

รายงานการวิจัย  
การพัฒนาระบบคอมพิวเตอร์สำหรับเด็กที่ป่วยเป็นโรคออทิสซึม  
โดยใช้เทคโนโลยีสื่อประสมและความจริงเสมือน  
Development of a computer system for training an autistic child using  
multimedia and virtual reality technologies.



รศ.ดร.นพพร โชติภักดิ์

นส.กรัญญา สิทธิสงวน

พ.ญ.พัชรภาพร เอื้อ

น.ส.อัจฉรา ณะชะบุญ

นายอนุสรณ์ บรรเทิง

นส.สุธาฉวี นพคุณธิ

ได้รับทุนสนับสนุนงานวิจัยจากเงินงบประมาณแผ่นดิน

ประจำปีงบประมาณ 2552

สำนักงานคณะกรรมการวิจัยแห่งชาติ (วช.)

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

## รายงานการวิจัย

การพัฒนาระบบคอมพิวเตอร์สำหรับเด็กที่ป่วยเป็นโรคออทิสซึม

โดยใช้เทคโนโลยีสื่อประสมและความจริงเสมือน

Development of a computer system for training an autistic child using  
multimedia and virtual reality technologies

รศ.ดร.นพพร โชติกกำธร

นส.กรัญญา สิริธสงวน

พ.ญ.พัชรพร เอ้า

น.ส.อัจฉรา เดชะบุญ

นายอนุสรณ์ บรรเทิง

นส.สุชาสินี นพฤทธิ

RCH

0A

46.9

V5

KA94

เลขหมู่..... 0-2

เลขทะเบียน..... 116090

วัน,เดือน,ปี..... 2 11A 2554

ได้รับทุนสนับสนุนงานวิจัยจากเงินงบประมาณแผ่นดิน

ประจำปีงบประมาณ 2552

สำนักงานคณะกรรมการวิจัยแห่งชาติ (วช.)

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มี

b. 16317671  
i.....

ชื่อ โครงการ: การพัฒนาระบบคอมพิวเตอร์สำหรับเด็กที่ป่วยเป็นโรคออทิสซึมโดยใช้เทคโนโลยีสื่อประสมและความจริงเสมือน

Development of a computer system for an autistic child using multimedia and virtual reality technologies

ได้รับทุนอุดหนุนการวิจัยจาก เงินงบประมาณแผ่นดิน สำนักงานคณะกรรมการวิจัยแห่งชาติ (วช.)

ประจำปี.....2552..... จำนวนเงิน..... 289,220 บาท.....

ระยะเวลาทำการวิจัย 3 ปี ตั้งแต่ ตุลาคม 2549 ถึง กันยายน 2552 (ระยะเวลาในการทำวิจัยที่ใช้จริง มีนาคม 2550 – กันยายน 2553 รวม 3 ปี 7 เดือน)

#### รายชื่อคณะกรรมการวิจัย

รศ.ดร.นพพร โชติกคำธร \*

นส.กรัญญา สิทธิสงวน \*

พ.ญ.พัชรพร เอ้า \*\*

น.ส.อัจฉรา เดชะบุญ \*\*

นายอนุสรณ์ บรรเทิง \*

นส.สุธาสินี นพฤทธิ์ \*

\* คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง กรุงเทพฯ

\*\* โรงพยาบาลเชียงรายประชานุเคราะห์ จังหวัดเชียงราย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## กิตติกรรมประกาศ

ขอขอบคุณโรงพยาบาลเชียงรายประชานุเคราะห์ เจ้าหน้าที่ในแผนกที่เกี่ยวข้อง และผู้ปกครองของเด็กออทิสซึม ที่ได้ให้ความร่วมมือและความอนุเคราะห์ในการนำระบบคอมพิวเตอร์ช่วยสอนที่พัฒนาขึ้นภายใต้โครงการนี้ ไปทำการทดสอบการใช้งาน ทำให้การวิจัยสามารถดำเนินไปได้ตามเป้าหมายที่กำหนด

คณะผู้วิจัย



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทคัดย่อ

รายงานฉบับนี้นำเสนอผลการวิจัยการพัฒนาระบบคอมพิวเตอร์สำหรับเด็กที่ป่วยเป็นโรคออทิสซึมโดยใช้เทคโนโลยีสื่อประสมและความจริงเสมือน ในงานวิจัยได้ทำการศึกษาและพัฒนาระบบคอมพิวเตอร์สำหรับเด็กออทิสติกในด้านต่าง ๆ อันได้แก่ การฝึกฝนความรู้พื้นฐานเรื่องสีและรูปร่าง การฝึกฝนการประสานการควบคุมกล้ามเนื้อมือกับการมองเห็น และการฝึกฝนการซื้อสินค้าจากร้านสะดวกซื้อในชีวิตประจำวัน ในส่วนของระบบคอมพิวเตอร์เพื่อการฝึกฝนความรู้พื้นฐานเรื่องสีและรูปร่าง ผู้วิจัยได้ประยุกต์ใช้ส่วนต่อประสานกับผู้ใช้แบบจับต้องได้ หรือ ทิวไอ (TUI – Tangible User Interface) โดยศึกษาประสิทธิภาพในแง่ของความยากง่ายในการใช้งาน และสัมฤทธิ์ผลในการเรียนรู้ เมื่อเปรียบเทียบกับการใช้ส่วนต่อประสานกับผู้ใช้แบบปกติซึ่งอาศัยเมาส์หรือจอสัมผัสเป็นอุปกรณ์เชื่อมต่อ ผลจากการศึกษาพบว่าการใช้ส่วนต่อประสานแบบจับต้องได้ ได้ผลดีกว่าส่วนต่อประสานแบบอื่น ๆ ทั้งในแง่ของความยากง่ายในการใช้งาน และในแง่ประสิทธิภาพในการเรียนรู้ นอกจากนี้ ผู้วิจัยยังได้พัฒนาโปรแกรมเกมตามมิติเพื่อช่วยในการฝึกฝนการประสานการควบคุมกล้ามเนื้อมือกับการมองเห็น โดยใช้อุปกรณ์พวงมาลัยอิเล็กทรอนิกส์ในการควบคุมเกม ซึ่งเป็นอีกลักษณะหนึ่งอุปกรณ์ส่วนต่อประสานแบบจับต้องได้

ในส่วนของพัฒนาระบบคอมพิวเตอร์เพื่อการฝึกฝนการใช้ชีวิตประจำวัน ได้มีการประยุกต์ใช้เทคโนโลยีความจริงเสมือน (Virtual Reality) และเทคโนโลยีผสมจริง (Mixed Reality) ในการพัฒนาระบบคอมพิวเตอร์จำลองสถานการณ์การซื้อสินค้าจากร้านขายของชำ ข้อดีของระบบดังกล่าวคือ ความสามารถในการปรับเปลี่ยนสินค้า และสภาพแวดล้อมจำลองในระบบ ให้มีความหลากหลาย เพื่อให้สามารถฝึกฝนการประยุกต์ใช้ทักษะที่เรียนในสถานการณ์ที่แตกต่างกันได้ โดยในการจำลองสภาพแวดล้อมภายในร้าน ในงานวิจัยนี้ได้อาศัยเทคนิคการสร้างภาพหลายมุมมองจากภาพถ่าย (Multi-view Image-based Rendering) เพื่อให้สามารถสร้างสภาพแวดล้อมที่ใกล้เคียงหรือเหมือนจริงได้จากภาพที่ถ่ายสถานที่จริงในมุมมองต่าง ๆ

## Abstract

This report describes results from a research project on development of a computer system for an autistic child using multimedia and virtual reality technologies. In this project, computer systems were developed for teaching and training autistic children on these skills: basic color and shape understanding, muscle control with eye-hand coordination, skill for purchasing goods from a convenient store in every-day life. In developing a computer system for basic color and shape understanding, use of a tangible user interface (TUI) was studied in terms of ease-of-use and learning efficacy. The system was compared with two similar computer-based training systems, one of which was based on a standard computer mouse while the other used a touch screen. It was found that the system based on TUI interface offered improved performance, in terms of both ease of use and learning efficacy. In addition, another computer-based training system was developed as a 3D computer game for training muscle control with eye-hand coordination skill. The latter system employed an electronic steering wheel joystick, which can be considered another form of tangible user interface.

The third computer-based training system, developed under this project, is a system for teaching every-day life skill. By applying virtual reality and mixed reality technologies, a simulation-based computer system was developed for teaching every-day life purchasing skill. The developed system offers flexibility in changing purchasing scenarios, each of which is defined by the type and number of goods to be purchased as well as the purchasing environment. This will give an opportunity for a trainee to learn how to adapt the skill for different situations. By simulating store environment using an image-based rendering technique, realistic simulation is achieved by scene reconstruction from multi-view images.

# สารบัญ

	หน้า
กิตติกรรมประกาศ.....	I
บทคัดย่อ.....	II
สารบัญ.....	IV
สารบัญตาราง.....	VI
สารบัญรูป.....	VII
บทที่ 1 บทนำ.....	1
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 วัตถุประสงค์ของการศึกษา.....	2
1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย.....	2
1.4 ขอบเขตของการวิจัย.....	3
1.5 ขั้นตอนการศึกษา.....	3
1.6 ประโยชน์ที่คาดว่าจะได้รับ.....	4
1.7 สรุปผลการดำเนินการ.....	4
บทที่ 2 การพัฒนาระบบคอมพิวเตอร์ช่วยสอนสำหรับเด็กออทิสซึม.....	6
2.1 แนวคิดเบื้องต้นในการออกแบบ.....	6
2.2 การออกแบบระบบคอมพิวเตอร์ช่วยสอนเรื่องรูปทรง.....	7
2.2.1 โปรแกรมเวอร์ชันที่ใช้ WIMP Interface.....	8
2.2.2 โปรแกรมเวอร์ชันที่ใช้ TUI Interface.....	8
บทที่ 3 การทดลองการใช้งานโปรแกรมฝึกทรงและสีและบทวิเคราะห์.....	12
3.1 ข้อมูลทั่วไปเกี่ยวกับการทดลอง.....	12
3.2 การทดลองที่ 1.....	13
3.2.1 ผู้ร่วมในการทดสอบ.....	13
3.2.2 การออกแบบการทดลอง.....	13
3.2.3 ผลการทดลองและประเมิน.....	14
3.3 การทดลองที่ 2.....	14
3.3.1 ผู้ร่วมในการทดสอบ.....	15
3.3.2 การออกแบบการทดลอง.....	15
3.3.3 ผลการทดลองและประเมิน.....	17

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ(ต่อ)

	หน้า
3.4 แนวทางการออกแบบโปรแกรมสำหรับเด็กก้อทิสซึ่ม.....	18
บทที่ 4 เกมฝึกการใช้มือโดยใช้จอยสติคส์แบบพวงมาลัย.....	19
4.1 หลักการทำงานของเบลนเดอร์เกมเอนจิน (Blender Game Engine) .....	19
4.2 การเชื่อมต่ออุปกรณ์.....	20
4.3 การเชื่อมต่อพวงมาลัยอิเล็กทรอนิกส์เข้ากับ โปรแกรมเบลนเดอร์ .....	20
4.4 เกม Happy Monkey .....	20
บทที่ 5 โปรแกรมจำลองสถานการณ์สำหรับเด็กก้อทิสซึ่ม .....	22
5.1 โครงสร้างของโปรแกรมจำลองสถานการณ์สำหรับเด็กก้อทิสซึ่ม.....	22
5.2 เครื่องมือในการพัฒนาโปรแกรม.....	27
5.3 การแสดงภาพของสภาพแวดล้อมภายนอกแบบสเตอริโอ .....	28
5.4 การแก้ปัญหาหระหะในการแสดงภาพของอุปกรณ์แสดงภาพแบบสวมศีรษะ.....	30
5.5 การซ้อรูปภาพหลังแบบบลูสกรีน (Blue Screen Matting).....	31
5.6 การสร้างภาพ 3 มิติโดยใช้ภาพถ่าย (Image-Based Rendering).....	32
5.7 เทคนิค Light Field เพื่อการสร้างภาพ 3 มิติจากภาพถ่าย.....	33
5.7.1 ความรู้พื้นฐานเกี่ยวกับ Light Field Rendering .....	33
5.7.2 การออกแบบและสร้างอุปกรณ์ในการถ่ายภาพ .....	34
5.7.3 โปรแกรมในการถ่ายภาพ .....	35
5.7.4 การปรับเทียบกล้อง (Camera Calibration) .....	36
5.7.5 การสร้างภาพใหม่ (Image Reconstruction).....	36
5.8 การหาตำแหน่งของผู้เล่น .....	39
5.9 การตรวจจับวัตถุ .....	40
5.10 การทดลอง .....	41
5.11 สรุปผล.....	44
บรรณานุกรม.....	45
ภาคผนวก .....	48

# สารบัญตาราง

ตารางที่	หน้า
3.1 จำนวนครั้งที่ผู้ร่วมการทดสอบต้องการความช่วยเหลือในการใช้งานจากนักกิจกรรมบำบัด – กรณีโปรแกรมเวอร์ชัน WIMP Interface .....	14
3.2 จำนวนครั้งที่ผู้ร่วมการทดสอบต้องการความช่วยเหลือในการใช้งานจากนักกิจกรรมบำบัด – กรณีโปรแกรมเวอร์ชัน TUI Interface.....	14
3.3 ผลคะแนน pre-test และ post-test สำหรับการฝึกฝนด้วยวิธีการที่ใช้ในปัจจุบัน.....	17
3.4 คะแนน pre-test และ post-test สำหรับการฝึกฝนด้วยระบบที่พัฒนาขึ้น .....	18
5.1 เฟรมเรตของโปรแกรม .....	42
5.2 ค่าตำแหน่งในแนวลึก .....	43
5.3 ผลการตรวจจับวัตถุ .....	43



# สารบัญรูป

รูปที่	หน้า
2.1 ตัวอย่างหน้าจอของ โปรแกรมเวอร์ชันที่ใช้ WIMP Interface.....	8
2.2 องค์ประกอบส่วนฮาร์ดแวร์ของ โปรแกรมเวอร์ชันที่ใช้ TUI Interface.....	9
2.3 ขั้นตอนการทำงานของ โปรแกรมเวอร์ชันที่ใช้ TUI Interface.....	9
2.4 ตัวอย่างภาพที่ถ่ายได้จากกล้องเว็บแคม เมื่อมีการนำบล็อกไม้มาวางไว้บนโต๊ะ.....	10
2.5 ตัวอย่างรูปแบบการจัดวางหลอดไฟ บนผิวบล็อกไม้.....	10
2.6 แสดงอุปกรณ์ TUI ที่สร้างขึ้นและใช้ในการทดลอง ก) บล็อกไม้ เพื่อใช้ในการฝึกเรื่องรูปทรง และสี ข) ของเล่น ใช้ในการฝึกฝนเรื่องสี .....	11
3.1 ภาพการฝึกฝนเรื่องรูปทรงและสีด้วยวิธีการในปัจจุบัน (ที่ไม่ได้อาศัยเครื่องคอมพิวเตอร์) .....	12
3.2 ผู้ร่วมทดสอบขณะกำลังใช้งาน โปรแกรม (ในภาพเป็นกรณีเวอร์ชัน WIMP Interface).....	13
3.3 แสดงหน้าจอที่โปรแกรมแสดงในการฝึกฝนระดับที่ 1 .....	16
3.4 หน้าจอที่โปรแกรมแสดงในการฝึกฝนระดับที่ 2 .....	16
4.1 แสดงการทำงานของ เบลนเดอร์เกมเอ็นจิน .....	19
4.2 แสดงการกำหนดเหตุการณ์ในเกมด้วยเบลนเดอร์เกมเอ็นจิน .....	19
4.3 ลักษณะของการเชื่อมต่อชุดอุปกรณ์เข้ากับเครื่องคอมพิวเตอร์ .....	20
4.4 การเรียกใช้อุปกรณ์ผ่าน Blender Game Engine.....	20
4.5 การกำหนด Sensor เพื่อรับค่าข้อมูลจากพวงมาลัย .....	20
4.6 การบังคับลิงให้ไปทางซ้ายและขวา.....	21
4.7 ภาพผลไม้ที่หล่นลงมา.....	21
4.8 ภาพเมื่อรับผลไม้ได้.....	21
5.1 โครงสร้างของโปรแกรมจำลองสถานการณ์สำหรับฝึกเด็กออทิสซึม .....	23
5.2 อุปกรณ์แสดงภาพแบบสวมศีรษะที่ติดตั้งกล้องเว็บแคมเพิ่มเติม .....	23
5.3 เครื่องหมายที่ถูกติดลงไปบนสิ่งของต่างๆ.....	23
5.4 ภาพแสดงจากสีฟ้าที่ใช้เป็นพื้นหลัง .....	24
5.5 แสดงการใช้งานจริง – มองจากด้านหลังผู้เล่น .....	24
5.6 แสดงการใช้งานจริง - มองจากด้านหน้าผู้เล่น .....	25
5.7 สิ่งที่ผู้เล่นมองเห็น — จากหลังสีฟ้าถูกซ่อนด้วยภาพจากหลังใหม่.....	25
5.8 แสดงการใช้งานจริง – ผู้เล่นวางสิ่งของที่จับ.....	26

## สารบัญรูป (ต่อ)

รูปที่	หน้า
5.9 แสดงการใช้งานจริง – ผู้เล่นวางจำนวนเงินที่ต้องชำระ.....	26
5.10 สิ่งที่คุณเล่นมองเห็น – โมเดลจากคอมพิวเตอร์ถูกวางลงบนวัตถุที่คุณเล่นวางบนโต๊ะ .....	26
5.11 การสร้างภาพแบบสเตอริโอ .....	30
5.12 การขยายภาพเฉพาะส่วน .....	31
5.13 Light Field Rendering .....	34
5.14 อุปกรณ์ในการถ่ายภาพ.....	35
5.15 การแบ่งการทำงานออกเป็นหลายๆ เซรด .....	35
5.16 ขั้นตอนการตรวจจับวัตถุ.....	41



# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

โรคออทิสซึม เป็นโรคทางจิตเวชโรคหนึ่งซึ่งเป็นผลมาจากความผิดปกติทางหน้าที่ของระบบประสาทบางส่วน สาเหตุที่แท้จริงยังไม่สามารถสรุปได้ แต่สามารถวินิจฉัยตั้งแต่ในวัยเด็ก เป็นกลุ่มอาการที่มีพัฒนาการล่าช้าและผิดปกติ ทางด้านพัฒนาการทางสังคม พัฒนาการด้านการสื่อความหมายและภาษา และ มีการกระทำและความสนใจซ้ำๆ ปัจจุบันพบว่ามีผู้ป่วยออทิสซึมอยู่มาก ซึ่งพบได้ 4-5 : 10,000 ตามเกณฑ์มาตรฐานการวินิจฉัย DSM IV และ 21:10,000 ตามการวินิจฉัยแบบ Autistic Spectrum ในประเทศไทยพบในเด็กชายมากกว่าเด็กหญิง 3.3 เท่า และมีโอกาสที่จะเกิดได้ในทุกเชื้อชาติ ถึงแม้ว่าในปัจจุบันยังไม่มียาหรือวิธีการรักษาโรคที่จำเพาะเจาะจงให้หายขาดได้ แต่ก็สามารถช่วยเหลือให้มีการพัฒนาการที่ดีขึ้น สามารถเรียนรู้ และ ใช้ชีวิตอยู่ร่วมกับสังคมได้ โดยเฉพาะการฝึกฝนทักษะทางด้านสังคม และการประยุกต์ใช้ทักษะในสถานการณ์ต่าง ๆ (generalization) ซึ่งเป็นความบกพร่องที่สำคัญของบุคคลที่ป่วยโรคออทิสซึม จึงมีความจำเป็นที่ผู้ป่วยต้องได้รับการฝึกฝนเป็นพิเศษ เพื่อให้พวกเขาดำรงชีวิตอยู่ในสังคมได้ใกล้เคียงปกติมากที่สุด

โดยทั่วไปการบำบัดจะใช้การฝึกฝนให้รู้จัก สี สิ่งของ ตัวเลข รูปทรง และนำไปประยุกต์ใช้กับสภาพแวดล้อมในปัจจุบัน รวมถึงการจำลองเหตุการณ์ หรือ สถานการณ์ทางสังคมต่างๆ เพื่อให้ผู้ป่วยได้ทดลองปฏิบัติ จนเกิดความคุ้นเคย และความชำนาญ แต่ก็ยังคงพบปัญหาอยู่มาก เนื่องจากผู้ป่วยยังไม่สามารถแยกแยะ และนำไปประยุกต์ใช้กับสถานการณ์จริงในปัจจุบันได้ รวมถึงปัญหาในเรื่องของเวลาที่ผู้ปกครองจะพาผู้ป่วยมารับการฝึกฝนอย่างสม่ำเสมอ อีกทั้งพื้นที่ที่มืออยู่อย่างจำกัด หรือแม้กระทั่งบุคคลากรที่มีอยู่อย่างจำกัด รวมถึงสถานการณ์สมมุติที่ไม่เหมือนจริงทำให้ผู้ป่วยไม่คุ้นเคย ยังผลให้เกิดอาการกลัวเมื่อต้องอยู่ร่วมกับผู้อื่นในสังคม

ในปัจจุบันได้มีผู้นำเทคโนโลยีคอมพิวเตอร์มาใช้ในวงการแพทย์เพื่อช่วยในการฝึกฝนและ บำบัด ผู้ป่วยรวมถึงผู้พิการ เพื่อให้การรักษามีประสิทธิภาพมากขึ้น สำหรับโรคออทิสซึมนั้น ได้มีผู้ที่ทำการวิจัยเพื่อ ใช้คอมพิวเตอร์ช่วยในการฝึกฝนมาพอสมควร หากแต่ส่วนใหญ่จะเป็นการศึกษาเกี่ยวกับเด็กในกลุ่มที่เรียกกันว่า High-functioning autism (HFA) เป็นหลัก ปัญหาหนึ่งที่พบในการใช้คอมพิวเตอร์กับเด็กออทิสติกในกลุ่ม Low-functioning autism (LFA) คือความยากง่ายในการใช้งาน และความสนใจหรือสมาธิที่มีกับบทเรียน ซึ่งในงานวิจัยนี้ ได้แก้ปัญหาดังกล่าวโดยการนำเทคโนโลยีสื่อประสม และส่วนต่อประสานกับผู้ใช้แบบจับต้องได้ มาประยุกต์ใช้

สำหรับการฝึกฝนการใช้ชีวิตประจำวัน ได้มีการนำเทคโนโลยีคอมพิวเตอร์ที่เรียกกันว่า เทคโนโลยีวีอาร์ เข้ามาช่วยในฝึกฝนและเรียนรู้ทักษะในด้านต่าง ๆ ของเด็กออทิสติก โดยใช้เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ต่างๆ เช่น อุปกรณ์แสดงภาพแบบสวมศีรษะ (Head Mounted Display) ซึ่งจะแสดงภาพจำลองสถานการณ์ต่างๆ แล้วให้ผู้ป่วยมีปฏิสัมพันธ์ตอบโต้ต่อสิ่งที่เห็นและได้ยินนั้น ภาพที่ใช้จำลองเหตุการณ์มักจะเป็นภาพนิ่งที่ถ่ายจากสถานที่จริง หรืออาจจะเป็นภาพ 3 มิติที่สังเคราะห์ด้วยคอมพิวเตอร์ ซึ่งต่างก็มีข้อจำกัด นั่นคือ หากเป็นภาพนิ่งสิ่งที่ผู้ป่วยมองเห็นจะถูกจำกัดด้วยมุมมองของภาพที่ถ่ายมาเท่านั้น ซึ่งแตกต่างจากสถานที่จริงที่สามารถมองได้รอบด้าน ในขณะที่การใช้ภาพ 3 มิติที่สังเคราะห์ด้วยคอมพิวเตอร์นั้น วิธีโดยทั่วไปในการสร้างภาพลักษณะนี้จะใช้วิธีการสร้างโมเดลที่ประกอบไปด้วยรูปหลายเหลี่ยม (Polygon) เพื่อใช้เป็นตัวแทนของวัตถุต่างๆ ในธรรมชาติ ซึ่งการสร้างภาพด้วยวิธีการนี้ยังมีข้อจำกัดหลายประการไม่ว่าจะเป็น ในด้านความสมจริงของภาพ ซึ่งหากต้องการให้สร้างภาพได้ในเวลาจริง (Real-Time) จำเป็นจะต้องลดความสมจริงของภาพลง และการสร้างฉากใหม่ ๆ ขึ้นมานั้นจะมีความยุ่งยากและใช้เวลานาน ทำให้ผู้ป่วยจะยังไม่สามารถปรับตัวเข้ากับสถานการณ์จริงได้ เพราะสิ่งที่เห็นแตกต่างไปจากโปรแกรมจำลองสถานการณ์จำลองที่ได้ฝึกฝนมา ตลอดจนข้อจำกัดที่ใช้ฝึกฝนได้เพียงในบางสถานการณ์ ซึ่งในงานวิจัยนี้ได้ทำการแก้ไขปัญหาดังกล่าว โดยการนำเทคโนโลยีผสมจริง (Mixed reality) และเทคนิคการสร้างภาพหลายมุมมองจากภาพถ่าย (Multi-view Image-based Rendering) มาประยุกต์ใช้ร่วมกับเทคนิคความจริงเสมือนเดิม เพื่อให้สามารถสร้างหรือจำลองสภาพแวดล้อมต่าง ๆ ได้ง่ายขึ้นและเหมือนจริง

## 1.2 วัตถุประสงค์ของการศึกษา

- 1) เพื่อพัฒนาเทคนิคการเชื่อมต่อ รูปแบบหน้าจอ และการปฏิสัมพันธ์ระหว่างคอมพิวเตอร์กับผู้ใช้ (Human Computer Interface) ที่เหมาะสมกับเด็กที่ป่วยเป็นโรคออทิสซึม
- 2) เพื่อพัฒนาโปรแกรมช่วยสอนทักษะพื้นฐานในการจำแนกวัตถุ รูปทรง สี ตัวเลข ฯลฯ ที่จำเป็นต่อการดำเนินชีวิตและการเล่าเรียน โดยอาศัยเทคโนโลยีสื่อประสม
- 3) เพื่อพัฒนาระบบจำลองสถานการณ์ด้วยเทคโนโลยีวีอาร์และคอมพิวเตอร์กราฟฟิกเพื่อการฝึกฝนการใช้ชีวิตประจำวันในสถานการณ์ต่าง ๆ

## 1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการวิจัย

ในงานวิจัยนี้ เป็นการศึกษาการนำเทคโนโลยีสื่อประสม วีอาร์และเออาร์ เข้ามาบำบัดอาการของโรคออทิสซึมให้มีประสิทธิภาพมากขึ้น โดยเน้นการฝึกทักษะทางสังคม และการประยุกต์ใช้ทักษะในสถานการณ์ต่าง ๆ ที่จะทำให้ผู้ป่วยคุ้นเคยกับการใช้ชีวิตประจำวัน โดยจะแบ่งการศึกษาและพัฒนาออกเป็น 2 ส่วน โดยส่วนแรกจะเป็นการพัฒนาโปรแกรมและระบบคอมพิวเตอร์เพื่อช่วยในการฝึกทักษะขั้นเริ่มต้น ซึ่งได้แก่ การรู้จักสี และรูปทรง พื้นฐาน ทำให้

สามารถรับรู้ แยกแยะ วัตถุต่างๆ ได้ถูกต้อง นอกจากนี้ในงานวิจัยจะได้ทำการพัฒนาเทคนิคการปฏิสัมพันธ์ระหว่างผู้ใช้กับเครื่องคอมพิวเตอร์ที่เหมาะสมกับผู้ป่วยโรคออทิสซึม เพื่อแก้ปัญหาสมาธิสั้น และพฤติกรรมอื่น ๆ อันเป็นลักษณะของผู้ป่วยเป็นโรคดังกล่าว ซึ่งเป็นอุปสรรคต่อการใช้งานเครื่องคอมพิวเตอร์ในการฝึกทักษะข้างต้น โดยจะทำการพัฒนาและประยุกต์ใช้แนวคิดของส่วนต่อประสานกับผู้ใช้แบบจับต้องได้ในงานวิจัยนี้

ในส่วนของ 2 ของการวิจัย จะเป็นการพัฒนาระบบคอมพิวเตอร์โดยอาศัยเทคโนโลยีวีอาร์ และเออาร์ เพื่อใช้ฝึกทักษะด้านสังคม (Social life) โดยการจำลองเหตุการณ์ หรือ สถานการณ์ต่าง ๆ กัน เพื่อให้ผู้ป่วยได้เกิดความคุ้นเคย และความชำนาญ จนสามารถที่จะไปอยู่ร่วมกับสังคมได้ใกล้เคียงปกติมากที่สุด ซึ่งจะลดอุปสรรคติดตามตัวให้น้อยลง โดยการใช้เทคโนโลยีผสมจริง (Mixed reality) มาประกอบกับเทคโนโลยีความจริงเสมือน

#### 1.4 ขอบเขตของการวิจัย

- 1) พัฒนาโปรแกรมและระบบคอมพิวเตอร์สำหรับการฝึกฝนทักษะพื้นฐานในการดำรงชีวิตประจำวันใน 2 ด้าน ได้แก่ ในการจำแนกวัตถุ รูปทรง สี ตัวเลข ฯลฯ และทักษะตลอดจนความสามารถในการปรับตัวให้สามารถทำกิจกรรมพื้นฐานได้ในสภาพแวดล้อมที่แตกต่างกันในชีวิตประจำวัน
- 2) การทดลองในงานวิจัยกระทำภายใต้สภาพแวดล้อมที่มีการควบคุม (laboratory experiment)
- 3) กลุ่มเป้าหมายของงานวิจัย ได้แก่ เด็กที่มีอายุระหว่าง 4-12 ปี โดยศึกษาทั้งกลุ่มที่มีระดับของการเจ็บป่วยในขั้นที่ต้องฝึกฝนทักษะพื้นฐาน และขั้นที่ใช้ชีวิตภายใต้สภาพแวดล้อมที่คุ้นเคยได้ (high function) แต่ยังคงฝึกทักษะการใช้ชีวิตในสังคม
- 4) งานวิจัยมุ่งเน้นประสิทธิภาพในการใช้งานเป็นอันดับแรก โดยคำนึงถึงความคุ้มค่าเชิงธุรกิจในการนำไปใช้งานเป็นปัจจัยประกอบ

#### 1.5 ขั้นตอนการศึกษา

- 1) ทบทวนเอกสารที่เกี่ยวข้อง (literature survey)
- 2) ทำการพัฒนาโปรแกรมช่วยสอนทักษะพื้นฐานในการจำแนกวัตถุ รูปทรง สี ตัวเลข ฯลฯ ที่จำเป็นต่อการดำเนินชีวิตและการเล่าเรียน และทำการทดลองประเมินผล
- 3) ทำการพัฒนาเทคนิคการเชื่อมต่อระหว่างคอมพิวเตอร์กับผู้ใช้ โดยอาศัยเทคโนโลยีสื่อประสมสำหรับเด็กออทิสซึม เพื่อการติดต่อและสั่งงานคอมพิวเตอร์ และทำการทดลองประเมินประสิทธิภาพตามหลักการของการออกแบบ HCI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 4) ทำการพัฒนาระบบคอมพิวเตอร์เพื่อการฝึกฝนการใช้ชีวิตประจำวันในสถานการณ์ต่าง ๆ โดยอาศัยเทคโนโลยีความจริงเสมือน และเทคโนโลยีผสมจริง

## 1.6 ประโยชน์ที่คาดว่าจะได้รับ

- 1) เป็นองค์ความรู้ในการวิจัยต่อไป
- 2) เป็นประโยชน์ต่อประชากรกลุ่มเป้าหมาย โดยระบบที่พัฒนาขึ้นบางส่วนสามารถจะนำไปใช้งานได้จริง

หน่วยงานที่สามารถนำผลการวิจัยไปใช้ประโยชน์ได้แก่ กระทรวงสาธารณสุข โรงพยาบาลหรือสถานพยาบาลที่มีศูนย์บำบัดผู้ป่วยเป็นโรคออทิสซึม ตลอดจนผู้ปกครองของเด็กที่ป่วยเป็นโรคดังกล่าว

## 1.7 สรุปผลการดำเนินการ

จากการที่โครงการนี้ ได้ดำเนินการมาเป็นระยะเวลา 3 ปี ครั้ง สรุปผลการดำเนินการได้ดังนี้

- 1) ได้ทำการศึกษาและพัฒนาเทคนิคการเชื่อมต่อกับผู้ใช้ โดยอาศัยหลักการของส่วนต่อประสานกับผู้ใช้แบบจับต้องได้
- 2) ได้ทำการพัฒนาระบบคอมพิวเตอร์ช่วยสอนเด็กออทิสซึมในเรื่องสี และรูปทรง โดยอาศัยหลักการส่วนต่อประสานกับผู้ใช้แบบจับต้องได้
- 3) ได้นำผลจากการศึกษาการใช้งานระบบคอมพิวเตอร์ช่วยสอนเด็กออทิสซึมในเรื่องสี และรูปทรงด้วยหลักการส่วนต่อประสานกับผู้ใช้แบบจับต้องได้ มาวิเคราะห์และสรุปเป็นแนวทางหรือแนวปฏิบัติที่ดีในการออกแบบ (Design Guidelines)
- 4) ได้ทำการทดสอบระบบตามข้อ 2 กับผู้ใช้งานหนึ่ง โดยผลทดลองและวิจัยได้รับการเผยแพร่ในเอกสารการวิจัยระดับนานาชาติ
- 5) ได้ทำการพัฒนาแบบคอมพิวเตอร์ประเภทเกมสามมิติ โดยใช้อุปกรณ์พวงมาลัยอิเล็กทรอนิกส์ เพื่อช่วยในการฝึกฝนทักษะการบังคับมือ และการประสานระหว่างมือกับการมองเห็น
- 6) ได้ทำการพัฒนาแบบจำลองสถานการณ์การใช้ชีวิตประจำวัน (การซื้อสินค้าจากร้านค้าของเบ็ดเตล็ด) โดยอาศัยเทคโนโลยีความจริงเสมือน เทคโนโลยีผสมจริง และการสร้างภาพสามมิติหลายมุมมองจากภาพถ่าย ประกอบกัน

ซึ่งผลงานโดยรวมตลอดระยะเวลาที่ผ่านมา สอดคล้องและครอบคลุมขอบเขตการวิจัย  
ตามที่ได้ระบุไปในข้อเสนอ โครงการวิจัย และมีงานเพิ่มจากที่ไม่ได้ระบุไว้บางส่วน เช่นการพัฒนา  
โปรแกรมสามมิติเพื่อการฝึกฝนทักษะการบังคับมือ เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

# การพัฒนาระบบคอมพิวเตอร์ช่วยสอนสำหรับเด็กออทิสซึม

ในบทนี้ จะได้กล่าวถึงแนวคิดของการพัฒนาระบบคอมพิวเตอร์ช่วยสอนสำหรับเด็กออทิสติก การออกแบบในส่วนต่าง ๆ และเทคนิคที่ใช้

### 2.1 แนวคิดเบื้องต้นในการออกแบบ

โดยทั่วไปพบว่าเด็กออทิสซึมส่วนใหญ่ชอบการเล่นหรือใช้คอมพิวเตอร์ ทั้งนี้ส่วนหนึ่งมีสาเหตุมาจากการที่เด็กออทิสซึมไม่ชอบที่จะสื่อสารกับคนด้วยตนเอง อันอาจเป็นผลมาจากการที่การสื่อสารระหว่างมนุษย์มีความซับซ้อนและเกี่ยวข้องกับการรับข่าวสารข้อมูลผ่านโสตประสาทหลาย ๆ ทางพร้อม ๆ อีกทั้งโปรแกรมคอมพิวเตอร์ก็มีสิ่งดึงดูดใจได้ ทำให้ลบลบปัญหาอันเกิดจากลักษณะสมาธิสั้นของเด็กออทิสซึมจำนวนหนึ่งได้ อย่างไรก็ดี เนื่องจากเด็กออทิสซึมเป็นจำนวนมากมีปัญหาเรื่องการประสานการทำงานระหว่างมือกับตา (eye-hand coordinate) ตลอดจนบางส่วนมีความสามารถที่จำกัดในการทำความเข้าใจความสัมพันธ์ระหว่างการใช้อุปกรณ์เชื่อมต่อกับคอมพิวเตอร์ที่ใช้โดยทั่วไปเป็น เม้าส์หรือคีย์บอร์ด กับผลที่ได้จากการควบคุม ทำให้ในงานวิจัยนี้ได้ทำการศึกษาการพัฒนาระบบช่วยสอนโดยอาศัยรูปแบบการเชื่อมต่อระหว่างคอมพิวเตอร์กับผู้ใช้ที่เรียกกันว่าส่วนต่อประสานแบบจับต้องได้ (Tangible User Interface - TUI) เพื่อลดหรือแก้ไขปัญหาดังกล่าว

TUI เป็นแนวคิดของการออกแบบการเชื่อมต่อระหว่างผู้ใช้กับคอมพิวเตอร์ (human-user interaction) ที่เกิดขึ้นเมื่อไม่กี่ปีที่ผ่านมา โดยหลักสำคัญของแนวคิดดังกล่าว คือการแทนบางส่วนของหรือทั้งหมดของ virtual object หรือสิ่งที่ปรากฏอยู่ในคอมพิวเตอร์ (ซึ่งไม่สามารถจับต้องได้โดยตรง) ด้วยอุปกรณ์หรือวัตถุภายนอกของระบบคอมพิวเตอร์ (คือเป็นวัตถุในโลกจริง) ทำให้ผู้ใช้สามารถจับต้องได้ โดยเชื่อว่าการออกแบบการเชื่อมต่อในลักษณะดังกล่าว จะทำให้การเรียนรู้และใช้งานคอมพิวเตอร์สามารถกระทำได้ง่ายขึ้น

ในงานวิจัยนี้ ได้ทำการออกแบบและพัฒนาระบบคอมพิวเตอร์ช่วยสอนโดยใช้ TUI เพื่อวัตถุประสงค์ในการสนับสนุนหรือช่วยหรือนักกิจกรรมบำบัด (occupational therapist) ในการสอนหรือฝึกฝนทักษะเบื้องต้นแก่เด็กออทิสซึม ให้สามารถเรียนรู้หรืออย่างมีประสิทธิภาพมากขึ้น ซึ่งในการฝึกฝนในรูปแบบที่ใช้อยู่ในปัจจุบัน ต้องการเวลาและจำนวนของนักกิจกรรมบำบัดที่ค่อนข้างมาก เพื่อให้การฝึกเป็นไปอย่างได้ผล ทั้งนี้ส่วนหนึ่งมีสาเหตุมาจากพฤติกรรมก้าวร้าวของเด็กทำให้มักจะต่อต้านการเรียนรู้กับนักกิจกรรมบำบัด ที่มักจะเกิดจากการถูกบังคับให้เรียน ดังนั้น

การใช้คอมพิวเตอร์มาช่วยในการฝึกฝน จำเป็นต้องออกแบบเพื่อให้มีสิ่งดึงดูดใจให้ผู้เรียนมีความพอใจและสนใจในการเรียน อันจะทำให้สัมฤทธิ์ผลของการเรียนดีขึ้น ปัจจัยที่มีผลต่อการบรรลุเป้าหมายดังกล่าว มีอาทิ เช่น ความยากง่ายในการใช้งาน ความน่าสนใจหรือดึงดูดใจให้ใช้ของระบบ เป็นต้น

ดังที่ได้กล่าวไปแล้วว่า การออกแบบการเชื่อมต่อโดยอาศัย TUI มีข้อดีประการสำคัญ คือ ทำให้การเรียนรู้ในการใช้งานง่าย นอกจากนี้ การที่อุปกรณ์เชื่อมต่อ TUI สามารถสัมผัส จับต้องได้น่าจะช่วยในการพัฒนาทักษะการควบคุมอวัยวะต่าง ๆ (เช่นมือและนิ้ว) รวมถึงเป็นการกระตุ้นการทำงานของสมองไปด้วย ตามหลักทฤษฎี sensory integration ที่ให้เน้นการกระตุ้นประสาทสัมผัสหลาย ๆ ด้าน พร้อม ๆ กันในการฝึกเด็กออทิสซึม การนำเอา TUI มาใช้ ได้มีผู้ศึกษาไว้ว่าสามารถใช้ให้เกิดประโยชน์ได้ โดยเฉพาะกับผู้พิการหรือผู้ใช้ที่ไม่ค่อยคุ้นเคยกับการใช้คอมพิวเตอร์ [1] และได้มีงานวิจัยที่นำไปประยุกต์ใช้เพื่อจัดทำระบบช่วยสอนในลักษณะของ Play to Learn เช่นงานวิจัยของ Price และคณะ [2] งานวิจัยที่ศึกษาในแง่ usability มีอาทิ เช่นงานของ Jacob [3]

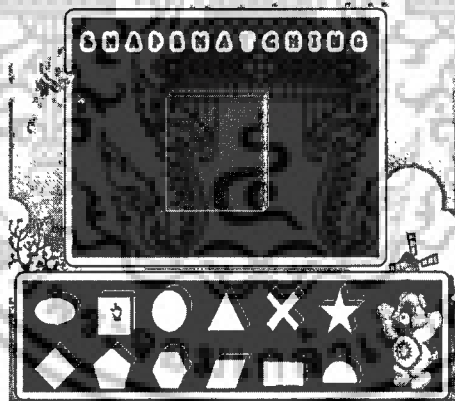
ในการจัดทำระบบช่วยสอนดังกล่าวนี้ ได้เลือกการฝึกฝนทางด้านสีและรูปร่าง เนื่องจากเป็นประเด็นของการฝึกฝนพื้นฐานที่ส่วนใหญ่ต้องเริ่มจากการฝึกฝนด้านดังกล่าว สำหรับเด็กออทิสซึมที่เป็นประเภท LFA

## 2.2 การออกแบบระบบคอมพิวเตอร์ช่วยสอนเรื่องรูปร่าง

การพัฒนาโปรแกรมหรือระบบคอมพิวเตอร์เพื่อช่วยในการฝึกฝนการแยกแยะรูปร่างพื้นฐาน ได้จัดทำขึ้นเป็น 2 รูปแบบ โดยรูปแบบแรกได้ออกแบบโดยใช้หลักการเชื่อมต่อกับผู้ใช้ในรูปแบบ WIMP Interface (Windows, Icon, Menu, Pointing device) ซึ่งก็คือรูปแบบออกแบหน้าจอกที่ใช้ในโปรแกรมคอมพิวเตอร์ส่วนใหญ่ในปัจจุบัน ที่อาศัยการควบคุมและสั่งการโดยผู้ใช้งานเมาส์และคีย์บอร์ด และในอีกรูปแบบหนึ่งที่ออกแบบโดยอาศัยหลักการเชื่อมต่อแบบ TUI นอกเหนือจากความแตกต่างในแง่ของส่วนเชื่อมต่อกับผู้ใช้แล้ว โปรแกรมทั้ง 2 เวอร์ชัน มีลักษณะหรือขั้นตอนในการเรียนที่คล้ายคลึงกัน คือ โปรแกรมจะสุ่มแสดงตัวอย่างภาพของวัตถุที่มีรูปร่างพื้นฐานทางเรขาคณิต (ทรงกลม สี่เหลี่ยม วงรี เป็นต้น) บนหน้าจอ และให้ผู้ใช้เลือกรูปร่างที่ตรงกัน โดยสำหรับโปรแกรมเวอร์ชันที่ใช้ WIMP Interface ผู้ใช้จะต้องใช้เมาส์คลิกลากไอคอนของรูปร่างที่ตรงกันไปวางบนตัวอย่างภาพบนหน้าจอ ส่วนในกรณีของ TUI Interface ผู้ใช้จะเลือกบล็อกไม้ที่มีรูปร่างที่ตรงกับตัวอย่างไปวางบนหน้าจอตรงส่วนที่แสดงตัวอย่างภาพ

### 2.2.1 โปรแกรมเวอร์ชันที่ใช้ WIMP Interface

ดังที่ได้กล่าวไปแล้ว โปรแกรมที่ใช้รูปแบบการเชื่อมต่อในลักษณะของ WIMP Interface อาศัยอุปกรณ์เมาส์และคีย์บอร์ดในการควบคุมสั่งงานกับโปรแกรม โดยผ่านการควบคุมเมนู (ใช้เมาส์หรือคีย์บอร์ด) และการคลิกลากไอคอนที่ปรากฏบนหน้าจอไปมา (โดยใช้เมาส์) ซึ่งในกรณีที่ใช้เมาส์เป็นหลักในการคลิกหรือลากไอคอน หน้าจอของโปรแกรมในเวอร์ชันนี้แสดงดังในรูปที่ 2.1 จากภาพ ดังที่ได้กล่าวไปแล้วข้างต้น ว่ารูปแบบของการเรียนทั้ง 2 เวอร์ชัน มีลักษณะคล้ายคลึงกัน กล่าวคือ โปรแกรมจะสุ่มแสดงภาพรูปทรงต่าง ๆ บริเวณตรงกลางของหน้าจอ โดยที่ในกรณีของเวอร์ชันที่ใช้ WIMP Interface ผู้ใช้จะต้องใช้เมาส์คลิกเลือกรูปทรงที่เหมือนกัน จากตัวเลือกต่าง ๆ ที่เรียงอยู่ตรงด้านล่างของหน้าจอ (ด้วยเมาส์ปุ่มซ้ายมือ) จากนั้นเลื่อนเมาส์ ซึ่งจะมีผลทำให้ไอคอนที่เลือกลากตาม mouse pointer ไป จากนั้นให้คลิกอีกครั้งเมื่อไอคอนอยู่เหนือรูปทรงตัวอย่าง เพื่อเป็นการยืนยันคำตอบ ซึ่งจะเรียกรูปแบบการปฏิสัมพันธ์กับคอมพิวเตอร์แบบนี้ว่าเป็นแบบ point-and-click interaction style สาเหตุที่เลือกรูปแบบดังกล่าวแทนที่จะใช้รูปแบบ drag-and-drop style (คลิกเมาส์ค้างไว้ ลากไปไว้เหนือรูปทรงตัวอย่าง แล้วปล่อยมือที่กดเมาส์ปุ่มซ้ายมือ) เนื่องจากพบว่าเป็นการยากสำหรับผู้ใช้กลุ่มเป้าหมายที่เป็นเด็กออทิสซึม จะเข้าใจและสามารถควบคุมเมาส์ในลักษณะดังกล่าวได้

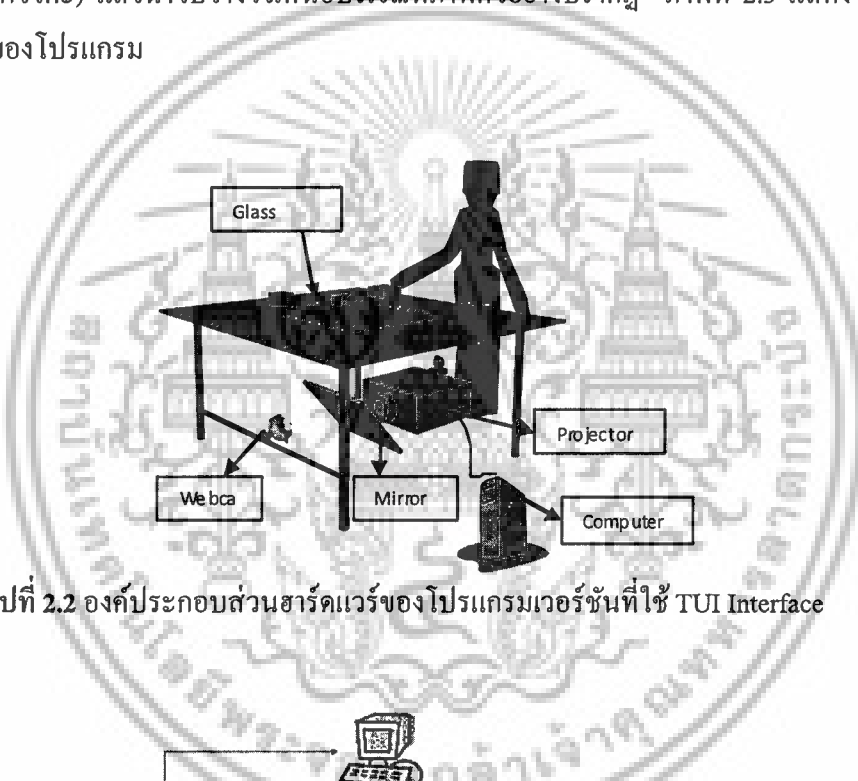


รูปที่ 2.1 ตัวอย่างหน้าจอของ โปรแกรมเวอร์ชันที่ใช้ WIMP Interface

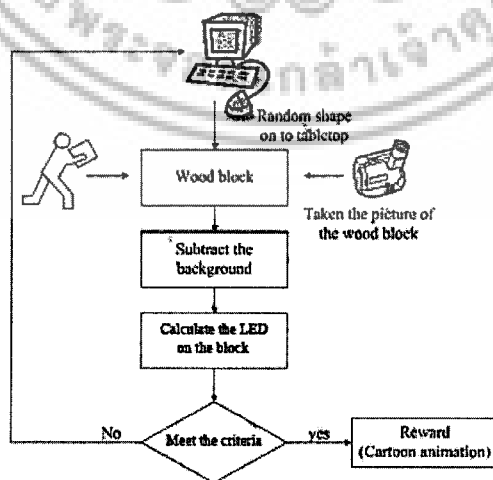
### 2.2.2 โปรแกรมเวอร์ชันที่ใช้ TUI Interface

โปรแกรมในเวอร์ชันที่ใช้ TUI Interface นี้ ออกแบบเพื่อให้ผู้ใช้สามารถติดต่อกับคอมพิวเตอร์ได้โดยการเลือก เคลื่อนย้าย และจับวางวัตถุจริงที่สามารถจับต้องได้ เหมือนหรือคล้ายคลึงกับการเล่นของเล่นทั่วไป ในรูปแบบการเชื่อมต่อแบบนี้ เนื่องจากจำเป็นต้องใช้อุปกรณ์ I/O ที่แตกต่างไปจากอุปกรณ์มาตรฐานที่มีมาพร้อมกับเครื่องคอมพิวเตอร์ที่มีจำหน่ายทั่วไป จึงมีความจำเป็นต้องทำการออกแบบและพัฒนาอุปกรณ์ในส่วนของฮาร์ดแวร์เพิ่มเติม นอกเหนือจากเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนที่เป็นซอฟต์แวร์ องค์ประกอบของระบบแสดงได้ดังรูปที่ 2.2 จากรูป การทำงานของระบบถูกควบคุมโดยส่วนของโปรแกรม ซึ่งจะแสดงภาพตัวอย่างของรูปทรงพื้นฐานออกทางหน้าจอคอมพิวเตอร์ ซึ่งในกรณีนี้ จะใช้การฉายภาพจากเครื่องฉายภาพ (projector) ที่อยู่ด้านล่างของโต๊ะที่ออกแบบมาเป็นการเฉพาะ ภาพที่ฉายจะไปปรากฏบนผิวหน้าของโต๊ะที่โปร่งแสงบางส่วน โดยด้านใดของโต๊ะ จะมีอุปกรณ์กล้องถ่ายภาพ (web camera) ที่คอยจับภาพวัตถุที่จะถูกนำมาวางบนโต๊ะ ในขณะที่การแสดงผลภาพตัวอย่างของโปรแกรมทั้ง 2 เวอร์ชัน มีลักษณะเหมือน ๆ กัน ในการเลือกคำตอบ สำหรับเวอร์ชันที่ใช้ TUI Interface นี้ ผู้ใช้จะเลือกบล็อกไม้ ที่มีรูปทรงต่าง ๆ กันให้เลือก (วางอยู่บริเวณด้านหนึ่งของโต๊ะที่ติดกับผู้เล่น) โดยเลือกรูปทรงที่เหมือนกับตัวอย่างที่แสดงบนจอ (บนผิวโต๊ะ) แล้วนำไปวางไว้ในเหนือบริเวณที่ภาพตัวอย่างปรากฏ ภาพที่ 2.3 แสดงขั้นตอนการทำงานของโปรแกรม



รูปที่ 2.2 องค์ประกอบส่วนฮาร์ดแวร์ของโปรแกรมเวอร์ชันที่ใช้ TUI Interface

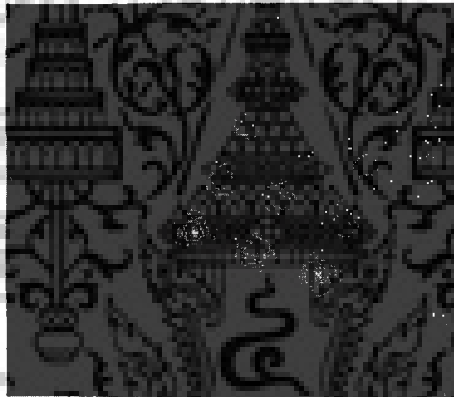


รูปที่ 2.3 ขั้นตอนการทำงานของ โปรแกรมเวอร์ชันที่ใช้ TUI Interface

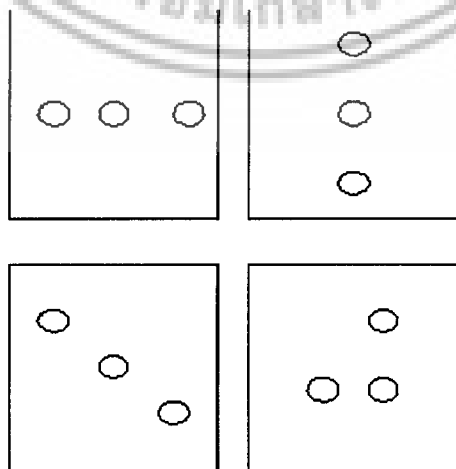
## การออกแบบและพัฒนาส่วนฮาร์ดแวร์

ในส่วนของฮาร์ดแวร์ที่ได้มีการออกแบบและพัฒนาขึ้น เพื่อให้โปรแกรมสามารถตรวจสอบบล็อกไม้ที่ถูกเลือกไปวางได้ นั้น ผู้วิจัยได้อาศัยเทคนิคการประมวลผลภาพ ประกอบกับการติดตั้งหลอดไฟอินฟราเรดขนาดเล็กไว้ทั้ง 2 ด้านของบล็อกไม้แต่ละชิ้น ซึ่งเมื่อมีการวางบล็อกไม้ไปบนผิวโต๊ะ กล้องเว็บแคมที่อยู่ภายในโต๊ะ จะสามารถจับภาพจุดสว่างต่าง ๆ ที่อยู่บนผิวหน้าของบล็อกไม้ได้ โดยที่บล็อกไม้แต่ละรูปทรง จะมีจำนวนและรูปแบบการจัดวางหลอดไฟที่แตกต่างกัน โดยอาศัยฟังก์ชันส่วนจับและประมวลผลภาพของโปรแกรม ทำให้โปรแกรมสามารถทราบได้ว่าผู้ใช้เลือกบล็อกไม้ชิ้นใด ก็จะสามารถตรวจสอบความถูกต้องของการเลือกดังกล่าวได้ หลอดไฟที่ฝังไปบนผิวของบล็อกไม้ดังที่ได้อธิบายไปแล้ว

รูปที่ 2.4 แสดงตัวอย่างภาพที่สามารถบันทึกได้จากกล้องเว็บแคม ที่อยู่อีกด้านหนึ่งของผิวโต๊ะ จุดสว่างในภาพเกิดจากหลอดไฟที่ฝังอยู่บนผิวบล็อกไม้ที่ถูกนำมาวาง รูปที่ 2.5 แสดงตัวอย่างรูปแบบการจัดวางหลอดไฟบนบล็อกไม้



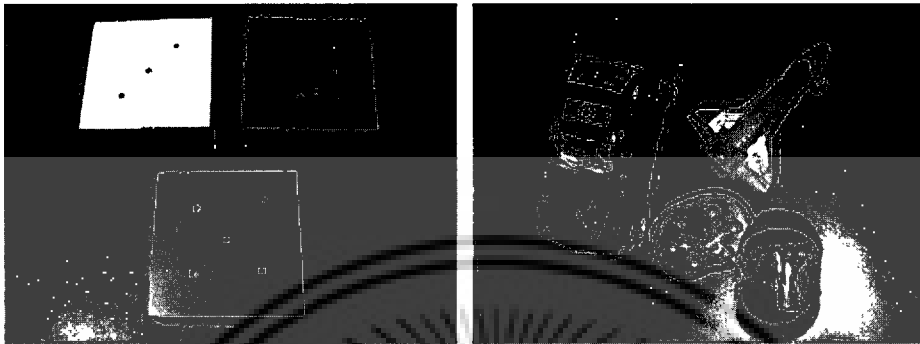
รูปที่ 2.4 ตัวอย่างภาพที่ถ่ายได้จากกล้องเว็บแคม เมื่อมีการนำบล็อกไม้มาวางไว้บนโต๊ะ



รูปที่ 2.5 ตัวอย่างรูปแบบการจัดวางหลอดไฟ บนผิวบล็อกไม้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีกรนำไปใช้

รูปที่ 2.6 ก) แสดงภาพของบล็อกไม้ที่สร้างขึ้น ส่วนรูปที่ 2.6 ข) เป็นการนำเอาของเล่นต่างๆ มาติดหลอดไฟเพื่อใช้ในการทดลอง ในกรณีหลังนี้ เป็นการนำมาใช้เพื่อการฝึกฝนเรื่องสี



รูปที่ 2.6 อุปกรณ์ TUI ที่สร้างขึ้นและใช้ในการทดลอง ก) บล็อกไม้ เพื่อใช้ในการฝึกเรื่องรูปทรง และสี ข) ของเล่น ใช้ในการฝึกฝนเรื่องสี



### บทที่ 3

## การทดลองการใช้งานโปรแกรมฝึกกรูปร่างและลี

ผลการทดลองที่แสดงในรายงานฉบับนี้ เป็นการทดลองเพิ่มเติมในปีที่ 3 ของโครงการ การทดลองที่ทำในปีที่ 1 และ 2 สามารถค้นคว้าได้เพิ่มเติมจากผลงานการตีพิมพ์ของผู้วิจัย ดังปรากฏในท้ายรายงาน

### 3.1 ข้อมูลทั่วไปเกี่ยวกับการทดลอง

การทดลองแบ่งออกเป็น 2 การทดลองหลัก ๆ เพื่อประเมินผลในส่วนของความยากง่ายในการใช้งาน (usability) และเพื่อประเมินผลในส่วนของประสิทธิภาพในการเรียนรู้ ในส่วนของการทดลองที่ 1 จะเป็นการเปรียบเทียบคุณสมบัติในแง่ความยากง่ายในการใช้งานระหว่างโปรแกรมที่ใช้ WIMP Interface กับโปรแกรมที่ใช้ TUI Interface โดยในการทดลอง จะเป็นใช้โปรแกรมในการฝึกฝนเรื่องรูปร่าง ในส่วนของการทดลองที่ 2 จะเป็นการเปรียบเทียบประสิทธิภาพในการเรียนรู้โดยการใช้โปรแกรมฝึกฝนที่พัฒนาขึ้น (เวอร์ชัน TUI Interface) กับผลที่ได้จากการฝึกฝนด้วยวิธีการที่ใช้ในปัจจุบัน ซึ่งสำหรับการฝึกเรื่องลี และรูปร่าง รูปแบบการฝึกฝนในปัจจุบัน นักกิจกรรมบ๊อบบี้จะ ตุ่มเลือกและแสดงการ์ดหรือแผ่นกระดาษที่มีภาพวัตถุที่มีรูปร่างหรือลี แตกต่างกันไปในแต่ละการ์ด แล้วให้ผู้เรียนทำการเลือกแผ่นกระดาษหรือการ์ดที่มีรูปภาพที่สอดคล้องกัน (ลี หรือรูปร่าง แล้วแต่กรณี) หรือในบางครั้งอาจเลือกใช้อุปกรณ์ของเล่นจริง ก็ได้ ดังตัวอย่างภาพการฝึกฝนจริงในรูปที่ 3.1



รูปที่ 3.1 ภาพการฝึกฝนเรื่องรูปร่างและลีด้วยวิธีการในปัจจุบัน (ที่ไม่ได้อาศัยเครื่องคอมพิวเตอร์)

## 3.2 การทดลองที่ 1

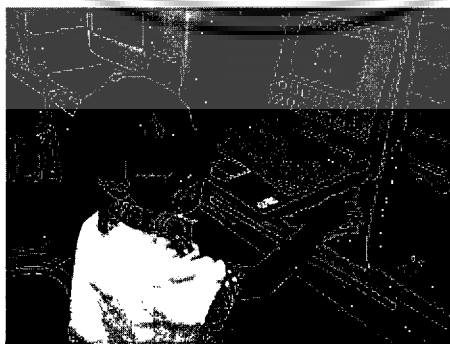
ในการทดลองเปรียบเทียบความยากง่ายในการใช้งานนี้ ได้ใช้วิธีการนับจำนวนครั้งที่พบว่าผู้ใช้ต้องการคำแนะนำหรือช่วยเหลือในการใช้งาน (โดยการสังเกตของนักกิจกรรมฯ)

### 3.2.1 ผู้ร่วมในการทดสอบ

ผู้ร่วมในการทดสอบ ประกอบไปด้วยเด็กออทิสซึม จำนวน 6 คน อายุอยู่ระหว่าง 3 ถึง 7 ขวบ โดยความยินยอมของผู้ปกครอง และโดยการประสานงานของนักกิจกรรมบำบัด โรงพยาบาลเชียงรายประชานุเคราะห์ ทุกคนเป็นเพศชาย (เนื่องจากสามารถหาผู้ร่วมทดสอบได้ง่ายกว่าที่เป็นเพศหญิง ซึ่งเป็นไปตามสถิติโดยทั่วไปของเด็กที่มีอาการออทิสซึม) โดยมีการคัดกรองเบื้องต้นเพื่อให้มีพื้นฐานใกล้เคียงกัน โดยเฉพาะอย่างยิ่งระดับความสามารถในการเรียนรู้ ผู้ร่วมทดสอบถูกแบ่งออกเป็น 2 กลุ่ม เท่า ๆ กัน กลุ่มหนึ่งให้ทดลองใช้โปรแกรมในเวอร์ชัน WIMP Interface กลุ่มที่เหลือทดลองโดยใช้โปรแกรมในเวอร์ชัน TUI Interface การทดลองทั้งหมดกระทำในคลินิกนอกเวลา เสริมจากการฝึกฝนตามปกติ ทั้งนี้เพื่อไม่ให้เกิดการทดลองกระทบการฝึกฝนตามปกติอันอาจทำให้เด็กที่ร่วมในการทดสอบเสียโอกาส ซึ่งอาจผลต่อพัฒนาการที่ควรจะได้รับ

### 3.2.2 การออกแบบการทดลอง

การทดลองแบ่งออกเป็น 6 ช่วง (sessions) แต่ละช่วงเริ่มด้วยนักกิจกรรมบำบัด ทำการอธิบายและสาธิตวิธีการใช้งาน โปรแกรมแก่ผู้ร่วมทดสอบ (การทดลองกระทำเป็นรายคน) เป็นระยะเวลา 10 นาที หลังจากนั้น ให้ผู้ร่วมทดสอบได้ใช้โปรแกรมในการฝึกฝนด้วยตนเองเป็นเวลา 10 นาที ระหว่างที่กำลังฝึก นักกิจกรรมบำบัดจะทำหน้าที่เป็นผู้สังเกตการณ์ และเมื่อเห็นว่าผู้ร่วมทดสอบไม่สามารถใช้งาน โปรแกรมหรือระบบ ได้ (ไม่มีกิจกรรมที่เกี่ยวกับการฝึกหรือใช้งาน โปรแกรมเป็นระยะเวลาเกิน 1 นาที) นักกิจกรรมฯ จึงจะเข้าไปช่วยเหลือและให้คำแนะนำหรือบอกให้ผู้ร่วมทดสอบใช้งาน โปรแกรมเพื่อการฝึกฝนต่อ จนกว่าจะครบเวลาที่กำหนด ตัวอย่างภาพแสดงการใช้งาน โปรแกรมระหว่างทดสอบ แสดงดังในรูปที่ 3.2



รูปที่ 3.2 ผู้ร่วมทดสอบขณะกำลังใช้งาน โปรแกรม (ในภาพเป็นกรณีเวอร์ชัน WIMP Interface)

### 3.2.3 ผลการทดลองและประเมิน

ผลการทดลองแสดงดังตารางที่ 3.1 และ 3.2 จากตารางแสดงว่าเวอร์ชัน WIMP Interface ต้องการความช่วยเหลือในการใช้งานมากกว่าเวอร์ชันที่ใช้ TUI Interface อย่างชัดเจน โดยในช่วงการทดสอบครั้งหลัง ๆ ไม่พบว่ามีผู้ร่วมทดสอบใดต้องการความช่วยเหลือในการใช้งานโปรแกรมเวอร์ชัน TUI Interface อีกเลย ในขณะที่กรณีของเวอร์ชัน WIMP Interface แม้ว่าแนวโน้มในการต้องการความช่วยเหลือจะลดน้อยลง แต่ก็ยังเป็นไปในอัตรการลดลงที่ค่อนข้างช้ากว่ามาก

ผู้ร่วมทดสอบ	ช่วงของการทดสอบ (Session – Sx)					
	S1	S2	S3	S4	S5	S6
D1	20	14	15	10	10	10
D2	30	13	11	8	7	1
D3	20	17	16	8	10	7

ตารางที่ 3.1 จำนวนครั้งที่ผู้ร่วมการทดสอบต้องการความช่วยเหลือในการใช้งานจากนักกิจกรรมบำบัด – กรณีโปรแกรมเวอร์ชัน WIMP Interface

ผู้ร่วมทดสอบ	ช่วงของการทดสอบ (Session – Sx)					
	S1	S2	S3	S4	S5	S6
T1	2	0	0	0	0	0
T2	0	0	0	0	0	0
T3	8	8	3	0	0	0

ตารางที่ 3.2 จำนวนครั้งที่ผู้ร่วมการทดสอบต้องการความช่วยเหลือในการใช้งานจากนักกิจกรรมบำบัด – กรณีโปรแกรมเวอร์ชัน TUI Interface

### 3.3 การทดลองที่ 2

ในการทดลองนี้ จะทำการทดสอบประสิทธิภาพในการเรียนรู้ของระบบช่วยสอนที่พัฒนาขึ้น (โดยเวอร์ชัน TUI Interface ในการทดลอง) โดยเปรียบเทียบกับรูปแบบการฝึกฝนที่ใช้อยู่ในปัจจุบัน ในการทดลองเปรียบเทียบ จะอาศัยการทำแบบทดสอบก่อน และหลังการฝึก (pre-test และ post-test) เป็นตัวชี้วัดประสิทธิภาพดังกล่าว

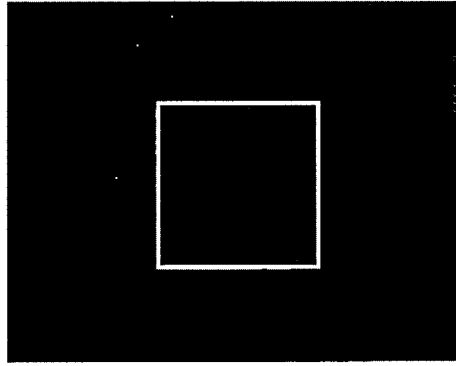
### 3.3.1 ผู้ร่วมในการทดสอบ

ผู้ร่วมในการทดสอบ ประกอบไปด้วยเด็กออทิสซึม เพศ จำนวน 16 คน อายุอยู่ระหว่าง 3 ถึง 7 ขวบ โดยความยินยอมของผู้ปกครอง และโดยการประสานงานของนักกิจกรรมบำบัด โรงพยาบาลเชียงรายประชานุเคราะห์ โดยมีการคัดกรองเบื้องต้น เพื่อให้มีพื้นฐานใกล้เคียงกัน โดยเฉพาะอย่างยิ่งระดับความสามารถในการเรียนรู้ ผู้ร่วมทดสอบถูกแบ่งออกเป็น 2 กลุ่ม เท่า ๆ กัน กลุ่มที่หนึ่งให้ทำการฝึกฝนโดยอาศัยวิธีการที่ใช้ในปัจจุบัน ส่วนที่เหลือใช้ระบบที่พัฒนาขึ้นในการฝึกฝน ทั้งนี้การฝึกฝนทั้งหมดอันเป็นส่วนหนึ่งของการทดสอบ เป็นการดำเนินการเพิ่มเติม นอกเหนือจากการฝึกตามปกติ เพื่อไม่ให้ผู้ร่วมทดสอบเสียโอกาสในการเรียนรู้และพัฒนา

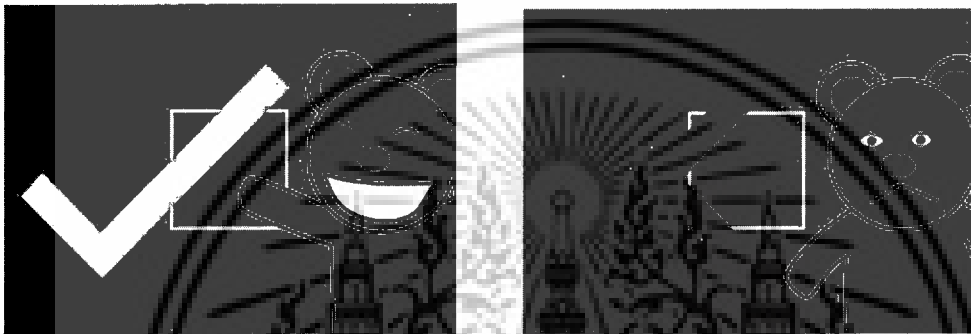
### 3.3.2 การออกแบบการทดลอง

ในการทดลองเกี่ยวกับการฝึกฝนการเรียนรู้เรื่องสี โดยการฝึกฝนแบ่งออกเป็น 3 ระดับ แต่ละระดับแบ่งการทดสอบออกเป็น 6 ช่วง (sessions) ช่วงแรกและช่วงสุดท้ายใช้เป็นแบบทดสอบเปรียบเทียบเพื่อประเมินผล (pre-test และ post-test) การทดสอบในแต่ละระดับมีรายละเอียดดังนี้

- 1) ในระดับที่ 1 กำหนดให้ผู้ร่วมทดสอบแต่ละคนทำการเลือกวัตถุ (การ์ดหรือแผ่นสีในกรณีของการฝึกฝนด้วยวิธีการตามปกติ หรือเป็นบล็อกไม้ในกรณีของการใช้ระบบที่พัฒนาขึ้น) ที่มีสีตรงกับสีที่นักกิจกรรมฯ หรือระบบเลือก ในกรณีของระบบช่วยสอน หากเลือกสีถูกต้อง ระบบจะแสดงตัวการ์ตูนการ์ตูนแอนิเมชันพร้อมเสียงเพลงและเสียงปรบมือ แต่หากผิด ระบบจะมีเสียงให้ลองใหม่อีกครั้ง พร้อมทั้งเสียงฮอด (ดูรูปที่ 3.3 ประกอบ)
- 2) ในระดับที่ 2 กำหนดให้ผู้ร่วมทดสอบแต่ละคนทำการเลือกของเล่น (ทั้งการฝึกด้วยวิธีปัจจุบันและการฝึกด้วยระบบช่วยสอน) ที่มีสีตรงกับสีที่นักกิจกรรมฯ หรือระบบเลือก ลักษณะของผลลัพธ์ที่ระบบแสดงในกรณีของการตอบถูกและผิดเป็นไปในลักษณะเดียวกับในระดับที่ 1 (ดูรูปที่ 3.4 ประกอบ) อนึ่ง ความแตกต่างจากระดับที่ 1 ก็คือ ภาพที่แสดงกับวัตถุที่ต้องเลือกมีความเหมือนกันเพียงอย่างเดียวคือสี ขนาดและรูปร่างอาจแตกต่างกันโดยสิ้นเชิง เพื่อเป็นการทดสอบความสามารถในการปรับใช้ความเข้าใจกับสถานการณ์ใหม่ ๆ (generalization)
- 3) สำหรับในระดับที่ 3 จะไม่มีการแสดงภาพตัวอย่างให้ดู นักกิจกรรมฯ หรือระบบฯ จะบอกด้วยเสียงว่าให้เลือกวัตถุที่มีสีใดเท่านั้น เป็นการทดสอบความเข้าใจเชิงแนวคิดและสัญลักษณ์ (ภาษา)



(ก)



(ข)

(ค)

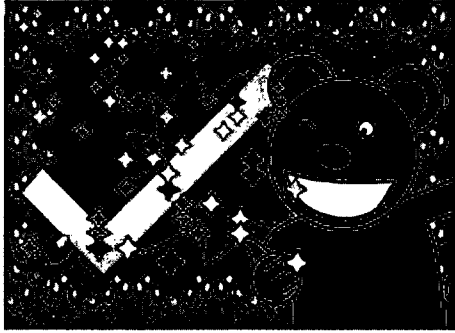
รูปที่ 3.3 หน้าจอที่โปรแกรมแสดงในการฝึกฝนระดับที่ 1 ก) เมื่อโปรแกรมกำหนดให้ผู้เล่นเลือก  
บลิ๊อคสี ข) เมื่อเลือกถูก ค) เมื่อเลือกผิด



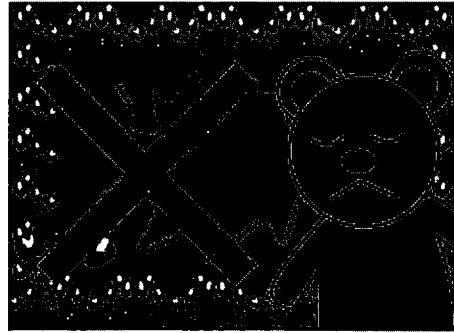
(ก)

รูปที่ 3.4 หน้าจอที่โปรแกรมแสดงในการฝึกฝนระดับที่ 2 ก) เมื่อโปรแกรมกำหนดให้ผู้เล่นเลือก  
ของเล่นสี ข) เมื่อเลือกถูก ค) เมื่อเลือกผิด

## สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



(ข)



(ค)

รูปที่ 3.4 (ต่อ)

แต่ละช่วงของการทดลองจะเริ่มต้นด้วยการแนะนำบทเรียนและการทดสอบ โดยนักกิจกรรมบำบัดจะแนะนำสีต่าง ๆ ที่แตกต่างกัน 5 สี (ดำ ขาว แดง เขียว และน้ำเงิน) และวิธีการหรือขั้นตอนในการฝึกฝน หลังจากนั้นจะเริ่มเข้าสู่เนื้อหาของบทเรียน

### 3.3.3 ผลการทดลองและประเมิน

จากการสรุปผลการทำแบบทดสอบก่อนและหลังการทดลอง โดยการหาผลต่างระหว่าง post-test และ pre-test เพื่อใช้เป็นตัววัดประสิทธิภาพของการเรียนรู้ ผลการวิเคราะห์แสดงได้ดังในตารางที่ 3.3 และ 3.4

ผู้ร่วมทดสอบ	คะแนน pre-test	คะแนน post-test
C1	0	1
C2	0	2
C3	0	1
C4	0	2
C5	0	3
C6	1	2
C7	0	0
C8	0	1

ตารางที่ 3.3 คะแนน pre-test และ post-test สำหรับการฝึกฝนด้วยวิธีการที่ใช้ในปัจจุบัน

ผู้ร่วมทดสอบ	คะแนน pre-test	คะแนน post-test
T1	1	6
T2	0	6
T3	0	6
T4	0	4
T5	0	6
T6	0	5
T7	0	6
T8	0	6

ตารางที่ 3.4 คะแนน pre-test และ post-test สำหรับการฝึกฝนด้วยระบบที่พัฒนาขึ้น

จากตารางดังกล่าวข้างต้น โดยการเปรียบเทียบผลต่างของคะแนน post-test และ pre-test พบว่าการฝึกฝนด้วยระบบที่พัฒนาขึ้น มีประสิทธิภาพในการเรียนรู้ที่สูงกว่าการฝึกฝนด้วยวิธีการปัจจุบันอย่างชัดเจน

### 3.4 แนวทางการออกแบบโปรแกรมสำหรับเด็กออทิสซึม

ผลจากการศึกษา พัฒนา และทดลองใช้โปรแกรมฝึกฝนในรูปแบบต่าง ๆ สำหรับเด็กออทิสซึม ผู้วิจัยได้นำผลที่ได้มาประมวลสรุปเป็นแนวทางการออกแบบ (design guidelines) โปรแกรมสำหรับเด็กออทิสติกได้ดังต่อไปนี้

- 1) ขนาดของ tangible object: เนื่องจากเด็กออทิสซึมส่วนใหญ่มีปัญหาในการควบคุมแขนขา หรือมือ (fine motor skill) ขนาดวัตถุที่จะใช้สำหรับเป็น tangible object ไม่ควรใหญ่หรือเล็กจนเกินไป
- 2) พื้นที่การทำงานหรือใช้งาน: เช่น ขนาดของโต๊ะที่ใช้ฉายภาพและวางวัตถุ ควรอยู่ในระยะที่เด็กสามารถเอื้อมถึง หากใหญ่หรือกว้างเกินไป อาจเป็นผลทำให้เด็กเสียสมาธิได้
- 3) การใช้เสียง : ควรระมัดระวังในการใช้เสียง เด็กออทิสซึมบางคนไม่สามารถทนกับเสียงบางประเภทได้ เช่น เสียงฮูด หรือเสียงดังรบกวน
- 4) หน้าจอ: ควรออกแบบให้เรียบง่าย เนื่องจากเด็กออทิสซึมอาจถูกดึงความสนใจจากบทเรียนได้ง่าย หากมีตัวการ์ตูนตกแต่งหน้าจอ
- 5) การตอบสนอง (feedback) เพื่อแสดงว่าตอบถูกหรือผิด ควรแสดงด้วยคำพูด สัญลักษณ์ หรือข้อความที่สื่อสารได้ง่าย ๆ บางครั้งการใช้ตัวการ์ตูนที่สื่อถึงด้านบวกหรือด้านลบ (เช่น หน้าเศร้า หรือหน้าดีใจ) เด็กออทิสซึมอาจแปลความหมายผิดไปจากที่ต้องการได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

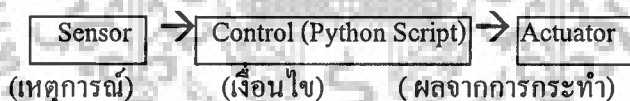
## บทที่ 4

# เกมฝึกการใช้มือโดยใช้จอยสติ๊กส์แบบพวงมาลัย

ส่วนหนึ่งของโครงการวิจัยนี้ คือการพัฒนาโปรแกรมคอมพิวเตอร์เพื่อช่วยสอนหรือฝึกฝนเด็กออทิสซึม ปัญหาหนึ่งของเด็กออทิสซึมจำนวนหนึ่ง คือความสามารถในการควบคุมการใช้มือให้สัมพันธ์กับสิ่งที่เห็น (eye-hand coordination) ในงานวิจัยนี้ ได้ทำเทคโนโลยีด้านคอมพิวเตอร์กราฟิก มาประกอบกับอุปกรณ์ติดต่อที่มีลักษณะของ TUI interface (จอยสติ๊กส์แบบพวงมาลัย หรือพวงมาลัยอิเล็กทรอนิกส์) เพื่อทำการพัฒนาเกมที่ให้เด็กออทิสซึม หรือผู้ที่มีปัญหาในลักษณะคล้ายคลึงกัน ได้ใช้ในการฝึกฝนการควบคุมมือให้สัมพันธ์กับภาพที่เห็น รายละเอียดของการพัฒนาโปรแกรมปรากฏในหัวข้อต่าง ๆ ดังต่อไปนี้ โดยจะเริ่มจากการให้ข้อมูลเบื้องต้นเกี่ยวกับเกมเอ็นจินที่ใช้ในการพัฒนา (Blender) การเชื่อมต่ออุปกรณ์พวงมาลัยอิเล็กทรอนิกส์เข้ากับเกมเอ็นจิน และสุดท้ายจะเป็นการนำเสนอรายละเอียดในส่วนของตัวเกม

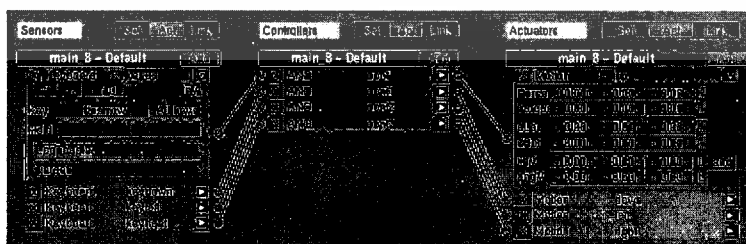
### 4.1 หลักการทำงานของเบลนเดอร์เกมเอ็นจิน (Blender Game Engine)

เบลนเดอร์เกมเอ็นจิน ประกอบด้วย สามส่วนที่ทำงานร่วมกัน โดยในการกำหนดการควบคุมในแต่ละครั้ง ทั้งสามส่วนจะทำงานสัมพันธ์กันตามแผนภาพดังนี้



รูปที่ 4.1 การทำงานของ เบลนเดอร์เกมเอ็นจิน

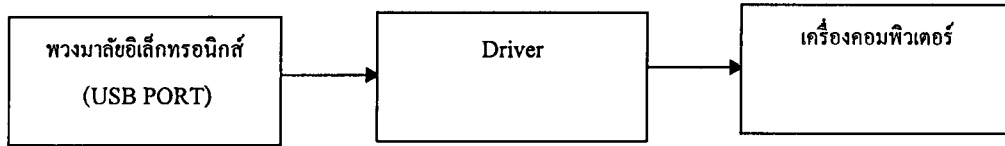
ตัวอย่างการกำหนดเหตุการณ์ในเกมด้วยเบลนเดอร์เกมเอ็นจิน (Blender Game Engine) แสดงได้ดังรูปต่อไปนี้



รูปที่ 4.2 การกำหนดเหตุการณ์ในเกมด้วยเบลนเดอร์เกมเอ็นจิน

## 4.2 การเชื่อมต่ออุปกรณ์

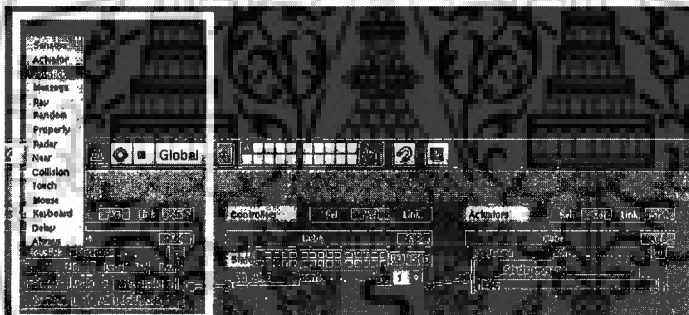
การเชื่อมต่ออุปกรณ์ พวงมาลัยอิเล็กทรอนิกส์ นั้น โดยจะทำการเชื่อมต่อผ่าน USB port ซึ่งจะต้องทำการลงไดรเวอร์ของชุดอุปกรณ์ก่อนเครื่องคอมพิวเตอร์จึงจะสามารถมองเห็นได้



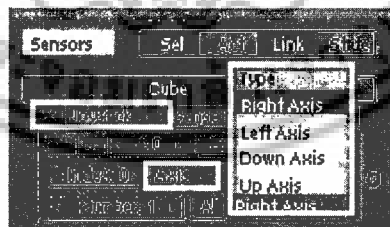
รูปที่ 4.3 ลักษณะของการเชื่อมต่อชุดอุปกรณ์เข้ากับเครื่องคอมพิวเตอร์

## 4.3 การเชื่อมต่อพวงมาลัยอิเล็กทรอนิกส์เข้ากับโปรแกรมเบลนเดอร์

พวงมาลัยอิเล็กทรอนิกส์สามารถเชื่อมต่อเข้ากับโปรแกรมเบลนเดอร์โดยใช้ Blender Game Engine ซึ่ง Blender Game Engine จะมองเห็นพวงมาลัยอิเล็กทรอนิกส์ผ่านส่วนของ sensor ประเภท joystick ดังรูปที่ 4.4



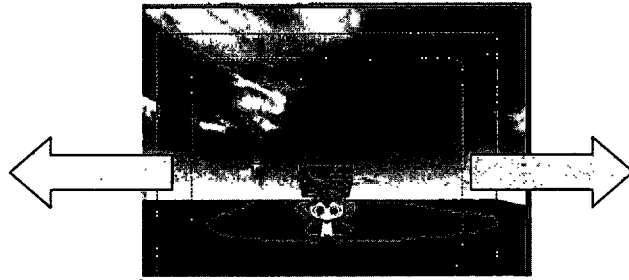
รูปที่ 4.4 การเรียกใช้อุปกรณ์ผ่าน Blender Game Engine



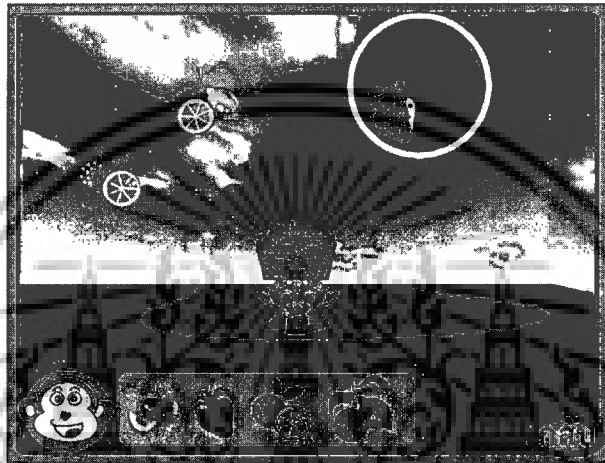
รูปที่ 4.5 การกำหนด Sensor เพื่อรับค่าข้อมูลจากพวงมาลัย

## 4.4 เกม Happy Monkey

เกมถูกออกแบบให้ผู้เล่นรับบทเป็นลิงคอยรับผลไม้ที่ตกลงมาจากท้องฟ้า โดยการหมุนพวงมาลัยเพื่อเลื่อนให้ลิงไปทางซ้าย และทางขวา เพื่อเก็บผลไม้ ดังรูปที่ 4.6 โดยผลไม้จะร่วงลงมาอย่างช้าๆ ดังรูปที่ 4.7



รูปที่ 4.6 การบังคับลิงให้ไปทางซ้ายและขวา



รูปที่ 4.7 ภาพผลไม้ที่หล่นลงมา

เมื่อเก็บผลไม้ได้ ที่หน้าจอด้านล่างจะมีรูปผลไม้ขึ้น สีสดใส และมีลิงออกมาจากด้านข้าง แสดงความยินดี ดังรูปที่ 4.8



รูปที่ 4.8 ภาพเมื่อรับผลไม้ได้

## บทที่ 5

# โปรแกรมจำลองสถานการณ์สำหรับฝึกเด็กออทิสซึม

เทคโนโลยีความจริงเสมือน (Virtual Reality) ถูกนำมาใช้ช่วยฝึกเด็กออทิสซึมเป็นระยะเวลานานแล้ว เนื่องจากสามารถสร้างฉากและสถานการณ์จำลองต่างๆ ได้หลากหลายโดยไม่ต้องเสียค่าใช้จ่ายมากนัก และยังให้ความปลอดภัยในการฝึกต่อเด็กอีกด้วย

การสร้างระบบความจริงเสมือน โดยทั่วไปนั้นจำเป็นต้องตอบสนองต่อความต้องการดังต่อไปนี้

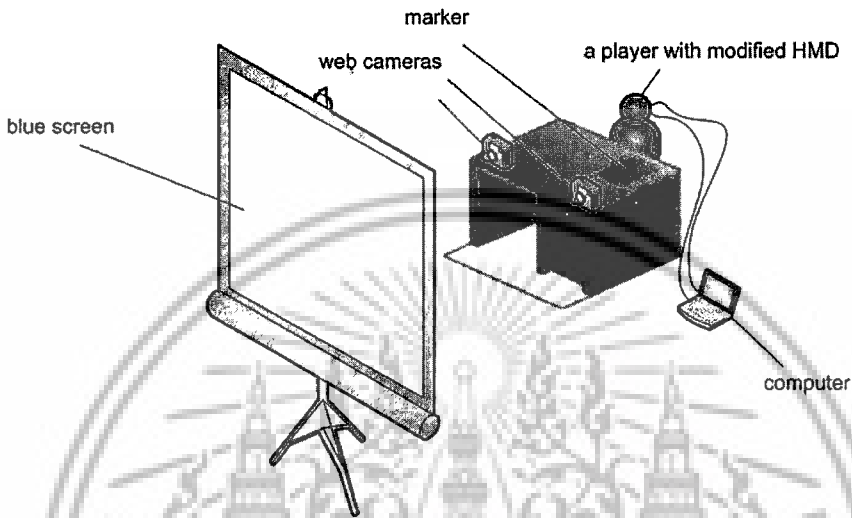
- ความสมจริง (Realistic) ภาพและเสียงที่ผู้ใช้รับรู้จากระบบนั้นจะต้องมีความสมจริงใกล้เคียงกับสิ่งที่พบเห็นในธรรมชาติ
  - การตอบสนองในเวลาจริง (Real-Time) ภาพและเสียงนั้นจะต้องมีการเปลี่ยนแปลงอย่างทันทีทันใด เมื่อเกิดเหตุการณ์ต่างๆ ขึ้นในระบบ
  - ปฏิสัมพันธ์ (Interaction) ผู้ใช้ต้องสามารถตอบโต้กับระบบได้
- เพื่อให้โปรแกรมสามารถตอบสนองต่อความต้องการข้างต้น จึงจำเป็นต้องใช้เทคโนโลยีหลายๆ อย่างมาช่วยในการพัฒนา

### 5.1 โครงสร้างของโปรแกรมจำลองสถานการณ์สำหรับฝึกเด็กออทิสซึม

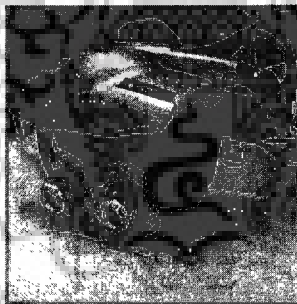
โปรแกรมที่พัฒนาขึ้นนี้เป็นโปรแกรมจำลองสถานการณ์สำหรับฝึกการซื้อของจากร้านค้าสำหรับเด็กออทิสซึม โดยมีจุดมุ่งหมายให้เด็กออทิสซึมมีความคุ้นเคยกับการเลือกซื้อสินค้าและจ่ายเงินให้กับพนักงานคิดเงิน จนสามารถนำไปใช้ในชีวิตประจำวันได้

ภาพรวมของโปรแกรมแสดงดังรูปที่ 5.1 ผู้เล่นจะสวมอุปกรณ์แสดงผลภาพแบบสวมศีรษะ (Head Mounted Display) ซึ่งเชื่อมต่อกับคอมพิวเตอร์ โดยอุปกรณ์แสดงผลภาพแบบสวมศีรษะนี้จะถูกดัดแปลงให้มีการติดตั้งเว็บแคม จำนวน 2 ตัว ดังรูปที่ 5.2 เพื่อใช้ถ่ายภาพจากสภาพแวดล้อมภายนอก และนำมาแสดงที่จอแสดงผลของอุปกรณ์แสดงผลภาพแบบสวมศีรษะ ดังนั้นเมื่อผู้เล่นสวมอุปกรณ์แสดงผลภาพแบบสวมศีรษะแล้ว ผู้เล่นจะมองเห็นทั้งภาพจากสภาพแวดล้อมจริงดังเช่นที่มองเห็นด้วยตาเปล่า และภาพสังเคราะห์ที่สร้างจากจากคอมพิวเตอร์ควบคู่กันไปด้วย เบื้องหน้าของผู้เล่นจะมีโต๊ะที่ใช้สำหรับวางสินค้าและเงินที่ใช้ในการซื้อของ โดยสินค้าและเงินนั้นจะถูกแทนด้วยเครื่องหมาย (marker) แบบพิเศษ ซึ่งจะเป็นภาพสัญลักษณ์ต่างๆ พิมพ์อยู่บนกระดาษแล้วนำไปติดลงบนวัตถุซึ่งมีรูปร่างและขนาดใกล้เคียงกับสิ่งที่ต้องการแทน เช่น เครื่องหมายที่ใช้แทนขนมชนิดหนึ่งซึ่งบรรจุอยู่ในกล่องรูปสี่เหลี่ยม ก็จะถูกติดลงบนกล่องกระดาษรูปสี่เหลี่ยมที่ไม่มีลวดลายใดๆ เป็นต้น ดังแสดงในรูปที่ 5.3 ถัดจากโต๊ะจะเป็นฉากสีฟ้าขนาดใหญ่ ดังในรูปที่ 5.4 ซึ่งฉากสีฟ้า

นี้จะถูกใช้เป็นพื้นที่ในการแสดงภาพร้านค้าจำลอง นอกจากนี้แล้วจะมีการติดตั้งเว็บแคมอีก 2 ตัวที่หันหน้าเข้าหาผู้เล่น ซึ่งกล้องดังกล่าวจะใช้ในการหาตำแหน่งของผู้เล่น (ในกรณีที่ต้องการจำลองผู้ขายด้วย ในการพัฒนาในเบื้องต้นนี้ ใช้คนหรือนักกิจกรรมบ่าบดอยู่อีกด้านหนึ่งของโต๊ะ เบื้องหน้าจอแสดงภาพฉากหลังสีฟ้า)



รูปที่ 5.1 โครงสร้างของโปรแกรมจำลองสถานการณ์สำหรับฝึกเด็กออทิสซึม

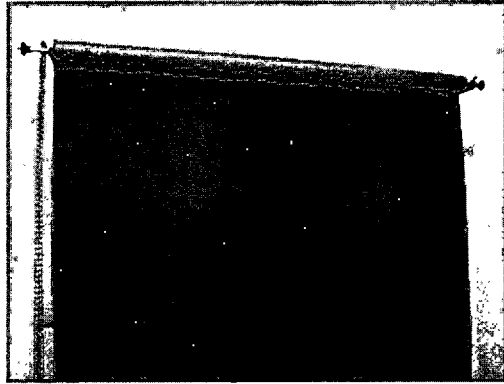


รูปที่ 5.2 อุปกรณ์แสดงภาพแบบสวมศีรษะที่ติดตั้งกล้องเว็บแคมเพิ่มเติม



รูปที่ 5.3 เครื่องหมายที่ถูกติดลงไปบนสิ่งของต่างๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 5.4 ภาพแสดงฉากสีฟ้าที่ใช้เป็นพื้นหลัง

เมื่อเริ่มโปรแกรมผู้เล่นสวมอุปกรณ์แสดงภาพแบบสวมศีรษะ ยืนอยู่ที่หน้าโต๊ะ แล้วมองไปยังฉากสีฟ้า ดังรูป 5.5 และ 5.6 ผู้เล่นจะมองเห็นสภาพแวดล้อมภายนอกตามภาพที่ส่งมาจากกล้อง โดยภาพนั้นจะแสดงในแบบสเตอริโอ 3 มิติ ซึ่งผู้เล่นจะมองเห็นความลึกของสิ่งต่างๆ ได้ชัดเจนมากขึ้นกว่าภาพที่เห็นจากกล้องโดยทั่วไป นอกจากนี้แล้วหากผู้เล่นต้องการขยายภาพเข้าออกเพื่อให้ใกล้เคียงกับการมองเห็นด้วยตาเปล่านั้นก็สามารถทำได้เช่นกัน ส่วนฉากสีฟ้าที่อยู่ด้านหน้าของผู้เล่นนั้นจะถูกแทนที่ด้วยภาพของฉากร้านค้า ดังรูปที่ 5.7 ซึ่งภาพดังกล่าวจะมีการเปลี่ยนแปลงโดยสัมพันธ์กับตำแหน่งของผู้เล่น ดังนั้น เมื่อผู้เล่นเคลื่อนที่ไปมา ภาพของร้านค้าที่เห็นบนฉากสีฟ้าก็จะเปลี่ยนแปลงตามไปด้วย



รูปที่ 5.5 แสดงการใช้งานจริง – มองจากด้านหลังผู้เล่น



รูปที่ 5.6 แสดงการใช้งานจริง - มองจากด้านหน้าผู้เล่น



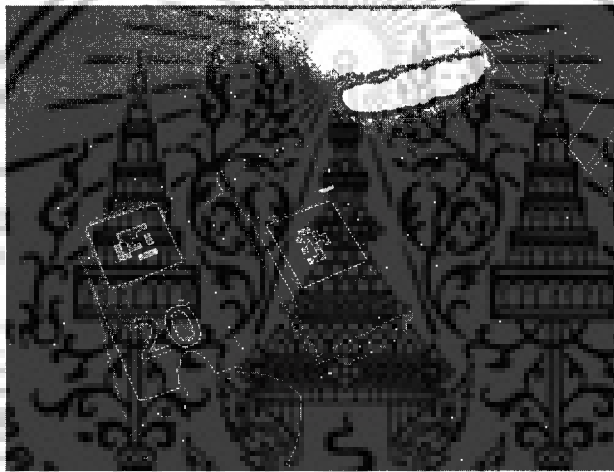
รูปที่ 5.7 สิ่งที่ผู้เล่นมองเห็น - ฉากหลังสีฟ้าถูกซ่อนด้วยภาพฉากหลังใหม่

ในระหว่างนั้นผู้เล่นจะได้ยินเสียงของผู้บรรยายจากโปรแกรมซึ่งจะบอกให้วางสินค้าที่  
ต้องการจะซื้อ เมื่อผู้เล่นเลือกวัตถุที่เป็นตัวแทนของสินค้าที่ต้องการจะซื้อและนำมาวางลงบนโต๊ะ  
แล้วดังรูปที่ 5.8 เสียงบรรยายจะแจ้งให้ทราบว่าสินค้าที่เลือกซื้อนั้นทั้งหมดราคาเท่าใด จากนั้นผู้เล่น  
จะเลือกวางวัตถุที่เป็นตัวแทนของเงินลงบนโต๊ะ ดังรูปที่ 5.9 (ในกรณีที่ใช้คนเป็นผู้ขายจริง จะใช้  
เสียงคนจริง ๆ แทน)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก้นำไปใช้



รูปที่ 5.8 แสดงการใช้งานจริง – ผู้เล่นวางสิ่งของที่ซื้อ



รูปที่ 5.9 แสดงการใช้งานจริง – ผู้เล่นวางจำนวนเงินที่ต้องชำระ



รูปที่ 5.10 สิ่งที่ผู้เล่นมองเห็น – โมเดลจากคอมพิวเตอร์ถูกวางลงบนวัตถุที่ผู้เล่นวางบนโต๊ะ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีก้นำไปใช้

ผู้เล่นจะมองเห็นภาพของสินค้าที่แท้จริงถูกวางลงบนวัตถุที่เป็นตัวแทนนั้น เช่นเดียวกับวัตถุที่เป็นตัวแทนของเงินก็จะถูกแทนที่ด้วยภาพของธนบัตรจริงๆ ดังรูปที่ 5.10 โดยทั้งภาพสินค้าและภาพธนบัตรนั้นเป็นภาพสังเคราะห์จากคอมพิวเตอร์ทั้งสิ้น แล้วเสียงบรรยายก็จะแจ้งให้ทราบอีกครั้งว่าเงินที่จ่ายนั้นพอดีกับราคาสินค้าหรือไม่ หากเกินเท่าใด ผู้เล่นจะต้องวางเงินให้เพียงพอกับราคาสินค้าทั้งหมดที่ต้องการซื้อ จึงจะจบขั้นตอนของการซื้อสินค้าได้ ทั้งหมดเป็นภาพรวมการทำงานของโปรแกรมจำลองนี้ ในหัวข้อต่อไปจะเป็นการอธิบายถึงรายละเอียดในการพัฒนาในแต่ละส่วน

## 5.2 เครื่องมือในการพัฒนาโปรแกรม

โปรแกรมจำลองสถานการณ์สำหรับฝึกเด็กออทิสซึมนี้พัฒนาขึ้นโดยใช้ภาษา C++ ใช้เครื่องมือ Microsoft Visual C++ 2008 ร่วมกับไลบรารีต่างๆ ดังนี้

- OpenCV เป็นไลบรารีสำหรับงานทางด้าน Computer Vision และ Image Processing ใช้ในการติดต่อกับกล้อง นำภาพที่ได้จากกล้องมาประมวลผล และจัดเก็บรูปภาพ
- OpenGL เป็นไลบรารีสำหรับการเรนเดอร์ภาพ 3 มิติ ใช้สำหรับการนำภาพจากกล้องทั้งสองและภาพจากชุดภาพมาเรนเดอร์ในระนาบของโลก 3 มิติ แล้วสร้างการเรนเดอร์แบบสเตอริโอ
- glut เป็นไลบรารีสำหรับสร้าง GUI ให้กับโปรแกรมที่ใช้ OpenGL ใช้สร้างและจัดการ window ของโปรแกรม
- ARToolkitPlus เป็นไลบรารีสำหรับการทำ template matching ใช้ในการตรวจสอบ marker ที่ติดกับวัตถุ เพื่อจำแนกว่าผู้ใช้วางวัตถุใดเข้ามาในฉาก และวัตถุนั้นอยู่ในตำแหน่งและทิศทางใด
- OpenAL เป็นไลบรารีสำหรับงานด้านเสียง ใช้ในการเล่นไฟล์เสียงบรรยายของโปรแกรมทั้งหมด

พัฒนาฟังก์ชันต่างๆ ของโปรแกรกดังต่อไปนี้

- ฟังก์ชันในการติดต่อกับกล้องเว็บแคม
- ฟังก์ชันในการนำภาพของกล้องมาซ้อนภาพฉากหลังแบบบลูสกรีน
- ฟังก์ชันในการตัดภาพบางส่วนของกล้องมาขยาย
- ฟังก์ชันในการนำภาพจากกล้องทั้งสองมาแสดงผลในแบบสเตอริโอ
- ฟังก์ชันในการแปลงภาพจาก OpenCV มาใช้ใน OpenGL
- ฟังก์ชันในการสร้างภาพใหม่จากชุดภาพถ่าย
- ฟังก์ชันในการเรนเดอร์ภาพด้วย OpenGL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ฟังก์ชันในการหาตำแหน่งของผู้เล่น
- ฟังก์ชันในการตรวจจับวัตถุที่ผู้เล่นนำมาวางในฉาก
- ฟังก์ชันในการเล่นไฟล์เสียงตามเหตุการณ์ที่เกิดขึ้น
- ฟังก์ชันหลัก ที่ควบคุมการทำงานทั้งหมด

### 5.3 การแสดงภาพของสภาพแวดล้อมภายนอกแบบสเตอริโอ

อุปกรณ์แสดงภาพแบบสวมศีรษะแสดงภาพที่ส่งมาจากคอมพิวเตอร์ได้ โดยจะทำให้ผู้สวมใส่มองเห็นเฉพาะภาพที่แสดงบนจอแสดงผลของอุปกรณ์เท่านั้น ต่างจากการมองภาพบนจอมอนิเตอร์ ที่ผู้ใช้จะมองเห็นสภาพแวดล้อมภายนอกไปด้วย การใช้อุปกรณ์แสดงภาพแบบสวมศีรษะจึงให้ความรู้สึกที่สมจริงกว่า อย่างไรก็ตาม โปรแกรมนี้ต้องการให้ผู้เล่นเกิดความรู้สึกสมจริงเหมือนดูภาพด้วยตาของตนเอง จึงใช้ภาพของสภาพแวดล้อมภายนอกมาแสดงบนอุปกรณ์แสดงภาพแบบสวมศีรษะแทนการใช้ภาพที่สร้างจากคอมพิวเตอร์ ซึ่งทำได้โดยการติดเว็บแคม 2 ตัว ไว้ที่ด้านหน้าของอุปกรณ์แสดงภาพแบบสวมศีรษะ แล้วนำภาพที่กล้องส่งมายังคอมพิวเตอร์ มาแสดงที่จอภาพของอุปกรณ์แสดงภาพแบบสวมศีรษะ

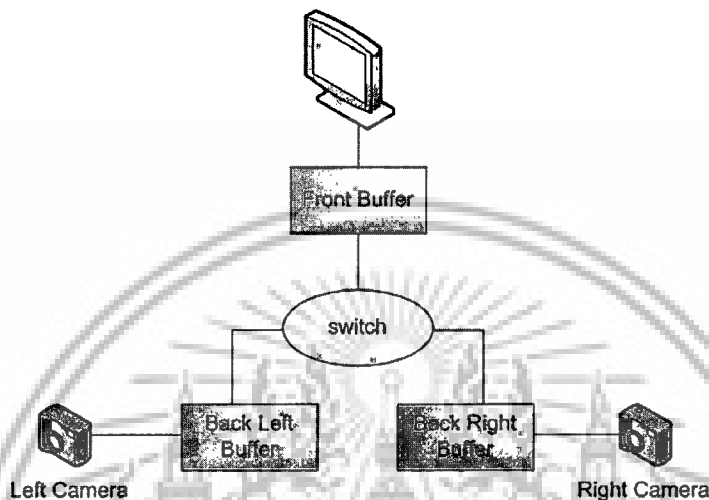
ภาพที่แสดงบนจอภาพโดยทั่วไปนั้นจะเป็นภาพ 2 มิติ ซึ่งแตกต่างจากการมองภาพด้วยตาของมนุษย์ที่มองเห็นภาพเป็น 3 มิติ การจำลองให้ภาพที่มองเห็นบนจอภาพเป็น 3 มิติใกล้เคียงกับการมองภาพของมนุษย์นั้นจะใช้เทคนิคการสร้างภาพแบบสเตอริโอ (Stereoscopic Imaging) ซึ่งจะเป็นการสร้างภาพขึ้นมาสองภาพจากตำแหน่งของกล้องที่ห่างกันเล็กน้อย ซึ่งเมื่อนำภาพทั้ง 2 มาฉายลงบนจอภาพพร้อมกัน โดยใช้เทคนิคบางอย่างที่ทำให้ตาแต่ละข้างมองเห็นเพียงภาพจากกล้องใดกล้องหนึ่ง เมื่อสมองของมนุษย์นำภาพจากทั้งสองด้านนั้น ไปประมวลผลก็จะทำให้เกิดเป็นภาพ 3 มิติ ซึ่งเป็นภาพที่สามารถรับรู้ความลึกของวัตถุต่างๆ ที่มองเห็นบนจอภาพได้ ทำให้ภาพที่มองผ่านอุปกรณ์แสดงภาพแบบสวมศีรษะนั้นใกล้เคียงกับการมองด้วยสายตาตนเองมากยิ่งขึ้น

การจะสร้างภาพแบบสเตอริโอได้นั้นจำเป็นต้องมีอุปกรณ์รับภาพจำนวน 2 ตัว ในที่นี้ก็คือเว็บแคม ซึ่งต้องติดกล้องทั้ง 2 ตัว เอาไว้บนอุปกรณ์แสดงภาพแบบสวมศีรษะ โดยมีระยะห่างที่ใกล้เคียงกับระยะห่างระหว่างตาซ้ายและตาขวาของมนุษย์คือประมาณ 4 ซม. แล้วใช้ภาพที่ได้จากกล้องซ้ายมาเก็บไว้ในเฟรมบัฟเฟอร์ (หน่วยความจำของการ์ดแสดงผล) ด้านหลังข้างซ้าย (Back Left Frame Buffer) ภาพที่ได้จากกล้องขวามาเก็บไว้ในเฟรมบัฟเฟอร์ด้านหลังข้างขวา (Back Right Frame Buffer) [28] ภาพที่ถูกเก็บเอาไว้ในเฟรมบัฟเฟอร์ทั้งสองจะยังไม่ถูกนำไปวาดลงบนหน้าจอ เพราะสิ่งที่จะถูกวาดลงบนหน้าจอจะต้องถูกเก็บเอาไว้ที่เฟรมบัฟเฟอร์ด้านหน้า (Front Frame Buffer) เนื่องจากการใช้เฟรมบัฟเฟอร์ด้านหน้าเพียงอย่างเดียวนั้นจะทำให้เกิดปัญหาการกระพริบ (flicker) บนหน้าจอในช่วงเวลาที่เปลี่ยนจากเฟรมหนึ่งไปอีกเฟรมหนึ่ง ดังนั้นเพื่อแก้ปัญหาดังกล่าว จึงต้องมีการเพิ่มเฟรมบัฟเฟอร์ด้านหลัง (Back Frame Buffer) เพื่อใช้เก็บข้อมูลของภาพที่เตรียมจะ

ถูกนำไปวาดลงบนหน้าจอในเฟรมถัดไป โดยเมื่อวาดภาพจากเฟรมบัฟเฟอร์ด้านหน้าลงบนจอภาพเรียบร้อยแล้ว เฟรมบัฟเฟอร์ด้านหน้าจะถูกสลับหน้าทีกลายเป็นเฟรมบัฟเฟอร์ด้านหลัง และเฟรมบัฟเฟอร์ด้านหลังก็จะถูกสลับมาเป็นเฟรมบัฟเฟอร์ด้านหน้า และนำข้อมูลของภาพที่เตรียมไว้ไปวาดลงบนหน้าจอได้ทันที บัฟเฟอร์ทั้งสองจะถูกสลับการทำงานเช่นนี้ไปเรื่อยๆ ซึ่งเทคนิคนี้เรียกว่าการทำบัฟเฟอร์คู่ (Double Buffering) ซึ่งช่วยแก้ปัญหาการกระพริบของภาพบนหน้าจอได้ ซึ่งสำหรับการแสดงภาพโดยทั่วไปนั้นจะมีเฟรมบัฟเฟอร์ด้านหลังเพียงอันเดียว แต่สำหรับการแสดงภาพแบบสเตอริโอ นั้นจะต้องมีเฟรมบัฟเฟอร์ด้านหลังสองตัว คือ ซ้ายซ้ายและข้างขวา เมื่อเวลาจะสลับเฟรมบัฟเฟอร์ระหว่างด้านหลังและด้านหน้าจะมีเพียงหนึ่งเฟรมบัฟเฟอร์เท่านั้นที่ถูกเลือกปัญหาต่อไปก็คือ จะเลือกอย่างไรและจะทำอย่างไรจึงจะบังคับให้ตาของผู้เล่นแต่ละข้างมองเห็นภาพจากกล้องด้านใดด้านหนึ่งเท่านั้น ซึ่งในที่นี้จะใช้เทคนิคที่เรียกว่าเพจฟลิปปีง (Page Flipping) นั่นก็คือในแต่ละเฟรม คอมพิวเตอร์จะส่งภาพจากเฟรมบัฟเฟอร์ด้านใดด้านหนึ่ง (ซ้ายหรือขวา) มาแสดงผลบนหน้าจอ และในเฟรมถัดไปก็จะส่งภาพจากเฟรมบัฟเฟอร์อีกด้านหนึ่งมาแสดงผลแทนสลับกันไปเรื่อยๆ เช่นนี้ ซึ่งหากมองดูด้วยตาเปล่าบนจอภาพ จะเห็นภาพจากทั้งกล้องด้านซ้ายและด้านขวาแสดงพร้อมๆ กัน โดยภาพจะมีการเหลื่อมกันอยู่เล็กน้อย และมีการกระพริบ (ความถี่ในการแสดงผลสูงมากพอที่จะทำให้ตาไม่สามารถแยกได้ว่าเป็นคนละภาพกัน) การจะมองเห็นภาพที่เหลื่อมกันอยู่เป็นภาพเดียวในแบบสเตอริโอได้นั้นจะต้องมีอุปกรณ์ช่วย ซึ่งอาจจะเป็นแว่นแบบสลับภาพ (Shutter Glass) ที่จะทำหน้าที่ในการปิดตาข้างใดข้างหนึ่งเมื่อภาพที่ส่งมาเป็นภาพจากกล้องอีกข้าง เช่น ภาพที่ส่งมาเป็นของกล้องด้านซ้าย เลนส์ทางด้านตาขวาของผู้สวมใส่ก็จะถูกปิดโดยใช้สีดำ ทำให้มีเพียงตาซ้ายเท่านั้นที่มองเห็นภาพดังกล่าว สลับกันไปเรื่อยๆ เช่นนี้โดยให้สัมพันธ์กับอัตราการวาดภาพบนหน้าจอ ก็จะทำให้ผู้สวมใส่มองเห็นรวมกันเป็นภาพเดียวแบบสเตอริโอได้ ซึ่งในที่นี้อุปกรณ์แสดงภาพแบบสวมศีรษะจะทำหน้าที่ดังกล่าวทำให้ผู้เล่นมองเห็นภาพจากทั้งสองกล้องในแบบสเตอริโอที่รับรู้ถึงความลึกของสิ่งต่างๆ ในภาพได้ รูปที่ 5.11 แสดงการสร้างภาพแบบสเตอริโอ

ในด้านการพัฒนาโปรแกรมสำหรับขั้นตอนนี้ จะใช้ฟังก์ชัน `cvCreateCameraCapture` ของ `OpenCV` ในการติดต่อกับกล้องทั้งสองตัวและนำภาพจากกล้องแต่ละตัวมาเก็บไว้ในโครงสร้างแบบ `IplImage` ซึ่งเป็นโครงสร้างในการเก็บข้อมูลรูปภาพของ `OpenCV` ด้วยฟังก์ชัน `cvQueryFrame` จากนั้นนำภาพของแต่ละกล้องที่ได้มาแปลงให้เป็นโครงสร้างข้อมูลแบบพื้นผิว (Texture) ของ `OpenGL` ด้วยฟังก์ชันที่เขียนขึ้นเอง ขั้นตอนต่อไปเป็นการสร้างภาพที่ต้องการด้วย `OpenGL` โดยการเลือกบัฟเฟอร์ที่จะใช้เก็บภาพเสียก่อน โดยการใช้ฟังก์ชัน `glDrawBuffer` แล้วกำหนดพารามิเตอร์ให้เป็นบัฟเฟอร์ข้างซ้าย (`GL_BACK_LEFT`) แล้ววาดแผ่นสี่เหลี่ยมที่มีพื้นผิวเป็นภาพจากกล้องด้านซ้าย เสร็จแล้วใช้ฟังก์ชัน `glDrawBuffer` กำหนดพารามิเตอร์ให้เป็นบัฟเฟอร์ข้างขวา (`GL_BACK_RIGHT`) แล้ววาดแผ่นสี่เหลี่ยมที่มีพื้นผิวเป็นภาพจากกล้องด้านขวา โดย

แผ่นสีเหลืองนี้จะมีตำแหน่งในแวนอนห่างจากแผ่นสีเหลืองด้านซ้ายอยู่เล็กน้อย ขั้นตอนสุดท้ายคือการใช้ฟังก์ชัน `glutSwapBuffers` ของ `glut` ในการสลับบัฟเฟอร์ด้านหลังไปเป็นบัฟเฟอร์ด้านหน้าซึ่งจะทำให้ภาพของแผ่นสีเหลืองทั้งสองนี้ถูกวาดสลับในอัตราที่สัมพันธ์กับการกะพริบ ปิดเปิดจอแสดงภาพแต่ละข้างของอุปกรณ์แสดงภาพแบบสเตอริโอ

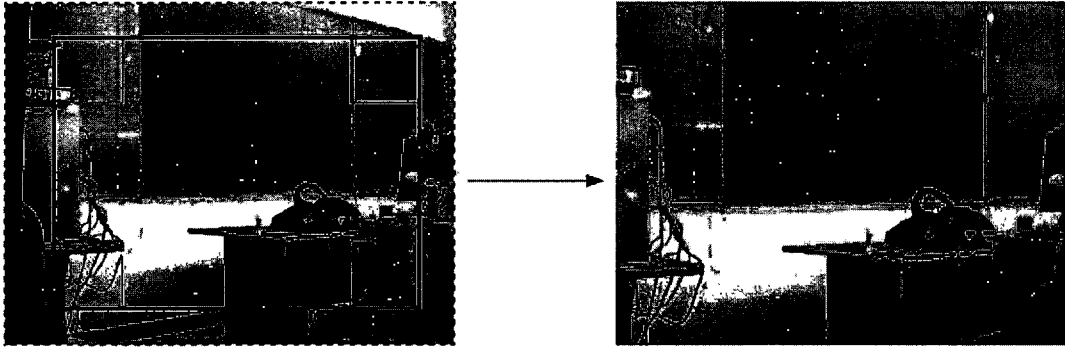


รูปที่ 5.11 การสร้างภาพแบบสเตอริโอ

#### 5.4 การแก้ปัญหาหระหะในการแสดงภาพของอุปกรณ์แสดงภาพแบบสเตอริโอ

เนื่องจากระหว่างกระจกที่เรามองภาพกับจอแสดงภาพของอุปกรณ์แสดงภาพแบบสเตอริโอนั้นมีระยะห่างพอสมควร ทำให้ภาพที่ปรากฏดูห่างออกไปจากตำแหน่งสายตา ซึ่งแตกต่างกับภาพที่มองเห็นด้วยตาตนเอง ในที่นี้แก้ไขโดยการทำให้โปรแกรมสามารถดึงภาพมาขยายให้ใหญ่ขึ้นในระยะเวลาที่ต้องการได้ซึ่งจะทำให้ภาพที่มองดูไกลออกไปนั้นถูกดึงเข้ามาใกล้มากขึ้น ดังรูปที่ 5.12 โดยผู้เล่นสามารถปรับระยะในการขยายภาพได้ตามต้องการ ช่วยให้ภาพที่มองผ่านอุปกรณ์แสดงภาพแบบสเตอริโอใกล้เคียงกับการมองด้วยสายตาตนเองมากขึ้น

การขยายภาพทำได้โดยการเลือกพื้นที่บางส่วนของภาพจากกล้อง โดยพื้นที่นี้จะถูกกำหนดด้วยระยะการขยายเข้าออกโดยผู้เล่น ส่วนที่ถูกเลือกนี้จะนำมาขยายให้มีขนาดเท่ากับความละเอียดของภาพเดิม ซึ่งในการพัฒนาโปรแกรมนั้นจะใช้ฟังก์ชัน `cvSetImageROI` ในการกำหนดพื้นที่บางส่วนของรูปภาพที่ได้มาจากกล้อง จากนั้นทำการคัดลอกข้อมูลในส่วนที่เลือกไปเก็บไว้เป็นภาพใหม่ โดยใช้ฟังก์ชัน `cvCopy` จากนั้นขยายภาพใหม่นี้ให้มีขนาดเท่ากับภาพเดิมโดยใช้ฟังก์ชัน `cvResize`



รูปที่ 5.12 การขยายภาพเฉพาะส่วน

### 5.5 การซ้อนภาพฉากหลังแบบบลูสกรีน (Blue Screen Matting)

การซ้อนภาพฉากหลังคือการแทนที่บริเวณที่เป็นฉากหลัง (Background) ของภาพที่ถ่ายได้ให้เป็นภาพฉากหลังใหม่ โดยคงส่วนที่เป็นวัตถุของภาพ (Foreground) เอาไว้ เทคนิคที่นิยมใช้กันมากคือเทคนิคการซ้อนภาพฉากหลังแบบบลูสกรีน [29] โดยจะต้องจัดเตรียมฉากที่เป็นสีใดสีหนึ่งทั้งหมด ซึ่งโดยส่วนใหญ่จะเป็นสีฟ้าหรือสีเขียว แล้วให้วัตถุอยู่หน้าฉากนี้ ภาพที่ถ่ายได้ออกมาจะเป็นภาพที่มีสีพิกเซลของทั้งส่วนที่เป็นวัตถุและฉากหลังรวมกัน ดังสมการ 5.1

$$u_c = \alpha u_f + (1 - \alpha) u_b \quad (5.1)$$

$u_c$  คือสีพิกเซลของวัตถุที่ซ้อนทับกับฉากหลัง

$u_f$  คือสีพิกเซลของวัตถุ

$u_b$  คือสีพิกเซลของฉากหลัง

$\alpha$  ค่าความโปร่งแสง (มีค่าตั้งแต่ 0-1)

จากสมการนี้ หากเราทราบสีพิกเซลของฉากหลัง ก็จะสามารถแยกส่วนที่เป็นวัตถุและส่วนที่เป็นฉากหลังออกจากกันได้ โดยค่าความโปร่งแสงนั้นเป็นตัวที่ใช้กำหนดว่าจะให้ฉากหลังมีความโปร่งแสงเท่าใด หากกำหนดเป็น 1 นั่นคือให้ฉากหลังโปร่งแสงทั้งหมด ซึ่งจะทำให้ฉากหลังถูกแทนที่ด้วยฉากหลังใหม่อย่างสมบูรณ์

สิ่งที่สำคัญสำหรับขั้นตอนนี้ก็คือการหาพิกเซลที่เป็นส่วนฉากหลังของภาพ โดยวัดได้จากค่าสีของแต่ละพิกเซลว่าเป็นสีของฉากหลังซึ่งเป็นสีฟ้าหรือไม่ ซึ่งแบบจำลองของสีที่ใช้กันโดยทั่วไปคือแบบจำลอง RGB ซึ่งประกอบไปด้วยแม่สีหลัก 3 สี คือ แดง เขียว น้ำเงิน โดยสีที่เกิดจากแบบจำลองนี้ก็เกิดจากการผสมแม่สีทั้งสามเข้าด้วยกันด้วยความเข้มของแต่ละแม่สีที่ต่างกัน อย่างไรก็ตามแบบจำลองนี้ค่อนข้างจะนำมาใช้ในการหาส่วนที่เป็นฉากหลังสีฟ้าได้ยาก เพราะ

เนื่องจากสีฟ้าที่มีในภาพนั้นเป็นสีฟ้าที่มีความเข้มหลายระดับ ซึ่งสีฟ้าที่มีความเข้มในแต่ละระดับนี้จะถูกผสมด้วยค่า RGB ที่แตกต่างกันมาก จึงเป็นการยากที่จะกำหนดช่วงของค่า RGB ที่เป็นโทนสีฟ้าได้อย่างแม่นยำ ดังนั้นการใช้แบบจำลองแบบอื่นจึงมีความเหมาะสมกว่า ซึ่งในที่นี้จะใช้แบบจำลอง HSV โดยแบบจำลองนี้ประกอบไปด้วยค่า Hue ที่ใช้กำหนดโทนสี ค่า Saturation ที่ใช้บอกความสดใสที่เกิดจากแสงสีขาว และค่า Value ที่หมายถึงความเข้มแสง แบบจำลองแบบนี้ใกล้เคียงกับการรับรู้สีของมนุษย์มากกว่าแบบจำลอง RGB และค่า Hue เป็นค่าที่ใช้กำหนดโทนสี ดังนั้นหากทราบว่าโทนสีฟ้าในแบบจำลองนี้มีค่าอยู่ระหว่างค่าใดแล้ว ก็จะสามารถหาพิกเซลที่เป็นโทนสีฟ้าเป็นของฉากหลังได้อย่างแม่นยำมากขึ้น สำหรับการพัฒนาโปรแกรมนั้นเนื่องจากภาพที่เก็บด้วยโครงสร้าง IplImage ของ OpenCV นั้นจะถูกจัดเก็บด้วยแบบจำลอง BGR (ซึ่งคล้ายกับแบบจำลอง RGB เพียงแต่เปลี่ยนลำดับแม่สี) จึงต้องใช้ฟังก์ชันในการแปลงค่าสีจากแบบจำลอง BGR ไปเป็นแบบจำลอง HSV เสียก่อน แล้วจึงนำค่าสีในแต่ละพิกเซลของภาพที่ได้จากกล้องมาตรวจสอบว่าเป็นค่าสีในโทนสีฟ้าหรือไม่ หากใช่ ให้แทนที่ค่าสีของพิกเซลนั้นด้วยค่าสีของภาพฉากหลังใหม่ที่ตำแหน่งพิกเซลเดียวกัน โดยภาพฉากหลังให้จะเป็นภาพรั้นค่าจำลองซึ่งภาพที่ได้จะเกิดจากขั้นตอนการสร้างภาพด้วยวิธีการ Image-Based Rendering ซึ่งจะกล่าวถึงในหัวข้อถัดไป

### 5.6 การสร้างภาพ 3 มิติโดยใช้ภาพถ่าย (Image-Based Rendering)

การสร้างภาพ 3 มิติโดยใช้ภาพถ่าย (Image-Based Rendering) [30] เป็นวิธีการในการสร้างภาพใหม่จากชุดของภาพถ่ายที่มีอยู่ นั่นคือ เมื่อเราถ่ายภาพสถานที่หรือวัตถุในเวลาหนึ่งๆ เป็นจำนวนหลายๆ ภาพ โดยแต่ละภาพจะถูกถ่ายด้วยกล้องที่วางอยู่ในตำแหน่งและ/หรือทิศทางของกล้องที่แตกต่างกัน หากเราต้องการสร้างภาพสถานที่หรือวัตถุเดิมจากตำแหน่งและทิศทางของกล้องที่แตกต่างไปจากที่ถ่ายมา เราสามารถสร้างได้โดยการประมาณค่าโดยอาศัยข้อมูลจากรูปภาพที่ได้ถ่ายมาแล้ว และในทางทฤษฎีหากเราสามารถสร้างภาพที่ตำแหน่งและทิศทางของกล้องใดๆ ได้ จะทำให้เราสามารถรับรู้ความลึกของวัตถุที่อยู่ในภาพนั้น ซึ่งอาจกล่าวได้ว่าเป็นการสร้างภาพ 3 มิติ

ข้อดีของวิธีการนี้คือ

- ใช้การคำนวณไม่มาก จึงเหมาะสมสำหรับการใช้กับแอปพลิเคชันที่ตอบสนองในเวลาจริง (Real-Time)

- เวลาที่ใช้ในการสร้างภาพ ไม่ขึ้นกับความซับซ้อนของฉาก

- ชุดของภาพที่จะนำมาใช้สามารถสังเคราะห์จากคอมพิวเตอร์หรือใช้ภาพถ่ายจริงตามธรรมชาติก็ได้ และยังสามารถที่จะนำภาพจากทั้งสองแหล่งมาผสมกันได้

ข้อเสียของวิธีการนี้คือ

- การสร้างภาพในทุกตำแหน่งทุกทิศทางของกล้องนั้นต้องใช้ชุดของภาพเป็นจำนวนมาก ซึ่งเป็นเรื่องยากในทางปฏิบัติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ต้องใช้หน่วยความจำเป็นจำนวนมากในการจัดเก็บภาพ
- เวลาที่ใช้ในการสร้างภาพขึ้นอยู่กับความละเอียดของภาพ
- เนื่องจากขาดข้อมูลทางด้านรูปทรงและตำแหน่งของวัตถุในฉาก ทำให้การเปลี่ยนแปลงภาพในฉากและการสร้างปฏิสัมพันธ์ต่อผู้ใช้ทำได้ยาก

จากข้อดีและข้อเสียดังกล่าว โปรแกรมจำลองสถานการณ์ที่สร้างขึ้นจึงเลือกใช้การสร้างภาพฉากหลังด้วยวิธีการสร้างภาพ 3 มิติโดยใช้ภาพถ่าย และสร้างส่วนที่ต้องมีปฏิสัมพันธ์กับผู้ใช้ด้วยวิธีการสร้างภาพแบบดั้งเดิมโดยใช้โมเดล

## 5.7 เทคนิค Light Field เพื่อการสร้างภาพ 3 มิติจากภาพถ่าย

### 5.7.1 ความรู้พื้นฐานเกี่ยวกับ Light Field Rendering

Light Field Rendering [31] เป็นเทคนิคหนึ่งในการสร้างภาพ 3 มิติจากภาพถ่าย ใช้แนวคิดที่ว่าหากต้องการจะเก็บข้อมูลของลำแสงในโลก 3 มิติ จะต้องใช้ระนาบ 2 ระนาบที่ขนานกันไปวางไว้ ซึ่งจุดตัดของลำแสงบนระนาบทั้งสองนี้จะเป็นพารามิเตอร์ของฟังก์ชัน 4 มิติที่สามารถใช้ระบุตำแหน่งในการหาค่าความสว่างของลำแสงนั้นๆ ได้

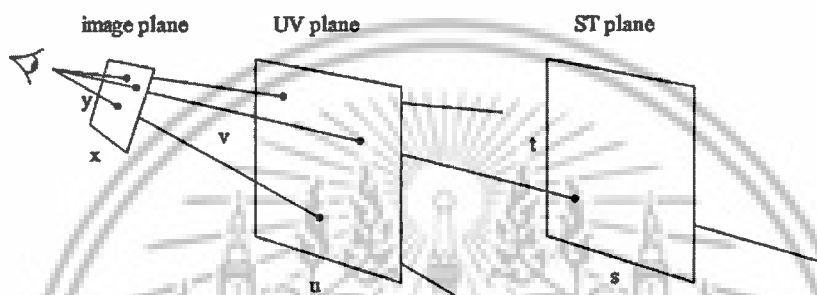
ระนาบทั้งสองประกอบไปด้วยระนาบของกล้อง (camera plane) ซึ่งแทนด้วย  $(u, v)$  ระนาบนี้จะเป็นตำแหน่งของกล้องที่ใช้ถ่ายภาพกลุ่มตัวอย่างทั้งหมด โดยจะวางในลักษณะคล้ายช่องตารางที่เท่า ๆ กัน ส่วนระนาบที่สองคือระนาบของภาพ (focal plane) ซึ่งแทนด้วย  $(s, t)$  ระนาบนี้จะเป็นระนาบของภาพที่จะถูกฉายจากกล้องในระนาบ  $(u, v)$  ทั้งหมด ดังนั้นฟังก์ชันของ Light Field สามารถเขียนเป็นฟังก์ชัน 4 มิติได้ดังสมการ 5.2

$$L = (u, v, s, t) \quad (5.2)$$

นั่นคือฟังก์ชันของ Light Field จะถูกกำหนดด้วยตำแหน่ง  $u, v, s, t$

การถ่ายภาพนั้น สามารถทำได้ทั้งจากการถ่ายภาพในธรรมชาติด้วยกล้องจริง และการถ่ายภาพสังเคราะห์จากกล้องเสมือน เนื่องจากวิธีการนี้ไม่ได้อาศัยข้อมูลของรูปทรง ทำให้จำเป็นต้องวางกล้องให้อยู่ในระนาบเดียวกันและมีระยะห่างระหว่างกล้องเท่าๆ กัน เพื่อให้ตัวอย่างที่ได้มีความสม่ำเสมอ (uniform) วิธีการนี้ยังชุดภาพมีจำนวนมากและระยะห่างระหว่างกล้องมีค่าน้อย จะยิ่งทำให้ภาพที่สร้างขึ้นใหม่มีคุณภาพที่ดี หรือหากระยะห่างระหว่างกล้องมีค่ามากก็จำเป็นต้องใช้กล้องที่มีความยาวโฟกัสสั้นเพื่อให้ปริมาตรการมองเห็น (Field of View) เพิ่มมากขึ้น นอกจากนี้แล้วหากการถ่ายภาพไม่สามารถถ่ายพร้อมกันได้ทุกกล้อง ก็จำเป็นต้องควบคุมปริมาณของแสงของฉากที่จะถ่ายให้คงที่

การสร้างภาพในตำแหน่งและทิศทางใหม่ที่ไม่มีในชุดภาพนั้น ทำได้โดยการลากเส้นจากจุดมองไปยังจุดกึ่งกลางของแต่ละพิกเซลในระนาบของรูปภาพที่ต้องการแสดงผล (image plane) เส้นที่ลากไปตัดระนาบ  $(u, v)$  นั้นจะทำให้ได้ตำแหน่งบนระนาบ  $(u, v)$  และจุดตัดของเส้นกับระนาบ  $(s, t)$  นั้นจะทำให้ทราบตำแหน่งบนระนาบ  $(s, t)$  ค่า  $u, v, s, t$  ที่ได้มานี้จะต้องนำมาหาต่อไปว่าตำแหน่งนี้ใกล้เคียงกับพิกเซลใดจากกล้องใดบ้าง จากนั้นใช้วิธีการประมาณค่าแบบใดแบบหนึ่งเพื่อที่จะหาค่าสีของพิกเซลที่ต้องการอย่างเหมาะสม ทำเช่นนี้กับทุกๆ พิกเซลก็จะได้ภาพจากตำแหน่งและทิศทางของกล้องที่ต้องการ ดังแสดงในรูปที่ 5.13

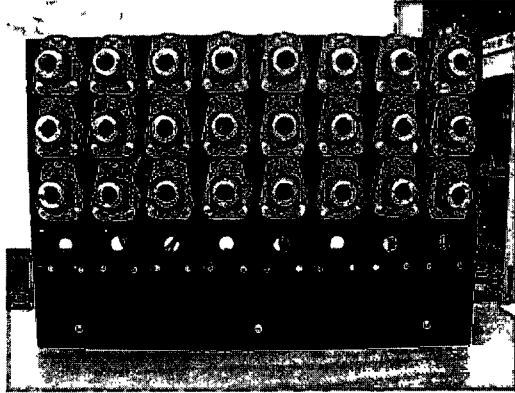


รูปที่ 5.13 Light Field Rendering [32]

สำหรับโปรแกรมนี้จะใช้วิธีการ Light Field ในการสร้างภาพจากหลังของร้านค้าเพื่อนำไปซ้อนกับฉากหลังสีฟ้า แต่การจะทำเช่นนี้ได้จำเป็นต้องทราบตำแหน่งและทิศทางของผู้เล่นซึ่งจะกล่าวถึงในหัวข้อถัดไป

### 5.7.2 การออกแบบและสร้างอุปกรณ์ในการถ่ายภาพ

กล้องที่ใช้ถ่ายภาพต้องมีจำนวนมากพอสมควรและเป็นกล้องรุ่นเดียวกัน ในที่นี้จึงเลือกใช้เว็บแคมเนื่องจากมีราคาถูกและมีขนาดเล็ก โดยเลือกใช้กล้อง AlphaCam M2 (MJ-541) จำนวน 24 ตัว โดยเมื่อได้กล้องมาแล้วจะต้องนำกล้องมาติดตั้งเข้ากับแผ่นพลาสติกที่เจาะรูเตรียมเอาไว้ เพื่อยึดให้กล้องแต่ละตัวอยู่ในระนาบเดียวกัน และวางในระยะห่างเท่าๆ กัน โดยกำหนดให้วางกล้องไว้ 8 ตัวตามแนวนอน เป็นจำนวน 3 แถว จุดศูนย์กลางของกล้องจะวางในระยะห่าง 3.5 เซนติเมตรตามแนวนอน และ 4.5 เซนติเมตรตามแนวตั้ง จากนั้นนำสาย USB ที่อยู่ด้านหลังของกล้องแต่ละตัวมาเชื่อมต่อเข้ากับเครื่องคอมพิวเตอร์ แต่เนื่องจากพอร์ต USB ของคอมพิวเตอร์แต่ละเครื่องนั้นมีจำกัด จึงต้องนำสาย USB จากกล้องแต่ละตัวมาต่อเข้ากับ USB Hub แล้วจึงนำสาย USB จาก USB Hub ต่อเข้ากับคอมพิวเตอร์ อุปกรณ์ถ่ายภาพที่ได้ เป็นดังรูปที่ 5.14

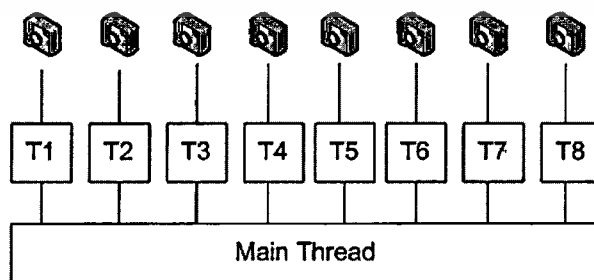


รูปที่ 5.14 อุปกรณ์ในการถ่ายภาพ

### 5.7.3 โปรแกรมในการถ่ายภาพ

การถ่ายภาพจากกล้องเป็นจำนวนมากพร้อมๆ กันนี้จำเป็นต้องพัฒนาโปรแกรมเพื่อใช้ในการถ่ายภาพในลักษณะนี้โดยเฉพาะ โดยโปรแกรมดังกล่าวจะต้องสามารถแสดงภาพจากกล้องแต่ละตัวให้ผู้ใช้ได้เห็น และเมื่อกดคีย์ที่ใช้สำหรับการถ่ายภาพ ภาพจากกล้องทั้งหมดจะถูกจัดเก็บลงเป็นไฟล์รูปภาพเพื่อนำไปใช้ต่อไป อย่างไรก็ตามเนื่องจากข้อจำกัดทั้งในด้านการแสดงภาพของเว็บแคมที่ใช้ซึ่งมีความเร็วเพียง 30 ภาพต่อวินาที และข้อจำกัดของบัส USB 2.0 ทำให้การจัดเก็บภาพจากกล้องหลายตัวพร้อมกันเป็นสิ่งที่ทำได้ยาก สำหรับปัญหาแรกในเรื่องความเร็วของเว็บแคมสามารถแก้ปัญหาได้บ้างด้วยการแบ่งการทำงานของโปรแกรมออกเป็นหลายๆ เธรด (Multithread) โดยให้กล้องแต่ละตัวทำงานเป็นหนึ่งเธรด ทำให้การรับภาพจากแต่ละกล้องนั้นไม่จำเป็นต้องรอกการทำงานจากกล้องอื่น ดังแสดงในรูปที่ 5.15 หากโปรแกรมมีเพียงเธรดเดียวโปรแกรมจะทำงานได้ช้ามาก เพราะต้องรอให้กล้องทุกตัวส่งภาพมาให้เสร็จสิ้นเสียก่อน เธรดการทำงานหลักของโปรแกรมจึงจะสามารถทำงานต่อไปได้

อย่างไรก็ดีเนื่องจากข้อจำกัดของบัส USB 2.0 ทำให้ยังไม่สามารถถ่ายภาพจากกล้องทั้ง 24 ตัวได้พร้อมกัน เพราะจะมีปัญหากกล้องบางตัวไม่สามารถรับภาพและส่งภาพเข้ามาได้ทัน จึงต้องแบ่งถ่ายภาพไปที่ละแถว ซึ่งมีแถวละ 8 ตัว



รูปที่ 5.15 การแบ่งการทำงานออกเป็นหลายๆ เธรด

อุปกรณ์ที่ใช้ในการถ่ายภาพจะใช้ในสร้าง Light Field ให้กับฉากจำลอง โดยการถ่ายภาพจากกล้องทั้ง 24 ไปยังร้านจำลองที่สร้างขึ้น ภาพที่ได้จากการถ่ายนี้จะถูกนำไปใช้ในการสร้างภาพใหม่ต่อไป

#### 5.7.4 การปรับเทียบกล้อง (Camera Calibration)

เนื่องจากกล้องแต่ละตัวอาจมีความคลาดเคลื่อนทั้งตำแหน่งและทิศทางตามการติดตั้ง อีกทั้งยังอาจเกิดจากความคลาดเคลื่อนของเลนส์ ทำให้ภาพที่ถ่ายได้อาจไม่เป็นภาพที่ถ่ายจากกล้องที่อยู่ในระนาบเดียวกันตามทฤษฎี นอกจากนี้แล้วตำแหน่งของกล้องตามความเป็นจริงนั้นจะมีหน่วยวัดที่แตกต่างไปจากตำแหน่งของกล้องในโลกเสมือนของ โปรแกรมที่สร้างขึ้น ดังนั้นการปรับเทียบกล้องจึงเป็นสิ่งจำเป็น การปรับเทียบกล้องจะทำให้เราทราบว่าภาพแต่ละภาพนั้นเกิดจากกล้องที่วางอยู่ในตำแหน่งและทิศทางใด รวมทั้งทราบความยาวโฟกัสของกล้องในหน่วยพิกเซล ซึ่งค่าที่ได้จากการปรับเทียบกล้องนี้จะถูกนำไปใช้ในขั้นตอนการสร้างภาพต่อไป โดยสำหรับ โปรแกรมนี้ จำเป็นต้องมีการปรับเทียบกล้องทั้งกล้องที่ใช้สำหรับการตรวจจับวัตถุ กล้องที่ใช้หาตำแหน่งของผู้เล่น และกล้องที่ใช้สำหรับการสร้างภาพด้วยเทคนิค Light Field

ในที่นี้การปรับเทียบกล้อง ทำได้โดยการใช้โปรแกรม GML C++ Camera Calibration [33] Toolbox โดยการสร้างตารางหมากกรุกที่มีขนาดและจำนวนตามที่โปรแกรมกำหนด เพื่อให้เป็นแบบให้โปรแกรมสามารถจำแนกตำแหน่งของตารางหมากกรุกที่เปลี่ยนไปในแต่ละภาพ ซึ่งจะช่วยให้ทราบตำแหน่งและทิศทางที่เปลี่ยนแปลงในแต่ละภาพได้

#### 5.7.5 การสร้างภาพใหม่ (Image Reconstruction)

จากชุดภาพที่ถ่ายมาได้ นั้นเป็นเพียงภาพจากกล้องในตำแหน่งหนึ่งๆ เท่านั้น แต่สำหรับการสร้างภาพในโลก 3 มิติ นั้นจะต้องสามารถสร้างภาพไม่ว่ากล้องจะไปอยู่ในตำแหน่งหรือทิศทางใดๆ ก็ได้ ดังนั้นจากข้อมูลชุดภาพที่มีอยู่จะถูกนำมาใช้ในการสร้างภาพใหม่เมื่อตำแหน่งของกล้องเปลี่ยนแปลงไปซึ่งในที่นี้ก็คือ เมื่อตำแหน่งของผู้เล่นเปลี่ยนแปลงไปภาพฉากจำลองนี้ก็ต้องเปลี่ยนแปลงตามให้สัมพันธ์กันด้วย

เริ่มต้นจากการหาตำแหน่งของผู้เล่น ซึ่งจะกล่าวถึงในหัวข้อ 5.8 เมื่อทราบตำแหน่งของผู้เล่นในโลก 3 มิติ (World Coordinate) แล้ว ก็จะใช้ตำแหน่งนี้เป็นจุดศูนย์กลางการ โปรเจกชัน (Center of Projection) ของกล้อง ขั้นตอนต่อไปคือการสร้างระนาบของรูปภาพที่ต้องการจะสร้างใหม่ (Image Plane) ในโลก 3 มิติ โดยสิ่งที่ต้องรู้ก็คือ ความยาวโฟกัสของกล้อง ความละเอียดของภาพที่จะสร้าง และขนาดของมุมในการมองของกล้องในแนวนอน (Horizontal Field of View) จากนั้นนำค่าต่างๆ เหล่านี้มาคำนวณหาความสูงและความกว้างของระนาบรูปภาพที่ต้องการจะสร้าง โดยก่อนอื่นให้คำนวณหาอัตราส่วนระหว่างแนวตั้งและแนวนอน (aspect ratio) ของรูปภาพก่อน โดยหาได้จากสมการ 5.3

$$aspect = w/h \quad (5.3)$$

$w$  คือ ความกว้างของภาพ

$h$  คือ ความสูงของภาพ

$aspect$  คือ อัตราส่วนระหว่างแนวดิ่งและแนวนอน

จากนั้น คำนวณหาความสูงและความกว้างของระนาบรูปภาพที่ต้องการจะสร้าง จากสมการ 5.4 และ 5.5 [34]

$$H = f \times \tan\left(\frac{\theta}{2}\right) \quad (5.4)$$

$$W = H \times aspect \quad (5.5)$$

$H$  คือ ความสูงครึ่งหนึ่งของระนาบรูปภาพ

$W$  คือ ความกว้างครึ่งหนึ่งของระนาบรูปภาพ

$f$  คือ ความยาวโฟกัสของกล้อง

$\theta$  คือ มุมในการมองแนวนอนของกล้อง

เนื่องจากจุดกึ่งกลางการโปรเจกชันอยู่ที่กึ่งกลางของระนาบรูปภาพ ดังนั้นจากจุดกึ่งกลางการโปรเจกชันไปยังขอบบนในแนวดิ่งของระนาบรูปภาพจึงมีความสูงเป็น  $H$  และจุดกึ่งกลางการโปรเจกชันไปยังขอบล่างในแนวดิ่งของระนาบรูปภาพจึงมีความสูงเป็น  $-H$  ซึ่งเป็นไปในทำนองเดียวกันกับในแนวนอน

ขั้นตอนต่อไปเป็นการแปลงตำแหน่งของพิกเซลในรูปภาพ (Pixel Coordinate) ที่ต้องการจะสร้างไปเป็นตำแหน่งบนระนาบรูปภาพ โดยใช้สมการ 5.6 และ 5.7

$$u_c = -W + W \frac{2c}{w} \quad (5.6)$$

$$v_r = -H + H \frac{2r}{h} \quad (5.7)$$

$u_c$  คือ ตำแหน่งในแนวนอนของพิกเซลบนระนาบรูปภาพ

$v_r$  คือ ตำแหน่งในแนวดิ่งของพิกเซลบนระนาบรูปภาพ

$c$  คือ ตำแหน่งในแนวนอนของพิกเซลใน pixel coordinate

$r$  คือ ตำแหน่งในแนวดิ่งของพิกเซลใน pixel coordinate

เสร็จแล้วแปลงตำแหน่งของพิกเซลบนระนาบรูปภาพไปเป็นตำแหน่งในโลก 3 มิติ โดยใช้

สมการ 5.8 5.9 และ 5.10

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$x_c = u_c + p_x \quad (5.8)$$

$$y_r = v_r + p_y \quad (5.9)$$

$$z = -f + p_z \quad (5.10)$$

$x_c$  คือ ตำแหน่งในแนวนอนของพิกเซลในโลก 3 มิติ

$y_r$  คือ ตำแหน่งในแนวตั้งของพิกเซลในโลก 3 มิติ

$z$  คือ ตำแหน่งในแนวลึกของพิกเซลในโลก 3 มิติ

$p_x$   $p_y$   $p_z$  คือ ตำแหน่งของจุดศูนย์กลางโปรเจกชันในแนวนอน แนวตั้ง และแนวลึกตามลำดับ

เมื่อได้ตำแหน่งของแต่ละพิกเซลในโลก 3 มิติแล้วก็สามารถสร้างเวกเตอร์ทิศทาง (direction vector) ที่ลากจากจุดศูนย์กลางโปรเจกชันไปยังแต่ละจุดพิกเซลบนระนาบรูปภาพ ได้จากสมการ 5.11

$$dir_{rc} = p_0 + q_{rc} \quad (5.11)$$

$dir_{rc}$  คือ เวกเตอร์ทิศทางของพิกเซล  $rc$

$p_0$  คือ ตำแหน่งจุดศูนย์กลางโปรเจกชัน

$q_{rc}$  คือ ตำแหน่งของพิกเซล  $rc$  ในโลก 3 มิติ

เมื่อได้เวกเตอร์ทิศทางของแต่ละพิกเซลแล้ว ขั้นตอนต่อไปเป็นการหาจุดตัดของเวกเตอร์เมื่อเคลื่อนที่ไปยังระนาบ  $uv$  และระนาบ  $st$  ซึ่งสามารถหาได้จากสูตร

$$t = -(p_0 \cdot N + d) / (dir_{rc} \cdot N) \quad (5.12)$$

$$p = p_0 + tV \quad (5.13)$$

$N$  คือ Normal Vector ของระนาบ  $uv$  หรือ  $st$

$d$  คือ ระยะห่างที่ตั้งฉากระหว่างจุดกำเนิด (origin) กับระนาบ  $uv$  หรือ  $st$

$p$  คือ จุดตัดบนระนาบ  $uv$  หรือ  $st$

เมื่อได้จุดตัดบนระนาบทั้งสองแล้ว ขั้นตอนถัดไปเป็นการแปลงตำแหน่ง  $u, v, s, t$  ที่ได้กลับมาเป็นตำแหน่งของกล้องบนระนาบ  $u, v$  และตำแหน่งของพิกเซลบนระนาบ  $s, t$  แล้วทำการประมาณค่าของพิกเซลที่ต้องการ โดยวิธีการประมาณค่านั้นมีหลายวิธี วิธีการอย่างง่าย ๆ นั่นคือดูว่าตำแหน่งของ  $u, v$  ใกล้กับตำแหน่งของกล้องไหนก็เลือกพิกเซลจากกล้องนั้น และดูว่าตำแหน่งของ

$s, t$  นั้นใกล้กับตำแหน่งของพิกเซลไหนก็ใช้พิกเซลนั้น เรียกวิธีการนี้ว่า Nearest Neighbor Interpolation ซึ่งเป็นวิธีการที่ให้ผลที่ไม่ดีนัก วิธีการที่ดีกว่าคือการประมาณค่าจากจุดที่ใกล้กับจุดที่ต้องการจำนวน 4 จุด แล้วนำค่าของพิกเซลจาก 4 จุดนี้มาเฉลี่ยกันและถ่วงน้ำหนักด้วยระยะทาง ซึ่งเรียกวิธีการนี้ว่า Bilinear Interpolation จำนวนได้จากสมการ 5.14

$$\begin{aligned}
 f(x, y) \approx & \frac{f(Q_{11})}{(x_2 - x_1)(y_2 - y_1)} (x_2 - x)(y_2 - y) \\
 & + \frac{f(Q_{21})}{(x_2 - x_1)(y_2 - y_1)} (x - x_1)(y_2 - y) \\
 & + \frac{f(Q_{12})}{(x_2 - x_1)(y_2 - y_1)} (x_2 - x)(y - y_1) \\
 & + \frac{f(Q_{22})}{(x_2 - x_1)(y_2 - y_1)} (x - x_1)(y - y_1)
 \end{aligned} \tag{5.14}$$

$f(x, y)$  คือ ค่าสีที่ต้องการประมาณ

$Q_{11}$  คือ ค่าสีของพิกเซลในจุดที่ใกล้เคียงซึ่งอยู่มุมล่างซ้าย

$Q_{21}$  คือ ค่าสีของพิกเซลในจุดที่ใกล้เคียงซึ่งอยู่มุมล่างขวา

$Q_{12}$  คือ ค่าสีของพิกเซลในจุดที่ใกล้เคียงซึ่งอยู่มุมบนซ้าย

$Q_{22}$  คือ ค่าสีของพิกเซลในจุดที่ใกล้เคียงซึ่งอยู่มุมบนขวา

$x$  คือ ตำแหน่งในแนวแกนอนของจุดที่ต้องการประมาณ

$y$  คือ ตำแหน่งในแนวแกนตั้งของจุดที่ต้องการประมาณ

$x_1$  คือ ตำแหน่งขวาสุดในแนวแกนอนของจุดใกล้เคียง

$x_2$  คือ ตำแหน่งซ้ายสุดในแนวแกนอนของจุดใกล้เคียง

$y_1$  คือ ตำแหน่งล่างสุดในแนวแกนตั้งของจุดใกล้เคียง

$y_2$  คือ ตำแหน่งบนสุดในแนวแกนตั้งของจุดใกล้เคียง

## 5.8 การหาตำแหน่งของผู้เล่น

การหาตำแหน่งและทิศทางการมองของผู้เล่นนั้นทำได้หลายวิธี เช่น ใช้อุปกรณ์ตรวจจับตำแหน่ง แต่ในที่นี้จะหาค่าพิกเซลของเว็บแคมที่ติดตั้งเอาไว้บนอุปกรณ์แสดงผลแบบสวมศีรษะ แล้วใช้กล้องอีก 2 ตัวที่วางในด้านตรงข้ามกับผู้เล่นจับภาพหลอดไฟนี้ ภาพที่ได้จะถูกนำไปประมวลผลเพื่อหาตำแหน่งพิกเซลที่เป็นจุดกึ่งกลางของหลอดไฟ จากตำแหน่งของพิกเซลจะถูกนำไปหาตำแหน่งในระนาบรูปภาพของกล้องแต่ละตัว เมื่อได้ตำแหน่งของกล้องและตำแหน่งของหลอดไฟที่ถูกฉายลงบนระนาบรูปภาพแล้ว จะสามารถสร้างเวกเตอร์ของลำแสงนี้ที่วิ่งเข้าสู่กล้องได้ และถ้าสามารถหาจุดตัดของเวกเตอร์ทั้งสองนี้ได้ ก็จะทราบตำแหน่งของวัตถุได้

อย่างไรก็ตามโดยทั่วไปแล้วเวกเตอร์ในโลกร 3 มิตินั้นจะไม่ตัดกัน การจะหาจุดตัดของเวกเตอร์ทั้งสองนี้จึงเป็นเพียงการหาค่าประมาณของจุดตัดเท่านั้น โดยการแก้สมการเส้นตรง 5.15 และ 5.16

$$P(\alpha) = P_0 + \alpha(P_1 - P_0) \tag{5.15}$$

$$Q(\beta) = Q_0 + \beta(Q_1 - Q_0) \tag{5.16}$$

$P_0$  เป็นจุดกึ่งกลางโปรเจกชันของกล้องตัวที่ 1

$P_1$  เป็นจุดกึ่งกลางของหลอดไฟบนระนาบรูปภาพของกล้องตัวที่ 1

$Q_0$  เป็นจุดกึ่งกลางโปรเจกชันของกล้องตัวที่ 2

$Q_1$  เป็นจุดกึ่งกลางของหลอดไฟบนระนาบรูปภาพของกล้องตัวที่ 2

$\alpha, \beta$  เป็นค่าใดๆ

$P(\alpha)$  จุดใดๆ บนเส้นตรงที่แทนลำแสงจากกึ่งกลางของหลอดไฟมายังจุดกึ่งกลางโปรเจกชันของกล้องตัวที่ 1

$Q(\beta)$  จุดใดๆ บนเส้นตรงที่แทนลำแสงจากกึ่งกลางของหลอดไฟมายังจุดกึ่งกลางโปรเจกชันของกล้องตัวที่ 2

แก้สมการเพื่อหาค่า  $\alpha$  และ  $\beta$  ก็จะสามารถหาจุดใดๆ บนเส้นตรงนี้ได้ และถ้าทราบจุดบนเส้นตรงทั้งสองที่มีระยะห่างน้อยที่สุด (ลากเส้นระหว่างสองจุดนี้แล้วตั้งฉากกับเส้นตรงทั้งสอง) กึ่งกลางของจุดทั้งสองนี้คือจุดประมาณที่เส้นตรงสองเส้นนี้จะตัดกันและเป็นตำแหน่งของหลอดไฟตามที่ต้องการ

## 5.9 การตรวจจับวัตถุ

ผู้เล่นจะต้องวางวัตถุที่ต้องการซื้อลงไปบนโต๊ะ รวมทั้งวางเงินที่ใช้ซื้อสินค้า ซึ่งการที่โปรแกรมจะสามารถตรวจจับและจำแนกวัตถุต่างๆ ได้ จำเป็นจะต้องมีการติดเครื่องหมาย (marker) เอาไว้บนตัววัตถุ เครื่องหมายจะต้องมีรูปแบบที่แน่นอนและมีลักษณะที่เหมือนกัน แต่จะมีลวดลายที่แตกต่างกัน โดยขั้นตอนการทำงานของ การตรวจจับวัตถุมีดังนี้

- แปลงรูปภาพที่ได้จากกล้องให้เป็นภาพขาวดำ แล้วทำการค้นหาเครื่องหมายที่อยู่ในรูปภาพโดยอาศัยขอบสีดำของเครื่องหมาย

- กำหนดพิกัดตำแหน่งและทิศทางของเครื่องหมายที่สัมพันธ์กับตำแหน่งและทิศทางของกล้อง (ตำแหน่งของกล้องเป็นจุดกำเนิด)

- สัญลักษณ์ที่อยู่ในเครื่องหมายจะถูกนำมาจำแนกด้วยวิธีการจับคู่กับแม่แบบ (Template Matching) ที่อยู่ในหน่วยความจำ เพื่อใช้ในการหาว่าเป็นสัญลักษณ์ในภาพคือวัตถุใด

- นำเอา model-view matrix ของวัตถุที่ต้องการวาดมาคูณเข้ากับ transformation matrix ของเครื่องหมาย

- วาดภาพของวัตถุที่ตรงกับสัญลักษณ์ลงไปบนตำแหน่งของเครื่องหมาย ซึ่งภาพวัตถุนี้เป็นภาพสังเคราะห์จากคอมพิวเตอร์ เท่ากับว่าภาพที่ผู้เล่นเห็นจะเป็นภาพผสมจากคอมพิวเตอร์และภาพจากสภาพแวดล้อมจริงที่ได้จากกล้อง

ขั้นตอนการทำงานทั้งหมดแสดงดังในรูปที่ 5.16



รูปที่ 5.16 ขั้นตอนการตรวจจับวัตถุ [35]

ในการพัฒนาโปรแกรมในส่วนนี้นั้น จะใช้ไลบรารี ARToolkitPlus ในการหาตำแหน่งและทิศทางของเครื่องหมายในภาพ โดยจะต้องแปลงภาพที่ได้จากกล้องให้เป็นภาพขาวดำ โดยใช้ฟังก์ชัน cvCvtColor ของ OpenCV ในการแปลง จากนั้นตัดเฉพาะส่วนของข้อมูลแต่ละพิกเซลของภาพ (ไม่เอา header ของ IpImage) แล้วส่งไปให้ฟังก์ชัน calc ของ ARToolkitPlus ซึ่งผลลัพธ์ที่ได้จากฟังก์ชันนี้จะถูกเก็บไว้ในโครงสร้างข้อมูลแบบ ARMarkerInfo ซึ่งจะระบุจำนวนและหมายเลขของเครื่องหมายที่พบ รวมไปถึงตำแหน่งและทิศทางของเครื่องหมายอีกด้วย ส่วนการวาดรูปวัตถุลงไปในฉากนั้นจะใช้ฟังก์ชันของ OpenGL ในการวาด ซึ่งจะต้องแปลงตำแหน่งของเครื่องหมายให้มาอยู่ในรูปที่สามารถใช้กับเมทริกซ์ของ OpenGL ได้ โดยใช้ฟังก์ชัน alcOpenGLMatrixFromMarker

## 5.10 การทดลอง

การทดลองความสามารถของโปรแกรมจะแบ่งเป็นส่วนต่างๆ ดังนี้

1. การทดสอบความเร็วในการแสดงผล โปรแกรมจะต้องตอบสนองต่อผู้ใช้ได้ในเวลาจริง (Real-Time) จึงต้องมีการวัดความเร็วในการแสดงผล ซึ่งวัดได้จากเฟรมเรต

(frame rate) ของโปรแกรม ซึ่งก็คือจำนวนเฟรมที่สามารถแสดงผลได้ใน 1 วินาที เช่น 60 FPS หมายถึง สามารถแสดงผลได้ 60 เฟรมต่อ 1 วินาที หากเฟรมเรตมีค่าประมาณ 12 FPS ขึ้นไป ถือว่ามีความเร็วมากพอที่ผู้ใช้จะไม่สังเกตเห็นว่าภาพมีการกระพริบ การวัดจะใช้ค่าเฉลี่ยของเฟรมเรตในเวลา 1 นาที โดยทดลองเปิดปิดฟังก์ชันการทำงานสเตอริโอและการสร้างภาพด้วย Image-Based Rendering แล้วนำไปแทนที่ในฉากหลังสีฟ้า พบว่าค่าที่ได้เป็นดังนี้

ฟังก์ชันการทำงาน	เฟรมเรตเฉลี่ยใน 1 นาที
เปิดทุกฟังก์ชัน	9.28
ปิดเฉพาะการแสดงผลแบบสเตอริโอ	18.72
ปิดทั้งการแสดงผลแบบสเตอริโอและการแทนที่ฉากหลังสีฟ้าด้วยภาพจาก IBR	27.45

ตารางที่ 5.1 เฟรมเรตของโปรแกรม

พบว่าความเร็วในการแสดงผลนั้นยังถูกจำกัดอยู่ที่ความเร็วในการแสดงผลของเว็บแคม และเมื่อเปิดการแสดงผลแบบสเตอริโอนั้นจะทำให้เฟรมเรตลดลงถึงครึ่งหนึ่ง เพราะต้องรอภาพจากกล้องเพิ่มขึ้นอีกหนึ่งตัว ความเร็วเมื่อไม่มีการเปิดการแสดงผลในแบบสเตอริโอนั้นถือว่าอยู่ในเกณฑ์ที่รับได้สามารถตอบสนองได้ในเวลาจริง แต่เมื่อเปิดการใช้สเตอริโอนั้นเฟรมเรตยังถือว่าต่ำเกินไป ซึ่งจะต้องหาวิธีในการแก้ไขปัญหานี้ต่อไป

## 2. การทดสอบวัดความถูกต้องของการหาดำแหน่งผู้เล่น

การหาดำแหน่งของผู้เล่นตามวิธีการในข้อ 5.8 นั้นเป็นเพียงค่าประมาณ ซึ่งเมื่อเปรียบเทียบกับตำแหน่งของผู้เล่นของโดยการวัดจริงนั้นจะมีความแตกต่างกัน โดยจากการวัดเฉพาะตำแหน่งตามแนวลึก (แกน Z) ที่ได้จากการคำนวณด้วยโปรแกรมเปรียบเทียบกับค่าจากการวัดจริงทั้งหมด 4 ค่าแล้วเป็นดังนี้

ค่าที่คำนวณจากโปรแกรม (ชม.)	ค่าจากการวัด (ชม.)
-1.39697	-1.5
-1.80683	-2.0
-2.20156	-2.5
-2.61742	-3.0

ตารางที่ 5.2 ค่าตำแหน่งในแนวลึก

พบว่าค่าที่ได้มีความแตกต่างกัน ซึ่งสามารถนำมาคำนวณหาค่าความผิดพลาดกำลังสองเฉลี่ย (Mean Squared Error (MSE)) ได้จากสมการ 5.17

$$MSE = \frac{\sum(x_0 - x)^2}{n} \quad (5.17)$$

$x_0$  คือ ค่าจริง

$x$  คือ ค่าประมาณ

$n$  คือ จำนวนตัวอย่าง

จากข้อมูลเปรียบเทียบจะพบว่าค่าที่คำนวณจาก โปรแกรมมีความผิดพลาดกำลังสองเฉลี่ย อยู่ที่ 0.0708409 ซึ่งถือว่าอยู่ในเกณฑ์ที่ยอมรับได้

3. การทดสอบวัดความถูกต้องของการตรวจจับวัตถุที่นำมาวางในฉาก เมื่อ โปรแกรมทำงานและผู้เล่นนำวัตถุต่างๆ มาวางในฉาก ผลการตรวจสอบวัตถุของ โปรแกรมเป็นดังนี้

วัตถุ	ผลการตรวจจับ
ขนมประเภทที่ 1	ถูกต้อง
ขนมประเภทที่ 2	ถูกต้อง
ธนบัตรใบละยี่สิบ	ถูกต้อง
ธนบัตรใบละห้าสิบ	ถูกต้อง
ธนบัตรใบละหนึ่งร้อย	ถูกต้อง

ตารางที่ 5.3 ผลการตรวจจับวัตถุ

จากผลการทดลองจะพบว่า โปรแกรมสามารถตรวจจับวัตถุได้ถูกต้องทั้งหมด

## 5.11 สรุปผล

โปรแกรมที่พัฒนาขึ้นสามารถตอบสนองความต้องการในการสร้างระบบความจริงเสมือน เพื่อใช้ในการฝึกสอนเด็กออทิสซึมได้ เพราะจากผลการทดลอง โปรแกรมสามารถสร้างภาพและเสียงที่สมจริง สามารถตอบสนองได้ในเวลาจริง และสามารถมีปฏิสัมพันธ์กับผู้ใช้ได้

อย่างไรก็ตาม โปรแกรมยังมีส่วนที่ต้องพัฒนาปรับปรุงต่อไป ไม่ว่าจะเป็นการพัฒนาวิธีการในการสร้างภาพด้วยเทคนิค Light Field ให้ภาพที่ได้มีความถูกต้องมากขึ้น การพัฒนาวิธีการในการหาตำแหน่งของผู้เล่นที่ถูกต้องมากยิ่งขึ้น รวมไปถึงการพัฒนาความเร็วในการแสดงผลให้มากขึ้น

นอกจากการปรับปรุงความถูกต้องและความเร็วแล้ว การเพิ่มความสามารถของโปรแกรมก็เป็นสิ่งที่ควรจะได้รับการพัฒนาต่อไป เช่น การให้ผู้ฝึกสอน หรือผู้ปกครอง สามารถเข้าไปมีส่วนร่วมในการรับบทบาทเป็นคณขยลินคว์ในโปรแกรมได้ หรือมีระบบแกนกลอ์ดโนมิติที่สามารถรับและทอนเงินให้กับผู้เล่นได้



## บรรณานุกรม

- [1] Paul M. (2007). Do tangible interfaces enhance learning? In: Proceeding of TEI'07, Baton Rouge, LA, USA, pp 163-169.
- [2] Price, S., Rogers, Y., Scaife, M., Stanton, D., & Neale, H. (2003). Using 'tangibles' to promote novel forms of playful learning. *Interacting with computers, Special Issue: Interaction design and children.*
- [3] Jacob, R.J.K., Ishii, H., Pangaro, G. and Patten, J. (2002), A tangible interface for organizing information using a grid. In: Proceeding of CHI'02, pp 339-346
- [4] Autism. Available at [http://www.kidshealth.org/teen/school\\_jobs/school/autism.html](http://www.kidshealth.org/teen/school_jobs/school/autism.html)
- [5] Barry, M., Pitt, I (2006) Interaction Design: a Multidimensional Approach for Learners with Autism, In: Proceeding of IDC'06 ,Tampere, Finland , pp 33-36
- [6] Bernard-Opitz, V., Sriram, and Nakoda-Sapuan, S (2001) Enhancing social problem solving in children with autism and normal children through computer-assisted instruction, *Journal of Autism and Development Disorders*, 377-384
- [7] Bosseler, A., Massaro, D., W (2003) Development and evaluation of computer-animated tutor for vocabulary and language learning, In: Proceeding of 15th International Congress of Phonetic Sciences , Barcelona ,Spain
- [8] Fails, J. A., Druin, A., Guha, M.L., Chipman, G., Simm, S. and Churaman, W. (2005). Child's Play: a comparison of desktop and physical interactive environment. In: Proceeding of IDC'05, pp 48-55
- [9] Dautenhahn, K (2000) Design issues on interactive environments for children with autism, In: Proceeding 3rd Intl Conf. Disability, Virtual reality & Assoc, Alghero, Italy
- [10] Evdokimos, L., K., Andrej, L., Maria, M., N., Magda, H., and Panagiotis, D., B. (2009). Using Affective Avatars and Rich Multimedia content for education of Children with Autism. In: Proceeding of Petra'09, Corfu, Greece,
- [11] Heimann, M., Nelson, K., Tjus, T., & Gillberg, C. (1995). Increasing reading and communication skills in children with autism through an interactive multimedia computer program. *Journal of Autism and Developmental Disorders*, 25(5), 459-480.

- [12] Hoffman, H. G., Garcia-Palacios, A., Carlin, C., Furness, T.A. III and Botella-Arbona, C. (2003). Interfaces that heal: Coupling real and virtual objects to cure spider phobia. *International Journal of Human-Computer Interaction*, 15, 469-486.
- [13] Jerry, A., F., Allison, D., Gene, C., Sante, S., and Wayne, C (2005). Child's Play: A Comparison of Desktop and Physical Interactive Environment. In *Proceedings of IDC'05 Boulder, CO*, pp 48-55
- [14] Jin-Ling L., Tung-yun L., Hao-hua C., His-Chin C., Jen-hao C., Jane Yuag-jen H., and Plolly H (2007). Playful Tray: Adopting Ubicomp and Persuasive Techniques into Play-Based Occupational Therapy for Reducing Eating Behavior in Young Children. In *UbiComp LNCS4717 ,Berlin, Heidelberg* pp 38-55.
- [15] Kuhnafel U., Cakma H., and Maass H (2000). Endoscopic surgery training using virtual reality and deformable tissue simulation, *Computers and Graphics*, v24(1), pp. 671-682.
- [16] Lesley X., Alissa N., and Nima M (2008). Are Tangible more fun? Comparing Children's enjoyment and Engagement Using Physical, Graphical and Tangible user Interface. In: *Proceeding of TEI'2008, Bonn, Germany 2008*, pp 191-198.
- [17] Leonard A., Mitchell, P., and Parson, S (2002). Finding a Place to Sit: A Preliminary Investigate into the Effectiveness of Virtual Environments for Social Skills Training for People with Autistic Spectrum Disorders. In: *Proceeding of 4th Intl Conf. Disability, Virtual Reality & Assoc. Tech., Veszprém, Hungary*.
- [18] M. North, S. North and J.R. Coble, *Virtual Reality Therapy: An Effective Treatment for the Fear of Public Speaking*, *International Journal of Virtual Reality*, 1998, Vol. 3, No. 2, 2-6.
- [19] Madhure K., and Ali M. *Teaching Table (2007): A tangible mentor for pre-K math education*. In: *Proceeding of TEI'07, LA, USA 2007* pp 191-194.
- [20] Moore, M., & Calvert, S. (2000). Vocabulary acquisition for children with autism: Teacher or computer instruction. *Journal of Autism and Developmental Disorders*, 30, 359-362.
- [21] Ouriel G., Jean-Claude M., and Jacqueline N(2005). Human Computer Interface for Autism: Assessing the Influence of Task Assignment and Output Modalities. In *Proceeding of CHI'2005, Portland, USA*, pp 1419-22.
- [22] Parrott, M., Bowman, D., & Ollendick, T. (2004). A methodology for designing specific animal phobia stimuli for virtual reality exposure therapy. *Cyberpsychology & behavior*, 7(3), 300-301.

- [23] S. Roy, E. Klinger, P. Légeron, F. Lauer, I. Chemin and P Nugues (2003). Definition of a VR-based protocol to treat social phobia, *CyberPsychology and Behavior*, 6(4), 411-420
- [24] Sitdhisanguan, K., Chotikakamthorn N., Dechaboon, A., and Out, P (2007). Comparative study of WIMP and Tangible User Interfaces in training shape matching skill for autistic children, in *Proceeding of TENCON IEEE Region 10 Conference ,Taipei Taiwan*, pp 1-4.
- [25] Strickland, D., Marcus, L., M., Mesibov, G., B., and Hogan, K(1996). Brief report: Two case studies using virtual reality as a learning tool for autistic children. *Journal of Autism and developmental Disabilities*, pp 651-659.
- [26] Wang P, Becker AA, Glover AT, Jones IA, Benford SD, Vloeberghs, M. (2005). Designing a virtual reality simulator for neurosurgery. In: Richir S, Taravel B, editors. *Proceedings of the seventh international conference on virtual reality*. Laval, France, pp. 35-9.
- [27] Werry, I., and Dautenhahn, K. (2004). Toward Interactive Robots in Autism Therapy: Background, Motivation and Challenges, *Pragmatics and Cognition*, 12(1), pp. 1-35.
- [28] Paul Bourke. "3D Stereo Rendering Using OpenGL (and GLUT)," [Online]. Available: <http://local.wasp.uwa.edu.au/~pbourke/miscellaneous/stereographics/stereorender/>.
- [29] Porter, T. and Duff, T. (1984): Compositing digital images. *SIGGRAPH*, pp. 253–259.
- [30] S.C. Chan, Heung-Yeung Shum, and King-To Ng (2007). Image-Based Rendering and Synthesis, *Signal Processing Magazine, IEEE*, Vol. 24, Issue 6, pp. 22-23.
- [31] Levoy, M. and Hanrahan, P. (1996). Light Field Rendering. *SIGGRAPH*, pp. 31–42.
- [32] Linden, D. V. J., C. (2003). Multiple Light Field Rendering, *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia*, pp 197.
- [33] CMC department, Moscow State University. "GML C++ Camera Calibration Toolbox," [Online]. Available: <http://graphics.cs.msu.ru/en/science/research/calibration/cpp>.
- [34] F.S. Hill, JR. (2007). *Computer Graphics Using OpenGL*. 3<sup>rd</sup> Edition, pp. 634-635.
- [35] Philip Lamb. "How does ARToolKit work ?," [Online]. Available: <http://www.hitl.washington.edu/artoolkit/documentation/userarwork.htm>.

## ภาคผนวก : รายชื่อบทความวิจัยเผยแพร่ผลงานในโครงการ

- [1] Sitdhisanguan, K., Chotikakamthorn N., Dechaboon, A., and Out, P (2007). Comparative study of WIMP and Tangible User Interfaces in training shape matching skill for autistic children, *Proceeding of TENCON IEEE Region 10 Conference* ,Taipei Taiwan, pp 1-4.
- [2] Sitdhisanguan, K., Chotikakamthorn N., Dechaboon, A., and Out, P. (2008). Evaluation the Efficacy of Computer - Based Training Using Tangible User Interface for Low-Function Children with Autism, *Proceedings of the 2008 Second IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning*. Banff, Canada, pp 70-74.

