ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

การพัฒนาเกมคอมพิวเตอร์สื่อประสม
โดยอาศัยการประมวลผลภาพวิดีโอ

DEVELOPMENT OF A MULTIMEDIA COMPUTER GAME
USING VIDEO PROCESSING

H005975

โดย

ปัญญนุช ลักษณวงษ์

PANYANUCH LUKSANAWONG

อาจารย์ที่ปรึกษา

รศ.ดร.นพพร โชติกกำธร

QA.
ป523ก
2551

เลขหมู่.....
เลขทะเบียน 05975
วัน,เดือน,ปี ๒ 3 ก.พ. 2553

b. 12174208
i.....

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน

หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคฤดูร้อน ปีการศึกษา 2551

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DEVELOPMENT OF A MULTIMEDIA COMPUTER GAME USING VIDEO PROCESSING

PANYANUCH LUKSANAWONG

A SYSTEM DEVELOPMENT PROJECT

OF THE REQUIREMENT FOR THE DEGREE OF

MASTER OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

SUMMER/ 2008

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใ... เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COPYRIGHT 2009

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	การพัฒนาเกมคอมพิวเตอร์สื่อประสม โดยอาศัยการประมวลผลภาพวิดีโอ
นักศึกษา	นางสาวปัญญนุช ลักษณวงษ์
รหัสนักศึกษา	47066148
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2551
อาจารย์ที่ปรึกษา	รศ.ดร.นพพร โชติกกำธร

บทคัดย่อ

ปัจจุบัน วิธีการเล่นเกมยังคงใช้วิธีการเล่นแบบเดิมๆ นั้นคือ ต้องนั่งเล่นอยู่หน้าจอคอมพิวเตอร์ โดยใช้อุปกรณ์จำพวกเมาส์และคีย์บอร์ดควบคุมการเล่นเกมอยู่ ซึ่งวิธีการเล่นเกมแบบนี้ อาจทำให้ผู้เล่นเกิดความเบื่อหน่ายและเมื่อยล้าได้ ดังนั้นจึงได้มีการจัดทำเกมที่ใช้วิธีการเล่นในรูปแบบใหม่ เพื่อดึงดูดใจและเพิ่มความน่าสนใจให้กับการเล่นเกมมากยิ่งขึ้น ด้วยการออกแบบและพัฒนาเกมคอมพิวเตอร์สื่อประสมที่อาศัยการประมวลผลภาพจากกล้องวิดีโอเว็บแคม โดยตัวเกมจะมีระบบตรวจจับการเคลื่อนไหวของตำแหน่งขาผู้เล่น แล้วจึงนำข้อมูลที่ตรวจจับได้ไปควบคุมการเคลื่อนที่ของตัวละครภายในเกม และจากลักษณะการทำงานของเกมนี้ สามารถแบ่งส่วนการทำงานออกได้เป็น 3 ส่วน คือ ส่วนที่ 1 เป็นส่วนที่ใช้ในการตรวจจับและติดตามตำแหน่งขาของผู้เล่น โดยอาศัยการประมวลผลภาพจากกล้องวิดีโอเว็บแคม ส่วนที่ 2 เป็นส่วนที่ใช้ในการสร้างเกม โดยใช้โปรแกรม Flash และส่วนที่ 3 เป็นส่วนที่ใช้ทำการเชื่อมต่อระหว่างโปรแกรม 2 ส่วนนี้ เพื่อทำให้ส่วนที่ 2 สามารถรับค่าจากส่วนที่ 1 ได้ โดยเกมนี้จัดทำขึ้นเพื่อรณรงค์ให้เด็กๆได้มีความรู้และตระหนักถึงพิษภัยของยาเสพติดผ่านทางการเล่นเกม ซึ่งจะแบ่งออกเป็น 2 บท คือ บทการเรียนรู้และบทการปราบปราม โดยบทแรก จะให้ผู้เล่นได้เรียนรู้เรื่องราวเกี่ยวกับยาเสพติด ผ่านทางการตอบคำถาม เล่นด้วยการเลือกเหยียบคำตอบที่คิดว่าถูกต้อง และบทที่ 2 จะให้ผู้เล่นทำการปราบปรามยาเสพติด ด้วยการกระโดดไปยังจุดที่ยาเสพติดอยู่ เพื่อต่อสู้กับยาเสพติดนั้นๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	Development of A Multimedia Computer Game Using Video Processing
Student	Miss. Panyanuch Luksanawong
Student ID.	47066148
Degree	Master of Science
Programme	Information Technology
Academic Year	2008
Advisor	Assoc. Prof. Dr. Nopporn Chotikakarmthorn

ABSTRACT

Nowadays, A method for playing a game depends on standard input devices. A player sits in front of a computer screen using mouse and keyboard for controlling the game. This makes a player bored and tired. Therefore, we develop a game with the new interaction for increasing the attractiveness of the game. The design and development of this game applies video processing with webcam and an algorithm for detecting the movement of the legs position player. A detected position is used to control a movement of a character in the game. The process of this game can be divide into three parts. The first part is an implementation of a leg position tracking algorithm. It is responsible for estimating a player by position leg. The second part is a game development using flash. The third part is used to connect 2 programs from part 1 and 2 for sending data from part 1 to part 2. This game is made for children to increase knowledge about narcotic. It is divided into 2 chapters, Learning chapter and Subdue chapter. In chapter 1 a player answers the question by tread on the right answer and chapter 2 a player jump to a narcotic point for fighting.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ รศ. ดร. นพพร โชติกกำธร อาจารย์ที่ปรึกษาโครงงาน ที่ให้ความกรุณา ให้ความรู้ ให้คำปรึกษา และคำแนะนำต่างๆ ในการพัฒนาโครงงานนี้ จนโครงงางนี้สามารถสำเร็จลุล่วงไปด้วยดี

ขอกราบขอบพระคุณคณาจารย์ทุกท่าน ในคณะเทคโนโลยีสารสนเทศ ที่กรุณาถ่ายทอดความรู้ และเทคนิคต่างๆ ในการศึกษา ตลอดระยะเวลาที่ศึกษาในสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ขอขอบคุณพี่นรงค์ศักดิ์ ที่คอยให้คำปรึกษาเรื่องหลักการ วิธีการและอัลกอลิทึมที่ใช้ในการประมวลผลภาพวิดีโอ

ขอขอบคุณพี่เอ๊ะ อ๊อฟ โอ น้องมายด์ และเพื่อนๆ ที่ช่วยให้คำปรึกษาเรื่องต่างๆ และเรื่องวิธีการที่ใช้ในการแก้ปัญหาโปรแกรมยามติดขัด

สุดท้ายขอกราบขอบพระคุณคุณพ่อ คุณแม่ และบุคคลในครอบครัว ที่ให้กำลังใจและให้การอุปการะมาโดยตลอด ขอกราบขอบพระคุณเป็นอย่างสูง

คุณความดีและประโยชน์ที่ได้รับจากโครงงานนี้ ผู้เขียนขอมอบแด่ผู้มีพระคุณทุกท่าน

ปัญญนุช ลักษณวงษ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....I

บทคัดย่อภาษาอังกฤษ.....II

กิตติกรรมประกาศ.....III

สารบัญ.....IV

สารบัญตาราง.....VII

สารบัญรูป.....VIII

บทที่ 1 บทนำ.....1

1.1 ความเป็นมาและความสำคัญของปัญหา.....1

1.2 ความมุ่งหมายและวัตถุประสงค์ของโครงการ.....1

1.3 ประโยชน์ที่คาดว่าจะได้รับ.....2

1.4 ขอบเขตของการศึกษา.....2

1.5 ขั้นตอนและวิธีการดำเนินโครงการ.....3

บทที่ 2 หลักและเทคนิคพื้นฐานที่ใช้ในการพัฒนาเกม.....4

2.1 หลักการพื้นฐานของเกมโดยทั่วไป.....4

2.1.1 ประเภทของเกม.....4

2.1.2 มุมมองภายในเกม.....5

2.1.3 สิ่งสำคัญที่ต้องคำนึงในการออกแบบเกม.....7

2.1.4 การเพิ่มความน่าสนใจให้กับเกม.....8

2.2 หลักการและเทคนิคที่ใช้ในการพัฒนาเกมด้วยโปรแกรม Flash.....10

2.2.1 ระบบพิกัดที่ใช้ในโปรแกรม Flash.....10

2.2.2 การเขียน ActionScript เพื่อควบคุมการทำงานภายในโปรแกรม Flash.....11

2.2.3 การสร้างกลไกการทำซ้ำ (Loop) ของ Flash.....12

2.2.4 เทคนิคการสร้างงาน 3 มิติ ใน Flash.....14

บทที่ 3 อัลกอริทึมในการตรวจวัดตำแหน่งขาด้วยกล้องวีดีโอ.....16

3.1 การประมวลผลภาพดิจิตอล.....16

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

3.2 การตรวจวัดการเคลื่อนไหวของร่างกายมนุษย์......................................16

3.2.1 ชนิดของการตรวจวัดตำแหน่ง 16

3.2.2 เปรียบเทียบข้อดีและข้อเสียของวิธีการตรวจวัดตำแหน่ง......................................17

3.3 อัลกอริทึมในการตรวจวัดตำแหน่งขา......................................18

3.3.1 กระบวนการลบภาพพื้นหลัง......................................19

3.3.2 กระบวนการแยกภาพขา......................................19

3.3.3 กระบวนการแยกโครงสร้างขา......................................20

3.3.4 กระบวนการคำนวณหาตำแหน่งส้นเท้า......................................21

3.3.5 กระบวนการคำนวณตำแหน่งของผู้ใช้......................................22

3.4 อัลกอริทึมสำหรับแก้ไขข้อผิดพลาด......................................23

บทที่ 4 การเชื่อมต่อและการสื่อสารระหว่าง Flash กับ Visual C++......................................24

4.1 ความรู้เบื้องต้นเกี่ยวกับ Flash ที่ใช้ใน Visual C++......................................24

4.1.1 ความสำคัญของ Flash ที่ต้องนำมาใช้กับ Visual C++......................................24

4.1.2 การเขียนโปรแกรม และภาษาที่ใช้กับ Flash......................................24

4.1.3 File format ของ Flash ที่ใช้ใน Visual C++......................................24

4.2 การเพิ่ม Component Shockwave Flash Object ของ Flash เข้าไปใน Visual C++.........25

4.3 การเชื่อมต่อระหว่าง Flash กับ Visual C++......................................27

4.4 การติดต่อสื่อสารระหว่าง Flash กับ Visual C++......................................27

4.4.1 การส่งค่าจาก Visual C++ ไปยัง Flash......................................28

4.4.2 การรับค่าจาก Flash ของ Visual C++......................................29

บทที่ 5 การวิเคราะห์และออกแบบระบบ......................................31

5.1 ยูสเคสไดอะแกรม......................................31

5.2 คำอธิบายยูสเคสไดอะแกรม......................................33

5.3 แอคติวิตี้ไดอะแกรม......................................46

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ (ต่อ)

หน้า

บทที่ 6 การออกแบบและพัฒนาเกม..55

6.1 ความต้องการของระบบที่ใช้ในการพัฒนาเกม..55

6.2 รูปแบบและวิธีการเล่นเกม..55

6.3 เนื้อเรื่องของเกม..56

6.4 รายละเอียดของส่วนหน้าจอ..57

6.4.1 หน้าจอเริ่มแรกและหน้าจอแสดงการประมวลผลภาพจากกล้องวิดีโอ.......57

6.4.2 Learning Game..61

6.4.3 Subdue Game..68

บทที่ 7 สรุปผลการวิจัยและข้อเสนอแนะ..74

7.1 สรุปผลการพัฒนาระบบ..74

7.2 ข้อเสนอแนะสำหรับการพัฒนาในอนาคต..75

เอกสารอ้างอิง..77

ประวัติผู้เขียน..78

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่ หน้า

5.1 ยูสเคส Leg Track Image Processing 33
5.1 (ต่อ)..34
5.1 (ต่อ)..35
5.2 ยูสเคส OCX Flash Connect..35
5.3 ยูสเคส Select Games 36
5.4 ยูสเคส How to Play 36
5.5 ยูสเคส ยูสเคส How to Set Camera 37
5.6 ยูสเคส Play Learning Game และ Subdue Game.......................................37
5.6 (ต่อ)..38
5.7 ยูสเคส Learning Character Control 38
5.8 ยูสเคส Learning Q&A Control...39
5.8 (ต่อ)..40
5.9 ยูสเคส Subdue Character Control..41
5.9 (ต่อ)..42
5.10 ยูสเคส Subdue Monster Control 42
5.10 (ต่อ)..43
5.11 ยูสเคส Win & Lose...44
5.11 (ต่อ)..45

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่ **หน้า**

2.1 ตัวอย่างจากเกม Backtrack..........5
2.2 ตัวอย่างจากเกม Dragon Ball..........6
2.3 ตัวอย่างจากเกม Castlevania..........6
2.4 ตัวอย่างจากเกม Super Robot Taisen..........7
2.5 แสดงความแตกต่างของระบบ Screen Coordinate กับ Cartesian Coordinate..........10
3.1 โครงสร้างและขั้นตอนของอัลกอริทึมในการตรวจวัดตำแหน่งขา..........12
4.1 File format ของ Flash และการใช้งานบน Web browser และ Visual C++..........25
4.2 วิธีการเพิ่ม Component ต่างๆ เข้าสู่ Project ใน Visual C++..........25
4.3 การเพิ่ม Component Shockwave Flash Object ของ Flash เข้าไปใน Visual C++..........26
4.4 เครื่องมือ Shockwave Flash Object ที่เพิ่มเข้ามาให้นำไปใช้งาน..........26
4.5 การเชื่อมต่อระหว่าง Flash กับ Visual C++..........27
4.6 การส่งค่าจาก Visual C++ ไปยัง Flash ด้วยฟังก์ชั่น SetVariable..........28
4.7 การรับค่าจาก Flash ของ Visual C++ ด้วยฟังก์ชั่น GetVariable..........30
5.1 Use Case Diagram ของเกมปราบปรามยาเสพติด..........31
5.2 แอคทิวิตี้ไดอะแกรม Leg Track Image Processing..........46
5.3 แอคทิวิตี้ไดอะแกรม OCX Flash Connect..........47
5.4 แอคทิวิตี้ไดอะแกรม Select Game..........47
5.5 แอคทิวิตี้ไดอะแกรม How to Play..........48
5.6 แอคทิวิตี้ไดอะแกรม How to Set Camera..........48
5.7 แอคติวิตี้ไดอะแกรม Learning Game And Subdue Game..........49
5.8 แอคติวิตี้ไดอะแกรม Learning Character Control..........50
5.9 แอคติวิตี้ไดอะแกรม Learning Q&A Control..........51
5.10 แอคทิวิตี้ไดอะแกรม Subdue Character Control..........52
5.11 แอคทิวิตี้ไดอะแกรม Subdue Monster Control..........53
5.12 แอคติวิตี้ไดอะแกรม Win & Lose..........54
6.1 แสดงหน้าจอเริ่มแรกและหน้าจอแสดงการประมวลผลภาพจากกล้องวิดีโอ..........57
6.2 แสดงหน้าหลักของเกม (Main Menu)..........58

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป(ต่อ)

รูปที่ หน้า

6.3 แสดงภาพการเลือก How to Play....58
6.4 แสดงหน้าจอ How to Play ของ Learning Game....59
6.5 แสดงหน้าจอ How to Play ของ Subdue Game....59
6.6 แสดงภาพการเลือก How to Set Camera....60
6.7 แสดงภาพ How to Set Camera....60
6.8 แสดงการเลือกเล่นเกม Learning Game จากหน้าเลือกเกม....61
6.9 แสดงการเหยียบปุ่ม "เริ่มเกม" เพื่อให้เกม Learning Game เริ่มทำงาน....61
6.10 แสดงการถามคำถามและคำตอบของเกม Learning Game....62
6.11 แสดงการเหยียบเลือกคำตอบ และแสดงปุ่ม ตกลง เพื่อยืนยันข้อที่เลือก....63
6.12 แสดงการเฉลยคำตอบที่ถูกต้อง....63
6.13 แสดง Time Out เมื่อเวลาในการตอบคำถามของแต่ละข้อหมด....64
6.14 แสดงการชนะ โดยจะขึ้นข้อความ Success....64
6.15 แสดงการชนะ โดยจะขึ้นข้อความ You Win เมื่อชนะในด่านสุดท้าย....65
6.16 แสดงการแพ้ โดยจะขึ้นข้อความ You Lose....65
6.17 แสดงหน้าจอตัวเลือกหลังพ่ายแพ้....66
6.18 แสดงหน้าจอของฉากที่ 2 และแสดงมอนสเตอร์เจ้าของด่าน....66
6.19 แสดงหน้าจอของฉากที่ 3 และแสดงมอนสเตอร์เจ้าของด่าน....67
6.20 แสดงหน้าจอของฉากที่ 4 และแสดงมอนสเตอร์เจ้าของด่าน....67
6.21 แสดงการเลือกเล่นเกม Subdue Game....68
6.22 แสดงการเหยียบปุ่ม เริ่มเกม เพื่อให้เกม Subdue Game เริ่มทำงาน....69
6.23 แสดงฉากที่ 1 ซึ่งจะมีปิศาจยาเสพติดออกมาคราวละตัว....70
6.24 แสดงฉากที่ 2 จะมีปิศาจยาเสพติดออกมาคราวละ 2 ตัว....70
6.25 แสดงฉากที่ 3 จะมีปิศาจยาเสพติดออกมาคราวละ 3 ตัว....71
6.26 แสดงฉากที่ 4 จะมีปิศาจยาเสพติดออกมาคราวละ 4 ตัวเลย....71
6.27 แสดงข้อความ Success เมื่อต่อสู้ชนะในด่านแต่ละด่าน....72
6.28 แสดงข้อความ You Win เมื่อต่อสู้ชนะในด่านสุดท้าย....72
6.29 รูปแสดงการแพ้....73
6.30 แสดงหน้าจอตัวเลือกหลังพ่ายแพ้....73

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบัน โลกได้ก้าวเข้าสู่ยุคแห่งเครื่องมืออีเล็กทรอนิคและเทคโนโลยีต่างๆ มากมาย โดยเฉพาะอย่างยิ่ง ก็คือ กล้อง Video Web Cam ซึ่งจะเห็นได้ว่า มีการนำมาใช้งานกันอย่างแพร่หลายเพิ่มมากขึ้น ไม่ว่าจะเป็นร้านที่ให้บริการอินเตอร์เน็ต หรือบ้านเรือนประชาชนทั่วไปก็ตาม ดังนั้นจะเห็นได้ว่ากล้อง Video Web Cam นั้นจึงแทบจะเป็นอุปกรณ์ต่อพ่วงคอมพิวเตอร์ที่จะขาดเสียไม่ได้แล้วในโลกยุคปัจจุบันนี้ ทั้งนี้ก็เนื่องมาจากราคาของกล้อง Video Web Cam ที่มีราคาถูกลงอย่างมาก อีกทั้งยังมีประสิทธิภาพที่สูงมากขึ้นอีกด้วย และด้วยเหตุนี้นี้เอง จึงทำให้เกิดแนวความคิดที่จะนำกล้อง Video Web Cam เข้ามามีส่วนร่วมในการควบคุมการทำงาน เพื่อเพิ่มความสะดวกสบายและก่อให้เกิดประโยชน์ในการใช้งานมากยิ่งขึ้น

และเนื่องจาก ปัจจุบันนี้เทคโนโลยีของเกมคอมพิวเตอร์กำลังได้รับความนิยมอย่างสูง และเข้ามามีบทบาทอย่างมากต่อเยาวชนไทยและเยาวชนทั่วโลก ซึ่งเกมโดยส่วนใหญ่ ก็จะใช้วิธีการนั่งเล่นและควบคุมผ่านเมาส์และคีย์บอร์ดซึ่งหากนั่งนานเกินไปก็จะมีผลเสียต่อสุขภาพ ด้วยเหตุนี้จึงทำให้เกิดแนวความคิดที่จะสร้างเกมที่จะทำให้เยาวชนได้ลุกขึ้นมาออกกำลังกาย เพื่อเสริมสร้างสุขภาพและพัฒนาการที่ดีของเยาวชน และเนื่องจาก Flash เป็นโปรแกรมที่มีประสิทธิภาพสูง แล้วสามารถสร้างเกมได้ในขนาดไฟล์ที่เล็ก อีกทั้งกำลังเป็นที่สนใจอย่างมากในปัจจุบัน ด้วยเหตุนี้จึงทำให้สามารถเผยแพร่ได้ง่ายและสามารถเปิดให้บริการได้ผ่านทางเว็ปไซต์ทั่วไป

จากเหตุผลทั้งหมดที่กล่าวมา จึงได้เกิดแนวความคิดที่จะพัฒนาเกมคอมพิวเตอร์ที่สร้างจากโปรแกรม Flash และภาษา ActionScript แล้วสามารถควบคุมการเล่นผ่านทางกล้อง Video Web Cam ในรูปแบบ Real-time ได้ เพื่อให้เกิดเป็นเกมคอมพิวเตอร์แนวใหม่ ที่ใช้การตรวจจับตำแหน่งของผู้เล่น โดยอาศัยการประมวลผลภาพที่ได้จากกล้อง Video Web Cam

1.2 ความมุ่งหมายและวัตถุประสงค์ของโครงการ

1. เพื่อศึกษาหลักการและกระบวนการพัฒนาเกมคอมพิวเตอร์ด้วยโปรแกรม Flash และภาษา ActionScript
2. เพื่อศึกษาหลักการและกระบวนการประมวลผลภาพจากกล้องวีดีโอ
3. เพื่อศึกษาหลักการและวิธีการตรวจจับตำแหน่งผู้เล่นจากกล้อง Video Web Cam

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. เพื่อให้สามารถประยุกต์ใช้หลักการประมวลผลภาพจากกล้องวีดีโอกับเกมคอมพิวเตอร์ที่พัฒนาโดยโปรแกรม Flash และภาษา ActionScript ได้

5. เพื่อให้สามารถสร้างการเชื่อมต่อ และใช้งานกล้องวีดีโอ กับเกมคอมพิวเตอร์ที่พัฒนาโดยโปรแกรม Flash และภาษา ActionScript ได้

6. เพื่อให้สามารถควบคุมการทำงานผ่านทางกล้องวีดีโอในรูปแบบ Real-time ได้

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1. ได้รับความรู้ความเข้าใจและเล็งเห็นถึงพิษภัยของยาเสพติดผ่านทางการเล่นเกม Subdue Narcotic หรือเกมปราบปรามยาเสพติด

2. เกิดสติปัญญาพร้อมกับมีร่างกายที่แข็งแรงควบคู่กัน หลังจากเล่นเกม Subdue Narcotic เพราะจะต้องใช้การขยับทั้งร่างกายในการเล่นเกม Subdue Narcotic นี้

3. เป็นส่วนหนึ่งของกิจกรรมรณรงค์ต่อต้านยาเสพติดในอนาคต เพื่อดึงดูดให้เด็กๆ หันมาสนใจกับปัญหายาเสพติดมากขึ้น

4. สามารถนำเอาเกม Subdue Narcotic มาใช้เป็นตัวอย่างในการพัฒนาเกมคอมพิวเตอร์สื่อประสมที่นำเอาการประมวลผลภาพจากกล้องวิดีโอราคาถูกมาใช้ควบคุมการเล่นเกม

5. สามารถนำเกมหรือโปรแกรมที่พัฒนาขึ้นจากโปรแกรม Flash มาเชื่อมต่อและใช้งานร่วมกับโปรแกรม Visual C++ ได้ เพื่อเพิ่มขีดความสามารถ และนำไปประยุกต์ใช้งานที่หลากหลายมากขึ้นในอนาคต

1.4 ขอบเขตของการศึกษา

1. ใช้กล้องเวบแคมจำนวน 1 ตัว ที่มีการเชื่อมต่อแบบ USB

2. ขนาดของพื้นที่ที่สามารถทำงานได้มีความกว้างไม่เกิน 2x2 เมตรและมีผู้ใช้เพียง 1 คนอยู่ในพื้นที่ที่กำหนด

3. พื้นที่ที่ใช้ทำการทดลองมีการควบคุมสี และการสะท้อนแสง

4. พัฒนาเกมส์คอมพิวเตอร์ ที่นำวิธีการตรวจวัดตำแหน่งของผู้ใช้มาใช้งาน และสามารถทำงานได้แบบเวลาจริง (Real time)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.5 ขั้นตอนและวิธีการดำเนินโครงการ

1. ศึกษาหลักการและกระบวนการในการพัฒนาเกมส์คอมพิวเตอร์
2. ศึกษาภาษา และกระบวนการที่ใช้ในการพัฒนาเกมส์คอมพิวเตอร์ ที่สร้างจากโปรแกรม Macromedia Flash
3. ออกแบบเนื้อเรื่อง ตัวละคร และฉากภายในเกมส์คอมพิวเตอร์
4. ดำเนินการพัฒนาเกมส์คอมพิวเตอร์ โดยใช้โปรแกรม Macromedia Flash
5. ศึกษาหลักการและกระบวนการในการประมวลผลภาพผ่านกล้องวีดีโอ
6. ศึกษาหลักการและอัลกอริทึมที่ใช้ในการประมวลผลภาพ
7. ศึกษาวิธีการเชื่อมต่อระหว่างโปรแกรม Visual C++ กับโปรแกรม Macromedia Flash
8. ดำเนินการเชื่อมต่อระหว่างโปรแกรม Visual C++ กับโปรแกรม Macromedia Flash โดยนำค่า x,y จากโปรแกรม Visual C++ มาใช้ในการเล่นเกมส์ ที่ถูกสร้างจากโปรแกรม Macromedia Flash
9. ทดลองและทดสอบเพื่อแก้ไขข้อผิดพลาดในระหว่างการเล่นเกมส์คอมพิวเตอร์
10. สรุปผลการดำเนินงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2
หลักและเทคนิคพื้นฐานที่ใช้ในการพัฒนาเกม

2.1 หลักการพื้นฐานของเกมโดยทั่วไป

2.1.1 ประเภทของเกม

เกม (Games) คือ โปรแกรมรูปแบบหนึ่งที่มีจุดประสงค์เพื่อดึงดูดผู้ใช้หรือผู้เล่นให้ตอบสนองต่อเหตุการณ์ต่างๆ อาจจะเป็นการแก้ปัญหาหรือการดำเนินเนื้อเรื่องไปตามแต่ละเกม หรืออาจจะเรียกได้ว่าเกมก็คือ รูปแบบของการพัฒนาโปรแกรมอีกประเภทหนึ่ง ที่มีจุดประสงค์หลักเพื่อให้การบันเทิงแก่ผู้ใช้หรือผู้เล่น โดยตัวเกมนั้นจะใช้ทั้งความสร้างสรรค์ ความทันสมัย และระบบประสาทสัมผัสของมนุษย์ ไม่ว่าจะเป็นภาพ เสียง และการเคลื่อนไหวต่างๆ เพื่อดึงดูดใจและเพื่อทำให้น่าสนใจมากที่สุด

ประเภทของเกมนั้นจะแบ่งออกตามลักษณะเด่นของเกม ไม่ว่าจะเป็นการดำเนินเรื่อง หรือวิธีการเล่นเป็นตัวกำหนด และส่วนใหญ่นิยมแบ่งได้เป็นประเภทต่างๆดังนี้

2.1.1.1 Action

เป็นเกมที่อาศัยศักยภาพของร่างกายสูง ไม่ว่าจะเป็นสายตา การกะจังหวะ เพื่อตอบสนองต่อเหตุการณ์ของเกม ซึ่งต้องอาศัยความรวดเร็วและความถูกต้อง โดยส่วนใหญ่เกมแนวนี้จะมีลักษณะเด่นคือ ให้ผู้เล่นกระทำเหตุการณ์เพื่อผ่านฉากหรือภารกิจ และดำเนินเนื้อเรื่องไปให้ถึงจุดหมายของเกม หรืออาจจะเป็นการเก็บคะแนนให้ได้มากที่สุด ตัวอย่างของเกมประเภทนี้ก็ได้แก่ Mario, Far Cry และ Crysis

2.1.1.2 Adventure

เกมประเภทนี้จะใช้เนื้อเรื่องเป็นตัวกำหนดทิศทางของเกม การแสดงออกของตัวละครในเกมก็จะถูกสร้างให้ซับซ้อนกว่าเกมประเภท Action มาก และการแสดงออกของตัวละครก็จะมีผลต่อการผ่านภารกิจหรือฉากสูง ตัวอย่างของเกมประเภทนี้ได้แก่ Beyond Good & Evil, Zelda และ Prince of Persia

2.1.1.3 Sport

เป็นเกมที่จำลองกีฬาชนิดต่างๆมาเป็นเกม มีลักษณะการเล่นที่เร้าใจ สนุกสนาน และสามารถเล่นได้ทุกเพศ ทุกวัย ตัวอย่างเกมส์ในประเภทนี้ได้แก่ Winning Eleven (เกมฟุตบอล) และ Need For Speed (เกมขับรถ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.1.4 Puzzle

เกมประเภทนี้ถูกเรียกอย่างง่ายๆว่า เกมลับสมอง เป็นเกมที่ต้องใช้ความคิดมากกว่าศักยภาพทางร่างกาย เพราะส่วนใหญ่เป็นเกมที่ต้องใช้การแก้ไขปัญหาในการผ่านด่าน ตัวอย่างของเกมในประเภทนี้ได้แก่ Tetris, Puyo Puyo และ เกมกระดานต่างๆ

2.1.1.5 Strategy

เป็นเกมที่ต้องใช้ความคิด เพื่อวางแผนและควบคุมสิ่งต่างๆในเกม ไม่ว่าจะเป็นกำลังพล ทรัพยากรต่างๆ ส่วนใหญ่ภารกิจของเกมประเภทนี้จะแบ่งได้ 2 ชนิด ได้แก่การสร้างและพัฒนา กับการสร้างเพื่อทำลาย (ในรูปแบบของสงคราม) ตัวอย่างของเกมประเภทนี้ได้แก่ Sim City, Red Alert และ Age of Empire

2.1.1.6 Role-Playing Game (RPG)

หรือก็คือเกมจำลองเหตุการณ์นั่นเอง ตัวละครในเกมจะถูกสมมติให้เป็นตัวของผู้เล่น ผู้เล่นมีหน้าที่พัฒนาตัวละครในเกมในทุกๆด้าน และตัวละครที่เราบังคับก็จะตอบสนองกับเหตุการณ์ต่างๆที่เกิดขึ้นในโลกของเกม ผลลัพธ์ของการดำเนินเรื่องราวจะแตกต่างกันไป ตามแต่ตัวเราจะเป็นผู้เลือกเอง ตัวอย่างเกมในประเภทนี้ได้แก่ Final Fantasy, Dragon Quest

2.1.2 มุมมองภายในเกม

เกมแต่ละเกมก็จะแตกต่างกันออกไป และสิ่งที่แตกต่างเป็นอย่างมากนั่นก็คือมุมมองภายในเกม มุมมองภายในเกมมีการพัฒนาไปมากกว่าอดีตมากมาย ภายในอดีตนั้น มุมมองที่แพร่หลายและเหมาะกับทรัพยากรที่มีมากที่สุดก็คือ Bird Eye View มุมมองที่นิยมนำมาใช้เพื่อถ่ายทอดภาพสู่สายตาของผู้เล่นมีหลายรูปแบบดังนี้

2.1.2.1 First Person

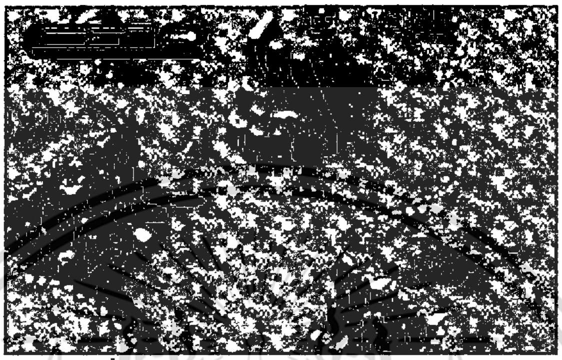
เป็นมุมมองที่นิยมนำไปใช้กับเกมประเภท จำลองการยิง โดยจะใช้วิธีถ่ายทอดสายตาของตัวละครที่เล่นออกมาทางหน้าจอ ปัจจุบันมุมมองแบบนี้เป็นที่นิยมมาก ไม่ว่าจะเป็นเครื่องเล่นตามบ้าน (Console) หรือเครื่องเล่น PC ที่นิยมเล่นมุมมองนี้มานานมากแล้ว เกมที่แนะนำเช่น Half Life, Far Cry และ Crysis

รูปที่ 2.1 ตัวอย่างจากเกม Backtrack

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.2 Isometric

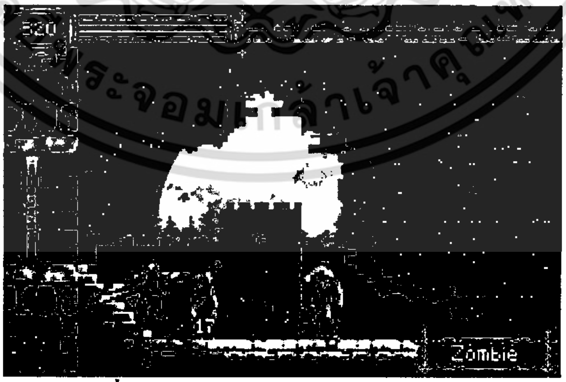
จะแสดงภาพเหมือนเรามองเฉียงลงมาเบื้องล่าง เป็นมุมมองแบบ 3 มิติที่นิยมนำมาใช้กันมากที่สุด เพราะสามารถสร้างเกมแบบ 3 มิติ โดยไม่ต้องใช้เทคนิคการเขียนโปรแกรมแบบ 3 มิติขั้นสูง เกมส์ในยุคแรกที่สร้างชื่อเสียงให้มุมมองแบบนี้ก็คือ Diablo และเกมมากมายในเครื่องยุคใหม่ เช่น Zelda, Final Fantasy และ Tomb Raider



รูปที่ 2.2 ตัวอย่างจากเกม Dragon Ball

2.1.2.3 Side

เป็นมุมมองที่นิยมใช้ในการพัฒนาเกมแบบ 2 มิติ โดยจะถ่ายทอดภาพจากมุมมองบุคคลที่ 3 (มองเห็นตัวละครชัดเจน) เกมในรูปแบบนี้จะมีอยู่แต่เฉพาะเครื่องเล่นเกมที่มีคุณภาพไม่สูงเท่านั้น แต่ถ้ามองย้อนกลับไปในอดีต มุมมองแบบนี้ถือเป็นมุมมองที่สามารถแสดงศักยภาพของเกมได้เป็นอย่างดี ตัวอย่างของเกมประเภทนี้ก็คือ Mario, Sonic และ Contra

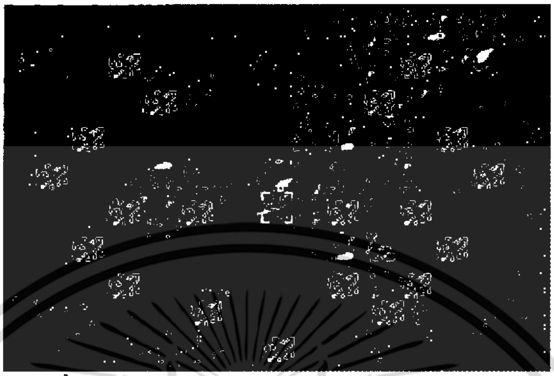


รูปที่ 2.3 ตัวอย่างจากเกม Castlevania

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2.4 Top Down

เป็นมุมมองแบบมองลงมาตรงๆ นิยมใช้ในเกมที่มีรูปแบบง่ายๆไม่ซับซ้อน หรือที่เรียกมุมมองแบบนี้ว่า Bird Eye View ตัวอย่างของเกมที่ใช้มุมมองประเภทนี้ก็คือ Pac Man, Battle Tank และ เกมในยุคเริ่มแรกทั้งหลาย



รูปที่ 2.4 ตัวอย่างจากเกม Super Robot Taisen

2.1.3 สิ่งสำคัญที่ต้องคำนึงถึงในการออกแบบเกม

เกมที่ได้รับความนิยมมักขึ้นอยู่กับองค์ประกอบต่างๆภายในเกม เช่น เนื้อหา จุดประสงค์ และกลุ่มผู้เล่นเป้าหมาย เพราะฉะนั้นก่อนจะเริ่มต้นพัฒนาเกมใดๆนั้น จึงควรที่จะวางแผนและพิจารณาองค์ประกอบต่างๆภายในเกมให้ดีเสียก่อน โดยมีหลักสำคัญในการพิจารณาดังต่อไปนี้

2.1.3.1 กลุ่มอายุและเพศของผู้เล่นที่เป็นเป้าหมาย

สิ่งสำคัญอันดับแรกเลยก็คือกลุ่มผู้เล่นที่เป็นเป้าหมาย เราจำเป็นจะต้องนำอายุและเพศของกลุ่มผู้เล่นที่เป็นเป้าหมายมาพิจารณาก่อนที่จะออกแบบเกมส์เสมอ อย่างเช่นถ้าเป็นกลุ่มผู้เล่นที่เป็นเด็ก ก็ควรจะต้องออกแบบเกมให้เข้าใจง่ายไม่ซับซ้อน อย่างเด็กเล็กก็ควรใช้ภาพเป็นการ์ตูนเพื่อจะได้ดึงดูดความสนใจจากเด็กๆได้ง่าย และถ้าจะออกแบบเกมไว้บน Web ก็ควรศึกษาอายุและเพศของผู้เยี่ยมชมส่วนใหญ่ก่อนว่าอยู่ในกลุ่มใด เพื่อที่จะสามารถออกแบบเกมได้ตรงกับกลุ่มคนที่เข้ามาเยี่ยมชม Web นั้นๆได้มากที่สุด

2.1.3.2 ประสิทธิภาพของคอมพิวเตอร์ที่จะใช้เล่นเกม

อีกปัจจัยที่ไม่สามารถละเลยได้เลย ถ้าเราคิดที่จะออกแบบเกมสักเกมหนึ่ง เพราะประสิทธิภาพของคอมพิวเตอร์ มีส่วนสำคัญในการประมวลผลภาพ และการเคลื่อนไหวต่างๆของเกมให้ดูดี นุ่มนวลและเสมือนจริงมากขึ้น โดยในหนึ่งเกมจะมีการประมวลผลสิ่งต่างๆภายในเกมมากมาย เพราะฉะนั้นจะต้องใช้ทรัพยากรของเครื่องมากพอดู ถ้าประสิทธิภาพของเครื่องไม่สูงพอ ก็จะทำให้การแสดงผลล่าช้า ขาดความต่อเนื่อง ดังนั้นก่อนจะออกแบบและพัฒนาเกม ควรจะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พิจารณาด้วยว่าเกมดังกล่าวใช้เครื่องคอมพิวเตอร์ที่มีประสิทธิภาพสูงเกินไปหรือไม่ เพราะไม่ว่าเกมจะออกมาดีเพียงใด แต่ถ้ามีคนที่สามารถเล่นได้น้อยมากก็นับว่าไม่เกิดประโยชน์

โดยเฉพาะการทำเกมที่พัฒนาขึ้นจากโปรแกรม Flash จำเป็นที่จะต้องพิจารณาถึงจุดนี้เอาไว้มากๆ เนื่องจากเกมที่พัฒนาขึ้นจากโปรแกรม Flash นั้นควรจะสามารถเล่นได้ทุกเครื่อง แม้จะเป็นเครื่องที่มีประสิทธิภาพต่ำๆก็ตาม แต่ถึงจะเป็นอย่างนั้น เกมที่พัฒนาขึ้นจากโปรแกรม Flash นั้นก็มีหลายจุดที่ถ้าวางแผนพัฒนาไม่ดี อาจทำให้ต้องใช้การประมวลผลคอมพิวเตอร์อย่างหนักเกินความจำเป็น และอาจทำให้เกิดการแสดงผลล่าช้า เครื่องอืด หรือไม่ก็เครื่องแฮงค์ไปเลย ดังนั้นจึงควรหาทางหลีกเลี่ยงเอาไว้ด้วย อย่างเช่น ภาพกราฟฟิคที่ถูกสร้างจาก Flash มักจะเป็นภาพแบบ vector ซึ่งถูกสร้างขึ้นจากการคำนวณทางคณิตศาสตร์ และยิ่งมีการไล่เฉดสีหรือไล่ความโปร่งใส มีการเคลื่อนย้ายตำแหน่ง แล้วถ้าทำพร้อมกันจำนวนมากๆ ก็จะทำให้คอมพิวเตอร์ต้องทำการคำนวณอย่างหนัก และอาจทำให้การแสดงภาพช้าลงอย่างมาก ดังนั้นจึงควรหาทางลดการประมวลผลด้วยการแปลงเป็นภาพแบบ Bitmap ซะก่อน ซึ่งก็จะได้ผลไม่แตกต่างกันมากแต่ช่วยลดการประมวลผลของคอมพิวเตอร์ลงไปอย่างมาก แต่ถ้าใช้ภาพแบบ vector ในจำนวนไม่มาก และ Effect ไม่เยอะก็สามารถใช้ได้ ไม่ส่งผลกระทบเท่าไหร่

2.1.3.3 วัตถุประสงค์และเนื้อหาของเกม

สำหรับการกำหนดวัตถุประสงค์ของเกมให้มีความเหมาะสมกับเกมนั้น จะทำให้เกมมีความน่าสนใจมากยิ่งขึ้น เช่นเกมที่มีการผจญภัยไปเรื่อยๆ เมื่อถึงเส้นชัยก็จะชนะ แต่ถ้าเราใส่อุปสรรคเช่นศัตรู หรือใส่สิ่งของที่ต้องค้นหาก่อนถึงเส้นชัย ก็จะทำให้เกมมีความน่าสนใจมากขึ้น ส่วนเนื้อหาก็เป็นสิ่งสำคัญไม่แพ้กัน ปัจจุบันเราจะเห็นการพัฒนาของเนื้อหาของเกมมากกว่าอดีตมาก อย่างเช่นเกมแข่งรถในอดีตก็จะมีเป้าหมายอยู่ที่การเข้าเส้นชัยเป็นที่ 1 เพียงอย่างเดียว แต่ปัจจุบันใส่ความเป็น Drama เพิ่มเข้าไปอย่างมาก เช่นจะต้องเอาชนะให้ได้เพื่อสร้างชื่อเสียง และนำชื่อเสียงที่มากขึ้นไปแข่งในสนามที่ยากขึ้นและเงินรางวัลสูงขึ้น เป็นต้น

2.1.4 การเพิ่มความน่าสนใจให้กับเกม

เกมที่ดีและน่าสนใจ ล้วนแล้วแต่มีองค์ประกอบมากมายหลายอย่าง เราสามารถแจกแจงสิ่งที่ทำให้เกมน่าสนใจได้ดังต่อไปนี้

2.1.4.1 จำนวนชีวิตและพลังชีวิต (Life and Health)

จำนวนชีวิตและพลังชีวิตของตัวละคร 2 สิ่งนี้เป็นสิ่งที่สร้างให้เกิดความตื่นเต้นและตื่นตัวในเกม เพราะเกมจะเดินหน้าต่อไปหรือกลับมาเริ่มต้นใหม่ก็ด้วย 2 สิ่งนี้ เช่นถ้าพลังชีวิตของเราหมด ก็จะเสียจำนวนชีวิต และเมื่อจำนวนชีวิตเป็น 0 เราจะต้องเริ่มต้นใหม่ตั้งแต่ต้น หรือเริ่มต้นใหม่ตรงจุดที่เราบันทึกเกม เพราะฉะนั้นจำนวนชีวิตและพลังชีวิตจะสามารถใช้ควบคุมระยะเวลาในการเล่นเกมได้ และถ้ามองดูดีๆแล้วทั้ง 2 สิ่งจะมีความสัมพันธ์ไปถึงการออกแบบฉาก

เอกสารนี้เป็นเอกสารที่ใช้งานเพื่อการศึกษาหรือการวิจัยเท่านั้น ไม่อนุญาตให้นำไปใช้เชิงพาณิชย์หรือเพื่อการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เลยทีเดียว เช่นจุด Climax ของฉากที่มีไว้ลดพลังชีวิตของผู้เล่น ตัวหัวหน้าที่ถ้าเล่นไม่ผ่านหรือพลังชีวิตหมดก่อนก็ต้องกลับไปเริ่มต้นใหม่ และเงื่อนไขที่ว่าต้องทำให้ตัวหัวหน้าพลังชีวิตหมดถึงผ่านฉากได้ ก็ล้วนเกี่ยวพันกับการออกแบบพลังชีวิตด้วย

2.1.4.2 คะแนน (Score)

คะแนนเป็นระบบที่มีไว้เพื่อวัดความสามารถและประสิทธิภาพของผู้เล่น อีกทั้งยังสามารถใช้เปรียบเทียบกับผู้เล่นคนอื่นๆได้อีกด้วย ระบบคะแนนจึงเป็นสิ่งที่ทำให้เกมน่าสนใจมากขึ้น ระบบคะแนนยังมีส่วนสำคัญในการเล่นเกมด้วย เช่นถ้าทำคะแนนได้เกินเป้าที่ตั้งไว้ก็จะชนะ หรือเกมจะเริ่มจากง่ายที่สุด และจะเริ่มเพิ่มระดับความยากขึ้นไปเมื่อทำคะแนนได้ตามที่เกมได้ตั้งไว้ เป็นต้น

ปัจจุบันคะแนนยังสามารถนำไปใช้แลกสิ่งจำเป็นในเกมอีกด้วย อย่างเครื่องเล่นเกม XBOX 360 ได้ใช้คะแนนในการเล่น เป็นตัวล่อให้นักเล่นเกมหันมาเล่น Online กันมากขึ้น เพราะสามารถนำคะแนนที่ได้จากการเล่นเกมมาแลกสิ่งของในเกมได้ เช่นอาวุธหรือเครื่องแต่งกาย ยิ่งนำคะแนนมาแลกสิ่งต้องใช้คะแนนสูงมากมาได้ นั่นยิ่งบ่งบอกถึงระดับของนักเล่นเกมเอง

2.1.4.3 จำนวนฉากและระดับความยาก (Stage and Level)

จำนวนฉากและระดับความยาก (Stage and Level) ก็คือ ระบบที่นำมาใช้เพื่อเพิ่มความน่าสนใจและน่าติดตามของเกม โดยเราสามารถนำระบบการเพิ่มฉากและระบบการเพิ่มระดับความยากของเกมมาใช้ควบคุมจำนวนการวนซ้ำในการเล่นเกมได้ เพราะคงไม่มีใครอยากเล่นเกมที่มีอุปสรรคและเป้าหมายเหมือนกันไปทุกฉาก ซึ่งถ้ายิ่งเพิ่มฉากมากขึ้นก็ควรมีระดับความยากเพิ่มมากขึ้นด้วย ความท้าทายในการเล่นเกมจึงจะเกิดขึ้น โดยส่วนมากเกมจะเริ่มตั้งแต่ง่ายที่สุด เพื่อสอนพื้นฐานการเล่นต่างๆให้กับผู้เล่น และจะเริ่มยากขึ้นไปเรื่อยๆตามความสามารถและคะแนนที่ทำได้ของผู้เล่น

ในหนึ่งฉาก เราอาจจะวางแผนให้ผู้เล่นได้พบกับอุปสรรคต่างๆ และทำให้ผู้เล่นได้ใช้ความสามารถที่ได้รับมาใช้อย่างเต็มที่ โดยต้องไม่ง่ายเกินไปและไม่ยากเกินไป ผู้ออกแบบเกมจะต้องลองเล่นดูจนกว่าจะพอใจว่ามันง่ายเกินไปหรือว่ายากจนไม่สามารถผ่านได้หรือไม่ เมื่อทดลองเล่นจนเป็นที่พอใจจึงค่อยปล่อยเกมหรือฉากนั้นๆสู่มือผู้เล่น

2.1.4.4 สิ่งตอบแทนและการเพิ่มพลัง (Bonus and Power Up)

สิ่งตอบแทนและการเพิ่มพลังหรือเพิ่มความสามารถ เป็นสิ่งที่ทำให้เกมมีความสลับซับซ้อนและน่าสนใจมากยิ่งขึ้น อย่างเช่นเรานำ Bonus หรือสิ่งของไปซุกซ่อนไว้ในฉาก ถ้าผู้เล่นสามารถหาได้จนครบ เมื่อจบฉากหรือผ่านภารกิจ ตัวละครก็จะได้รับการ Power Up เพื่อเพิ่มพลังหรือเพิ่มความสามารถของตัวละครให้มากยิ่งขึ้น เป็นต้น ซึ่งนี่ก็เป็นแค่หนึ่งในแนวทางการใช้ Bonus และ Power Up เท่านั้นเอง โดยยังสามารถนำไปใช้ได้อีกหลายหลายรูปแบบ

เอกสารนี้เป็นเอกสารที่จัดทำขึ้นเพื่อการศึกษาเท่านั้น ห้ามนำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.4.5 ระดับของข้อผิดพลาดที่พบ (Bug)

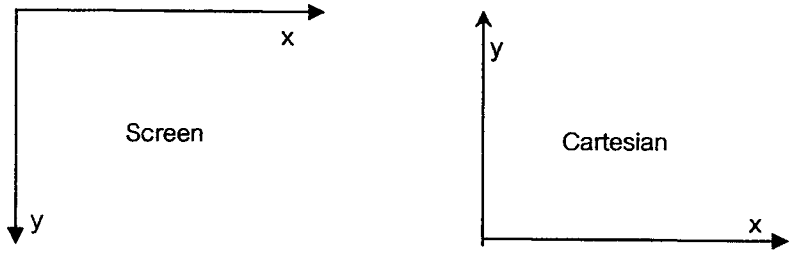
ถึงแม้ว่าจะเป็นเกมที่เก่าที่สุด หรือเกมที่ใหม่ที่สุด ที่ออกในเครื่องยุคใหม่ แต่สิ่งที่ยังคงมีเหมือนๆกันอาจไม่ได้มีแค่ความสนุก แต่กลับกลายเป็นข้อผิดพลาด (Bug) ที่สามารถพบได้ทั่วไปในเกม โดยระดับของความผิดพลาดที่ผู้เล่นพบนั้น จะเป็นสิ่งที่ทำให้ความสนุกในการเล่นเกมลดลงอย่างมาก ยิ่งมีมากขึ้นเท่าไหร่ก็จะยิ่งทำให้ผู้เล่นเสียอารมณ์และเสียความรู้สึกมากขึ้นเท่านั้น เพราะฉะนั้นการพัฒนาเกมใดๆก็ตาม จึงควรที่จะตรวจสอบข้อผิดพลาดที่อาจเกิดขึ้นได้ ให้ครบถ้วนและเรียบร้อยเสียก่อนที่จะนำออกมาให้เล่น

2.2 หลักการและเทคนิคที่ใช้ในการพัฒนาเกมด้วยโปรแกรม Flash

2.2.1 ระบบพิกัดที่ใช้ในโปรแกรม Flash

ระบบพิกัดที่ใช้ในโปรแกรม Flash คือระบบ Cartesian Coordinate System ซึ่งเป็นระบบที่นิยมนำมาใช้ในการอ้างอิงและกำหนดตำแหน่งต่างๆ ของวัตถุ โดยจะให้เส้นในแนวนอนเป็นแกน x และเส้นในแนวตั้งเป็นแกน y และจุดที่เส้นทั้ง 2 ตัดกันจะกำหนดให้เป็นจุด Origin คือ ค่า x เป็น 0 และค่า y เป็น 0 เพื่อเป็นการกำหนดจุดอ้างอิงเริ่มต้นให้กับระบบ โดยพื้นที่ด้านขวาของจุด Origin จะเป็นค่า +x ส่วนพื้นที่ด้านซ้ายของจุด Origin จะเป็นค่า –x และพื้นที่ด้านบนของจุด Origin จะเป็นค่า +y ส่วนพื้นที่ด้านล่างของจุด Origin จะเป็นค่า –y ซึ่งก็จะทำให้สามารถนำไปใช้ระบุตำแหน่งต่างๆบนพื้นที่ของระบบ Cartesian Coordinate System ได้

สำหรับการระบุตำแหน่งของวัตถุบนจอภาพจะใช้หลักการเดียวกับ Cartesian Coordinate System แต่ในการอ้างอิงตำแหน่งบนจอภาพของคอมพิวเตอร์นั้นมีความแตกต่างกับระบบ Cartesian Coordinate System อยู่บ้าง ตรงที่การพิจารณาตำแหน่งของวัตถุบนจอภาพคอมพิวเตอร์ หรือในระบบ Screen Coordinate System จะมองว่าพื้นที่ด้านล่างของจุด Origin เป็นค่า +y และพื้นที่ด้านบนของจุด Origin เป็นค่า –y ซึ่งจะกลับกันกับค่า y ในระบบ Cartesian Coordinate System ส่วนค่า x ยังคงใช้รูปแบบเดียวกันอยู่



รูปที่ 2.6 แสดงความแตกต่างของระบบ Screen Coordinate กับ Cartesian Coordinate

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งในโปรแกรม Flash นั้นจะใช้ระบบพิกัดเดียวกับ Screen Coordinate System โดยให้พื้นที่ซึ่งใช้สำหรับแสดงเนื้อหาต่างๆใน Flash เรียกว่า Stage และถ้าวัตถุใดถูกวางหรือกำหนดตำแหน่งไว้ภายนอก Stage คือ ถูกวางไว้บริเวณที่ค่าของ x หรือ y เป็นลบแสดงว่าวัตถุนั้นจะไม่ถูกแสดงบน Stage เพราะอยู่นอกขอบเขตการแสดงผล ซึ่งต้องระวังให้ดีในเรื่องของการกำหนดตำแหน่งของวัตถุ ไม่เช่นนั้นวัตถุนั้นอาจไม่ถูกแสดง เพราะอยู่นอกขอบเขตการแสดงผลได้

2.2.2 การเขียน ActionScript เพื่อควบคุมการทำงานภายในโปรแกรม Flash

การทำงานของ Movie ใน Flash นั้นจะกระทำแบบต่อเนื่องจากต้นจนจบ เพราะฉะนั้นจึงจำเป็นจะต้องมีการเขียนเหตุการณ์เข้าไปควบคุม ที่เรียกว่าการใช้ ActionScript ซึ่งเป็นภาษาที่ใช้เพื่อควบคุมการทำงานต่างๆภายในโปรแกรม Flash

ActionScript เป็นภาษาแบบ script language มีลักษณะในเชิงวัตถุ (object-oriented programming language) เหมือนกับ JavaScript ซึ่งเป็นมาตรฐานชนิดเดียวกันเรียกว่า ECMA-262 ถูกกำหนดขึ้นโดย European Computers Manufacturers Association (ECMA) ดังนั้นหากผู้ศึกษา ActionScript มีความสามารถในการเขียน JavaScript จะสามารถเข้าใจได้ง่ายขึ้น โดย ActionScript นั้นจะใช้สำหรับกำหนดการทำงานด้านต่างๆ ซึ่งจะเรียกการทำงานนี้ว่า Action โดยการทำงานต่างๆนี้ จะขึ้นอยู่กับเหตุการณ์หรือ Event ที่เกิดขึ้น หากไม่มีเหตุการณ์ก็จะไม่เกิดการทำงาน ฉะนั้นเหตุการณ์ คือสิ่งสำคัญที่ถูกสร้างขึ้น โดยถูกออกแบบให้ตอบสนองการทำงานที่หลากหลาย เช่น การคลิกส์เมาท์ การหยุด การเล่นและอื่นๆอีกมากมาย ซึ่งสามารถควบคุมเหตุการณ์เหล่านี้ได้ด้วย Action เพราะฉะนั้น Action คือ คำสั่งที่ใช้สำหรับกำหนดการทำงานให้กับวัตถุ หรือรูปแบบการกระทำต่างๆ เมื่อมีเหตุการณ์ใดๆเกิดขึ้น โดยจะมี Action ให้เลือกใช้งานหลากหลายรูปแบบ เช่น การหยุด (Stop), การเล่น (Play) หรือ การโหลด (Load) เป็นต้น

ซึ่งการกำหนด ActionScript นั้นจะสามารถระบุได้หลายๆเหตุการณ์ในคราวเดียว หรือในเหตุการณ์อาจจะมีหลายชุดคำสั่งก็ได้ และ สามารถกำหนดหรือเขียน ActionScript ให้กับ

1. Button หรือ ปุ่ม เพื่อเป็นการกำหนดการทำงานให้กับตัวปุ่ม โดยจะเขียน ActionScript ไว้ที่ตัวปุ่มนั้นๆ เลย เช่น เมื่อคลิกเมาส์บนปุ่มแล้วจะเกิดเหตุการณ์อะไร เป็นต้น
2. Movie Clip เพื่อให้สามารถสั่งงาน Movie Clip ได้ตามต้องการ โดยจะเขียน ActionScript ไว้ที่ตัว Movie Clip นั้นๆ เลย เช่น เมื่อ Movie Clip ทำการโหลดขึ้นมาแล้วให้จะให้เกิดเหตุการณ์อะไร เป็นต้น
3. Frame หรือ เฟรม การกำหนดการทำงานให้กับเฟรม จะต้องกำหนดให้เฟรมนั้นมีสภาวะเป็นคีย์เฟรม (Key Frame) หรือคีย์เฟรมว่าง (Blank Key Frame) เสียก่อน เฟรมที่ถูกกำหนดการทำงานหรือถูกเขียน ActionScript ลงไปจะปรากฏสัญลักษณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

a ขึ้นบน Frame ที่วางActionScripไว้ โดยคำสั่งนั้นจะมีผลก็ต่อเมื่อ playhead เล่นมาถึงตำแหน่งนั้นแล้วเท่านั้น หรือก็คือเมื่อ Movie ทำงานมาถึง ณ ตำแหน่งเฟรมที่กำหนดไว้นั้นเอง ซึ่งลักษณะการเขียนเกมที่ใช้ในโปรเจคนี้จะใช้การเขียนการควบคุมจากเฟรม ๆ เดียวเป็นหลัก คือ จะกำหนดไว้ทีเฟรมแรกในแต่ละเกม เพือควบคุมการทำงานของเกมนั้นๆ

2.2.3 การสร้างกลไกการทำซ้ำ (Loop) ของ Flash

คำสั่งที่ใช้ในกลุ่มการทำงานแบบวนซ้ำนี้ มีอยู่ด้วยกันหลายแบบ ซึ่งแต่ละแบบก็มีรูปแบบการใช้งานทีแตกต่างกันออกไป แล้วแต่เราจะเลือกใช้ให้เหมาะสมกับรูปแบบและความต้องการในโปรแกรมนั้นๆ

2.2.3.1 while

โดยรูปแบบคำสั่งของ while จะประกอบด้วยเงื่อนไขภายในวงเล็บ เมื่อเงื่อนไขนั้นเป็นจริง ก็จะไปทำงานตามลำดับของ statement ภายในเครื่องหมาย {} จนหมด และจะวนมาตรวจสอบเงื่อนไขอีกครั้ง ถ้าเงื่อนไขยังคงเป็นจริงอยู่ก็จะวนซ้ำ ทำเช่นนี้ไปเรื่อยๆ และจะหลุดจากการทำงานของ Loop while ก็ต่อเมื่อเงื่อนไขเป็นเท็จ และจะข้ามไปทำ statement ที่อยู่ด้านล่างของเครื่องหมาย } ซึ่งมีรูปแบบคำสั่งดังนี้

```
while (condition) {
        statement;
}
```

2.2.3.2 do...while

สำหรับคำสั่งของ do...while นี้ จะทำคำสั่ง statement ภายในเงื่อนไขก่อน 1 ครั้ง โดยไม่สนใจเงื่อนไข แล้วจึงค่อยมีการตรวจสอบเงื่อนไข และเมื่อเงื่อนไขเป็นจริงอีกจึงค่อยวนไปทำ statement เหล่านั้นใหม่อีกครั้ง และจะวนซ้ำไปเรื่อยๆจนกว่าเงื่อนไขจะเป็นเท็จ ซึ่งมีรูปแบบคำสั่งดังนี้

```
do {
        statement;
} while (condition);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.3.3 for

สำหรับคำสั่ง for นั้นจะทำการกำหนดค่าเริ่มต้น(init) เงื่อนไข(condition) และการเปลี่ยนแปลงค่าเริ่มต้น(next) ซึ่งมักจะใช้การเพิ่มค่าหรือลดค่ามาเป็นตัวกำหนดการวนซ้ำ โดยจะทำการกำหนดกันตั้งแต่บรรทัดแรกของคำสั่งภายในวงเล็บเลย แล้วใช้เครื่องหมาย ; กั้นระหว่างข้อกำหนดทั้ง 3 โดยจะออกจาก loop for เมื่อเงื่อนไขเป็นเท็จ ซึ่งมีรูปแบบคำสั่งดังนี้

```
for ( init ; condition ; next ) {
        statement;
}
```

2.2.3.4 for…in

สำหรับคำสั่ง for…in นั้น จะทำการวนซ้ำโดยไม่มีการกำหนดค่าเริ่มต้น(init) เงื่อนไข(condition) และการเปลี่ยนแปลงค่าเริ่มต้น(next) เหมือนกับคำสั่ง for ที่ผ่านมา เพราะคำสั่ง for…in นั้นสร้างขึ้นมาเพื่อให้ทำงานกับ Object หรือ Array ในการวนซ้ำตามจำนวน Property ของ Object หรือตามจำนวนสมาชิกภายใน Array โดยให้ varIterant เป็นตัวแปรที่เอามารับค่า Property ของ Object หรือสมาชิกภายใน Array ที่กำหนดออกมา แล้วทำตาม statement ที่อยู่ภายใน ซึ่งจะวนซ้ำไปเรื่อยๆ จนกระทั่งเรียก Property ของ Object หรือสมาชิกภายใน Array ครบทุกตัวแล้ว

```
For ( varIterant in object ) {
        statement;
}
```

2.2.3.5 Timeline Loop

สำหรับ Timeline Loop นั้นใช้เพื่อสร้างการวนซ้ำที่มีช่วงเวลาแตกต่างกันออกไป เพื่อให้เห็นถึงความเปลี่ยนแปลงของการทำงานที่เกิดขึ้น ทำได้โดยวางคำสั่งควบคุมการทำงานไว้ในแต่ละ Frame แยกกัน แล้วสั่งให้ไปทำงานยัง Frame นั้นๆ แล้วสั่งวนเรียกกันไปมาจนเกิดเป็น Loop ขึ้น

2.2.3.6 Event Loop

สำหรับ Event Loop เป็นการสร้างกลไกการทำซ้ำด้วย event โดยใช้คำสั่งควบคุมการทำงานของ event คือ enterFrame มาสร้างให้เกิดเป็นลูป ซึ่งจะวนซ้ำตามอัตรา frame rate หรือจำนวนเฟรมต่อวินาทีที่ได้ทำการกำหนดไว้ โดยใช้เฟรมบน Timeline เพียงเฟรมเดียว ซึ่งก็จะให้ผลการทำงานเหมือนกับ Timeline Loop ได้โดยไม่ต้องเพิ่มเฟรม

2.2.3.7 การวนซ้ำด้วย interval

สำหรับการวนซ้ำด้วย interval เป็นการวนซ้ำโดยใช้ฟังก์ชั่น setInterval กำหนดเวลาที่จะให้เกิดการทำงานที่ระบุไว้ทุกๆ รอบเวลาที่กำหนด ซึ่งมีลักษณะการทำงานคล้ายกับ onEnterFrame แต่ interval จะสามารถทำซ้ำโดยกำหนดความถี่ในการทำงานได้เองมีหน่วยเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

มิลลิวินาที นอกจากนี้ยังสามารถยกเลิกการทำงานจาก Actionscript โดยตรงได้อีกด้วย มีรูปแบบการใช้งาน 3 แบบ คือ

1) กำหนดฟังก์ชั่นภายในคำสั่ง setInterval เลย ด้วย

```
setInterval ( function() {statement; }, intervalTime);
```

2) กำหนดฟังก์ชั่นไว้ก่อนแล้วเรียกชื่อฟังก์ชั่นภายในคำสั่ง setInterval ดังนี้

```
function myFunction (arg) {
    statement;
}
setInterval ( myFunction, intervalTime, arg);
```

3) กำหนดเป็นออบเจ็ค และเรียกใช้เมธอดของออบเจ็คนั้นจากคำสั่ง setInterval

```
myObj = new Object();
myObj.myMethod = function (arg) {
    statement;
}
setInterval ( myObj, "myMethod", intervalTime, arg);
```

สำหรับการยกเลิกการทำงานของ interval จะใช้คำสั่ง clearInterval ซึ่งการเรียกนั้นต้องใช้ IntervalID เพื่อระบุ interval ที่จะยกเลิกด้วย โดยค่า IntervalID นั้นจะได้มาตอนกำหนด setInterval โดยเป็นค่าที่จะถูก return ออกมาในตอนที่เรียกใช้ setInterval นั้นๆ ดังนี้

```
IntervalID = setInterval (myFunction, intervalTime);
```

และเมื่อต้องการจะยกเลิกการทำงานของ interval ก็ใช้ตัวแปรที่เก็บค่า IntervalID ไว้มาใช้ระบุให้กับ คำสั่ง clearInterval ดังนี้

```
clearInterval (intervalID);
```

2.2.4 เทคนิคการสร้างงาน 3 มิติใน Flash

เนื่องจาก Flash เป็นโปรแกรมที่ออกแบบมาเพื่อใช้ในการสร้างงาน Animation ในรูปแบบ 2 มิติเป็นหลัก ดังนั้นการทำงานในรูปแบบ 3 มิตินั้นจึงยังไม่สามารถทำได้อย่างสมบูรณ์นัก ซึ่งในขณะนี้ทาง Flash ก็ยังคงสามารถทำได้เพียงแค่ให้ดูเหมือนเป็น 3 มิติเท่านั้น ยังไม่สามารถที่จะ Reader ออกมาในรูปแบบ 3 มิติได้ ดังนั้นการจะทำงานให้ดูเป็น 3 มิติจึงต้องใช้เทคนิคทางด้านภาพเข้ามาช่วย ด้วยการไปสร้างภาพในโปรแกรมสร้างงาน 3 มิติโดยเฉพาะ แล้วนำภาพที่ได้มาใช้ใน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flash หรืออาจจะสร้างภาพในรูปแบบ 2 มิติแล้วทำการใส่แสงเงาให้กับภาพด้วยการไล่ระดับสีด้วยเครื่องมือเกรเดียนต์ ซึ่งวิธีนี้จำเป็นต้องใช้ความสามารถทางศิลปะสูง และถ้าหากทำการสร้างภาพจาก Flash ด้วยวิธีนี้ ภาพที่ได้จะเป็นภาพแบบ Vector ซึ่งใช้การคำนวณทางคณิตศาสตร์ในการแสดงผล ดังนั้นถ้าหากใช้มากเกินไปอาจทำให้เครื่องต้องรับภาระหนักมากเกินไปในการประมวลผล แต่ก็สามารถแก้ได้ด้วยการบันทึกภาพให้เป็นภาพแบบ Bitmap ซะก่อนแล้วค่อยนำมาใช้ ก็จะสามารถช่วยลดการประมวลผลลงไปได้อย่างมากเลยทีเดียว

หลังจากที่ได้ภาพที่ดูเป็น 3 มิติแล้ว จากนั้นจึงต้องนำมากำหนดระดับความลึกหรือความใกล้ไกลของมุมมองภายในงานกันเอาเองอีกที โดยให้ภาพที่อยู่ใกล้ตัวเรามีขนาดใหญ่กว่าภาพที่อยู่ไกลจากตัวเรา ด้วยการเพิ่มหรือลดขนาดไล่กันไปตามระดับความใกล้ไกลของวัตถุที่อยู่ในภาพ โดยกำหนดจากเครื่องมือ Transform หรือใช้คำสั่ง _xscale หรือ _yscale ในการกำหนดขนาดเอาก็ได้

และเนื่องจากรูปแบบการจัดวางวัตถุภายใน Flash จะเป็นแบบลำดับชั้นหรือ Layer จึงทำให้วัตถุที่ถูกวางทีหลังหรือสร้างขึ้นมาทีหลัง อยู่ทับวัตถุที่วางหรือสร้างขึ้นมาอยู่ก่อนแล้ว ดังนั้นจึงต้องมีการกำหนดลำดับชั้นหรือระดับความลึกกันใหม่ ด้วยการใช้งานฟังก์ชั่น swapDepths(); เพื่อใช้สลับค่าความลึก หรือฟังก์ชั่น getNextHighestDepth(); เพื่อให้มาอยู่ระดับบนสุดหรืออยู่หน้าสุดเป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

อัลกอริทึมในการตรวจวัดตำแหน่งขาด้วยกล้องวีดีโอ

3.1 การประมวลผลภาพดิจิตอล (Digital image processing)

การประมวลผลภาพดิจิตอลหมายถึง กระบวนการหรือวิธีการใดๆ ที่กระทำกับภาพดิจิตอล เพื่อที่จะแก้ไข ปรับปรุง หรือวิเคราะห์ภาพดิจิตอล เพื่อทำการปรับปรุงภาพดิจิตอลให้ดีขึ้น หรือแยกส่วนของภาพที่เราต้องการออกจากสภาพแวดล้อมหรือสิ่งที่รบกวนอยู่ ซึ่งกระบวนการนี้จะใช้เครื่องคอมพิวเตอร์และซอฟท์แวร์ที่ประกอบ ด้วยฟังก์ชันทางคณิตศาสตร์ สำหรับดำเนินการกับรูปภาพในลักษณะต่างๆ ดังนี้

3.2 การตรวจวัดการเคลื่อนไหวของร่างกายมนุษย์

ระบบการตรวจวัดการเคลื่อนไหวของมนุษย์ เป็นระบบที่สามารถตรวจวัดตำแหน่งการเคลื่อนไหวของมนุษย์ได้แบบเวลาจริง ในการตรวจวัดอาจจะมีการใช้อุปกรณ์ติดตั้งลงบนตัวของมนุษย์ เช่น อุปกรณ์ตรวจวัดด้วยสนามแม่เหล็กไฟฟ้า หรือตรวจวัดด้วยกล้องวีดีโอ ซึ่งอาจจะเป็นการตรวจวัดทั้งหมดของร่างกาย หรือใช้เฉพาะบางส่วนของร่างกายก็ได้

3.2.1 ชนิดของการตรวจวัดตำแหน่ง

การตรวจวัดการเคลื่อนไหวของมนุษย์นั้นมีอยู่ด้วยกันหลากหลายวิธีการ โดยในปัจจุบันสามารถแบ่งวิธีการที่ใช้ในการตรวจวัดตำแหน่งได้ 3 วิธีด้วยกันคือ

3.2.1.1 Non-vision base วิธีการนี้จะใช้เซ็นเซอร์ติดไว้กับตัวมนุษย์ เพื่อใช้สำหรับรวบรวมข้อมูลตำแหน่งการเคลื่อนไหว หรือการเปลี่ยนแปลงตำแหน่งของร่างกายมนุษย์ โดยสามารถจำแนกเทคโนโลยีที่ใช้ในการตรวจวัดคือ เครื่องกล, การตรวจจับความเฉื่อย, การตรวจจับคลื่นเสียง, คลื่นวิทยุหรือไมโครเวฟ และการตรวจวัดสนามแม่เหล็ก สำหรับเทคโนโลยีต่างๆที่นำมาใช้ในการตรวจวัดนี้ มีทั้งข้อดีและข้อจำกัดแตกต่างกันไป ซึ่งข้อจำกัดนี้จะประกอบไปด้วยข้อกำหนดของรูปแบบการใช้งาน ข้อจำกัดของการตรวจวัด และข้อกำหนดเกี่ยวกับสภาพแวดล้อม จึงทำให้มีการนำเทคโนโลยีนี้ไปใช้งานในสภาวะแวดล้อมที่แตกต่างกันไป และเนื่องจากอุปกรณ์หรือเครื่องมือสำหรับวิธีการนี้มักจะมีราคาแพง ซึ่งไม่เหมาะกับการนำมาใช้ในเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ห้องสมุดคณะเทคโนโลยีสารสนเทศ พระจอมเกล้าลาดกระบัง

3.2.1.2 Vision based with markers วิธีนี้จะใช้อุปกรณ์ทางแสงหรือกล้องวีดีโอร่วมกับการทำเครื่องหมายบนร่างกาย สำหรับเป็นจุดอ้างอิงในการตรวจจับการเคลื่อนไหวของร่างกายมนุษย์ วิธีการนี้สามารถนำไปใช้ในการตรวจจับการเคลื่อนไหวของร่างกายมนุษย์ ซึ่งมีโครงสร้างที่มีความซับซ้อนสูงได้เป็นอย่างดี ทำให้วิธีการตรวจวัดตำแหน่งแบบนี้จึงถูกนำไปใช้ในงานวิจัยทางด้านการแพทย์, วิทยาศาสตร์การกีฬา, ด้านวิศวกรรม และการสร้างภาพอนิเมชันที่ต้องการการเคลื่อนไหวที่มีความสมบูรณ์แบบ

3.2.1.3 Vision based without markers วิธีการนี้จะใช้อุปกรณ์ทางแสงหรือกล้องวีดีโอเป็นเครื่องมือในการตรวจวัดตำแหน่ง โดยจะจับร่างกายของมนุษย์หรือส่วนใดส่วนหนึ่งที่ต้องการ ซึ่งจะขึ้นอยู่กับวิธีของผู้ที่นำวิธีการนี้ไปใช้ว่าต้องการจะตรวจวัดจากส่วนใดของร่างกาย เช่น ศีรษะ, หน้า, ดวงตา, แขน, มือ หรือขา เป็นต้น สำหรับวิธีการนี้ถ้าเป็นการเจาะจงไปยังส่วนใดส่วนหนึ่งของร่างกาย เช่น ดวงตา จะต้องใช้กล้องที่มีความละเอียดสูง และระยะการใช้งานจะไม่สูงมาก เนื่องจากข้อจำกัดดังกล่าว อีกทั้งประสิทธิภาพที่ได้ยังขึ้นอยู่กับเครื่องคอมพิวเตอร์ที่นำมาประมวลผลด้วย (Huiyu and Housheng. 2004 : 3-5)

3.2.2 เปรียบเทียบข้อดีและข้อเสียของวิธีการตรวจวัดตำแหน่ง

การตรวจวัดตำแหน่งโดยวิธีการที่กล่าวมามีทั้งข้อดีและข้อเสียแตกต่างกัน โดยสามารถเปรียบเทียบได้ดังต่อไปนี้

3.2.2.1 ราคา สำหรับวิธีการตรวจวัดตำแหน่งด้วย Non-vision based tracking จะใช้อุปกรณ์ที่มีราคาแพงกว่า วิธีตรวจวัดตำแหน่งแบบ Vision based แต่จะให้คุณภาพของข้อมูลตำแหน่งที่สูงกว่า แต่เนื่องจากราคาที่แพงของการตรวจวัดตำแหน่งแบบ Non-vision based จึงไม่เหมาะสมกับการนำมาใช้งานในการพัฒนาแอปพลิเคชันราคาต่ำ

3.2.2.2 ความสะดวกสบาย สำหรับวิธีการตรวจวัดตำแหน่งด้วย Non-vision bade tracking จะใช้อุปกรณ์มาติดตั้งไว้ที่ร่างกายและพื้นที่ใช้งานก็ถูกจำกัด สำหรับวิธีการแบบ Vision based ร่วมกับมาร์คเกอร์ จำเป็นจะต้องติดตั้งมาร์คเกอร์ไว้กับร่างกาย ซึ่งทำให้มีขั้นตอนในการใช้งานที่ยุ่งยาก ส่วนวิธีการแบบ Vision based แบบไม่ใช้มาร์คเกอร์ จะมีความสะดวกสบายในการติดตั้งและการใช้งาน นอกจากนี้ยังไม่ถูกจำกัดพื้นที่ใช้งานอีกด้วย

3.2.2.3 สัญญาณรบกวน (Noise) เมื่อเปรียบเทียบระหว่างวิธีที่ไม่ใช้กล้องวีดีโอกับวิธีที่ใช้กล้องวีดีโอ สัญญาณรบกวนของวิธีที่ไม่ใช้กล้องวีดีโอ เช่น วิธีที่ใช้สนามแม่เหล็กไฟฟ้าอาจจะเกิดสัญญาณรบกวนจากคลื่นวิทยุ และอุปกรณ์ที่สามารถส่งสัญญาณสนามแม่เหล็กไฟฟ้าได้ สำหรับวิธีที่ใช้กล้องวิดีโอจะมีสัญญาณรบกวนที่เกิดจากปัจจัยหลายอย่าง เช่น แสงสว่าง, ความแตกต่างของสีเป็นต้น ซึ่งถ้ามีการติดตั้งระบบที่ดีก็จะสามารถทำให้สัญญาณรบกวนน้อยลงได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
05975

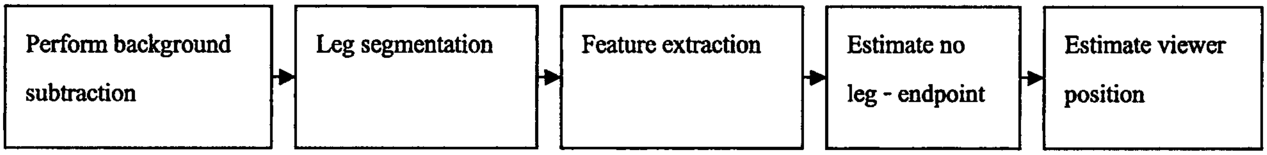
3.2.2.4 ความละเอียด (Accuracy) ความละเอียดของข้อมูลสำหรับการตรวจวัดตำแหน่งด้วยกล้องวีดีโอแบบไม่ใช้มาร์คเกอร์ อาจจะให้ความละเอียดของข้อมูลต่ำ เมื่อเทียบกับการตรวจวัดตำแหน่งแบบไม่ใช้กล้องวีดีโอ แต่ก็มีความเหมาะสมเพียงพอกับแอปพริเคชันราคาต่ำ เช่น เกมที่ไม่ต้องการความละเอียดและความถูกต้องของข้อมูลสูงมาก

3.3 อัลกอริทึมในการตรวจวัดตำแหน่งขา

จากหัวข้อที่ผ่านมา จะเห็นได้ว่าวิธีการตรวจวัดตำแหน่งของผู้ใช้ด้วยกล้องวีดีโอแบบไม่ใช้มาร์คเกอร์ เป็นอีกวิธีหนึ่งที่มักถูกนำมาใช้ในการตรวจวัดตำแหน่งของผู้ใช้ เนื่องจากสะดวกในการติดตั้งและสะดวกต่อผู้ใช้งาน โดยเฉพาะเรื่องของราคาและความซับซ้อนของระบบ ก็เป็นอีกปัจจัยที่มีความสำคัญอย่างมาก ดังนั้นวิธีการที่ใช้ในการคำนวณหาตำแหน่งจุดยืนของผู้ใช้ในโปรเจคนี้นั้นจึงควรใช้วิธีการที่ง่าย รวดเร็วและใช้อุปกรณ์ราคาต่ำ ซึ่งในการคำนวณหาตำแหน่งดังกล่าวนี้นั้น เราจะทำการแบ่งส่วนของร่างกายมนุษย์ออกเป็น 5 ส่วนด้วยกันคือ ศีรษะ ลำตัว ขา และเท้า แล้วเราจะอาศัยเพียงกล้องวีดีโอ Web Cam ราคาถูก 1 ตัว ทำการจับภาพโดยเจาะจงไปที่ส่วนใดส่วนหนึ่งของร่างกายมนุษย์ ซึ่งก็คือ ส่วนขาและเท้าของผู้ใช้ โดยไม่ต้องอาศัยมาร์คเกอร์ หลังจากนั้นจึงนำไปใช้หาตำแหน่งจุดยืนในระนาบของพื้นต่อไป ซึ่งการจับภาพร่างกายท่อนล่างของผู้ใช้นั้น มีข้อดีหลายประการ ได้แก่

- สามารถควบคุมภาพพื้นหลังได้ง่ายเนื่องจากพื้นที่ผู้ใช้งานอยู่จะไม่มีการเปลี่ยนแปลงของพื้น เช่น การเดินผ่านของบุคคลอื่น เป็นต้น
- ง่ายต่อการคำนวณค่าตำแหน่งด้วย Homography
- ผู้ใช้งานรู้สึกมีความเป็นส่วนตัวมากกว่า การตรวจวัดตำแหน่งที่ใช้กล้องจับภาพที่ห้าหรือทั้งร่างกายของผู้ใช้งาน

สำหรับขั้นตอนการทำงานของอัลกอริทึมที่ใช้ในการตรวจวัดตำแหน่งของขานั้น มีดังต่อไปนี้



รูปที่ 3.1 โครงสร้างและขั้นตอนของอัลกอริทึมในการตรวจวัดตำแหน่งขา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.1 กระบวนการลบภาพพื้นหลัง (Perform background subtraction)

กระบวนการลบภาพพื้นหลัง ก็คือ วิธีการที่ใช้ในการลบภาพพื้นหลังหรือสภาพแวดล้อมที่เราไม่ต้องการออกจากส่วนของภาพที่เราต้องการ โดยมีจุดประสงค์เพื่อแยกเอาเฉพาะส่วนของภาพที่เราต้องการออกมาใช้งาน ซึ่งมีขั้นตอนการทำงานดังต่อนี้

1) ทำการเก็บภาพพื้นหลังหรือสภาพแวดล้อมที่ไม่ต้องการเอาไว้ก่อน จากนั้นจึงนำภาพพื้นหลังหรือสภาพแวดล้อมที่ได้มาเปลี่ยนเป็นภาพแบบ Grayscale
2) นำภาพหรือทำการเก็บภาพที่ต้องการลบภาพพื้นหลังออก มาทำการเปลี่ยนให้เป็นภาพแบบ Grayscale
3) นำภาพแบบ Grayscale ทั้ง 2 ภาพที่ได้ ไปผ่านกระบวนการ Background Subtraction
4) ซึ่งจะได้ออกมาเฉพาะส่วนของภาพที่เราต้องการ โดยทำการลบภาพพื้นหลังหรือสภาพแวดล้อมที่เราไม่ต้องการออกจากภาพเรียบร้อยแล้ว

โดยที่กระบวนการ Background Subtraction สามารถคำนวณได้จากสมการต่อไปนี้

$$B(x,y) = f_{bw}\,|I_1(x,y) - I_2(x,y)| \tag{3.1}$$

เมื่อ $I_1(x,y)$ คือ ภาพพื้นหลังหรือสภาพแวดล้อมที่ไม่ต้องการ

$I_2(x,y)$ คือ ภาพที่ต้องการลบภาพพื้นหลังหรือสภาพแวดล้อมที่ไม่ต้องการออก

$B(x,y)$ คือ ภาพที่ได้จากวิธีการ Background subtraction

3.3.2 กระบวนการแยกภาพขา (Leg segmentation)

กระบวนการแยกภาพขา คือ กระบวนการที่ใช้ในการแยกแยะว่าส่วนไหนคือส่วนที่เป็นขา โดยทำการแยกแยะจากภาพที่ได้มาจากกระบวนการลบภาพพื้นหลัง (Background subtraction) ในกระบวนการที่แล้ว ซึ่งมีขั้นตอนการทำงานดังต่อนี้

1) ทำการค้นหาส่วนที่เป็นขาของผู้ใช้จากภาพที่ได้มาจากกระบวนการลบภาพพื้นหลัง (Background subtraction) ในกระบวนการที่แล้ว ซึ่งโดยปกติเวลาที่มนุษย์ยืนตรง ขาก็จะมีลักษณะเป็นเส้นตรงด้วย
2) นำภาพขาของผู้ใช้มาคำนวณหาจุดกึ่งกลางของภาพตามแนวนอนในแต่ละแถวด้วยวิธี Center of mass
3) แล้วนำไปวาดเป็นกราฟ ก็จะได้ตำแหน่งที่ผู้ใช้ยืนอยู่ในภาพออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ซึ่งวิธีการ Center of mass สามารถคำนวณได้จากสมการที่ 3.13

$$r_x = \frac{\sum_{y=1}^{N} B(x, y).y}{\sum_{y=1}^{N} B(x, y)} \quad (3.2)$$

เมื่อ r_x คือ จุดกึ่งกลางของภาพตามแนวนอนที่คำนวณได้

3.3.3 กระบวนการแยกโครงสร้างขา (Leg skeleton extraction)

กระบวนการแยกโครงสร้างขา คือ กระบวนการที่จะทำการตัดสิ่งที่ไม่ต้องการในภาพออกให้เหลือเฉพาะโครงสร้างของขาเท่านั้น เพราะภาพที่ได้จากกระบวนการลบภาพพื้นหลังอาจจะมีบางส่วนในภาพที่ไม่ต้องการติดมาด้วย อย่างเช่น เงา ซึ่งมีหลักการทำงานดังต่อนี้

1) หาค่า Threshold เพื่อจะนำมาใช้สำหรับเลือกจุดที่เหมาะสม จากกระบวนการแยกภาพขา (Leg segmentation) ซึ่งสามารถหาได้จากสมการข้างล่างนี้

$$S_x = \sum_{y=1}^{N} B(x, y) \quad (3.3)$$

เมื่อ S_x คือ ผลรวมของค่า Pixel ทั้งหมดตามแนวนอน ณ แถวที่ x

$$Th_l = \sum_{x=1}^{M} S_x \quad (3.4)$$

เมื่อ Th_l คือ ค่า Threshold ที่นำมาใช้สำหรับเลือกจุดที่เหมาะสม

2) เลือกจุดที่ต้องการจากกระบวนการแยกภาพขา (Leg segmentation) โดยการเลือกจุด r_x ณ แถวที่ x ที่มีผลรวมของจำนวนจุดภาพตามแนวนอนมากกว่าค่า Th_l หรือค่า Threshold ที่คำนวณได้ ดังสมการต่อไปนี้

$$R_x = \{x|, S_x > Th_l\} \quad (3.5)$$

เมื่อ R_x คือ เซ็ตของแถวที่ถูกเลือก ซึ่งมีค่า $S_x > Th_l$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) **นำจุด r_x , $x \in R_x$ ที่ได้มาคำนวณหาตัวแปร m และ b จากสมการเน้นตรง $y = mx + b$ เพื่อใช้เป็นเส้นอ้างอิงในการคำนวณหาตำแหน่งของจุดที่ผู้ใช้ยืนในขั้นตอนต่อไป การคำนวณจะใช้วิธีการ Least squares approximation ดังสมการที่ 3.6 และ 3.7**

$$m = \frac{P\left(\sum_{x\in R_x} x.r_x\right) - \left(\sum_{x\in R_x} x\right)\left(\sum_{x\in R_x} r_x\right)}{P\left(\sum_{x\in R_x} X^2\right) - \left(\sum_{x\in R_x} x\right)^2} \tag{3.6}$$

$$b = \frac{\left(\sum_{x\in R_x} r_x\right)\left(\sum_{x\in R_x} X^2\right) - \left(\sum_{x\in R_x} x\right)\left(\sum_{x\in R_x} x.r_x\right)}{P\left(\sum_{x\in R_x} X^2\right) - \left(\sum_{x\in R_x} x\right)^2} \tag{3.7}$$

เมื่อ m คือ ค่าความชันของสมการเส้นตรง

b คือ จุดที่เส้นตรงตัดกับแกน y

P คือ จำนวนของจุดที่นำใช้ในการคำนวณ (จำนวนสมาชิกของ R_x)

3.3.4 กระบวนการคำนวณหาตำแหน่งส้นเท้า (Leg-endpoint Estimation)

กระบวนการคำนวณหาตำแหน่งส้นเท้า เป็นกระบวนการที่ใช้คำนวณหาจุดที่เป็นตำแหน่ง**ของส้นเท้า** C_{xy} จากโครงสร้างของขาที่ได้จากกระบวนการที่แล้ว ซึ่งมีขั้นตอนการทำงานดังต่อนี้

1) นำจุดที่เป็นโครงสร้างของขาที่ได้ มาเปรียบเทียบกับค่าจากสมการเส้นตรง ว่ามีระยะ ห่างจากสมการเส้นตรงเท่าใด ดังสมการต่อไปนี้

$$e_{x'} = |r_{x'} - (mx' + b)|,\ x' \in R_x \tag{3.8}$$

เมื่อ $e_{x'}$ คือ ค่าระยะห่างของจุดโครงสร้างขาจากสมการเส้นตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2) หาค่าเฉลี่ยของค่าระยะห่างจากค่า $e_{x'}$ ที่ได้ ดังสมการต่อไปนี้

$$A = \frac{\sum_{1}^{P} e_{x'}}{P} \tag{3.9}$$

เมื่อ A คือ ค่าเฉลี่ยของค่าระยะห่างของจุดโครงสร้างขาจากสมการเส้นตรง

3) ทำการเลือกจุดที่เป็นโครงสร้างของขาจาก r_x โดยเลือกเฉพาะจุดที่มีค่าระยะห่างของจุดน้อยกว่าค่าเฉลี่ยที่คำนวณได้ โดยพิจารณาจากจุดที่เลือกที่มีค่า x มากที่สุดไปหาค่าน้อยที่สุด ถ้าจุดใดมีค่าระยะห่างจากสมการเส้นตรงน้อยกว่าค่าของ A เป็นตำแหน่งแรก ให้เลือกจุดนั้นเป็นตำแหน่งส้นเท้า C_{xy}

3.3.5 กระบวนการคำนวณตำแหน่งของผู้ใช้ (Estimate viewer position)

กระบวนการคำนวณตำแหน่งของผู้ใช้ คือ กระบวนการที่ใช้ในการคำนวณหาตำแหน่งของผู้ใช้ใน World plane ซึ่งจะแทนด้วย W_{xy} โดยสามารถคำนวณได้จากการนำค่าตำแหน่งส้นเท้า C_{xy} จากขั้นตอนที่ผ่านมา ไปแทนค่าในสมการที่ 3.10 และ สมการที่ 3.11 ซึ่งเป็นสมการที่ได้มาจากหลักการของ (Criminisi,A. et.al. 1999 : 625-634) Homography หรือ Projective transformation โดยอาศัยหลักการฉายภาพจาก World plan ไปยัง Image Plan และ Transformation matrix ขนาด 3×3 ซึ่งสามารถนำไปประยุกต์ใช้ในการคำนวณหาตำแหน่งใน World plane จากภาพเพียงภาพเดียวได้ สำหรับสมการที่ใช้ในการคำนวณหาตำแหน่ง X,Y ของผู้ใช้ใน World plane นี้ก็คือ

$$X = \frac{ax + by + c}{gx + hy + 1} \tag{3.10}$$

$$Y = \frac{dx + ey + f}{gx + hy + 1} \tag{3.11}$$

เมื่อนำค่าค่าตำแหน่งส้นเท้า C_{xy} จากขั้นตอนที่ผ่านมา ไปแทนค่าก็จะได้ตำแหน่ง X,Y ของผู้ใช้ใน World plane ออกมาแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 อัลกอริทึมสำหรับแก้ไขข้อผิดพลาด

จากอัลกอริทึมในการตรวจวัดตำแหน่งขาที่ผ่านมา อาจจะมีความผิดพลาดเกิดขึ้นได้ เช่น เกิดจากการเปลี่ยนแปลงของแสง หรือ เงาที่พื้น เป็นต้น ดังนั้นเพื่อที่จะทำให้ค่าตำแหน่งที่คำนวณได้มีความเหมาะสม จึงต้องมีอัลกอริทึมสำหรับแก้ไขข้อผิดพลาดของตำแหน่งที่คำนวณได้ ซึ่งก็คือ อัลกอริทึมที่จะใช้ปรับค่าตำแหน่งให้เหมาะสม หรือ Smoothing algorithm โดยอัลกอริทึมนี้จะใช้ปรับค่าตำแหน่งของผู้ใช้ที่คำนวณได้ให้มีความเหมาะสมที่จะนำไปใช้ โดยนำค่าตำแหน่งมาคำนวณดังสมการต่อไปนี้

$$W_{i,xy}(t) = \frac{\sum_{m=1}^{N-1} W_{xy}(t-m)}{N} \qquad (3.12)$$

เมื่อ $W_{i,xy}$ คือ ค่าตำแหน่ง W_{xy} ตัวที่ I ในเวลาใด t ใดๆ

N คือ จำนวนของข้อมูลตำแหน่งที่นำมาใช้ในการคำนวณ

t คือ ค่าดัชนีเวลาของค่าตำแหน่ง W_{xy}

(ณรงค์ศักดิ์ พุดเผือด. 2548 : 32-36)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
การเชื่อมต่อและการสื่อสารระหว่าง Flash กับ Visual C++

4.1 ความรู้เบื้องต้นเกี่ยวกับ Flash ที่ใช้ใน Visual C++

4.1.1 ความสำคัญของ Flash ที่ต้องนำมาใช้กับ Visual C++

ในโลกยุคปัจจุบันนี้ การพัฒนาโปรแกรมนั้น ไม่ได้มีเพียงแค่เวลา หน่วยความจำ และพื้นที่เท่านั้นที่สำคัญ แต่ภาพลักษณ์ที่แสดงออกมาก็เป็นเรื่องที่สำคัญมากไม่ยิ่งหย่อนไปกว่ากัน ดังเช่น เมื่อเรามองดูเว็ปไซต์ เราจะพบว่ามันมี interface ที่เต็มไปด้วย multi-media และ visual effect ที่สวยงามและดึงดูดใจเป็นอย่างมาก แต่ในทางตรงกันข้าม โปรแกรมที่ใช้งานบนหน้าจอคอมพิวเตอร์ทั่วไปที่ถูกพัฒนาจาก Visual Basic หรือ Visual C++นั้น กลับมีรูปแบบของ interface เป็นสีเทา ที่น่าเบื่อ ซ้ำซาก และไม่ดึงดูดใจเลย ดังนั้นจึงจำเป็นที่จะต้องนำเอา Flash เข้ามาเพิ่มสีสันและความน่าสนใจให้มากยิ่งขึ้น และที่สำคัญก็คือ การออกแบบหน้าจอโดยใช้ Flash นั้นทำได้อย่างง่ายดายและสนุกสนาน นั้นก็เพราะ Flash มีเครื่องไม้เครื่องมือและ components บรรจุอยู่อย่างครบถ้วนให้เลือกใช้นั้นเอง

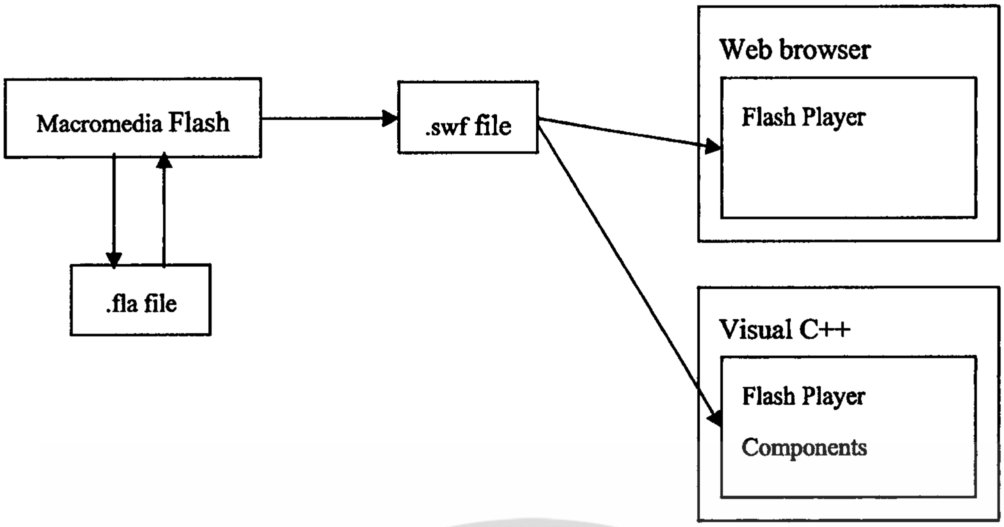
4.1.2 การเขียนโปรแกรม และภาษาที่ใช้กับ Flash

โปรแกรม Flash นั้นเป็นโปรแกรมที่โดดเด่นด้านการ Design และการทำ Animation เป็นอย่างมาก เนื่องจากเป็นโปรแกรมที่มีเครื่องไม้เครื่องมือที่เพียบพร้อม ครบครัน และใช้งานง่าย ดังนั้น มันจึงถูกนำไปใช้ในการสร้างสรรค์ผลงานต่างๆ มากมาย และเป็นโปรแกรมที่นิยมกันอย่างกว้างขวางในยุคปัจจุบัน แต่ไม่เพียงเท่านั้น มันยังสามารถพัฒนาให้มีความฉลาดมากขึ้น ด้วย การเขียนโปรแกรมควบคุมการทำงานภายใน Flash โดยใช้ภาษาโปรแกรมที่เรียกว่า ActionScript ซึ่งเป็นภาษาที่ถูกสร้างขึ้นมาเฉพาะ เพื่อใช้ควบคุมการทำงานต่างๆ ภายในโปรแกรม Flash นอกจากนี้ ยังสามารถสร้างและเพิ่มเติมส่วนของ components เข้าไปใหม่ได้เองตามความต้องการ

4.1.3 File format ของ Flash ที่ใช้ใน Visual C++

ไฟล์ในโปรแกรม Flash ที่ใช้ในการสร้าง ตกแต่ง และแก้ไข ผลงานการออกแบบและทำ Animation นั้น คือ ไฟล์ .fla ซึ่งเป็นไฟล์ที่จะใช้เฉพาะกับโปรแกรม Flash เท่านั้น โดยไฟล์นี้จะทำการเก็บส่วนของ Graphics และ ActionScript ที่ใช้ในงานไว้ ดังนั้น เวลาที่เราจะเอาผลงานไปใช้กับโปรแกรมอื่น เราจะต้องทำการแปลงไฟล์จาก .fla ไปเป็น .swf เสียก่อน โดยใช้คำสั่ง Export Movie จากเมนู File ซึ่งไฟล์ .swf นั้น เราจะสามารถพบเห็นได้ทั่วไปบนเว็บไซต์ โดยจะถูกเล่นจากโปรแกรม Flash Player ที่แสดงอยู่ภายใน web browser และไฟล์นี้นั้นเองที่จะถูกนำไปใช้ในโปรแกรม Visual C++ ด้วยเช่นกัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

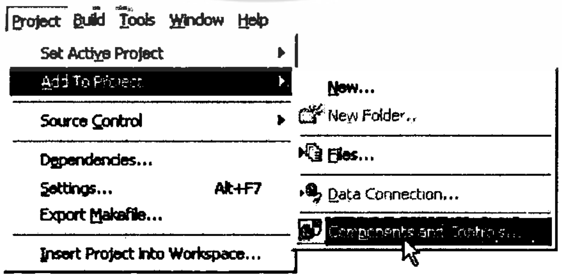


รูปที่ 4.1 File format ของ Flash และการใช้งานบน Web browser และ Visual C++

4.2 การเพิ่ม Component Shockwave Flash Object ของ Flash เข้าไปใน Visual C++

ด้วยเทคโนโลยี Microsoft's COM/Active X ทำให้ Shockwave Flash OCX control เป็น Active X Component ที่สมบูรณ์และเพียบพร้อมในการใช้งาน และด้วยเหตุนี้เอง จึงทำให้ไฟล์ .swf ของ Flash นอกจากที่จะใช้ใน web browser แล้วยังสามารถนำไปใช้ในภาษาโปรแกรมอื่นๆ ได้อีกหลากหลายและง่ายดาย ขอเพียงแค่เป็นภาษาโปรแกรมที่สนับสนุน OCX Format เท่านั้น ก็สามารถนำไปใช้งานได้แล้ว ตัวอย่างเช่น Visual C++ , Visual Basic , Delphi และอื่นๆ

โดย Shockwave Flash OCX Active X Component นี้จะถูกติดตั้งอยู่ในระบบของเรา ก็ต่อเมื่อ เราได้ทำการติดตั้งโปรแกรม Flash หรือ Flash Player ไว้แล้วเท่านั้น ซึ่งในตอนแรกนั้น เราจะต้องทำการเพิ่ม Shockwave Flash OCX Active X Component เข้าไปในโปรแกรม Visual C++ เสียก่อน โดยเข้าไปที่เมนู Project เลือก Add to Project แล้วคลิกเลือก Components and Controls… ดังรูปที่ 4.2



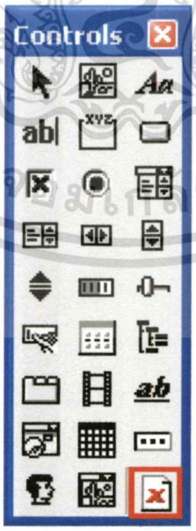
รูปที่ 4.2 วิธีการเพิ่ม Component ต่างๆ เข้าสู่ Project ใน Visual C++

แล้วเลือก Shockwave Flash Object เพื่อเพิ่ม Component ของ Flash เข้าไปใน Visual C++ ดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.3 การเพิ่ม Component Shockwave Flash Object ของ Flash เข้าไปใน Visual C++

เมื่อเพิ่ม Component Shockwave Flash Object ของ Flash เรียบร้อยแล้วจะปรากฏ เครื่องมือ Shockwave Flash Object ดังรูปที่ 4.4 ขึ้นมาให้นำใช้งาน



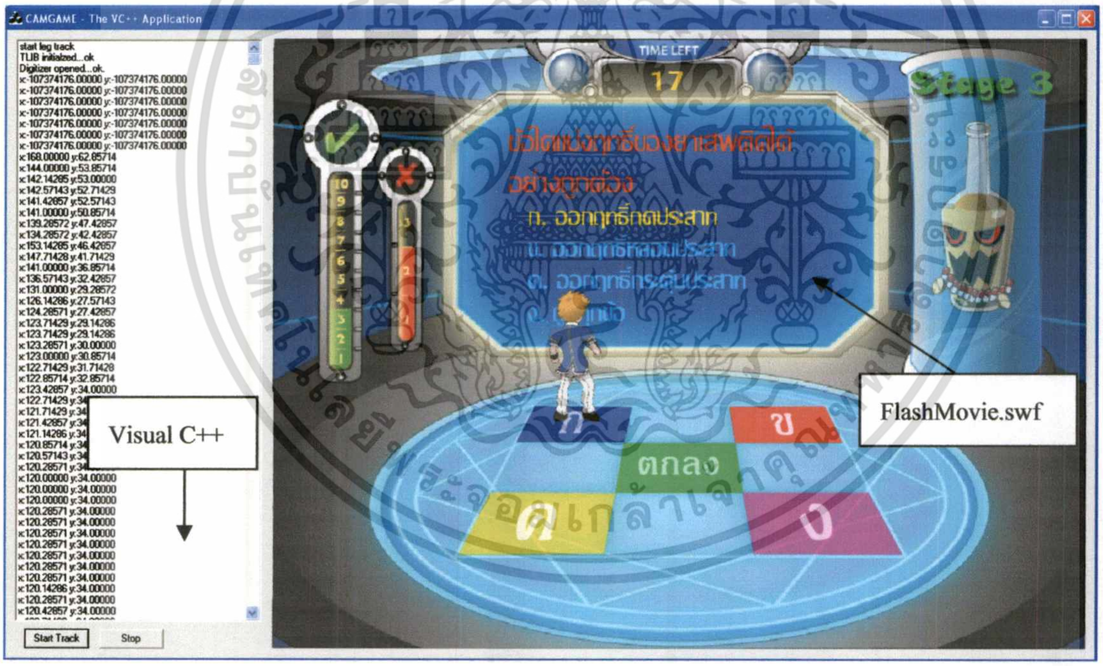
รูปที่ 4.4 เครื่องมือ Shockwave Flash Object ที่เพิ่มเข้ามาให้นำไปใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 การเชื่อมต่อระหว่าง Flash กับ Visual C++

หลังจากทำการติดตั้ง Component Shockwave Flash Object ของ Flash เข้าไปใน Visual C++ เป็นที่เรียบร้อยแล้ว ก็สามารถคลิกลากเครื่องมือ Shockwave Flash Object นี้มาใช้ใน Visual C++ ได้เลย โดยหลังจากที่นำ Shockwave Flash Object มาใช้งานใน Project แล้ว ก็ต้องมาทำการกำหนดค่าตัวแปลที่จะนำมาใช้เรียก Component Shockwave Flash Object นี้เสียก่อน โดยคลิกขวาที่ Shockwave Flash Object ใน Project แล้วเลือก ClassWizard... ก็จะปรากฏหน้าต่าง MFC ClassWizzard ขึ้นมา เข้าไปที่ Tab Member Variable จากนั้นเลือก Add Variable กำหนดชื่อให้กับ Component Shockwave Flash Object นี้ เช่น m_flashplayer เพียงเท่านี้ก็จะได้ Component Shockwave Flash Object มาไว้ใช้งานได้แล้ว

หลังจากนั้นให้ทำการโหลดไฟล์ .swf ของ Flash เข้ามาในโปรแกรม Visual C++ ด้วยการใช้ฟังก์ชั่น SetMovie(); ซึ่งเป็นหนึ่งในฟังก์ชั่นที่มีมาให้พร้อมกับ Component Shockwave Flash Object เช่น *m_flashplayer.SetMovie("C:\\TLIB\\examples\\camgame\\flash\\SNgame.swf");* ซึ่งสิ่งนี้จะทำให้เราสามารถที่จะเล่นไฟล์ .swf ของ Flash ได้ในโปรแกรม Visual C++ ได้แล้ว ดังรูปที่ 4.5



รูปที่ 4.5 การเชื่อมต่อระหว่าง Flash กับ Visual C++

4.4 การติดต่อสื่อสารระหว่าง Flash กับ Visual C++

หลังจากที่ทำการโหลดไฟล์ .swf ของ Flash เข้ามาในโปรแกรม Visual C++ ด้วยการใช้ฟังก์ชั่น SetMovie(); แล้วนั้น เราก็พร้อมที่จะทำการติดต่อสื่อสารกับไฟล์ Flash นั้นๆแล้ว

โดยใน Shockwave Flash OCX Active X Component นั้น มีหลาย Functions และหลาย Events ให้เราสามารถเลือกนำมาใช้งานเพื่อให้เหมาะสมกับประเภทและความต้องการของแต่ละงานนั้นๆ ซึ่ง

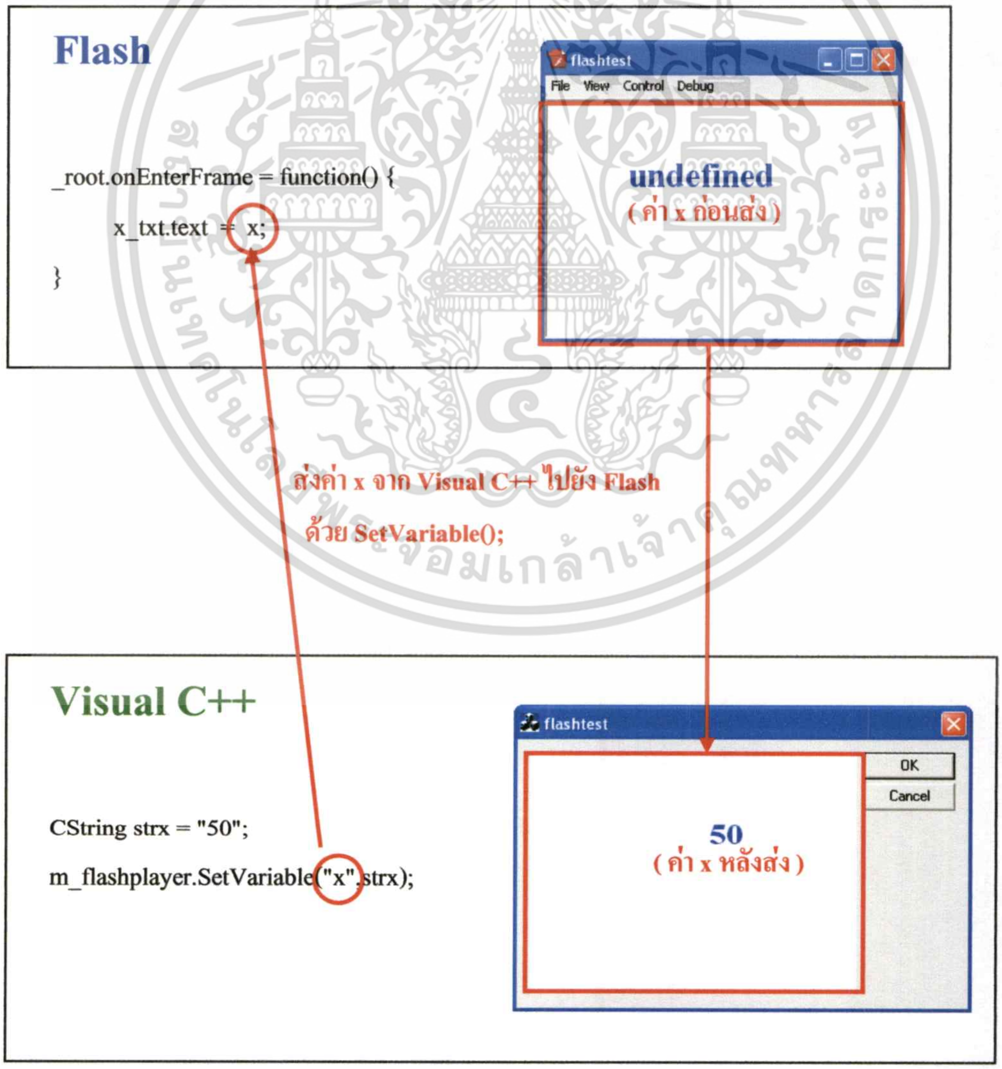
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ฟังก์ชั่นสำคัญๆ ที่ใช้ในการสื่อสารระหว่าง Flash กับ Visual C++ ก็คือ ฟังก์ชั่นที่ใช้ในการรับส่งค่าระหว่าง Flash กับ Visual C++ นั้นเอง โดยฟังก์ชั่น SetVariable จะเป็นการส่งค่าจาก Visual C++ ไปยัง Flash และฟังก์ชั่น GetVariable จะเป็นการรับค่าจาก Flash เข้ามายัง Visual C++ ซึ่งค่าที่จะใช้รับส่งกันระหว่าง Flash กับ Visual C++ นั้นจะต้องเป็น String เท่านั้น

4.4.1 การส่งค่าจาก Visual C++ ไปยัง Flash

การส่งค่าจาก Visual C++ ไปยัง Flash นั้นสามารถทำได้โดยใช้ฟังก์ชั่น SetVariable ที่ให้มาพร้อมกับ Component Shockwave Flash Object ของ Flash ซึ่งมีรูปแบบการใช้งาน ดังต่อไปนี้คือ*void SetVariable (LPCTSTR name, LPCTSTR value);* โดยที่ *LPCTSTR name* คือ ชื่อตัวแปลที่ถูกตั้งขึ้นเพื่อจะนำไปใช้ใน Flash และจะต้องมีชนิดเป็น String เท่านั้น ส่วน *LPCTSTR value* นั้นก็คือ ค่าที่ Visual C++ กำหนดให้กับตัวแปล LPCTSTR name ซึ่งก็จะต้องมีชนิดเป็น String ด้วยเช่นเดียวกัน

ตัวอย่างการส่งค่าจาก Visual C++ ไปยัง Flash โดยใช้ฟังก์ชั่น SetVariable มีดังต่อไปนี้



รูปที่ 4.6 การส่งค่าจาก Visual C++ ไปยัง Flash ด้วยฟังก์ชั่น SetVariable

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปข้างบนจะเห็นได้ว่าเราสามารถส่งค่าไปยัง Flash ได้ง่ายๆ โดยส่งค่าผ่านทางฟังก์ชั่น SetVarible(); เท่านั้นเอง และฟังก์ชั่นนี้ก็สามารถที่จะส่งค่าในแบบ Real-time ได้ด้วย ซึ่งจะขึ้นอยู่กับค่า strx ที่เราจะกำหนดนั้นเอง ซึ่งในโปรเจคนี้ก็จะใช้การส่งค่า strx ในแบบ Real-time โดยให้ strx เป็นค่าในแกน x ที่ได้จากการประมวลผลภาพจากกล้องวิดีโอ และ stry เป็นค่าในแกน y ที่ได้จากการประมวลผลภาพจากกล้องวิดีโอ ซึ่งค่าทั้ง 2 นี้จะเปลี่ยนอยู่ตลอดเวลาตามการเคลื่อนไหวของผู้เล่น จึงต้องทำการส่งค่าอยู่ตลอดเวลาในรูปแบบ Real-time นั้นเอง

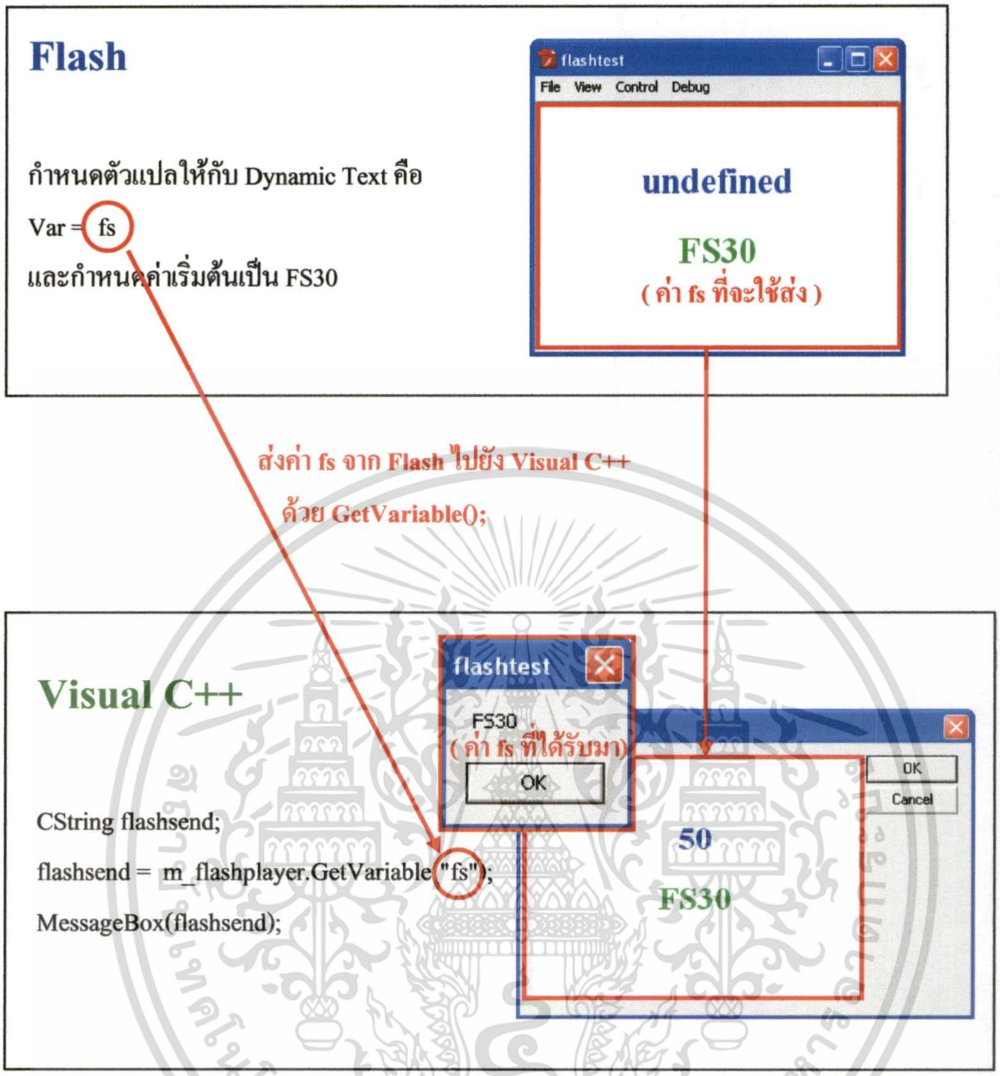
แต่จากการรับค่าใน Flash จะกระทำเพียงครั้งแรกครั้งเดียวที่เชื่อมต่อกันเท่านั้น ดังนั้นเพื่อให้ Flash สามารถรับการส่งค่าที่มาในแบบ Real-time ได้นั้น เราจึงต้องกำหนดให้ Flash วนลูปเพื่อเช็คค่าที่จะเข้ามาอยู่ตลอดเวลา โดยเราจะใช้ฟังก์ชั่นการทำงานของ onEnterFrame ในการวนเช็คค่าที่จะส่งผ่านเข้ามายัง Flash ซึ่งฟังก์ชั่นการทำงานของ onEnterFrame นั้นจะทำการวนซ้ำทุกๆ รอบ Framerate หรือ Frame/Second ที่กำหนดไว้ในไฟล์ Flash เพียงเท่านี้เราก็จะสามารถรับค่าจาก Visual C++ ในแบบ Real-time ได้แล้ว

4.4.2 การรับค่าจาก Flash ของ Visual C++

การรับค่าจาก Flash ของ Visual C++ นั้น สามารถทำได้โดยใช้ฟังก์ชั่น GetVariable ที่ให้มาพร้อมกับ Component Shockwave Flash Object ของ Flash ด้วยเช่นเดียวกัน ซึ่งมีรูปแบบการใช้งานที่แตกต่างจาก SetVariable อยู่บ้าง ดังต่อไปนี้ คือ *CString GetVariable(LPCTSTR name);* โดยที่ *LPCTSTR name* คือ ชื่อตัวแปลที่ถูกตั้งขึ้นเพื่อใช้รับค่าที่ Flash จะส่งค่ามาให้ โดยจะเป็นชื่อตัวแปลเดียวกันกับที่ Flash ใช้ส่งค่ามา และจะต้องมีชนิดเป็น String เท่านั้น

ตัวอย่างการรับค่าจาก Flash ของ Visual C++ โดยใช้ฟังก์ชั่น GetVariable มีดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 การรับค่าจาก Flash ของ Visual C++ ด้วยฟังก์ชั่น GetVariable

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

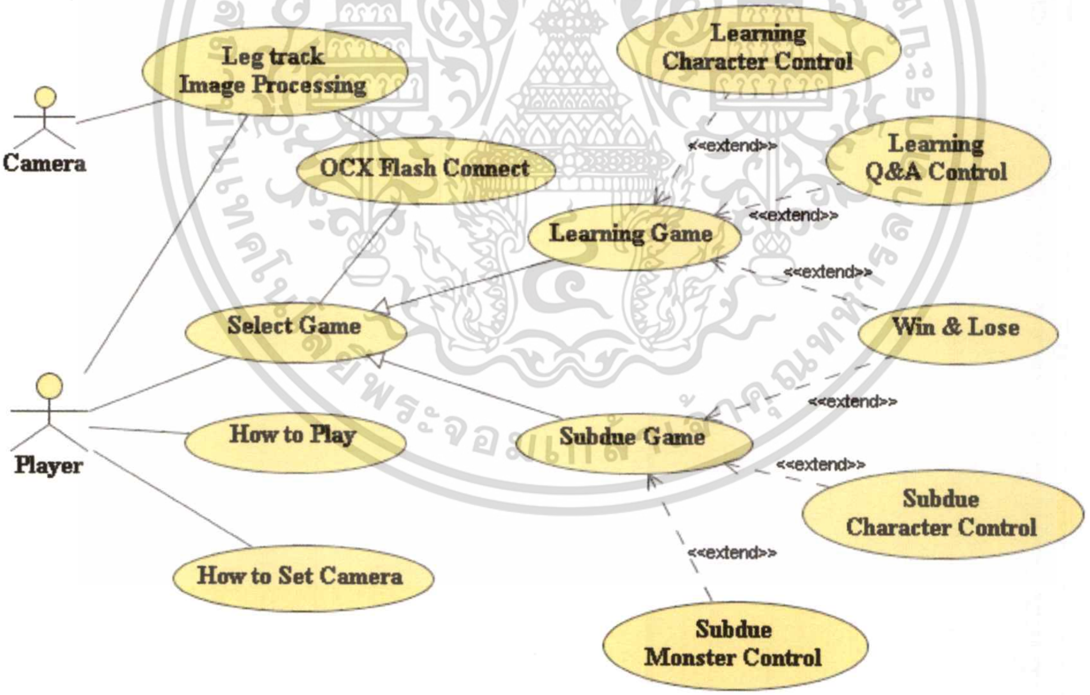
บทที่ 5

การวิเคราะห์และออกแบบระบบ

ในบทนี้จะกล่าวถึงการวิเคราะห์และออกแบบระบบการทำงานของตัวเกม โดยจะใช้ Use Case Diagrams และ Activity Diagrams เป็นเครื่องมือที่ใช้อธิบายภาพรวมและลำดับขั้นตอนการทำงานของระบบทั้งหมด

5.1 ยูสเคสไดอะแกรม (Use Case Diagram)

หลังจากที่ได้วิเคราะห์ความต้องการและข้อจำกัดของเกมแล้ว จะสามารถนำมาสร้างเป็นยูสเคสไดอะแกรม เพื่อใช้อธิบายฟังก์ชันการทำงานของระบบ ซึ่งสามารถแสดงให้เห็นถึงการใช้งานระบบอย่างครบถ้วน ดังที่แสดงให้เห็นดังรูปที่ 5.1



รูปที่ 5.1 Use Case Diagram ของเกมปราบปรามยาเสพติด

จากแผนภาพด้านบน จะเห็นได้ว่า Use Case Diagram นั้นก็คือแผนภาพที่แสดงความสัมพันธ์ของการทำงานโดยรวมของระบบ ซึ่งจะอธิบายว่าในระบบมีการดำเนินงานอะไรบ้าง โดยจะแสดงการติดต่อระหว่างระบบกับผู้ใช้ ซึ่งประกอบด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- **Actor** หมายถึง สิ่งใดๆก็ตามที่ใช้งานระบบหรือมีส่วนร่วมกับ Use Case ภายในระบบ โดยสิ่งดังกล่าวอาจเป็นคน อุปกรณ์ต่างๆหรือระบบอื่นๆ เป็นต้น โดย Actor จะมีการแลกเปลี่ยนข้อมูลข่าวสารกับระบบและมีสัญลักษณ์ที่ใช้คือ รูปคน ในที่นี้ก็ได้แก่ ผู้เล่นที่เข้ามาเล่นเกมกับระบบ และ กล้องวีดีโอ ที่ใช้จับภาพและประมวลผลภาพของผู้เล่นเพื่อบอกตำแหน่งจุดยืนของผู้เล่นให้กับระบบมาใช้ในการเล่นเกม
- **Use Case** คือ กิจกรรมหลักๆที่เกิดขึ้นภายในระบบซึ่งอาจเป็นกิจกรรมระหว่างผู้ใช้กับระบบหรือระบบกับระบบ ซึ่งจะใช้รูปวงรีเป็นสัญลักษณ์
- **Relationship** เป็นการแสดงความสัมพันธ์แบบต่างๆ ระหว่าง Use Case กับ Use Case และ Use Case กับ Actor สำหรับความสัมพันธ์ระหว่าง Use Case สามารถแบ่งออกได้เป็นสองแบบ คือ
    1. ความสัมพันธ์แบบขยาย (Extends Relationship) คือ Use case หนึ่งไปมีมีผลต่อการทำงานตามปกติของอีก Use case หนึ่ง นั่นหมายถึงว่า Use case ที่มาขยายมัน[27] จะมีผลทำให้การดำเนินการของ Use case ที่ถูกขยายมีการเปลี่ยนกิจกรรมไป สัญลักษณ์ที่ใช้คือ <<extends>> (เป็นเส้นประพร้อมหัวลูกศรที่ชี้จาก Use case ที่ขยายไปยัง Use case ที่ถูกขยาย โดยมีคำว่า <<extend>> กำกับอยู่บนเส้นลูกศร)
    2. ความสัมพันธ์แบบใช้ (Include Association) คือการที่ Use case หนึ่ง เรียกใช้บริการจาก Use case อีกอันดับหนึ่ง คล้าย ๆ กับการเรียกใช้งานโปรแกรมย่อยโดยโปรแกรมหลัก สัญลักษณ์ที่ใช้คือ <<include>> (เป็นเส้นประพร้อมหัวลูกศรชี้ไปยัง Use case ที่ถูกเรียกใช้งาน โดยมีคำว่า << include >> กำกับอยู่บนเส้นลูกศร)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 คำอธิบายยูสเคสไดอะแกรม (Use Case Diagram Description)

ตารางที่ 5.1 ยูสเคส Leg Track Image Processing

ชื่อยูสเคส:	Leg Track Image Processing	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	การตรวจวัดตำแหน่งขาของผู้เล่นด้วยกล้องวีดีโอ	
สิ่งกระตุ้นเหตุการณ์:	ตำแหน่งจุดยืนของผู้เล่นบนพื้น	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการเก็บภาพพื้นหลัง 2. ระบบทำการเปลี่ยนภาพพื้นหลังที่ได้ให้เป็นภาพแบบ Grayscale
	3. ผู้เล่นเข้ามายืนในบริเวณกล้อง	
		4. เก็บภาพขาที่ต้องการจะลบภาพพื้นหลังออก 5. ระบบทำการเปลี่ยนภาพขาที่ได้ให้เป็นภาพแบบ Grayscale 6. ระบบนำภาพแบบ Grayscale ทั้ง 2 ภาพไปผ่านกระบวนการ Background Subtraction 7. ระบบจะได้ภาพขาที่ลบภาพพื้นหลังออกเรียบร้อยแล้ว 8. ระบบทำการนำภาพขาของผู้เล่นมาคำนวณหาจุดกึ่งกลางของภาพในแต่ละแถว 9. ระบบใช้วิธีการ Center of Mass คำนวณหาจุดกึ่งกลางของภาพตามแนวนอน 10. ระบบนำไปวาดเป็นกราฟ ซึ่งจะทำให้ได้ตำแหน่งที่ผู้เล่นยืนอยู่ในภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.1 (ต่อ)

ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		11. ระบบทำการเลือกจุดที่มีผลรวมของจำนวนจุดภาพตามแนวนอน มากกว่าค่า threshold 12. ระบบนำจุดที่ได้มาคำนวณหาตัวแปร m และ b ของสมการเส้นตรง y = mx+b เพื่อใช้เป็นเส้นอ้างอิงที่จะใช้คำนวณหาตำแหน่งจุดยืนของผู้เล่น 13. ระบบได้จุดที่เป็นโครงสร้างขาโดยเฉพาะ หลังจากตัดเงาออกแล้ว 14. ระบบทำการคำนวณหาจุดที่เป็นตำแหน่งของส้นเท้า โดยนำจุดที่เป็นโครงสร้างขามาเปรียบเทียบกับค่าจากสมการเส้นตรงที่คำนวณไว้ ว่ามีระยะห่างจากสมการเส้นตรงใด 15. ระบบทำการหาค่าเฉลี่ยของค่าระยะห่างของจุดโครงสร้างขาจากสมการเส้นตรง 16. ระบบทำการเลือกจุดที่เป็นโครงสร้างของขา โดยพิจารณาจากจุดที่เลือกที่มีค่า x มากที่สุดไปหาค่าน้อยที่สุด ถ้าจุดใดมีระยะห่างจากสมการเส้นตรงน้อยกว่าค่าของ A เป็นตำแหน่งแรกให้เลือกจุดนั้นเป็นตำแหน่งส้นเท้า 17. ระบบทำการหาค่าตำแหน่งของผู้เล่นใน World Plane ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.1 (ต่อ)

ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		การนำค่าตำแหน่งส้นเท้าที่ได้ มาแทนค่า 18. ระบบทำการนำค่าตำแหน่งที่ได้มาปรับค่าตำแหน่งให้เหมาะสม เพื่อแก้ไขข้อผิดพลาดที่อาจเกิดขึ้นได้ เช่นการเปลี่ยนแปลงของแสงหรือเงาที่พื้น
เหตุการณ์ทางเลือก:		

ตารางที่ 5.2 ยูสเคส OCX Flash Connect

ชื่อยูสเคส:	OCX Flash Connect	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Program	
คำอธิบาย:	ทำหน้าเชื่อมต่อและรับส่งข้อมูลที่ได้จากการประมวลผลภาพจากกล้องวีดีโอใน Visual C++ กับไฟล์ swf ของ Flash	
สิ่งกระตุ้นเหตุการณ์:	ค่าตำแหน่ง x ,y ที่ได้จากกล้องวีดีโอ	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการเชื่อมต่อ Shockwave Flash OCX Active X Component เข้ากับโปรแกรม Visual C++ 2. ระบบทำการโหลดไฟล์ .swf ที่ต้องการใช้งานของ Flash เข้ามาในโปรแกรม Visual C++ ด้วยฟังก์ชัน SetMovie();
	3.ผู้เล่นทำการเปลี่ยนตำแหน่งจุดยืน	4. ระบบทำการส่งค่า x, y จาก Visual C++ ไปยัง Flash ในแบบ Real-time ด้วยฟังก์ชัน SetVariable();
เหตุการณ์ทางเลือก:		

เอกสารนี้มีลิขสิทธิ์สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.3 ยูสเคส Select Games

ชื่อยูสเคส:	Select Games	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	ทำหน้าที่เลือกเกมที่จะเล่น	
สิ่งกระตุ้นเหตุการณ์:	การเลือกเกมของผู้เล่น	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการแสดงปุ่ม "Select Games"
	2. คลิกเลือกปุ่ม "Select Games"	
		3. ระบบทำการแสดงหน้าจอ Select Game
		4. ระบบแสดงปุ่มเกม "Learning Game" และ "Subdue Game" ขึ้นมาให้เลือก
	5. คลิกเลือกเกมที่สนใจ	
เหตุการณ์ทางเลือก:		

ตารางที่ 5.4 ยูสเคส How to Play

ชื่อยูสเคส:	How to Play	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	เป็นการอธิบายวิธีเล่นของเกม	
สิ่งกระตุ้นเหตุการณ์:	การเลือกเข้าไปชมวิธีการเล่นเกมของแต่ละเกม	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการแสดงปุ่ม "How to Play"
	2. คลิกเลือกปุ่ม "How to Play"	
		3. ระบบแสดงเกมขึ้นมาให้เลือก พร้อมกับอธิบายวิธีการเล่นเกมของแต่ละเกม
เหตุการณ์ทางเลือก:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.5 ยูสเคส How to Set Camera

ชื่อยูสเคส:	How to Set Camera	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	อธิบายวิธีการตั้งกล้อง	
สิ่งกระตุ้นเหตุการณ์:	การเลือกปุ่ม How to Set Camera	
ลำดับการไหลของเหตุการณ์:	แอ๊กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการแสดงปุ่ม "How to Set Camera"
	2. คลิกเลือกปุ่ม "How to Set Camera"	
		3. ระบบแสดงหน้าจออธิบายรายละเอียดของวิธีการตั้งกล้อง
เหตุการณ์ทางเลือก:		

ตารางที่ 5.6 ยูสเคส Play Learning Game และ Subdue Game

ชื่อยูสเคส:	Learning Game และ Subdue Game	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	แสดงการ Load Game ตั้งแต่เริ่มแรกจนกระทั่งเข้าสู่เกม	
สิ่งกระตุ้นเหตุการณ์:	คลิกปุ่มเลือกเกม	
ลำดับการไหลของเหตุการณ์:	แอ๊กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการตรวจสอบ Stage ปัจจุบัน 2.ระบบทำการ Load และแสดงหน้าจอของเกม 3.แสดงข้อความ Stage ปัจจุบันกลางหน้าจอ 4.ระบบตรวจสอบจุดยืนของผู้เล่น 5.แสดงตัวละครของผู้เล่น 6.ระบบแสดงปุ่ม "เริ่มเกม" ตรงกลาง Stand Stage
	7.ผู้เล่นเหยียบปุ่ม "เริ่มเกม"	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.6 (ต่อ)

ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		8.ระบบของเกมเริ่มทำงาน โดยแสดง Animation Loop ขององค์ประกอบในฉาก ตามแต่ละ Stage
เหตุการณ์ทางเลือก:		

ตารางที่ 5.7 ยูสเคส Learning Character Control

ชื่อยูสเคส:	Learning Character Control	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	ควบคุมการแสดงท่าทางของตัวละครและการเคลื่อนตัวละครตามตำแหน่งจุดยืนของผู้เล่น	
สิ่งกระตุ้นเหตุการณ์:	การเคลื่อนที่ของผู้เล่น	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบตรวจสอบจุดยืนของผู้เล่น 2.เมื่อผู้เล่นอยู่เฉยๆ ระบบจะตั้งท่าเตรียมพร้อม (ready)
	3.ผู้เล่นเปลี่ยนตำแหน่งจุดยืนใหม่	
		4.ระบบทำการเคลื่อนตัวละครไปยังตำแหน่งใหม่ ตามค่าตำแหน่งจุดยืนที่ได้มาใหม่ 5. ทำซ้ำตั้งแต่1-4 ไปเรื่อยๆจนกว่าจะรู้แพ้หรือชนะ 6. ถ้าตอบถูกครบตามที่กำหนด ไว้คือครบ 10 ข้อ จะชนะทันทีและจะไปยัง Stage ต่อไป 7. ถ้าผิดครบตามจำนวนที่กำหนดไว้ในแต่ละ Stage จะแพ้ทันที โดยระบบจะหยุดเกม และให้ตัวละครจะแสดงท่าตาย (Die)
เหตุการณ์ทางเลือก:		

เอกสารนี้เป็นลิขสิทธิ์ที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.8 ยูสเคส Learning Q&A Control

ชื่อยูสเคส:	Learning Q&A Control	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	เป็นการควบคุมการทำงานของระบบคำถามในเกม	
สิ่งกระตุ้นเหตุการณ์:	การเลือกคำถามของผู้เล่น ที่มีส่วนสำคัญในการเอาชนะเกม	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการตรวจสอบ Stage ปัจจุบัน 2.ระบบทำการ reset เวลาที่ใช้ในแต่ละรอบ 3.ระบบสุ่มคำถามพร้อมตัวเลือก 4.ถ้าเป็นคำถามที่เคยถามไปแล้วใน Stage นั้นๆ ระบบจะทำการสุ่มคำถามจนกว่าจะได้คำถามใหม่ 5.ระบบแสดงคำถามพร้อมตัวเลือกที่ได้จากการสุ่มขึ้นบนจอ 6.ระบบแสดงปุ่มตัวเลือก "ก, ข, ค และ ง" ขึ้นมาให้เหยียบ 7.ระบบทำการตรวจเช็คเวลา ถ้าเวลาหมด จะแสดงข้อความว่า "Time Out" พร้อมเปิดไฟแดงตรงตอบผิดขึ้น 1 ดวง พร้อมกับเพิ่มจำนวนข้อที่ผิด 1 ข้อ 8. ระบบเฉลยคำตอบ โดยแสดงข้อที่ถูกต้อง 9. ระบบจะตรวจสอบจำนวนข้อที่ผิด ถ้าครบตามที่กำหนดไว้ในแต่ละ Stage เกมจะหยุดลงทันที และระบบจะแสดงการแพ้ โดยจำนวนข้อผิดที่กำหนดไว้ในแต่ละ Stage คือ Stage 1 ตอบผิดครบ 7 ข้อ, Stage 2 ตอบผิดครบ 5

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.8 (ต่อ)

ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		ข้อ, Stage 3 ตอบผิดครบ 3 ข้อ, Stage 4 ตอบผิดครบ 1 ข้อ จะแพ้ทันที
	10.ผู้เล่นเหยียบคำตอบก่อนหมดเวลา	
		11. ระบบทำการเปลี่ยนสีตัวอักษรตามข้อที่เลือก 12. จากนั้นระบบจะแสดงปุ่ม "ตกลง" ขึ้นมาให้เหยียบ เพื่อเป็นการยืนยันคำตอบที่เลือก 13.ระบบทำการตรวจสอบคำตอบที่ได้ว่าถูกต้องหรือไม่ 14.ถ้าคำตอบถูก เปิดไฟเขียวตรงตอบถูก 1 ดวง เพิ่มจำนวนข้อที่ถูกขึ้น 1 ข้อ 15. ถ้าคำตอบผิด เปิดไฟแดงตรงตอบผิด 1 ดวง เพิ่มจำนวนข้อที่ผิด 1 ข้อ จากนั้นจึงเฉลยโดยการแสดงข้อที่ถูกให้เห็น 16. ระบบจะทำการสุ่มคำถามอีกในกรณีที่ยังถูกไม่ถึง 10 ข้อ และผิดยังไม่ครบตามที่กำหนดไว้ในแต่ละ Stage คือยังไม่รู้แพ้ชนะ 17. โดยถ้าถูกครบกำหนดก่อนเกมจะหยุดลงทันที พร้อมทั้งแสดงการชนะ แต่ถ้าผิดครบกำหนดก่อน เกมก็จะหยุดลงทันทีเช่นกัน ตัวละครจะแสดงท่าตายและแสดงการแพ้
เหตุการณ์ทางเลือก:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.9 ยูสเคส Subdue Character Control

ชื่อยูสเคส:	Subdue Character Control	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	ควบคุมการแสดงท่าทางของตัวละครและการเคลื่อนตัวละครตามตำแหน่งจุดยืนของผู้เล่น	
สิ่งกระตุ้นเหตุการณ์:	การเคลื่อนที่ของผู้เล่น	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
		1.ระบบทำการตรวจสอบจุดยืนของผู้เล่น 2.ถ้าไม่มีการเปลี่ยนตำแหน่งของผู้เล่น จะแสดงท่ายืนเตรียมพร้อม (ready)
	3.ถ้ามีการเปลี่ยนตำแหน่งของผู้เล่น	
		4.ระบบแสดงการเคลื่อนที่ไปยังตำแหน่งใหม่ 5.ในกรณีที่เหยียบโดน Monster ระบบจะสุ่มท่าที่จะใช้โจมตี 6.ระบบแสดงท่าโจมตี (attack) นั้น 7.ระบบทำการสุ่มค่าพลังโจมตี 8.เมื่อได้ค่าพลังโจมตีแล้ว ระบบแสดงค่าพลังโจมตี 9.ระบบเพิ่มคะแนน (score) 10.ระบบลดเลือดของ Monster ตามค่าพลังโจมตีที่ได้ 11.หลังจากแสดงท่าโจมตีแล้วตัวละครจะกลับเข้าสู่ท่าเตรียมพร้อม (ready) 12. กรณีที่เหยียบไม่โดน Monster Monster จะแสดงท่าโจมตี แล้วปล่อยพลังออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.9 (ต่อ)

ลำดับการไหลของ	แอ็กเตอร์กระทำ	ระบบตอบสนอง
เหตุการณ์:		13.ถ้าตัวละครถูกพลังโจมตีของ Monster ระบบแสดงท่าโดนโจมตี (damage) เป็นตัวกระพริบ 14.ระบบทำการลดเลือดของตัวละคร 15.ในกรณีที่เลือดของผู้เล่นน้อยกว่าหรือเท่ากับ 0 เกมจะหยุดลงทันที และตัวละครจะแสดงท่าตายพร้อมทั้งแสดงการแพ้ 16. แต่ถ้าเลือดของตัวละครยังไม่หมด จะตัวละครเข้าสู่ท่าเตรียมพร้อม (ready) และเล่นเกมต่อไป
เหตุการณ์ทางเลือก:		

ตารางที่ 5.10 ยูสเคส Subdue Monster Control

ชื่อยูสเคส:	Subdue Monster Control	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	แสดงการเคลื่อนไหว รวมทั้งท่าทางต่างๆของ Monster	
สิ่งกระตุ้นเหตุการณ์:	การโจมตีของ Monster รวมทั้งการโจมตีของผู้เล่น	
ลำดับการไหลของ	แอ็กเตอร์กระทำ	ระบบตอบสนอง
เหตุการณ์:		1.ระบบตรวจสอบ Stage ปัจจุบัน 2.ตรวจสอบจำนวน Monster ที่ต้องแสดงใน Stage นั้น 3.ระบบทำการสุ่มชนิดของ Monster แต่ละตัว ถ้ามีการซ้ำ ระบบจะกระทำการสุ่มใหม่ 4.เมื่อได้ Monster ที่ไม่ซ้ำกันใน Stage ระบบจะกระทำการสุ่มตำแหน่งที่จะแสดงของ Monster

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.10 (ต่อ)

ลำดับการไหลของ	แอ็กเตอร์กระทำ	ระบบตอบสนอง
เหตุการณ์:		5.ในกรณีที่มี Monster อื่นอยู่ ระบบจะกระทำการสุ่มตำแหน่งที่จะแสดง Monster ใหม่ 6.Monster แสดงท่ารอ (ready) ตามความเร็วของ Monster แต่ละตัว 7.ในกรณีที่ถูกโจมตีก่อน แสดงท่าโดนโจมตี (damage) และลบ Monster นั้น 8.ระบบจะลดเลือดของ Monster 9. ถ้าเลือด Monster น้อยกว่า 0 เกมจะจบลงทันที พร้อมทั้งแสดงการชนะ และไปยัง Stage ถัดไป 10.ในกรณีที่ Monster ไม่โดนโจมตีและเป็นฝ่ายปล่อยพลังโจมตี ระบบสุ่มพลังที่จะปล่อย 11. เมื่อได้พลังที่จะปล่อย จะแสดงท่าปล่อยพลัง (attack) ที่ได้ ซึ่งจะแตกต่างกันในแต่ละท่า และแต่ละตัวละคร 12.สุ่มค่าพลังโจมตีตามชนิดของ Monster ซึ่งจะหนักเบาตามชนิดของยาเสพติดที่ Monster นั้นเป็นตัวแทน 13.ลดเลือดของตัวละคร ตามค่าพลังโจมตีที่สุ่มได้ 14. ซึ่งจะเล่นกันไปจนกว่าจะรู้แพ้ชนะกัน โดยเลือดฝ่ายใดหมดก่อนจะเป็นฝ่ายพ่ายแพ้ไป
เหตุการณ์ทางเลือก:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.11 ยูสเคส Win & Lose

ชื่อยูสเคส:	Win & Lose	
ผู้ใช้ที่เกี่ยวข้องโดยตรง:	Player	
คำอธิบาย:	แสดงการชนะและการพ่ายแพ้ของผู้เล่น	
สิ่งกระตุ้นเหตุการณ์:	การที่ผู้เล่นผ่านจุดประสงค์ที่เกมตั้งไว้ หรือพ่ายแพ้ต่อจุดประสงค์ที่เกมตั้งไว้	
ลำดับการไหลของเหตุการณ์:	แอ็กเตอร์กระทำ	ระบบตอบสนอง
	1. ผู้เล่นผ่านจุดประสงค์ที่เกมตั้งไว้	
		2. ระบบทำการตรวจสอบว่าเป็น Stage สุดท้ายหรือไม่
		3. ถ้าไม่ใช่ Stage สุดท้าย แสดงข้อความ "Success" กลางหน้าจอ
		4. ระบบทำการเปลี่ยนเป็น Stage ถัดไป
		5. ถ้าใช่ Stage สุดท้าย แสดงข้อความ "You Win" กลางหน้าจอ
		6. ระบบทำการเป็นเป็นหน้าจอเลือกเกม
	7. ผู้เล่นไม่ผ่านจุดประสงค์ที่เกมตั้งไว้	
		8. ระบบทำการแสดงข้อความ "You Lose" กลางหน้าจอ
		9. ระบบทำการแสดงข้อความ "คุณต้องการจะเล่นด่านนี้ต่อหรือไม่"
		10. ระบบทำการแสดงปุ่ม "เล่นต่อ" และ "เลือกเกม" ขึ้นมาให้เลือกเหยียบ
	11. ผู้เล่นเลือกเหยียบปุ่ม "เล่นต่อ"	
		12. ระบบทำการ Load Stage ขึ้นมาเล่นอีกครั้ง

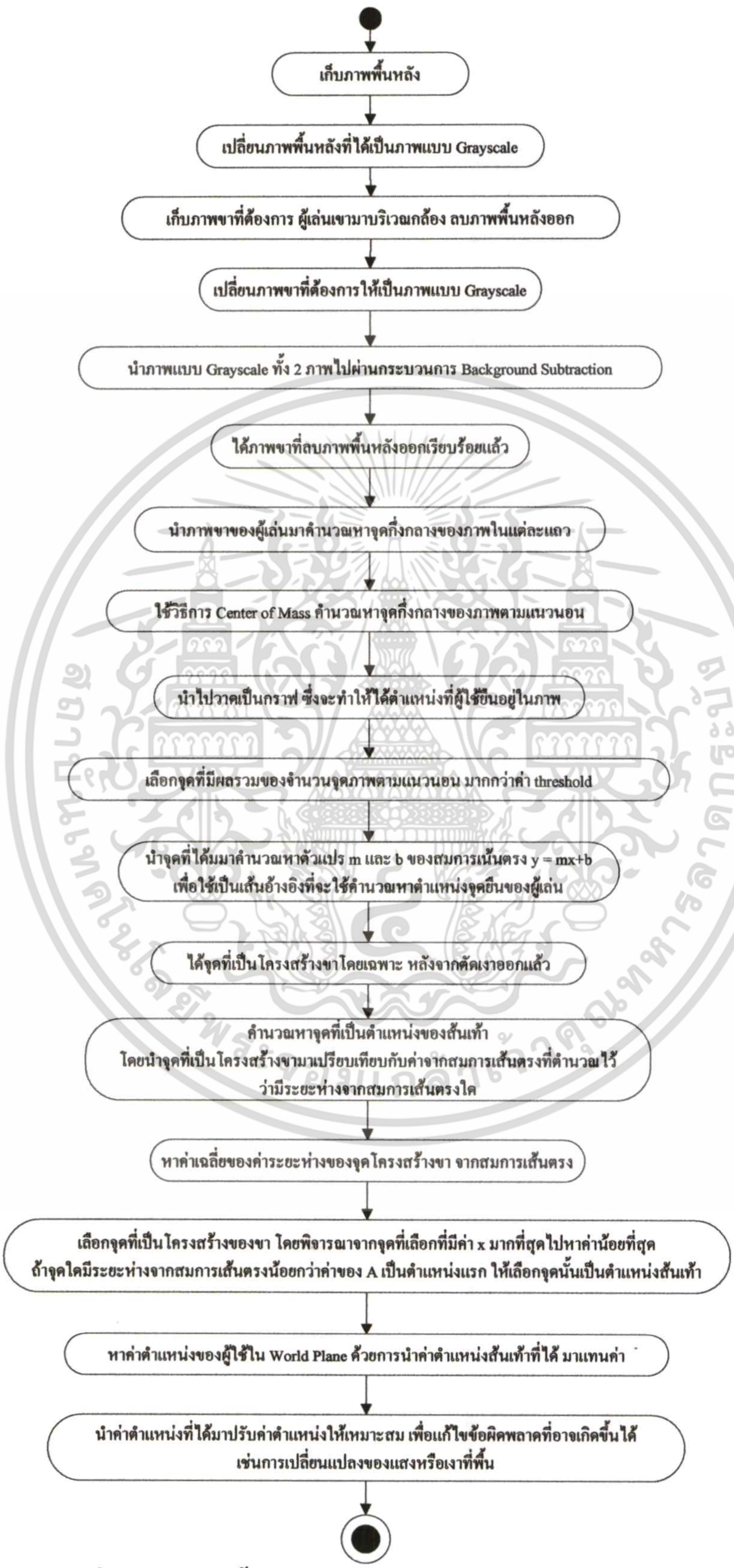
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 5.11 (ต่อ)

ลำดับการไหลของ	แอ๊กเตอร์กระทำ	ระบบตอบสนอง
เหตุการณ์:		12. ระบบทำการ Load Stage ขึ้นมาเล่นอีกครั้ง
	13. ผู้เล่นเลือกเหยียบปุ่ม "เลือกเกม"	
		14. ระบบทำการส่งไปที่หน้าจอเลือกเกม
เหตุการณ์ทางเลือก:		

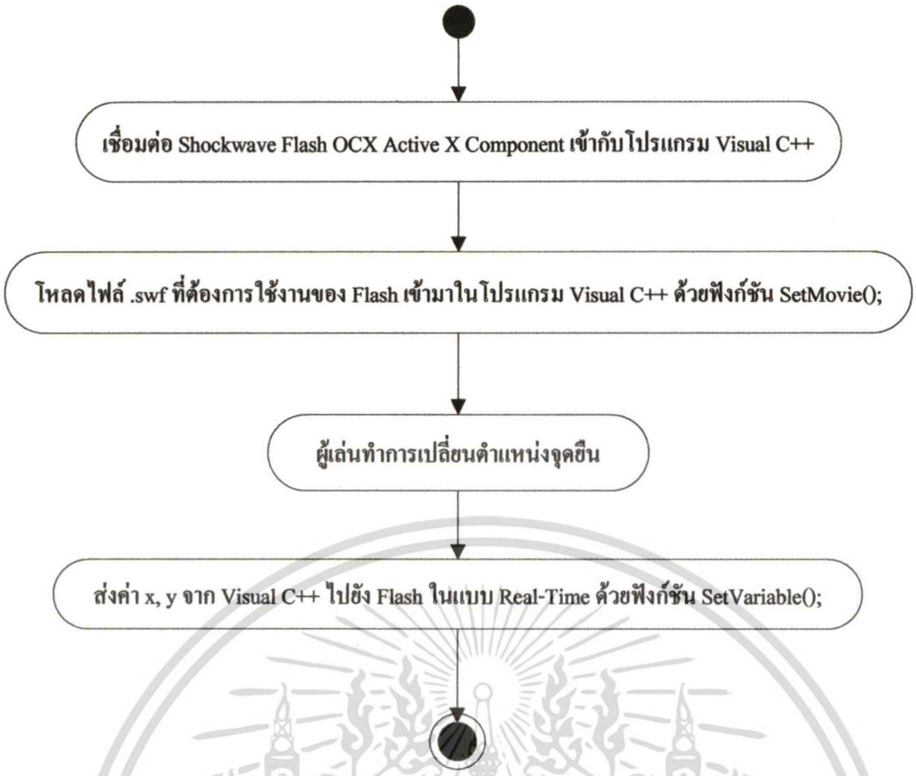
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 แอคติวิตี้ไดอะแกรม (Activity Diagrams)

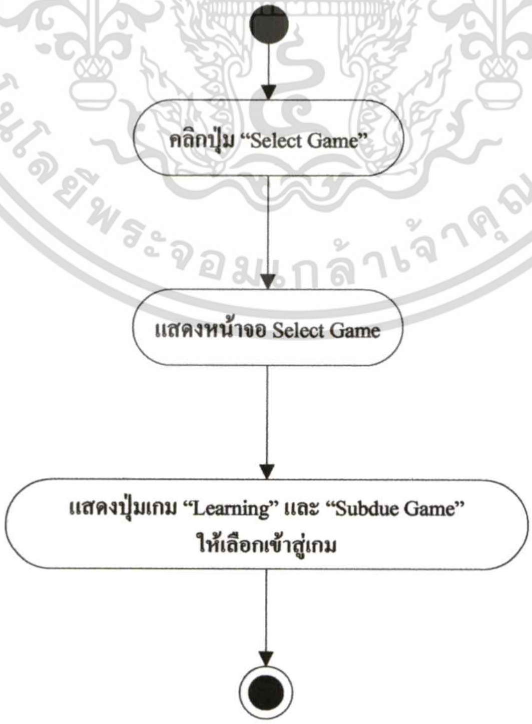


รูปที่ 5.2 แอคทิวิตี้ไดอะแกรม Leg Track Image Processing

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



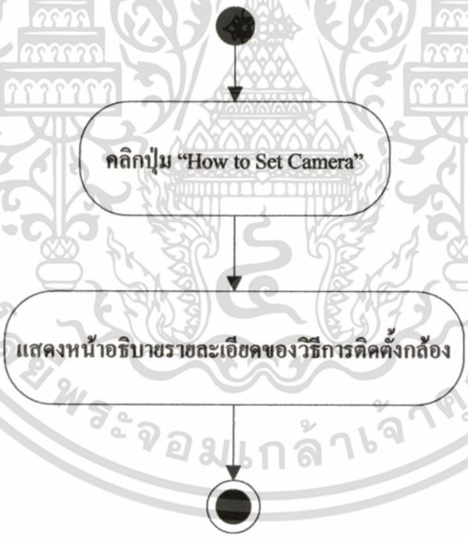
รูปที่ 5.3 แอคทิวิตี้ไดอะแกรม OCX Flash Connect



รูปที่ 5.4 แอคทิวิตี้ไดอะแกรม Select Game

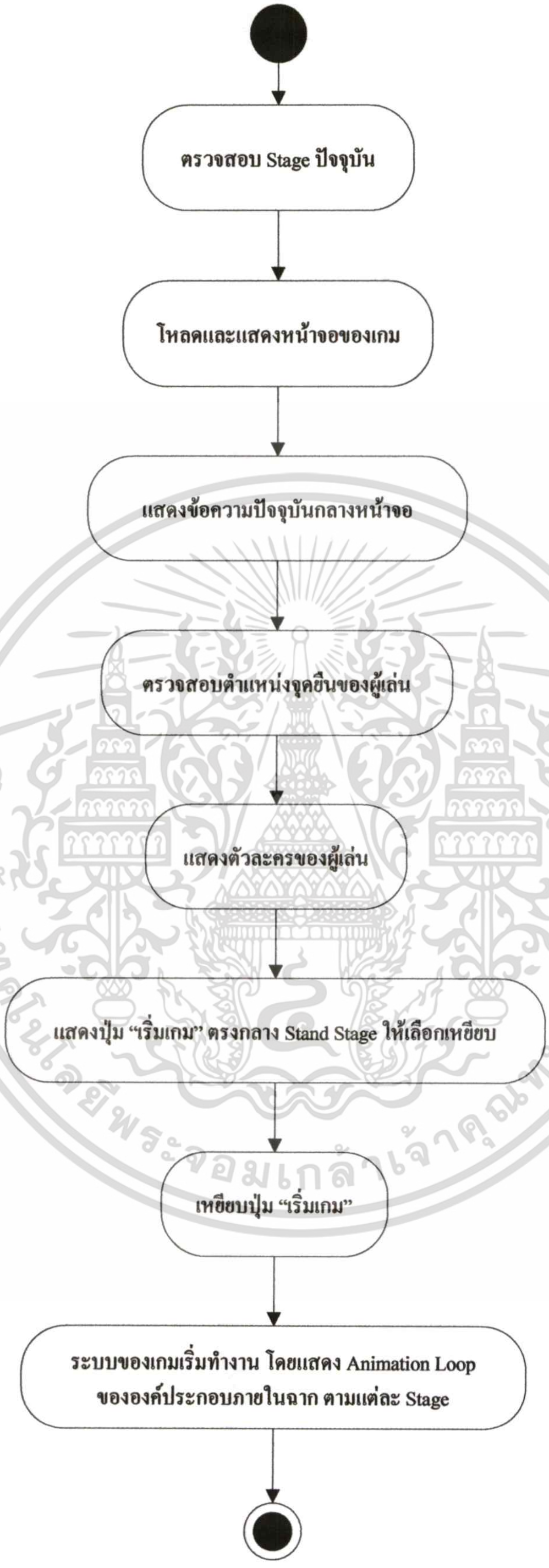
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 5.5 แอคทิวิตี้ไดอะแกรม How to Play



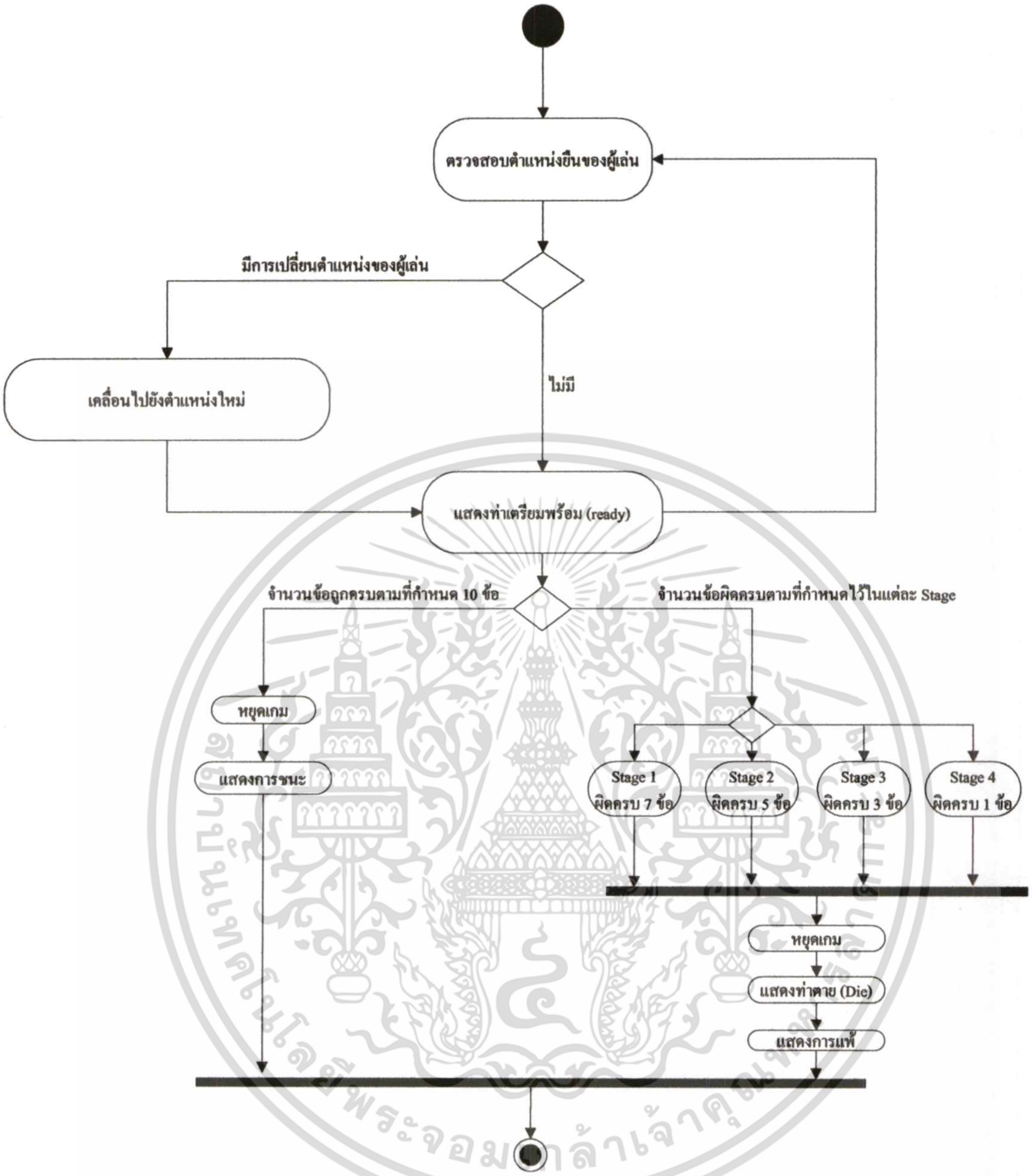
รูปที่ 5.6 แอคทิวิตี้ไดอะแกรม How to Set Camera

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



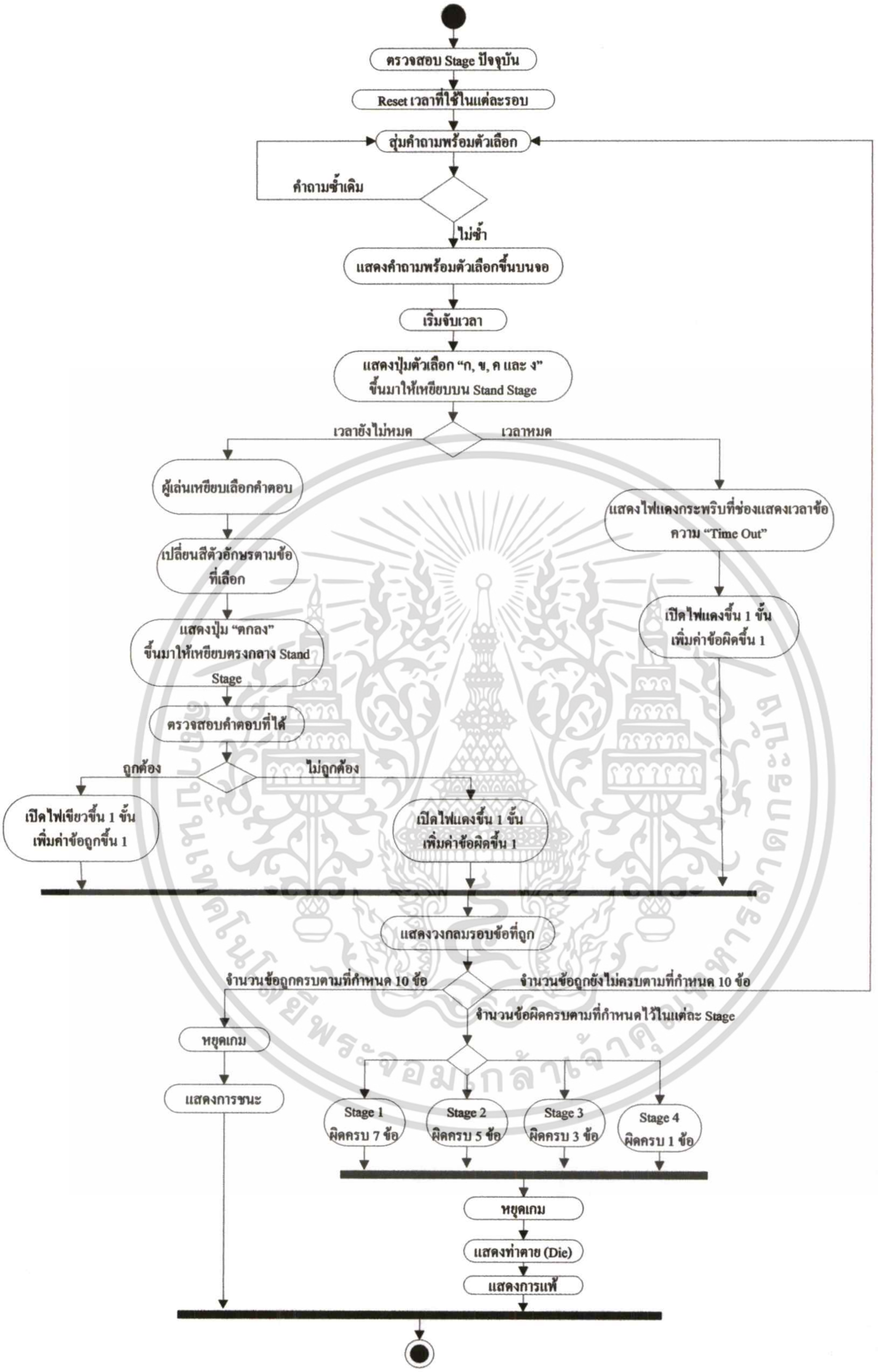
รูปที่ 5.7 แอคติวิตี้ไดอะแกรม Learning Game And Subdue Game

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



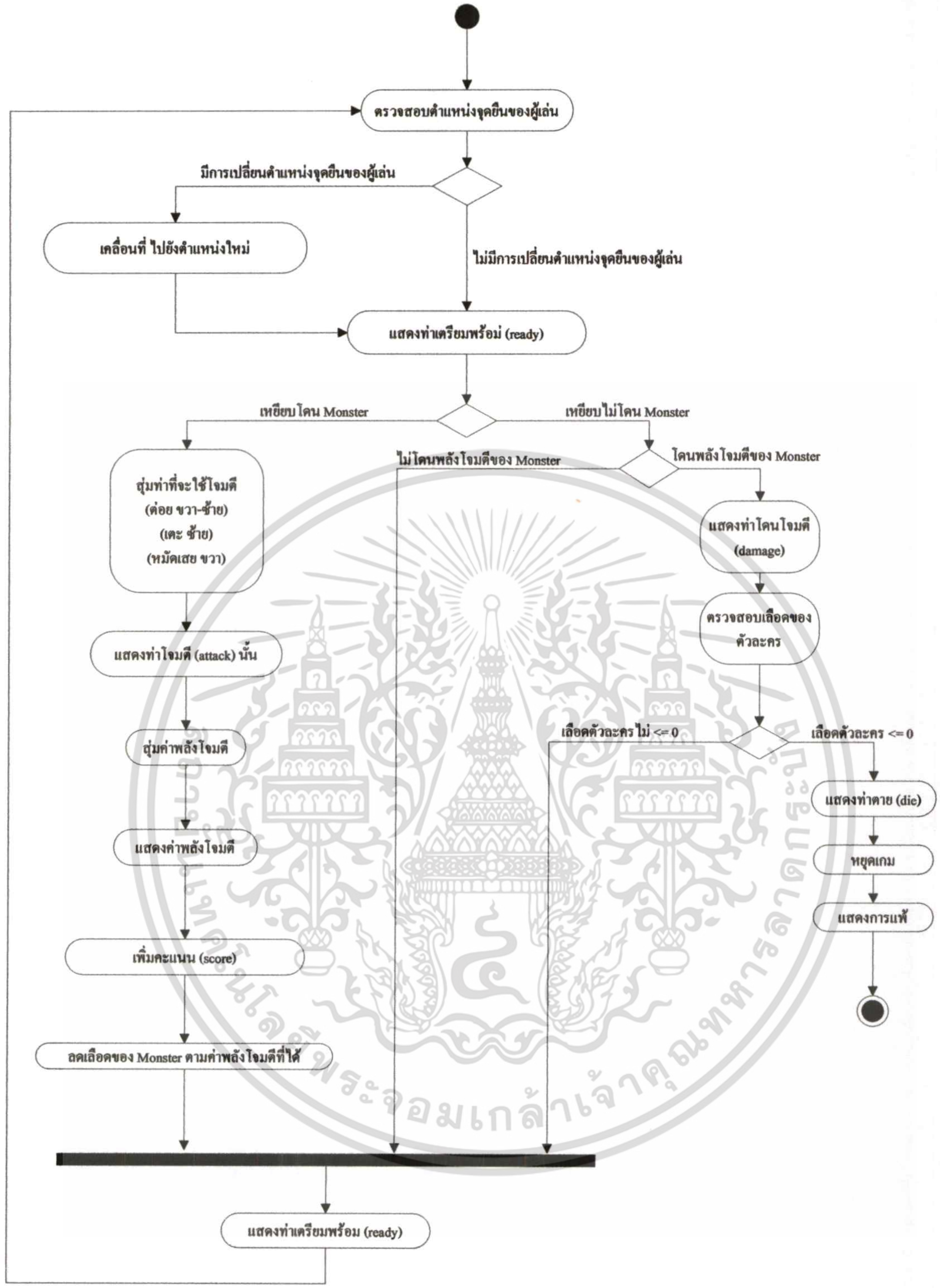
รูปที่ 5.8 แอคติวิตี้ไดอะแกรม Learning Character Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



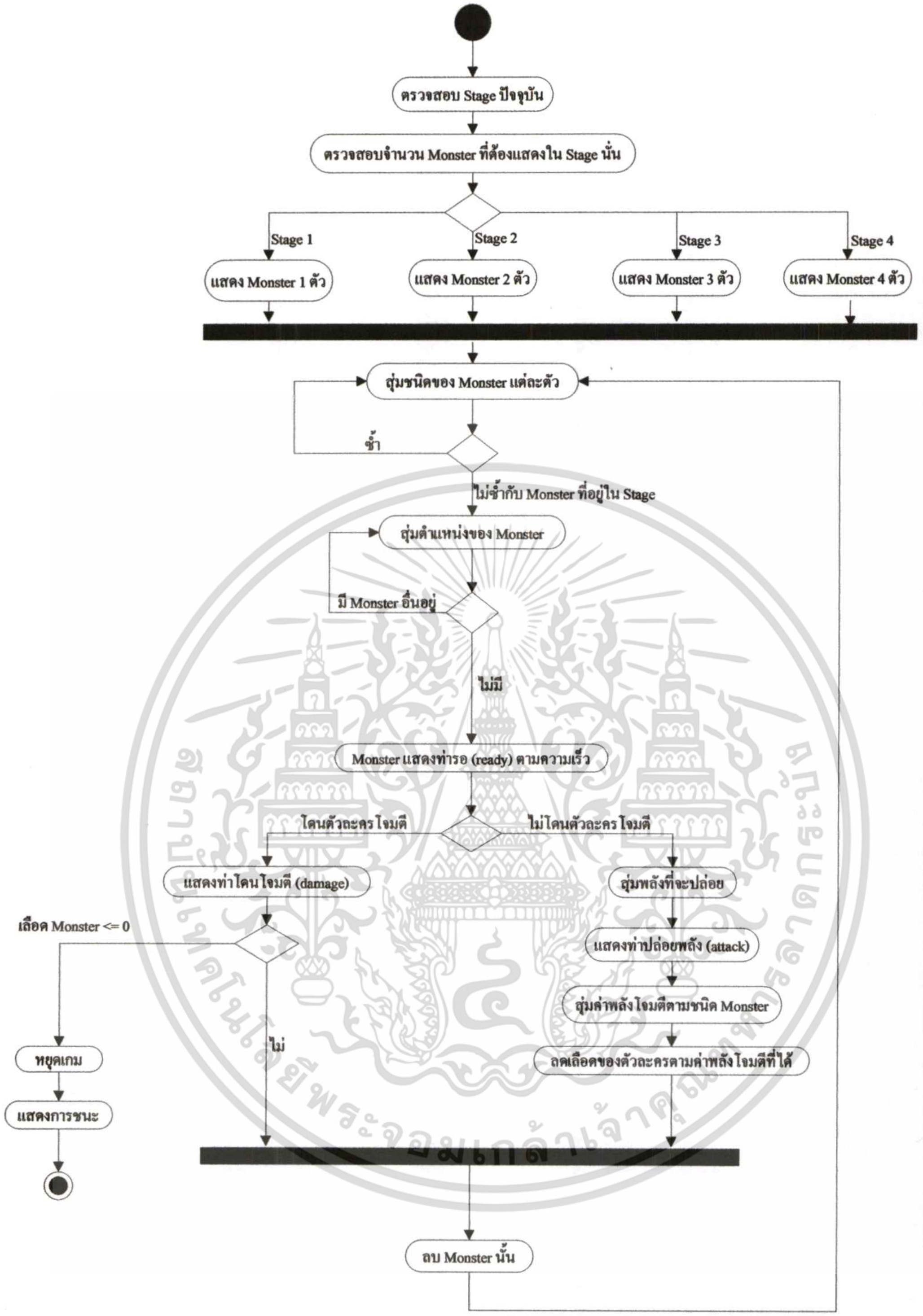
รูปที่ 5.9 แอคติวิตี้ไดอะแกรม Learning Q&A Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



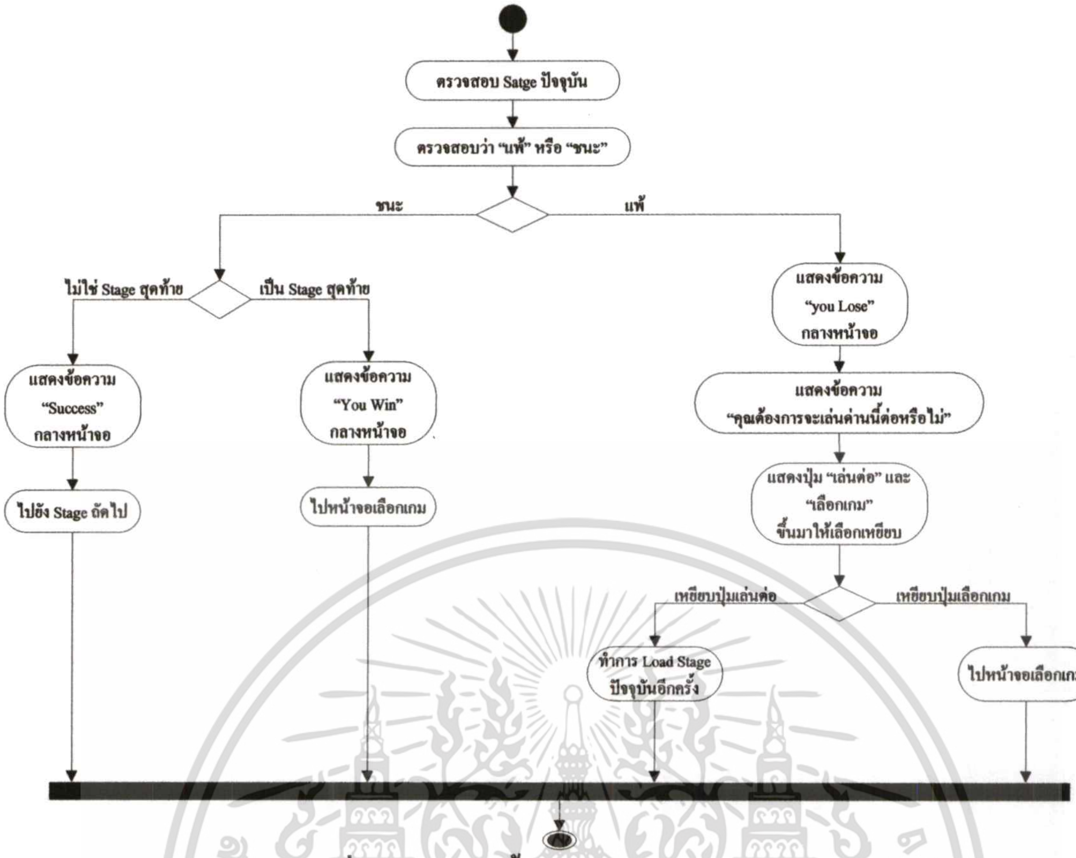
รูปที่ 5.10 แอคทิวิตี้ไดอะแกรม Subdue Character Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.11 แอคทิวิตี้ไดอะแกรม Subdue Monster Control

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.12 แอคติวิตี้ไดอะแกรม Win & Lose

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

การออกแบบและพัฒนาเกม

ในบทนี้จะกล่าวถึงการออกแบบเกมไม่ว่าจะเป็นรูปแบบของเกม วิธีการเล่นเกม เนื้อเรื่อง ตัวละคร ฉาก และ User Interface ที่จำเป็นต้องใช้ภายในเกม ซึ่งอาจจะต้องใช้เวลากันพอสมควรในการที่จะทำให้ออกมาเป็นเกมที่สมบูรณ์พร้อมได้นั้น ก็เนื่องมาจากว่าองค์ประกอบเหล่านี้ล้วน แล้วแต่เป็นองค์ประกอบที่มีความสำคัญสำหรับเกม ทั้งนี้ก็เพราะการที่เกมใดเกมหนึ่งจะประสบความสำเร็จหรือไม่นั้น ย่อมขึ้นอยู่กับความน่าสนใจ และความสนุกสนานของตัวเกม ที่ได้จากการออกแบบสิ่งต่างๆ เหล่านี้นี่เอง

6.1 ความต้องการของระบบที่ใช้ในการพัฒนาเกม

- เครื่องคอมพิวเตอร์ PC Pentium II ขึ้นไป
- กล้องวีดีโอเว็บแคม ความละเอียดของภาพวีดีโอตั้งแต่ 320 x 240 จุดภาพขึ้นไป และรองรับ USB 2.0 และ 1.1 (สำหรับโปรเจคนี้ใช้กล้องวีดีโอเว็บแคม Logitech รุ่น QuickCam Pro 4000 ในการพัฒนา)
- ระบบปฏิบัติการ Microsoft Windows 98 ขึ้นไป
- โปรแกรม Microsoft Visual C++ 6.0 และ ใช้ TLIB :Tracking Library ในการพัฒนาโปรแกรมประมวลผลภาพที่ได้จากกล้องวีดีโอเพื่อตรวจวัดตำแหน่งขา
- โปรแกรม Adobe Flash Professional CS3 เพื่อใช้ในการพัฒนาเกม Subdue Narcotic
- โปรแกรม Adobe Flash Player 9.0 เพื่อใช้เล่นไฟล์ .swf ที่ได้จากการพัฒนาเกม Subdue Narcotic

6.2 รูปแบบและวิธีการเล่นเกม

เนื่องจากเราต้องการทำเกมที่สามารถเล่นผ่านทางกล้อง Video Web Cam ในรูปแบบ Real-time ได้ ด้วยการใช้เทคนิคการประมวลผลภาพที่ได้จากกล้อง Video Web Cam โดยทำการตรวจจับตำแหน่งของผู้เล่นเอาไว้อยู่ตลอดเวลา ดังนั้นเกมที่จะสร้างจึงต้องใช้ตำแหน่งพิกัดจุดยืนของผู้เล่นเป็นตัวดำเนินเรื่องในการเล่นเกม แต่การใช้กล้อง Video Web Cam ในการเล่นนั้นก็มีขีดจำกัดอยู่ด้วย นั้นก็คือ ระยะของกล้องมีขอบเขตไม่กว้างนัก อีกทั้งความแม่นยำของค่าตำแหน่งจุดยืนที่ได้ก็ไม่เที่ยงตรงมากนักและมีการเปลี่ยนแปลงอยู่ตลอดเวลา ขนาดที่จะนำมาใช้เป็นจุดตัดสินชี้ขาดเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดๆไปได้ ดังนั้นจึงควรจะใช้ค่าที่ได้เป็นช่วงมากกว่า ซึ่งก็ขึ้นอยู่กับปัจจัยหลายอย่าง เช่น ตำแหน่งการวางเท้าของผู้เล่น ระยะห่างจากกล้อง ซึ่งการออกแบบเกมนั้นจำเป็นจะต้องให้สอดคล้องกับปัญหาเหล่านี้ด้วย

เกมแรก คือ เกมตอบคำถามเกี่ยวกับยาเสพติด ซึ่งนับเป็นก้าวแรกของการเรียนรู้เรื่องราวเกี่ยวกับยาเสพติด จึงได้ออกแบบเนื้อเรื่องโดยให้ตัวเอกหรือก็คือผู้เล่นมาทำการเรียนรู้เกี่ยวกับยาเสพติดชนิดต่างๆด้วยกัน ตัวเอกที่ถูกดัดแปลงให้ต่อสู้กับปีศาจยาเสพติดนั้น ก็ต้องรอบรู้เกี่ยวกับเรื่องราวของยาเสพติดด้วย โดยจะเรียนรู้ผ่านคำถามแบบ 4 ตัวเลือก โดยเราจะต้องตอบถูกให้ได้ 10 ข้อ และในแต่ละด่านจะสามารถผิดได้ไม่เท่ากัน ด่านแรกถ้าผิดครบ 7 ข้อ ก็จะแพ้ทันที ด่านที่ 2 ถ้าผิดครบ 5 ข้อ ก็จะแพ้ทันทีเช่นกัน ส่วนด่านที่ 3 ถ้าผิดครบ 3 ข้อ ก็จะแพ้ทันที และด่านสุดท้ายไม่สามารถผิดได้เลย

ส่วนเกมที่ 2 รูปแบบของเกมจะเป็นเกมแนวต่อสู้ (Survival) ซึ่งจะทำการต่อสู้กับเหล่าปีศาจยาเสพติด ทั้งนี้ก็เพื่อเป็นการรณรงค์ให้เห็นอันตรายของยาเสพติด โดยจะใช้วิธีการต่อสู้ด้วยการกระโดดเข้าไปสู้กับคู่ต่อสู้ซึ่งก็คือเหล่าปีศาจยาเสพติดชนิดต่างๆ ตามช่องที่กำหนด โดยจะกำหนดจำนวนช่องเป็น 9 ช่องในรูปแบบ 3x3 ซึ่งสาเหตุที่ต้องใช้วิธีการกระโดดเข้าไปเหยียบคู่ต่อสู้ก็เพราะตำแหน่งของผู้เล่นที่ได้จากการประมวลผลภาพของกล้อง Video Web Cam นั้นจะเป็นค่าเฉลี่ยระหว่างเท้าทั้ง 2 ข้าง โดยถ้าหากโจมตีด้วยการกระโดดเข้าไปเหยียบปีศาจยาเสพติด ถ้าโดนก็จะทำการโจมตี ซึ่งจะเป็นการยับยั้งการโจมตีของปีศาจยาเสพติดได้ ส่วนปีศาจยาเสพติดคู่ต่อสู้ของเรานั้น จะหายตัวแล้วไปปรากฏตามจุดต่างๆ แบบสุ่ม แล้วจะทำการโจมตีมายังเราตามระยะเวลาที่กำหนด ซึ่งความเร็วในการเข้าโจมตีมายังเรานั้นจะขึ้นอยู่กับค่าความความเร็วของปีศาจแต่ละตัว และเมื่อโจมตีเสร็จแล้วหรือถูกยับยั้งการโจมตีก็จะทำการหายตัวไปปรากฏยังที่ใหม่ ซึ่งการต่อสู้จะดำเนินเช่นนี้ไปเรื่อยๆจนว่าฝ่ายหนึ่งฝ่ายใดจะเลือดหมดแล้วตายจากไป

6.3 เนื้อเรื่องของเกม

เนื้อเรื่องของเกม คือ ตัวเอกที่ชื่อ Mark เป็นเด็กชายอายุ 15 ปี มีความเกลียดชังยาเสพติดมากเนื่องจากทั้งครอบครัวถูกคนติดยาเสพติดทำร้ายจนเสียชีวิต และมีเพียงเขาที่ถูกองค์กรต่อต้านยาเสพติดโลก(ลับ) ซึ่งเป็นองค์กรลับที่จัดตั้งขึ้นโดยกระทรวงควบคุมและป้องกันปัญหายาเสพติดระดับโลก นำมาดัดแปลงร่างกาย เพื่อให้มาต่อสู้กับปีศาจยาเสพติด ทำให้เขามีพลังทางร่างกายและจิตใจแข็งแกร่งพอที่จะต่อสู้กับปีศาจยาเสพติดได้ แต่ก่อนจะออกไปจัดการกับปีศาจยาเสพติดที่ดุร้าย Mark จะต้องทำการเรียนรู้เกี่ยวกับเรื่องของยาเสพติดชนิดต่างๆก่อน เมื่อเรียนรู้จนสามารถวางใจได้แล้ว ทางองค์กรจึงได้มอบภารกิจง่ายๆให้ก่อน โดยให้เริ่มต่อสู้กับปีศาจยาเสพติดในระดับล่างที่มีความรุนแรงไม่มากก่อน เพื่อเป็นการฝึกฝนและทดสอบความสามารถ เมื่อเริ่มมีทักษะสูงขึ้น

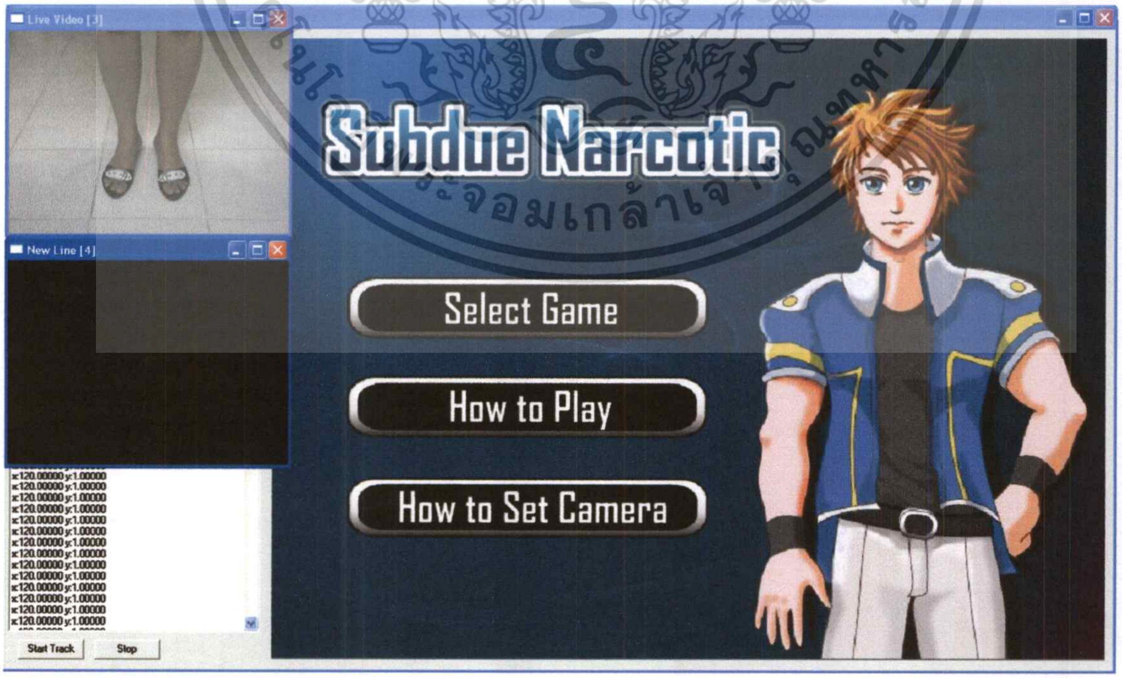
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และมีความสามารถพอแล้วก็จะพิจารณามอบหมายงานในภารกิจต่อๆไปให้ จากระดับล่างไปที่มีความรุนแรงไม่มากไปจนถึงระดับสูง ตามลำดับความรุนแรง โดยผ่านการบอกเล่าและมอบหมายจากองค์กรต่อต้านยาเสพติดโลก

เขาได้เริ่มต่อสู้กับปีศาจยาเสพติดที่มีโทษไม่ร้ายแรงนักอย่างปีศาจยาเสพติดชนิดสูบ ซึ่งเขาได้เรียนรู้จากการต่อสู้ว่า ปีศาจยาเสพติดชนิดนี้มีความร้ายกาจ ซึ่งสามารถทำให้ถึงตายได้ถ้าประมาท ซึ่งจากการที่ตัวเอกมีฝีมือที่เก่งกาจนั้นเอง ทำให้ทางองค์กรไว้วางใจและมอบหมายงานให้เขาอย่างต่อเนื่อง ด้วยการให้เขาไปกำจัดปีศาจยาเสพติดที่มีโทษร้ายแรงเพิ่มมากขึ้นตามลำดับ ซึ่งเขาเองก็ดีใจและภูมิใจเป็นอย่างมากที่จะได้ไปปราบปรามปีศาจยาเสพติดที่เป็นศัตรูตัวฉกาจเหล่านั้น ดังนั้นเขาจึงดั้นด้นออกเดินทางเพื่อไปกำจัดเหล่าปีศาจยาเสพติดเหล่านั้นให้สิ้นซาก ในฉากแรกเหล่าปีศาจยาเสพติดยังดูแคลน Mark มาก เพราะพวกมันผลัดกันออกมาสู้กับเขาแค่คราวละตัว และเมื่อ Mark สามารถเอาชนะได้ พวกมันจึงค่อยเพิ่มระดับของความยากขึ้น จนสุดท้าย Mark จะต้องต่อสู้กับพวกปีศาจยาเสพติดด้วยความยากระดับสูงสุด อนาคตของมนุษยชาติขึ้นอยู่กับการต่อสู้ในครั้งนี้แล้ว

6.4 รายละเอียดของส่วนหน้าจอ

6.3.1 หน้าจอเริ่มแรกและหน้าจอแสดงการประมวลผลภาพจากกล้องวิดีโอ



รูปที่ 6.1 แสดงหน้าจอเริ่มแรกและหน้าจอแสดงการประมวลผลภาพจากกล้องวิดีโอ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราเลือกที่ Start Track ที่มุมซ้ายล่างแล้ว กล้องจะทำการบันทึกภาพของพื้นหลัง และทำการตรวจจับตำแหน่งของผู้เล่น โดยใช้การประมวลผลภาพที่ได้จาก Video Web Cam และจะเป็นการเข้าสู่หน้าหลักของเกมทันที

รูปที่ 6.2 แสดงหน้าหลักของเกม (Main Menu)

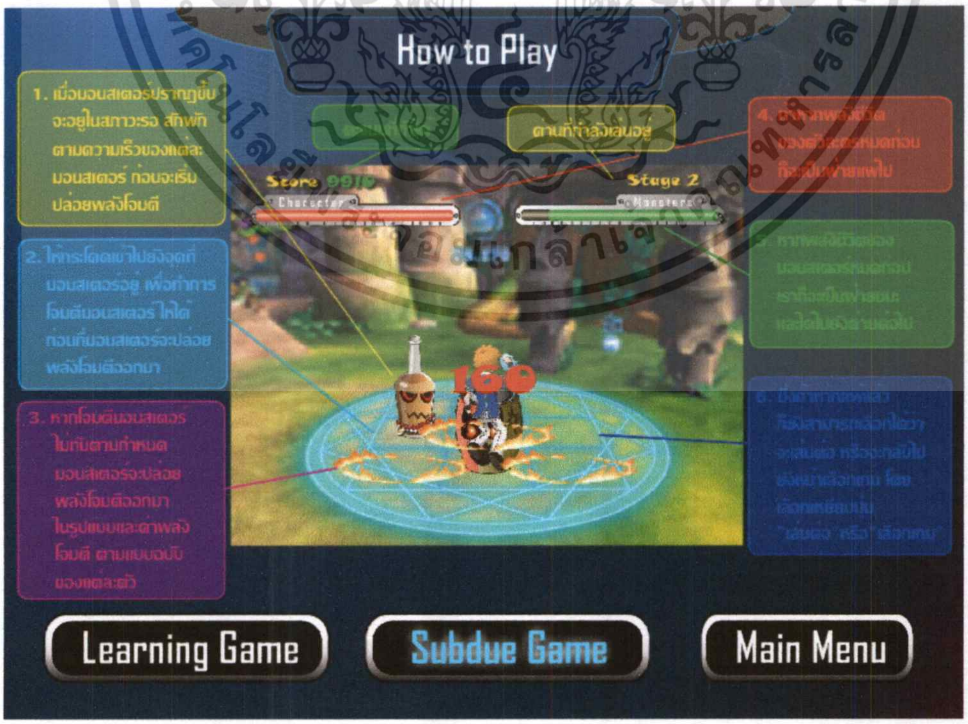
หน้านี้จะแสดงชื่อเกม Subdue Narcotic และตัวละคร Mark แบบครึ่งตัว และแสดง Menu ของเกม เมื่อเราเลือก Select Game จะเข้าสู่หน้าเลือกเกม ซึ่งมีทั้งหมด 2 เกม เมื่อเราเลือก How to Play ก็จะเข้าสู่หน้าสอนวิธีการเล่นเกม เมื่อเราเลือก How to Set Camera จะเข้าสู่หน้าสอนวิธีการติดตั้งและการวางตำแหน่งกล้อง

รูปที่ 6.3 แสดงภาพการเลือก How to Play

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเลือกที่ How to Play ระบบจะทำการแสดงหน้าจออธิบายการเล่น สำหรับมือใหม่หรือคนที่สนใจ สามารถศึกษาหน้าจอและวิธีการเล่นเบื้องต้นได้จากที่นี่

รูปที่ 6.4 แสดงหน้าจอ How to Play ของ Learning Game

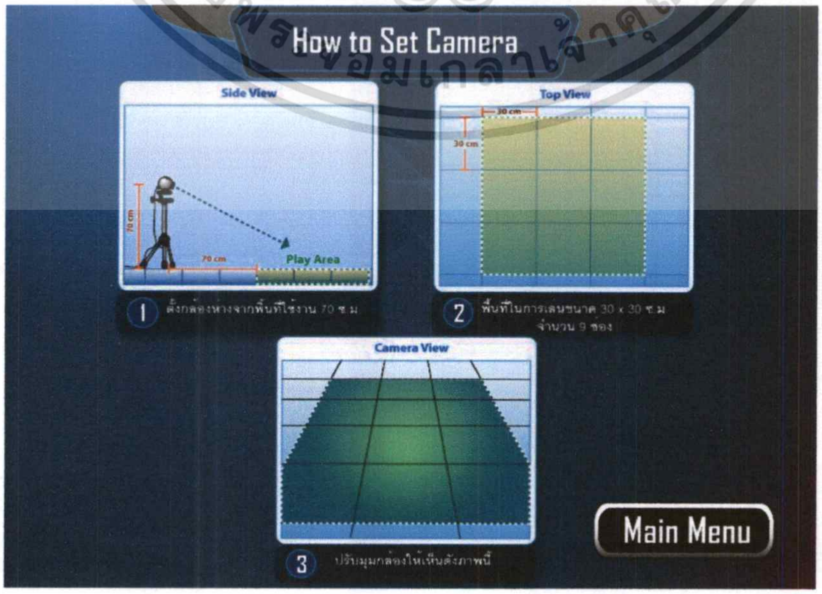


รูปที่ 6.5 แสดงหน้าจอ How to Play ของ Subdue Game

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.6 แสดงภาพการเลือก How to Set Camera

เมื่อเราเลือกที่ How to Set Camera แล้ว ระบบจะแสดงหน้าจอการจัดตั้งกล้องให้ทราบทันที ซึ่งมีรายละเอียดดังนี้ คือ ให้ตั้งกล้องไว้ด้านหน้า หันหน้ากล้องเข้าหาตัวเรา จากนั้นตั้งกล้องให้สูงจากพื้น 70 ซม. และห่างจากพื้นที่ที่จะใช้เล่น 70 ซม. โดยพื้นที่ที่จะใช้เล่นนั้น มีขนาด 90 x 90 ซม. ซึ่งจะแบ่งเป็น 9 ช่อง ขนาดช่องล่ะ 30 x 30 ซม. ให้ทำการตั้งกล้องไว้ตรงจุดกึ่งกลางพื้นที่ที่จะใช้เล่น แล้วปรับมุมกล้องให้เหมือนกับข้อ 3 ในรูปที่ 6.7 โดยให้เส้นกั้นแนวขวางเส้นแรกห่างจากขอบล่างของภาพที่เห็นในกล้องนิดหนึ่ง และให้ขอบเส้นกั้นแนวขวางเส้นที่ 2 อยู่ชิดขอบขวาและขอบด้านซ้ายพอดี



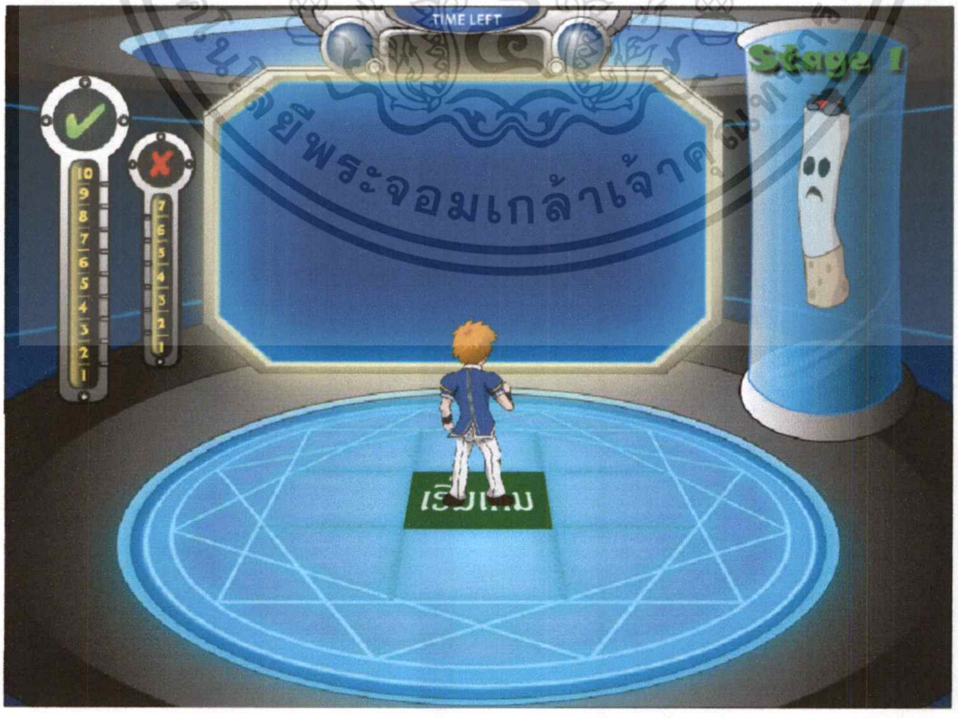
รูปที่ 6.7 แสดงภาพ How to Set Camera

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการให้บริการเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3.2 Learning Game

ตัวเกมจะมีทั้งหมด 2 เกม คือเกม Learning Game และ Subdue Game เมื่อเลือกเกมแรก จะเข้าสู่ตัวเกม Learning Game ทันที

รูปที่ 6.8 แสดงการเลือกเล่นเกม Learning Game จากหน้าเลือกเกม



รูปที่ 6.9 แสดงการเหยียบปุ่ม "เริ่มเกม" เพื่อให้เกม Learning Game เริ่มทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยหน้าจอของเกม Learning Game จะแบ่งออกเป็น 4 ส่วน ส่วนแรกทางซ้ายมือสุดจะเป็นแถวเครื่องหมายถูก ถ้าเราตอบถูกไฟสีเขียวจะขึ้น 1 ดวง ซึ่งเราจะต้องตอบให้ถูก 10 ข้อต่อ 1 ฉาก ถัดมาจากแถวเครื่องหมายผิด ซึ่งจำนวนการตอบผิดจะขึ้นอยู่กับฉาก อย่างด่านแรกถ้าตอบผิดครบ 7 ข้อจะถือว่าแพ้ทันที ด่านที่ 2 ถ้าตอบผิดครบ 5 ข้อจะถือว่าแพ้ทันที ส่วนด่านที่ 3 ถ้าตอบผิดครบ 3 ข้อจะถือว่าแพ้ทันที และด่านที่ 4 ถ้าตอบผิดแม้แต่ข้อเดียวจะถือว่าแพ้ทันที

ส่วนที่ 2 จอตรงกลางฉาก จอนี้จะเป็นการแสดงคำถามและคำตอบ และเมื่อเราเลือกคำตอบ คำตอบจะเป็นสีเหลือง และจะเฉลยข้อที่ถูก เป็นสีชมพูให้ผู้เล่นได้ทราบหลังจากตรวจสอบคำตอบจากที่ผู้เล่นได้เลือกไปแล้วว่าถูกหรือผิด ส่วนด้านบนของจอคำถามและคำตอบจะมีนาฬิกาคอยจับเวลาอยู่ ซึ่งจะให้เวลา 20 วินาทีต่อ 1 คำถาม เมื่อนับถอยหลังจนถึง 0 จะถือว่าผู้เล่นตอบผิดทันที

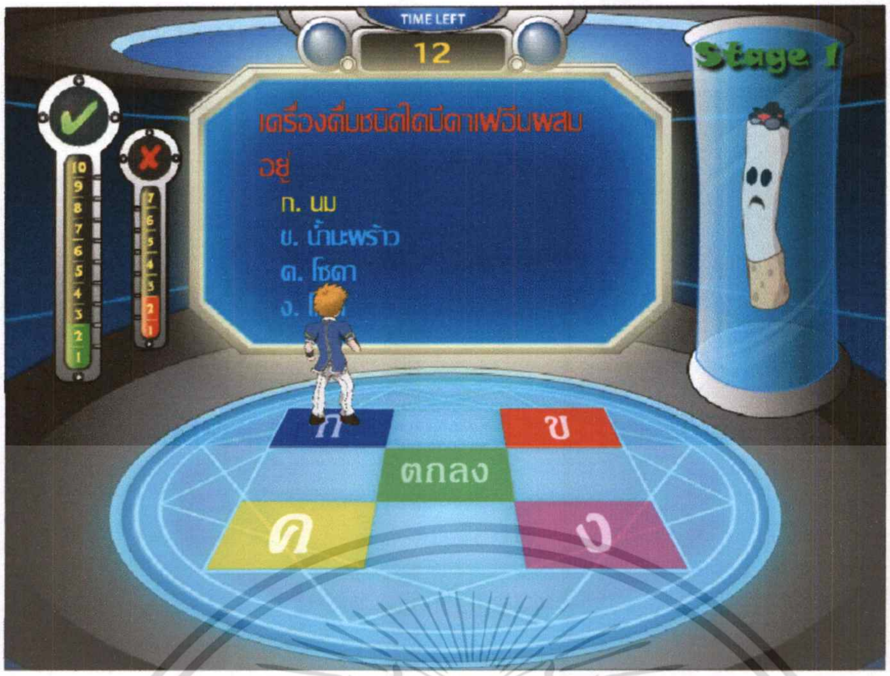
ส่วนที่ 3 เป็นส่วนการแสดง Stage ปัจจุบันที่เล่น และปีศาจยาเสพติดเจ้าของด่าน

ส่วนที่ 4 จะเป็นส่วนที่ผู้เล่นใช้เลือกคำตอบที่ถูกต้อง ก, ข, ค และ ง เมื่ออ่านคำถามและคำตอบในส่วนที่ 2 แล้ว เราจะต้องเลือกคำตอบที่ถูกต้อง หลังจากมั่นใจในคำตอบและจะไม่เปลี่ยนแปลงแล้ว จึงกลับมาเหยียบปุ่ม ตกลง เกมจะทำการเฉลยคำตอบข้อที่ถูกเป็นสีชมพู ถ้าเราตอบถูก ไฟสีเขียวจะขึ้นที่แถบเครื่องหมายถูก แต่ถ้าเราตอบผิด ไฟสีแดงจะขึ้นที่แถบเครื่องหมายผิด

การเล่นเกมเราจะบังคับ Mark ตัวเอกของเกมนี้ โดยให้ Mark เลือกคำตอบที่ถูกต้อง เพื่อเพิ่มพูนความรู้ในการจัดการกับพวกปีศาจยาเสพติดต่างๆ

รูปที่ 6.10 แสดงการถามคำถามและคำตอบของเกม Learning Game

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.11 แสดงการเหยียบเลือกคำตอบ และแสดงปุ่ม ตกลง เพื่อยืนยันข้อที่เลือก

เมื่อเรามาเหยียบปุ่มตกลงตรงกลาง จะเป็นการยืนยันว่าเราตกลงเลือกข้อนั้นแล้ว และจะไม่มีเปลี่ยนแปลงคำตอบอีก ระบบก็จะนำข้อที่เลือกนั้นไปตรวจสอบว่าถูกหรือผิด

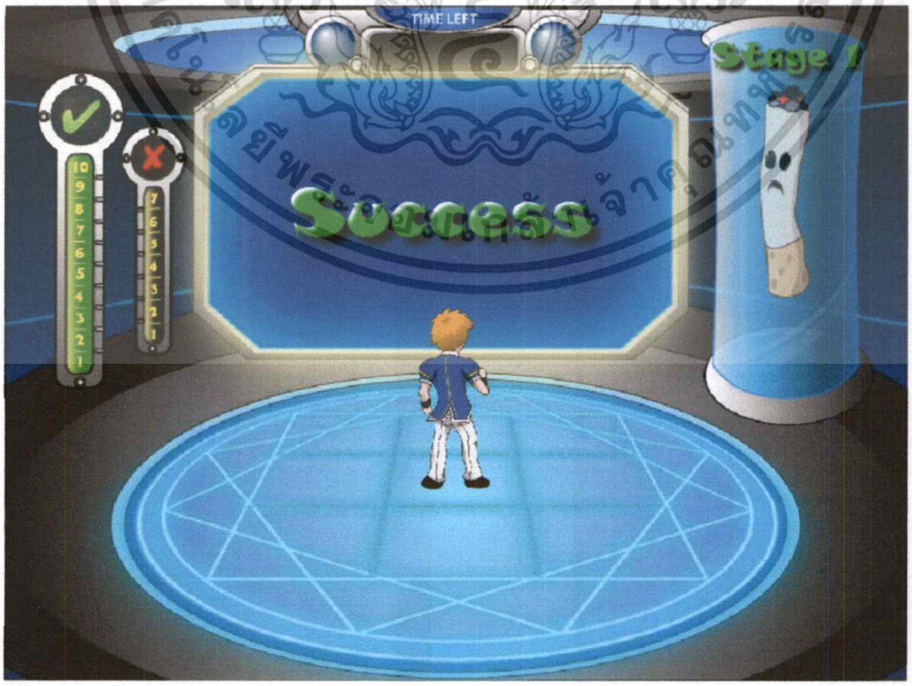
หลังจากนั้นก็จะเฉลยข้อที่ถูกให้เราทราบเป็นตัวอักษรสีชมพู โดยระบบจะให้เวลาในการคิดแต่ละข้อเท่ากับ 20 วินาที เพราะฉะนั้นจะต้องรู้จักใช้เวลาให้เป็นประโยชน์ เพราะถ้าตอบไม่ทันภายในเวลาที่กำหนด จะถือว่าตอบผิดทันที

รูปที่ 6.12 แสดงการเฉลยคำตอบที่ถูกต้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ... ให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

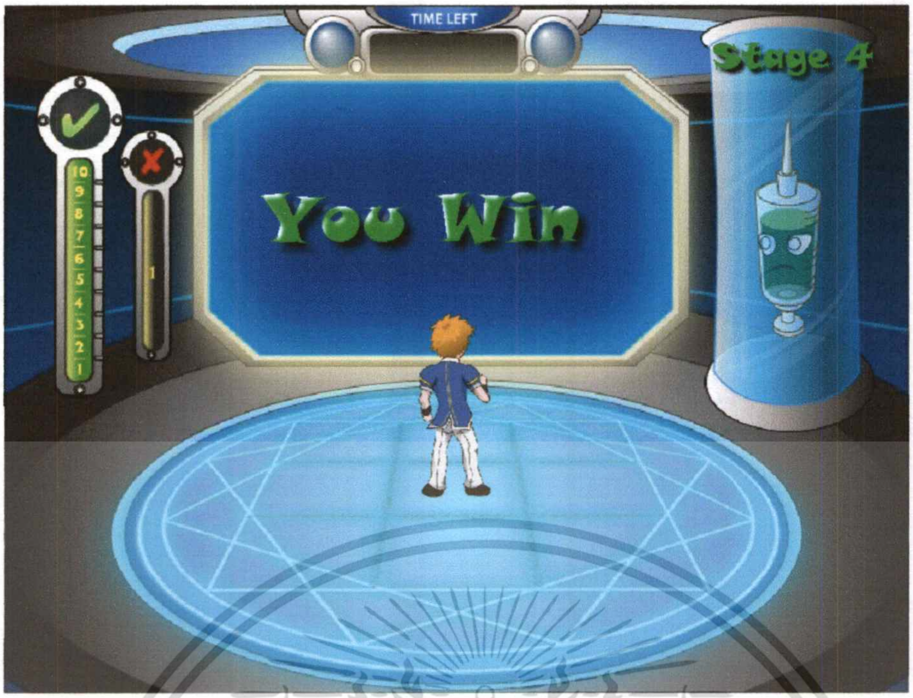
รูปที่ 6.13 แสดง Time Out เมื่อเวลาในการตอบคำถามของแต่ละข้อหมด

เมื่อเราตอบถูกครบ 10 ข้อแล้ว ระบบจะแสดงหน้าจอของการชนะ โดยจะแสดงข้อความ Success ขึ้นมากลางหน้าจอ และจะผ่านไปสู่ฉากต่อไปทันที แต่ถ้าหากด่านที่ผ่านเป็นด่านสุดท้ายแล้วจะแสดงข้อความ You Win ขึ้นมากลางหน้าจอแทน แล้วจะไปยังหน้าเลือกเกมแทน



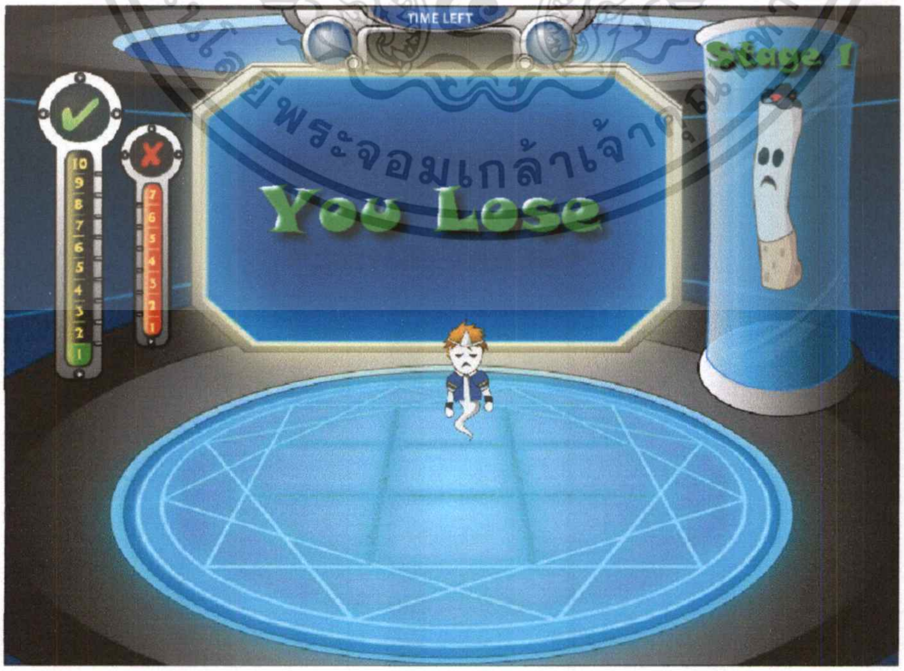
รูปที่ 6.14 แสดงการชนะ โดยจะขึ้นข้อความ Success

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.15 แสดงการชนะ โดยจะขึ้นข้อความ You Win เมื่อชนะในด่านสุดท้าย

แต่ถ้าตอบผิดครบตามที่กำหนดไว้ในแต่ละด่านแล้ว เราแพ้ทันที โดยระบบจะแสดงข้อความ You Lose ขึ้นมากลางหน้าจอ แล้วจากนั้นจึงแสดงหน้าจอตัวเลือกหลังพ่ายแพ้ ถ้าเราเลือก "เล่นต่อ" เราได้กลับมาเล่นแก้ตัวยังฉากที่เราพ่ายแพ้ ถ้าเราเลือก "เลือกเกม" ระบบจะกลับมาที่หน้าเลือกเกมให้ทันที

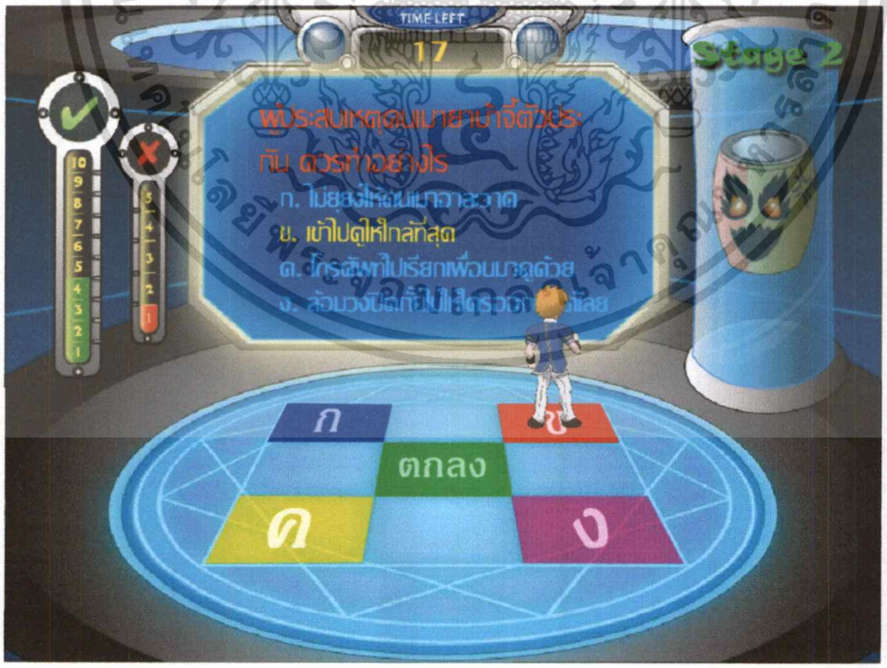


รูปที่ 6.16 แสดงการแพ้ โดยจะขึ้นข้อความ You Lose

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.17 แสดงหน้าจอตัวเลือกหลังพ่ายแพ้

จากที่ผ่านมาในด่านแรก เราจะเห็นได้ว่าเราจะแพ้ทันที เมื่อตอบผิดครบ 7 ข้อ แต่ในด่านที่ 2 นั้น ถ้าตอบผิดครบ 5 ข้อจะถือว่าแพ้ทันที



รูปที่ 6.18 แสดงหน้าจอของฉากที่ 2 และแสดงมอนสเตอร์เจ้าของด่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนในด่านที่ 3 นั้น ถ้าตอบผิดครบ 3 ข้อจะถือว่าแพ้ทันที

รูปที่ 6.19 แสดงหน้าจอของฉากที่ 3 และแสดงมอนสเตอร์เจ้าของด่าน

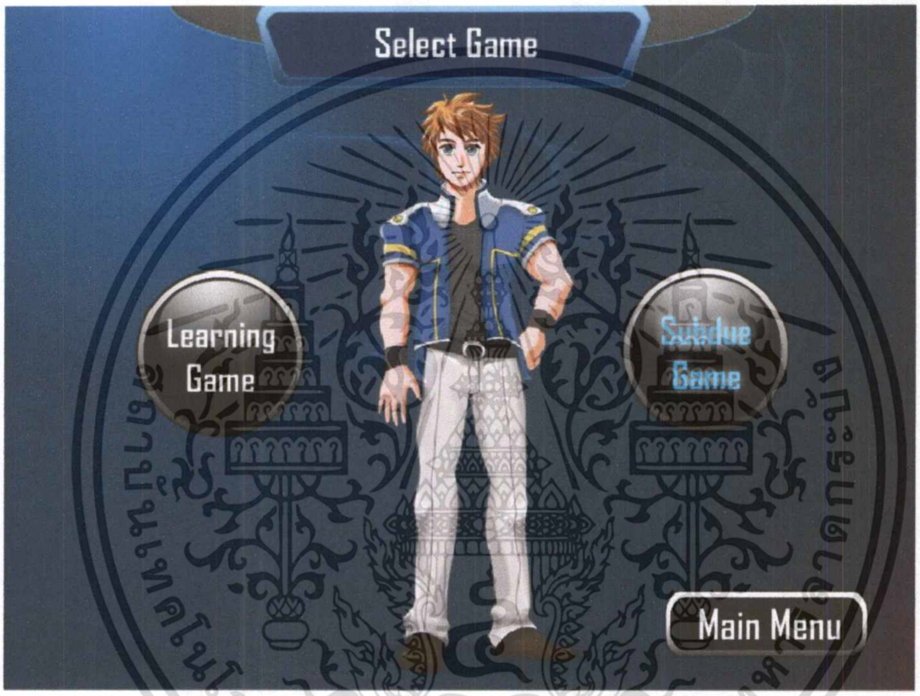
ส่วนในด่านที่ 4 นั้น ถ้าตอบผิดถือว่าแพ้ในทันที

รูปที่ 6.20 แสดงหน้าจอของฉากที่ 4 และแสดงมอนสเตอร์เจ้าของด่าน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเราจะสรุปได้ว่า เกม Learning Game นั้นจะมีทั้งหมด 4 ฉาก ความยากและความง่ายจะอยู่ที่จำนวนของการตอบผิด ฉากแรกถ้าตอบผิดครบ 7 ข้อจะถือว่าแพ้ ฉากที่ 2 ถ้าเราตอบผิดครบ 5 ข้อจะถือว่าแพ้ ฉากที่ 3 ถ้าเราตอบผิดครบ 3 ข้อจะถือว่าแพ้ และฉากที่ 4 ฉากสุดท้าย ถ้าตอบผิดจะถือว่าแพ้เลยทันที เพราะฉะนั้นถ้าเราใส่ใจกับถำถามทั้งหมดที่ผ่านตามา เราจะสามารถผ่านเกม Learning Game ได้ไม่ยากเย็นอย่างที่คิดเลย

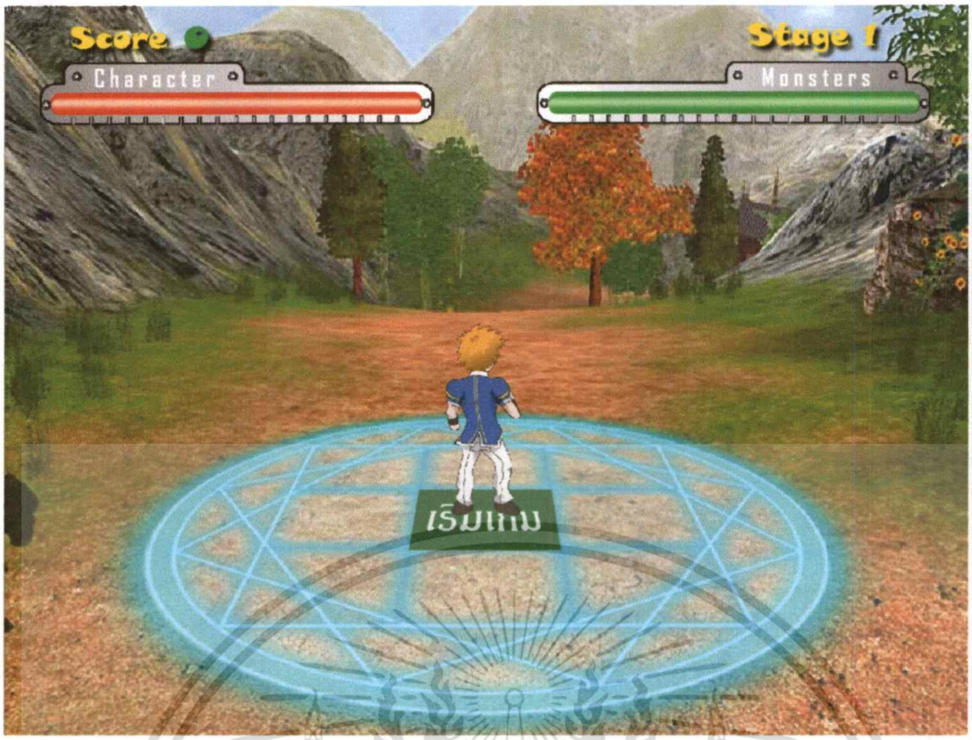
6.3.3 Subdue Game



รูปที่ 6.21 แสดงการเลือกเล่นเกม Subdue Game

เกมที่ 2 จะเป็นเกมต่อสู้ เราจะต้องบังคับ Mark ให้ไปเหยียบพวกปีศาจยาเสพติด ก่อนที่พวกมันจะปล่อยพลังออกมาทำร้าย ปีศาจยาเสพติดแต่ละชนิดก็จะมีพลังให้การโจมตีแตกต่างกันไป ความรุนแรงก็จะขึ้นอยู่กับแต่ละชนิดด้วย เกม Subdue Game นี้ จะมีด้วยกันทั้งสิ้น 4 ฉาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 6.22 แสดงการเหยียบปุ่ม "เริ่มเกม" เพื่อให้เกม Subdue Game เริ่มทำงาน

เมื่อเข้าสู่เกมแล้วจะปรากฏปุ่ม "เริ่มเกม" ขึ้นมาให้ผู้เล่นเหยียบ เพื่อให้ผู้เล่นมายืนประจำที่เตรียมพร้อมก่อน เมื่อพร้อมแล้ว ก็ให้เหยียบปุ่ม "เริ่มเกม" ระบบของเกมก็จะเริ่มทำงานในทันที

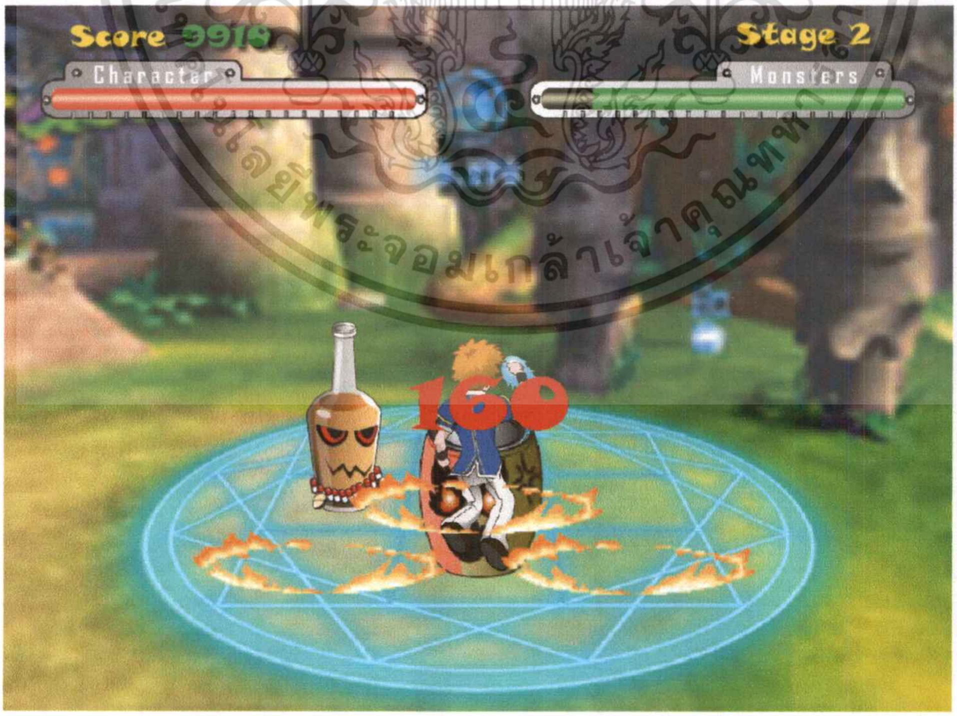
หน้าจอของเกม Subdue Game จะแบ่งออกเป็น 3 ส่วน ส่วนที่ 1 ทางซ้ายบนสุด จะมีแถบพลังสีแดง แถบนี้คือแถบพลังชีวิตของ Mark ถ้าหมดเมื่อไรก็จะแพ้ทันที และด้านบนของแถบพลังชีวิตของ Mark คือคะแนนที่ Mark สามารถทำได้จากการต่อสู้กับปีศาจยาเสพติด ส่วนที่ 2 ทางขวาบนจะเป็นแถบพลังสีเขียวคือพลังชีวิตของพวกปีศาจยาเสพติด และด้านบนของแถบพลังชีวิตของปีศาจยาเสพติดนั้น ก็จะเป็นตัวเลขแสดงฉากที่เล่นอยู่ ณ ปัจจุบัน และส่วนที่ 3 วงแหวนเวทย์กลางจอ ส่วนนี้จะเป็นส่วนของพื้นที่เล่นของผู้เล่น โดยปีศาจยาเสพติดจะผลัดกันโผล่มาโจมตีเป็นระยะๆ ตามจุดต่างๆ เมื่อผู้เล่นกระโดดไปยังจุดที่ปีศาจยาเสพติดอยู่ก็จะเป็นการโจมตีปีศาจเหล่านั้น

Subdue Game จะแบ่งออกเป็นทั้งหมด 4 ฉาก ไม่มีเวลากำหนด แต่ความยากของเกมนี้ ก็คือผู้เล่นจะต้องทำการโจมตีปีศาจยาเสพติดเหล่านั้นให้ทัน ก่อนที่จะถูกปีศาจยาเสพติดเหล่านั้นโจมตีกลับมา โดยเมื่อปีศาจยาเสพติดมาปรากฏอยู่บนหน้าจอ ให้ทำการกระโดดไปยังจุดที่ปีศาจยาเสพติดอยู่ ถ้าทัน Mark จะทำการโจมตีปีศาจยาเสพติดนั้นๆทันที ซึ่งท่าทางการโจมตีของ Mark นั้นจะมีทั้งต่อย เตะ และเสย โดยระบบจะแสดงออกมาแบบสุ่ม ส่วนพวกปีศาจยาเสพติดนั้นจะมีรูปแบบการโจมตี ระยะการในการโจมตี ที่แตกต่างกันออกไป ซึ่งบางตัวก็อาจจะมีการโจมตีอยู่หลายรูปแบบ ดังนั้นผู้เล่นจึงต้องคอยทำการสังเกตุให้ดีๆด้วย สำหรับในส่วนค่าความรุนแรงของพลังโจมตีของปีศาจยาเสพติด ก็จะเป็นไปตามชนิดความรุนแรงของยาเสพติดที่ปีศาจยาเสพติดนั้นเป็นตัวแทนด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยฉากที่ 1 จะมีปีศาจยาเสพติดพลัดกันสุ่มออกมาคราวละ 1 ตัว และฉากที่ 2 จะค่อยๆเพิ่มระดับความยากโดยจะมีปีศาจยาเสพติดพลัดกันสุ่มออกมาคราวละ 2 ตัว

รูปที่ 6.23 แสดงฉากที่ 1 ซึ่งจะมีปีศาจยาเสพติดออกมาคราวละตัว

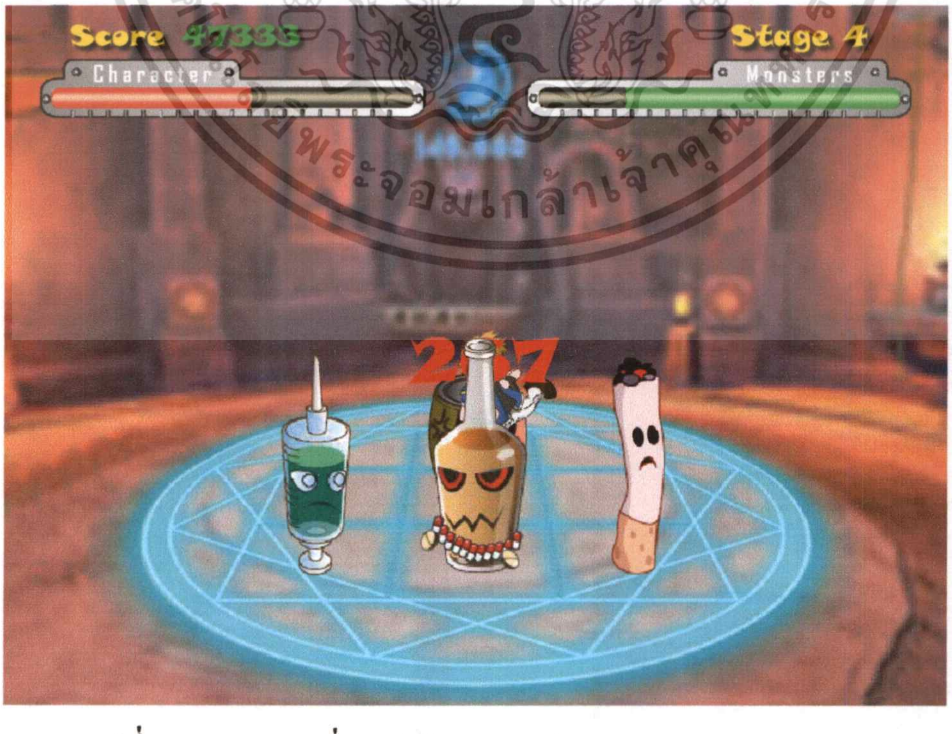


รูปที่ 6.24 แสดงฉากที่ 2 จะมีปีศาจยาเสพติดออกมาคราวละ 2 ตัว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนในฉากที่ 3 จะมีปีศาจยาเสพติดพลัดกันสุ่มออกมาคราวละ 3 ตัว และฉากที่ 4 ซึ่งเป็นฉากสุดท้ายจะมีปีศาจยาเสพติดพลัดกันสุ่มออกมาคราวละ 4 ตัวเลยทีเดียว ซึ่งจะเห็นได้ว่าความยากจะเพิ่มขึ้นอย่างมากมายเลยทีเดียว

รูปที่ 6.25 แสดงฉากที่ 3 จะมีปีศาจยาเสพติดออกมาคราวละ 3 ตัว



รูปที่ 6.26 แสดงฉากที่ 4 จะมีปีศาจยาเสพติดออกมาคราวละ 4 ตัวเลย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อเราเอาชนะได้ระบบจะแสดงหน้าจอของการชนะ โดยจะแสดงข้อความ Success ขึ้นมากลางหน้าจอ และจะผ่านไปสู่ฉากต่อไปทันที แต่ถ้าหากด่านที่ผ่านเป็นด่านสุดท้ายแล้วจะแสดงข้อความ You Win ขึ้นมากลางหน้าจอแทน แล้วจะไปยังหน้าเลือกเกม

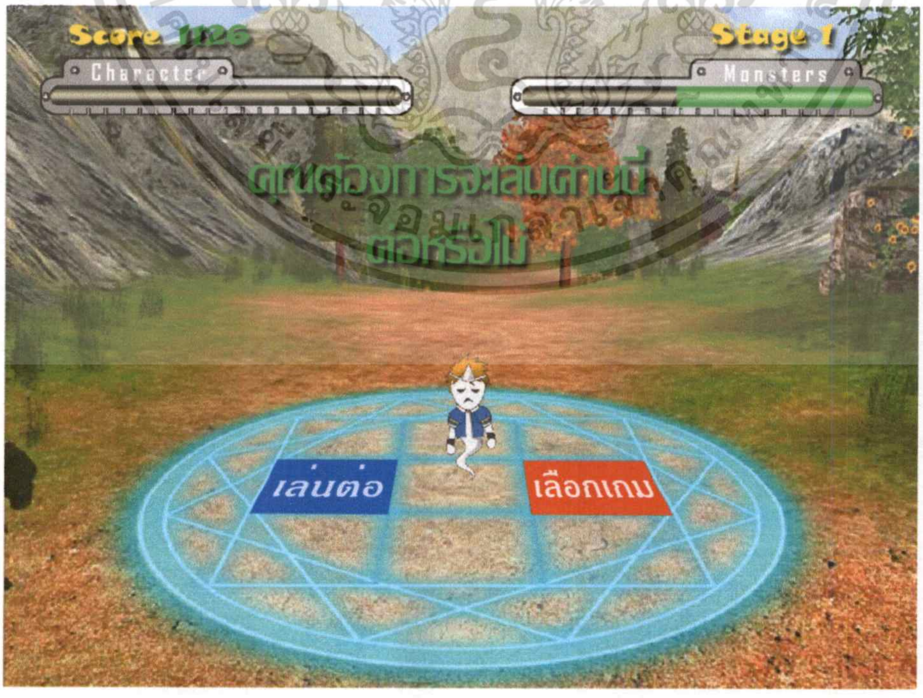
รูปที่ 6.27 แสดงข้อความ Success เมื่อต่อสู้ชนะในด่านแต่ละด่าน

รูปที่ 6.28 แสดงข้อความ You Win เมื่อต่อสู้ชนะในด่านสุดท้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แต่ถ้าเราถูกปีศาจยาเสพติดรุมทำร้ายจนพ่ายแพ้ เราจะมีโอกาสแก้ตัวได้ไม่จำกัด เรียนรู้เรื่องรัศมีโจมตีของปีศาจยาเสพติดแต่ละชนิดให้ดีๆ แล้วเราจะสามารถเอาชนะมันได้อย่างง่ายๆ และสิ่งที่สำคัญที่สุดก็คือ เราจะต้องตื่นตัวอยู่เสมอ มิเช่นนั้นเราจะต้องตกเป็นฝ่ายถูกโจมตีอยู่ฝ่ายเดียว

รูปที่ 6.29 รูปแสดงการแพ้



รูปที่ 6.30 แสดงหน้าจอตัวเลือกหลังพ่ายแพ้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7
สรุปผลการวิจัยและข้อเสนอแนะ

7.1 สรุปผลการพัฒนาระบบ

ในโครงการฉบับนี้ ได้นำเสนอการประยุกต์ใช้งานอัลกอริทึม ในการตรวจวัดตำแหน่งขาด้วยกล้องวีดีโอ โดยอาศัยหลักการประมวลผลภาพจากกล้องวีดีโอ ผ่านทางเกม "Subdue Narcotic" หรือ เกมปราบปรามยาเสพติด เพื่อแสดงให้เห็นว่า เราสามารถนำหลักการและวิธีประมวลผลภาพจากกล้องวีดีโอ Web Cam ราคาถูก ที่มีใช้ทั่วไปตามบ้านในปัจจุบัน มาประยุกต์ใช้กับเกมที่อยู่ต่าง Platform ต่างภาษากันได้ ซึ่งก็ได้เลือกใช้โปรแกรม Flash และภาษา ActionScript มาใช้ในการพัฒนาเกม เนื่องจากเป็นรูปแบบงานเกมที่บุคคลทั่วไปสนใจกันมาก มีรูปแบบการนำเสนอที่สวยงาม และสามารถใช้ลูกเล่นได้หลากหลาย โดยเฉพาะการสร้างงาน Animation ที่สามารถ Interactive ได้ อีกทั้งยังสามารถนำความรู้นี้ไปประยุกต์ต่อยอดได้โดยง่าย

ในส่วนของการเชื่อมต่อระหว่างโปรแกรม Flash และโปรแกรม Visual C++ ที่ใช้ในการประมวลผลภาพจากกล้องวีดีโอ นั่นก็สามารถกระทำได้โดยง่าย เพียงแค่เราติดตั้งโปรแกรม Flash ก็จะมี Shockwave Flash OCX Active X Component เพิ่มเข้ามาให้นำไปใช้ได้ใน Visual C++ พร้อมกับฟังก์ชันต่างๆมากมาย ให้ใช้ในการรับส่งค่าระหว่าง Flash และ Visual C++ ซึ่งจุดนี้ไม่เพียงแต่จะใช้กับเกม Flash เท่านั้น ยังสามารถนำ Flash มาสร้างเป็น User Interface เพื่อสร้างสีสันให้กับโปรแกรม แทนหน้าจอแบบเดิมๆที่สร้างจาก Visual C++ ได้อีกด้วย

สำหรับตัวเกม Subdue Narcotic หรือเกมปราบปรามยาเสพติดนั้น ถูกคิดขึ้นเพื่อให้ความรู้และกระตุ้นเตือนเด็กๆ ให้ตระหนักถึงพิษภัยและอันตราย ที่จะเกิดขึ้นจากยาเสพติดหลากหลายชนิด ผ่านรูปแบบของเกมเพื่อเพิ่มความน่าสนใจ และดึงดูดใจจากเด็กๆ แบ่งออกเป็น 2 เกม คือ Learning Game และ Subdue Game โดยเกมแรกจะเป็นเกมที่จะให้เรียนรู้เกี่ยวกับยาเสพติดชนิดต่างๆ ผ่านทางการตอบคำถาม เล่นด้วยการเลือกเหยียบคำตอบที่คิดว่าถูกต้อง สำหรับเกมที่ 2 นั้นจะให้ทำการปราบปรามยาเสพติด ด้วยการกระโดดไปยังจุดที่ยาเสพติดอยู่เพื่อทำการต่อสู้กับยาเสพติดนั้นๆ โดยสาเหตุที่ต้องใช้รูปแบบวิธีการเล่นแบบกระโดดเข้าไปเหยียบยังจุดต่างๆ ก็เพื่อให้เหมาะสมกับอัลกอริทึมที่ใช้ประมวลผลภาพจากกล้องวีดีโอ โดยจะกำหนดจุดให้เหยียบแตกต่างกันไปในแต่ละเกม เพื่อชี้ให้เห็นว่าเราสามารถนำค่าที่ได้จากการประมวลภาพจากกล้องวีดีโอมาใช้ได้หลากหลายรูปแบบ ตามแต่ที่เราจะกำหนด แต่ก็ยังมีข้อจำกัดอยู่หลายประการ

สำหรับลักษณะการทำงานของอัลกอริทึม ที่เลือกนำมาใช้ในการตรวจวัดตำแหน่งขาและวิธีการประมวลผลภาพจากกล้องวีดีโอ นั้นสามารถสรุปลักษณะการทำงานได้ดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1) ลักษณะการยืนหันหลังให้กับกล้องวีดีโอ 180 องศา จะมีค่าความถูกต้องของค่าตำแหน่งที่คำนวณได้สูงที่สุด และการยืนหันหน้าให้กับกล้องวีดีโอ 0 องศา และระยะปลายเท้าทั้ง 2 ด้านห่าง 40 เซนติเมตร จะมีค่าความถูกต้องของค่าตำแหน่งที่คำนวณได้ต่ำที่สุด เนื่องจากอัลกอริทึมในการตรวจวัดตำแหน่งขาเป็นการคำนวณหาจุดปลายเท้าบนพื้นระนาบ ดังนั้นการยืนหันหน้าให้กล้องวีดีโอจะมีส่วนปลายเท้ายื่นออกมาทำให้การตรวจวัดตำแหน่งขาที่ได้เป็นตำแหน่งของปลายเท้าแทน และการยืนหันหลังให้กล้องวีดีโอ 180 องศา จะไม่มีส่วนของเท้ายื่นออกมาทำให้การตรวจวัดค่าตำแหน่งมีความถูกต้องสูง แต่จากค่าที่ได้จากการยืนหันหน้าให้กล้องวีดีโอก็มีความถูกต้องสูงพอที่จะใช้ในการเล่นเกมในลักษณะนี้ได้อย่างไม่มีปัญหา เพราะเกมนี้จะใช้ค่าเป็นช่วงเพียงพอที่จะยืนได้ทั้ง 2 ขา

2) ระยะห่างจากกล้องวีดีโอสูงขึ้น ความถูกต้องของค่าตำแหน่งที่คำนวณได้จะมีความถูกต้องน้อยลง เนื่องจากเมื่อระยะห่างสูงขึ้นขนาดของขาของผู้ทดลองในภาพวีดีโอก็จะมีขนาดเล็ก เมื่อเทียบกับขนาดของขาที่มีระยะห่างจากกล้องวีดีโอน้อยกว่า ดังนั้นเมื่อมีระยะห่างสูง ขนาดของขาในภาพวีดีโอก็มีขนาดเล็ก จึงทำให้จำนวนของจุดภาพที่นำมาใช้ในการคำนวณน้อย ทำให้มีค่าความผิดพลาดจากการคำนวณค่าตำแหน่งสูง ดังนั้นในการออกแบบเกมนี้จึงออกแบบให้จุดที่จะยืนไกลสุดไม่ห่างจากกล้องมาก จึงไม่เป็นปัญหาในการคำนวณหาค่าตำแหน่ง และยังคงมีความถูกต้องสูงพอที่จะใช้เล่นเกมได้อย่างไม่มีปัญหา

3) ทิศทางการยืนในแนวขวางกับทิศทางของกล้อง จะมีค่าความถูกต้องของค่าตำแหน่งที่คำนวณได้สูงที่สุด เนื่องจากอัลกอริทึมในการคำนวณค่าตำแหน่งจะสามารถคำนวณค่าตำแหน่งได้ถูกต้องสูง และเท้าทั้งสองข้างควรอยู่ในลักษณะสมมาตร แต่การกระโดดโดยมีการยกเท้าขึ้น ทำให้เสียความสามาตรของเท้าทั้ง 2 ไปบ้าง แต่ก็ไม่เป็นปัญหากับช่วงที่กำหนดไว้ในเกม และตำแหน่งสุดท้ายที่กระโดดลงมาก็จะยืนด้วยเท้าคู่ตามลักษณะการเล่น จึงไม่มีผลเท่าใดนักกับเกมที่สร้างถ้าเล่นตามวิธีการที่กำหนดไว้ในการเล่นเกม

7.2 ข้อเสนอแนะสำหรับการพัฒนาในอนาคต

1) เนื่องจากลักษณะการยืนหันหน้า หันหลัง การแยกปลายเท้า ทิศทางการวางเท้า และระยะห่างจากกล้องวีดีโอ ต่างกันจะทำให้ความแม่นยำในการตรวจวัดตำแหน่งขาต่างกัน ดังนั้นการกำหนดลักษณะวิธีการเล่นเกม จึงไม่ควรใช้ความแม่นยำของตำแหน่งที่สูงเกินไป

2) อัลกอริทึมที่ใช้ในโครงงานนี้ใช้วิธีการลบภาพพื้นหลังแบบง่าย ทำให้เมื่อมีการเปลี่ยนแปลงของแสง เช่น มากขึ้นหรือน้อยลง ทำให้อัลกอริทึมลบภาพพื้นหลัง ไม่สามารถทำงานได้อย่างมีประสิทธิภาพ จึงจำเป็นจะต้องมีการพัฒนาอัลกอริทึมลบภาพพื้นหลังที่มีความคงทนต่อการเปลี่ยนแปลงของแสงมาใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3) อัลกอริทึมที่ใช้ในการคำนวณค่าตำแหน่งในโครงงานนี้ ยังไม่สามารถคำนวณหาได้ว่าผู้ใช้งานยืนหันหน้าในทิศทางใด ดังนั้นหากต้องการทำเกมหรือระบบที่ต้องทราบว่าผู้เล่นหันหน้าทิศใด จำเป็นต้องมีการพัฒนาอัลกอริทึมในการคำนวณหาทิศทางการยทนของผู้ใช้งานเพิ่ม

4) ระบบภาษา ActionScript ของเกม Flash ส่วนใหญ่จะรองรับการทำงานที่เกิดจากการใช้เมาส์และคีย์บอร์ดเป็นหลัก ดังนั้นในการพัฒนาเกมโดยไม่ใช้เมาส์และคีย์บอร์ดในการควบคุมนั้นจึงทำให้เกิดความยุ่งยาก เนื่องจากการกระทำบางอย่างระบบไม่รองรับหรือไม่สนับสนุน ซึ่งทำให้ต้องหาหนทางอื่นมาใช้ในการแก้ไขปัญหา หรือตรวจสอบแทน ซึ่งก็จะทำให้ระบบยุ่งยากและซับซ้อนมากยิ่งขึ้น ดังนั้นในการออกแบบเกมจึงควรหาคิดหาวิธีการเพื่อเอาไว้ด้วย

5) ระบบของเกม Flash ยังไม่รอบรับการทำงานในรูปแบบ 3 มิติเท่าใดนัก ดังนั้นการทำเกมจึงทำได้เพียงแค่ทำให้ดูเหมือนว่าเป็น 3 มิติ ด้วยการกำหนดขนาดและความลึก พร้อมกับออกแบบฉากและตัวละครให้ดูมีมิติเอาเอง ซึ่งก็อาจจะทำให้ระบบยุ่งยากและซับซ้อนมากยิ่งขึ้น แต่ก็ดูดีและมีมิติมากขึ้น ซึ่งในอนาคตทางผู้พัฒนา Flash ก็น่าจะมีระบบ 3D เข้ามา Support ในจุดนี้มากขึ้นแล้ว

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

กำพล ลีลาภรณ์. 2551. **Flash ActionScript**. กรุงเทพฯ: Provision.

กำพล ลีลาภรณ์ และสุรพงษ์ วีระรักษ์เดชา. 2546. **Flash MX ActionScript Inspire**. กรุงเทพฯ: Provision.

ณรงค์ศักดิ์ พุดเผือด. 2548 **"อัลกอริทึมในการตรวจวัดตำแหน่งขาด้วยกล้องวีดีโอตัวเดียวสำหรับระบบแสดงภาพสเตอริโอแบบเปลี่ยนมุมมองได้."** วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ บัณฑิตวิทยาลัย, สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง.

ยุทธนา ลีลาศวัฒนกุล. 2546. **คู่มือการเขียนโปรแกรมและใช้งาน Visual C++ .NET ฉบับสมบูรณ์.** นนทบุรี : อินโฟเพรส.

สุธี พงศาสกุลชัย และสุรเชษฐ์ วงศ์ชัยพรพงษ์. 2548. **คัมภีร์ Flash MX 2004 Actionscript.** กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.

สุธี พงศาสกุลชัย และสุรเชษฐ์ วงศ์ชัยพรพงษ์. 2548. **คัมภีร์ การออกแบบและพัฒนาเกมส์ด้วย Flash MX 2004**. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.

Anne Goebel, editor. 2007. **Actionscript 3.0 Game Programming University.** United States of America : Que Publishing.

Criminisi, A. Reid, I. and Zisserman, A. 1999. "A Plane Measuring Device." **Image and Vision Computing.** 17(8) : 625-634.

Huiyu Zhou and Housheng Hu. 2004. "A Survery – Human Movement Tracking and Stroke Rehabilitation" **Department of Computer Sciences University of Essex United Kingdom.** December : 3-5.

Nik Level. 2004. **Flash MX 2004 Games Art to ActionScrip**. England : Elsevier.

Tonya Simpson. 2003. **Macromedia Flash MX Actionscript For Fun & Games.** United States of America : Que Publishing.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประวัติผู้เขียน

ชื่อผู้เขียน นางสาวปัญญนุช ลักษณวงษ์

วันเดือนปีเกิด 10 กรกฎาคม 2524

สถานที่เกิด สุพรรณบุรี

ที่อยู่ปัจจุบัน 222/32 ตึก AF3 แฮปปี้แลนด์ทาวเวอร์ แขวงคลองจั่น เขตบางกะปิ จังหวัด กรุงเทพมหานคร

ประวัติการศึกษา <u>ระดับปริญญาตรี</u>

ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
มหาวิทยาลัยหอการค้าไทย

<u>ระดับปริญญาโท</u>

แขนงวิชาวิทยาการสารสนเทศ คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้