เกมควบคุมท่าทางการเคลื่อนไหวด้วยกล้องเว็บแคม

MOTION CONTROLLER GAME USING WEB CAMERA

กันตพงศ์ สิริบัญชาพร

ฐาปนี ชาญวรรณ์

โครงงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

เกมควบคุมท่าทางการเคลื่อนไหวด้วยกล้องเว็บแคม

MOTION CONTROLLER GAME USING WEB CAMERA

กันตพงศ์ สิริบัญชาพร

ฐาปนี สาสุวรรณ์

โครงงานพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2556

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

MOTION CONTROLLER GAME USING WEB CAMERA

KANTAPONG SIRIBUNCHAPORN

THAPANEE SASUWAN

A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR OF SCIENCE

IN COMPUTER SCIENCE

FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2013

เอกสารนี้เป็นม... ...ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลง... ...เจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อปัญหาพิเศษ	เกมควบคุมท่าทางการเคลื่อนไหวด้วยกล้องเว็บแคม		
	MOTION CONTROLLER GAME USING WEB CAMERA		
ชื่อนักศึกษา	นายกันตพงศ์	สิริบัญชาพร	53050935
	นางสาวฐาปนี	สาสุวรรณ์	53050971
ปริญญา	วิทยาศาสตรบัณฑิต		
ภาควิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2556		
อาจารย์ที่ปรึกษา	ผศ.ดร.กรกช ประชุมรักษ์		

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์ประจำปีการศึกษา 2556

คณะกรรมการสอบ	ลายมือชื่อ
ดร.รุ่งรัตน์ เวียงศรีพนาวัลย์ ประธานกรรมการ	
ดร.อนันตพร หรรษคุณาฒัย กรรมการ	
ผศ.ดร.กรกช ประชุมรักษ์ กรรมการและอาจารย์ที่ปรึกษา	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้งที่มีการนำไปใช้

ลิขสิทธิ์ของภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

หัวข้อปัญหาพิเศษ	เกมควบคุมท่าทางการเคลื่อนไหวด้วยกล้องเว็บแคม		
	MOTION CONTROLLER GAME USING WEB CAMERA		
ชื่อนักศึกษา	นายกันตพงศ์	สิริบัญชาพร	53050935
	นางสาวฐาปนี	สาสุวรรณ์	53050971
ปริญญา	วิทยาศาสตรบัณฑิต		
ภาควิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2556		
อาจารย์ที่ปรึกษา	ผศ.ดร.กรกช ประชุมรักษ์		

บทคัดย่อ

โครงงานพิเศษชิ้นนี้ได้ศึกษาและพัฒนาเกี่ยวกับการควบคุมเกมโดยใช้กล้องเว็บแคม และนำเสนอวิธีการตรวจจับสีผิวทั้งหมดโดยใช้สีผิวจากใบหน้า โดยใช้ไลบรารี่ของ OpenCV เพื่อรับอินพุตในการควบคุมการเคลื่อนไหวของตัวละครภายในเกม มีการใช้กระบวนการทำงานที่เกี่ยวข้องกับรูปภาพ (Image Processing) เพื่อวิเคราะห์และติดตามการเคลื่อนไหวของมือและใช้หลักการ Haar-like Feature ในการตรวจจับใบหน้า ส่วนวิธีการตรวจจับสีผิวนั้น ภายในปัญหาพิเศษนี้ได้นำเสนอวิธีใหม่โดยใช้สีผิวจากใบหน้ามากำหนดขอบเขตของสีผิว จากการศึกษาพบว่าวิธีการที่นำเสนอนี้ทำให้การตรวจจับสีผิวมีความแม่นยำมากยิ่งขึ้น และสามารถนำมาประยุกต์เข้ากับระบบในการติดตามการเคลื่อนไหวของมือและใบหน้าได้ ซึ่งการเคลื่อนไหวของตัวละครภายในเกมมีความสอดคล้องกับการควบคุมของผู้เล่น ทั้งนี้เพื่อนำเสนองานด้านการตรวจจับภาพเคลื่อนไหวให้มีประสิทธิภาพมากขึ้น

คำสำคัญ : การตรวจจับใบหน้า, การตรวจจับสีผิว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Title	MOTION CONTROLLER GAME USING WEB CAMERA
Student	Mister Kantapong Siribunchaporn 53050935
	Miss Thapanee Sasuwan 53050971
Degree	Bachelor of Science
Major Program	Computer Science
Academic Year	2013
Advisor	Asst. Prof. Dr. Korakot Prachumrak

ABSTRACT

The special project studies and develops "Motion Controller Game Using Web Camera". This project presents a new adaptive skin modeling approach based on face skin tone color. We apply OpenCV library control the motion of the game character. In our new approach, we apply Haar-like feature technique for face detection, then the face color is used to define the color range of skin color in the other part of the body. This program is able to detect face and hands from input and then enforce the predefined model to move corresponding to actual movement. The proposed method was able to show the validity and robust performance of the proposed algorithm and low - cost production.

Keywords : Face detection, Skin color detection

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กิตติกรรมประกาศ

การจัดทำปัญหาพิเศษในหัวข้อ เกมควบคุมท่าทางการเคลื่อนไหวด้วยกล้องเว็บแคม สำเร็จลุล่วงไปได้ด้วยดี เนื่องจากผู้จัดทำได้รับความช่วยเหลือจากบุคคลผู้มีพระคุณหลายท่าน ดังนี้

ขอขอบพระคุณ บิดา มารดา และสมาชิกในครอบครัวทุกคน ผู้ซึ่งอยู่เบื้องหลัง คอยอบรมสั่งสอน ให้กำลังใจตลอดในการจัดทำปัญหาพิเศษ และความอบอุ่นเสมอมา

ขอขอบพระคุณ ผศ.ดร.กรกช ประชุมรักษ์ อาจารย์ที่ปรึกษาในการทำหัวข้อปัญหาพิเศษนี้ที่ได้ให้คำปรึกษา และคำแนะนาในการแก้ปัญหาต่างๆ รวมถึงการตรวจสอบและแก้ไขการเขียนรายงานปัญหาพิเศษเล่มนี้อย่างละเอียด

ขอขอบพระคุณ ดร.รุ่งรัตน์ เวียงศรีพนาวัลย์ และ ดร.อนันตพร หรรษคุณาฒัย ประธานกรรมการที่ปรึกษาปัญหาพิเศษ ที่กรุณาเป็นกรรมการคุมสอบ และให้คำปรึกษาในปัญหาพิเศษนี้

ขอขอบพระคุณ อาจารย์สาขาวิทยาการคอมพิวเตอร์ทุกท่าน ที่อบรมสั่งสอน ให้ความรู้ทั้งภาคทฤษฎี และภาคปฏิบัติให้แก่ทางคณะผู้จัดทำมาตลอดระยะเวลา 4 ปี จนกระทั่งปัญหาพิเศษนี้ได้สำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ต้องขอบคุณ เพื่อนๆ น้องๆทุกคน ที่คอยให้ความช่วยเหลือ และให้กำลังใจเสมอมา

นอกจากนี้ยังมีบุคคลท่านอื่นๆ ที่ไม่ได้กล่าวไว้ ณ ที่นี้ จึงใคร่ขอขอบพระคุณทุกท่านที่กรุณามีส่วนร่วมในการให้ความช่วยเหลือ คำปรึกษา คำแนะนำ ตลอดจนกำลังใจในการทำปัญหาพิเศษฉบับนี้

นายกันตพงศ์ สิริบัญชาพร
นางสาวฐาปนี สาสุวรรณ์

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 ที่มาและความสำคัญของปัญหาพิเศษ	1
1.2 วัตถุประสงค์	1
1.3 ทฤษฎีหรือแนวความคิดที่ใช้ในการศึกษา	1
1.4 ขอบเขตของโครงงาน	2
1.5 ขั้นตอนในการดำเนินงาน	2
1.6 ประโยชน์ที่คาดว่าจะได้รับ	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	3
2.1 ความรู้พื้นฐานเกี่ยวกับประมวลภาพดิจิทัล	3
2.1.1 ความรู้พื้นฐานเกี่ยวกับภาพดิจิทัล	3
2.1.2 ข้อมูลเบื้องต้นเกี่ยวกับ Web camera	4
2.2 ความรู้พื้นฐานด้านคอมพิวเตอร์วิทัศน์	4
2.2.1 OpenCV (Open Source Computer Vision Library)	5
2.2.2 OpenCvSharp	5
2.2.3 Equalization	5
2.2.4 Haar-like Feature	7
2.2.5 การหาเส้นรูปร่าง	9
2.2.6 บริเวณที่สนใจ Region-of-interest (ROI)	11

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
2.3 ระบบสี	11
2.3.1 ความรู้พื้นฐานเกี่ยวกับภาพสีระบบอาร์จีบี (RGB Image)	11
2.3.2 ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ HSV	11
2.3.3 ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ YCbCr	12
2.4 งานวิจัยที่เกี่ยวข้อง	13
2.5 Unity	14
2.5.1 การแสดงผล	15
2.5.2 การเขียนสคริปต์	15
2.5.3 Asset Tracking	16
2.5.4 แพลตฟอร์ม	16
2.5.5 Asset Store	16
2.5.6 ฟิสิกส์	17
2.5.7 รุ่น	17
บทที่ 3 ขั้นตอนการออกแบบและพัฒนาแอพพลิเคชัน	19
3.1 หลักการและแนวคิด	19
3.1.1 วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า	20
3.2 การตรวจจับใบหน้า	20
3.3 ขั้นตอนการแยกสีผิวจากวิธีการกำหนดขอบเขตสีผิวจากใบหน้า	22
3.3.1 การเตรียมภาพ	22
3.3.2 การแบ่งภาพสีผิว	23
3.3.3 การปรับภาพให้เรียบโดยใช้มอร์โฟโลยี	25
3.4 การทดลอง	26
3.5 การวัดประสิทธิภาพ	30
3.6 สรุปผลการทดลอง	31

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

	หน้า
3.7 การทำงานของโปรแกรม	31
3.8 ขั้นตอนการประมวลผลท่าทาง	33
3.9 ส่วนประกอบต่างๆของโปรแกรม	36
บทที่ 4 ขั้นตอนการทำงานของโปรแกรม	38
4.1 ฮาร์ดแวร์และซอฟต์แวร์สำหรับพัฒนาโปรแกรม	38
4.2 อุปกรณ์ที่ใช้พัฒนาโปรแกรม	38
4.3 ขั้นตอนการทำงานของแอพพลิเคชัน	39
4.3.1 เมื่อเริ่มการทำงานของเกมแอพพลิเคชัน	39
4.3.2 การค้นหา และตรวจจับใบหน้าเพื่อใช้เป็นค่าเริ่มต้น	39
4.3.3 ขั้นตอนการตอบโต้และควบคุมการเคลื่อนไหวภายในเกม	40
4.3.4 ข้อจำกัดในการใช้โปรแกรมในสภาพแวดล้อมต่างๆกัน	41
บทที่ 5 ผลสรุปและข้อเสนอแนะ	57
5.1 สรุปผล	57
5.1.1 ปัญหาต่างๆ ที่พบระหว่างการพัฒนาโปรแกรม	57
5.2 ข้อเสนอแนะ	58
5.3 แนวทางการพัฒนาในอนาคต	58
เอกสารอ้างอิง	**59**
ภาคผนวก ก	**61**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
3.1 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 1	27
3.2 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 2	28
3.3 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 3	29
3.4 ผลการทดลองการตรวจจับสีผิวในภาพสี	30
3.5 เวลาในการตรวจจับสีผิวในภาพสี	30
3.6 พิกัดของหน้าจอ	33

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป

รูปที่		หน้า
2.1	โครงสร้างของภาพดิจิทัล	3
2.2	ภาพต้นแบบ	6
2.3	กราฟที่สอดคล้องกันของภาพต้นแบบ	6
2.4	ภาพหลังจากการทำ Equalization	6
2.5	กราฟที่สอดคล้องกันของภาพหลังจากการทำ Equalization	7
2.6	รูปแบบของรูปเหลี่ยมสำหรับการตรวจจับลักษณะแบบต่างๆ	8
2.7	ตัวอย่างการใช้รูปเหลี่ยมตรวจจับลักษณะต่างๆ	8
2.8	การคำนวณแบบ Integtral image	8
2.9	รูปแบบการเชื่อมติดกันของพิกเซล	9
2.10	ลำดับชั้นของส่วนประกอบที่เชื่อมติดกัน	10
2.11	แบบจำลองสี HSV	12
2.12	แบบจำลองสี YCbCr	12
3.1	แผนภาพการตรวจจับใบหน้า	21
3.2	ภาพที่ได้รับจากกล้อง	22
3.3	ภาพหลังจากการย่อขนาด	23
3.4	ภาพใบหน้าขยาย 2 เท่าและเปลี่ยนเป็นโมเดลสี HSV	23
3.5	กราฟข้อมูล 2 มิติ H-S	24
3.6	กรอบสี่เปลี่ยมล้อมรอบข้อมูลที่มีการรวมตัวกันหนาแน่นมากที่สุด	25
3.7	ผลลัพธ์ที่ได้จากวิธีการกำหนดขอบเขตสีผิวจากใบหน้า	25
3.8	ภาพก่อนและหลังการทำมอร์โฟโลยี	26
3.9	แผนภาพโดยรวมของระบบ	32
3.10	แผนภาพการตรวจจับสีผิว	34
3.11	ภาพต้นแบบ	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
3.12 ภาพการแบ่งพิกัดของหน้าจอ	35
4.1 ภาพตัวอย่างกล้องที่ใช้ติดต่อกับคอมพิวเตอร์	38
4.2 ภาพเปิดของเกมแอพพลิเคชัน	39
4.3 การตรวจจับติดตามใบหน้าและมือของผู้เล่น	40
4.4 ภาพแสดงเมื่อเริ่มเข้าสู่การทำงานของเกม	40
4.5 ท่าทางบังคับให้ตัวละครกระโดดขึ้นในแนวตรง	41
4.6 แสดงภาพตัวละครกระโดดในแนวตรง	41
4.7 ท่าทางบังคับให้ตัวละครกระโดดทางซ้ายหรือเดินทางซ้าย	42
4.8 แสดงภาพตัวละครกระโดดทางซ้าย	42
4.9 แสดงภาพตัวละครเดินไปทางซ้าย	43
4.10 ท่าทางบังคับให้ตัวละครกระโดดทางขวาหรือเดินทางขวา	43
4.11 แสดงภาพตัวละครกระโดดทางขวา	44
4.12 แสดงภาพตัวละครเดินไปทางขวา	44
4.13 แสดงภาพเมื่อจบเกม	45
4.14 แสดงภาพตัวละครยืนนิ่ง	45
4.15 ท่าทางในการกำจัดหมอกท่าทาง 1	46
4.16 ท่าทางในการกำจัดหมอกท่าทาง 2	47
4.17 แสดงภาพตัวละครเดินไปทางซ้าย	47
4.18 แสดงภาพตัวละครกระโดดทางขวา	48
4.19 แสดงภาพตัวละครยืนนิ่ง	49
4.20 ท่าทางบังคับให้ตัวละครเดินตรง	49
4.21 แสดงภาพตัวละครเดินตรง	50
4.22 ท่าทางบังคับให้ตัวละครเดินถอยหลัง	50

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
4.23 แสดงภาพตัวละครเดินถอยหลัง	51
4.24 แสดงภาพตัวละครเดินไปทางซ้าย	51
4.25 แสดงภาพตัวละครเดินไปทางขวา	52
4.26 ท่าทางบังคับให้ตัวละครต่อยด้านซ้าย	52
4.27 แสดงภาพตัวละครต่อยด้านซ้าย	53
4.28 ท่าทางบังคับให้ตัวละครต่อยด้านขวา	53
4.29 แสดงภาพตัวละครต่อยด้านขวา	54
4.30 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีสีวัตถุใกล้เคียงกับสีผิว	54
4.31 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีคนภายในฉากเกิน 1 คนกรณี 1	55
4.32 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีคนภายในฉากเกิน 1 คนกรณี 2	55
4.33 แสดงภาพการตรวจจับในระยะห่าง	56
ก-1 แสดงหน้าเว็บไซต์ของ Unity	62
ก-2 แสดงหน้าเว็บไซต์สำหรับการดาวน์โหลด	62
ก-3 แสดงไดเรกทอรี่ที่เก็บไฟล์ UnitySetup.exe	63
ก-4 แสดงหน้าเริ่มต้นในการติดตั้งโปรแกรม Unity	63
ก-5 โปรแกรมแสดงข้อตกลงและเงื่อนไขต่างๆ	64
ก-6 โปรแกรมแสดงรายละเอียดที่จะติดตั้ง	64
ก-7 ไดเรกทอรี่ที่จะติดตั้งโปรแกรม	65
ก-8 โปรแกรม Unity.exe	65
ก-9 หน้าโปรแกรม Unity	66
ก-10 หน้าต่างในการสร้างโปรเจค	66
ก-11 ตัวอย่างในการสร้างโปรเจค	67
ก-12 ตัวอย่างการสร้างวัตถุแบบแรก	67
ก-13 ตัวอย่างการสร้างวัตถุแบบที่สอง	68
ก-14 ภาพวัตถุภายในเกม	68

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
ก-15 แสดงการปรับขนาด ตำแหน่ง และการหมุนของวัตถุ	69
ก-16 แสดงการกำหนดตำแหน่งของวัตถุ	69
ก-17 แสดงการกำหนดการหมุนของวัตถุ	70
ก-18 แสดงการกำหนดขนาดของวัตถุ	70
ก-19 แสดงการตั้งค่าส่วนประกอบของวัตถุ	71
ก-20 แสดงคำสั่งต่างๆในส่วนประกอบของวัตถุ	71
ก-21 แสดงการนำเข้าวัตถุ 3 มิติมาใช้งาน	72
ก-22 ไดเรกทอรี่ที่เก็บไฟล์ 3 มิติ	72
ก-23 ตัวอย่างไฟล์ 3 มิติที่นำเข้ามา	73
ก-24 แสดงการนำไฟล์ 3 มิติเข้ามาภายในฉาก	73
ก-25 วิธีการสร้างคำสั่ง	74
ก-26 ขั้นตอนการสร้างคำสั่ง	74
ก-27 ตัวอย่างไฟล์คำสั่ง	75
ก-28 ตัวอย่างคำสั่ง	75
ก-29 ภาพทดลองการทำงาน	76
ก-30 แสดงการใส่แสง	76
ก-31 แสดงการใส่ Directional Light	77
ก-32 แสดงการเปลี่ยนประเภทแสง	77
ก-33 แสดงการเปลี่ยนสีของแสง	78
ก-34 แสดงการใส่ Point Light	78
ก-35 แสดงการใส่ Spotlight	79
ก-36 แสดงการใส่ Area Light	79
ก-37 การสร้าง Animator Controller	80
ก-38 การใส่ไฟล์ท่าทางการเคลื่อนไหวและการทำ Animator	80
ก-39 การใส่ท่าทางของตัวละคร	81
ก-40 การสร้าง Transition	81

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญรูป (ต่อ)

รูปที่	หน้า
ก-41 ตัวอย่าง Transition	81
ก-42 การกำหนด Condition	82
ก-43 ประเภทพารามิเตอร์ต่าง	82
ก-44 การสร้างพารามิเตอร์	83
ก-45 กำหนด Condition ด้วยพารามิเตอร์	83
ก-46 เงื่อนไขของ Condition	84
ก-47 การสร้าง Component ชื่อ Animatorที่ตัวละคร	84
ก-48 การกำหนด Animator ที่ตัวละคร	84
ก-49 ตัวอย่างคำสั่ง	85
ก- 50 แสดงการใส่เสียงภายในเกม	85
ก- 51 แสดงการเลือกประเภทเสียง	86
ก- 52 แสดงการใส่ไฟล์เสียงในโฟลเดอร์	86
ก- 53 แสดงไฟล์เสียงภายในโปรเจค	87
ก- 54 แสดงการใส่ไฟล์เสียงลงในเกม	87
ก- 55 ส่วนประกอบของเสียง	88
ก- 56 วิธีการบันทึกฉาก	88
ก- 57 วิธีการสร้างเกม	89
ก- 58 ตัวเลือกในการสร้าง	89
ก- 59 แสดงไดเรกทอรี่ที่ต้องการบันทึก	90
ก- 60 ไฟล์ที่ได้จากการสร้าง	90

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหาพิเศษ

ในโปรเจคนี้จะเป็นการนำเสนอเกมแอพพลิเคชัน โดยใช้กล้องเว็บแคมในการตรวจจับและควบคุมการเคลื่อนไหวของตัวละครจากผู้เล่น ซึ่งเกมแอพพลิเคชั่นถือเป็นเทคโนโลยีรูปแบบหนึ่งที่ได้รับความนิยมเป็นอย่างมาก มีการใช้งานแพร่หลายในคอมพิวเตอร์ส่วนบุคคล(PC) และคอมพิวเตอร์พกพา(Laptop) โดยในส่วนของการติดต่อกับผู้ใช้งาน ในส่วนมากจะเป็นการรับอินพุตจากแป้นพิมพ์หรือเม้าส์ แต่ภายในหัวข้อนี้จะเป็นการรับอินพุตจากกล้องเว็บแคม เพราะสามารถนำมาประยุกต์ใช้ในการตรวจจับการเคลื่อนไหวของร่างกาย ราคาถูก ใช้งานง่าย เป็นทางเลือกใหม่ในการเพิ่มความสนุกสนานและความบันเทิงในการเล่นเกมที่ถูกนำมาใช้กันอย่างแพร่หลายในปัจจุบัน

1.2 วัตถุประสงค์

1) เพื่อพัฒนาโปรแกรมที่รับอินพุตจากอุปกรณ์เว็บแคม
2) เกิดความรู้ความเข้าใจในการประมวลผลภาพดิจิทัล
3) นำผลลัพธ์ที่ได้มาประยุกต์ใช้ในงาน 3 มิติเชิงมัลติมีเดีย

1.3 ทฤษฎีหรือแนวความคิดที่ใช้ในการศึกษา

หลักการประมวลผลภาพ (Image Processing) ที่ทางผู้จัดทำได้นำมาใช้นั้น เป็นการนำไลบรารี่ของ OpenCvSharp มาใช้ในการประมวลผลภาพ ซึ่งเป็นไลบรารี่ไม่มีข้อจำกัดในเรื่องของฮาร์ดแวร์ (Hardware) จึงสามารถนำมาใช้งานได้ง่ายและใช้งานได้ในหลายภาษา ไม่ว่าจะเป็นภาษา C ภาษา C++ และภาษา Python โดยในไลบรารี่นี้มีชุดคำสั่งที่น่าสนใจ เช่น การค้นหาวัตถุในภาพ การจดจำท่าทางการเคลื่อนไหว การจำแนกวัตถุ และการติดตามการเคลื่อนไหวของวัตถุ ทางคณะผู้จัดทำได้นำทฤษฎีการตรวจจับและการติดตามการเคลื่อนไหวของร่างกายมาประยุกต์ใช้กับโปรแกรมและใช้ภาษาซีชาร์ป (C#) ในการพัฒนาควบคู่กัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 ขอบเขตของโครงงาน

1) สร้างโปรแกรมในการเชื่อมต่อกล้องเข้ากับโปรแกรม
2) สร้างโปรแกรมเกี่ยวกับการประมวลผลรูปภาพ เพื่อทำการตรวจจับและติดตามการเคลื่อนไหวของร่างกาย
3) พัฒนาเกมแอพพลิเคชันที่รับภาพจากกล้อง เพื่อนำมาประมวลผลลักษณะท่าทางของร่างกาย
4) แสดงการเคลื่อนไหวของตัวละครออกทางหน้าจอของโปรแกรม

1.5 ขั้นตอนในการดำเนินงาน

1) กำหนดขอบเขต จุดประสงค์ ความต้องการของโครงงาน
2) วางแผนขั้นตอนการทำงาน
3) ศึกษาหลักการเขียนโปรแกรม เช่น ภาษาซีชาร์ป และ OpenCvSharp เป็นต้น
4) ศึกษาทฤษฎี และข้อมูลที่เกี่ยวข้องกับกระบวนการทำงานที่เกี่ยวข้องกับรูปภาพ
5) วิเคราะห์และออกแบบระบบ
6) พัฒนาและเขียนโปรแกรม
7) ออกแบบส่วนที่ติดติดกับผู้ใช้
8) จัดทำเอกสาร

1.6 ประโยชน์ที่คาดว่าจะได้รับ

1) ช่วยเพิ่มทักษะในด้านการเขียนโปรแกรม และความรู้เกี่ยวกับการประมวลผลภาพ (image processing)
2) สามารถเรียนรู้การใช้งานการติดต่อกับกล้อง เพื่อเขียนโปรแกรมในการประมวลผลภาพ
3) สามารถพัฒนาเทคโนโลยีทางด้านมัลติมีเดีย เพื่อนำมาประยุกต์ใช้และอำนวยความสะดวก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

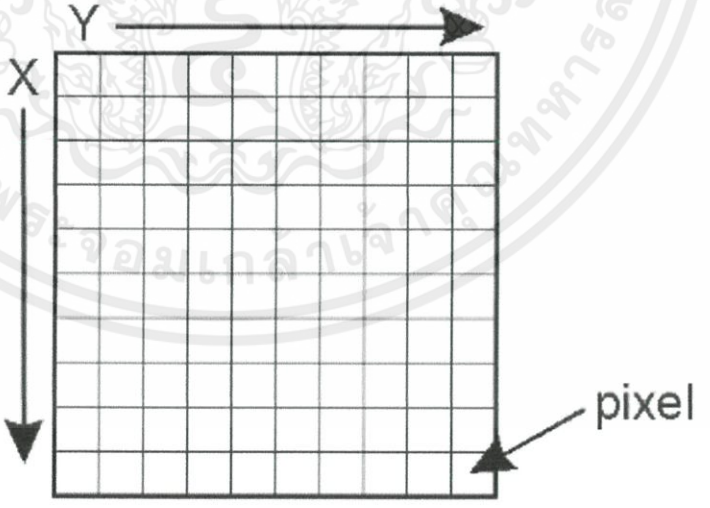
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1. ความรู้พื้นฐานเกี่ยวกับประมวลภาพดิจิทัล

ในปัญหาพิเศษนี้จะใช้ภาพดิจิทัลที่ได้จากกล้องเว็บแคมเป็นข้อมูลรับเข้าของระบบ เพื่อนำภาพเหล่านี้ไปวิเคราะห์ตรวจจับและติดตามลักษณะเด่นตามที่ต้องการ ดังนั้นในหัวข้อนี้จะอธิบายเกี่ยวกับความรู้พื้นฐานเกี่ยวกับการวิเคราะห์ประมวลผลภาพ

2.1.1. ความรู้พื้นฐานเกี่ยวกับภาพดิจิทัล

ภาพดิจิทัลนั้นคือ ภาพที่ถูกแปลงสัญญาณจากรูปแบบแอนะล็อก (Analog) ให้อยู่ในรูปของดิจิทัล (Digital) โดยหลังจากการทำเป็นภาพดิจิทัลแล้ว ภาพเหล่านี้จะถูกลงในหน่วยความคอมพิวเตอร์ในรูปแบบของอาร์เรย์หลายมิติ โดยส่วนที่เล็กที่สุดนั้นเรียกว่า “พิกเซล” (Pixel) อาจกล่าวได้ว่าภาพก็คือพิกเซลที่ประกอบกันเป็นจำนวนมากนั่นเอง และค่าในอาร์เรย์จะอยู่ในรูปของฟังก์ชันสองมิติ f(x,y) โดยที่ x และ y จะแทนพิกัดของจุดภาพ และค่า f เป็นฟังก์ชันแทนความเข้มของจุดภาพนั้น (Intensity)



รูปที่ 2.1 โครงสร้างของภาพดิจิทัล[1]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.1.2. ข้อมูลเบื้องต้นเกี่ยวกับ Web camera

เว็บแคม (Webcam) คือกล้องวีดีโอที่ใช้ในการส่งภาพแบบเวลาจริง (เวลาจริง) ให้กับคอมพิวเตอร์หรือเครือข่ายคอมพิวเตอร์เครื่องอื่นๆ ประโยชน์ของกล้องเว็บแคมที่นำมาใช้กันมากที่สุดนั่นก็คือ การเชื่อมต่อวีดีโอ ซึ่งเป็นการให้คอมพิวเตอรทำหน้าที่เป็นสถานีในการรับส่งวีดีโอ เช่น สามารถบันทึกภาพนิ่งหรือเคลื่อนไหวของเรา ให้ไปปรากฏในหน้าจอมอนิเตอร์และสามารถส่งภาพเคลื่อนไหวนี้ผ่านระบบเครือข่ายเพื่อให้คนอีกฟากหนึ่งสามารถเห็นตัวเราเคลื่อนไหวได้เหมือนอยู่ต่อหน้า ส่วนประโยชน์ในด้านอื่นที่นำมาใช้ได้แก่ การเฝ้าระวังความปลอดภัย ด้านคอมพิวเตอร์วิทัศน์ (Computer vision) และการแพร่ภาพวีดิทัศน์ (Video broadcasting)

สำหรับเซ็นเซ่อร์ที่กล้องเว็บแคมส่วนมากมีอยู่ 2 ชนิด คือ CCD และ CMOS แต่เซ็นเซ่อร์ที่นิยมใช้กันมากก็คือ CMOS เนื่องจากเหตุผลหลายๆประการและตัวเซ็นเซ่อร์แบบ CMOS ก็สามารถแบบออกได้อีก 2 ชนิดด้วยกัน คือ CLF Color CMOS Censor ที่มีความละเอียดของพิกเซลเพียง 110,000 พิกเซล (367 x 291) ในขณะที่ VGA Color CMOS Censor ให้ความละเอียดที่สูงกว่าที่ 350,000 พิกเซล (655 x 493) สำหรับเซ็นเซอร์แบบ CCD จะเป็นเซ็นเซอร์ที่นิยมใช้ในกล้องดิจิตอล เพราะให้ความละเอียดที่สูงกว่าและมีสัญญาณรบกวนไม่มากเหมือนกับเซ็นเซอร์แบบ CMOS

สำหรับการเชื่อมต่อของกล้องเว็บแคมในปัจจุบันส่วนใหญ่ จะเป็นแบบ USB เกือบทั้งหมด ส่วนกล้องเว็บแคมแบบไร้สายจะใช้การเชื่อมต่อในแบบ Wi-Fi หรือ Wireless LAN ทำให้สามารถเคลื่อนย้ายไปได้ทุกที่ แต่ก็ยังมีระยะที่จำกัดและเว็บแคมที่เชื่อมต่อในแบบ Wireless ค่อนข้างที่จะมีราคาสูงพอสมควร

2.2. ความรู้พื้นฐานด้านคอมพิวเตอร์วิทัศน์

คอมพิวเตอร์วิทัศน์เป็นการศึกษาเกี่ยวกับระบบการคิดวิเคราะห์ และการตัดสินใจเกี่ยวกับรูปภาพ ซึ่งเป็นศาสตร์แขนงหนึ่งทางคอมพิวเตอร์ เพื่อให้คอมพิวเตอร์สามารถเรียนรู้เกี่ยวกับรายละเอียดและข้อมูลที่น่าสนใจในภาพได้ โดนส่วนใหญ่นั้นจะเป็นการศึกษาเกี่ยวกับการตรวจจับและค้นหาวัตถุ การติดตามการเชื่อมโยงของวัตถุ และการประเมินผลเกี่ยวกับสิ่งที่สนใจในภาพ เป็นต้น ซึ่งในปัจจุบันได้มีการศึกษาค้นคว้าและวิจัยเกี่ยวกับศาสตร์แขนงนี้เป็นจำนวนมากและจะกล่าวถึงความรู้พื้นฐานที่เกี่ยวข้องกับด้านคอมพิวเตอร์วิทัศน์ที่ใช้ในปัญหาพิเศษนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.2.1. OpenCV (Open Source Computer Vision Library)

OpenCV คือไลบรารี่ของฟังก์ชันที่ใช้สำหรับการพัฒนางานในด้านการประมวลผลภาพและคอมพิวเตอร์วิทัศน์ที่มีฟังก์ชันมากมายให้เลือกใช้งานตามความเหมาะสม ซึ่ง OpenCV นั้นได้ถูกพัฒนาโดยบริษัทอินเทล (Intel) ด้วย OpenCV เป็นไลบรารีที่ไม่มีข้อจำกัดในเรื่องของฮาร์ดแวร์ (Hardware) จึงสามารถนำใช้งานได้ง่าย และใช้งานได้ในหลายภาษา ไม่ว่าจะเป็นภาษา C ภาษา C++ และภาษา Python เป็นต้น ในไลบรารี่นี้มีชุดคำสั่งที่น่าสนใจ ได้แก่ การค้นหาวัตถุในภาพ การจดจำท่าทางการเคลื่อนไหว การจำแนกวัตถุ และการติดตามการเคลื่อนไหวของวัตถุ เป็นต้น

2.2.2. OpenCvSharp

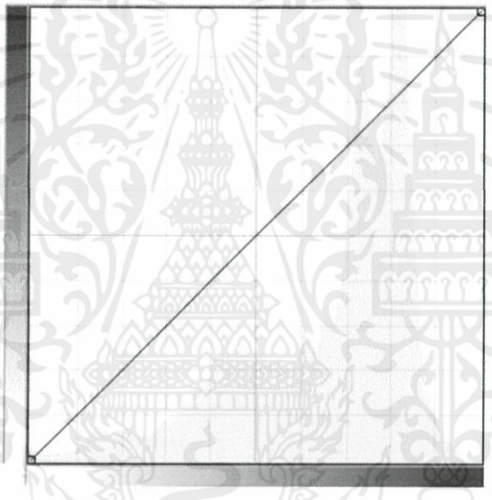
OpenCvSharp คือ OpenCV ที่เปลี่ยนแพลตฟอร์ม (Platform) ไปเป็น .NET Framework โดยใช้ภาษา C# ในการเขียน ซึ่งสามารถใช้การประมวลผลภาพและอัลกอริธึมของคอมพิวเตอร์วิทัศน์โดยใช้ภาษา C# VB .NET หรือภาษาอื่นๆในการพัฒนา อาจกล่าวได้ว่า OpenCvSharp คือ OpenCV ที่ใช้ภาษา C# ในการพัฒนานั่นเอง

2.2.3. Equalization

ขั้นตอนการตรวจจับใบหน้า จะมีขั้นตอนการทำงานดังนี้ เริ่มแรก ภาพที่รับเข้ามาเป็นได้ทั้งภาพระดับสีเทาและภาพสีที่ถูกแปลงภาพเป็นภาพระดับสีเทา ภาพสีเทาเหล่านั้นจะถูกนำไปปรับปรุงภาพด้วยวิธี Histogram Equalization เพื่อกระจายให้แต่ละระดับสีเทามีจำนวนพิกเซลใกล้เคียงกัน ซึ่งการทำการกระจายของฮิสโตแกรมนั้น ลักษณะความคมชัดและระดับความสว่างของภาพจะไม่มีผลต่อการแบ่งแยกใบหน้าคน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

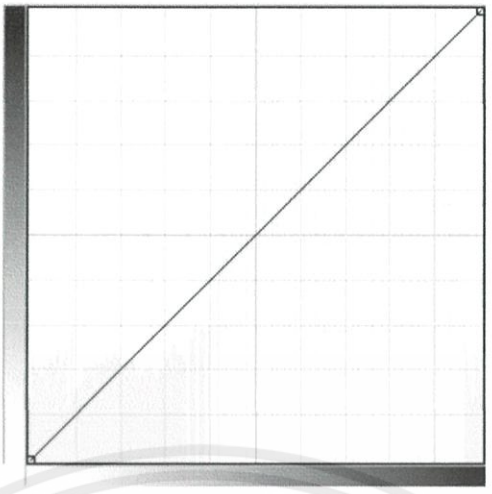
รูปที่ 2.2 ภาพต้นแบบ



รูปที่ 2.3 กราฟที่สอดคล้องกันของภาพต้นแบบ

รูปที่ 2.4 ภาพหลังจากการทำ Equalization

เอกสารนี้เป็นเอกสารที่สงวนไว้สำ ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิ ครั้งที่มีการนำไปใช้



รูปที่ 2.5 กราฟที่สอดคล้องกันของภาพหลังการทำ Equalization

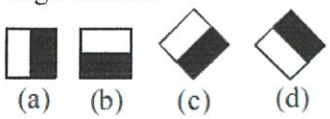
2.2.4. Haar-like Feature

การตรวจหาใบหน้าคนด้วย Haar-like Feature เป็นวิธีทางสถิติ สามารถแบ่งแยกใบหน้าคนได้โดยการเปรียบเทียบลักษณะของ Haar-like Feature จากฐานข้อมูล ที่ผ่านการฝึกให้จดจำลักษณะ Haar-like Feature ของใบหน้าคนและจดจำลักษณะ Haar-like Feature ของสิ่งที่ไม่ใช่ใบหน้าคนไว้แล้ว

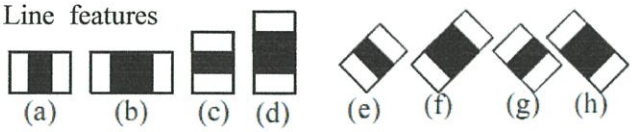
นำภาพที่ทำการปรับปรุงแล้วตรวจหาใบหน้าคน โดยทำการเลือกกรอบของพื้นที่ในภาพหาลักษณะ Haar-like Feature เปรียบเทียบกับลักษณะ Haar-like Feature จากฐานข้อมูลทั้งตัวอย่างที่ถูกและตัวอย่างที่ผิด ถ้าลักษณะ Haar-like Featue มีลักษณะของตัวอย่างที่ผิดให้ตัดสินว่าไม่มีใบหน้าคนอยู่ในกรอบที่พิจารณาทันที ทำการเปรียบเทียบกับฐานข้อมูลไปเรื่อยๆจนหมดฐานข้อมูล จึงตัดสินว่ากรอบที่พิจารณาอยู่นั้นมีใบหน้าคนอยู่ กรอบที่ใช้ในการพิจารณานี้เปลี่ยนขนาดในการตรวจหา เพื่อให้ตรวจหาใบหน้าคนได้ในหลายขนาดตามขนาดของกรอบ ขณะเปลี่ยนกรอบตรวจหานี้มีการทำงานพร้อมกันแบบขนาน เพื่อความรวดเร็วในการตรวจหาผลลัพธ์ที่ได้จาก Haar Detection คือตำแหน่งจุดศูนย์กลางของวัตถุที่ถูกพิจารณาว่าเป็นใบหน้าคน และขนาดของกรอบพื้นที่ในการตรวจหา แล้วจึงวงกลมล้อมรอบตำแหน่งที่พบบนภาพที่รับเข้ามาด้วยเส้นผ่านศูนย์กลางที่เท่ากับขนาดของกรอบพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

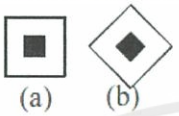
ข. Edge features



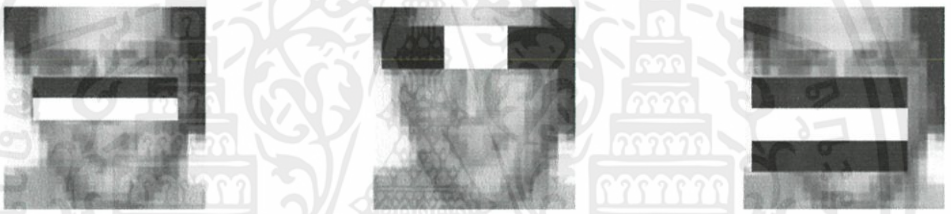
ข. Line features



ก. Center-surround features



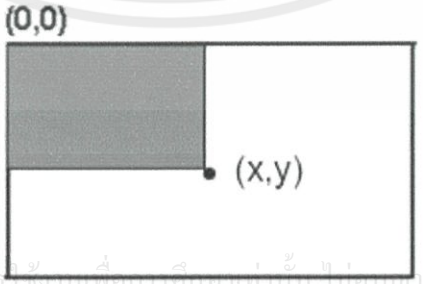
รูปที่ 2.6 รูปแบบของรูปเหลี่ยมสำหรับการตรวจจับลักษณะแบบต่างๆ[2]
(ก) ความสามารถของขอบ (ข) ความสามารถของเส้น
(ค) ความสามารถของบริเวณที่ล้อมรอบจุดตรงกลาง



รูปที่ 2.7 ตัวอย่างการใช้รูปเหลี่ยมตรวจจับลักษณะต่างๆ[3]

การคำนวณค่าของรูปเหลี่ยม (feature) นั้นใช้หลักการคำนวณแบบ Integral image คือผลรวมของค่าในทุกๆจุดภาพที่ตำแหน่ง (x, y) ใดๆดังสมการที่ 2.1

$$P(x,y) = \sum_{x' \le x, y' \le y} i(x', y') \tag{2.1}$$



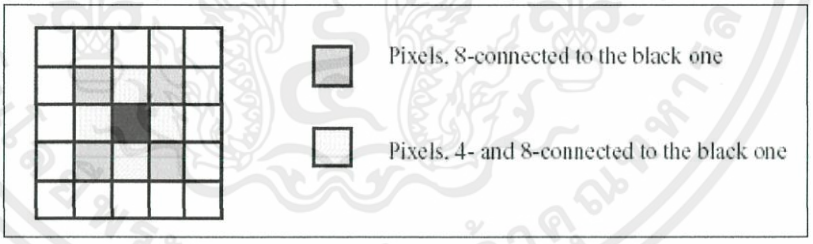
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 2.8 การคำนวณแบบ Integral image[2]

ในการทำ Haar like-feature นั้นจำเป็นต้องมีภาพตัวอย่างจำนวนมากซึ่งใช้ในการคัดเลือกลักษณะของรูปที่ต้องการตรวจจับและตีความหมายซึ่งมีสองลักษณะคือ Positive Image หรือรูปที่มีวัตถุนั้นๆประกอบอยู่ภายในภาพและ Negative Image หรือภาพใดๆที่ไม่มีวัตถุที่เราต้องการอยู่ภายในภาพจากนั้นใช้ขั้นตอนวิธีของ AdaBoost (Adaptive Boost) ซึ่งเป็นกระบวนการหารูปเหลี่ยมที่มีลักษณะใกล้เคียงและแตกต่างกับภาพนำเข้าสำหรับการจัดประเภทของภาพโดยการถ่วงน้ำหนักให้ส่วนต่างๆภายในภาพบนภาพ Positive และภาพ Negative เพื่อใช้หาลักษณะของวัตถุที่ “ใช่” และ “ไม่ใช่” ในลักษณะต่างๆ

2.2.5. การหาเส้นรูปร่าง

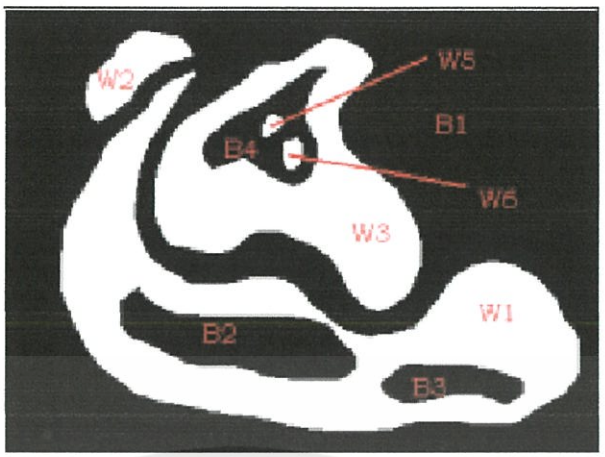
เป็นการใช้ลำดับขั้นตอนของการทำงานเกี่ยวกับเรื่องเวคเตอร์ ซึ่งลำดับขั้นตอนของการทำงานที่ใช้หารูปร่างของแรสเตอร์อิมเมจ (raster image) ในการแปลงเป็นภาพขาวดำ (binary image) ซึ่งภาพขาวดำคือภาพที่มีส่วนประกอบด้วย 0-พิกเซลซึ่งก็คือพิกเซลที่มีค่าเป็น 0 ที่ให้สีพิกเซลเป็นสีดำ และ 1-พิกเซลซึ่งก็คือพิกเซลที่มีค่าเป็น 1 ที่ให้สีพิกเซลเป็นสีขาว เซตของ 0-พิกเซล หรือ 1-พิกเซลที่เชื่อมติดกันจะทำให้เกิด 0-(1-) คอมโพเนนท์ (0-(-1)Component) โดยสามารถแบ่งประเภทของการเชื่อมติดกันเป็น 2 ชนิดได้แก่ แบบ 4-คอนเนคทิวิตี (connectivity) ซึ่งสามารถเขียนให้อยู่ในรูปสมการ $|x'-x''|+|y'-y''|=1$ และ 8-คอนเนคทิวิตี ซึ่งสามารถเขียนให้อยู่ในรูปสมการ โดยกำหนดให้พิกเซล 2 พิกเซลอยู่ที่พิกัด (x', y') และ (x'', y'')



รูปที่ 2.9 รูปแบบการเชื่อมติดกันของพิกเซล[3]

จากความสัมพันธ์ของสมการตามที่กล่าวมานี้รูปจะถูกแบ่งออกเป็นส่วนประกอบเป็นส่วนๆที่ไม่มีการทับซ้อนกัน 1-(0-) 4-คอนเนค (8-คอนเนค) ในแต่ละเซตจะประกอบไปด้วยพิกเซลที่มีค่าที่เท่ากันซึ่งก็คือทั้งหมดเป็นพิกเซลที่มีค่าเป็น 0 หรือ 1 และมีคู่ของพิกเซลที่อยู่ระหว่างจุดสองจุดของอีกเซตรูปส่วนประกอบต่างๆจะมีความสัมพันธ์กันดังรูปที่ 2.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.10 ลำดับชั้นของส่วนประกอบที่เชื่อมติดกัน[3]

1-คอมโพเนนท์ W1, W2 และW3 อยู่ภายในเฟรม (0-คอมโพเนนท์ B1) แสดงว่าถูกล้อมรอบด้วย B1

0-คอมโพเนนท์ B2 และ B3 อยู่ภายใน W1

1-คอมโพเนนท์ W5 และ W6 อยู่ภายใน B4 ซึ่งอยู่ภายใน W3 ดังนั้น 1-คอมโพเนนท์อยู่ภายใน W3 โดยตรงแต่ทั้ง W5 และ W6 ไม่ได้อยู่ติดกับส่วนอื่น หมายความว่าส่วนที่กล่าวถึงนี้อยู่คนละระดับชั้นกันกับส่วนอื่น

หากต้องการหลีกเลี่ยงความขัดแย้งที่เกิดจากโครงสร้าง 0-พิกเซลจะถือว่าเป็นพิกเซล 8-(4-) คอนเนคในกรณีที่ 1-พิกเซลจะเกี่ยวข้องกับ 4-(8-) คอนเนคโดยจะสมมติว่าจากภาพนี้ 1-พิกเซลเป็น 8-คอนเนคและ 0-พิกเซลเป็น 4-คอนเนค

จากการที่ 0-คอมโพเนนท์จะเสริมให้ 1-คอมโพเนนท์มีความสมบูรณ์และแยก 1-คอมโพเนนท์ให้ออกจากชิ้นส่วนอื่นที่ไม่ติดกัน ไลบรารี่ OpenCV จึงพิจารณาเพียงแค่ส่วนที่เป็นโครงสร้างในการศึกษาเท่านั้นส่วนที่เป็น 0-พิกเซลจะถือว่าเป็นภาพพื้นหลังในส่วนของ 0-คอมโพเนนท์ที่ถูกล้อมรอบด้วย 1-คอมโพเนนท์โดยตรงก็จะเรียกว่าหลุมโฮลล์ (hole) ของ 1-คอมโพเนนท์จุดในเส้นขอบของ 1-คอมโพเนนท์ก็คือพิกเซลใดๆที่อยู่ในส่วนนั้นและมี 4-คอนเนคเชื่อมกับ 0-พิกเซลเซตของจุดในเส้นขอบนั้นจะเรียกว่าเส้นขอบ (border)

ในแต่ละ 1-คอมโพเนนท์จะมีเส้นขอบภายนอก (outer border) ซึ่งจะแบ่งตัวเองออกจาก 0-คอมโพเนนท์ที่ล้อมรอบอยู่และอาจจะมีเส้นขอบของหลุมโฮลล์ (hole border) มากกว่า 1 หลุม โฮลล์ที่แบ่งส่วนของ 1-คอมโพเนนท์ออกจาก 0-คอมโพเนนท์ที่อยู่รอบๆจะเห็นได้ชัดว่าเส้นขอบภายนอกและเส้นขอบหลุมโฮลล์จะทำให้สามารถอธิบายรูปของส่วนประกอบนั้นได้จนครบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังนั้นเมื่อมีเส้นขอบทั้งหมดก็สามารถชี้ให้เห็นถึงเส้นรูปร่าง (contour) ของทุกๆชิ้นส่วนได้ด้วย การเก็บข้อมูลทั้งหมดเกี่ยวกับการจัดลำดับขั้นแทนด้วยการบีบอัดภาพของรูปไบนารีต้นแบบ

2.2.6. บริเวณที่สนใจ Region-of-interest (ROI)

Region-of-interest (ROI) คือบริเวณที่เราสนใจ โดยการตีกรอบล้อมรอบบริเวณที่สนใจด้วยวงกลม กรอบสี่เหลี่ยม หรือกรอบรูปเหลี่ยมใดๆ ซึ่งอาจจะเป็นบริเวณใดภายในภาพก็ได้ เพื่อนำภาพเฉพาะส่วนดังกล่าวมาประมวลผล หรือเปลี่ยนแปลงภาพตามต้องการ โดยไม่มีผลกระทบกับส่วนอื่นๆ ซึ่งภายในหนึ่งภาพสามารถกำหนดได้หลายๆบริเวณที่สนใจ

2.3. ระบบสี

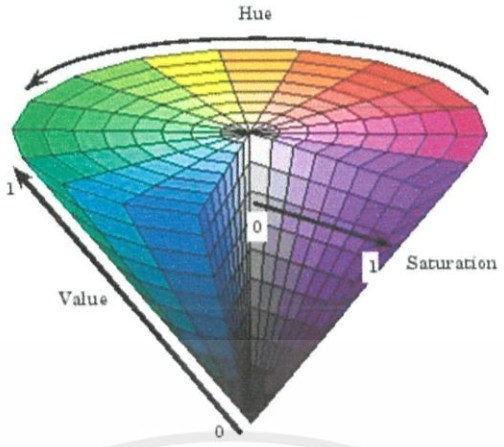
2.3.1. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบอาร์จีบี (RGB Image)

ภาพสีระบบอาร์จีบีเป็นที่นิยมอย่างมากในการบันทึกภาพดิจิทัล โดยภาพจะแสดงสีในแต่ละพิกัดออกมาจากการผสมกันของ สีแดง (Red) สีเขียว (Green) และสีน้ำเงิน (Blue) ภาพสีระบบอาร์จีบีนั้นจะถูกเก็บอยู่ในรูปแบบของอาร์เรย์ 3 มิติที่มีขนาด m x n x 3 โดยที่ m และ n แทนขนาดความยาวและความกว้างของภาพ ส่วนมิติสุดท้ายนั้นจะท าการเก็บค่าของสีแดง สีเขียว และสีน้ำเงิน

2.3.2. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ HSV

ประกอบด้วย 3 องค์ประกอบได้แก่ ค่า H (Hue) คือค่าโทนสี หรือค่าสีสัน ซึ่งมีค่าสีแตกต่างกันไปตามความถี่ของแสง ค่า S (Saturation) คือค่าความอิ่มตัวสี ซึ่งเป็นค่าที่แสดงระดับสีเมื่อเทียบกับค่าโทนสี และค่า V (Value)หรือB (Brightness)คือค่าบอกระดับความสว่างของภาพ ซึ่งที่ระดับความสว่างต่ำสุดหมายถึงสีดำ ไม่ว่าค่าโทนสีหรือค่าความอิ่มตัวสีเท่าใด และระดับความสว่างสูงสุดหมายถึงสีขาว ซึ่งเป็นสีที่สว่างที่สุดของค่าโทนสีและค่าความอิ่มตัวสี

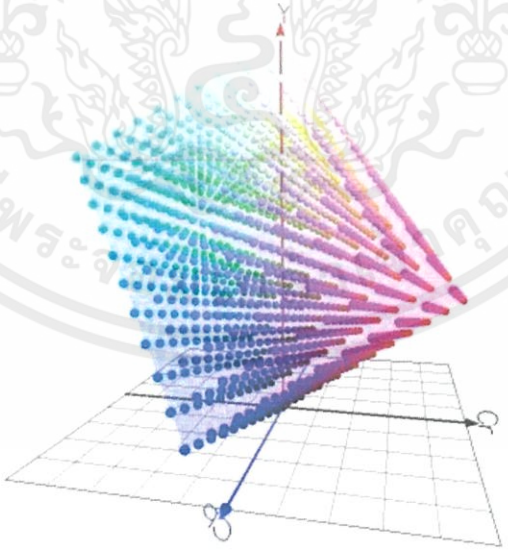
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.11 แบบจำลองสี HSV[4]

2.3.3. ความรู้พื้นฐานเกี่ยวกับภาพสีระบบ YCbCr

YCbCr , Y′CbCr, or Y Pb/Cb Pr/Cr, หรือสามารถเขียนเป็น YC_BC_R or $Y'C_BC_R$ เป็นหนึ่งในระบบการเข้ารหัสสัญญาณสีที่นำมาใช้ในวีดีโอ Component และระบบระบบภาพดิจิตอล ความแตกต่างระหว่าง YCbCr กับ RGB ก็คือ YCbCr แสดงภาพโดยใช้สัญญาณความสว่างและสัญญาณความต่างสี (Color Difference) อีกสองสัญญาณ ส่วน RGB แสดงภาพด้วยสัญญาณสีแดง เขียว และน้ำเงิน คำว่า YCbCr ตัวอักษร Y มาจากความสว่าง (Luminance) Cb คือสีน้ำเงินที่ตัดความสว่างออกไป (B-Y) Cr คือสีแดงที่ตัดความสว่างออกไป (R-Y)



รูปที่ 2.12 แบบจำลองสี YCbCr[5]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4. งานวิจัยที่เกี่ยวข้อง

จากการศึกษาของ Rahman et al.[6] เพื่อกำหนดขอบเขตพื้นที่สีผิวจากการวิเคราะห์การกระจายค่าข้อมูลสีผิว โดยใช้โมเดลสีทั้งสามได้แก่ RGB, HSV และ YCbCr ที่ใช้ในงานวิจัยนี้เกี่ยวกับการการตรวจจับพื้นที่ใบหน้าโดยใช้โมเดลสี RGB-H-CbCr ในโมเดล

กฎการกำหนดขอบเขตพื้นที่สีผิวจากโมเดลสี RGB นำเสนอโดย Peer et al.[7] ซึ่งกฎการกำหนดของเขตพื้นที่สีผิวโดยโมเดลสี RGB แบ่งเป็นบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางวันและบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางคืน ซึ่งบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางวันสามารถหาได้จากสมการดังนี้

$$\begin{aligned}&(R > 95)AND(G > 40)AND(B > 20)\ \ AND\\&(\max\{R,G,B\} - \min\{R,G,B\} > 15)\ \ \ AND\\&(|R - G| > 15)AND(R > G)AND(R > B)\end{aligned} \tag{2.2}$$

ในขณะบริเวณพื้นที่สีผิวที่ได้รับแสงในช่วงกลางคืน จะหาได้จากสมการดังนี้

$$\begin{aligned}&(R > 220)AND(G > 210)AND(B > 170)\ \ AND\\&(|R - G| \le 15)AND(R > B)AND(G > B)\end{aligned} \tag{2.3}$$

เพื่อพิจารณาเงื่อนไข จะใช้ลอจิก OR ในการรวมสมการทั้งสองเข้าด้วยกันคือ สมการที่ 2.2 OR กับสมการที่ 2.3 จะได้กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี RGB

$$\text{สมการที่}(2.2)\ \cup\ \text{สมการที่}(2.3) \tag{2.4}$$

สำหรับนำไปใช้ในการตรวจจับพื้นที่บริเวณสีผิว เพื่อกำหนดขอบเขตบริเวณพื้นที่สีผิว โดยวิเคราะห์การกระจายค่าข้อมูลสีผิวในโมเดลสี YCbCr พบว่าการกระจายค่าข้อมูลสีผิวในค่าข้อมูลสี Cb และ Cr จะเกาะกลุ่มอยู่ในช่วงตรงกลางระหว่าง 0 ถึง 255 จึงทำให้สามารถทราบค่าข้อมูลสีผิวที่ชัดเจน ทฤษฎีนี้จึงทำการประมาณการค่าข้อมูลโดยนำคณิตศาสตร์พื้นฐานสมการเชิงเส้นมาใช้ในการหาขอบเขตพื้นที่การกระจายค่าข้อมูลสีผิว จะได้สมการดังนี้

$$Cr \le 1.5862 \times Cb + 20 \tag{2.5}$$

$$Cr \ge 0.3448 \times Cb + 76.2069 \tag{2.6}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$Cr \ge -4.5652 \times Cb + 234.5652 \tag{2.7}$$

$$Cr \le -1.15 \times Cb + 301.75 \tag{2.8}$$

$$\mathbf{Cr} \leq \mathbf{-2.2857} \times \mathbf{Cb} + \mathbf{432.85} \quad (2.9)$$

เพื่อพิจารณาเงื่อนไข จะใช้ลอจิก AND ในการรวมสมการที่ 2.5-2.9 เข้าด้วยกันคือ สมการที่ 2.5 AND กับสมการที่ 2.6 AND กับสมการที่ 2.7 AND กับสมการที่ 2.8 AND กับสมการที่ 2.9 จะได้กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี YCbCr

$$\textbf{สมการที่}(2.5) \cap \textbf{สมการที่}(2.6) \cap \textbf{สมการที่}(2.7) \cap \textbf{สมการที่}(2.8) \cap \textbf{สมการที่}(2.9) \quad (2.10)$$

โมเดลสี HSV จากการวิเคราะห์การกระจายค่าข้อมูลสีผิวในโมเดลสี HSV พบว่าการกระจายค่าข้อมูลสีผิวในค่าสี (Hue) แสดงให้เห็นได้ชัดเจนที่สุดในการแบ่งแยกระหว่างสีผิวและไม่ใช่สีผิว ซึ่งดูได้จากความเข้มข้นของค่าสี (Hue) จะได้สมการดังนี้

$$(\mathbf{H} < \mathbf{5})\mathbf{OR}(\mathbf{H} > \mathbf{130}) \quad (2.11)$$

กฎการกำหนดขอบเขตพื้นที่สีผิวโดยโมเดลสี HSV

2.5. Unity

Unity คือเกมเอนจิ้นที่ถูกพัฒนาจากเทคโนโลยีของ Unity สำหรับนักพัฒนาทั่วโลกกว่าหนึ่งล้านคน เพื่อให้สามารถรองรับได้หลากหลายแพลตฟอร์มและมีเครื่องมือในการพัฒนาโปรแกรมเพื่ออำนวยความสะดวกต่างๆในการพัฒนาวีดีโอเกมที่ใช้สำหรับเล่นบนเว็บไซต์ แพลตฟอร์มของคอมพิวเตอร์ เครื่องมือควบคุมวีดีโอเกม (เช่น joystick) และอุปกรณ์มือถือ ในช่วงแรกของ Unity เป็นเครื่องมือในการพัฒนาเกมที่รองรับเพียงเฉพาะแพลตฟอร์มโอเอสเอกซ์ หลังจากนั้นจึงมีการพัฒนาเรื่อยมา จนกระทั่งสามารถรองรับได้หลายแพลตฟอร์มในปัจจุบัน และโดยใช้ภาษา C# หรือ C++ ในการพัฒนา มีการปรับปรุงรุ่นล่าสุดคือ Unity4.2 ซึ่งถูกปล่อยออกมาเมื่อเดือนกรกฎาคม ปีค.ศ.2013 โดยสามารถรองรับการพัฒนาจากทั้ง ไอโอเอส แอนดรอยด์ วินโดส์ แบล็กเบอร์รี่ 10 โอเอสเอกซ์ ลินุกซ์ เว็บบราว์เซอร์ แฟลช เพลย์สเตชัน 3 เอกซ์บอกซ์ 360 วินโดส์โฟน 8 และ วียู ซึ่งเปิดให้สามารถดาวน์โหลดแล้วทั้ง Unity และ Unity Pro

เอกสารนี้เป็น[illegible]ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.1. การแสดงผล

กราฟฟิคเอนจิ้นที่ใช้ได้แก่ Direct3D (วินโดส์ เอกซ์บอกซ์ 360) OpenGL (แมค วินโดส์ ลินุกซ์ PS3) OpenGL ES (แอนดรอยด์ ไอโอเอส) และ proprietary APIs (API ที่ถูกทำให้ใช้ได้เฉพาะกับอุปกรณ์ชนิดใดชนิดหนึ่งเท่านั้น(Wii)) โดยสามารถรองรับการทำความขรุขระของพื้นผิว การทำให้วัตถุมีความสามารถในการสะท้อน การทำให้พื้นผิวมีความลึกและสมจริง การเพิ่มความสมจริงของเงาจากแสงสะท้อนรอบทิศทาง การคำนวนเงาโดยใช้กระบวนการใส่เงาลงไปในงานคอมพิวเตอร์กราฟฟิคสามมิติ การนำแสงที่ได้จากการคำนวณจากการ render มาใส่ลงใน texture และ full-screen post-processing effects

Unity สามารถรองรับไฟล์งานจากโปรแกรม 3ds Max, Maya, Softimage, Blender, modo, Zbrush, Cinema 4D, Adobe Photoshop, Adobe Fireworksและ Allegorithmic Substance ซึ่งสามารถเพิ่มเข้าไปในโปรเจคเกมและจัดการไฟล์เหล่านั้นผ่านหน้าต่างการใช้งานกราฟฟิคของ Unity

ภาษาที่ใช้ในการทำ shaders คือภาษา ShaderLab สามารถรองรับทั้งหลักการเขียนโปรแกรมแบบfixed-function pipeline และ shader ที่ใช้เขียนใน GLSL(OpenGL Shading Language) หรือ Cg(C for Graphics) โดยทั้ง GLSL และ Cg ต่างก็เป็นภาษา shading ระดับสูงเช่นเดียวกัน ซึ่งภายใน shader สามารถมีตัวแปรที่หลากหลายและสามารถประกาศการย้อนกลับในกรณีเฉพาะ และยังอนุญาติให้ Unity ใช้ตรวจจับตัวแปรให้เหมาะสมกับการ์ดจอ ซึ่งถ้าหากไม่มีการ์ดจอที่เข้ากันได้ ก็จะทำการย้อนกลับเพื่อหาทางเลือกใหม่ของ shader ซึ่งอาจจะเสียคุณลักษณะบางประการในการแสดงผล

Unity 4.2 รุ่นล่าสุดที่ปล่อยออกมาเมื่อวันที่ 3 สิงหาคม 2013 นั้นได้อนุญาติให้นักพัฒนาอิสระสามารถใช้การให้เงาแบบ เวลาจริง เฉพาะสำหรับการให้แสงแบบมีทิศทางเท่านั้น และเพิ่มการรองรับ DirectX11 ที่ให้เงาที่มีความละเอียดของพิกเซลที่สมบูรณ์มากขึ้นเพิ่มพื้นผิวในการสร้างวัตถุสามมิติจากภาพระดับสีเทา กราฟฟิคใบหน้าที่มากขึ้น การทำอนิเมชันที่เรียบเนียนกว่าเดิม และรองรับการทำให้เป็น FPS

2.5.2. การเขียนสคริปต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สคริปต์ของเกมเอนจิ้นถูกสร้างขึ้นจาก Mono2.6 ซึ่งเป็นโอเพนซอร์ซที่ใช้ .NET Framework โดยโปรแกรมเมอร์สามารถใช้สคริปต์ Unity (ภาษาที่กำหนดขึ้นเองและใช้โครงสร้าง

จากสคริปต์ ECMA โดยอ้างอิงกับ JavaScript) C# หรือ Boo (ใช้โครงสร้างจากภาษา Python) โดยตั้งแต่รุ่น 3.0 เป็นต้นไปการปรับปรุงสคริปต์เพื่อแก้ไขจุดบกพร่องจากทีมพัฒนาของบริษัทโมโนจะถูกรวมเข้ากับ Unity ไว้ด้วยกัน

2.5.3. Asset Tracking

Unity มีการรวม Unity Asset Server ซึ่งเป็นวิธีการควบคุมรุ่นสำหรับผลงานและสคริปต์ของนักพัฒนาเกม มีการใช้ PostgreSQL เป็นส่วนในการจัดการระบบ ระบบของเสียงนั้นสร้างจากไลบรารี่ FMOD(ที่มีความสามารถในการเล่นเสียงที่ผ่านการบีบอัดมาแล้วของบริษัท Ogg Vorbis) ใช้ Theora codec ในการเล่นวีดีโอ มีเอนจิ้นการทำภูมิศาสตร์และพฤกศาสตร์(ใช้ Umbra ที่รองรับ tree billboarding และ Occlusion Culling)มี light mapping และการให้แสงแบบทั้งระบบ จาก Beast มีการใช้ RakNet ในการติดต่อกับผู้เล่นจำนวนหลายคน และมีการค้นหาเส้นทางแบบตาข่าย

2.5.4. แพลตฟอร์ม

สามารถรองรับการใช้งานหลากหลายแพลตฟอร์ม นักพัฒนาสามารถควบคุมโปรเจคที่จะพัฒนาสำหรับอุปกรณ์มือถือ เว็บบราว์เซอร์คอมพิวเตอร์และเครื่องมือควบคุมวีดีโอเกม(เช่น joystick) โดยสามารถตั้งค่าเฉพาะสำหรับการแปลงพื้นผิวและความละเอียดของเกมในแต่แพลตฟอร์มให้มีความแตกต่างกันออกไปได้อีกด้วย

ในปัจจุบันสามารถรองรับแพลตฟอร์ม ไอโอเอส แอนดรอยด์ วินโดส์ แมค ลินุกซ์ Unity web Player Adobe Flash เพลย์สเตชัน 3 เอกซ์บอกซ์ 360 และ Wii ซึ่งยังสามารถรองรับเพลย์สเตชัน Vita ที่สามารถพบเห็นได้ในเกม Escape Plan และ Oddworld:New'n'Tasty แต่ข่าวนี้ยังไม่ได้รับการยืนยันอย่างเป็นทางการจากทาง Unity

Unity รุ่นล่าสุดที่กำลังมาสามารถรองรับแพลตฟอร์ม แบล็กเบอร์รี่ 10 เพลย์สเตชัน 4 วียู เอกซ์บอกซ์วัน วินโดส์โฟน 8 ทั้งยังมีข่าวลืออีกว่า ในอนาคตจะสามารถรองรับ HTML และ Adobe plug-in รุ่นใหม่ที่จะมาแทนที่ Flash Player อีกด้วย

2.5.5. Asset Store

Unity Asset Store เปิดตัวครั้งแรกเมื่อเดือนพฤศจิกายนปี 2010 ซึ่งเป็นแหล่งทรัพยากรที่อยู่ภายใน Unity editor โดยมีการรวบรวมผลงานเป็นจำนวนมากกว่า 4,400 ชิ้น ประกอบไปด้วย โมเดลสามมิติ พื้นผิวและวัสดุ ชิ้นส่วนของระบบ เพลงและเสียงเอฟเฟค คู่มือและโปรเจคชุดคำสั่งสคริปต์ส่วนแก้ไขเพิ่มเติมและการบริการออนไลน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.5.6. ฟิสิกส์

Unity มีการรองรับ Nvidia's(ชื่อเรียกอย่างเป็นทางการคือ Ageia's) PhysX ซึ่งเป็นฟิสิกส์เอนจิ้น(นำมาใช้ตั้งแต่Unity 3.0) ที่รองรับการจำลองการเคลื่อนไหวของเสื้อผ้าและผิวหนังการคำนวนความทึบโปร่ง และการชนกันแบบ เวลาจริง

2.5.7. รุ่น

Unity รุ่นแรกได้เปิดตัวในงานประชุมนักพัฒนา Apple จากทั่วโลกในปี 2005 ซึ่งฟังก์ชันการทำงานสร้างมาเพื่อรองรับโปรเจคที่ทำงานบนคอมพิวเตอร์แมคเท่านั้นและเริ่มพัฒนาให้สามารถรองรับแพลตฟอร์มอื่นๆเรื่อยมา โดย Unity 3 ที่ถูกปล่อยออกมาในเดือนกันยายนปี 2010 ได้มุ่งเน้นไปที่การแนะนำเครื่องมือที่สตูดิโอระดับใหญ่สามารถนำไปใช้ได้ ซึ่งเป็นการทำให้บริษัทหันเหความสนใจไปที่กลุ่มนักพัฒนาที่มีขนาดใหญ่ขึ้น รวมไปถึงการทำเกมเอนจิ้นแพคเกจสำหรับนักพัฒนาอิสระและทีมพัฒนาขนาดเล็กให้สามารถซื้อหาได้ โดยรุ่น 4.0 ที่ถูกปล่อยออกมาล่าสุดในปลายปี 2012 ได้มีการเพิ่มการรองรับ Mecanium animation และ DirectX11

2.5.7.1. Unity 4

Unity 4 มีการประกาศในวันที่ 18 มิถุนายน 2012 ว่าจะมีการเพิ่มเทคโนโลยีเล็กๆน้อยๆ ดังจะคล้ายกับการปล่อยออกมาของรุ่นก่อนๆ ซึ่งมีทั้งการปรับปรุงและเพิ่มคุณสมบัติบางอย่างเช่น การปรับปรุงGUIใหม่ และมีการจำหน่ายจริงในวันที่ 14 พฤศจิกายน 2012

คุณสมบัติใหม่ที่เพิ่มขึ้นมานั้นได้แก่ การรองรับ DirectX 11 และ Mecanim animation ซึ่งภายในกราฟฟิคของอุปกรณ์สื่อสารมีการให้เงาแบบเวลาจริง skinned mesh instancing, ความสามารถในการใช้แผนที่ธรรมดาเมื่อมีการใช้ baking lightmaps และการคัดกรอง GPU profiler อย่างไรก็ตามการใช้งาน Adobe Flash add-on สามารถใช้กับ Unity รุ่น 4.0 ขึ้นไปเท่านั้น (รองรับเฉพาะกับเครื่องมือของ Unity 3.5 ที่เป็นรุ่น beta) และยังมีความลือว่า Unity 4.2 จะเพิ่มคุณสมบัติของเงาที่รองรับการทำงานสำหรับรุ่นฟรี แต่ก็ยังมีข้อจำกัดบางประการเช่น อาจจะจำกัดให้สามารถใช้แสงได้เพียงทิศทางเดียว เป็นต้น

Unity 4 เป็นการเพิ่มทางเลือกในการทำให้เกมสามารถเล่นได้หลากหลายแพลตฟอร์ม และมีการเริ่มต้นการพัฒนาไปสู่การใช้งานบน Ubuntu โดยไม่เสียค่าใช้จ่ายเพิ่มเติมใดๆทั้งสิ้น และตั้งแต่ Unity 4.0 เป็นต้นไป จะมีการร่วมงานกับทาง Facebook เพื่อรองรับสำหรับแพลตฟอร์มโซเชียลซึ่งมีชื่อเรียกว่า Unity Web Player

เอกสารนี้เป็น
ไม่ว่ากรณีใด

2.5.7.2.Mecanim

Mecanim คือเทคโนโลยีแอนิเมชันของ Unity ที่มีการพัฒนามาหลายปีแล้วจากบริษัทเล็กๆที่มีชื่อเดียวกันนี้ ซึ่งภายหลังทาง Unity สาขาประเทศแคนาดาได้เข้ามาซื้อกิจการ โดยเทคโนโลยีนี้เป็นการนำเอาสสารและการเคลื่อนไหวของธรรมชาติมาสร้างลักษณะ ซึ่งมีเครื่องมือสำหรับการสร้าง state machines, blend tree, IK rigging และแอนิเมชันการกำหนดเป้าหมายใหม่โดยอัตโนมัติที่อยู่ภายใน Unity editor

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3
ขั้นตอนการออกแบบและพัฒนาแอพพลิเคชัน

3.1 หลักการและแนวคิด

ปัจจุบันนี้มีการพัฒนารูปแบบการเล่นเกมให้ก้าวหน้าไปมาก จากเกม 2 มิติเป็น 3 มิติที่เน้นความสมจริงของภาพ หรือเน้นที่มีความสวยงามเป็นแฟนตาซีราวกับโลกในจินตนาการ ซึ่งรูปแบบการเล่นส่วนใหญ่ของเกมนั้น จะเป็นการอินพุตทางคีย์บอร์ดและเมาส์ใช้ในการควบคุมบังคับเกม จึงมีแนวคิดที่ว่า ในเมื่อเกมเน้นความสมจริงของภาพซึ่งมองเห็นด้วยตาแล้ว น่าจะเพิ่มความสมจริงโดยใช้การเคลื่อนไหวร่างกายในการควบคุมเกม จากการศึกษาหลักการและทฤษฎีการประมวลผลภาพต่างๆก็สามารถนำมาประยุกต์ใช้ในการพัฒนาเกม โดยที่ได้นำกล้องเว็บแคมมาเป็นอินพุตในการตรวจจับการเคลื่อนไหวของร่างกาย เนื่องกล้องเว็บแคมมีราคาถูก ส่วนใหญ่จะมีติดมากับเครื่องคอมพิวเตอร์แบบพกพา ทำให้ไม่ต้องเสียค่าใช้จ่ายติดตั้งอุปกรณ์เพิ่มเติม

ปัญหาพิเศษนี้มีเป้าหมายที่จะพัฒนาเกม โดยใช้การตรวจจับและการติดตามการเคลื่อนไหวของร่างกายมนุษย์ที่แสดงผ่านกล้องเว็บแคม ในการควบคุมการเคลื่อนไหวภายในเกม ซึ่งจะนำไลบรารี่ของ OpenCV มาใช้ในส่วนของการประมวลผลภาพ คือการตรวจจับ ติดตามการเคลื่อนไหวของใบหน้าและมือของผู้เล่น โดยที่จะออกแบบเกมโดยใช้เกมเอนจิ้นที่ชื่อว่า Unity ช่วยในสร้างเกมและองค์ประกอบต่างๆภายในเกม เกมนี้จะออกแบบมาเพื่อการใช้งานที่สอดคล้องกับการเคลื่อนไหวของร่างกายมนุษย์ที่ใช้ในการควบคุมเกม

โดยหลักการของปัญหาพิเศษนี้คือ การตรวจจับใบหน้าโดยใช้หลักการ Haar-like Feature และการตรวจจับสีผิวโดยวิธีการกำหนดขอบเขตสีผิวจากใบหน้า จากนั้นทำการกำหนดจุดกึ่งกลางบริเวณใบหน้าและบริเวณมือของผู้เล่น เพื่อนำไปใช้ในการวิเคราะห์ท่าทางและส่งไปยังส่วนของเกมเพื่อคุมควบการเคลื่อนไหวของผู้เล่นที่จะแสดงผลภายในเกม

3.1.1 วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า

ปัญหาพิเศษนี้ได้นำเสนอวิธีการใหม่ในการกำหนดขอบเขตของสีผิว โดยการนำสีผิวทั้งหมดจากสีผิวใบหน้ามาเป็นตัวกำหนดขอบเขตแทนการกำหนดด้วยค่าคงที่ ซึ่งค่าสีผิวที่ได้มานั้นจะเป็นค่าสีผิวของตัวผู้ใช้เอง และจะปรับเปลี่ยนขอบเขตของพื้นที่สีผิวให้เหมาะสม เป็นการเพิ่มประสิทธิภาพในการตรวจจับสีผิวและลดระยะเวลาในการรวบรวมข้อมูลเพื่อหาค่าคงที่ เนื่องจากกำหนดขอบเขตจากโมเดลสีแบบทั่วไป จำเป็นต้องมีการรับข้อมูลจำนวนมากเพื่อวิเคราะห์สีผิวแล้วจึงสามารถนำมาหาค่าเฉลี่ยเพื่อกำหนดเป็นกฎ แต่วิธีการกำหนดขอบเขตสีผิวจากใบหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สามารถลดระยะเวลาในขั้นตอนดังกล่าว เพราะวิธีนี้จะไม่ใช้การรวบรวมข้อมูล แต่จะเป็นการใช้ข้อมูลจากใบหน้าที่ได้รับมา โดยสามารถแบ่งการทำงานออกเป็น 3 ส่วนหลัก ได้แก่ ขั้นตอนการเตรียมภาพโดยใช้การตรวจจับใบหน้าด้วยหลักการ Haar-like Feature เพื่อนำข้อมูลสีผิวใบหน้ามาวิเคราะห์ในโมเดลสี HSV จากนั้นทำการพล็อตกราฟแบบ 2 มิติเพื่อนับความถี่สะสมของสีโดยใช้ช่องสี H และ S มาเป็นแกน สุดท้ายนำผลลัพธ์ที่ได้จากการพล็อตกราฟไปสร้างเป็นกฎการกำหนดขอบเขตพื้นที่สีผิว

ขั้นตอนการตรวจจับสีผิวมนุษย์นั้นใช้วิธีการตรวจจับสีภาพ เป็นวิธีที่ได้รับความนิยมเป็นอย่างมาก เนื่องจากมีการประมวลผลที่รวดเร็ว และผลลัพธ์ออกมาเป็นที่น่าพึงพอใจ งานวิจัยจำนวนมากได้นำเสนอเทคนิคและวิธีการจากโมเดลสี เช่น Peer et al.[7] ใช้โมเดลสี RGB ในการตรวจจับพื้นที่สีผิวของใบหน้า Chai et al.[8] ใช้โมเดลสี YCbCr และ Wang et al.[9] ใช้โมเดลสี HSV ในการตรวจจับพื้นที่สีผิวของใบหน้าจากภาพสี ซึ่งงานวิจัยโมเดลสีส่วนมากในการตรวจจับสีผิวมนุษย์นั้น เกิดจากการรวมรวบข้อมูลสีผิวจำนวนมาก หาค่าเฉลี่ยและนำมากำหนดเป็นค่าคงที่ โดยอุปสรรคของงานวิจัยประเภทนี้คือ เมื่อรวบรวมข้อมูลสีผิวจากหลายเชื้อชาติ จึงทำให้เกิดความต่างของสีผิว ค่าเฉลี่ยที่นำมากำหนดเป็นค่าคงที่นั้นอาจมีความคลาดเคลื่อน เช่น หากข้อมูลสีผิวที่รวบรวมมามีจำนวนข้อมูลของคนผิวสีมากกว่า จะทำให้ค่าคงที่ที่ได้มานั้น อาจจะใช้กับคนเอเชียหรือคนต่างเชื้อชาติไม่ได้ประสิทธิภาพ และอุปสรรคอีกด้านหนึ่งของงานวิจัยประเภทนี้คือขั้นตอนการรวบรวมข้อมูลสีผิวจะเป็นข้อมูลเฉพาะในสภาพแสงปกติ เมื่อนำไปใช้ในสภาวะแสงผิดปกติ เช่น แสงของสภาพแวดล้อมเป็นสีน้ำเงินหรือสีเขียว จะทำให้ผลที่ได้เกิดความคลาดเคลื่อน เพราะเมื่อแสงเปลี่ยนไปแต่ค่าคงที่ยังคงเดิม การตรวจจับโดยวิธีการใช้โมเดลสีที่กำหนดค่าไว้แล้วจึงเป็นไปได้ยาก

วิธีการกำหนดขอบเขตสีผิวจากใบหน้าที่เรานำเสนอ จะทำให้ลดความคลาดเคลื่อนที่เกิดจากปัญหาดังกล่าว และจากผลการทดลอง วิธีนี้ช่วยเพิ่มค่าความถูกต้องของการตรวจจับพิกเซลสีผิวเป็น 92.951%

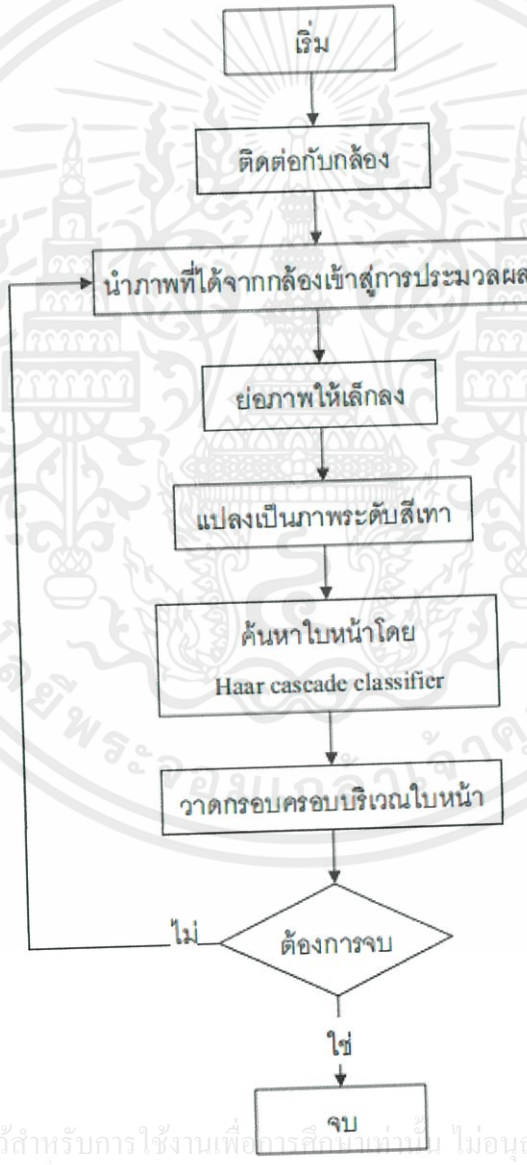
3.2 การตรวจจับใบหน้า

ขั้นตอนการเขียนโปรแกรมขั้นแรกเขียนโปรแกรมให้ติดต่อกับกล้องเพื่อนำภาพจากกล้องนำมาประมวลผล โดยการเขียนโปรแกรมให้โปรแกรมเชื่อมต่อกับกล้องเว็บแคม สามารถทำได้ด้วยการเรียกใช้ฟังก์ชันการทำงานภายในไลบรารี่ของ OpenCV เพื่อรับภาพมาประมวลผลภาพที่ได้จาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กล้องนั้นสามารถเลือกได้ตามความต้องการ ในที่นี้ต้องการภาพสีแบบ RGB เพื่อง่ายต่อการแสดงผล

การตรวจจับใบหน้านั้นทำได้โดยใช้ Haar cascade classifier ซึ่งเป็นฟังก์ชันใน OpenCV มาช่วยในการทำงาน เป็นการนำเอาไฟล์ประเภท XML ที่อยู่ในไลบรารีมาใช้ ซึ่งเป็นไฟล์ที่รวบรวมข้อมูลใบหน้ามนุษย์ที่ได้รับเรียนรู้จากรูปภาพเป็นจำนวนมาก เมื่อได้รับเฟรมภาพจากกล้องเว็บแคม ก็จะทำการย่อขนาดของภาพให้เล็กลงและแปลงภาพที่ได้มาเป็นภาพระดับเทาเพื่อให้การวิเคราะห์ได้รวดเร็วขึ้น จากนั้นประมวลผลเพื่อหาตำแหน่งของใบหน้า แล้ววาดกรอบครอบบริเวณใบหน้า



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.1 แผนภาพการตรวจจับใบหน้า

3.3 ขั้นตอนการแยกสีผิวจากวิธีการกำหนดขอบเขตสีผิวจากใบหน้า

3.3.1 การเตรียมภาพ

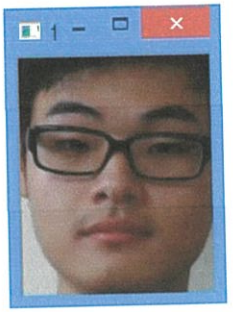
หลักการในการแยกสีผิวขั้นตอนแรกนั้น จะคล้ายกับขั้นตอนการแยกใบหน้าดังที่กล่าวข้างต้น โดยเมื่อรับภาพเข้ามาในระบบ ภาพที่รับมาจะทำการย่อขนาดลงดังแสดงในรูปที่ 3.3(ก) ซึ่งคุณภาพของภาพที่ได้จะลดลงตามไปด้วย แต่จะทำให้การประมวลผลของระบบมีการประมวลผลได้รวดเร็วมากยิ่งขึ้น จากนั้นนำภาพที่ทำการย่อขนาดแล้ว มาตรวจจับใบหน้าจากภาพโดยวิธี Haar-like Feature ซึ่งเป็นฟังก์ชันหนึ่งใน OpenCV ผลลัพธ์ที่ได้จากการตรวจจับจะเป็นรูปที่ตัดเอาเฉพาะบริเวณใบหน้ามาเท่านั้นดังแสดงในรูปที่ 3.3(ข) และเป็นการตัดส่วนที่เป็นพื้นหลังของภาพและสิ่งรบกวนต่างๆออกไป เช่น เส้นผม เพื่อการประมวลผลที่แม่นยำมากยิ่งขึ้น ในส่วนนี้เป็นขั้นตอนการหาภาพใบหน้าและนำผลลัพธ์ที่ได้ไปวิเคราะห์หาสีผิวในขั้นตอนถัดไป

ภายในเกม เนื่องจากมีการควบคุมตัวละครโดยการใช้มือร่วมด้วย ดังนั้นในส่วนนี้จะมีการวิเคราะห์มือของผู้เล่น โดยเริ่มการแบ่งกึ่งกลางภาพที่ได้รับจากกล้องดังแสดงในรูปที่ 3.2 เพื่อแยกภาพออกเป็นสองส่วน ส่วนแรกทางฝั่งซ้ายของภาพจะหมายถึงส่วนที่ค้นหามือซ้ายของผู้เล่น และส่วนที่สองทางฝั่งขวาของภาพจะหมายถึงส่วนที่ค้นหามือขวาของผู้เล่น ซึ่งการแบ่งภาพแต่ละครั้งไม่จำเป็นต้องแบ่งเป็นสองส่วนเท่าๆกันเสมอไป แต่จะพิจารณาจากตำแหน่งใบหน้าของผู้เล่นร่วมด้วย

รูปที่ 3.2 ภาพที่ได้รับจากกล้อง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(ก)



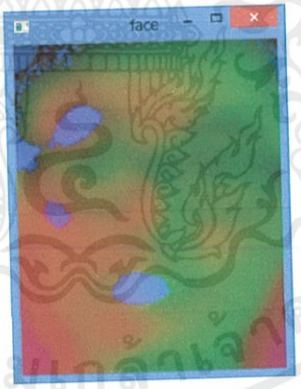
(ข)

รูปที่ 3.3 (ก) ภาพหลังจากการย่อขนาด

(ข) ภาพใบหน้าที่ได้จากตรวจจับโดยวิธี Haar-like Feature

3.3.2 การแบ่งภาพสีผิว

เริ่มต้นจากการนำภาพใบหน้าที่ได้จากการตรวจจับโดยวิธี Haar-like Feature มาทำการเพิ่มขนาดเป็น 2 เท่า เพื่อเพิ่มปริมาณของข้อมูลในการนำไปวิเคราะห์หาสีผิว หลังจากนั้นจึงทำการแปลงภาพสีบริเวณใบหน้าจากสีปกติให้อยู่ในโมเดลสี HSV เนื่องจากงานวิจัยของ Chaves-GonzÃ¡lez[10] ในการเปรียบเทียบการตรวจจับสีผิวของโมเดลสีต่างๆ โมเดลสี HSV นั้นมีอัตราความถูกต้องในการตรวจจับสีผิวได้มากกว่าโมเดลสีอื่นๆ

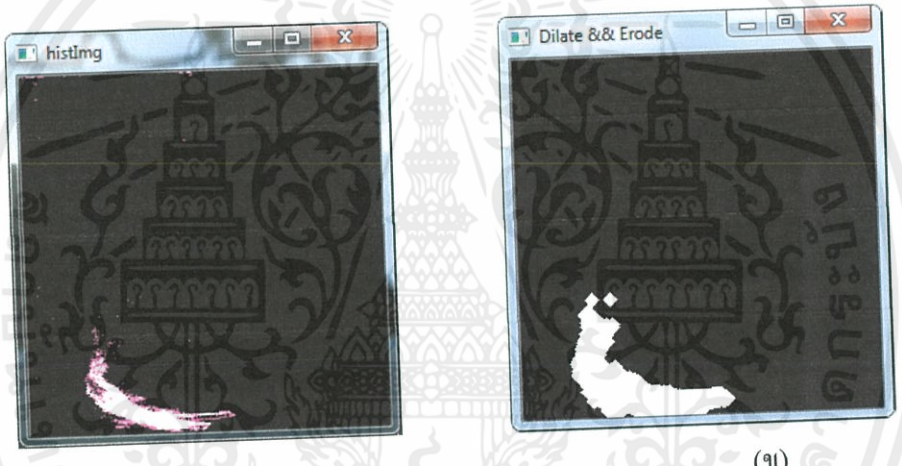


รูปที่ 3.4 ภาพใบหน้าขยาย 2 เท่าและเปลี่ยนเป็นโมเดลสี HSV

หลังจากแปลงภาพสีของใบหน้าจากโมเดลสีปกติให้อยู่ในโมเดลสี HSV(รูปที่ 3.4) แล้วจากนั้นจึงทำการสร้างตาราง 2 มิติ เพื่อเก็บรวบรวมข้อมูลความถี่สะสมของพิกเซลภายในภาพ โดยให้แกน X แทนช่องสี H ตั้งแต่ 0-255 และแกน Y แทนช่องสี S ตั้งแต่ 0-255 เช่นหากพิกเซลแรกของภาพใบหน้า(รูปที่ 3.4)มีค่า H=20 S=55 ในการเก็บข้อมูจะเก็บลงในแกน X=20 และ Y=55 เป็นต้น โดยจะเก็บข้อมูลไปเรื่อยๆจากทุกพิกเซลภายในภาพ หลังจากรวบรวมข้อมูลเสร็จเรียบร้อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้ว จะเป็นการพล็อตกราฟของข้อมูล 2 มิติดังแสดงในรูปที่ 3.5(ก) ซึ่งสีที่แสดงในภาพนั้นหมายถึงจำนวนข้อมูลที่มีความถี่สะสมมากน้อยแตกต่างกันไป โดยจะไล่สีตั้งแต่สีชมพูที่แสดงถึงความถี่สะสมของข้อมูลน้อยไปจนถึงสีขาวที่แสดงถึงความถี่สะสมของข้อมูลมาก จากนั้นนำรูปภาพของกราฟข้อมูลที่ได้ไปปรับปรุงคุณภาพของภาพโดยเทคนิคไดเลชัน(Dilation)และอีโรชัน(Erosion) เพื่อกำจัดสัญญาณรบกวน(Noise)ภายในภาพดังแสดงในรูปที่ 3.5(ข) หลังจากนั้นจะทำการค้นหาช่วงของข้อมูลที่มีข้อมูลรวมตัวกันหนาแน่นมากที่สุด โดยช่วงข้อมูลเหล่านั้นจะนำมากำหนดเป็นขอบเขตช่วงข้อมูลของสีผิว ซึ่งขอบเขตของสีผิวที่ได้ เป็นข้อมูลจากการวิเคราะห์ใบหน้า และจะเปลี่ยนค่าไปตามลักษณะสีผิวใบหน้าของผู้ใช้



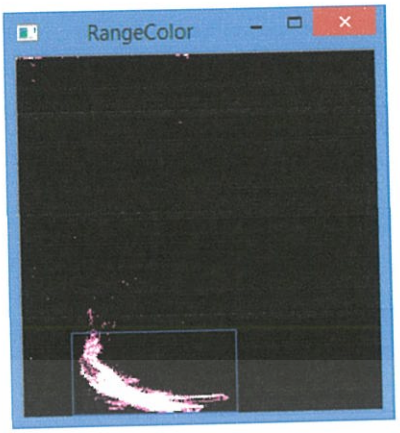
(ก) (ข)

รูปที่ 3.5 (ก) กราฟข้อมูล 2 มิติ H-S

(ข) กราฟข้อมูลหลังการปรับปรุงโดยเทคนิคไดเลชันและอีโรชัน

หลังจากการนำรูปภาพของกราฟข้อมูล ผ่านเทคนิคไดเลชันและอีโรชันดังแสดงในรูปที่ 3.5(ข)แล้ว วิธีการกำหนดขอบเขตสีผิวจากใบหน้านี้จะใช้ช่วงข้อมูลที่มีการรวมตัวกันหนาแน่นมากที่สุดมากำหนดเป็นขอบเขตของสีผิวดังกล่าว โดยจะวาดกรอบสี่เหลี่ยมครอบคลุมพื้นที่เหล่านั้น(รูปที่ 3.6) ขอบเขตของสีผิวที่ได้จึงเป็นลักษณะกราฟ 2 มิติ H-S จากนั้นนำภาพที่ได้รับจากกล้องมาเปรียบเทียบกับขอบเขตดังกล่าว โดยจะพิจารณาในแต่ละพิกเซล หากพิกเซลของภาพที่ได้รับจากกล้องมีค่าสีอยู่ในช่วงที่คาดว่าจะเป็นสีผิวมนุษย์ อีกนัยหนึ่งหมายถึง พิกเซลดังกล่าวมีค่าสีอยู่ในขอบเขตของสีผิวที่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.6 กรอบสี่เหลี่ยมล้อมรอบข้อมูลที่มีการรวมตัวกันหนาแน่นมากที่สุด

จะทำการเปลี่ยนค่าพิกเซลนั้นให้เป็นสีขาว ถ้าค่าที่นำมาเปรียบเทียบเกินกว่าช่วงของขอบเขตของช่วงสีที่กำหนดไว้ จะถือว่าพิกเซลตำแหน่งนี้เป็นพิกเซลที่ไม่ใช่สีผิว และทำการเปลี่ยนค่าในพิกเซลนั้นให้เป็นสีดำดังแสดงในรูปที่ 3.7

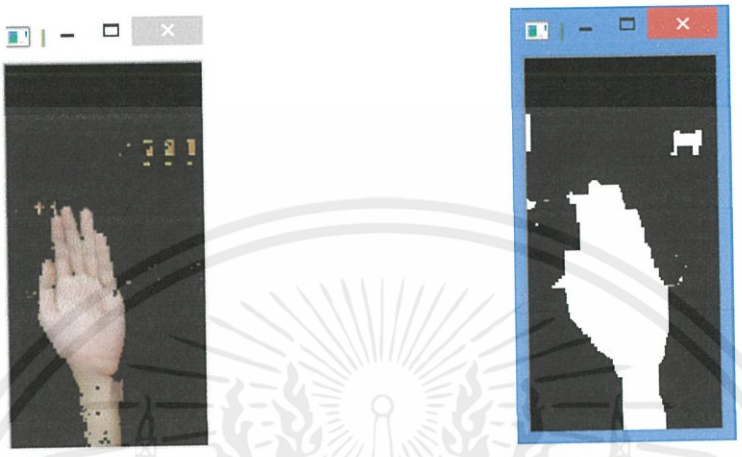
รูปที่ 3.7 ผลลัพธ์ที่ได้จากวิธีการกำหนดขอบเขตสีผิวจากใบหน้า

3.3.3 การปรับภาพให้เรียบโดยใช้มอร์โฟโลยี

ผลลัพธ์จากการแบ่งภาพสีผิวที่ได้นั้น จะมีจุดเล็กๆที่เกิดจากภาพที่มีจุดสีเพี้ยนทำให้การเลือกสีผิวผิดพลาดไป ซึ่งถือว่าเป็นสัญญาณรบกวน(Noise) และมีผลต่อการนำไปวิเคราะห์ผลในขั้นตอนการทำงานต่อไป ดังนั้นจึงต้องมีการกำจัดสัญญาณรบกวนเหล่านี้ทิ้งไป โดยนำภาพดังกล่าวมาทำการหลอมมอร์โฟโลยีจากเทคนิคไดเลชันตามด้วยเทคนิคอีโรชัน ซึ่งสามารถเรียกการกระทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แบบนี้ว่าโคลสซิง เพื่อให้ภาพใบหน้าและมือที่ได้ราบเรียบยิ่งขึ้น สามารถเชื่อมต่อเป็นเนื้อเดียวกัน ทำให้การตรวจจับมีความทนทานต่อสภาพแสงมากขึ้น



(ก) (ข)

รูปที่ 3.8 ภาพก่อน(ก)และหลัง(ข)การทำมอร์โฟโลยี

การประมวลผลมือของผู้เล่นนั้นจะกำหนดบริเวณที่เราสนใจ ที่คาดไว้ว่าเป็นมือของผู้เล่น (รูปที่ 3.8 (ข)) โดยพื้นที่นั้นจะต้องไม่เล็กกว่าค่าที่กำหนดไว้ เพื่อตัดวัตถุที่มีค่าสีใกล้เคียงกับสีผิวมนุษย์ทิ้งไป แล้วจึงหาจุดศูนย์กลางของบริเวณที่เราสนใจเพื่อกำหนดว่าตำแหน่งภาพที่ได้เป็นตำแหน่งมือของผู้เล่น และนำตำแหน่งที่ได้นี้ไปคำนวณท่าทางของผู้เล่นในการเล่นเกม

3.4 การทดลอง

รูปภาพที่นำมาใช้ในการทดสอบสามารถแบ่งออกเป็น 3 กรณีใหญ่ คือการตรวจจับใบหน้าภายในสภาพแวดล้อมและแสงปกติ การตรวจจับใบหน้าภายในสภาพแวดล้อมแสงปกติและพื้นหลังไม่มีวัตถุรบกวน และการตรวจจับใบหน้าภายในสภาพแวดล้อมแสงผิดปกติ ใบหน้าที่นำมาทดสอบมีจำนวนทั้งหมด 9 ใบหน้า ข้อมูลภาพสีที่นำมาทดสอบเป็นภาพสีที่ถ่ายจากกล้องเว็บแคมและจากอินเตอร์เน็ต ในแต่ละภาพจะมีสภาพแวดล้อมที่แตกต่างกัน ทั้งความแตกต่างของพื้นหลังและสีที่รบกวนภายในภาพ ซึ่งจะเปรียบเทียบการตรวจจับสีผิวด้วยกัน 4 วิธีคือ 1.วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า 2.วิธีการตรวจจับโดยใช้โมเดลสี RGB[7] 3.วิธีการตรวจจับโดยใช้โมเดลสี YCbCr[8] และ4.วิธีการตรวจจับโดยใช้โมเดลสี HSV[9]

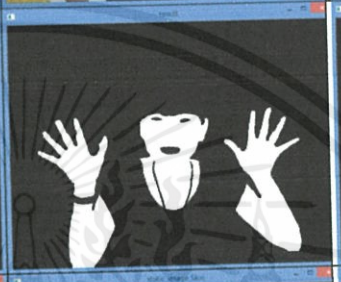
เอกสารนี้เป็น... ไม่ว่ากรณี... ...ทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.1 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 1

วิธีการ	กรณีที่ 1.1	กรณีที่ 1.2	กรณีที่ 2
ภาพต้นแบบ			
ผลที่คาดหวัง			
วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า			
RGB[7]			
YCbCr[8]			
HSV[9]			

เอกสารนี้เป็นเอกสารที่สงวน
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้ง

ตารางที่ 3.2 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 2

วิธีการ	กรณีที่ 3.1	กรณีที่ 3.2	กรณีที่ 4
ภาพต้นแบบ			
ผลที่คาดหวัง			
วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า			
RGB[7]			
YCbCr[8]			
HSV[9]			

เอกสารนี้เป็นเอกสารที่สงวน
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้ง

ตารางที่ 3.3 แสดงภาพผลการทดลองการตรวจจับสีผิวในภายสีโดยวิธีการต่างๆ 3

วิธีการ	กรณีที่ 5	กรณีที่ 6	กรณีที่ 7
ภาพต้นแบบ			
ผลที่คาดหวัง			
วิธีการกำหนดขอบเขตสีผิวทั้งหมดจากสีผิวใบหน้า			
RGB[7]			
YCbCr[8]			
HSV[9]			

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของผลงานทุกครั้ง

3.5 การวัดประสิทธิภาพ

เนื่องจากการนำเสนอวิธีการกำหนดขอบเขตสีผิวจากใบหน้านี้ เป็นวิธีการเพิ่มประสิทธิภาพการตรวจจับสีผิวภายในภาพ การพิจารณาจึงคิดจาก DSR(Detection Success Rate) ค่าความถูกต้องของการตรวจจับพิกเซลสีผิว ซึ่งเป็นอัตราส่วนของจำนวนพิกเซลสีผิวที่ตรวจจับได้ถูกต้องต่อจำนวนพิกเซลสีผิวทั้งหมดที่มีอยู่จริงในภาพ เขียนได้ดังสมการ

$$DSR = \frac{\text{จำนวนพิกเซลสีผิวที่ตรวจจับได้ถูกต้อง}}{\text{จำนวนพิกเซลสีผิวทั้งหมดที่มีอยู่จริงในภาพ}} \times 100\% \quad (3.1)$$

ตารางที่ 3.4 ผลการทดลองการตรวจจับสีผิวในภาพสี

กรณี	วิธีการกำหนดขอบเขตสีผิวจากใบหน้า DSR (%)	RGB[7] DSR (%)	YCbCr[8] DSR (%)	HSV[9] DSR (%)
1.1	92.463	76.349	99.216	56.425
1.2	97.896	84.975	99.493	90.477
2	90.205	67.487	38.868	0.3959
3.1	97.909	97.729	85.122	17.999
3.2	97.827	97.729	99.713	96.875
4	92.973	99.736	86.285	45.243
5	82.806	36.506	5.9414	0.6880
6	89.609	16.648	32.449	9.8039
7	94.865	89.645	98.994	90.958
ค่าเฉลี่ย	**92.951**	**70.318**	**71.787**	**45.429**

ตารางที่ 3.5 เวลาในการตรวจจับสีผิวในภาพสี

วิธีการ	วิธีที่นำเสนอ	RGB	YCbCr	HSV
เวลา(ms)	69.987	64.454	63.896	64.246

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.6 สรุปผลการทดลอง

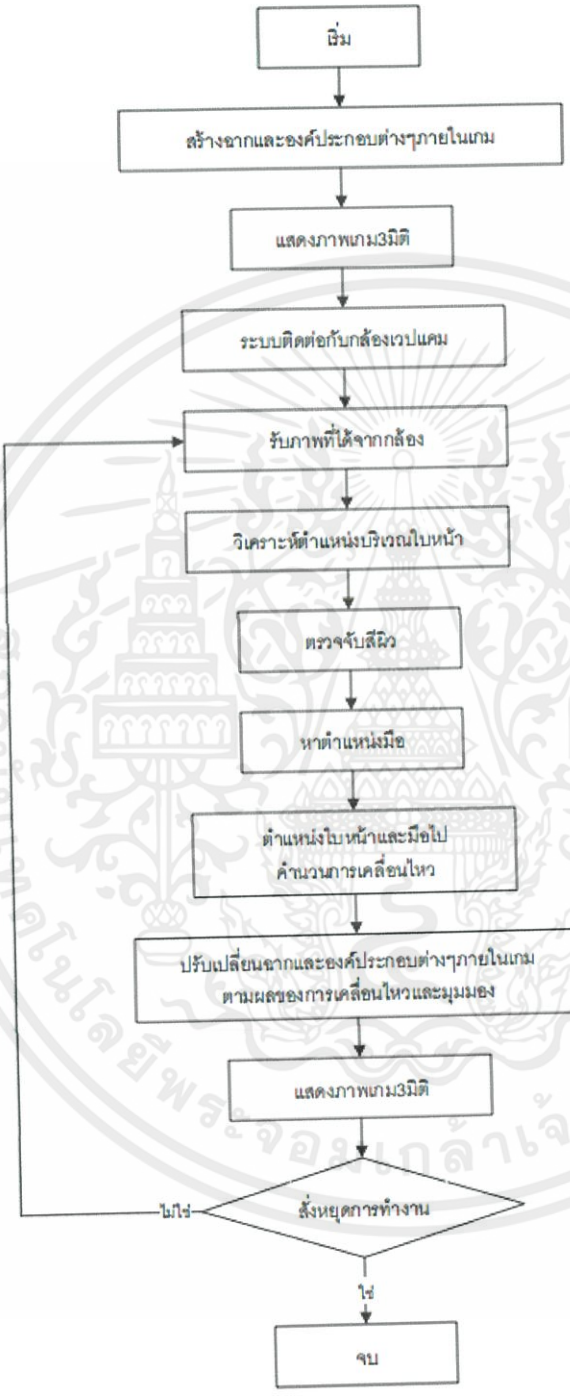
จากวิธีที่นำเสนอการกำหนดขอบเขตสีผิวจากใบหน้า ผลที่ได้เมื่อเปรียบเทียบกับงานวิจัยการใช้โมเดลสี RGB[7] YCbCr[8] และ HSV[9] ในสภาพแวดล้อมและแสงปกตินั้น ค่า DSR ที่ได้มีค่าสูงกว่างานวิจัยอื่นๆ และจากผลการทดลองดังแสดงในตารางกรณีที่ 2 ซึ่งเป็นกรณีที่สภาพแวดล้อมมีแสงน้อย จากการตรวจจับสีผิวด้วยวิธีที่นำเสนอ ค่าความถูกต้องของการตรวจจับมีค่าสูงกว่าวิธีอื่นอย่างเห็นได้ชัด ส่วนการตรวจจับสีผิวในสภาพแวดล้อมที่มีแสงปกติ แต่มีวัตถุรบกวนอยู่ภายในภาพ(กรณีที่ 3.1 และ 3.2)ผลลัพธ์ที่ได้มีค่าใกล้เคียงกับวิธีอื่น แต่ในสภาพแวดล้อมแสงผิดปกติ ในกรณีแสงสีน้ำเงินและแสงสีเขียวสะท้อนลงบนใบหน้า(กรณีที่ 5 และ 6) ค่า DSR ที่ได้จะสูงกว่าวิธีการอื่นๆอย่างเห็นได้ชัดเช่นกัน เพราะวิธีที่นำเสนอได้นำสีใบหน้า ณ ขณะนั้นที่เปลี่ยนแปลงไปตามสภาวะแสงดังกล่าวไปวิเคราะห์ จึงทำให้ค่ากำหนดขอบเขตของสีผิวเปลี่ยนตาม จากการทดลองกับข้อมูลทั้ง 9 ภาพ ผลลัพธ์ที่ได้มีค่าเฉลี่ยความถูกต้องของการตรวจจับพิกเซลสีผิว (DSR) สูงถึง 92.951% ส่วนเวลาในการตรวจจับสีผิวของวิธีการกำหนดขอบเขตสีผิวทั้งหมดจากใบหน้านั้น จะช้ากว่าประมาณ 8.721% ดังแสดงในตาราง 3.5 เนื่องจากมีการกำหนดขอบเขตของสีผิวใหม่ในทุกเฟรม ดังนั้นจึงทำให้การประมวลผลช้ากว่าวิธีการกำหนดขอบเขตสีผิวด้วยค่าคงที่

3.7 การทำงานของโปรแกรม

เมื่อเกมเริ่มการทำงาน ตัวเกมจะสร้างฉากและองค์ประกอบต่างๆภายในเกมเป็น 3 มิติตามที่ได้ออกแบบสภาพแวดล้อม แสง สี เสียงต่างๆไว้ จากนั้นตัวเกมจะทำงานตามชุดคำสั่งที่ได้ใส่ลงไป คือการติดต่อกับกล้องเว็บแคมเพื่อที่จะรับอินพุตจากกล้องต่อมา ส่วนของการประมวลผลภาพนำภาพมาวิเคราะห์การตรวจจับใบหน้าด้วยวิธี Haar-like Feature และทำการหาตำแหน่งจุดกึ่งกลางบริเวณใบหน้า เริ่มต้นบริเวณมือของผู้เล่นด้วยวิธีการตรวจจับสีผิว เพื่อหาตำแหน่งของมือจากนั้นเอาตำแหน่งที่ได้ของบริเวณใบหน้าและบริเวณมือไปคำนวณการเคลื่อนไหว ภายในเกมจะมีการเคลื่อนไหวตามผลของการคำนวณ และผลกระทบต่างๆที่เกิดจากเคลื่อนไหวไม่ว่าจะเป็นวัตถุต่างๆในเกม แสง เงา มุมมองของผู้เล่นจะเปลี่ยนไป ตัวเกมจะคำนวณการเคลื่อนไหวและดำเนินต่อไปเรื่อยๆ จนทำภารกิจภายในเกมเสร็จสิ้น หรือล้มเหลว โดยภารกิจในเกมที่ผู้เล่นได้รับมอบหมายนั้นจะเริ่มจากภารกิจระดับง่ายไปจนถึงระดับยาก ซึ่งจะเป็นการพัฒนาความสามารถของผู้เล่นในการเคลื่อนให้ดีขึ้นตามลำดับ และจะเกิดความท้าทายกับภารกิจต่อไปที่ยากขึ้นจึงทำให้การ

เอกสารนี้เป็น... ไม่ว่ากรณี...

ความสนุกในการเล่นเกม เมื่อเกมดำเนินมาจนถึงภารกิจสุดท้ายหรือภารกิจล้มเหลว ตัวเกมจะหยุดลงเพื่อให้ผู้เล่นได้เลือกดำเนินการต่อไปว่าจะกลับไปเล่นใหม่หรือออกจากเกม



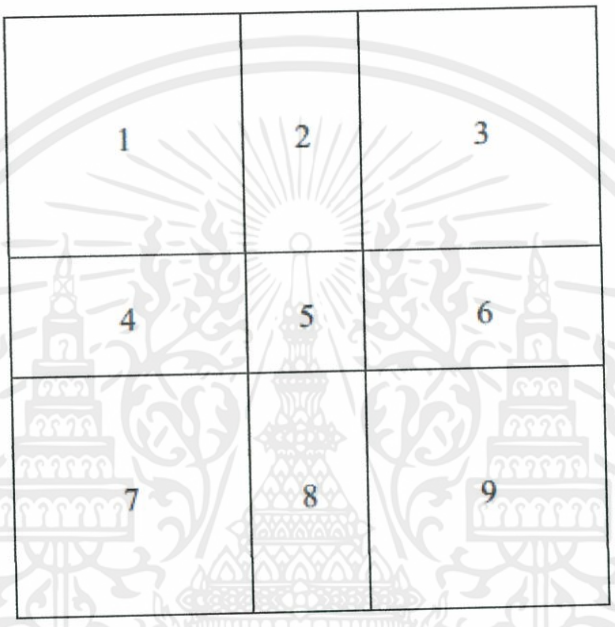
รูปที่ 3.9 แผนภาพโดยรวมของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.8 ขั้นตอนการประมวลผลท่าทาง

ในขั้นการประมวลผลท่าทางนั้น เริ่มจากการค้นหาใบหน้าของผู้เล่นแล้วจึงแบ่งภาพออกเป็นสองส่วนเพื่อค้นหาตำแหน่งมือ ซึ่งภาพฝั่งซ้ายเป็นการค้นหาตำแหน่งมือซ้ายของผู้เล่นส่วนภาพฝั่งขวาเป็นการค้นหาตำแหน่งมือขวาของผู้เล่น โดยจะนำตำแหน่งใบหน้าและมือที่ได้มาเทียบกับพิกัดของหน้าจอเพื่อกำหนดท่าทางของตัวละครในเกม

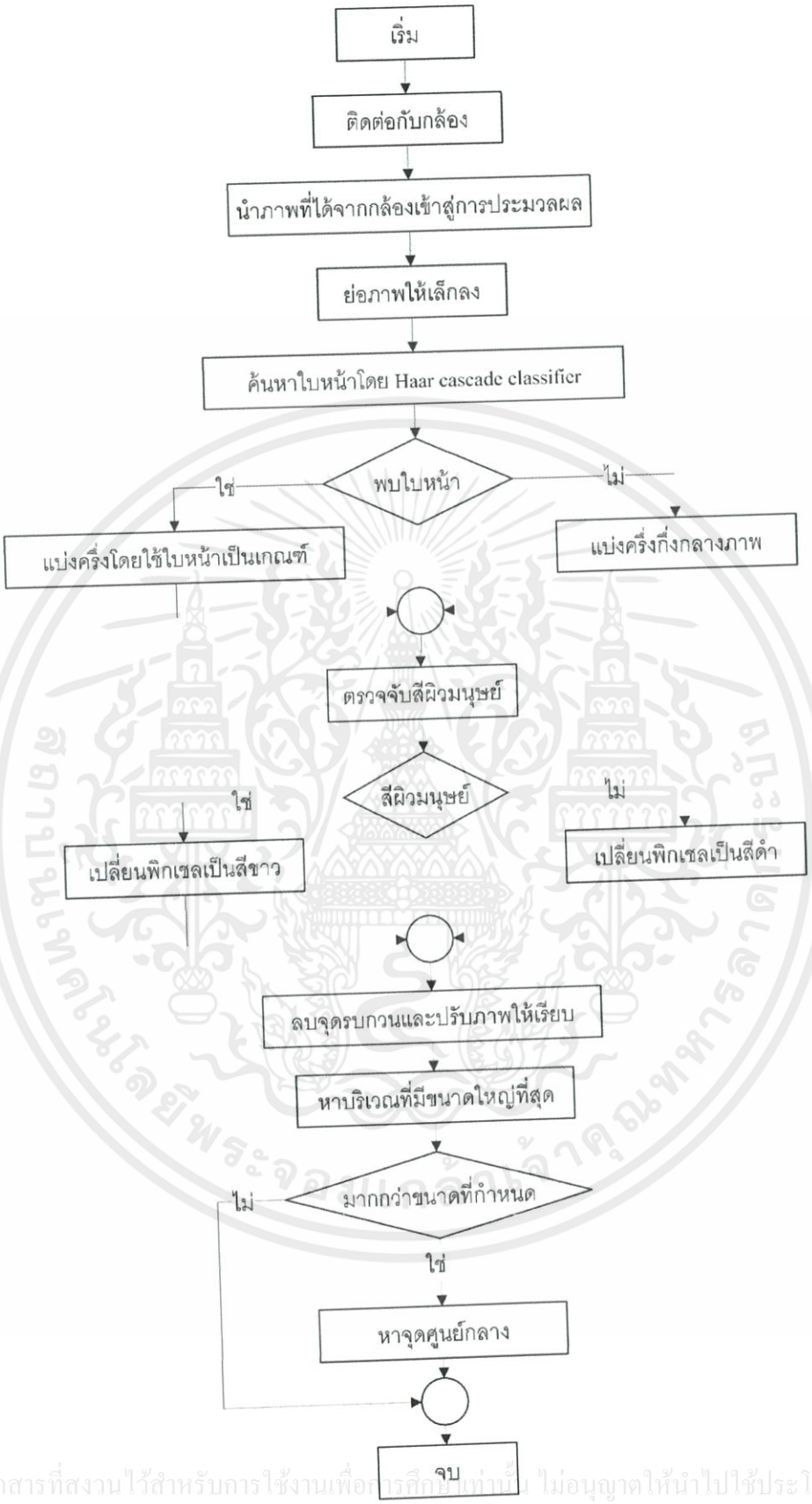
ตารางที่ 3.6 พิกัดของหน้าจอ



โดยจะแบ่งพิกัดดังภาพข้างต้น มีเงื่อนไขดังนี้

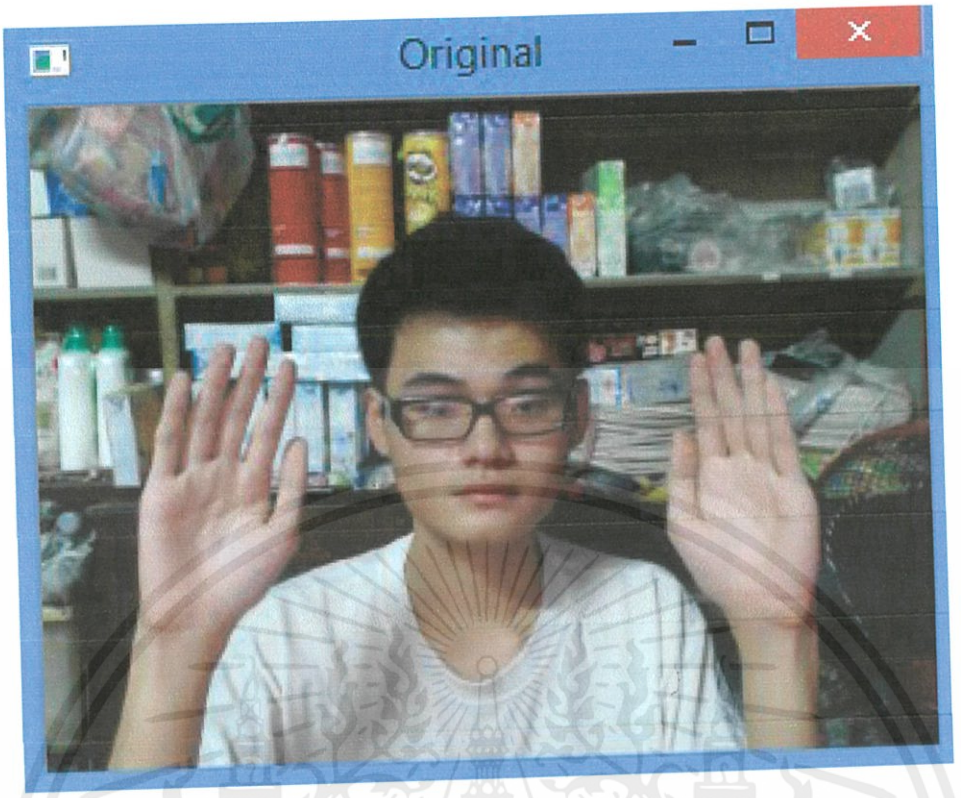
- ถ้าตรวจพบส่วนของศีรษะในส่วนที่ 5 จะหมายถึง การยืนอยู่กับที่
- ถ้าตรวจพบส่วนของศีรษะในส่วนที่ 3,6 และ 9 จะหมายถึง การเลี้ยวขวา
- ถ้าตรวจพบส่วนของศีรษะในส่วนที่ 1,4 และ 7 จะหมายถึง การเลี้ยวซ้าย
- ถ้าตรวจพบส่วนของศีรษะในส่วนที่ 1,2 และ 3 จะหมายถึง การเดินหน้า
- ถ้าตรวจพบส่วนของศีรษะในส่วนที่ 7,8 และ 9 จะหมายถึง การถอยหลัง
- ถ้าตรวจพบส่วนของมือในส่วนที่ 1,4 และ 7 จะหมายถึง การต่อยซ้าย
- ถ้าตรวจพบส่วนของมือในส่วนที่ 3,6 และ 9 จะหมายถึง การต่อยขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

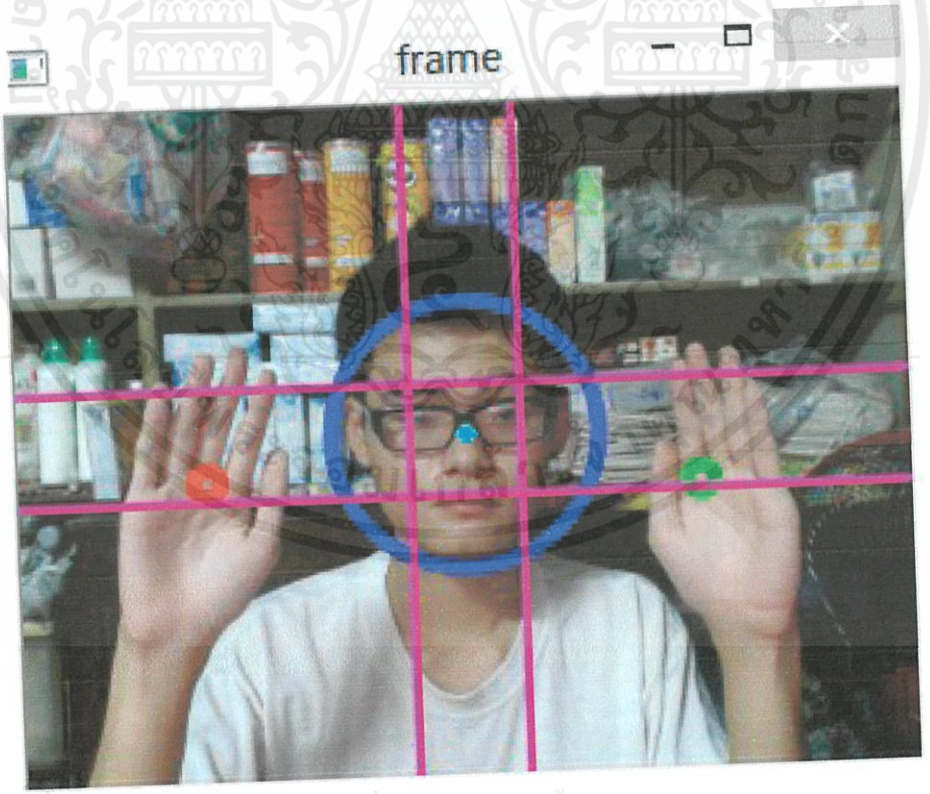


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.10 แผนภาพการตรวจจับสีผิว



รูปที่ 3.11 ภาพต้นแบบ



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 3.12 ภาพการแบ่งพิกัดของหน้าจอ

3.9 ส่วนประกอบต่างๆของโปรแกรม

- void init()

เป็นการกำหนดค่าเริ่มต้นของโปรแกรม ได้แก่ การกำหนด Haar like feature เพื่อใช้ในการตรวจจับใบหน้า การตั้งค่ากล้องเว็บแคม การกำหนดความกว้างและความสูงของภาพ

- void update()

นำรูปภาพที่ได้รับจากกล้องเว็บแคม นำภาพที่ได้มาแสดงผล และนำรูปไปประมวลผลในขั้นตอนถัดไป

- IplImage face(IplImage img)

ประมวลผลภาพแต่ละภาพที่ได้รับจากกล้อง โดยนำภาพมาย่อขนาด แปลงภาพเป็นภาพสีเทา แล้วนำไปปรับปรุงรูปภาพโดยวิธีการ Equalization จากนั้นจะทำการค้นหาใบหน้าด้วยวิธี Haar-like Feature ซึ่งผลลัพธ์ของการค้นหาที่ได้คือกรอบสี่เหลี่ยมที่ล้อมรอบบริเวณใบหน้า นำผลลัพธ์ที่ได้มาตัดขอบ เพื่อตัดส่วนเกินที่อาจจะไม่ใช่ใบหน้าออก นำภาพที่ได้มาปรับให้เรียบขึ้น โดยวิธี Gaussian หลังจากนั้นจึงแปลงภาพให้อยู่ในโมเดลสี HSV แล้วนำตำแหน่งของใบหน้ามาควบคุมการเคลื่อนไหวของตัวละครภายในเกม และแบ่งภาพออกเป็นสองส่วน เพื่อหาตำแหน่งมือในการประมวลผลขั้นตอนถัดไป

- CvPoint hand(IplImage imgHand,IplImage imgOrignal,CvRect hand)

นำภาพที่แบ่งออกเป็น 2 ส่วนมาทำการตรวจหาสีผิว จากนั้นปรับภาพให้เรียบโดยวิธีการโมฟอร์โลยี เสร็จแล้วจึงนำมาผ่านกระบวนการเพื่อหาช่วงข้อมูลที่มีจำนวนมากที่สุดและต้องมีขนาดใหญ่กว่าค่าที่กำหนดไว้ ซึ่งเป็นข้อกำหนดที่ทำไว้เพื่อป้องกันข้อผิดพลาดที่ภาพนั้นอาจจะไม่ใช่มือ จากนั้นนำมากำหนดขอบเขตแล้วหาจุดศูนย์กลางของมือ เพื่อนำตำแหน่งมาควบคุมการเคลื่อนไหวของตัวละครภายในเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- CvRect cal(IplImage src)

เป็นการนำภาพใบหน้ามาเก็บข้อมูลความถี่สะสมสีของช่องสี H และ S แล้วจึงนำข้อมูลเหล่านั้นมาพลอตกราฟ โดยรูปภาพของกราฟที่ได้จะนำไปผ่านกระบวนการปรับปรุงข้อมูลด้วยวิธีไดเลชันและอีโรชัน

- CvSeq<CvPoint> FindContours(IplImage img, CvMemStorage storage)

ค้นหาช่วงของข้อมูลที่มีจำนวนข้อมูลรวมตัวกันเป็นกลุ่มใหญ่ที่สุด หลังจากนั้นนำมาตีกรอบสี่เหลี่ยม เพื่อกำหนดเป็นขอบเขตของสีผิวมนุษย์

- IplImage skin(IplImage src, CvRect range)

เอาช่วงข้อมูลที่ได้จากการประมวลผล มากำหนดค่าสูงสุดและต่ำสุดของช่องสี H และ S จากนั้นนำภาพที่ได้รับจากกล้องเว็บแคม โดยแปลงเป็นภาพจากสีปกติเป็นโมเดลสี HSV แล้วนำมาเปรียบเทียบกับช่วงข้อมูลสีที่กำหนด ซึ่งถ้าหากพิกเซลของภาพจากกล้องมีค่าอยู่ในช่วง แสดงว่าพิกเซลที่พิจารณานั้นเป็นสีผิวมนุษย์ และพิกเซลนั้นจะแสดงผลเป็นสีขาว แต่ถ้าหากไม่อยู่ในช่วงของข้อมูลที่กำหนด แสดงว่าพิกเซลนั้นไม่ใช่สีผิวมนุษย์ และจะแสดงผลของพิกเซลเป็นสีดำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4
ขั้นตอนการทำงานของโปรแกรม

อุปกรณ์ที่ใช้ในการพัฒนาโปรแกรมเป็นปัจจัยสำคัญที่ทำให้ได้โปรแกรมที่มีประสิทธิภาพ ประสิทธิผล และต้องสามารถรองรับการทำงานของโปรแกรมในทุกๆสภาพการณ์ในการทดลองและการปฏิบัติงานจริง

4.1 ฮาร์ดแวร์และซอฟต์แวร์สำหรับพัฒนาโปรแกรม

- CPU Core i5
- Ram8 Gb
- ระบบปฏิบัติการ Windows 7 64 bit
- Web camera
- Microsoft Visual Studio 2012
- OpenCVSharp Library
- Unity 4.3.1

4.2 อุปกรณ์ที่ใช้พัฒนาโปรแกรม

กล้องที่ใช้ติดต่อกับคอมพิวเตอร์ (Web camera) 1 ตัว

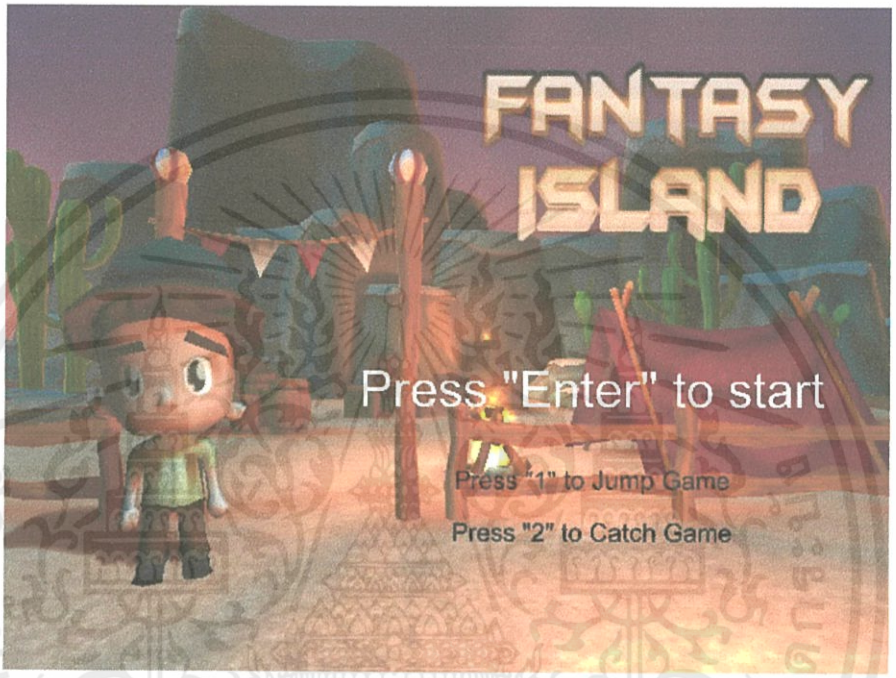
รูปที่ 4.1 ภาพตัวอย่างกล้องที่ใช้ติดต่อกับคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.3 ขั้นตอนการทำงานของแอพพลิเคชันเมื่อรวมส่วนของการประมวลผลภาพและส่วนของเกมเข้าด้วยกัน

4.3.1 เมื่อเริ่มการทำงานของเกมแอพพลิเคชัน

เมื่อเปิดโปรแกรมขึ้นมาโปรแกรมจะแสดงหน้าแรกของเกมเพื่อให้ผู้เล่นทำการเริ่มเล่นเกม

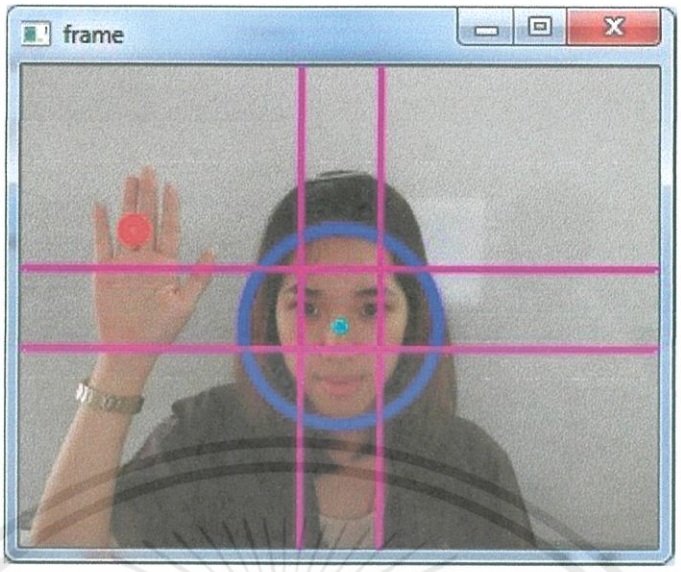


รูปที่ 4.2 ภาพเปิดของเกมแอพพลิเคชัน

4.3.2 การค้นหา และตรวจจับใบหน้าเพื่อใช้เป็นค่าเริ่มต้น

เมื่อผู้เล่นทำการเริ่มเกม โปรแกรมจะทำการติดต่อกับกล้องเวปแคม และเปิดกล้องเพื่อทำการค้นหาใบหน้า และทำการตรวจจับด้วยวิธีของ Haar like feature เมื่อมีใบหน้าของผู้เล่นเข้ามาในฉาก ส่วนของกระบวนการทำงานที่เกี่ยวข้องกับการประมวลผลภาพจะทำการค้นหา และตรวจจับใบหน้าเริ่มต้นของผู้เล่น พร้อมกับแสดงจุดกึ่งกลางบริเวณใบหน้าและมือของผู้เล่น ดังรูปที่ 4.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 ภาพการตรวจจับติดตามใบหน้า และมือของผู้เล่น

การตรวจจับติดตามใบหน้า และมือของผู้เล่นจะนำไปใช้ในการวิเคราะห์ท่าทางที่กำหนดไว้ และเมื่อกำหนดใบหน้า และมือเริ่มต้นได้เรียบร้อยแล้วก็จะเข้าสู่การทำงานของเกม

4.3.3 ขั้นตอนการตอบโต้และควบคุมการเคลื่อนไหวภายในเกม

แอพพลิเคชันจะวนลูปรอรับคำสั่งจากการกำหนดท่าทางอยู่ตลอดเวลา

เอกสารนี้เป็นเอกสาร ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น ไปใช้

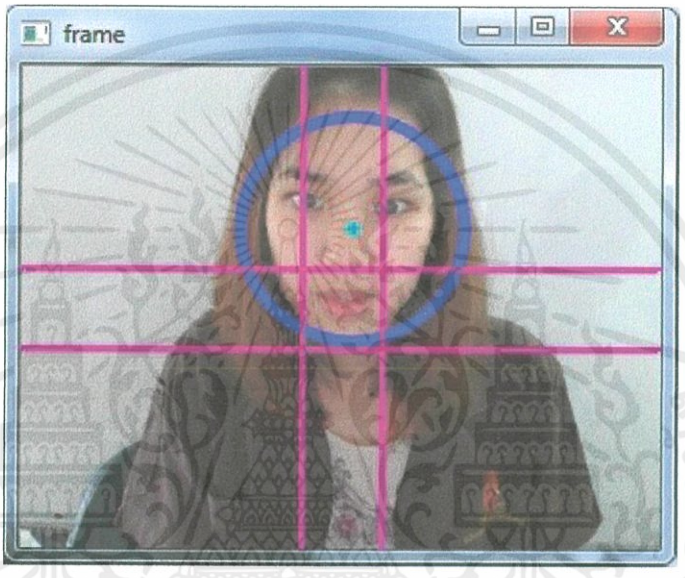
รูปที่ 4.4 ภาพแสดงเมื่อเริ่มเข้าสู่การทำงานของเกม

4.3.3.1 ท่าทางต่างๆที่ใช้ภายในเกมและการทำงานของเกมเมื่อได้รับคำสั่ง

ในการควบคุมท่าทางต่างๆภายในเกมจะแบ่งออกเป็น 3 แบบ โดยจะแบ่งตามประเภทภายในเกม ได้แก่เกมกระโดด เกมเก็บของ และเกมเขาวงกต

เกมกระโดด

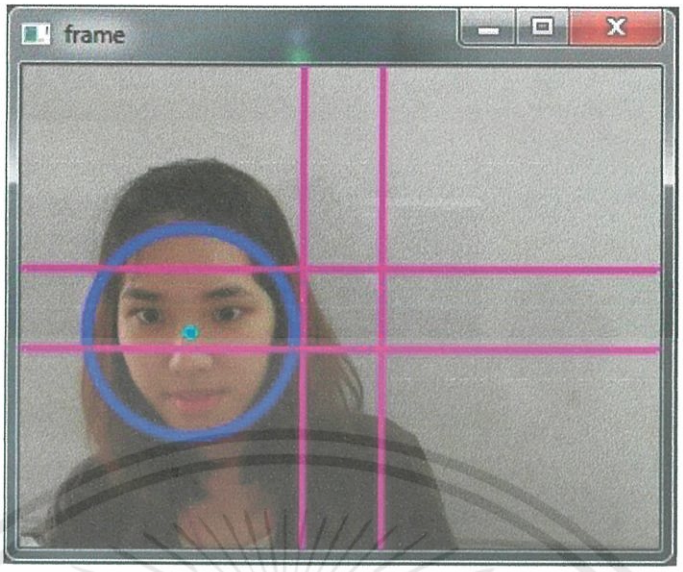
เกมกระโดดเป็นเกมด่านแรก เหมาะสำหรับฝึกการควมคุมตัวละครด้วยศีรษะผู้ของเล่นเป็นการควบคุมการกระโดด และบังคับตัวละครภายในเกมให้ไปทางด้านซ้ายหรือขวา



รูปที่ 4.5 ท่าทางบังคับให้ตัวละครกระโดดขึ้นในแนวตรง

รูปที่ 4.6 แสดงภาพตัวละครกระโดดในแนวตรง

เอกสารนี้เป็นเอกสาร ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไปใช้

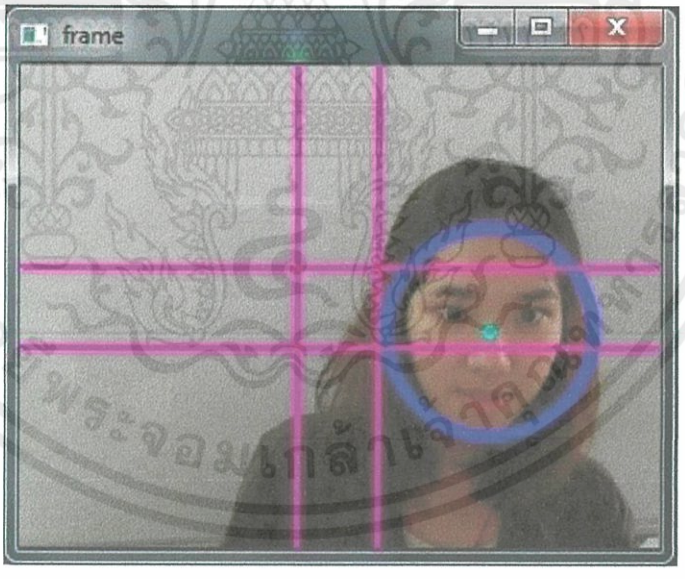


รูปที่ 4.7 ท่าทางบังคับให้ตัวละครกระโดดทางซ้ายหรือเดินไปทางซ้าย

รูปที่ 4.8 แสดงภาพตัวละครกระโดดทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.9 แสดงภาพตัวละครเดินไปทางซ้าย



รูปที่ 4.10 ท่าทางบังคับให้ตัวละครกระโดดทางขวาหรือเดินไปทางขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.11 แสดงภาพตัวละครกระโดดทางขวา

รูปที่ 4.12 แสดงภาพตัวละครเดินไปทางขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

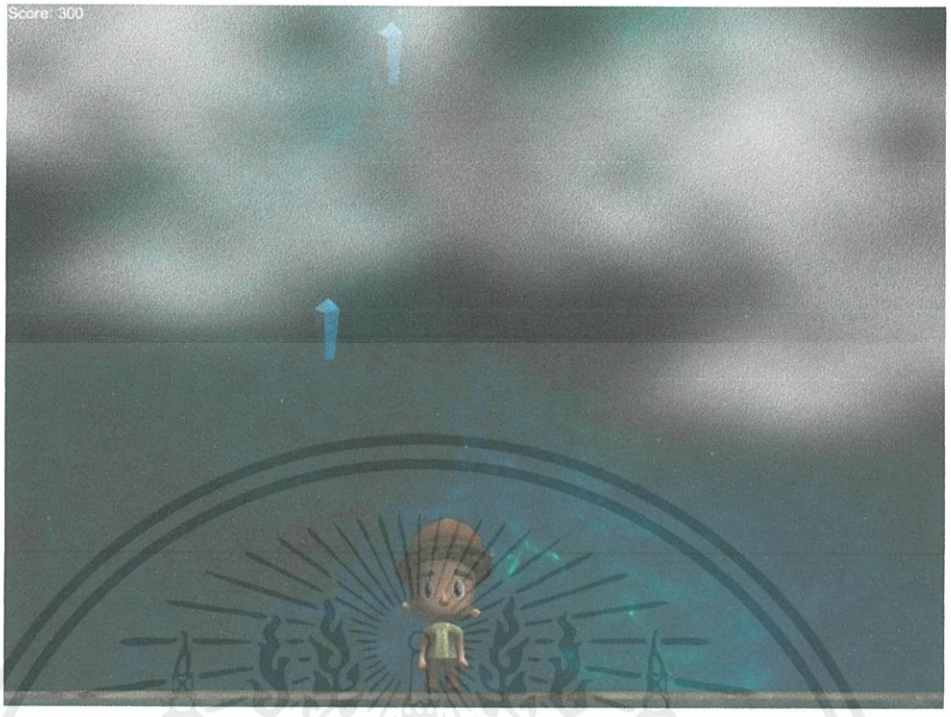
รูปที่ 4.13 แสดงภาพเมื่อจบเกม

โดยคะแนนจะเก็บตามระยะดับความสูงที่ผู้เล่นขึ้นได้สูงที่สุด และถ้าหากผู้เล่นขึ้นสูงถึงระยะทางที่กำหนดภายในเกม ผู้เล่นจะเข้าสู่ด่านถัดไปซึ่งเป็นด่านเก็บของ แต่ถ้าหากผู้เล่นกระโดดพลาด และหลุดจากตำแหน่งที่กำหนดเป็นขอบเขตไว้ภายในเกม จะเป็นการจบเกม

เกมเก็บของ

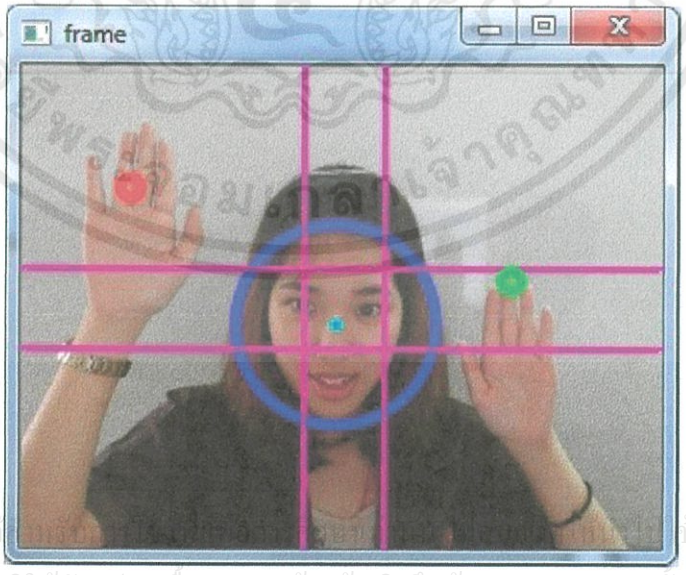
ภายในเกมจะเป็นการฝึกใช้มือในการควบคุมตัวละคร ร่วมกับการใช้ศีรษะของผู้เล่น โดยจะมีการใช้มือเพื่อกำจัดสิ่งที่เป็นอุปสรรคภายในเกม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



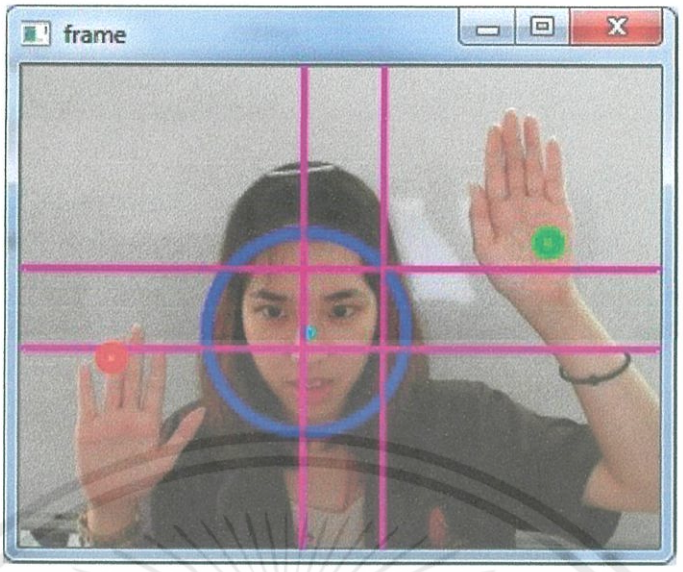
รูปที่ 4.14 แสดงภาพตัวละครยืนนิ่ง

ภายในด่านเก็บของนี้ หากผู้เล่นสามารถเก็บคริสตัลสีฟ้าได้จะมีค่าเท่ากับ 20 คะแนน และอุปสรรคในด่านนี้มี 3 อย่างด้วยกันคือ 1.หินอุกาบาต ถ้าหากผู้เล่นเก็บหิน จะปรากฏหมอกบังขึ้นบนหน้าจอให้ผู้เล่นกำจัดออกไป 2.หม้อเวทมนต์ ถ้าหากผู้เล่นเก็บหม้อได้ คะแนนจะถูกลบ 30 คะแนน และอย่างสุดท้ายคือคริสตัลสีแดง หากผู้เล่นเก็บคริสตัลดังกล่าวได้ เกมจะจบลงทันที



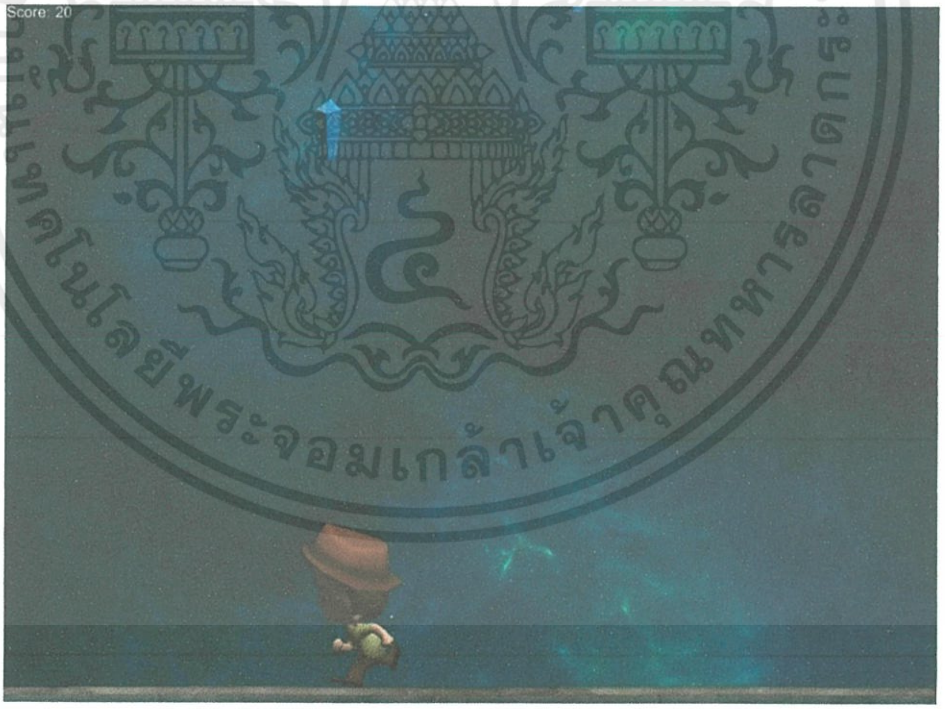
เอกสารนี้เป็นเอกสารที่สงวนไว้ ... ใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.15 ท่าทางในการกำจัดหมอกท่าทาง 1



รูปที่ 4.16 ท่าทางในการกำจัดหมอกท่าทาง 2

ซึ่งในการกำจัดหมอกนั้น ผู้เล่นจะต้องทำท่าทางในการกำจัดหมอกตามรูปที่ 4.15 และ 4.16 ควบคู่กัน จึงจะสามารถกำจัดหมอกที่บังหน้าจอออกไปได้



รูปที่ 4.17 แสดงภาพตัวละครเดินไปทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.18 แสดงภาพตัวละครเดินไปทางขวา

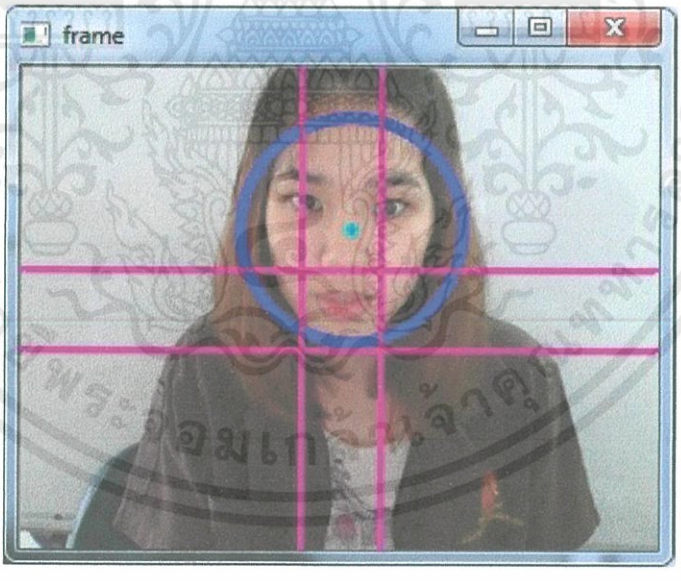
เมื่อผู้เล่นเก็บคะแนนได้ครบ 300 คะแนน เกมจะเปลี่ยนเป็นด่านถัดไปคือด่านเขาวงกต ซึ่งเป็นด่านสุดท้ายของเกม

เกมเขาวงกต

เกมนี้จะเป็นการใช้ศีรษะและมือในการควบคุมการเคลื่อนไหวภายในเกม ซึ่งการควบคุมตัวละครในด่านนี้จะยากกว่าด่านอื่นๆภายในเกม โดยการใช้มือในด่านนี้จะเป็นการควบคุมให้ตัวละครต่อยทางซ้ายและขวา เพื่อกำจัดสัตว์ประหลาดที่เป็นอุปสรรคในเกมเขาวงกต

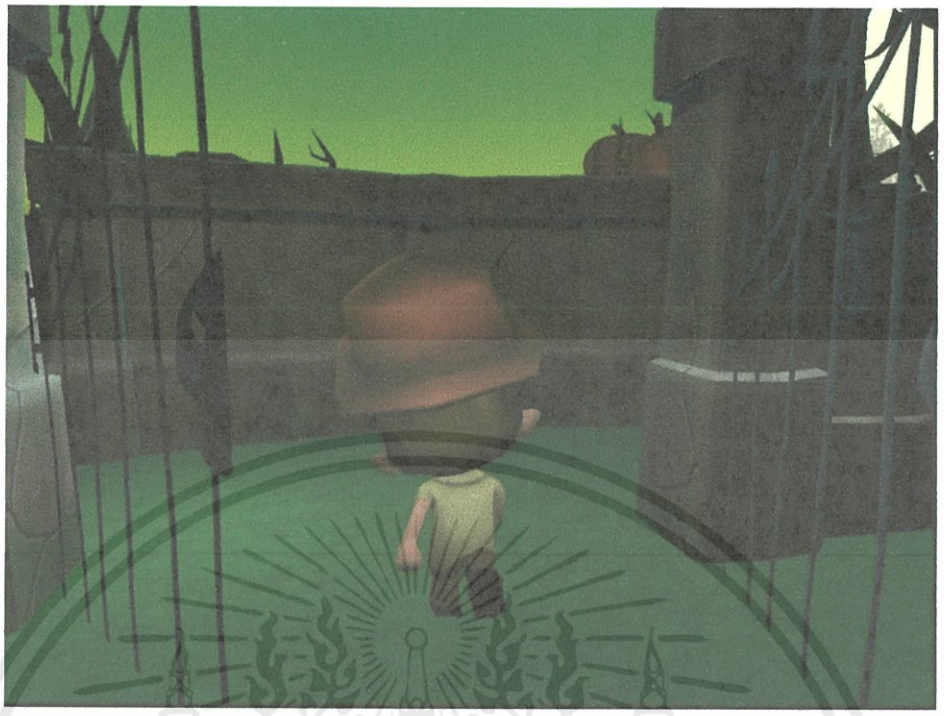
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.19 แสดงภาพตัวละครยืนนิ่ง

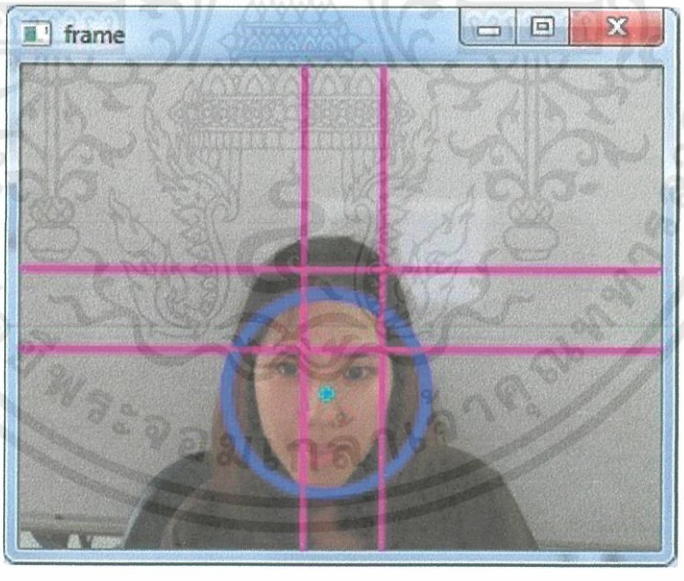


รูปที่ 4.20 ท่าทางบังคับให้ตัวละครเดินตรง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



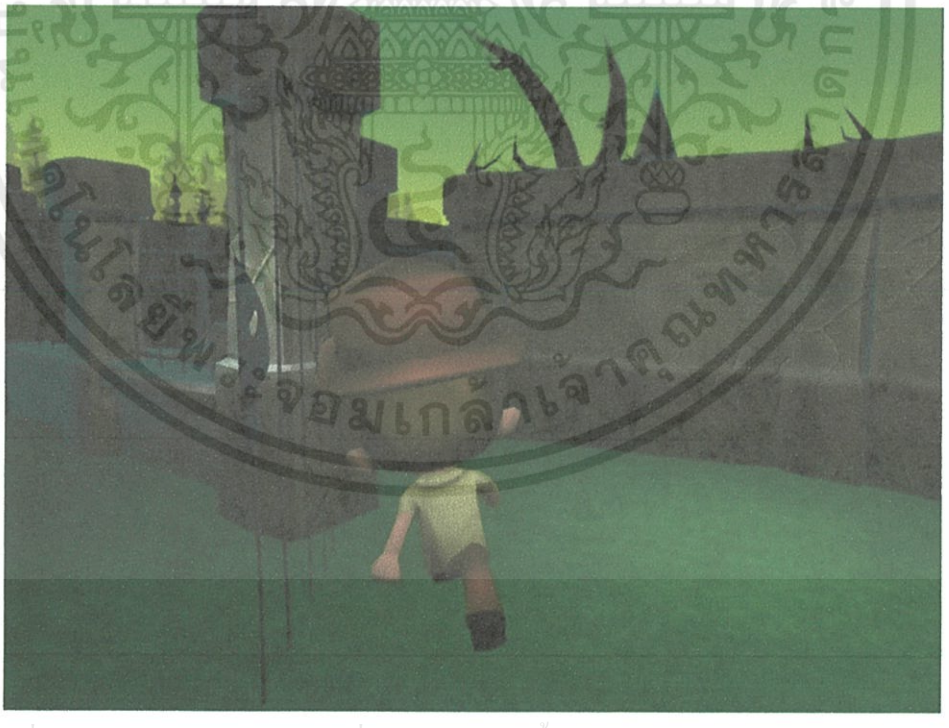
รูปที่ 4.21 แสดงภาพตัวละครเดินตรง



รูปที่ 4.22 ท่าทางบังคับให้ตัวละครเดินถอยหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

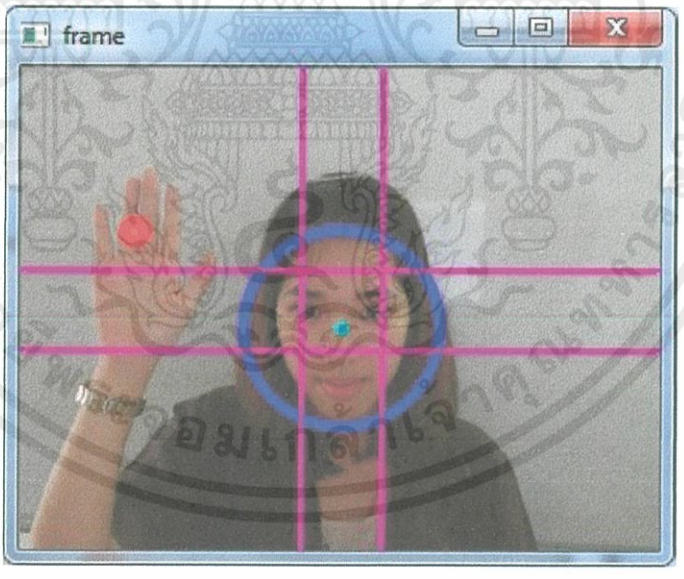
รูปที่ 4.23 แสดงภาพตัวละครเดินถอยหลัง



รูปที่ 4.24 แสดงภาพตัวละครเดินไปทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

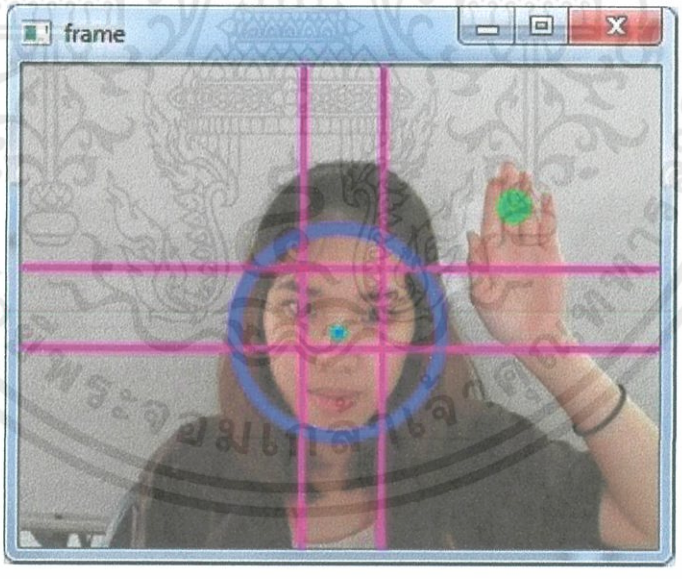
รูปที่ 4.25 แสดงภาพตัวละครเดินไปทางขวา



รูปที่ 4.26 ท่าทางบังคับให้ตัวละครต่อยด้านซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.27 แสดงภาพตัวละครต่อยด้านซ้าย



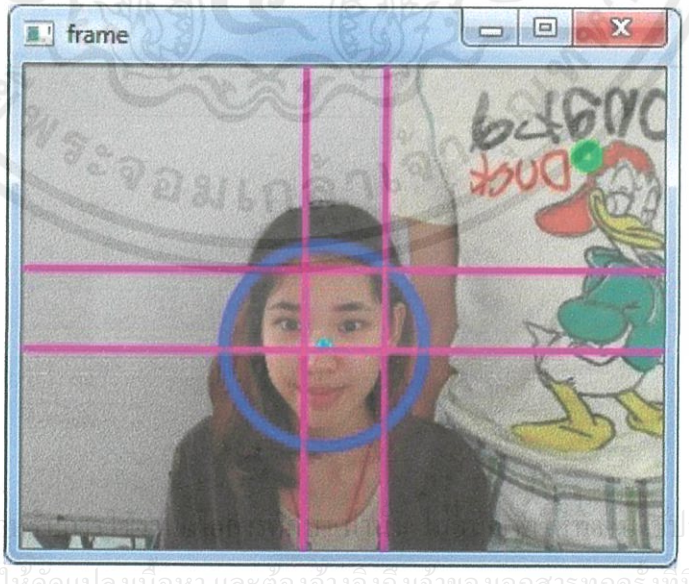
รูปที่ 4.28 ท่าทางบังคับให้ตัวละครต่อยด้านขวา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.29 แสดงภาพตัวละครต่อยด้านขวา

ในด่านเขาวงกตซึ่งเป็นด่านสุดท้ายนี้ ผู้เล่นต้องต่อสู้กับสัตว์ประหลาดและหาทางออกจากเขาวงกตให้ได้จึงจะเป็นการจบเกม

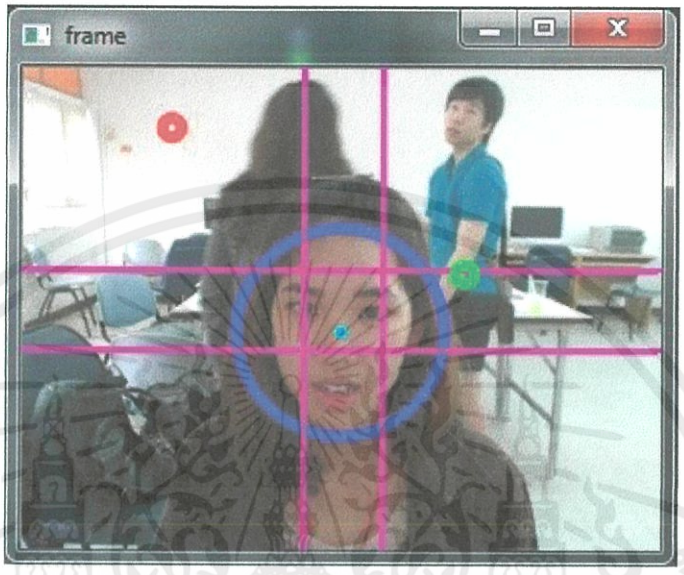
4.3.4 ข้อจำกัดในการใช้โปรแกรมในสภาพแวดล้อมต่างๆกัน



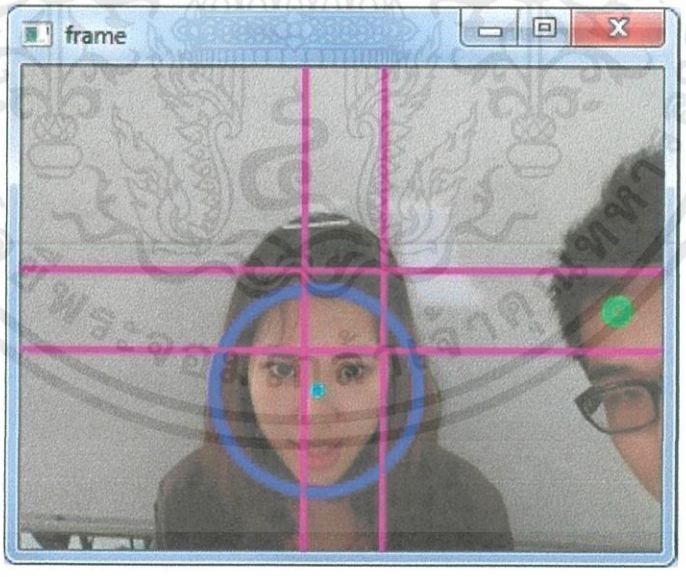
เอกสารนี้เป็นเอกสารที่สงวนไว้สำ ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.30 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีสีวัตถุใกล้เคียงกับสีผิว

รูปที่ 4.30 แสดงภาพกรณีที่มีวัตถุภายในฉากที่มีสีใกล้เคียงกับสีผิว โปรแกรมจะไม่สามารถแยกได้ว่าสีใดเป็นสีผิว สีใดเป็นสีของวัตถุเนื่องจากโปรแกรมมีการกำหนดช่วงของสีผิวไว้



รูปที่ 4.31 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีคนภายในฉากเกิน 1 คนกรณี 1

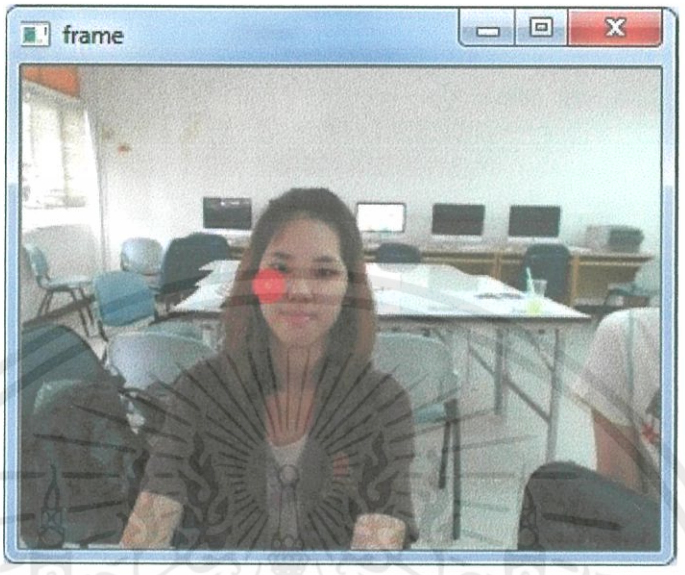


รูปที่ 4.32 แสดงภาพการตรวจจับในสภาพแวดล้อมที่มีคนภายในฉากเกิน 1 คนกรณี 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 4.31 และรูปที่ 4.32 แสดงภาพกรณีที่มีปริมาณของคนภายในฉากมากกว่า 1 คน จะทำให้โปรแกรมเกิดข้อผิดพลาดได้ เนื่องจากมีสีผิวภายในฉากเกินข้อกำหนด ซึ่งถ้าหากค้นเจอ

ตำแหน่งใบหน้าแล้ว ในส่วนอื่นของภาพที่เป็นสีผิวมนุษย์ หากมีจำนวนสีผิวเพียงพอ โปรแกรมจะประมวลผลว่าเป็นมือ จุดสีเขียวจึงย้ายตำแหน่งดังภาพ



รูปที่ 4.33 แสดงภาพการตรวจจับในระยะห่าง

รูปที่ 4.33 แสดงภาพการตรวจจับใบหน้า ในกรณีที่ใบหน้าอยู่ห่างจากกล้องเกินระยะการตรวจจับที่กำหนดไว้ภายในโปรแกรม ทำให้ปริมาณสีผิวที่นำมาพิจารณานั้นมีไม่เพียงพอ โปรแกรมจึงไม่สามารถตรวจจับใบหน้าได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5
ผลสรุปและข้อเสนอแนะ

5.1 สรุปผล

โปรแกรมสามารถตรวจจับการเคลื่อนไหวเพื่อควบคุมตัวละครภายในเกมได้ดี ภายใต้สภาพแวดล้อมที่มีการควบคุมเรื่องสีของวัตถุ จำนวนคน และแสงภายในฉาก เพื่อความสะดวกในการประมวลผลภาพ

ขั้นตอนการตรวจจับสีผิว ภายในปัญหาพิเศษนี้ได้ใช้วิธีการตรวจจับสีผิวทั้งหมดจากสีผิวใบหน้า เป็นการนำรูปภาพใบหน้าที่ได้จากการประมวล มาวิเคราะห์เพื่อกำหนดขอบเขตสีผิว ซึ่งค่าสีผิวที่ได้นั้นเป็นค่าสีผิวจากตัวผู้ใช้เอง จึงทำให้มีการปรับเปลี่ยนขอบเขตได้อย่างเหมาะสม และจากผลการทดลองพบว่า ประสิทธิภาพในการตรวจจับสีผิวมีความแม่นยำมากขึ้นเมื่อเปรียบเทียบกับวิธีการตรวจจับโดยใช้โมเดลสีอื่นๆ โดยวัดค่าความถูกต้องในการตรวจจับสีผิว (DSR) ได้ 92.951% ซึ่งสามารถตรวจจับได้ดีในสภาพแวดล้อมที่มีแสงน้อย แสงปกติและสภาพแวดล้อมที่มีแสงรบกวนสีน้ำเงิน แต่เวลาในการประมวลผลของวิธีที่นำเสนอนี้จะช้ากว่าวิธีการกำหนดขอบเขตสีผิวด้วยค่าคงที่เป็น 8.271% เนื่องจากวิธีการดังกล่าวมีการกำหนดขอบเขตสีผิวใหม่ในทุกเฟรม

ส่วนข้อจำกัดของโปรแกรมยังไม่สามารถทำงานได้ครอบคลุมทุกสภาพแวดล้อม เช่น กรณีที่มีแสงสว่างมากเกินไปจะไม่สามารถตรวจจับใบหน้าและสีผิวได้ จึงจำเป็นต้องมีการควบคุมสภาพแวดล้อมให้อยู่ภายในบริเวณที่ไม่สว่างและมืดจนเกินไป กรณีที่มีวัตถุภายในฉากมีสีคล้ายกับสีผิวจะทำให้โปรแกรมตรวจจับและประมวลผลผิดพลาด การวิเคราะห์ท่าทางยังไม่มีความแม่นยำ 100 เปอร์เซ็นต์ เช่น ในกรณีที่ตำแหน่งของผู้เล่นอยู่ระยะห่างจากจุดกำหนด จะทำให้โปรแกรมประมวลผลผิดพลาดหรือหลุดการตรวจจับได้

ในอนาคตคาดว่าจะสามารถพัฒนาเพื่อแก้ไขปัญหาดังกล่าวได้ นอกจากนี้ยังสามารถนำโปรแกรมไปพัฒนา และประยุกต์ใช้ในงานด้านต่างๆ ได้นอกจากด้านความบันเทิง ตลอดจนการพัฒนาสำหรับเป็นสื่อการเรียนการสอนได้อีกด้วย

5.1.1 ปัญหาต่างๆ ที่พบระหว่างการพัฒนาโปรแกรม

1. การพัฒนาโปรแกรมมีการนำไลบรารี่ของโอเพ่นซีวีชาร์พ (OpenCVSharp Library) มาใช้พัฒนา ซึ่งเป็นไลบรารี่ฟรีแวร์ (Freeware) สามารถโหลดมาใช้งานได้ฟรี สำหรับจัดการเรื่องการประมวลผลภาพ ทำให้ยังมีปัญหาและจุดบกพร่องต่างๆอยู่เล็กน้อย ตลอดจนข้อมูลทางด้านการประมวลผลภาพค่อนข้างเป็นเรื่องใหม่ ทำให้ยากในการค้นคว้าข้อมูลและการทดลองยังไม่กว้างขวางมากนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เนื่องจากการทำงานของโปรแกรมเป็นการทำงานแบบเรียลไทม์ (Real time) ทำให้กล้องที่ใช้จำเป็นต้องมีอัตราการเปลี่ยนภาพค่อนข้างสูง เพื่อให้ได้ภาพสำหรับการแปลความหมายต่อเนื่องกัน ซึ่งหากนำมาใช้งานกับกล้องรุ่นเก่าที่อัตราการเปลี่ยนภาพต่ำจะทำให้มีปัญหาในการประมวลผล อีกทั้งการหาสีผิวเป็นการวิเคราะห์สีผิวจากใบหน้าในทุกๆภาพ จึงใช้ทรัพยากรของเครื่องจำนวนมากเพื่อนำมาวิเคราะห์แปลความหมายภาพ ซึ่งส่งผลกระทบถึงการควบคุมตัวละครไม่ต่อเนื่อง หรืออาจทำให้ตัวละครหลุดจากการควบคุม
3. การพัฒนาในส่วนเกมแอพพลิเคชันมีการนำ Unity มาใช้พัฒนา ทำให้มีปัญหาในการศึกษาการใช้งานในบางเรื่อง เช่น การนำไลบรารี่อื่นมาใช้ร่วมกัน เป็นต้น รวมไปถึงการค้นคว้าข้อมูลต่างๆและข้อจำกัดบางประการในการพัฒนา
4. การตรวจจับของโปรแกรมภายใต้สภาพแวดล้อมต่างๆ เช่นปริมาณแสงที่มีมากหรือน้อยจนเกินไป วัตถุรบกวนที่มีสีใกล้เคียงกับสีผิวมนุษย์ จำนวนคนภายในภาพ อาจทำให้การตรวจจับเกิดข้อผิดพลาด

5.2 ข้อเสนอแนะ

1. การเลือกใช้กล้องสำหรับทำงานกับโปรแกรม ควรเลือกใช้กล้องที่มีอัตราความถี่ของการรับภาพและความละเอียดสูง เพื่อความเร็วในการรับภาพ และลดการกระตุกในการประมวลผล
2. การเลือกใช้คอมพิวเตอร์ ควรเลือกใช้คอมพิวเตอร์ที่มีประสิทธิภาพสูงจะทำให้ความสามารถในการประมวลผลมีประสิทธิภาพดีขึ้น

5.3 แนวทางการพัฒนาในอนาคต

1. พัฒนาฟังก์ชันของเกมแอพพลิเคชันให้มีมากขึ้น เพื่อเพิ่มความสนุก และความท้าทายที่เพิ่มขึ้น
2. พัฒนาแอพพลิเคชันให้สามารถตรวจจับได้ทั้งร่างกาย สามารถใช้ส่วนขามาควบคุมเกมได้เพิ่มขึ้น
3. พัฒนาวิธีการตรวจจับสีผิวให้มีประสิทธิภาพ และเพิ่มความแม่นยำในสภาพแวดล้อมต่างๆได้ดียิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารอ้างอิง

[1] นายนภชา วรวัฒน์. (2553). การติดตามใบหน้าและลักษณะเด่นบนใบหน้าและการสร้างศีรษะจากลำดับภาพสองมิติ(ออนไลน์). สืบค้นจาก:http://161.200.126.13/Download/2301499_Senior_Project/Report/Year_2553/ Comp/Face and Facial Feature Tracking and Head Reconstruction from 2D Image Sequence.pdf [10 ตุลาคม 2556]

[2] Kedija Kedir Idris. (2555). Pupil Center Location using HAAR Features (ออนไลน์). สืบค้นจาก : http://mmlab.disi.unitn.it/wiki/images/7/71/Haar.png [9 ตุลาคม 2556]

[3] Paul Viola. (2551). Robust Real-time Object Detection (ออนไลน์). สืบค้นจาก : http://www.hpl.hp.com/techreports/Compaq-DEC/CRL-2001-1.pdf [10 ตุลาคม 2556]

[4] Anonymous. (2550). Converting Color Data Between Color Spaces (ออนไลน์). สืบค้นจาก : http://www.mathworks.com/help/releases/R2013b/images/hsvcone.gif [10 ตุลาคม 2556]

[5] Nelson Filipe. (2552). Desenvolvimento de um Sensor de Tráfego baseado em Visão por Computador (ออนไลน์). สืบค้นจาก : http://www.couleur.org/spaces/YCbCrspace.jpg [15 กันยายน 2556]

[6] Nusirwan Anwar bin Abdul Rahman, Kit Chong Wei and John See. "RGB-H-CbCr Skin Colour Model for Human Face Detection", *MMU Intermational Symposium on Information & Communications Technologies (M2USIC 2006)*, PJ, Malaysia, 2006.

[7] P. Peer, J. Kovac, F. Solina, "Human Skin ColourClustering for Face Detection", *EUROCON1993*, Ljubljana,Slovenia, pp. 144-148, September 2003.

[8] Chai D and Ngan K.N, "Locating Facial Region of a Head-and-shoulders color image," *In :Proceedings of the 3rd International Conference on Automatic Face and Gesture Recognition. Nara*, Japan, pp.124~129, 1998.

[9] Wang Y. and Yuan, B., "A novel approach for human face detection from color images under complex background", *Pattern Recognition*, 34(10), 1983-1992, 2001.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

[10] J. Chaves-GonzÃ¡lez , M. Vega-RodrÃguez , J. GÃ³mez-Pulido and J. SÃ¡nchez-PÃ©rez "Detecting skin in face recognition systems: A colour spaces study", *Digital Signal Process.*, vol. 20, no. 3, pp.806 -823 2010.

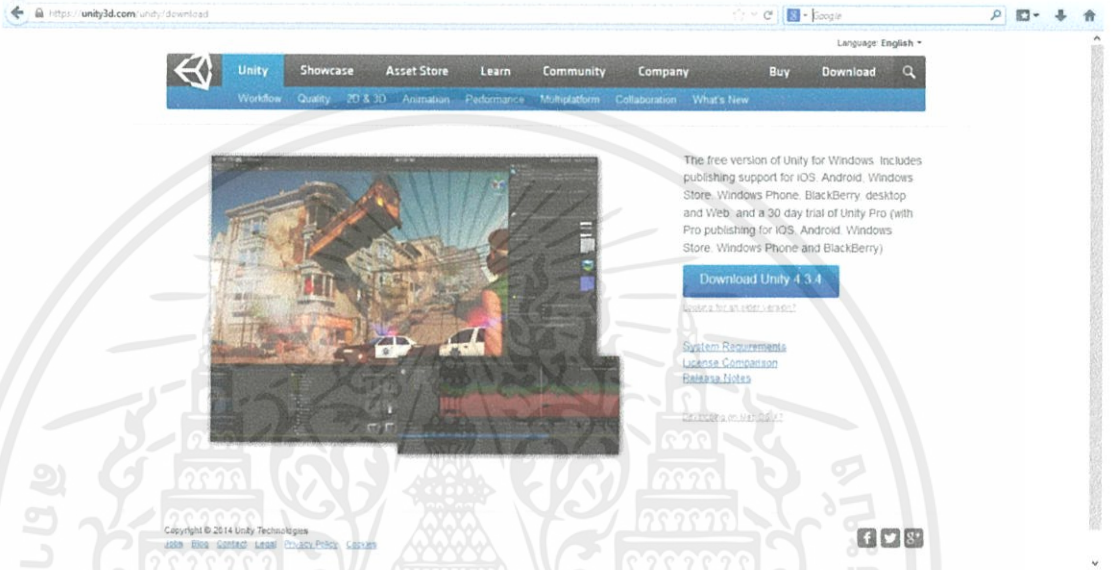
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก
การติดตั้งโปรแกรมและวิธีใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

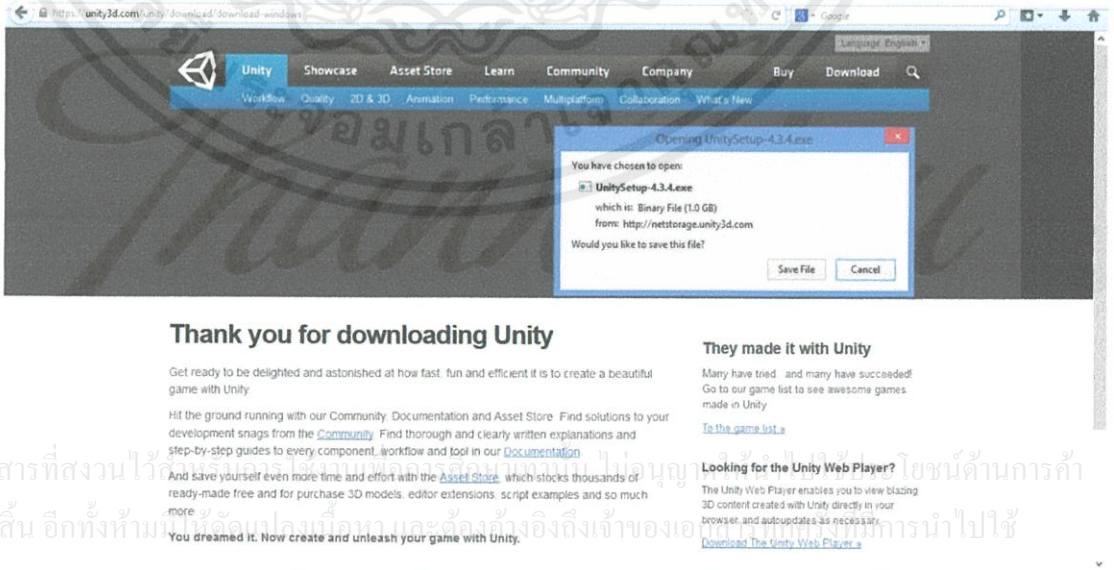
- **การติดตั้งโปรแกรม Unity**

1. การติดตั้งโปรแกรม Unity สามารถติดตั้งโดยการเข้าไปดาวน์โหลดโปรแกรมนี้ จากเว็บไซต์ http://unity3d.com/unity/download เมื่อปรากฏหน้าต่างเว็บไซต์ขึ้นมา ให้ทำการคลิกที่ปุ่มสีฟ้า “Download Unity 4.3.4”



รูปที่ ก-1 แสดงหน้าเว็บไซต์ของ Unity

2. หลังจากคลิกแล้ว จะปรากฏข้อความขึ้นมาดังแสดงในรูป ก-2 เพื่อดาวน์โหลดโปรแกรม ให้คลิกที่ปุ่ม “Save File”



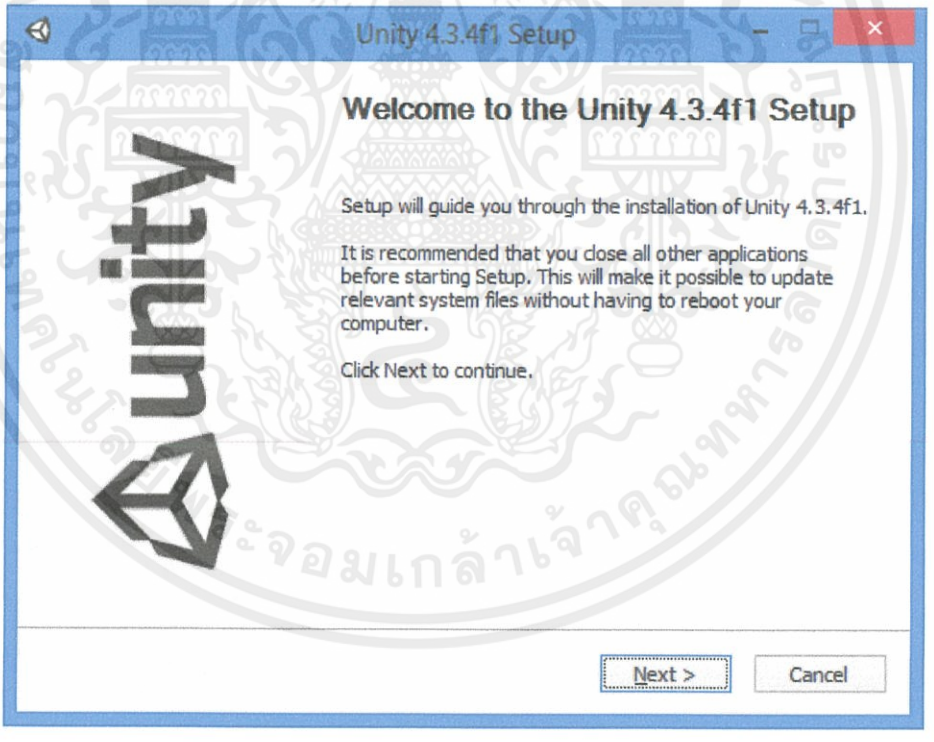
รูปที่ ก-2 แสดงหน้าเว็บไซต์สำหรับการดาวน์โหลด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมื่อทำการดาวน์โหลดเรียบร้อยแล้ว ให้เข้าไปในไดเรกทอรี่ที่เก็บไฟล์ UnitySetup.exe จากนั้นทำการเปิดโปรแกรม UnitySetup.exe

รูปที่ ก-3 แสดงไดเรกทอรี่ที่เก็บไฟล์ UnitySetup.exe

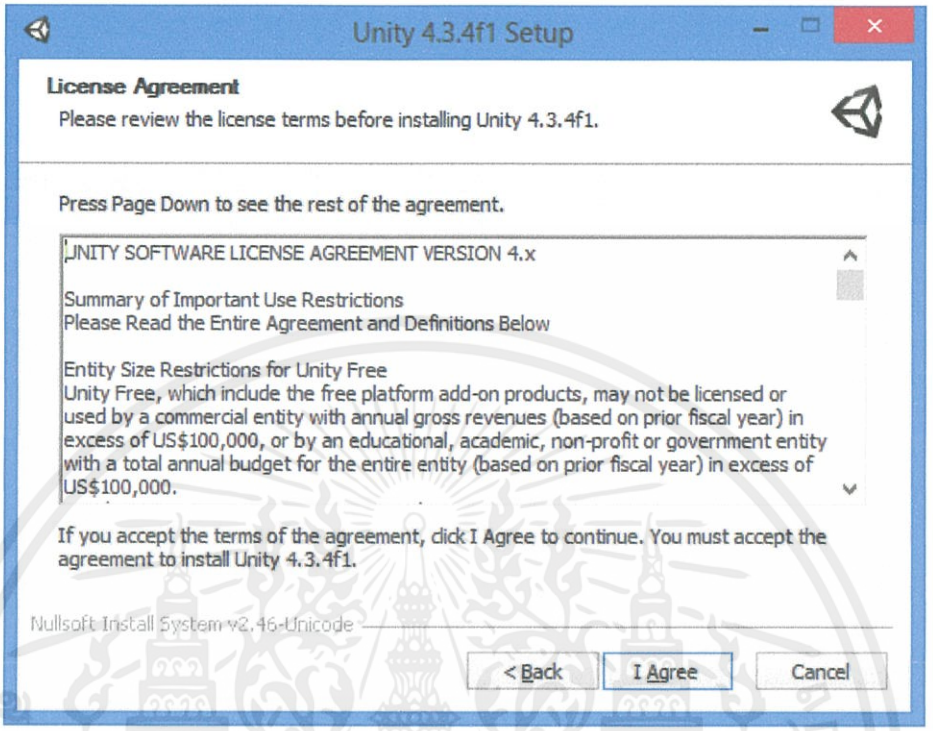
4. โปรแกรมจะแสดงคำแนะนำต่างๆก่อนที่จะเริ่มการติดตั้ง ให้กด “Next”



รูปที่ ก-4 แสดงหน้าเริ่มต้นในการติดตั้งโปรแกรม Unity

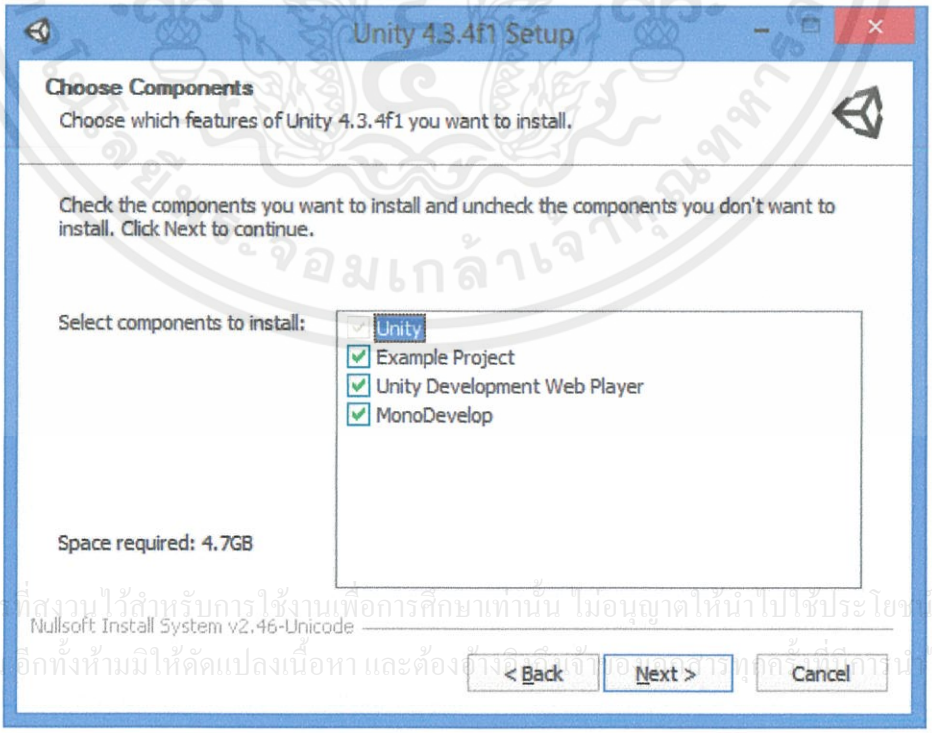
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5. โปรแกรมแสดงข้อตกลงและเงื่อนไขต่างๆของโปรแกรม ให้กด “I Agree” เพื่อยอมรับเงื่อนไขและข้อกำหนดต่างๆ



รูปที่ ก-5 โปรแกรมแสดงข้อตกลงและเงื่อนไขต่างๆ

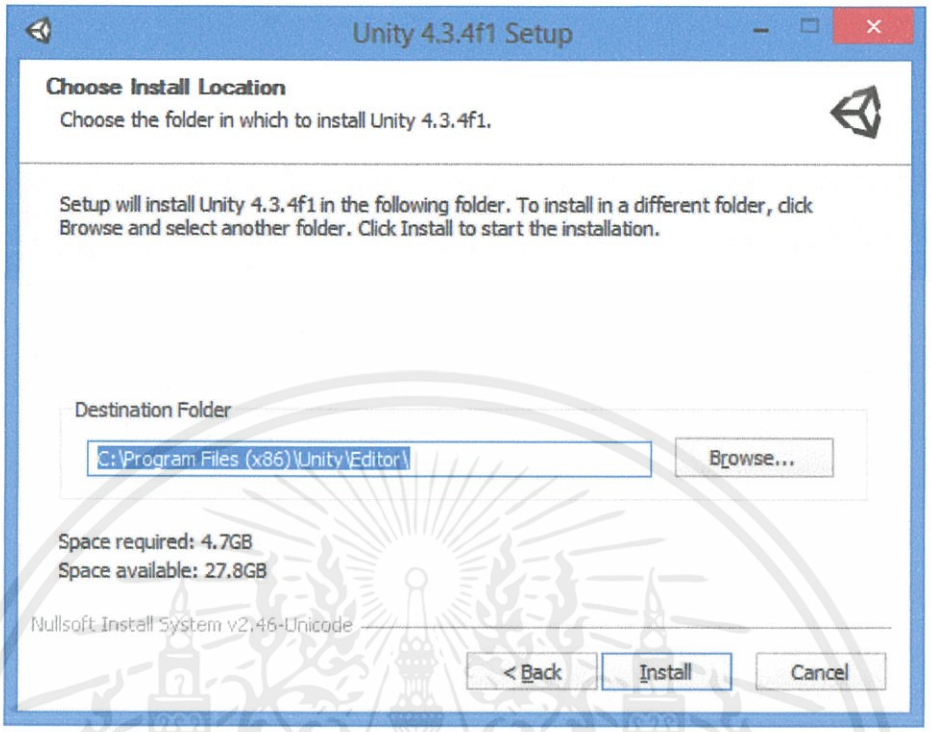
6. โปรแกรมจะให้เลือกรายละเอียดในการติดตั้ง



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก-6 โปรแกรมแสดงรายละเอียดที่จะติดตั้ง

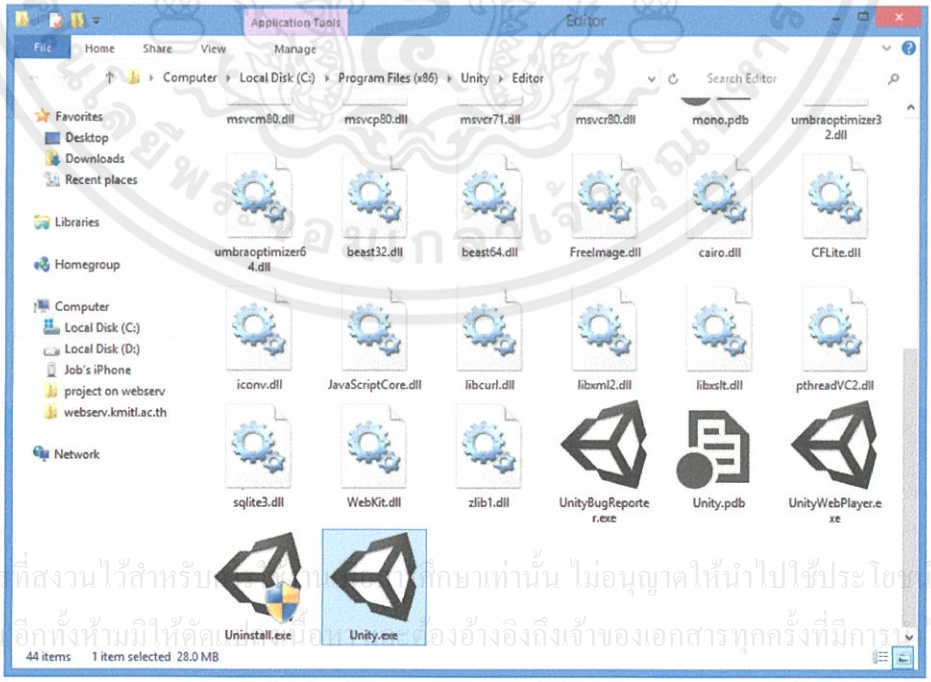
7. เลือกไดเรกทอรี่ที่ต้องการติดตั้ง จากนั้นกด “Install” จากนั้นรอโปรแกรมติดตั้งจนเสร็จ



รูปที่ ก-7 ไดเรกทอรี่ที่จะติดตั้งโปรแกรม

- **วิธีการใช้โปรแกรม Unity**

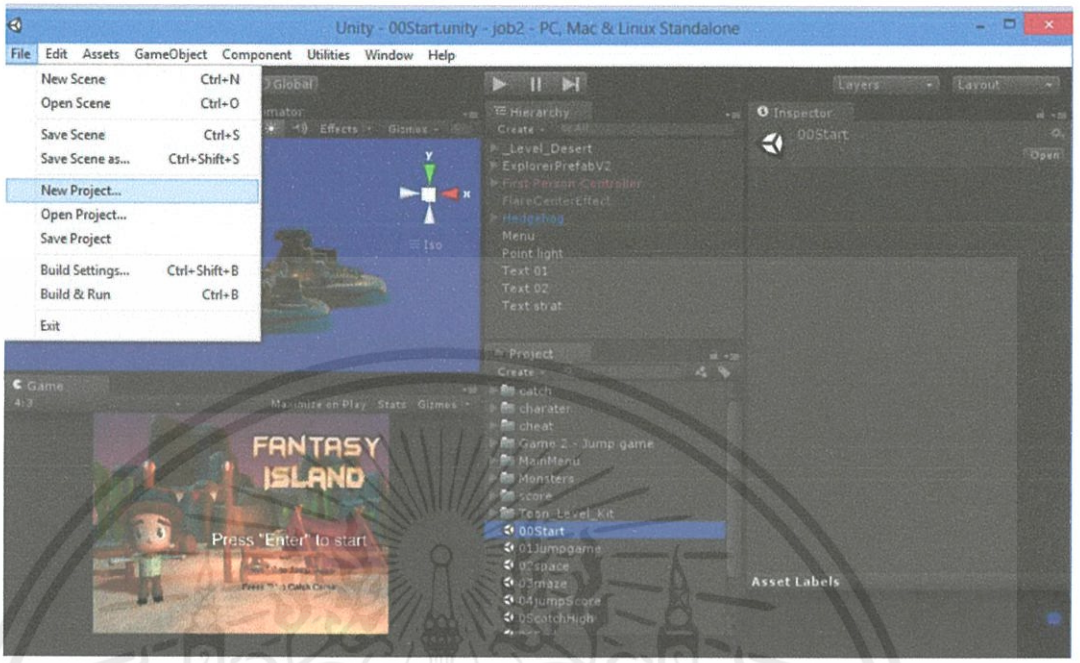
1. วิธีการใช้งานโปรแกรม Unity เริ่มจากให้เข้าไปในไดเรกทอรี่ที่ได้ทำการติดตั้งโปรแกรมไว้ เปิดโปรแกรมชื่อ Unity.exe



รูปที่ ก-8 โปรแกรม Unity.exe

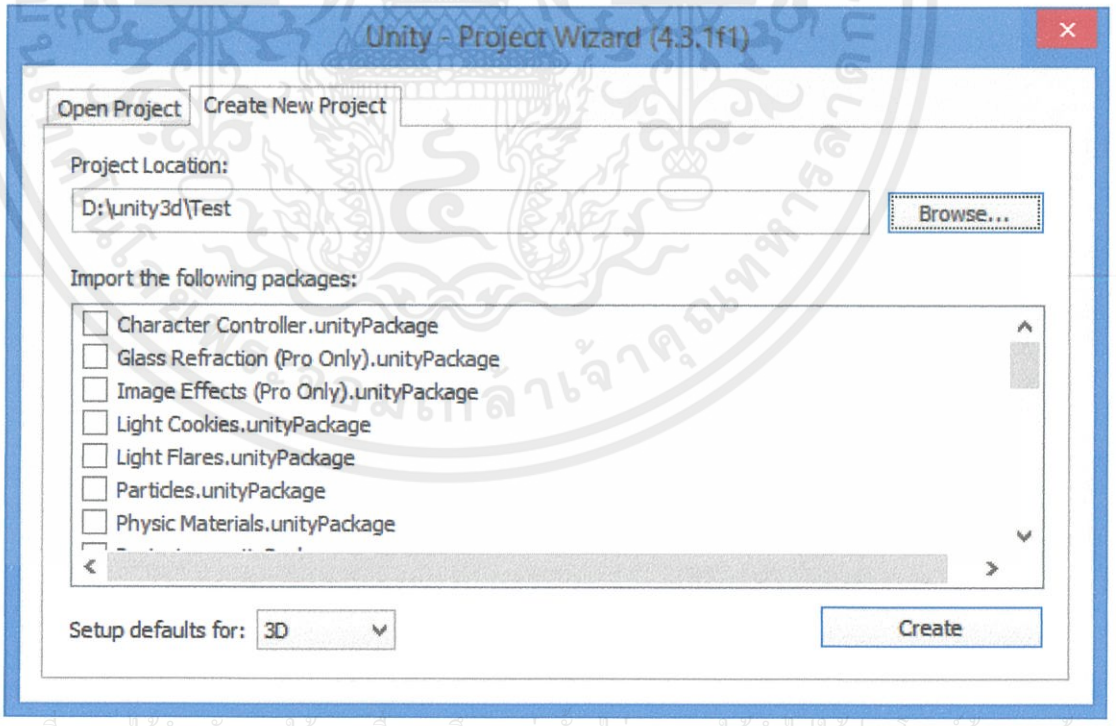
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เลือก File แล้ว New Project เพื่อสร้างงานขึ้นมาใหม่



รูปที่ ก-9 หน้าโปรแกรม Unity

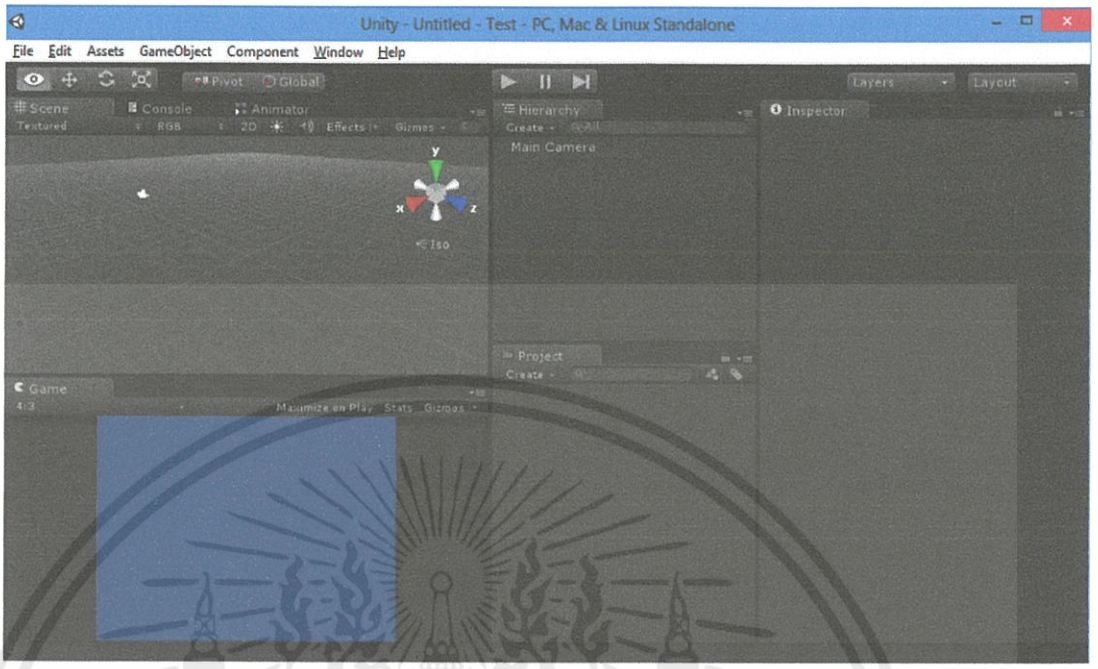
3. เลือกไดเรกทอรี่ที่ต้องการเก็บไฟล์งานนี้ จากนั้นกด “Create”



รูปที่ ก-10 หน้าต่างในการสร้างโปรเจค

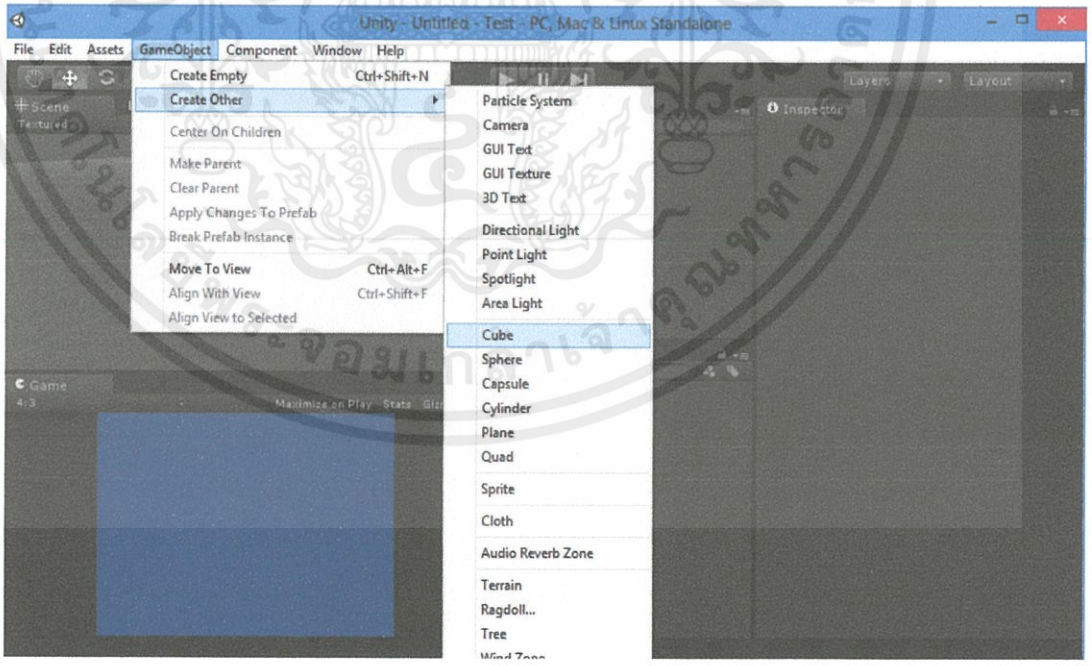
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4. จากนั้นโปรแกรมจะเปิดหน้าต่างขึ้นมาใหม่



รูปที่ ก-11 ตัวอย่างในการสร้างโปรเจค

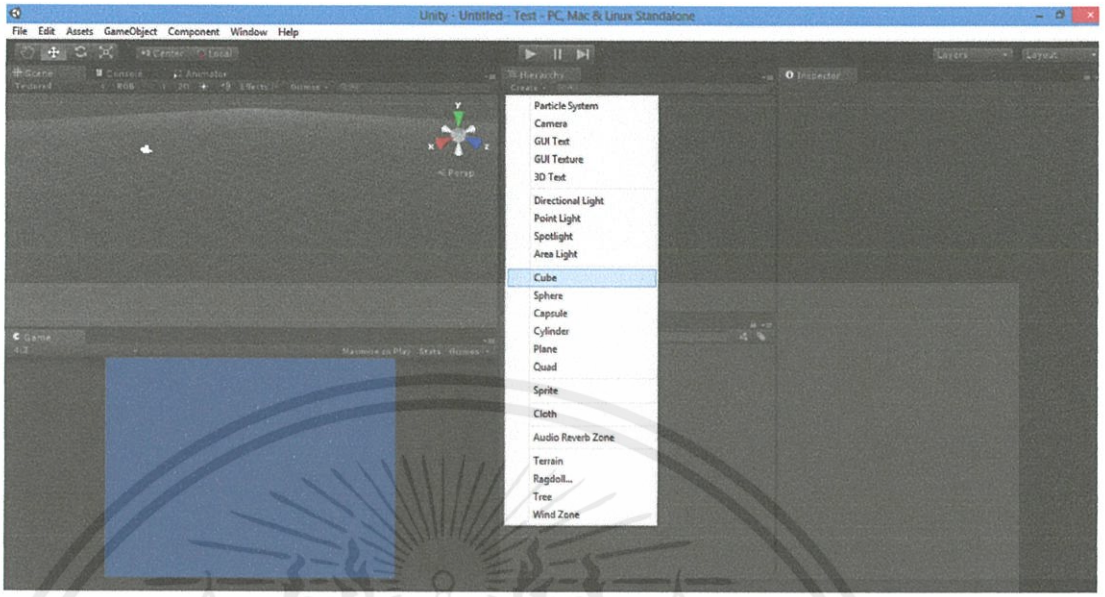
5. การสร้างวัตถุภายในเกมสามารถทำได้ 2 วิธี วิธีแรกให้ทำการคลิกที่ “GameObject” จากนั้นเลือก “Create Other” แล้วเลือกประเภทวัตถุที่ต้องการ



รูปที่ ก-12 ตัวอย่างการสร้างวัตถุแบบแรก

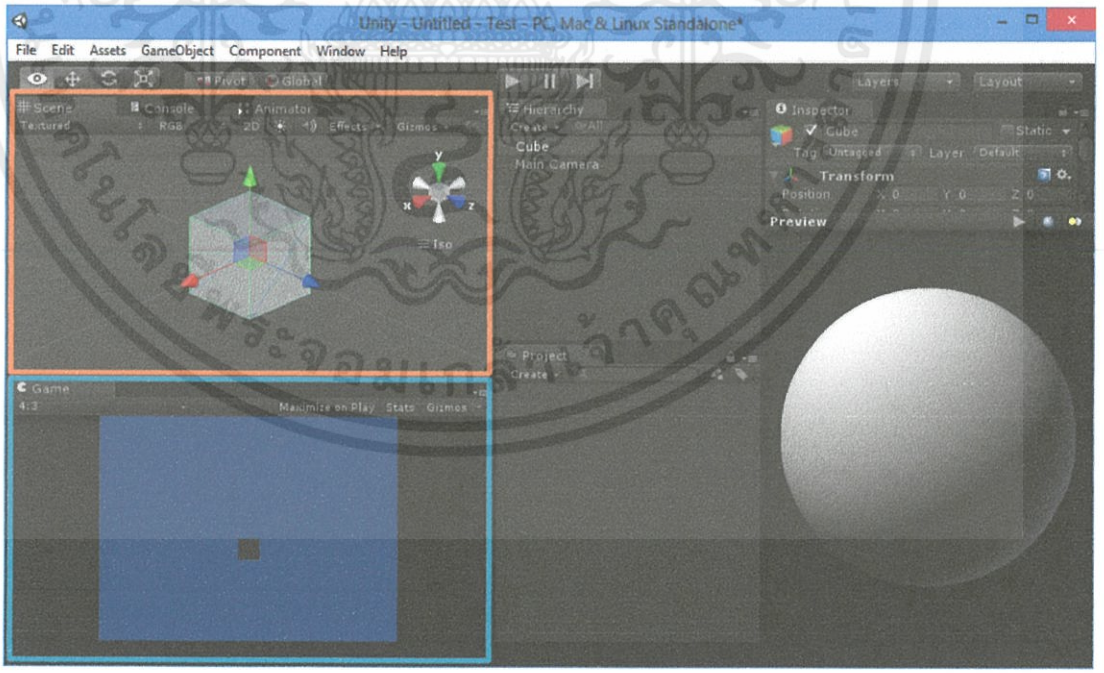
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนวิธีที่ 2 ให้ทำการคลิกที่ “Create” จากนั้นจึงเลือกประเภทวัตถุที่ต้องการสร้าง



รูปที่ ก-13 ตัวอย่างการสร้างวัตถุแบบที่สอง

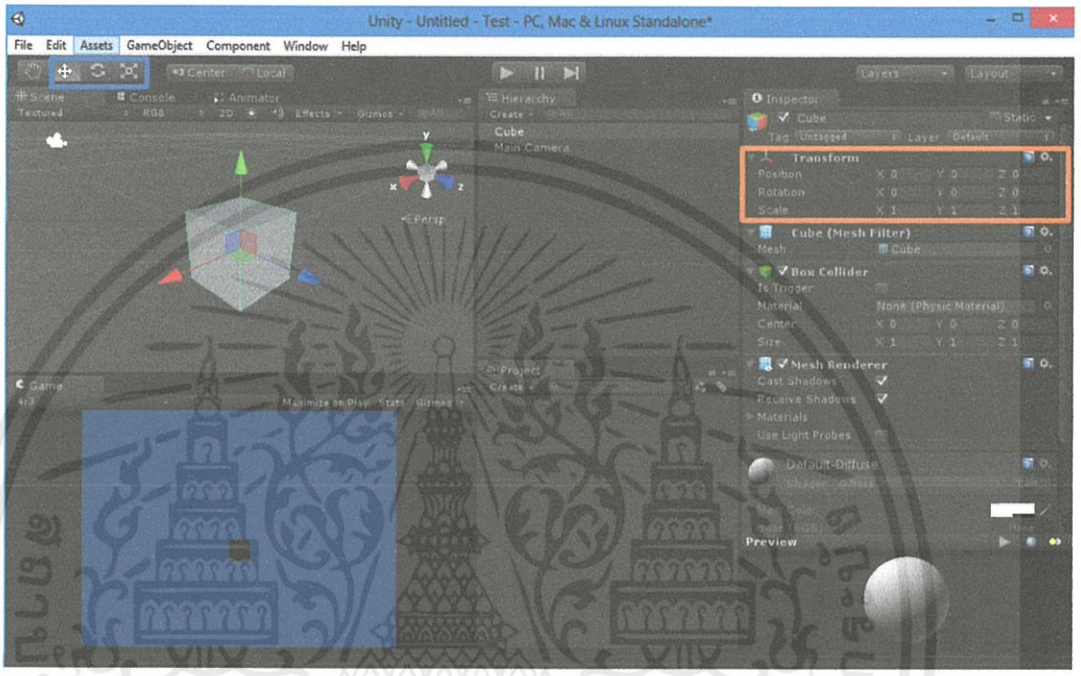
6. วัตถุจะปรากฏในฉากของเกม โดยหน้าต่างที่ปรากฏภายในโปรแกรม สามารถแบ่งออกเป็น 2 ส่วนหลักคือ “Scene” คือหน้าต่างที่ใช้ทำการสร้างเกม ส่วนหน้าต่างอีกแบบหนึ่งคือ “Game” คือหน้าต่างที่แสดงภาพในการเล่นเกมจริง



รูปที่ ก-14 ภาพวัตถุภายในเกม

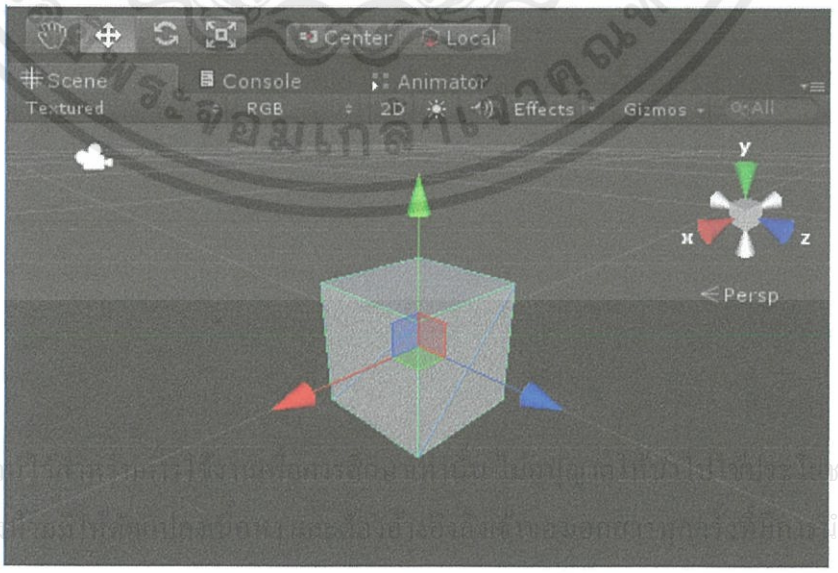
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งาน[illegible]ตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7. การปรับขนาดรูปร่าง ตำแหน่ง และทิศทางการหมุนของวัตถุภายในเกมนั้น สามารถกำหนดได้ 2 วิธี วิธีแรกคือการกำหนดโดยการใส่ค่าให้กับวัตถุนั้นๆลงในหัวข้อ “Transform” ซึ่งจะมีให้ระบุค่า Position(ตำแหน่งของวัตถุ) Rotation(ทิศทางการหมุนของวัตถุ) และ Scale(ขนาดของวัตถุ) ส่วนการกำหนดอีกแบบนั้น สามารถเลือกใช้คำสั่งโดยการคลิกไปที่รูปภาพเหล่านี้



รูปที่ ก-15 แสดงการปรับขนาด ตำแหน่ง และการหมุนของวัตถุ

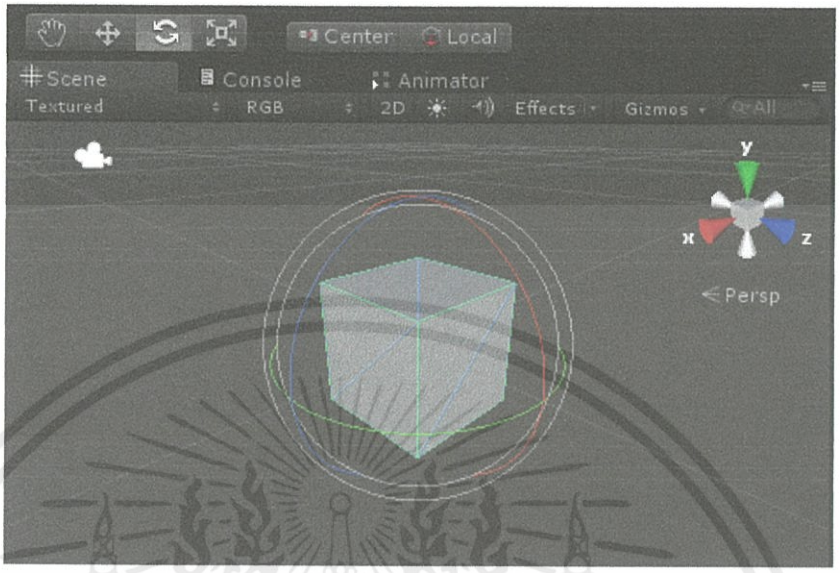
โดยปุ่มแรกหมายถึง การกำหนดตำแหน่งของวัตถุ สามารถเลื่อนตำแหน่งวัตถุภายในเกมโดยการเลื่อนที่ตำแหน่งลูกศรทั้ง 3 สี



รูปที่ ก-16 แสดงการกำหนดตำแหน่งของวัตถุ

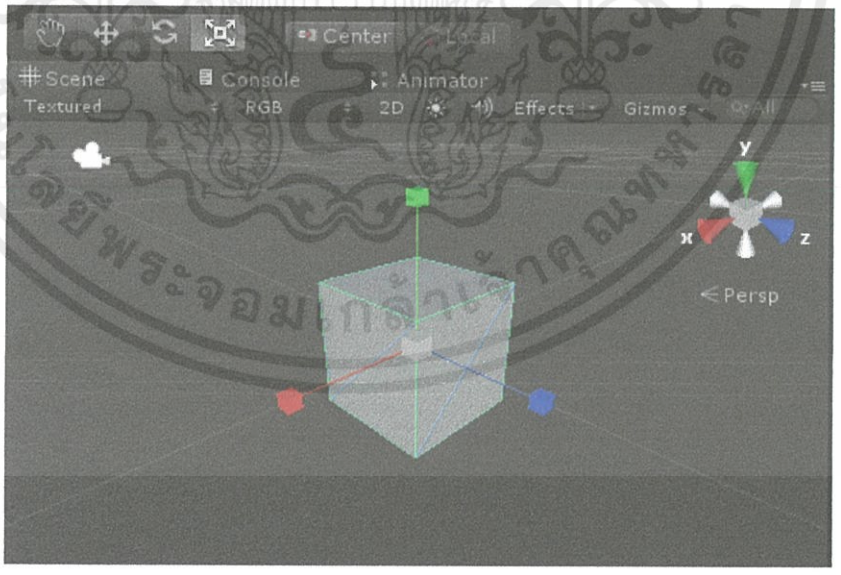
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปุ่มที่สองหมายถึง การกำหนดการหมุนของวัตถุ สามารถกำหนดการหมุนวัตถุภายในเกมโดยการคลิกหมุนที่เส้นวงกลมทั้ง 3 สี



รูปที่ ก-17 แสดงการกำหนดการหมุนของวัตถุ

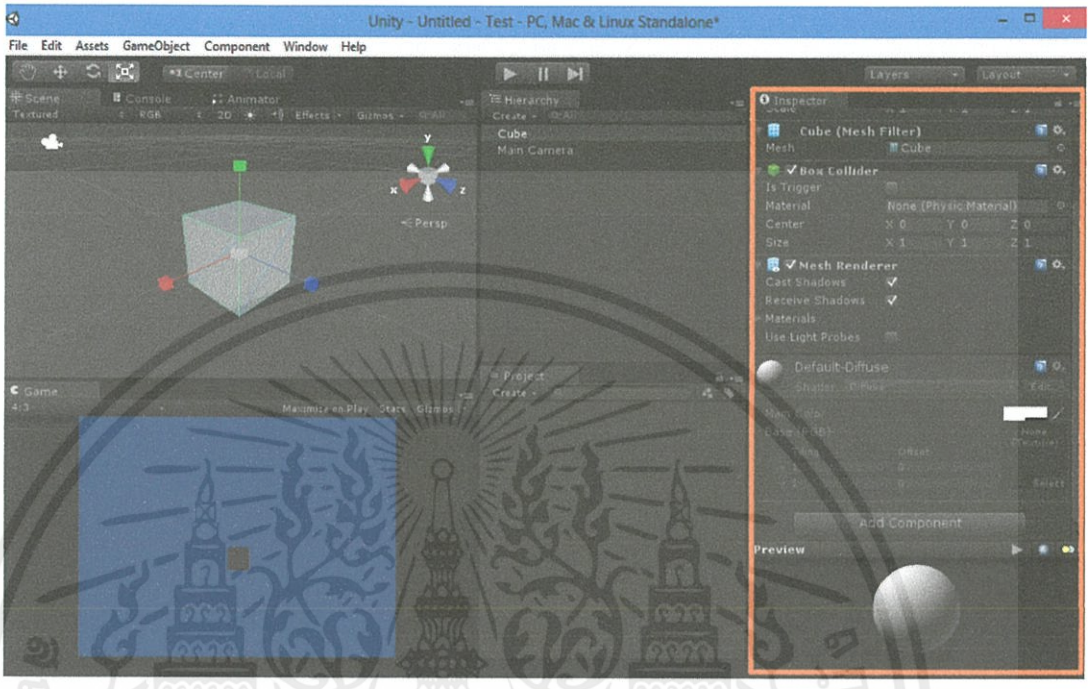
ปุ่มที่สามหมายถึง การกำหนดขนาดของวัตถุ สามารถกำหนดขนาดวัตถุภายในเกมโดยการคลิกที่กล่องสี่เหลี่ยมเล็กๆทั้ง 3 กล่อง โดยกล่องแต่ละสีจะขยายหรือย่อขนาดตามแกนต่างๆ ส่วนการขยายหรือย่อขนาดพร้อมกันทั้ง 3 แกน สามารถทำได้โดยการคลิกเลือกที่กล่องสีเทา



รูปที่ ก-18 แสดงการกำหนดขนาดของวัตถุ

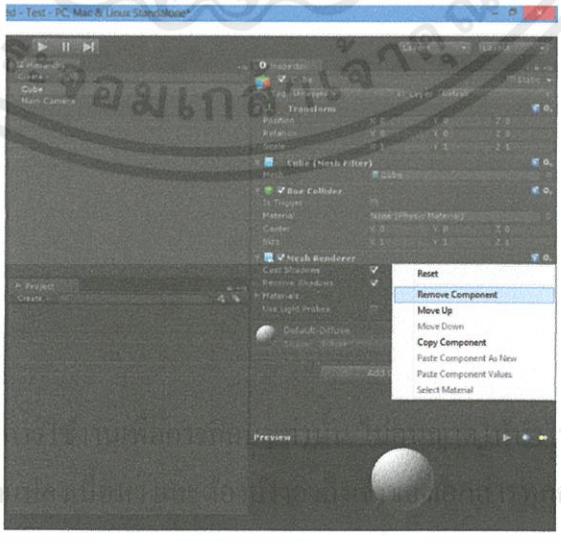
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8. ในการตั้งค่าส่วนประกอบอื่นๆของวัตถุ สามารถตั้งค่าได้จากช่อง “Inspector” โดยสามารถเพิ่มหรือลดส่วนประกอบได้ ซึ่งการเพิ่มส่วนประกอบสามารถทำได้โดยการคลิก “Add Component”



รูปที่ ก-19 แสดงการตั้งค่าส่วนประกอบของวัตถุ

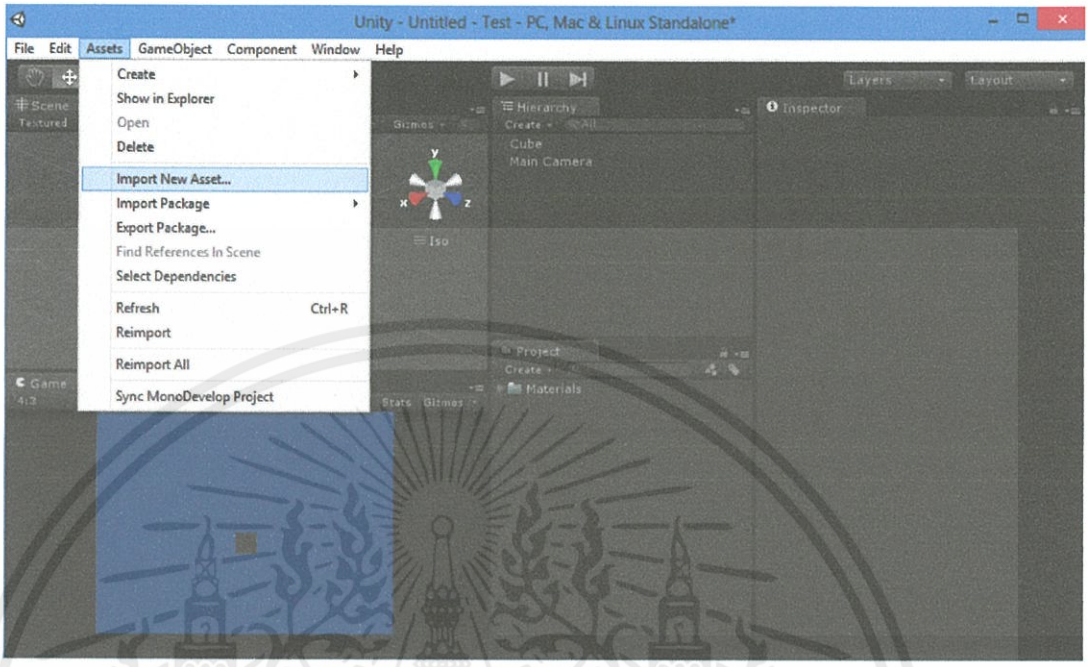
ส่วนการลดส่วนประกอบให้คลิกเลือกที่ จากนั้นเลือก “Remove Component” หรือสามารถตั้งค่าส่วนประกอบนั้นใหม่โดยการเลือก “Reset” สามารถเลื่อนตำแหน่งของส่วนประกอบให้อยู่ด้านบนโดยการเลือก “Move Up” และสามารถคัดลอกค่าของส่วนประกอบนั้นโดยการเลือก “Copy Component”



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ ปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัด ครั้งที่มีการนำไปใช้

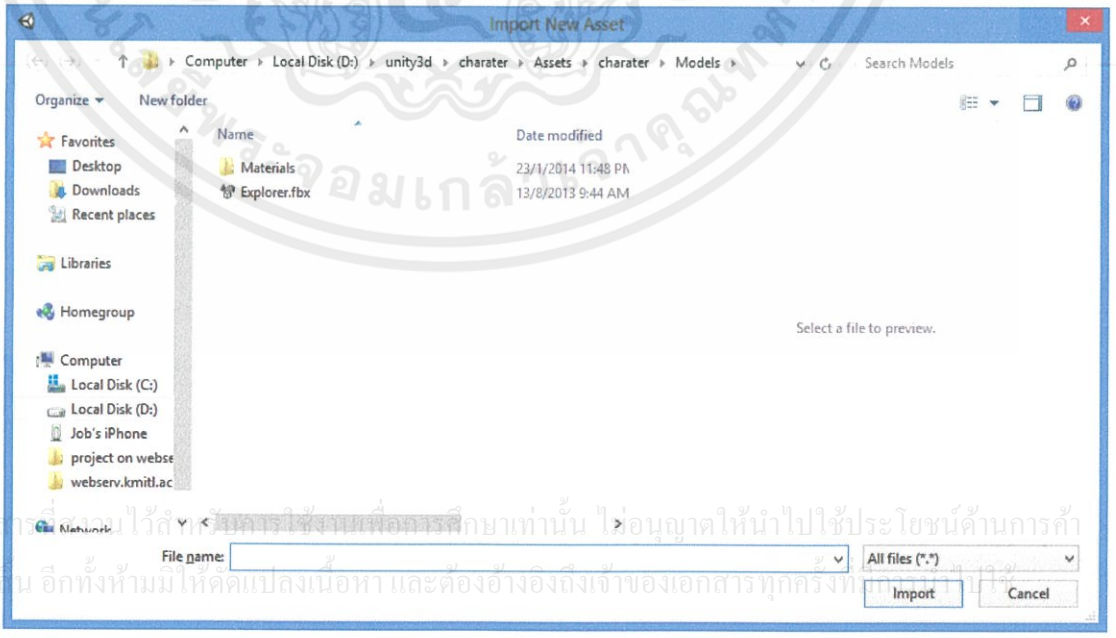
รูปที่ ก-20 แสดงคำสั่งต่างๆในส่วนประกอบของวัตถุ

9. การนำไฟล์วัตถุ 3 มิติมาใช้งานใน Unity ทำได้โดยการคลิกที่ “Assets” จากนั้น “Import New Assets”



รูปที่ ก-21 แสดงการนำเข้าวัตถุ 3 มิติมาใช้งาน

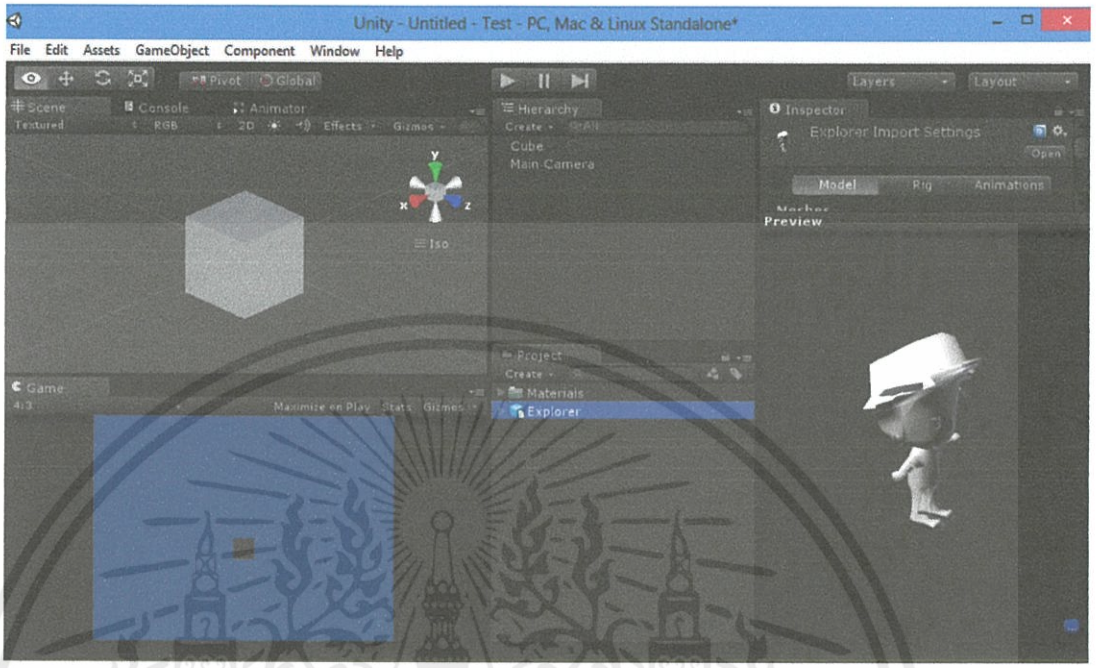
10. เลือกไฟล์วัตถุ 3 มิติที่ต้องการ โดยวัตถุเหล่านั้นต้องมาจากโปรแกรม 3 มิติที่ Unity รองรับ ได้แก่โปรแกรมดังต่อไปนี้ Maya, Cinema 4D, 3ds Max, Cheetah3D, Modo, Lightwave หรือ Blender นอกจากนั้นแล้ว ยังต้องบันทึกเป็นสกุลไฟล์ที่ Unity รองรับเช่นกัน ได้แก่ .fbx, .dae, .3DS, .dxf และ .obj ซึ่งนามสกุลไฟล์ที่คนส่วนใหญ่นิยมใช้ใน Unity คือ .fbx



รูปที่ ก-22 ไดเรกทอรี่ที่เก็บไฟล์ 3 มิติ

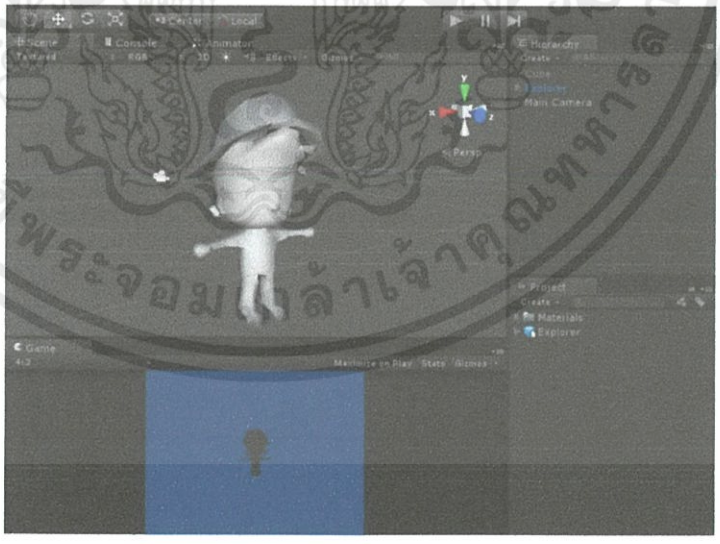
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

11. เมื่อทำการเลือกวัตถุที่ต้องการแล้ว หลังจากกด “Import” วัตถุนั้นจะเข้ามาอยู่ภายในโปรเจ ซึ่งสามารถนำไปใช้งานได้ทันที



รูปที่ ก-23 ตัวอย่างไฟล์ 3 มิติที่นำเข้ามา

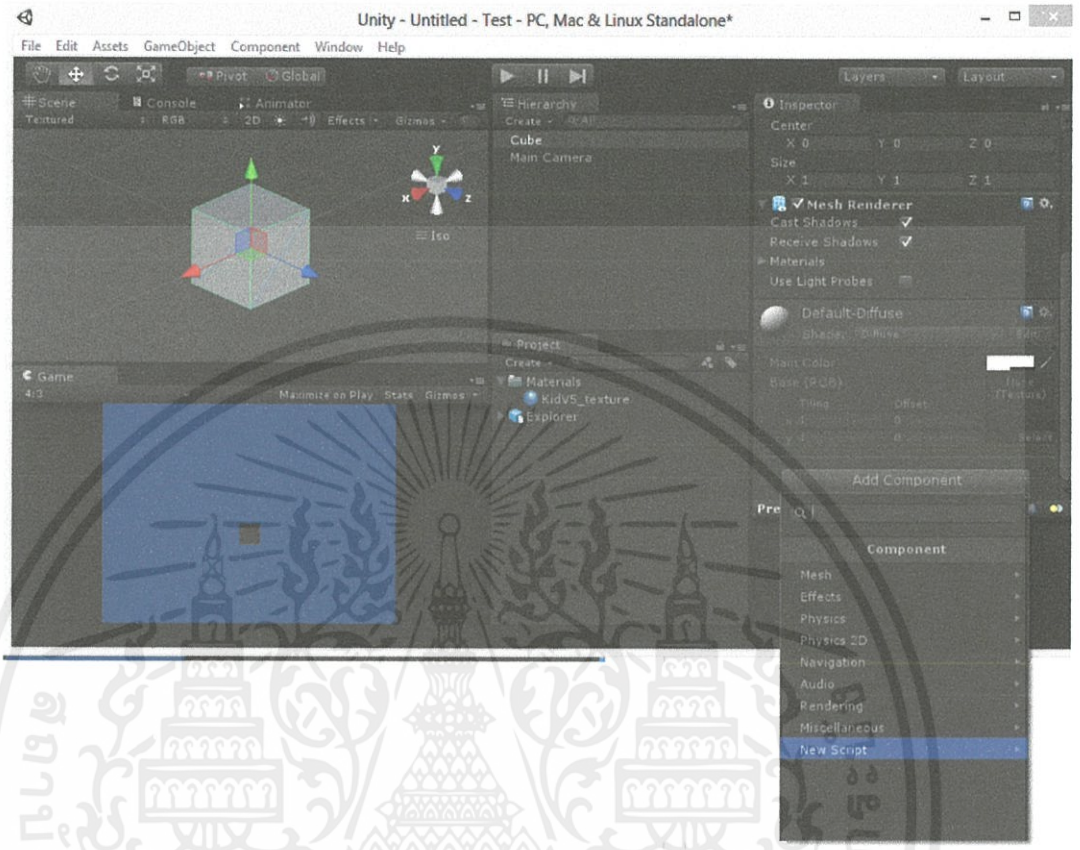
ซึ่งการนำวัตถุเข้ามาใช้งานภายในฉากที่กำลังสร้างอยู่นั้น ทำได้โดยการคลิกเลือกที่ชื่อของวัตถุที่นำเข้ามา (ในที่นี้ชื่อว่า “Explorer”) แล้วลากเข้าไปภายในฉากหรือ “Hierarchy”



รูปที่ ก-24 แสดงการนำไฟล์ 3 มิติเข้ามาภายในฉาก

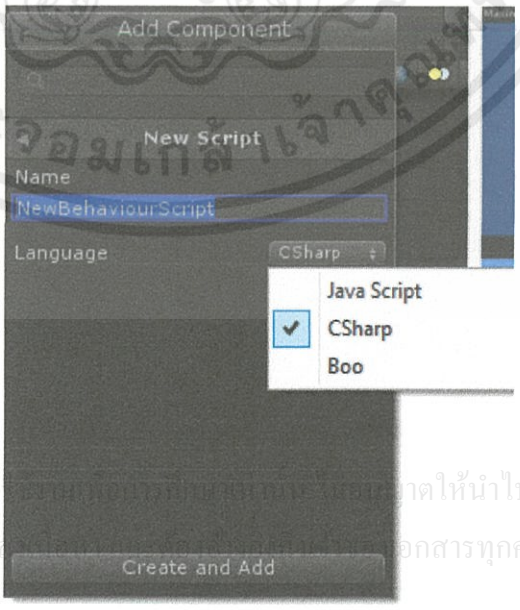
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

12. การเขียนโปรแกรมเพื่อควบคุมวัตถุ สามารถทำได้โดยการเลือกวัตถุที่ต้องการ จากนั้นคลิกที่ “Add Component” เลือก “New Script”



รูปที่ ก-25 วิธีการสร้างคำสั่ง

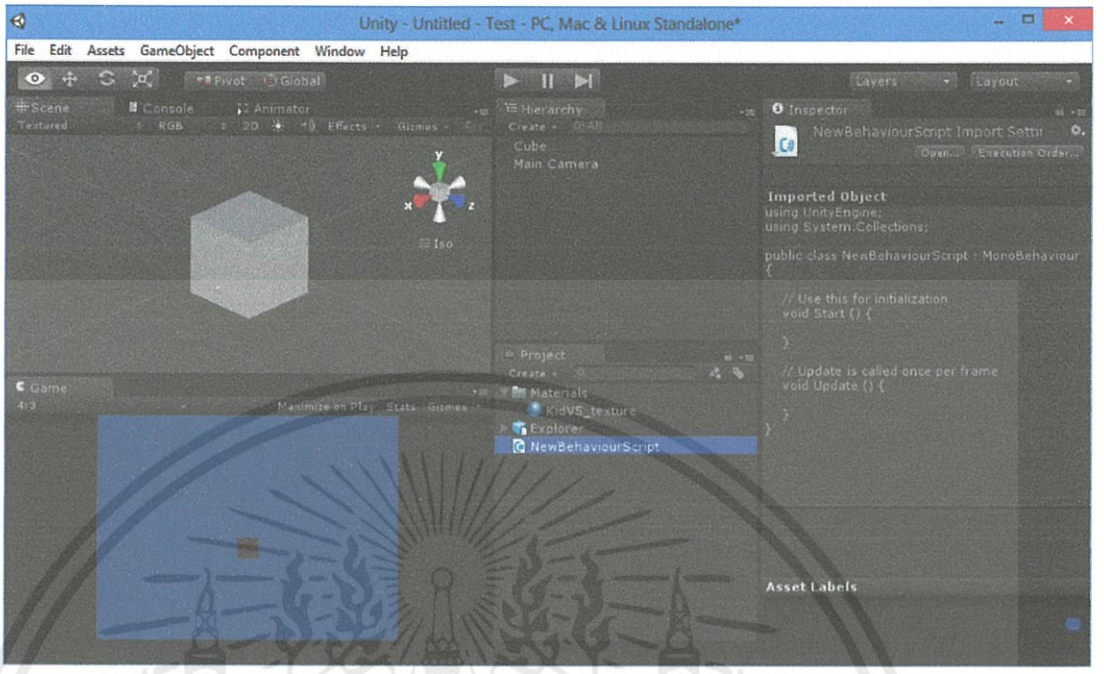
13. ตั้งชื่อคำสั่ง และเลือกภาษาที่ต้องการเขียน จากนั้นกด “Create and Add”



รูปที่ ก-26 ขั้นตอนการสร้างคำสั่ง

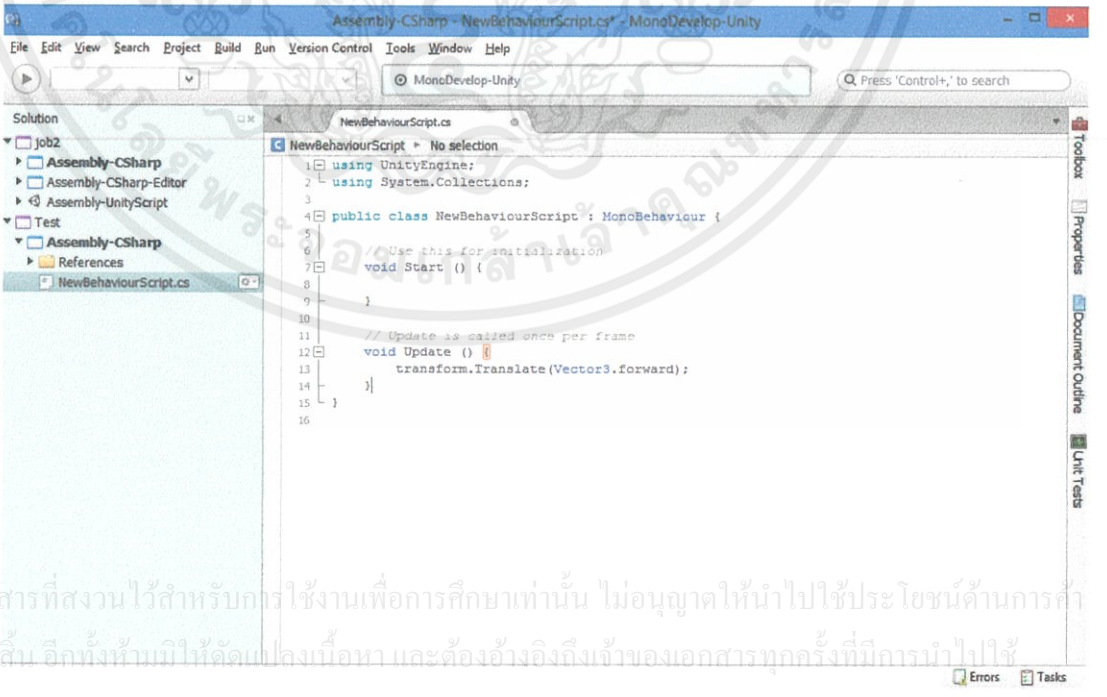
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการ
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแป
ตให้นำไปใช้ประโยชน์ด้านการค้า
อกสารทุกครั้งที่มีการนำไปใช้

14. เปิดคำสั่งที่สร้างขึ้นมาโดยกดที่คำสั่ง 2 ครั้ง



รูปที่ ก-27 ตัวอย่างไฟล์คำสั่ง

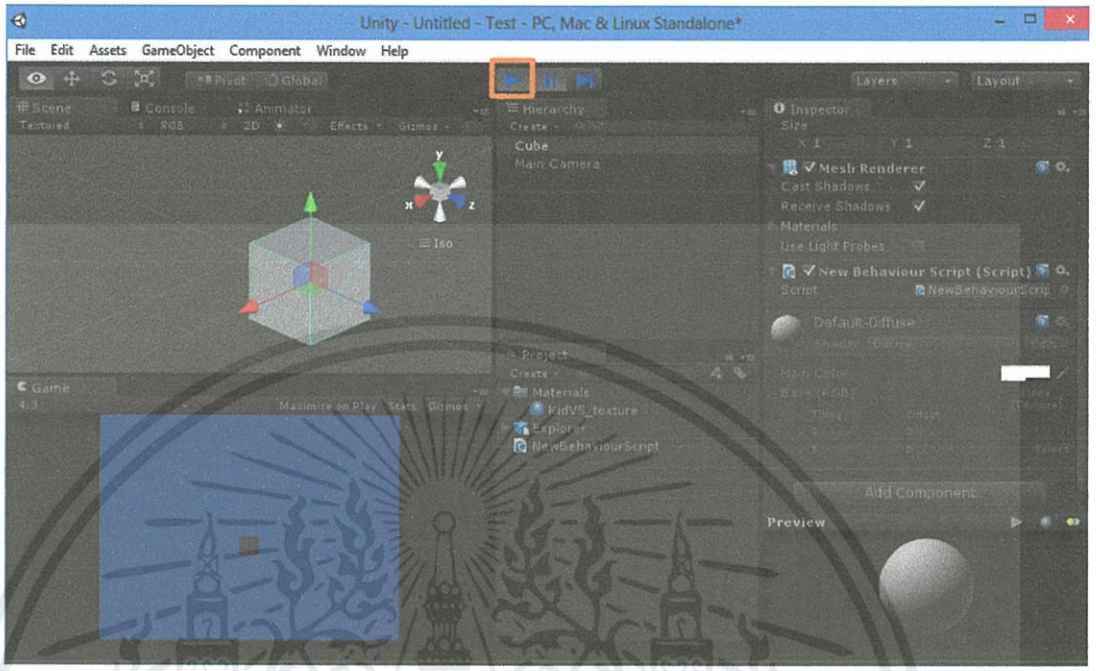
15. ทำการใส่คำสั่งที่ต้องการกำหนดลงไปในคำสั่ง เช่น transform.Translate(Vector3 .forward); คือการสั่งให้วัตถุเคลื่อนที่ไปข้างหน้า เป็นต้น จากนั้นบันทึกคำสั่งที่กำหนด โดยการกด Ctrl+S



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

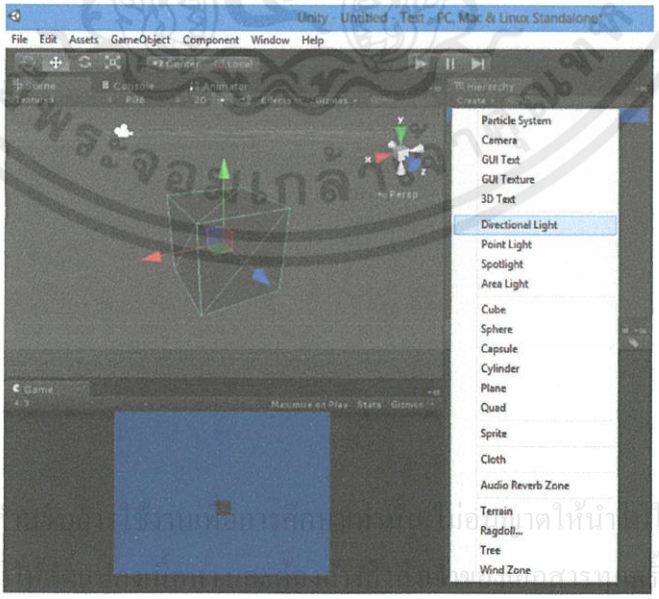
รูปที่ ก-28 ตัวอย่างคำสั่ง

16. สามารถทดลองคำสั่งที่เขียนขึ้นด้วยการกดปุ่มสามเหลี่ยมที่โปรแกรม Unity โปรแกรมจะทำงานตามคำสั่งที่ได้เขียนขึ้นมา



รูปที่ ก-29 ภาพทดลองการทำงาน

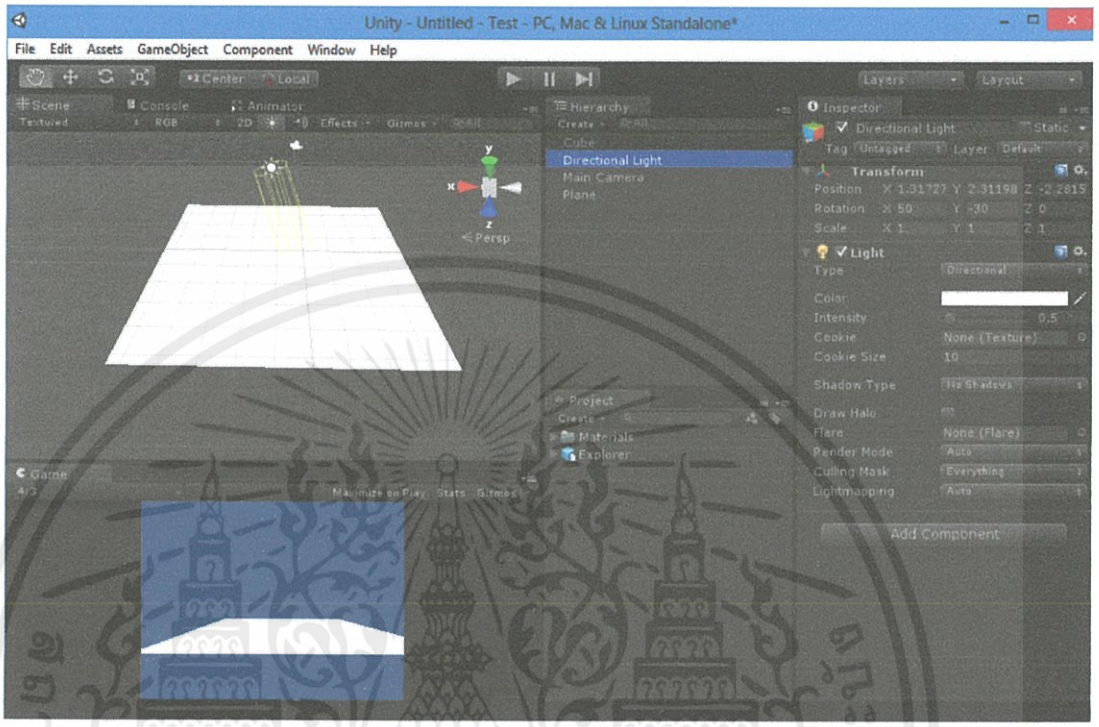
17. การใส่แสงภายในเกม ประเภทของแสงสามารถแบ่งออกได้เป็น 4 ประเภท ได้แก่ 1.Directional Light 2.Point Light 3.Sportlight และ 4.Area Light โดยการใส่แสงสามารถทำได้โดยการคลิกที่ “Create” แล้วเลือกประเภทแสงที่ต้องการใส่ ซึ่งแสงประเภทต่างๆจะมีความแตกต่างกันออกไป



รูปที่ ก-30 แสดงการใส่แสง

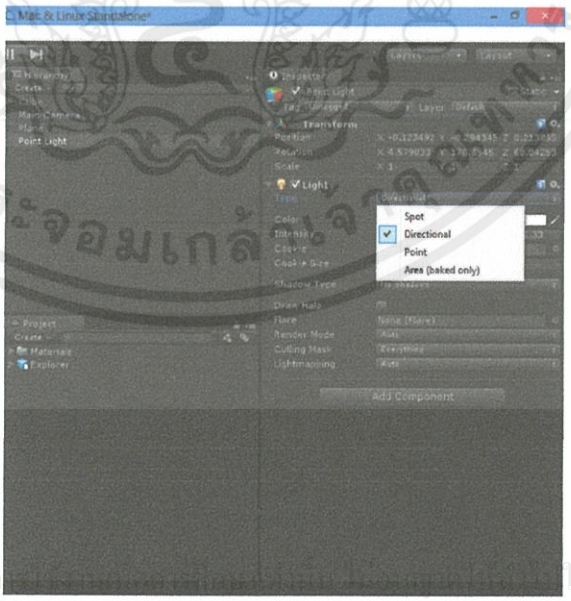
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17.1 การใส่แสงประเภท Directional Light ซึ่งหลังจากการใส่แสงแล้ว สามารถปรับเปลี่ยนประเภทแสงภายหลังในได้อีกด้วย โดยการคลิกเลือกที่ “Type”



รูปที่ ก-31 แสดงการใส่ Directional Light

โดยหลังจากคลิกเลือก “Type” แล้ว จะปรากฏประเภทของแสงให้เลือกดังภาพ

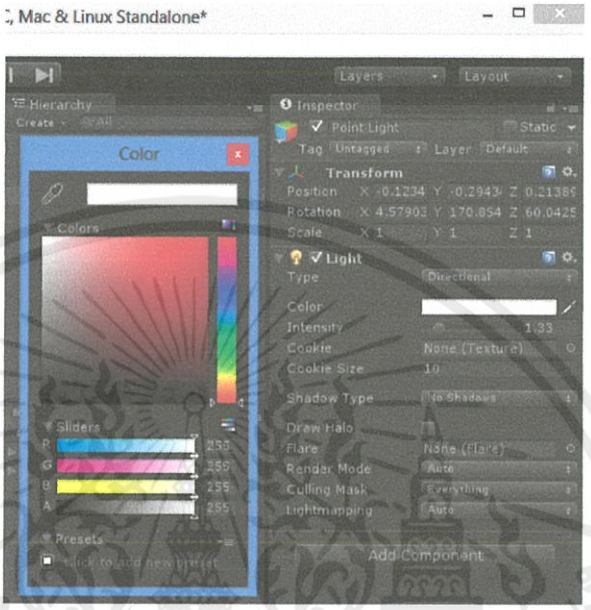


รูปที่ ก-32 แสดงการเปลี่ยนประเภทแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

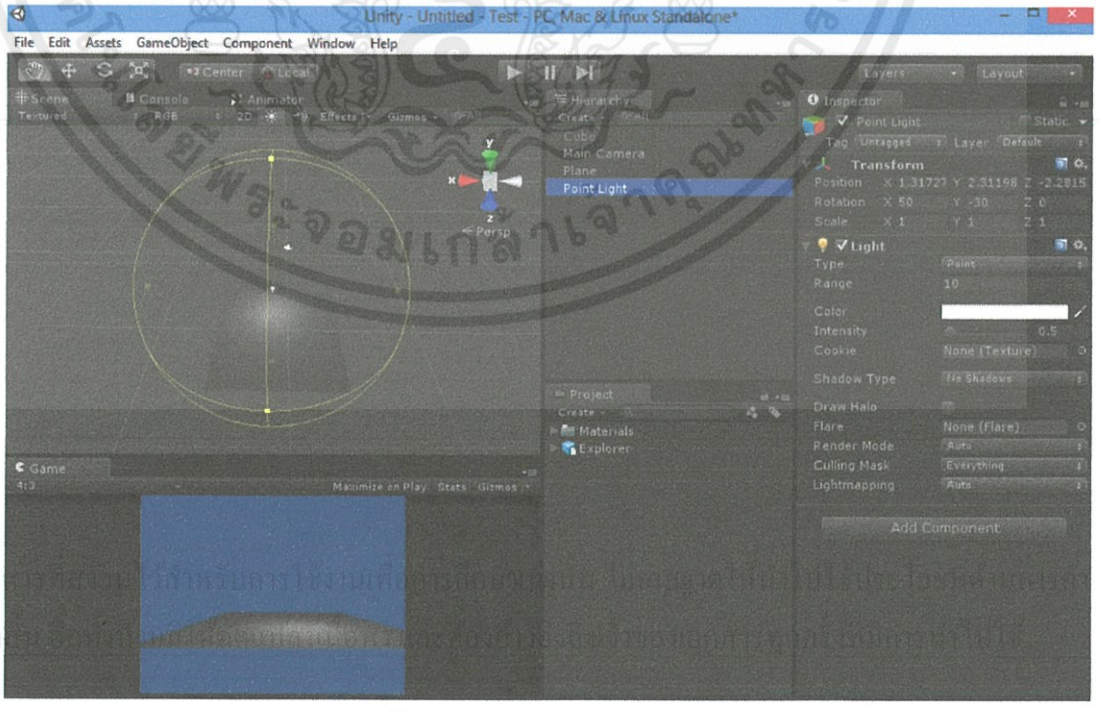
นอกจากการเปลี่ยนประเภทของแสงแล้ว ยังสามารถเปลี่ยนสีของแสงและความสว่างได้เช่นกัน โดยการเปลี่ยนสีของแสง เริ่มต้นจากการคลิกไปที่ช่อง

หลังจากนั้นจะปรากฏหน้าต่างเพื่อทำการเลือกสีได้ตามต้องการ และสามารถกำหนดค่าความสว่างของแสงได้ที่ช่อง “Intensity”



รูปที่ ก-33 แสดงการเปลี่ยนสีของแสง

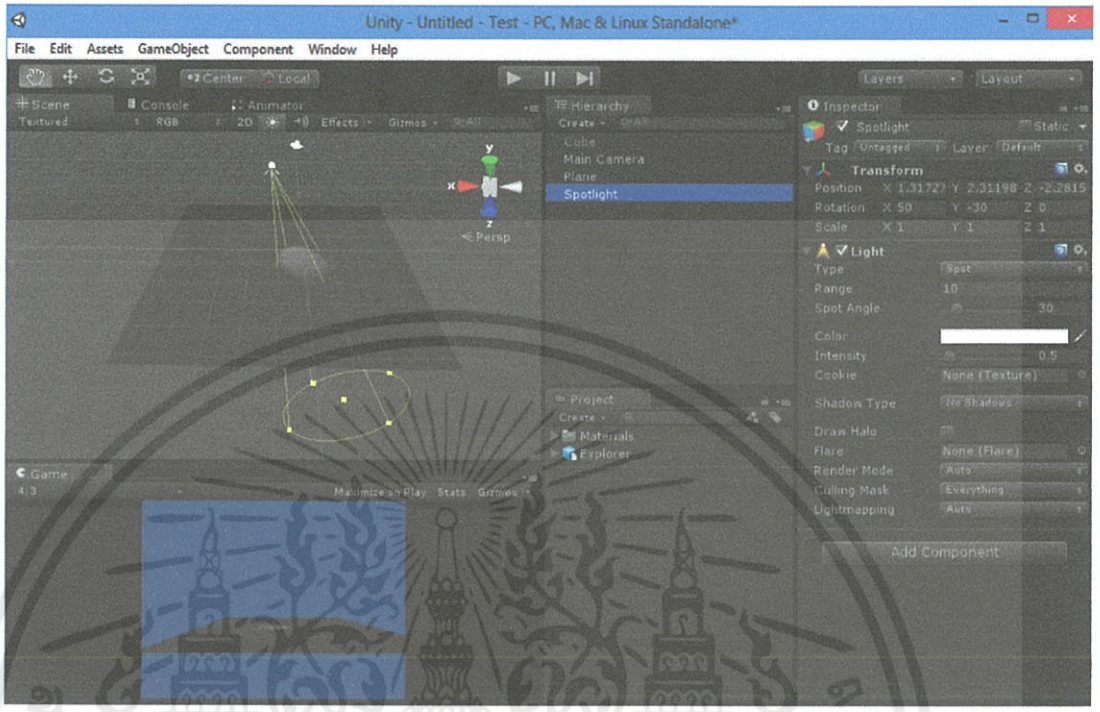
17.2 การใส่แสงประเภท Point Light การกำหนดค่าต่างๆจะคล้ายประเภท Directional Light ดังที่กล่าวไว้ข้างต้น แต่จะเพิ่มการกำหนดระยะของแสงที่คำสั่ง “Range”



รูปที่ ก-34 แสดงการใส่ Point Light

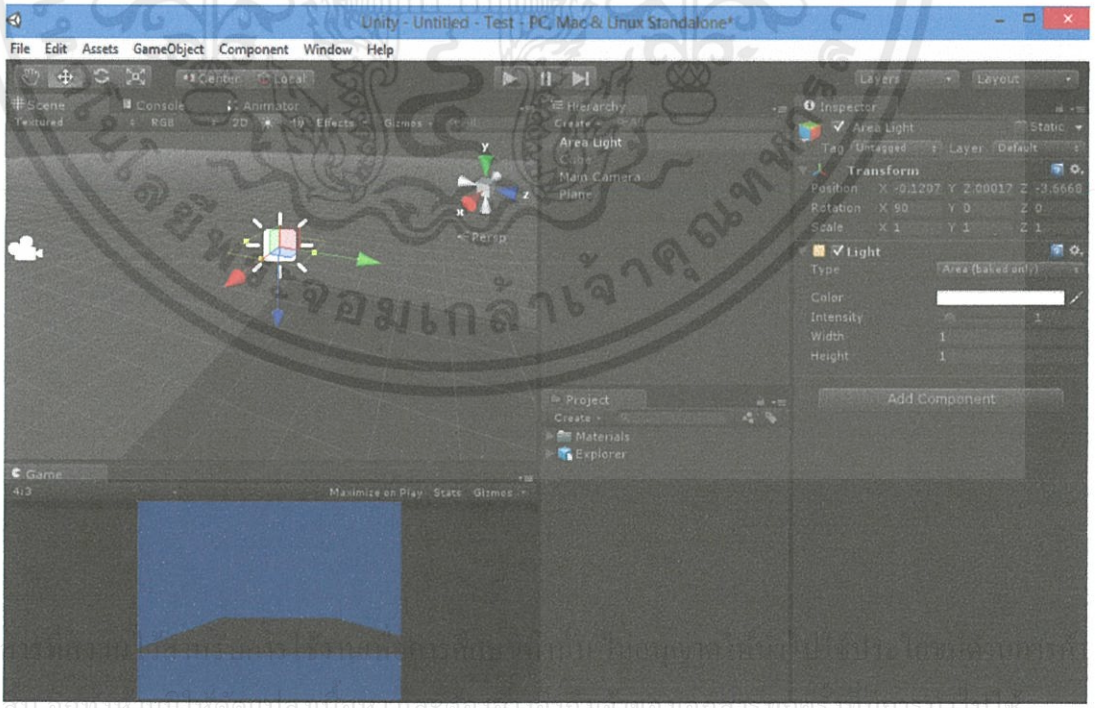
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น

17.3 การใส่แสงประเภท Spotlight การกำหนดค่าต่างๆจะคล้ายกับแสงประเภท Point Light แต่จะเพิ่มการกำหนดความกว้างของแสงที่คำสั่ง “Spot Angle”



รูปที่ ก-35 แสดงการใส่ Spotlight

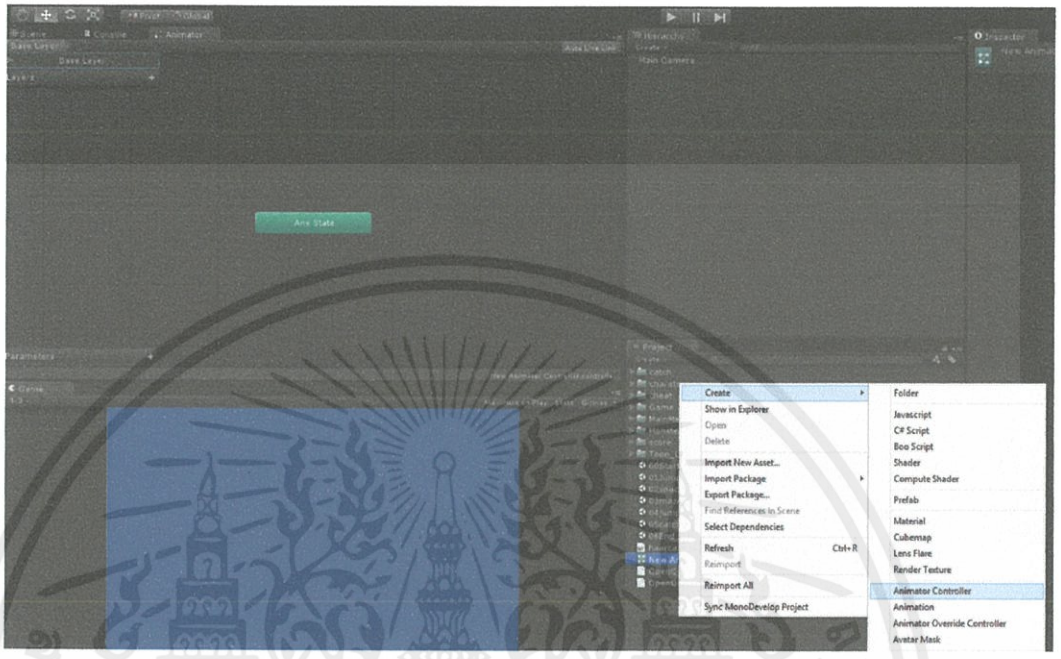
17.4 การแสงประเภท Area Light



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

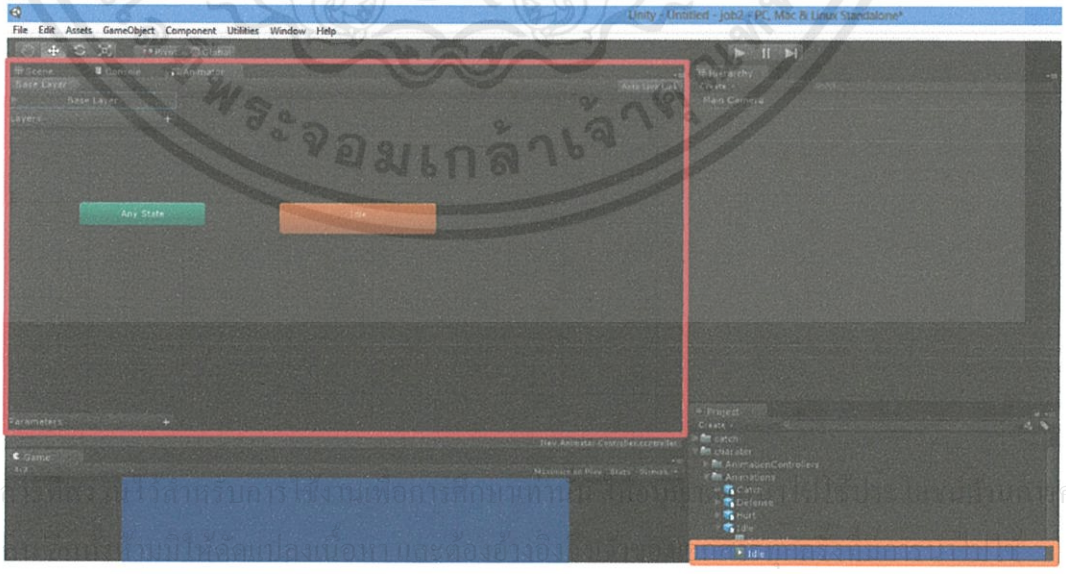
รูปที่ ก-36 แสดงการใส่ Area Light

18. การทำอนิเมชั่นของตัวละครเริ่มจากคลิกขวาที่ “Project” เลือก “Create” แล้วเลือกที่ “Animator Controller” จากนั้นเปิด Animator Controller ที่ได้สร้างขึ้นมา ซึ่งจะแสดงอยู่ในแถบ Animator



รูปที่ ก-37 การสร้าง Animator Controller

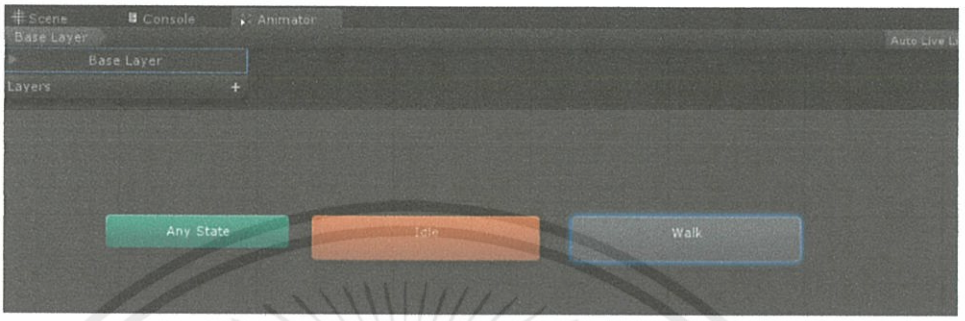
18.1การกำหนดท่าทางหรือการกระทำของตัวละคร สามารถทำได้โดยการลากไฟล์ .fbx ของตัวละครเข้ามาที่ “Animator” เช่น ต้องการให้ตัวละครทำท่าทางว่าง (Idle) เริ่มจากการลากไฟล์ท่าทางว่างเข้ามาที่ “Animator” ซึ่งท่าทางแรกที่ลากเข้ามาจะเป็นกล่องสี่เหลี่ยมสีส้ม และตัวละครจะแสดงท่าทางตามที่กำหนด



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น

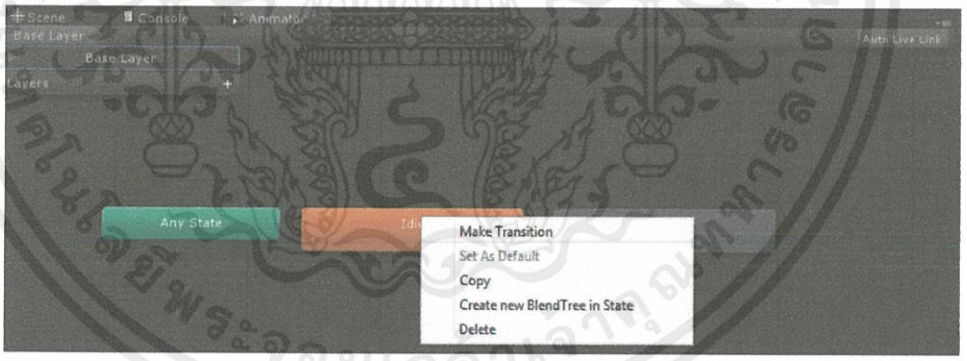
รูปที่ ก-38 การใส่ไฟล์ท่าทางการเคลื่อนไหวและการทำ Animator

18.2 ส่วนการกำหนดท่าทางอื่นๆเพิ่มเติม สามารถทำได้โดยการใส่ไฟล์ .fbx ของท่าทางอื่นเข้าไป เช่น การเดิน การวิ่ง เป็นต้น ตัวอย่างดังแสดงในภาพ คือถ้าต้องการให้ตัวละครจากสถานะอยู่นิ่งๆเปลี่ยนเป็นการเดิน การกำหนดท่าทางจะเริ่มจากการใส่ท่าทางนิ่ง (Idle) แล้วจากนั้นใส่ทางท่างการเดิน (Walk) ดังภาพ

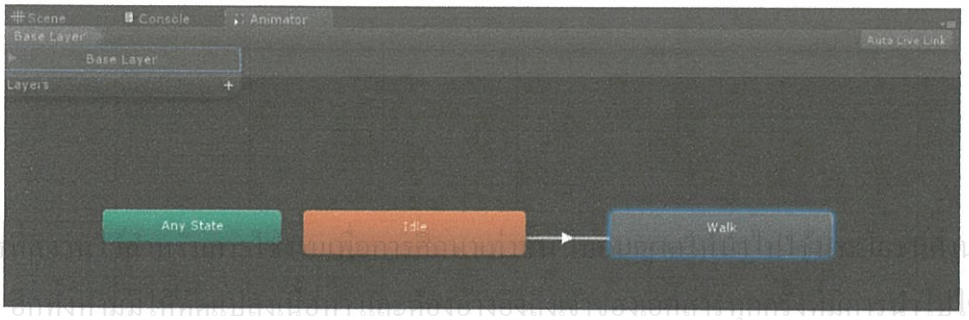


รูปที่ ก-39 การใส่ท่าทางของตัวละคร

18.3 การคุมควบการเปลี่ยนท่าทางสามารถทำได้หลายวิธี ไม่ว่าจะเป็นแบบแรก คือเมื่อทำท่าทางแรกเสร็จแล้วจึงทำท่าทางต่อไป หรือการกำหนดพารามิเตอร์ วิธีการเปลี่ยนท่าทางเมื่อทำท่าทางแรกเสร็จแล้วจึงทำท่าทางต่อไปนั้น ขั้นแรกให้คลิกขวาที่ท่าทางแรกจากนั้นเลือก “Make Transition” ดังรูป แล้วทำการคลิกท่าทางถัดไปจากนั้นเลือก Transition ที่สร้างขึ้นมา แล้วปรับ Condition เป็น “Exit time”

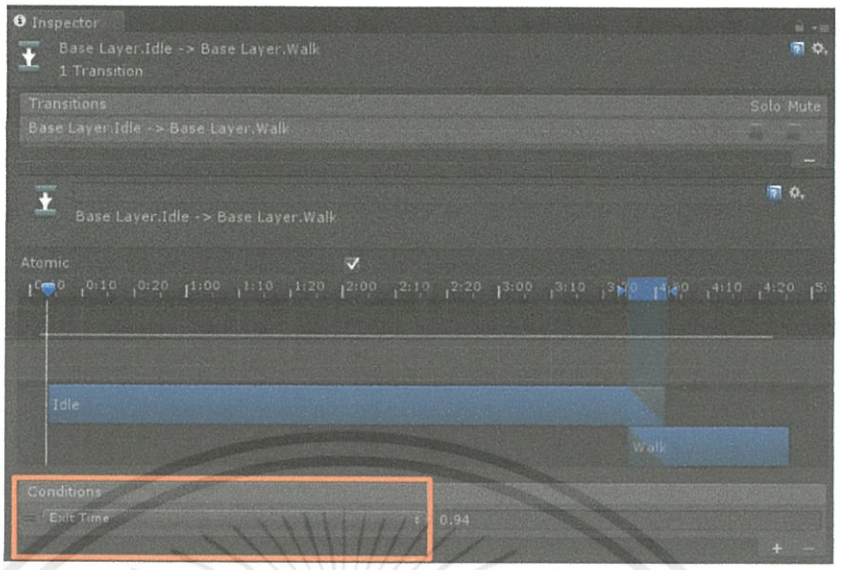


รูปที่ ก-40 การสร้าง Transition



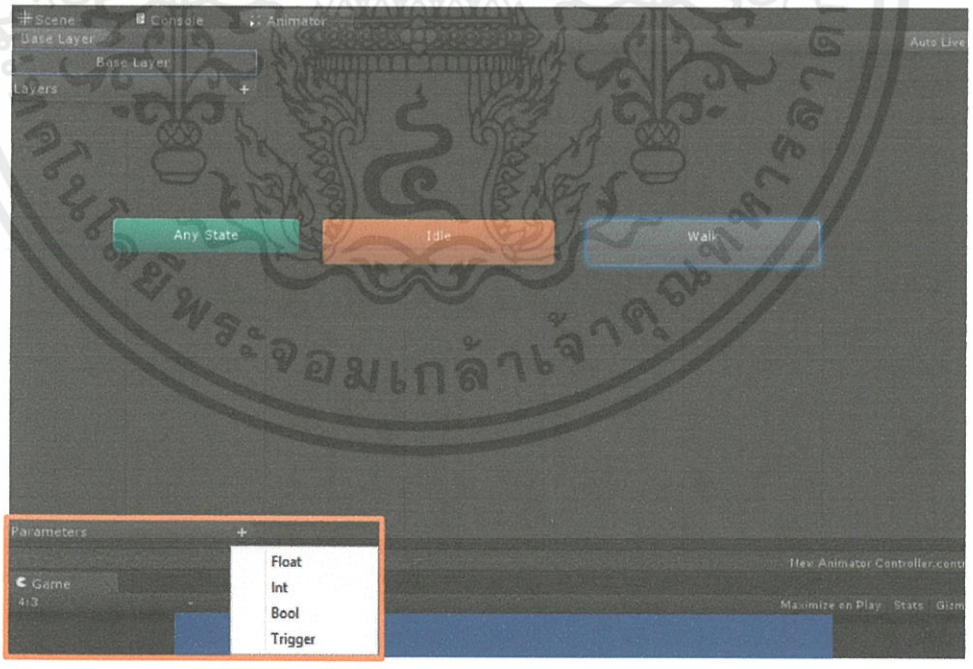
รูปที่ ก-41 ตัวอย่าง Transition

เอกสารนี้เป็นเอกสาร
ไม่ว่ากรณีใดๆทั้งสิ้น



รูปที่ ก-42 การกำหนด Condition

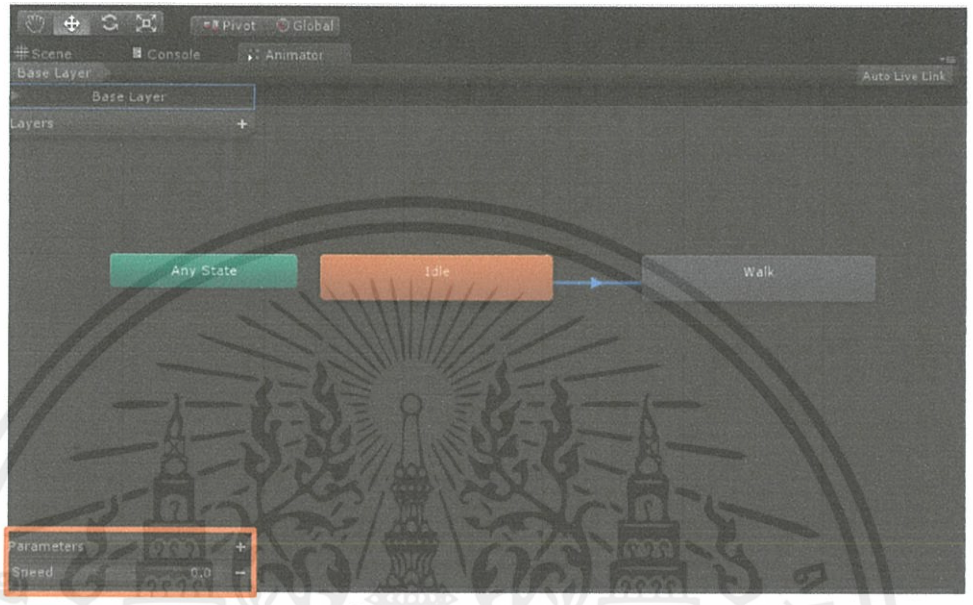
18.4วิธีการเปลี่ยนท่าทางโดยการใช้พารามิเตอร์ สามารถทำได้โดยการสร้าง Transition ดังที่กล่าวไว้ข้างต้น แล้วปรับ Condition เป็นพารามิเตอร์ที่สร้างขึ้น เช่นดังตัวอย่างที่แสดงในภาพคือ ขั้นแรกเพิ่มพารามิเตอร์สามารถทำได้โดยการกดเครื่องหมาย + ในแถบพารามิเตอร์ จากนั้นกำหนดประเภทของพารามิเตอร์เป็น Float และให้ชื่อว่า Speed



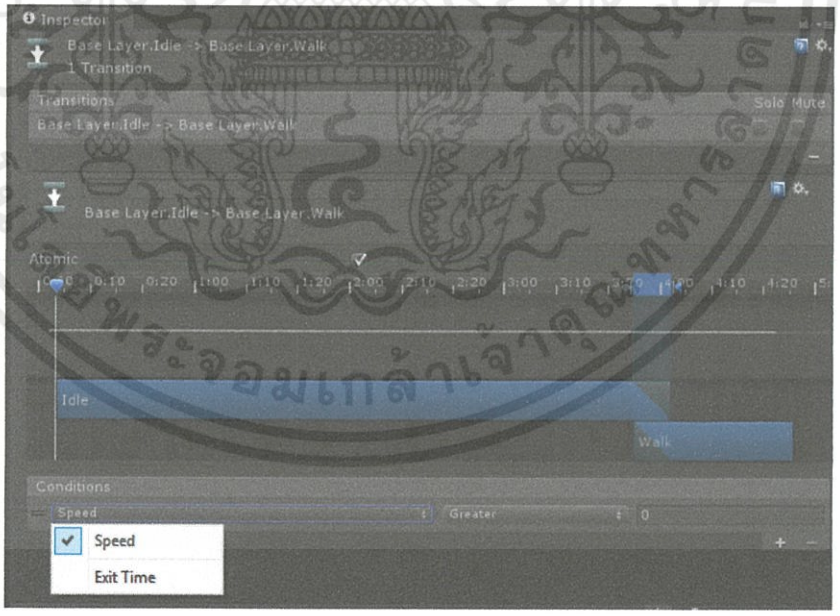
รูปที่ ก-43 ประเภทพารามิเตอร์ต่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แล้วปรับ Condition เป็น Speed ดังรูป ก-45 โดย Condition นี้จะมีการกำหนดเงื่อนไขอยู่ 2 แบบคือ Greater กับ less โดยที่ Greater คือ เมื่อพารามิเตอร์มีค่ามากกว่าที่กำหนด และ less คือ เมื่อพารามิเตอร์มีค่าน้อยกว่าที่กำหนด แล้วถ้าเงื่อนไงเป็นจริงจะทำการเปลี่ยนท่าทางของตัวละคร



รูปที่ ก-44 การสร้างพารามิเตอร์

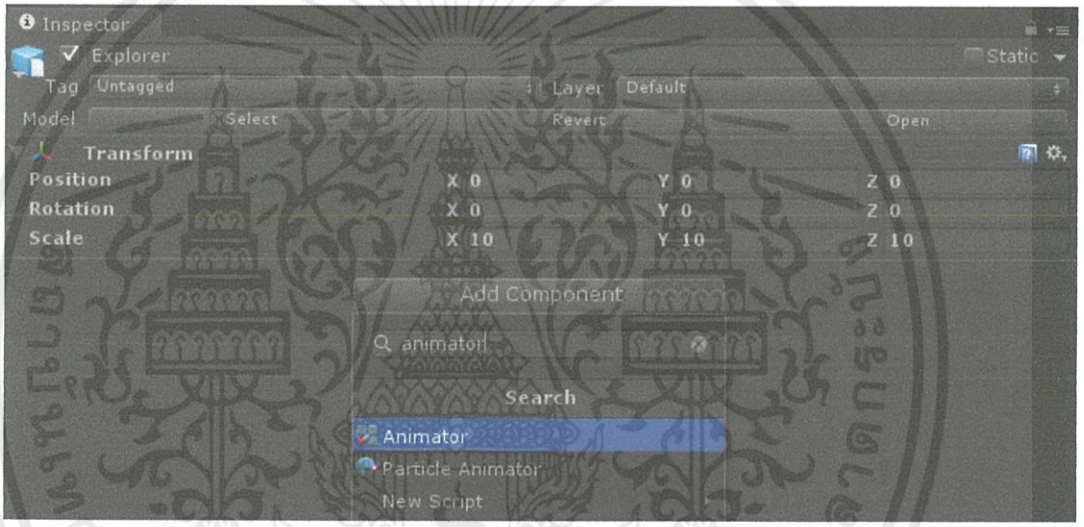


รูปที่ ก-45 กำหนด Condition ด้วยพารามิเตอร์

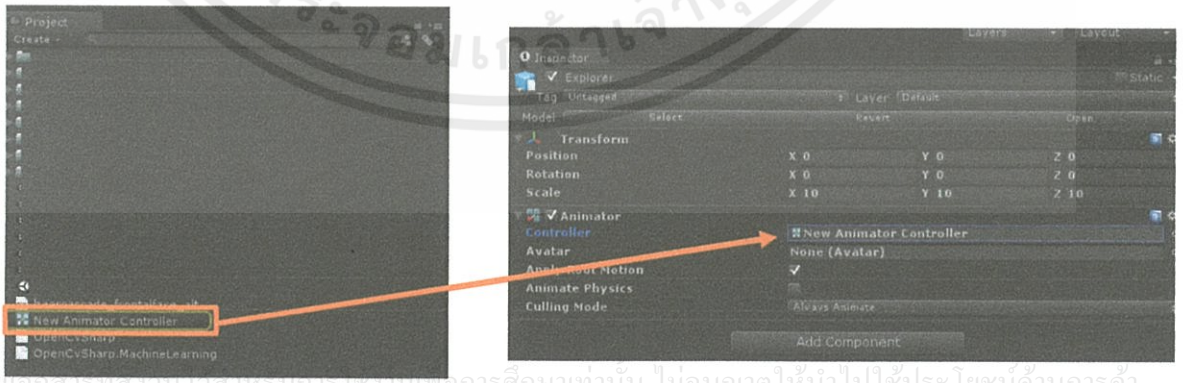
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ ก-46 เงื่อนไขของ Condition

18.5 การนำอนิเมชั่นไปใช้กับตัวละคร ทำได้โดยการ Add Component ที่ชื่อว่า “Animator” เข้าที่ตัวละคร ดังรูป ก-47 จากนั้นลาก Animator Controller ที่ได้สร้างขึ้นได้ที่ช่อง Controller ดังรูป ก-48



รูปที่ ก-47 การสร้าง Component ชื่อ Animatorที่ตัวละคร



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

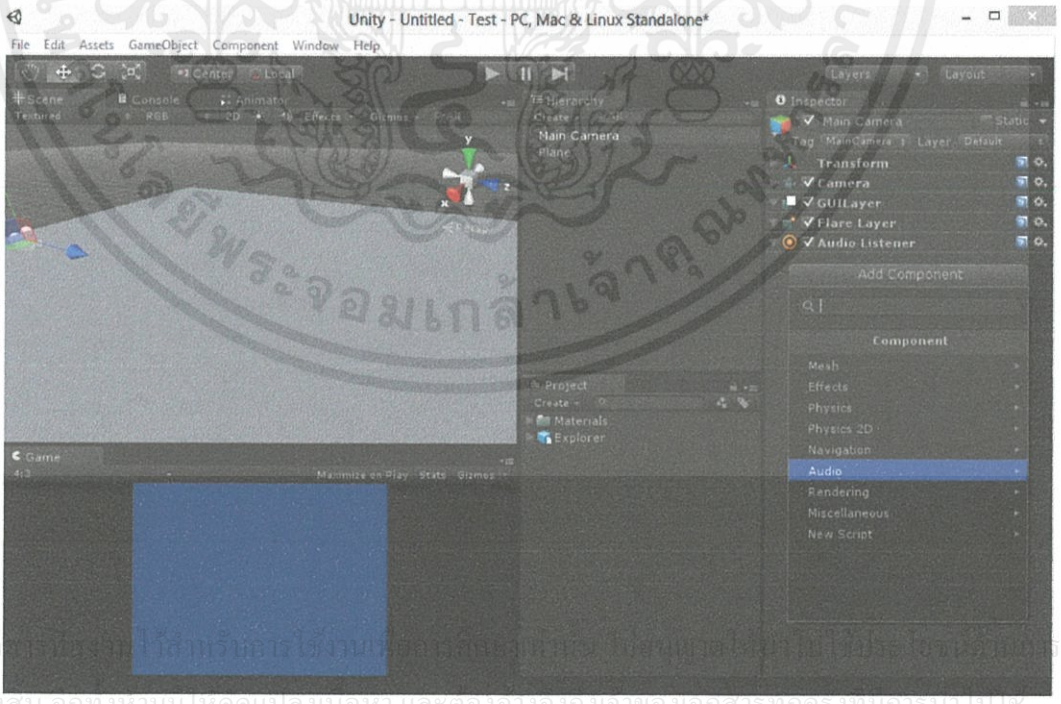
รูปที่ ก-48 การกำหนด Animator ที่ตัวละคร

18.6 การเปลี่ยนค่าพารามิเตอร์โดยใช้คำสั่งดังนี้ ขั้นแรกสร้างตัวแปรประเภท Animator จากนั้นทำการอ้างอิงกับ Component ที่เราใส่ให้ การเปลี่ยนแปลงค่าให้ใช้คำสั่ง SetFloat

```
using UnityEngine;
using System.Collections;

public class PlayerController : MonoBehaviour {

    private Animator anim;


    // Use this for initialization
    void Start () {
        anim = GetComponent<Animator>();
    }

    // Update is called once per frame
    void Update () {

        float h = Input.GetAxis("Horizontal");
        anim.SetFloat("Speed",h);

    }
```

รูปที่ ก-49 ตัวอย่างคำสั่ง

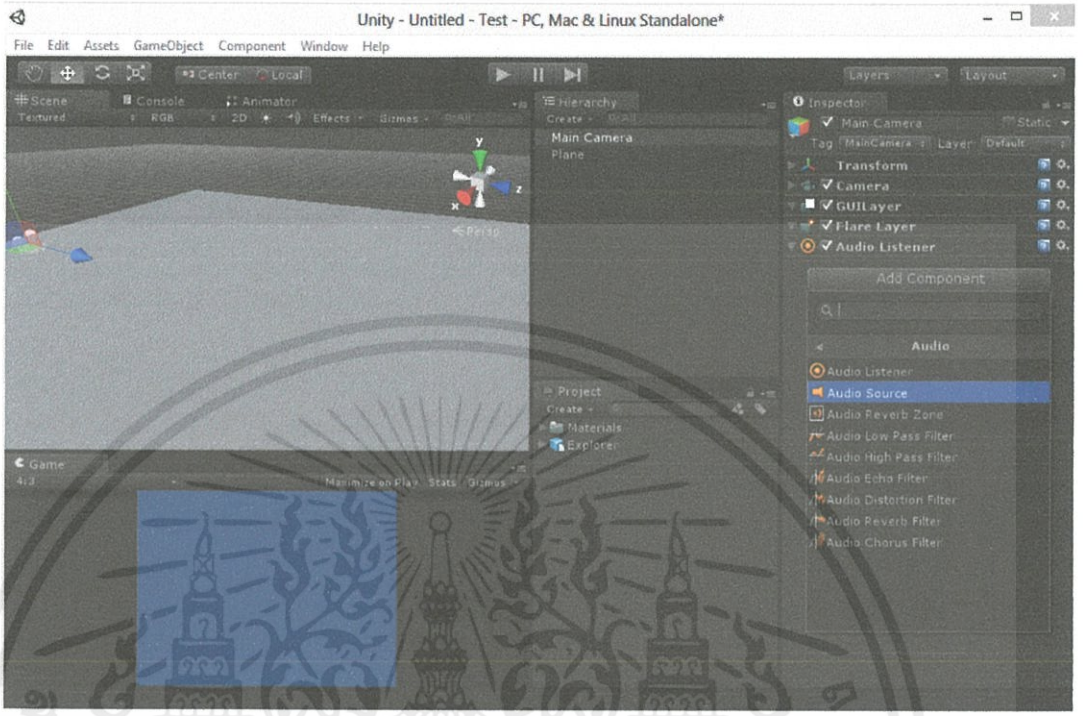
19. การใส่เสียงภายในเกม สามารถเริ่มต้นโดยการคลิกที่ “Add Component” จากนั้นเลือกที่ “Audio”



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

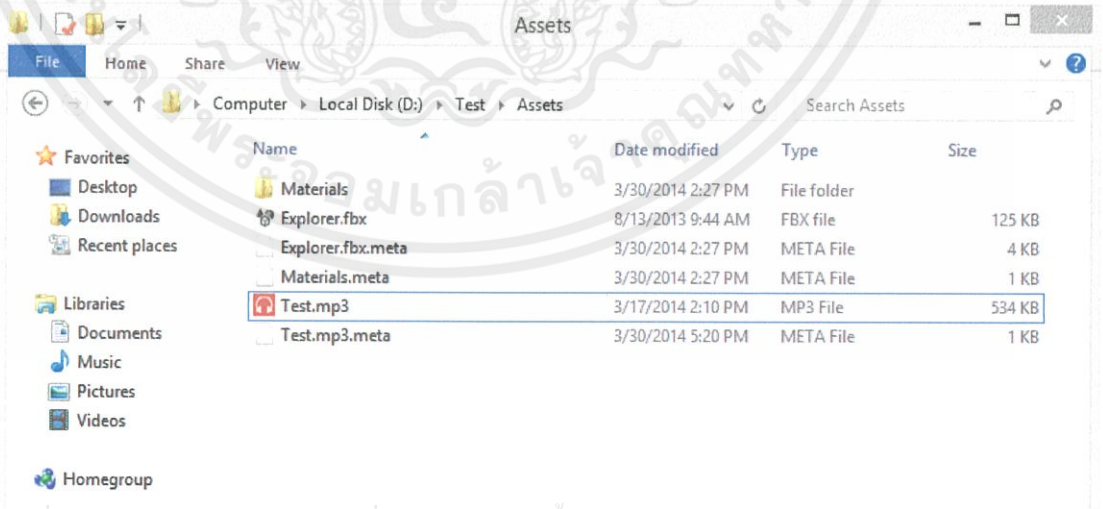
รูปที่ ก-50 แสดงการใส่เสียงภายในเกม

20. หลังจากคลิกเลือก “Audio” แล้ว ให้เลือกที่ “Audio Source”



รูปที่ ก-51 แสดงการเลือกประเภทเสียง

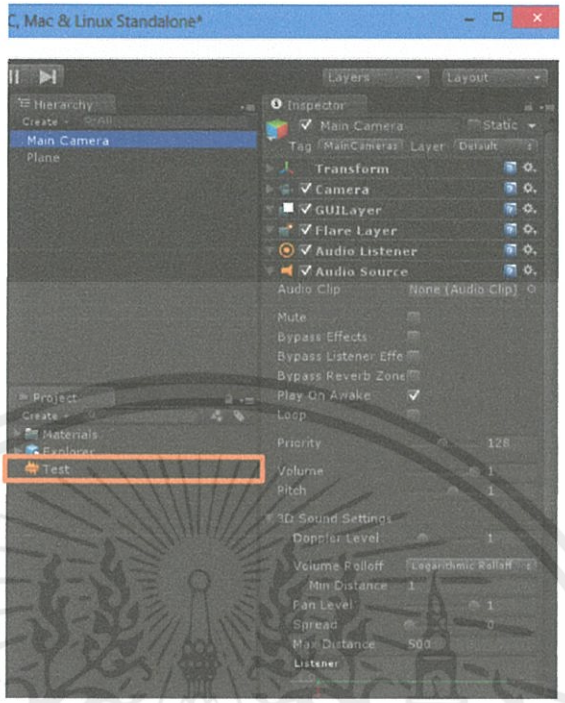
21. หลังจากเลือกประเภทแล้ว จะปรากฏหน้าต่าง Component ของ Audio Source ขึ้นมา หลังจากนั้นให้ทำการลากไฟล์เสียงที่ต้องการใช้ภายในเกม เข้าไปในโฟลเดอร์ “Assets” ของโปรเจคเกมที่สร้างขึ้นมา ซึ่งในที่นี้โปรเจคเกมมีชื่อว่า “Test”



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

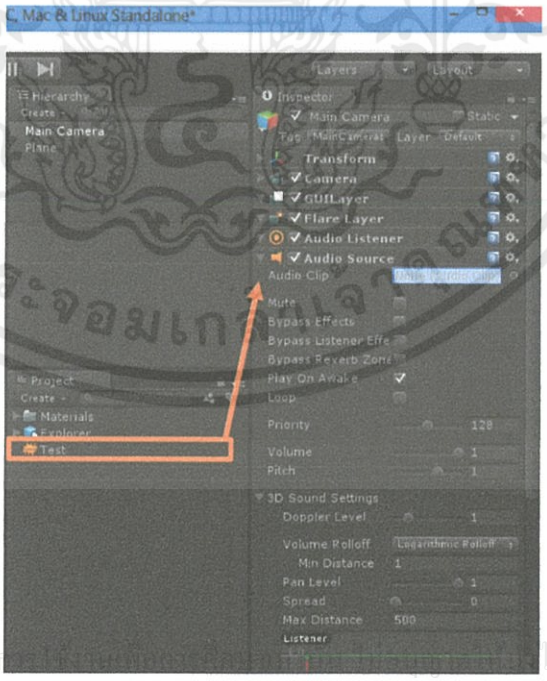
รูปที่ ก-52 แสดงการใส่ไฟล์เสียงในโฟลเดอร์

22. เมื่อทำการใส่ไฟล์เข้าไปในโปรเจคแล้ว จะปรากฏไฟล์ดังกล่าวที่แถบโปรเจค



รูปที่ ก-53 แสดงไฟล์เสียงภายในโปรเจค

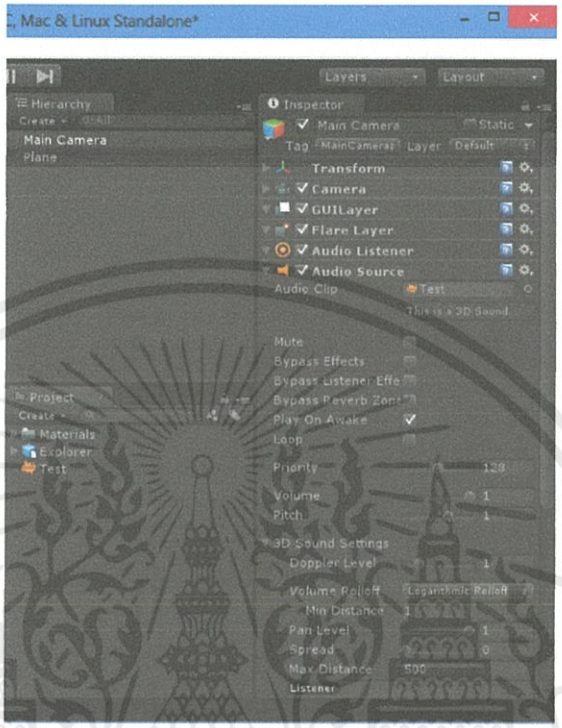
23. ซึ่งการนำไฟล์ที่ได้มานี้มาใช้ภายในเกม ทำได้โดยการคลิกที่ไฟล์แล้วลากไปที่ช่อง “Audio Clip”



รูปที่ ก-54 แสดงการใส่ไฟล์เสียงลงในเกม

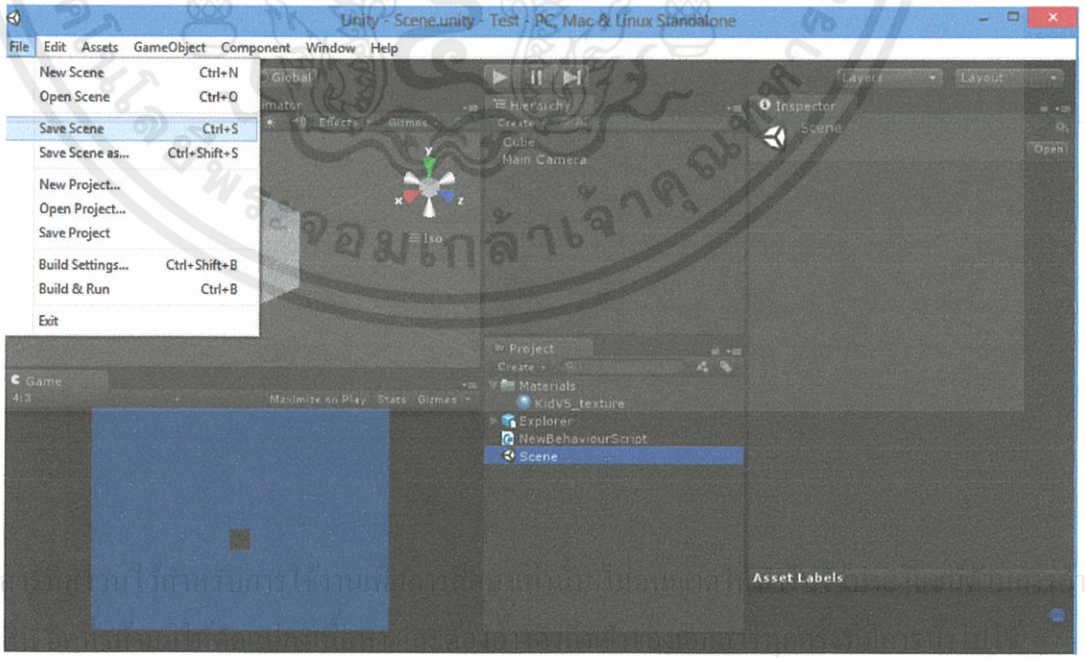
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแป ทุกครั้งที่มีการนำไปใช้

24. เมื่อใส่เสียงภายในเกมได้แล้ว ยังสามารถกำหนดค่าต่างๆของเสียงได้อีกด้วย เช่น Play On Awake หมายถึงการเล่นเสียงตั้งแต่เริ่มเกม หรือ Loop ซึ่งหมายถึงการเล่นเพลงวนซ้ำไปเรื่อยๆ เป็นต้น



รูปที่ ก-55 ส่วนประกอบของเสียง

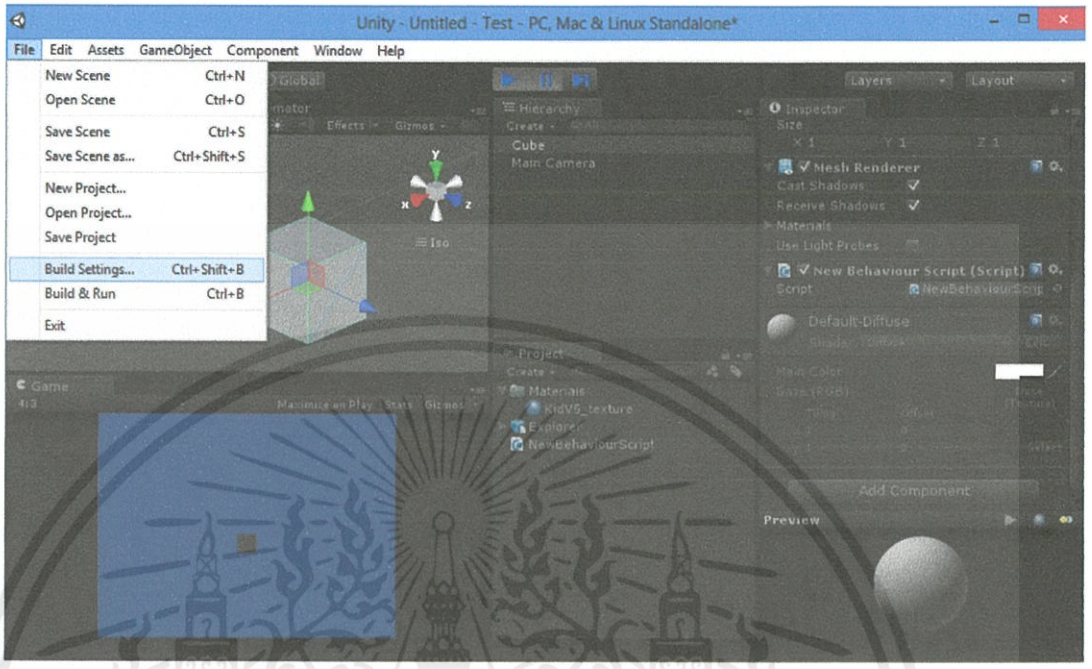
25. วิธีการบันทึกฉากเกมที่ได้สร้างขึ้น เริ่มต้นคลิกที่ “File” แล้วเลือกที่ “Save Scene”



รูปที่ ก-56 วิธีการบันทึกฉาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น

26. ส่วนขั้นตอนในการสร้างเกมเป็นไฟล์นามสกุล .exe สามารถทำได้โดยการไปที่ “File” จากนั้นเลือก “Build Setttings…”



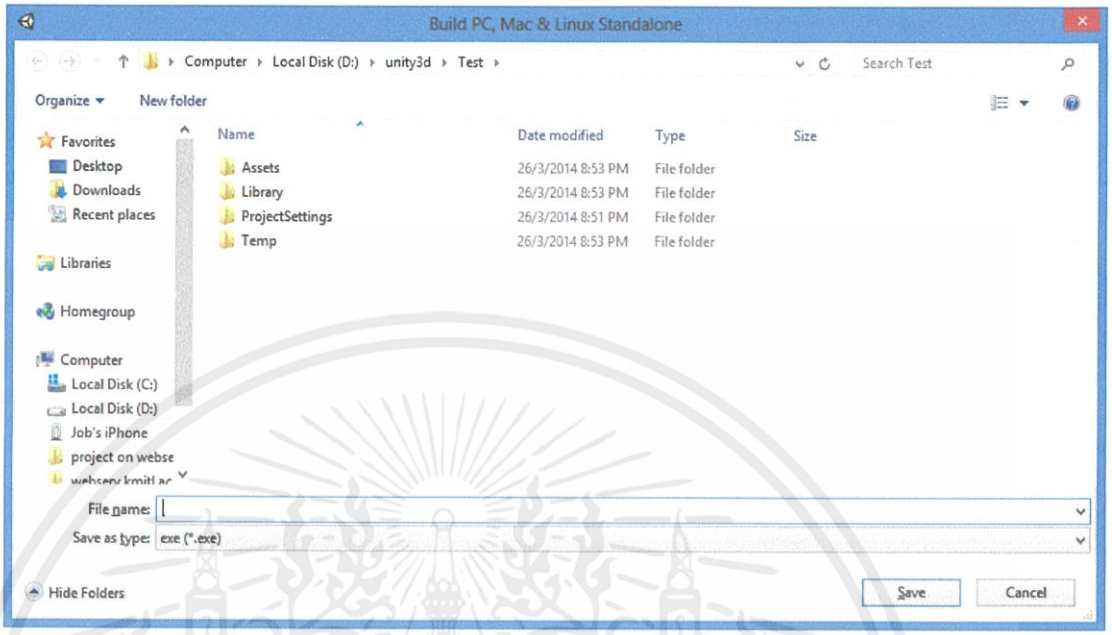
รูปที่ ก-57 วิธีการสร้างเกม

27. เลือกฉากเกมที่ต้องการจากงานที่สร้างขึ้นมา(ในที่นี้เลือก Scene.Unity) จากนั้นเลือกรูปแบบที่ต้องการสร้าง โดยจะมีให้เลือกอยู่ภายในช่อง “Platform” เช่น windows เป็นต้น หลังเลือกรูปแบบที่ต้องการได้แล้ว จากนั้นกด “Build” เพื่อสร้าง

รูปที่ ก-58 ตัวเลือกในการสร้าง

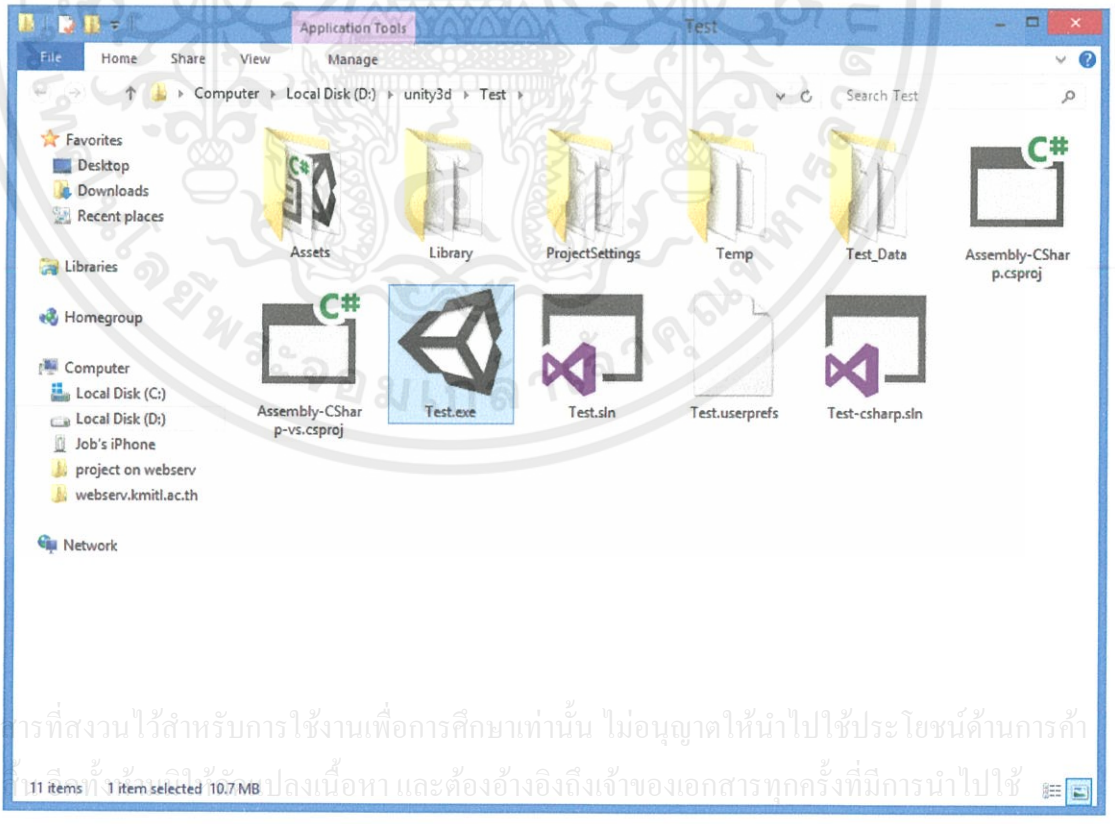
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับ ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัด ที่มีการนำไปใช้

28. เลือกไดเรกทอรี่ที่ต้องการบันทึก และตั้งชื่อไฟล์ที่สร้างขึ้น จากนั้นกด “Save” โปรแกรมจะทำการสร้างไฟล์เกมขึ้นมา



รูปที่ ก-59 แสดงไดเรกทอรี่ที่ต้องการบันทึก

29. ไฟล์ที่ได้จะอยู่ตามไดเรกทอรี่ที่ได้เลือกไว้



รูปที่ ก-60 ไฟล์ที่ได้จากการสร้าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้