอเพื่อบอกให้ผู้ใช้ทราบ

4.3.1.1 การสร้างภาพกราฟฟิกสองมิติ (2D Graphics)

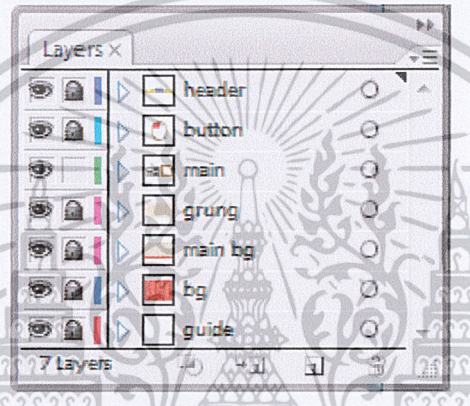
การสร้างภาพกราฟฟิกสองมิติเพื่อนำมาใช้ในการทำฉากประกอบ เมนู และไอคอนต่างๆ โดยสร้างด้วยโปรแกรม Illustrator CS3 และปรับแต่งรูปภาพด้วยโปรแกรม Photoshop CS3 ซึ่งทั้ง 2 โปรแกรมนี้เป็นโปรแกรมสำหรับการทำงานเกี่ยวกับภาพสองมิติโดยเฉพาะ อีกทั้งเป็นโปรแกรมที่ใช้งานง่าย และมีฟังก์ชันในการทำงานครบครัน โดยมีขั้นตอนการพัฒนาดังนี้

1. ขั้นตอนการสร้างไฟล์ใหม่ และเตรียมแม่แบบ
    - ทำการสร้างไฟล์ใหม่ในโปรแกรม Illustrator CS3 แล้วกำหนดขนาดของงานตามต้องการในหน่วยพิกเซล
    - กำหนดขอบเขตของงานโดยใช้เครื่องมือ Crop Area Tool (Ctrl+O)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ทำการสแกนรูปที่ได้ออกแบบไว้ แล้วใช้คำสั่ง Place Template เพื่อ import รูปที่ออกแบบไว้มาใช้เป็นแม่แบบในการสร้างผลงาน
- ทำการ Lock layer ที่เป็นแม่แบบ เพื่อไม่ให้ถูกเคลื่อนย้ายโดยไม่ตั้งใจ

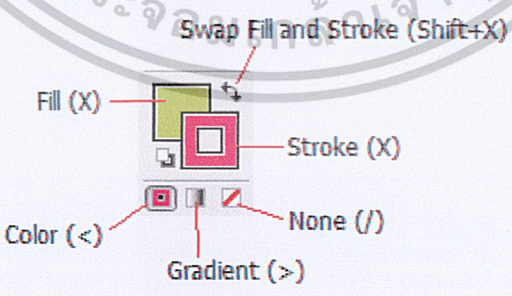
2. ขั้นตอนการวาดตามแม่แบบ
   - สร้าง layer ใหม่
   - ใช้เครื่องมือ Pen Tool (P) หรือ Pencil Tool (N) วาดตามแม่แบบ
   - หากรูปที่วาดมีรายละเอียดเยอะ ควรวาดแต่ละ Object แยก layer กัน เพื่อความสะดวกในการทำงาน



รูปที่ 4.25 การแยก Layer ของ Object

   - สามารถใช้คำสั่ง Group (Ctrl+G) ในการรวม Object ให้เป็น Object เดียวกัน

3. ขั้นตอนการลงสี
   - เลือก object ที่ต้องการลงสี แล้วเลือกสีจาก Fill และ Stroke ใน Tools Box ถ้าหากไม่ต้องการให้มีสีใน Fill หรือ Stroke ให้เลือก None(/)



รูปที่ 4.26 การเลือกสี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

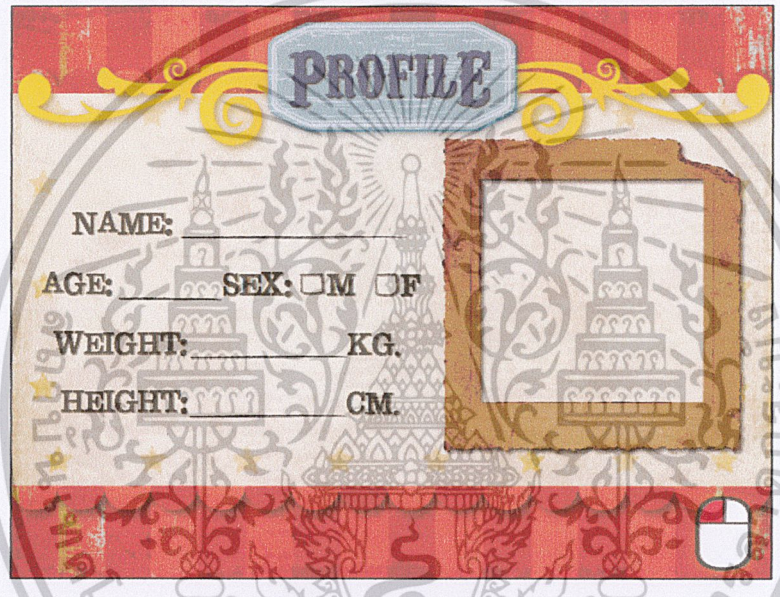
4. ขั้นตอนการตกแต่งภาพ สามารถทำได้ 2 วิธีคือ

    1) save file แล้วนำไปเปิดในโปรแกรม Photoshop CS3

    2) copy ภาพทั้งหมด แล้วนำมา paste ที่โปรแกรม Photoshop CS3 โดยเลือก option การ paste เป็นแบบ Smart Object เพื่อคงลักษณะภาพแบบเวกเตอร์

5. ขั้นตอนการบันทึกไฟล์

    - เลือก save ประเภทของไฟล์เป็น .png เนื่องจากไฟล์รูปภาพประเภท png จะมีคุณภาพสูง และสามารถกำหนดให้พื้นหลังเป็นแบบโปร่งใส (Transparent) ได้

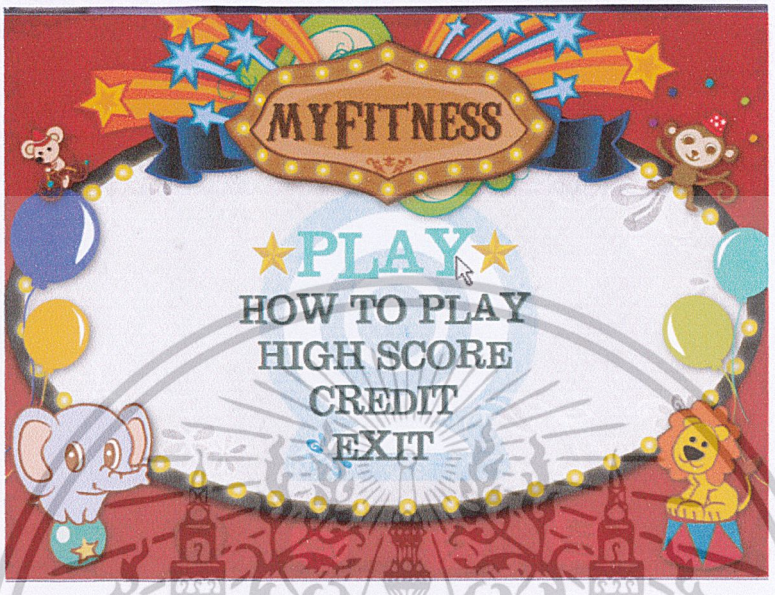


รูปที่ 4.27 ฉากของเกมที่เสร็จสมบูรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

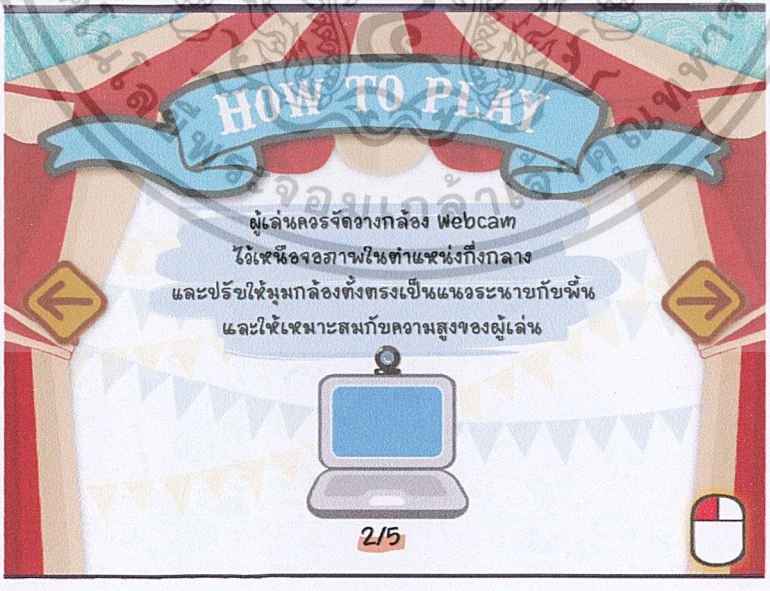
4.3.1.2 หน้าจอของส่วนติดต่อกับผู้ใช้ที่ควบคุมด้วยเมาส์และคีย์บอร์ด

1) หน้า Main Menu เป็นหน้าแรกของเกม แสดงเมนูต่างๆ ได้แก่ Play, How To Play, High Score Credit และ Exit



รูปที่ 4.28 หน้า Main Menu

2) หน้า How To Play เปรียบเสมือนคู่มือการเล่นเกม ที่แสดงรายละเอียดวิธีการเล่น การตั้งกล้อง การจัดสถานที่สำหรับเล่นเกม เป็นต้น



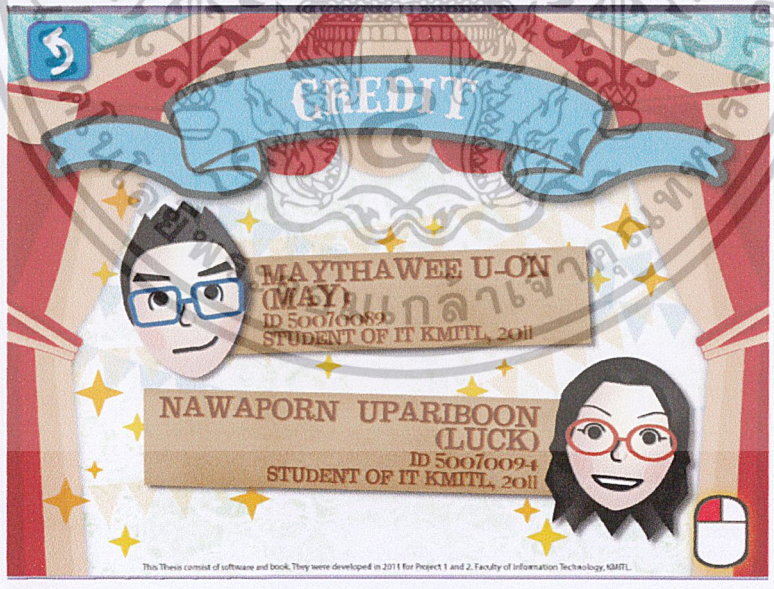
รูปที่ 4.29 หน้า How To Play

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) หน้า Leader Board แสดงอันดับของผู้เล่นในแต่ละเกมที่ทำคะแนนได้สูงสุด 3 อันดับแรก

รูปที่ 4.30 หน้า Leader Board

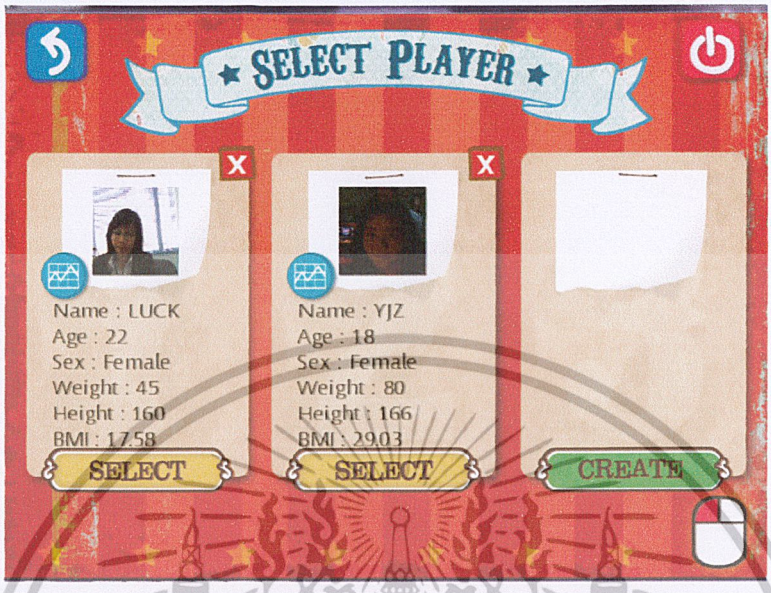
4) หน้า Credit แสดงรายชื่อผู้จัดทำเกม



รูปที่ 4.31 หน้า Credit

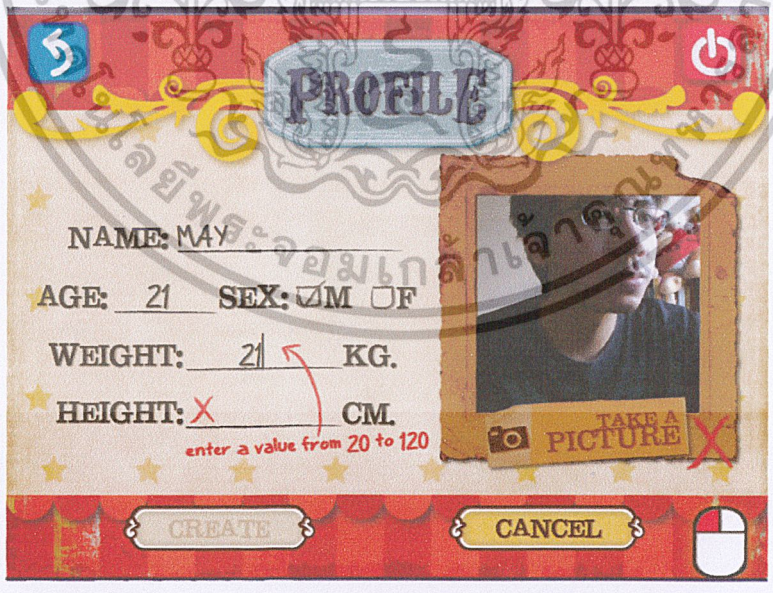
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5) หน้า Select Profile หลังจากเลือกเมนู Play ที่หน้า Main Menu จะเข้าสู่หน้าเลือกผู้เล่น ซึ่งมีให้เลือกทั้งหมด 3 ผู้เล่น



รูปที่ 4.32 หน้า Select Profile

6) หน้า Create New Profile เป็นหน้าสำหรับสร้างข้อมูลผู้เล่นใหม่ เมื่อเลือก Create ที่หน้า Select Profile



รูปที่ 4.33 หน้า Create New Profile

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7) หน้า BMI Detail แสดงความหมายของดัชนีมวลกาย (Body Mass Index) ก่อนเข้าสู่การคำนวณผลลัพธ์

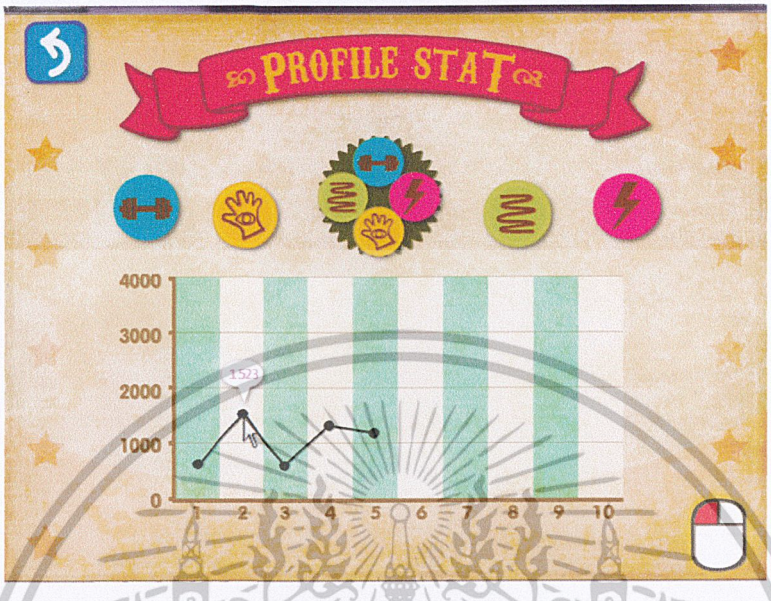
รูปที่ 4.34 หน้า BMI Detail

8) หน้า BMI Calculate แสดงผลลัพธ์ค่าดัชนีมวลกายของผู้เล่น หลังจากสร้างข้อมูลผู้เล่นแล้ว

รูปที่ 4.35 หน้า BMI Calculate

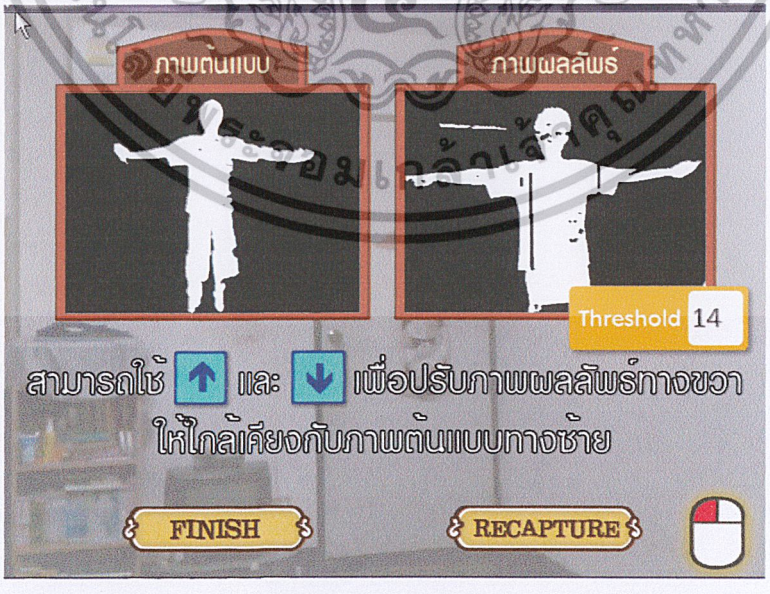
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9) หน้า Profile Stat แสดงข้อมูลประวัติการเล่นของผู้เล่นในแต่ละเกม โดยจะเก็บประวัติไว้ทั้งหมด 10 ครั้งล่าสุดที่เล่น



รูปที่ 4.36 หน้า Profile Stat

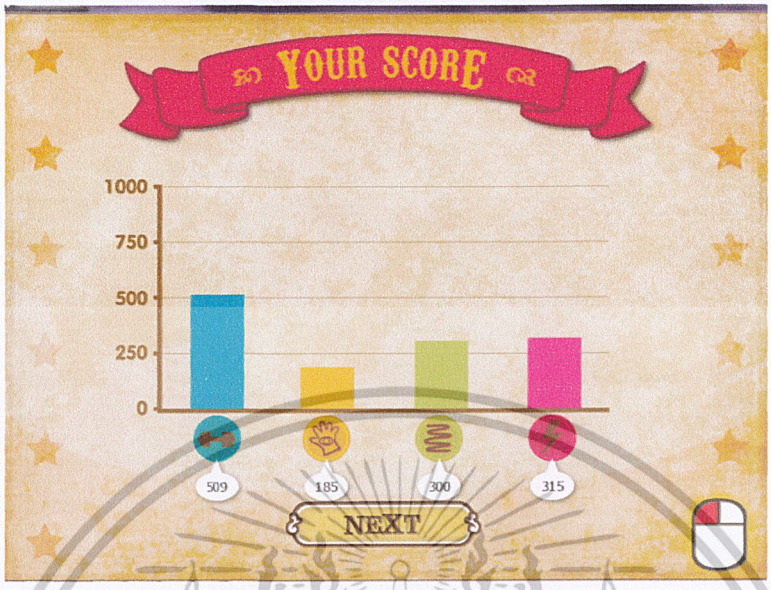
10) หน้า Background Subtraction Config เป็นหน้าของการเก็บภาพพื้นหลังก่อนการเริ่มเล่นเกม จะแสดงหน้านี้ก็ต่อเมื่อจะเริ่มเล่นเกมทดสอบสมรรถภาพความยืดหยุ่น และเกมทดสอบสมรรถภาพความคล่องแคล่ว โดยสามารถปรับค่าเทรชโฮลได้ตามต้องการ



รูปที่ 4.37 หน้า Background Subtraction Config

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11) หน้า Graph Result แสดงคะแนนหลังจากเล่นเกมจบในรูปแบบกราฟแท่ง



รูปที่ 4.38 หน้า Graph Result

12) หน้า Congratulated เป็นหน้าที่ให้เหรียญรางวัลโดยประเมินจากคะแนนที่ผู้เล่นทำได้ มีทั้งหมด 3 เหรียญ คือ เหรียญทอง เหรียญเงิน และเหรียญทองแดง

รูปที่ 4.39 หน้า Congratulated

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2 ส่วนติดต่อกับผู้ใช้ที่ควบคุมด้วยการสัมผัส

การควบคุมด้วยการสัมผัสจะเริ่มตั้งแต่หน้าวิธีการเล่น เพื่อให้เกิดความต่อเนื่องไปจนถึงตัวเกม ในหน้าของวิธีการเล่นจะยังใช้ภาพกราฟฟิกสองมิติแสดงผล

4.3.2.1 การสร้างปุ่มสัมผัส

การสร้างปุ่มสัมผัสแทนการควบคุมด้วยเมาส์และคีย์บอร์ดนั้น จะใช้ภาพกราฟฟิกสองมิติที่ตัวปุ่ม และใช้เทคนิคการประมวลผลภาพในการตรวจจับว่าผู้เล่นได้สัมผัสกับปุ่มนั้นๆหรือไม่ โดยแสดงระดับการถูกสัมผัสด้วยเอฟเฟกต์ที่สร้างจาก Mercury Particle Engine

(ก) (ข) (ค)

รูปที่ 4.40 การสร้างปุ่มสัมผัสในเกม

(ก) ภาพกราฟฟิกปุ่ม

(ข) ปุ่มเมื่อถูกสัมผัส แสดงระดับการถูกสัมผัสด้วยดวงไฟสีเขียว

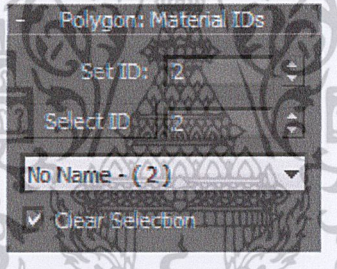
(ค) ปุ่มเมื่อถูกสัมผัสจนขึ้นดวงไฟสีแดง แสดงถึงการเลือกเมนูนั้นๆ

4.3.2.2 การสร้างโมเดลสามมิติ (3D Models)

การนำโมเดลสามมิติมาใช้ในเกมนั้นเพื่อต้องการให้เกิดความสมจริง และทำให้เกมดูมีมิติมากขึ้น โดยโมเดลทั้งหมดสร้างจากโปรแกรม 3ds Max ซึ่งเป็นโปรแกรมสำหรับสร้างงานสามมิติ โดยเฉพาะการสร้างโมเดลสามมิติเพื่อนำไปใช้กับ XNA Game Studio Express นั้น จำเป็นที่จะต้อง export โมเดลออกมาเป็น .x เสียก่อน โดยการเพิ่มปลั๊กอินที่ชื่อว่า kW X-port หรือ Pandasofts DirectX Exporter ซึ่งในที่นี้เลือกใช้ปลั๊กอิน Pandasofts DirectX Exporter ในการ export ไฟล์ .x โดยมีวิธีการทำดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

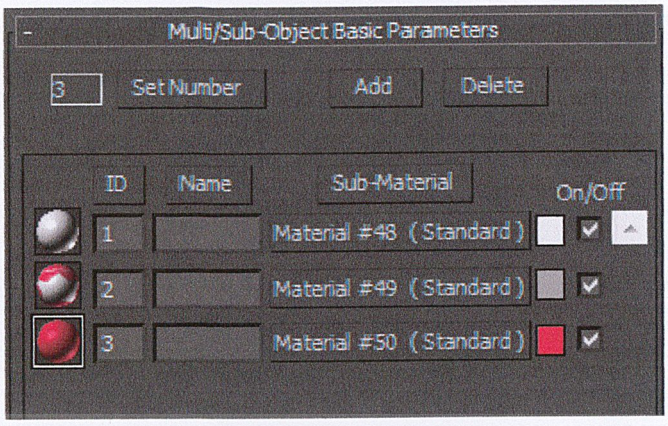
1. ขั้นตอนการสร้างโมเดล (Modeling)
   - ขึ้นโมเดลจากรูปทรงพื้นฐาน (Standard Primitive)
   - ทำการแปลงโมเดลให้เป็น Polygon โดยคลิกขวาที่โมเดล เลือกคำสั่ง Convert to Editable Poly
   - ทำการปรับแต่งแก้ไขรูปทรงให้มีรายละเอียดตามต้องการ
2. ขั้นตอนการใส่พื้นผิว (Material & Texture)
   - ในกรณีนี้ต้องการใส่หลายลวดลายลงในโมเดลชิ้นเดียว จึงต้องมีการกำหนด ID ให้กับตัวโมเดลเสียก่อน
     I. ที่ Modify Panel หัวข้อ Selection เลือกการทำงานกับ Polygon
     II. คลิกเลือก Polygon ที่ต้องการจะใส่ลวดลายแบบแรก
     III. ที่ Modify Panel หัวข้อ Polygon: Material IDs กำหนดค่า ID เป็น 1 เพื่อบอกให้โปรแกรมทราบว่า Polygon ส่วนนี้จะใส่ลวดลายแบบที่ 1 และทำอย่างนี้ไปจนครบลวดลายที่ต้องการจะใส่



รูปที่ 4.41 การกำหนด Material ID

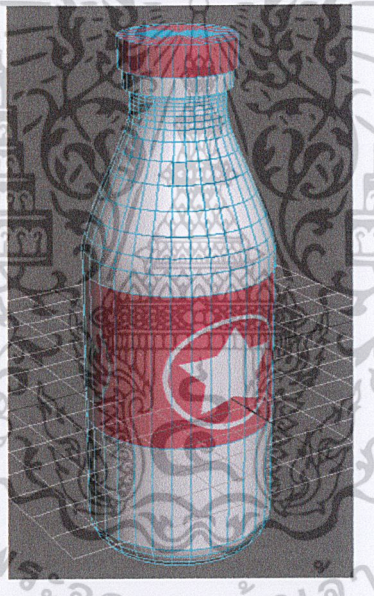
   - เปิด Material Editor (M) คลิกที่ Standard แล้วเลือก Multi/Sub-Object กำหนดจำนวน Material ตามจำนวน ID ที่ได้ระบุไว้
   - กำหนดลายละเอียดของ Material แต่ละ ID โดยคลิกที่ปุ่ม Material #no (Standard)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.42 Material Editor ในการทำงานแบบ Multi/Sub-Object

- แล้วคลิก Material จาก Material Slot ลากไปใส่ในโมเดล



รูปที่ 4.43 โมเดลที่ใส่ Material แบบ Multi/Sub-Object

3. ขั้นตอนการ Export (Exporting)
   - คลิกที่เมนูหลักของโปรแกรม แล้วเลือก Export
   - กำหนดชื่อไฟล์และที่อยู่ของไฟล์ พร้อมเลือกประเภทไฟล์เป็น Panda DirectX (*.X)
   - จากนั้นกำหนดค่าต่างๆตามต้องการ และกด OK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3.2.3 หน้าจอของส่วนติดต่อกับผู้ใช้ที่ควบคุมด้วยการสัมผัส

1) หน้า Tutorial เป็นหน้าที่บอกวิธีการเล่นของแต่ละเกม

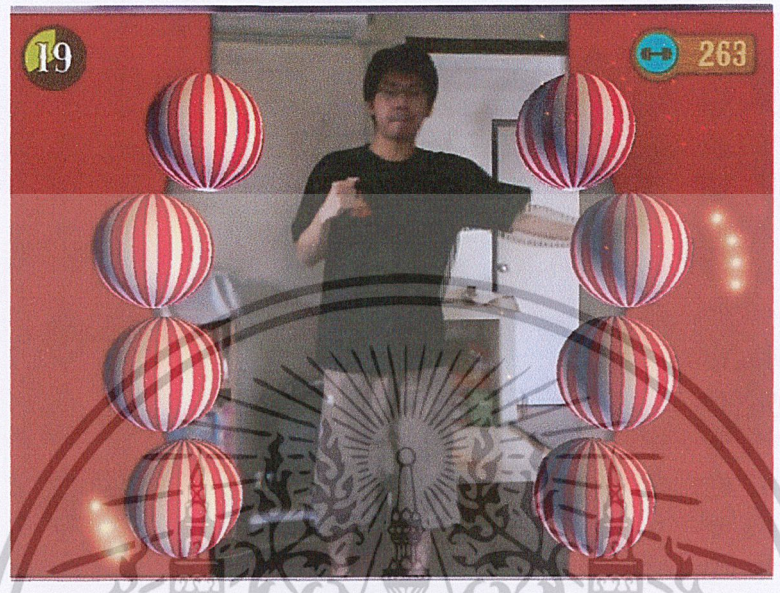
รูปที่ 4.44 หน้า Tutorial

2) หน้า Calibrate เป็นหน้าที่ไกด์ให้ผู้เล่นทราบว่าควรยืนเล่นที่ระยะใด

รูปที่ 4.45 หน้า Calibrate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) หน้าเกมหน้าเกมทั้ง 4 เกม จะมีรูปแบบที่เหมือนกันคือ มุมซ้ายบนมีนาฬิกา และมุมขวาบนมีแถบคะแนน ซึ่งแถบคะแนนในแต่ละเกมจะมีการใส่ไอคอนของเกมนั้นๆให้ผู้เล่นทราบด้วย



รูปที่ 4.46 หน้าเกม

รูปที่ 4.47 ไอคอนของแต่ละเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
สรุปผลโครงงาน และข้อเสนอแนะ

ในบทนี้จะกล่าวถึงบทสรุปของการทำงานให้เกิดเป็นเกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส โดยการทำงานร่วมกับเทคนิคการประมวลผลภาพตามที่ได้ศึกษากับเทคนิคในการสร้างเกมต่างๆ เช่น ระบบฟิสิกส์ และการนำโมเดลสามมิติมาใช้ร่วมกันภายในเกม ในบทนี้จะกล่าวถึงปัญหาและข้อจำกัดที่เกิดขึ้นระหว่างการพัฒนา รวมถึงข้อเสนอแนะที่จะมีประโยชน์ต่อผู้ที่สนใจศึกษา หรือนำไปพัฒนาให้เกมมีการทำงานหรือรูปแบบการเล่นที่ดียิ่งขึ้น

5.1 สรุปผลโครงงาน

การพัฒนาเกมเชิงปฏิสัมพันธ์โดยใช้การตรวจจับของร่างกายผ่านกล้องเว็บแคม: มายฟิตเนส เป็นการพัฒนาเกมในรูปแบบใหม่ ซึ่งอาศัยการเคลื่อนไหวร่างกายของผู้เล่นในการเล่นและควบคุมเกม โดยมีกล้องเว็บแคมเป็นอุปกรณ์ในการนำเข้าภาพ โดยมีจุดประสงค์ต้องการให้ผู้เล่นได้เคลื่อนไหวร่างกายแทนที่จะต้องนั่งเล่นเกมแบบปกติ รูปแบบของเกมจะเป็นการวัดสมรรถภาพของผู้เล่น 4 ด้าน ประกอบไปด้วย ความอดทนของกล้ามเนื้อ การประสานสัมพันธ์ ความยืดหยุ่น และความคล่องแคล่ว เพื่อความน่าสนใจมากยิ่งขึ้นจึงได้มีการนำระบบฟิสิกส์ ระบบเอฟเฟกต์ และการใช้โมเดลสามมิติมาใช้ร่วมกับเกมด้วย

ในการเล่นเกมจะต้องใช้เทคนิคการประมวลผลภาพในการประมวลผลและตรวจจับการเคลื่อนไหวของผู้เล่น โดยเกมของการวัดสมรรถภาพด้านความอดทนกล้ามเนื้อ และการประสานสัมพันธ์นั้นจะใช้การตรวจสอบส่วนที่มีการเคลื่อนไหว ซึ่งระบบสามารถตรวจจับได้อย่างถูกต้องและมีการแสดงผลสัมพันธ์กับการเคลื่อนไหวของผู้เล่นได้อย่างดี เกมของการวัดสมรรถภาพด้านความยืดหยุ่น และความคล่องแคล่วจะใช้การตรวจสอบโครงร่างของผู้เล่น ซึ่งระบบมาสามารถตรวจจับออกมาได้อย่างถูกต้อง และตัวเกมมีการแสดงผลได้สัมพันธ์กับการเคลื่อนไหวของผู้เล่นได้เป็นอย่างดี การทำงานในระบบฟิสิกส์ การแสดงผลเอฟเฟกต์ รวมถึงการแสดงผลโมเดลสามมิติ ก็มีการแสดงผลได้อย่างถูกต้องและสอดคล้องกับตัวเกม โดยรวมแล้วระบบสามารถช่วยให้ผู้เล่นได้เคลื่อนไหวร่างกาย สามารถลดความเสี่ยงเรื่องสุขภาพจากการนั่งเล่นเกมในรูปแบบเดิมๆได้ อีกทั้งยังสามารถสร้างความสนุกสนานให้แก่ผู้เล่นได้ไปพร้อมกันอีกด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 ปัญหาและข้อจำกัดในการทำงานของระบบ

ในระหว่างการพัฒนาระบบ ได้มีการศึกษาและทดลองปฏิบัติตามทฤษฎีต่างๆ จนถึงขั้นตอนการพัฒนาและทดลองระบบ สามารถสรุปปัญหาและข้อจำกัดของระบบได้ดังนี้

1. เนื่องจากข้อจำกัดของภาษาที่ใช้ทำให้ไม่สามารถแสดงผลรูปที่มาจากเว็บแคมได้โดยตรงต้องผ่านการแปลงข้อมูลก่อน ซึ่งทำให้การทำงานช้าลง รูปที่ได้จะล่าช้ากว่าการเคลื่อนไหวของผู้เล่นขณะนั้น โดยผู้จัดทำได้มีการแก้ไขในส่วนนี้ด้วยวิธีการแก้ไขปรับปรุงกระบวนการทำงาน ทำให้ช่วยลดปัญหาความล่าช้าของภาพไปได้มาก
2. เทคนิคการประมวลผลภาพส่วนใหญ่มีการทำงานที่ใช้เวลานาน บางเทคนิคเมื่อนำมาประยุกต์ใช้แล้วพบว่าใช้เวลาในการประมวลผลนานมาก ซึ่งไม่สามารถนำมาใช้กับตัวเกมได้ เพราะต้องคำนึงถึงเรื่องประสิทธิภาพ ทำให้ไม่สามารถนำเทคนิคในการประมวลผลภาพรูปแบบต่างๆมาใช้ประยุกต์กับการออกแบบเกมได้อย่างเต็มที่
3. เทคนิคการประมวลผลรูปที่ใช้พัฒนาระบบ ใช้การหาความแตกต่างของระดับสีเป็นตัวแยกแยะความแตกต่าง และใช้การตรวจสอบการเคลื่อนไหว ถ้าเกิดกรณีที่ผู้เล่นมีสีที่คล้ายหรือเหมือนกับพื้นหลังจะทำให้ระบบไม่สามารถแยกแยะความแตกต่างนั้นได้
4. ระบบไม่สามารถแยกแยะได้ว่าสิ่งที่เคลื่อนไหวอยู่นั้นเป็นผู้เล่นหรือสิ่งอื่นที่ที่มีการเคลื่อนไหว
5. มีการนำเทคนิคลบฉากหลังมาใช้ในเกม ซึ่งจุดอ่อนของเทคนิคนี้คือต้องมีการควบคุมสภาพแวดล้อม ถ้าเกิดมีการเปลี่ยนแปลงของสภาพแวดล้อม เช่น แสง หรือวัตถุอื่นที่เคลื่อนไหว ก็จะทำให้การประมวลผลผิดพลาด
6. การแสดงผลโมเดลสามมิติใน XNA ผิดเพี้ยน ซึ่งเกิดจากการตั้งค่าเกี่ยวกับโมเดลที่จะต้องให้เป็นไปตามข้อกำหนดของ XNA ถึงจะแสดงผลได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 ข้อเสนอแนะในการพัฒนาระบบต่อไป

1. พัฒนาเทคนิคการประมวลผลภาพในการตรวจจับสิ่งที่สนใจให้ดียิ่งขึ้น
2. พัฒนาประสิทธิภาพในการทำงานของระบบให้ดียิ่งขึ้น
3. เพิ่มเติมเทคนิคการประมวลผลภาพ และรูปแบบของเกมให้มีความหลากหลาย มีการทดสอบสมรรถภาพด้านต่างๆเพิ่มมากขึ้น
4. พัฒนาให้สามารถเล่นบนระบบเครือข่ายได้ ในรูปแบบออนไลน์
5. มีการนำข้อมูลของผู้เล่น เช่น เพศ อายุ ส่วนสูง น้ำหนัก ไปปรับปรุงให้เกมมีความเหมาะสมกับข้อมูลเหล่านั้น เช่น สามารถให้คะแนนสมรรถภาพแยกตามช่วงอายุได้ เป็นต้น
6. พัฒนาให้ผู้เล่นสามารถเลือกเกมตามต้องการได้ โดยไม่ต้องเล่นต่อเนื่องกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

[1] ชานน ศรวิเศษสม และ ธนศร คันธวัฒน์. 2549. **ระบบช่วยฝึกสอนการออกกำลังกายโดยการประยุกต์ใช้เทคนิคการตรวจจับการเคลื่อนไหวของมนุษย์จากวีดีทัศน์**. ปริญญานิพนธ์วิทยาศาสตรบัณฑิต คณะเทคโนโลยีสารสนเทศ สาขาวิชาเทคโนโลยีสารสนเทศ, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[2] ธนพล ภูวรัตนกุล และคณะ. 2551. **การวิเคราห์การเคลื่อนไหวของร่างกาย**. ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[3] ชานนทร์ เรือนจรัสศรี และคณะ. 2551. **การพัฒนาโลกเสมือนสามมิติด้วย XNA GAME STUDIO**. ปริญญานิพนธ์วิศวกรรมศาสตรบัณฑิต คณะวิศวกรรมศาสตร์ สาขาวิศวกรรมคอมพิวเตอร์, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

[4] สัจจะ จรัสรุ่งรวีวร. 2552. **เริ่มต้น Visual C# 2008 ฉบับสมบูรณ์**. นนทบุรี : ไอดีซี อินโฟดิสทริบิวเตอร์ เซ็นเตอร์.

[5] Alexandre Santos Lobão et al. 2009. **Beginning XNA 3.0 Game Programming From Novice to Professional**. California : Apress.

[6] Chad Carter. 2009. **Microsoft XNA Game Studio 3.0 Unleashed**. Indiana : Pearson.

[7] **XNA Community Thailand**. 2007. [Online]. Available : http://www.thaixna.com/index.php/webboard-topmenu-47.

[8] Bradski, G. and Kaehler, A. 2008. **Learning OpenCV**. California : O'Reilly Media.

[9] **Emgu CV: OpenCV in .NET (C#, VB, C++ and more)**. 2009. [Online]. Available : http://www.emgu.com/wiki/index.php/Main_Page.

[10] Microsoft. n.d. **MSDN Library XNA Game Studio 3.1**. [Online]. Available : http://msdn.microsoft.com/en-us/library/bb200104(v=XNAGameStudio.31).aspx

[11] CodePlex. n.d. **Mercury Particle Engine Documentation**. [Online]. Available : http://mpe.codeplex.com/documentation

[12] CodePlex. n.d. **Farseer Physics Engine 3.2 Manual**. [Online]. Available : http://farseerphysics.codeplex.com/documentation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม (ต่อ)

[13] คณะกรรมการส่งเสริมกีฬาและการออกกำลังกายเพื่อสุขภาพในสถาบันการศึกษา และการพัฒนาองค์ความรู้. 2549. **แบบทดสอบและเกณฑ์มาตรฐานสมรรถภาพทางกายที่สัมพันธ์กับสุขภาพ สำหรับเด็กไทย อายุ 7 – 18 ปี**. กรุงเทพฯ : สำนักงานกองทุนสนับสนุนการสร้างเสริมสุขภาพ (สสส.).

[14] ฝ่ายวิทยาศาสตร์การกีฬาร่วมกับศูนย์กกท. จังหวัด. 2549. **วิทยาศาสตร์การกีฬากับการทดสอบสมรรถภาพทางกาย**. กรุงเทพฯ : การกีฬาแห่งประเทศไทย.

[15] วิรัตน์ ถนอมทรัพย์. ม.ป.ป. **สมรรถภาพทางกาย**. [Online]. Available : http://pirun.ku.ac.th/~g4786057/s_03.htm

[16] Topendsports. 1997. **Fitness Testing**. [Online]. Available : http://www.topendsports.com/testing/tests.htm

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

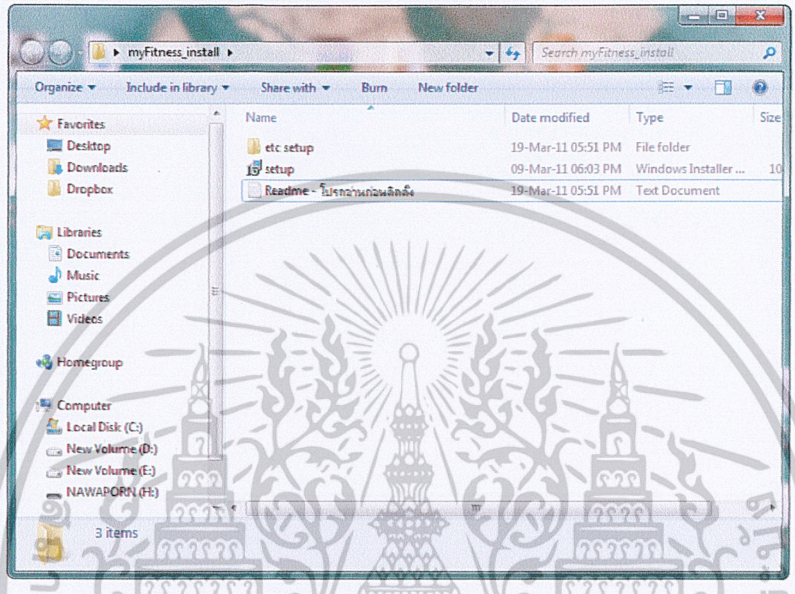
ภาคผนวก ก.

คู่มือการติดตั้งเกม myFitness

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คู่มือการติดตั้งโปรแกรม

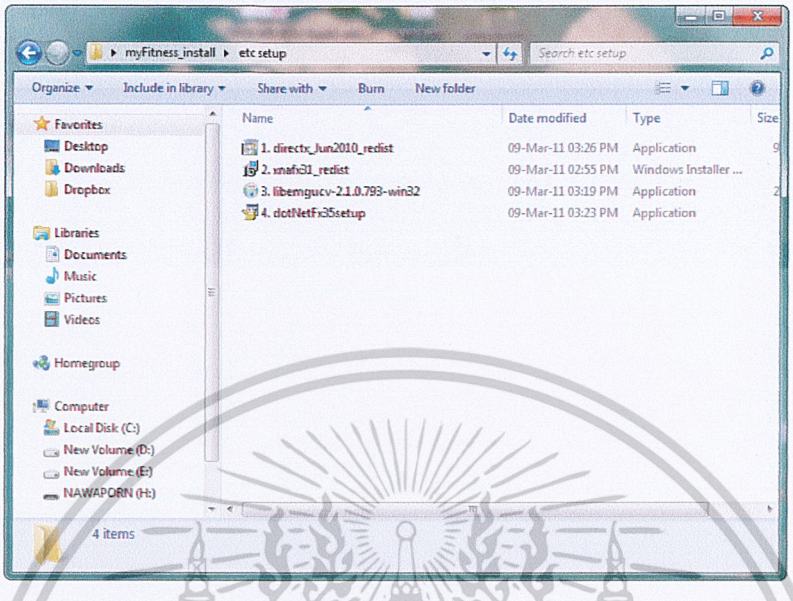
1. เมื่อเปิดโฟลเดอร์ myFitness_install ขึ้นมาจะพบ 3 ไฟล์ คือ โฟลเดอร์ etc setup, setup และ Text file ชื่อ Readme – โปรดอ่านก่อนติดตั้ง (แนะนำให้ผู้ใช้งานอ่านไฟล์ Readme ก่อน เพื่อที่จะได้ติดตั้งโปรแกรมได้อย่างถูกต้อง)



รูปที่ ก.1 แสดงหน้าจอโฟลเดอร์ myFitness_install

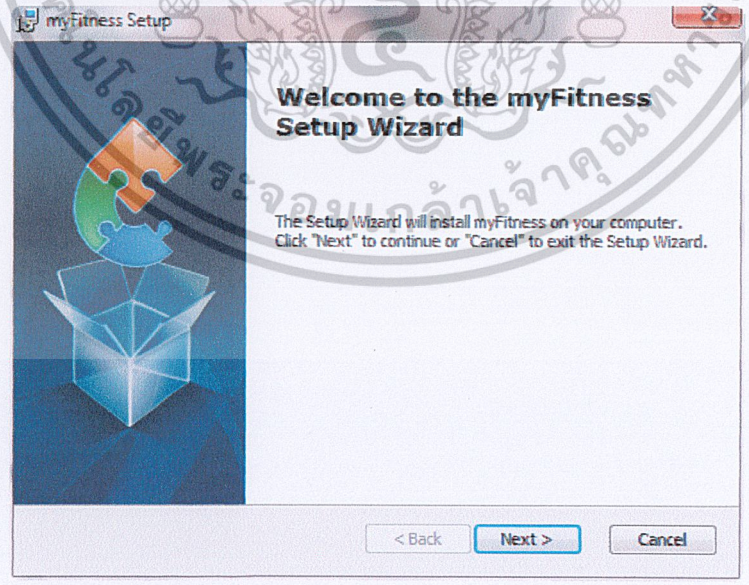
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. ให้เริ่มทำการติดตั้งโปรแกรมพื้นฐานที่อยู่ภายในโฟลเดอร์ etc setup ก่อน เมื่อเปิดขึ้นมาจะพบไฟล์ดังรูปที่ ก.2 และให้เริ่มลงโปรแกรมพื้นฐานตามที่ระบุไว้ในไฟล์ Readme



รูปที่ ก.2 แสดงหน้าจอไฟล์ที่อยู่ในโฟลเดอร์ etc setup

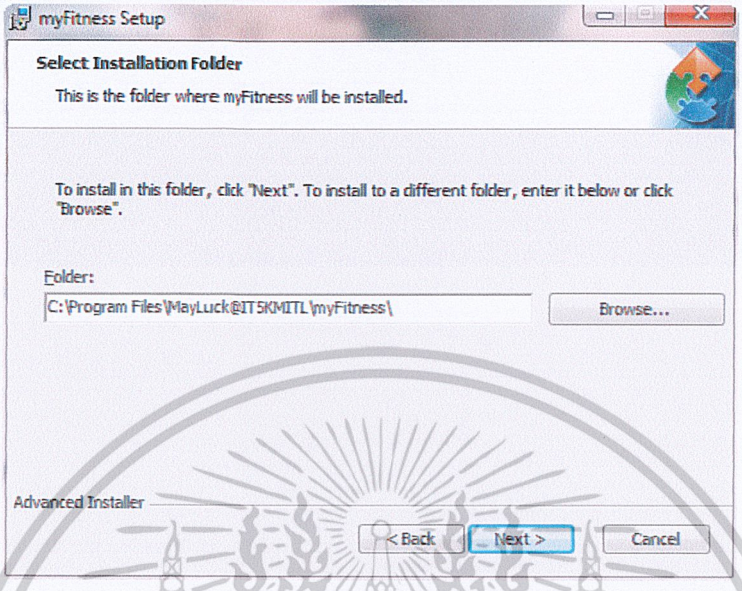
3. หลังจากลงไฟล์พื้นฐานทั้ง 4 ไฟล์เรียบร้อยแล้ว ให้กลับไปยังโฟลเดอร์ myFitness_install เพื่อทำการลงตัวโปรแกรมจริง โดยดับเบิ้ลคลิกที่ไฟล์ setup หลังจากดับเบิ้ลคลิกจะแสดงหน้าจอดังรูปที่ ก.3 จากนั้นคลิก “Next” เพื่อเริ่มการติดตั้งโปรแกรม



รูปที่ ก.3 แสดงหน้าจอการเริ่มติดตั้งโปรแกรม

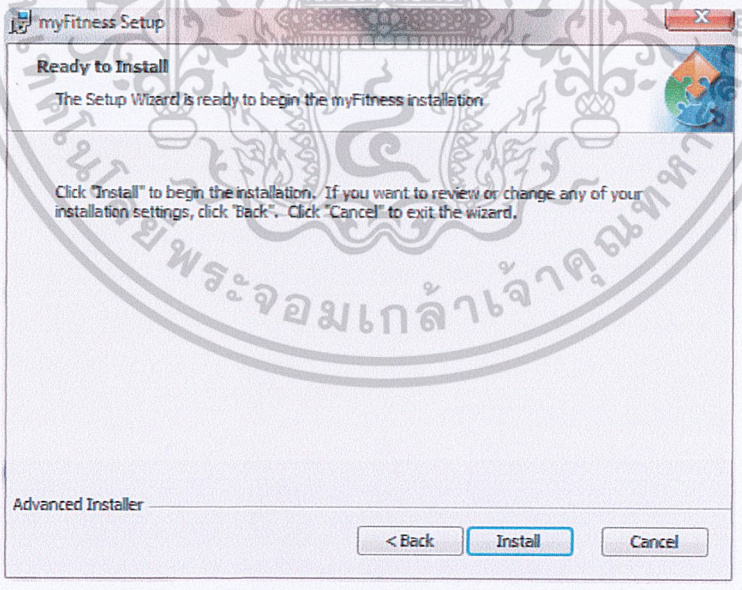
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จากนั้นจะเข้าสู่หน้าเลือกที่อยู่ของไฟล์ สามารถเลือกตำแหน่งที่อยู่ของไฟล์ได้โดยคลิก “Browse” เมื่อเสร็จเรียบร้อย คลิก “Next”



รูปที่ ก.4 แสดงหน้าจอเลือกที่อยู่ไฟล์

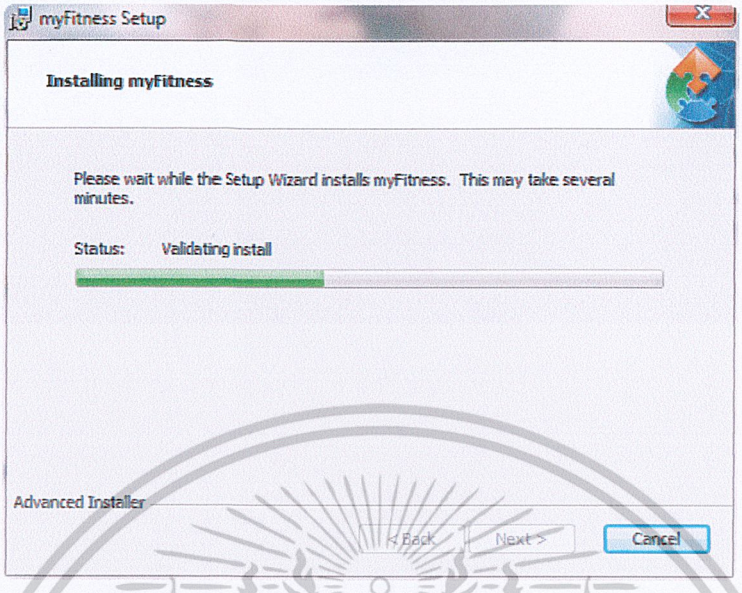
5. จากนั้นโปรแกรมจะเข้าสู่หน้ายืนยันการติดตั้งโปรแกรม คลิก “Install” เพื่อเริ่มการติดตั้ง



รูปที่ ก.5 แสดงหน้าจอยืนยันการติดตั้งโปรแกรม

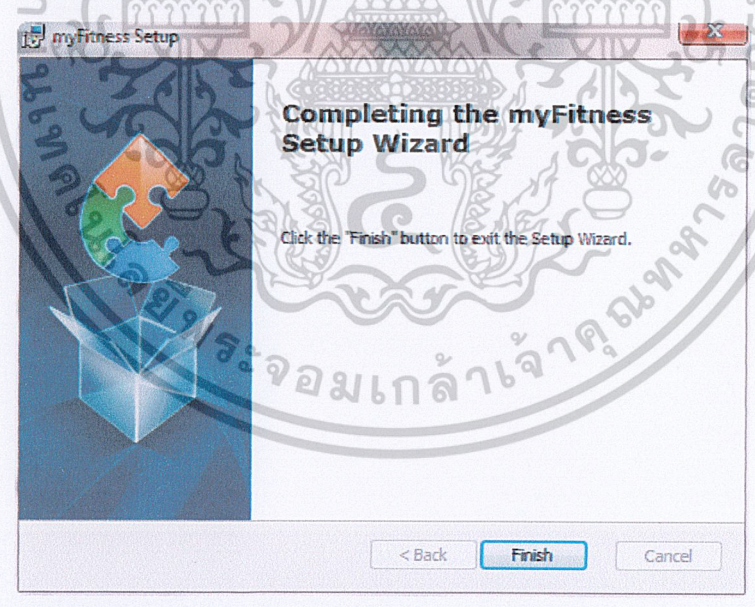
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6. จากนั้นโปรแกรมจะเริ่มทำการติดตั้งแสดงได้ดังรูปที่ ก.6



รูปที่ ก.6 แสดงหน้าจอการกำลังติดตั้งโปรแกรม

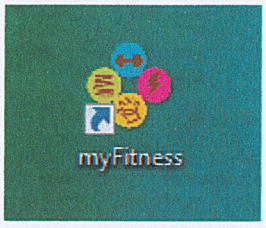
7. เมื่อติดตั้งโปรแกรมเสร็จจะแสดงหน้าจอดังรูปที่ ก.7



รูปที่ ก.7 แสดงหน้าจอเสร็จสิ้นการติดตั้งโปรแกรม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ที่ Desktop จะปรากฏไอคอนเกม ดังรูปที่ ก.8



รูปที่ ก.8 แสดงไอคอนเกมที่หน้า Desktop

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ข.

การวิเคราะห์ข้อมูลและออกแบบเกณฑ์สมรรถภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การวิเคราะห์ข้อมูลและออกแบบเกณฑ์สมรรถภาพ

จุดประสงค์เพื่อให้สามารถสร้างเกณฑ์การให้คะแนนในแต่ละเกมได้อย่างเหมาะสม และสามารถวัดได้จริง โดยทำการเก็บข้อมูลจากกลุ่มตัวอย่างนักเรียน นักศึกษา เพศชาย – หญิง อายุระหว่าง 18 – 23 ปี จำนวน 30 คนจากการทดลองเล่นเกมทั้ง 4 เกม ได้แก่ เกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ เกมทดสอบสมรรถภาพการประสานสัมพันธ์ เกมทดสอบสมรรถภาพความยืดหยุ่น และเกมทดสอบสมรรถภาพความคล่องแคล่ว ซึ่งจากการคำนวณ และวิเคราะห์ข้อมูลแล้ว สามารถสรุปเป็นเกณฑ์ในแต่ละเกมได้ดังนี้

ตารางที่ ข.1 เกณฑ์คะแนนเกมทดสอบสมรรถภาพความอดทนของกล้ามเนื้อ

ระดับ	คะแนน
ดีเลิศ (Excellent)	> 135
ดี (Good)	88 – 134
ต้องปรับปรุง (Poor)	< 87

ตารางที่ ข.2 เกณฑ์คะแนนเกมทดสอบสมรรถภาพการประสานสัมพันธ์

ระดับ	คะแนน
ดีเลิศ (Excellent)	> 38
ดี (Good)	17 – 37
ต้องปรับปรุง (Poor)	< 16

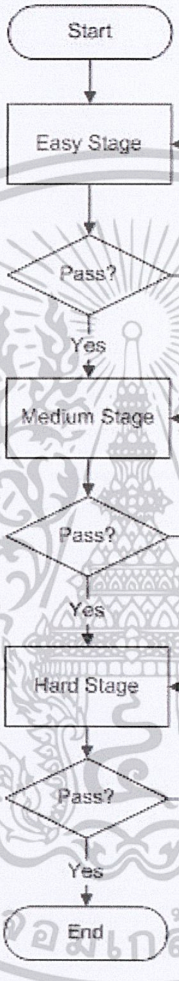
ตารางที่ ข.3 เกณฑ์คะแนนเกมทดสอบสมรรถภาพความคล่องแคล่ว

ระดับ	คะแนน	
	หลบของที่ต้องหลบ	เก็บของที่ต้องเก็บ
ดีเลิศ (Excellent)	> 26	> 22
ดี (Good)	18 – 25	> 22
ต้องปรับปรุง (Poor)	< 17	< 21

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในเกมทดสอบสมรรถภาพความยืดหยุ่นจะมีวิธีการคิดคะแนนที่แตกจากเกมอื่นๆ โดยการเก็บข้อมูลนั้นจะให้กลุ่มตัวอย่างทดลองเล่นในแต่ละด่าน ซึ่งมีทั้งหมด 21 ด่าน แล้วบันทึกจำนวนคนที่ผ่านในด่านนั้นๆ เพื่อจัดอันดับความยากง่ายของด่านทั้ง 21 ด่านนี้

การทำงานของเกมจะให้เริ่มเล่นที่ระดับง่าย 3 ด่านก่อน ถ้าหากผู้เล่นสามารถผ่านไปได้ ก็จะเข้าสู่ด่านระดับปานกลาง และถ้าหากผ่านระดับปานกลางได้ก็จะเข้าสู่ระดับยากต่อไป



รูปที่ ข.1 การทำงานของเกมทดสอบสมรรถภาพความยืดหยุ่น

นอกจากการให้คะแนนสมรรถภาพแล้ว ตัวเกมยังมีการคำนวณค่าดัชนีมวลกายด้วย ซึ่งใช้เกณฑ์ดังนี้

$$\text{BMI} = \frac{\text{weight (kg.)}}{\text{height (m.)}^2} \qquad \text{(ข.1)}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ ข.4 แสดงค่าดัชนีมวลกายแยกตามเพศ

	ผู้ชาย	ผู้หญิง
ผอม (Underweight)	< 20	< 19
สุขภาพดี (Healthy Range)	20 – 25	19 – 24
อ้วน (Overweight)	26 – 30	25 – 30
อ้วนเกินไป (Obese)	> 30	> 30

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อ – นามสกุล	นายเมธาวี อยู่อ่อน
วัน เดือน ปีเกิด	4 พฤษภาคม 2532
ที่อยู่	7/28 หมู่ 17 ถนน บางนา-ตราด ตำบล บางพลีใหญ่ อำเภอ บางพลี จังหวัด สมุทรปราการ 10540
โทรศัพท์	02-316-8245
อีเมล	maythawee_may@hotmail.co.th
ประวัติการศึกษา	
2553	วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ชื่อ – นามสกุล	นางสาวนวพร อุปริบุญ
วัน เดือน ปีเกิด	21 พฤศจิกายน 2531
ที่อยู่	977/4 ซอยพระราม3 57 ถนนพระรามที่ 3 แขวงบางโพงพาง เขตยานนาวา กรุงเทพฯ 10120
โทรศัพท์	089-206-4025
อีเมล	nawponz@gmail.com
ประวัติการศึกษา	
2553	วิทยาศาสตรบัณฑิต สาขาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้