ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี

AUGMENTED REALITY FOR FANCY MASK FITTING

นายโอภาส แก้วทา รหัส 52363219

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2555

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 20 ก.ค. 2558
เลขทะเบียน /682697O
เลขเรียกหนังสือ ม/ร.
มหาวิทยาลัยนเรศวร จ 948 ร
2555

ใบรับรองปริญญานิพนธ์

หัวข้อโครงงาน	ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี
ผู้ดำเนินโครงงาน	นายโอภาส แก้วทา รหัส 52363219
ที่ปรึกษาโครงงาน	อาจารย์รัฐภูมิ วรานุสาสน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2555

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

.....ที่ปรึกษาโครงงาน
(อาจารย์รัฐภูมิ วรานุสาสน์)

.....กรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.พนมขวัญ ริยะมงคล)

.....กรรมการ
(อาจารย์เศรษฐา ตั้งค้าวานิช)

.....กรรมการ
(ดร.พนัส นัถฤทธิ์)

ชื่อหัวข้อโครงงาน	ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี
ผู้ดำเนินโครงงาน	นายโอภาส แก้วทา รหัส 52363219
ที่ปรึกษาโครงงาน	อาจารย์รัฐภูมิ วรานุสาสน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2555

.....

บทคัดย่อ

โครงงานนี้จัดทำขึ้นเพื่อความบันเทิง ความสนุกสนานและเชื่อมตัวผู้ใช้เข้ากับโลกแห่งจินตนาการ ในโครงงานที่มีชื่อว่า "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" ที่สามารถตอบสนองกับภาพสัญลักษณ์เมื่อส่องผ่านกล้องซึ่งผู้ใช้สามารถเปลี่ยนโมเดลหน้ากากได้ ตามที่ต้องการโดยใช้เพียงภาพสัญลักษณ์สวมที่ศีรษะ แล้วโปรแกรมจะแสดงโมเดลหน้ากากแฟนซีขึ้นมาที่บริเวณหน้าของผู้ใช้ หากผู้ใช้ต้องการเปลี่ยนโมเดลหน้ากากแฟนซีก็สามารถใช้ภาพสัญลักษณ์ที่ใช้ในการเปลี่ยนโมเดลหน้ากากขึ้นมาในโปรแกรม โปรแกรมก็จะทำการเปลี่ยนโมเดลตามแบบที่ผู้ใช้ต้องการ ทั้งหมดใช้เพียงแค่ภาพสัญลักษณ์และคอมพิวเตอร์เท่านั้นเราก็สามารถที่จะสนุกไปกับ "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี"

Project title	Augmented Reality for Masks Fitting
Name	Mr.Aophat Gaeota ID. 52363219
Project adviser	Mr.Rattapoom Waranusart
Major	Computer Engineering
Department	Electrical and Computer Engineering
Academic year	2012

Abstract

This project under the title "Augmented Reality for Mask Fitting" is intended for entertainment and connecting a user's reality with fantasy. This program can respond to visual symbols or AR tags put on the user's head through a web camera. A user can try a virtual mask superimposing on her face using an AR tag put on her head. The user can change the virtual masks using another AR tag. This program can give a user a fun experience in trying fancy masks.

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงไปได้ด้วยความกรุณาเป็นอย่างยิ่งจากอาจารย์ที่ปรึกษาอาจารย์รัฐภูมิ วรานุสาสน์ ที่คอยให้คำปรึกษาและคอยแนะนำวิธีการแก้ปัญหามากมาย และให้ความกรุณาในการตรวจทานปริญญานิพนธ์ คณะผู้จัดทำโครงงานขอกราบขอบพระคุณเป็นอย่างสูงและขอระลึกถึงความกรุณาของท่านไว้ตลอดไป

ขอขอบพระคุณคณะกรรมการโครงงานทั้ง 3 ท่านผู้ช่วยศาสตราจารย์ ดร. พนมขวัญ ริยะ-มงคล อาจารย์เศรษฐา ตั้งค้าวานิช และดร.พนัส นัณฤทธิ์ ที่ช่วยแนะนำในสิ่งที่ควรปรับปรุงโครงงาน

ขอขอบคุณคณาจารย์ทุกท่านที่ประสิทธ์ประสาทวิชาความรู้ให้กับทางผู้จัดทำโครงงานจนทำให้โครงงานสำเร็จลุล่วง

ขอขอบคุณภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์ที่ช่วยอนุเคราะห์สถานที่ในการจัดทำโครงงาน

และสุดท้ายนี้ขอกราบขอบพระคุณคุณพ่อคุณแม่ที่คอยช่วยเหลือและคอยให้การสนับสนุนในทุกด้านจนทำให้โครงงานนี้สำเร็จลุล่วงไปได้ด้วยดี

นายโอภาส แก้วทา

สารบัญ

หน้า

ใบรับรองปริญญานิพนธ์.....ก
บทคัดย่อภาษาไทย.....ข
บทคัดย่อภาษาอังกฤษ.....ค
กิตติกรรมประกาศ.....ง
สารบัญ.....จ
สารบัญตาราง.....ซ
สารบัญรูป.....ฌ
บทที่ 1 บทนำ.....1
1.1 ที่มาและความสำคัญของโครงงาน.....1
1.2 วัตถุประสงค์ของโครงงาน.....2
1.3 ขอบเขตของโครงงาน.....3
1.4 ขั้นตอนการดำเนินงาน.....3
1.5 แผนผังการดำเนินโครงงาน.....3
1.6 ผลที่คาดว่าจะได้รับ.....5
1.7 รายละเอียดงบประมาณของโครงงาน.....5
บทที่ 2 ทฤษฎีและหลักการที่เกี่ยวข้อง.....6
2.1 ความรู้เบื้องต้นเกี่ยวกับเทคโนโลยีความจริงเสริม.....6
2.1.1 หลักการที่ใช้ในเทคโนโลยีความจริงเสริม.....7
2.1.2 ความก้าวหน้าของเทคโนโลยีความจริงเสริม.....7
2.2 ความรู้เบื้องต้นเกี่ยวกับการทำงานและไลบรารีของ OpenGL.....11
2.2.1 คุณลักษณะเด่นของ OpenGL.....12
2.2.2 การทำงานของ OpenGL.....13
2.2.3 ระบบสี RGB.....13
2.3 ความรู้เบื้องต้นเกี่ยวกับไลบรารี ARToolKit.....14

2.4 การหาค่าขีดแบ่ง.....15
2.5 การติดป้ายให้ส่วนประกอบที่ติดกัน.....16
2.6 การตรวจจับจุดตัดผ่านศูนย์.....16
2.7 การประมาณตำแหน่ง.....18
2.8 แมทริกซ์การแปลง.....21
2.9 การเรนเดอร์ภาพสามมิติ.....23
2.10 ความรู้เกี่ยวกับไลบรารี่ openFrameWork.....24

บทที่ 3 วิธีการดำเนินโครงการ.....25
3.1 ศึกษารายละเอียดและหลักการความรู้ ที่เกี่ยวข้องกับโครงงาน.....25
3.2 ออกแบบการทำงานทั้งหมด.....26
3.3 การทำงานของไลบรารี openFrameWork.....27
3.4 การออกแบบหน้ากาก 3 มิติ.....28
3.5 ออกแบบการใช้งานและค้นหาสัญลักษณ์.....32
3.5.1 การประมาณตำแหน่ง.....33
3.5.2 3D rendering.....33
3.6 การออกแบบการใช้งานของโปรแกรม.....34
3.6.1 การแสดงโมเดลหน้ากาก.....34
3.6.2 การออกแบบสถานะของโมเดลหน้ากากขณะแสดงผล.....35
3.6.3 การเปลี่ยนโมเดลหน้ากาก.....35

บทที่ 4 ผลการทดลอง.....37
4.1 ผลการทดลองการตรวจหาวัตถุสัญลักษณ์ (Marker).....37
4.2 ผลการทดลองการวาดโมเดลหน้ากากลงบนภาพสัญลักษณ์.....39
4.3 ผลการทดลองการใช้ภาพสัญลักษณ์ในการเปลี่ยนโมเดลหน้ากาก.....40
4.4 ปัจจัยที่มีผลต่อการทดลอง.....42
4.4.1 มุมของภาพสัญลักษณ์.....42

4.4.2 ระยะห่างของกล้องกับภาพสัญลักษณ์.....43
4.4.3 แสง.....46
4.5 สรุปผลการทดลอง.....48
บทที่ 5 สรุปผลและข้อเสนอแนะ.....49
5.1 สรุปผลการทดลอง.....49
5.2 ปัญหาที่พบระหว่างการทำโครงงาน.....49
5.3 ข้อเสนอแนะและวิธีการแก้ไขปัญหา.....50
5.4 ข้อเสนอแนะสำหรับงานในอนาคต.....50
เอกสารอ้างอิง.....51
ภาคผนวก.....54
ประวัติผู้เขียนโครงงาน.....62

สารบัญตาราง

ตารางที่ หน้า

1.1 ตารางแสดงระยะเวลาการดำเนินงาน.....3

สารบัญรูป

รูปที่ หน้า

2.1 แสดงการทำงานของเทคโนโลยีความจริงเสริม.....6

2.2 แสดงหลักการของเทคโนโลยีความจริงเสริม.....7

2.3 เกมเสมือนสามมิติ.....8

2.4 รูปแสดงตัวอย่างโปรแกรมเทคโนโลยีความจริงเสริมกับการศึกษา.....8

2.5 รูปที่นำเทคโนโลยีความจริงเสริมไปใช้การเชิงการแพทย์.....9

2.6 ภาพแสดงตัวอย่างตึกอาคารในเมือง.....10

2.7 การทำงานของ OpenGL.....13

2.8 รูปของระบบสี RGB.....14

2.9 แสดงภาพการแยกภาพค่าขีดแบ่ง.....15

2.10 แสดงการทำงานของกระบวนการการติดป้ายให้ส่วนประกอบที่ติดกัน.....16

2.11 แสดงภาพตั้งต้น.....17

2.12 แสดงอนุพันธ์อันดับสองของภาพตั้งต้น.....17

2.13 ความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame.....18

2.14 ความสัมพันธ์ของ Ideal Screen Coordinatesและ Observe Screen Coordinates.....19

2.15 กระบวนการคำนวณค่า 3D Poses.....20

2.16 แสดงความสัมพันธ์แมทริกซ์การแปลง.....21

2.17 แสดงความสัมพันธ์ของมุมระหว่างกล้องกับสัญลักษณ์.....22

2.18 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ.....23

2.19 ขั้นตอนการทำงานของ openFrameWork.....24

3.1 แสดงเครื่องมือที่ใช้.....25

3.2 แสดงขั้นตอนการทำงานของ "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี".....26

3.3 การสร้างโมเดลหน้าจากโมเดลต้นแบบโดยการใส่ภาพพื้นหลังลงบนโมเดลหน้ากาก.....30

3.4 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 1.....28

3.5 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 2.....29

3.6 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 3.....29
3.7 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 4.....30
3.8 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 5.....30
3.9 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 6.....31
3.10 แสดงหลักการอ่านและค้นหาสัญลักษณ์ (Marker).....33
3.11 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ.....33
3.12 การวาดโมเดลลงบนสัญลักษณ์.....34
3.13 ตัวอย่างภาพสถานะของโมเดลหน้ากาก.....35
3.14 เมื่อตรวจพบภาพสัญลักษณ์หมายเลข A จะทำการเพิ่ม Model Index.....36
3.15 เมื่อตรวจพบภาพสัญลักษณ์ หมายเลข B จะทำการลด Model Index 36
4.1 หมายเลขภาพสัญลักษณ์เท่ากับหมายเลข X.....38
4.2 หมายเลขภาพสัญลักษณ์เท่ากับหมายเลข A.....39
4.3 หมายเลขภาพสัญลักษณ์เท่ากับหมายเลข B.....39
4.4 การวาดโมเดลหน้ากากลงบนภาพสัญลักษณ์หมายเลขที่ X.....40
4.5 โปรแกรมตรวจพบภาพสัญลักษณ์หมายเลข A.....41
4.6 โปรแกรมตรวจพบภาพสัญลักษณ์หมายเลข B.....42
4.7 การแสดงผลของโมเดลหน้ากากเมื่อมุมของภาพสัญลักษณ์เปลี่ยนไป.....43
4.8 ภาพสัญลักษณ์มีระยะห่างจากกล้องประมาณ 30-50 เซนติเมตร.....44
4.9 ภาพสัญลักษณ์มีระยะจากกล้องระยะประมาณ 80 - 100 เซนติเมตร.....45
4.10 ภาพสัญลักษณ์มีระยะห่างจากกล้อง 120 เซนติเมตร.....45
4.11 ระยะห่างระหว่างกล้องกับภาพสัญลักษณ์มีระยะ 180 เซนติเมตรขึ้นไป.....46
4.12 ทดลองในที่ ที่แสงเหมาะสมทำให้การแสดงผลออกตามปกติ.....47
4.13 ทดลองในที่ ที่มีแสงน้อยทำให้ตรวจหาภาพสัญลักษณ์ไม่ได้.....47
4.14 ภาพแสดงการทำงานทั้งหมด.....48

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันเทคโนโลยีความจริงเสริม (Augmented Reality) เป็นเทคโนโลยีที่เข้ามามีบทบาทในด้านต่างๆ ของสังคมไม่ว่าจะ เป็นด้านการศึกษาด้านการแพทย์ ด้านธุรกิจและด้านการทหาร เทคโนโลยีความจริงเสริมเป็นสิ่งที่จะรวมเอาจินตนาการรวมเข้ากับโลกแห่งความเป็นจริง โดยการทำงานของเทคโนโลยีความจริงเสริมนั้น จะเริ่มจากการที่เรานำกล้องมาจับภาพที่ภาพสัญลักษณ์ (Marker) แล้วหาข้อมูลภาพสัญลักษณ์จากฐานข้อมูล (Marker Database) ที่ได้เก็บภาพ และจำข้อมูลของภาพสัญลักษณ์ไว้ จากนั้นจะคำนวณค่าตำแหน่งเชิง 3 มิติ (Pose Estimation) ของสัญลักษณ์ เพื่อนำมาวิเคราะห์รูปแบบของสัญลักษณ์เทียบกับกล้องแล้วเข้าสู่กระบวนการสร้าง ภาพ 2 มิติ จากโมเดล 3 มิติ (3D Rendering) เป็นการเพิ่มข้อมูลเข้าไปในภาพโดยใช้ค่าตำแหน่งเชิง 3 มิติ ที่โปรแกรมคำนวณได้จนได้ภาพเสมือนจริง เทคโนโลยีความจริงเสริมได้มีแรงบันดาลใจให้ผู้วิจัยสนใจที่จะพัฒนา "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" เพื่อทางเลือกในการเพิ่มความสนุกสนาน เพิ่มลูกเล่นให้ในชีวิตประจำวันของเรามีสีสันเพิ่มมากขึ้น

อีกทั้งผู้วิจัยยังคิดว่ายังสามารถพัฒนาต่อยอดไปทำกับสิ่งของอื่นๆ ได้ ไม่ว่าจะเป็นการลองเสื้อผ้า แว่นตา นาฬิกา หมวก ฯลฯ และ อาจจะพัฒนาและนำไปใช้ในเครือข่ายอินเตอร์เน็ททำให้เกิดการซื้อขายกันได้น่าสนใจและสะดวกสบายยิ่งขึ้น

1.2 วัตถุประสงค์ของโครงงาน

1.2.1 เพื่อสร้าง " ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี " (Augmented - Reality for Fancy Mask Fitting) ที่สามารถทดลองแบบหน้ากากที่มีอยู่ในโปรแกรมได้

1.2.2 เพื่อให้สามารถนำโปรแกรมไปประยุกต์ใช้ทางการตลาดในการส่งเสริมการขายได้

1.2.3 เพื่อศึกษาเทคโนโลยี และเทคนิควิธีการพัฒนาระบบความจริงเสริม (Augmented - Reality)

1.2.4 เพื่อศึกษาเทคนิควิธีการที่เกี่ยวข้อง เช่นคอมพิวเตอร์กราฟฟิกและทัศนศาสตรคอม-พิวเตอร์

1.3 ขอบเขตของโครงงาน

1.3.1 สร้าง " ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี " ที่สามารถลองหน้า-กากได้เมื่อส่องผ่านกล้อง

1.3.2 สร้าง " ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี " ที่สามารถลองหน้า-กากแฟนซีตามแบบที่มีในโปรแกรมได้

1.3.3 สามารถตอบโต้กับผู้ใช้ได้ โดยภาพที่แสดงจะเปลี่ยนไปตามการเลือกหมวกตามที่ต้องการ

1.4 ขั้นตอนการดำเนินงาน

1.4.1 ศึกษาความรู้ทั่วไปเกี่ยวกับระบบความจริงเสริม

1.4.2 ศึกษาและค้นคว้าข้อมูลเกี่ยวกับไลบรารี่ Open Framework

1.4.3 ศึกษาและค้นคว้าข้อมูลเกี่ยวกับไลบรารี่ ARToolkit

1.4.4 ศึกษาการทางานของไลบรารี OpenGL เพื่อสร้างภาพกราฟิก

1.4.5 ออกแบบโปรแกรมที่สามารถลองแบบหมวกได้

1.4.6 เขียนโปรแกรมส่วนโมเดลหน้ากากแฟนซีลงในโปรแกรม

1.4.7 เขียนโปรแกรมส่วนการรับข้อมูลจากวัตถุสัญลักษณ์ผ่านกล้อง

1.4.8 เขียนโปรแกรมส่วนการอ่านวัตถุสัญลักษณ์แล้วแสดงแบบหน้ากากที่ต้องการ

1.4.9 เขียนโปรแกรมสร้างกราฟิกแสดงผลที่ตำแหน่งวัตถุสัญลักษณ์

1.4.10 ทดสอบความสมบูรณ์และใช้งาน

1.4.11 สรุปผลและจัดทำรูปเล่มโครงงาน

1.5 แผนผังการดำเนินโครงงาน

1.1 ตารางแสดงระยะเวลาการดำเนินงาน

กิจกรรม	ปี 2555							ปี 2556				
	มิ.ย.	ก.ค.	ส.ค.	ก.ย.	ต.ค.	พ.ย.	ธ.ค.	ม.ค.	ก.พ.	มี.ค.	เม.ย	พ.ค.
1. ศึกษาความรู้ทั่วไปเกี่ยวกับระบบความจริงเสริม												
2. ศึกษาและค้นคว้าข้อมูลเกี่ยวกับไลบรารี่ Open Framework												
3. ศึกษาการทำงานของไลบรารี OpenGL เพื่อสร้างภาพกราฟิก												
4. ศึกษาและค้นคว้าข้อมูลเกี่ยวกับไลบรารี่ ARToolkit												
5. ออกแบบโปรแกรมที่สามารถลองแบบหน้ากากได้												

6. เขียนโปรแกรมส่วนโมเดลหน้ากากลงในโปรแกรม										
7. เขียนโปรแกรม ส่วนการรับข้อมูลจากวัตถุสัญลักษณ์ผ่านกล้อง										
8. เขียนโปรแกรมส่วนการอ่านวัตถุสัญลักษณ์แล้วแสดงแบบหน้ากากที่ต้องการ										
9. ทดสอบความสมบูรณ์และใช้งาน										
10. สรุปผลและจัดทำรูปเล่ม **โครงงาน**										

1.6 ผลที่คาดว่าจะได้รับ

1.6.1 "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" ที่สามารถทดลองแบบหน้ากากที่มีได้

1.6.2 สามารถนำเทคโนโลยีคอมพิวเตอร์กราฟิกและทัศนาศาสตร์คอมพิวเตอร์มาใช้ในการพัฒนาระบบความจริงเสริมได้

1.6.3 สามารถพัฒนาโปรแกรมไปประยุกต์ใช้กับวัตถุอื่นๆได้ เช่น แหวน แว่นตา นาฬิกา หมวก เป็นต้น

1.6.4 สามารถนำโปรแกรมไปประยุกต์ใช้ทางการตลาดในการส่งเสริมการขายสินค้าได้

1.7 รายละเอียดงบประมาณของโครงงาน

ค่าอุปกรณ์	เป็นเงิน	400 บาท
ค่าเอกสารทำรูปเล่ม	เป็นเงิน	600 บาท
รวม	เป็นเงิน	1,000 บาท

หมายเหตุ ถัวเฉลี่ยทุกรายการ

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

2.1 ความรู้เบื้องต้นเกี่ยวกับเทคโนโลยีความจริงเสริม

เทคโนโลยีความจริงเสริม (Augmented Reality) [1] เป็นเทคโนโลยีที่ผสานโลกเสมือนเข้ากับโลกแห่งความเป็นจริงให้เป็นหนึ่งเดียวกัน เทคโนโลยีมีหลักการทำงานโดยเริ่มจากการที่เรามีภาพสัญลักษณ์ แล้วเราจะนำมาส่องผ่านกล้อง จากนั้นกล้องจะนำภาพสัญลักษณ์ที่พบไปตรวจในส่วนประมวลผล แล้วดูว่าภาพสัญลักษณ์ตรงกับภาพ 2 มิติ หรือ 3 มิติ แบบใด จากนั้นโปรแกรมจะทำการวาดภาพกราฟฟิกดังกล่าวลงบนภาพสัญลักษณ์ เทคโนโลยีความจริงเสริมช่วยทำให้เพลิดเพลินสนุกสนาน อีกทั้งยังได้ศึกษาเกี่ยวกับ openGL, ARToolKit ,openFrameWork เทคโนโลยีความจริงเสริมนำไปประยุกต์ในงานต่างๆได้มากมาย ไม่ว่าจะเป็นงานโฆษณา กีฬา การบันเทิง หรือการศึกษา เป็นต้น

รูปที่ 2.1 แสดงการทำงานของเทคโนโลยีความจริงเสริม

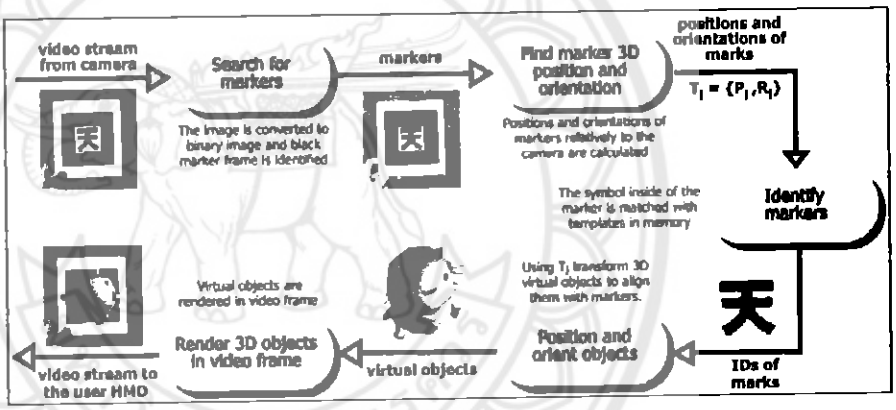
2.1.1 หลักการที่ใช้ในเทคโนโลยีความจริงเสริม [1]

1. สัญลักษณ์ (Marker หรือ AR code) ใช้ในการกำหนดตำแหน่งของวัตถุ

2. อุปกรณ์รับภาพ (Image input device) หรือกล้องเว็บแคม ใช้มองตำแหน่งสัญลักษณ์แล้วส่งเข้าส่วนประมวลผลความจริงเสริม (AR Engine)

3. ส่วนประมวลผลความจริงเสริม (AR Engine) ทำหน้าที่อ่านข้อมูลจากกล้องเข้ามาประมวลผลเพื่อคำนวณตำแหน่งของสัญลักษณ์

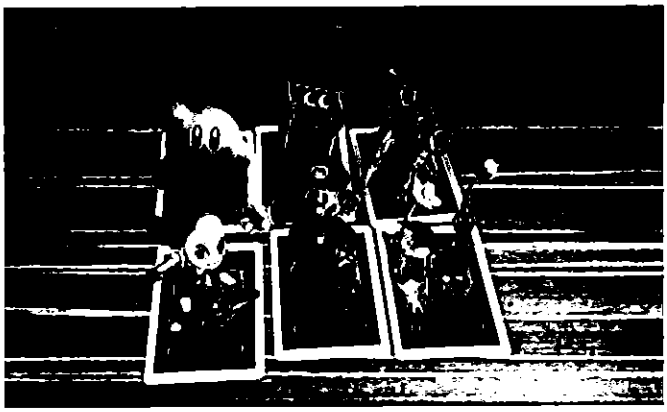
4. อุปกรณ์แสดงผล (Display device) จอแสดงผลข้อมูลที่ผ่านการประมวลผลแล้วส่งมาให้ในรูปแบบของวีดีโอ



รูปที่ 2.2 แสดงหลักการของเทคโนโลยีความจริงเสริม [11]

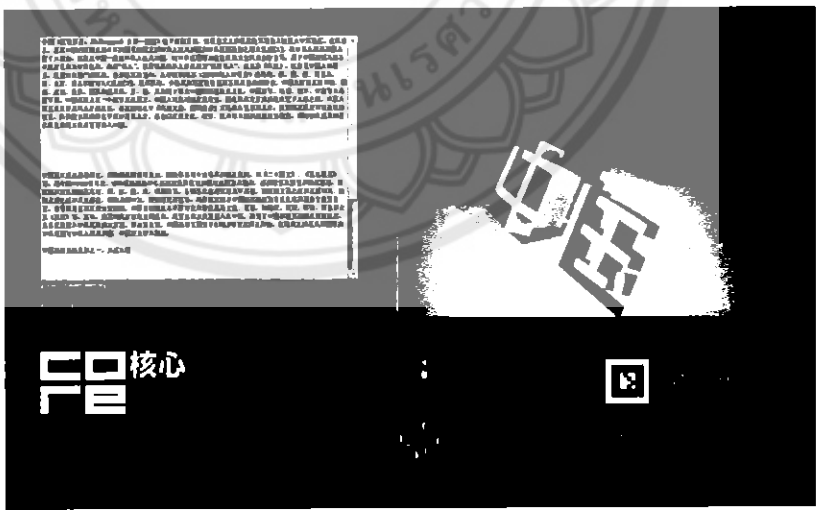
2.1.2 ความก้าวหน้าของเทคโนโลยีความจริงเสริม [1]

1. ด้านความบันเทิง ก็จะเป็นความก้าวหน้าในด้านเกม การ์ตูนหรือแม้กระทั่งภาพยนตร์ ซึ่งจะเห็นได้ว่าจะนำความก้าวหน้าทางด้านเทคโนโลยีความจริงเสริมมาใช้ร่วมด้วยดังรูปที่ 2.3



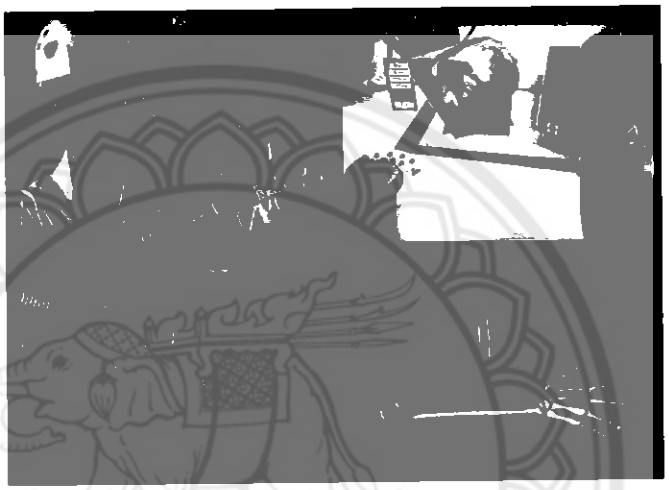
รูปที่ 2.3 เกมเสมือนสามมิติ [12]

2. ด้านการศึกษา เทคโนโลยีความจริงเสริมนั้นสามารถนำประยุกต์ในการศึกษาได้มากมายในปัจจุบันซึ่งเป็นการใช้เทคโนโลยีความจริงเสริมมาแสดงความอักษรเป็นแบบสามมิติซึ่งจะใช้ในวิชากราฟิกต่างๆหรือจะเป็นการนำเทคโนโลยีความจริงเสริมมาใช้ในการศึกษาระบบสุริยะ ซึ่งจะแสดงรายละเอียดของดาวเคราะห์ต่างๆเป็นต้น ดังรูปที่ 2.4



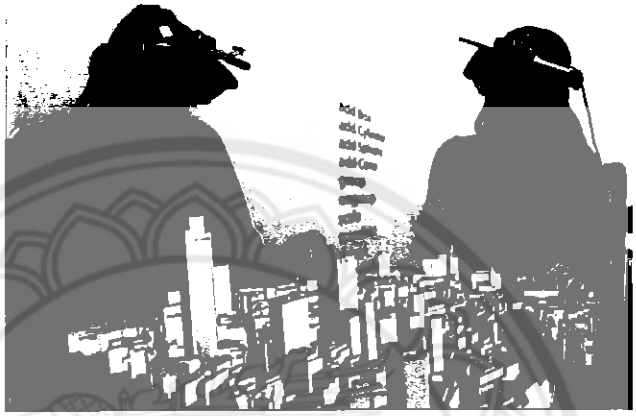
รูปที่ 2.4 รูปแสดงตัวอย่างโปรแกรมเทคโนโลยีความจริงเสริมกับการศึกษา [13]

3. ทางการแพทย์ เทคโนโลยีความจริงเสริมกับการแพทย์นั้นอาจจะฟังดูเหมือนห่างไกลแต่ความจริงนั้นในปัจจุบันได้มีการนำเทคโนโลยีความจริงเสริมกับเทคโนโลยีทางด้านการแพทย์มาใช้ร่วมการแล้ว ดังรูปที่ 2.5



รูปที่ 2.5 รูปที่นำเทคโนโลยีความจริงเสริมไปใช้การเชิงการแพทย์ **[14]**

4. ทางด้านธุรกิจ ส่วนในด้านธุรกิจนั้นก็ได้มีการนำเทคโนโลยีความจริงเสริมมาใช้และได้รับรับความนิยมเป็นอย่างมาก โดยเฉพาะด้านโฆษณา เพราะเราจะเห็นได้จากการใช้ภาพกราฟิกในงานโฆษณา ที่เห็นได้ตามสื่อต่างๆในปัจจุบัน ดังรูปที่ 2.6



รูปที่ 2.6 ภาพแสดงตัวอย่างตึกอาคารในเมือง [15]

2.2 ความรู้เบื้องต้นเกี่ยวกับการทำงานและไลบรารีของ OpenGL [2]

OpenGL คือ ซอฟท์แวร์ที่เชื่อมต่อกับฮาร์ดแวร์ที่แสดงผลทางด้านกราฟิก ซึ่งประกอบ ด้วยคำสั่งประมาณ 250 คำสั่ง OpenGL มีโครงสร้างเป็นอิสระจากฮาร์ดแวร์ (hardware-independent interface) และสามารถใช้ได้กับระบบปฏิบัติการหลายๆระบบไลบรารี่ OpenGL ซึ่งจัดทำโดย silicon graphic เป็นเทคโนโลยีเปิด ซึ่งใช้ในเครื่องยูนิกซ์มาก่อน โดยการเรียกใช้ผ่านภาษาซี และเนื่องจากเป็นที่นิยมใช้กันมากในซอฟท์แวร์ที่ใช้สร้างภาพสามมิติ โดยเฉพาะพวกเกมส์ต่างๆ จนกระทั่งปัจจุบันเป็นไลบรารี่ ที่ติดมากับ Ms windows ตั้งแต่ 95 ขึ้นไป ซึ่งดูได้จากที่ directory \windows\system\ จะเห็นว่ามี file ชื่อ opengl32.dll อยู่ นั่น คือ library ที่ใช้สร้างภาพสามมิติของ OpenGL การวาดรูปวัตถุต่างๆของ OpenGLจะใช้หน่วยพื้นฐานคือ จุด เส้น และรูปหลายเหลี่ยม ซึ่งเราสามารถเรียกใช้ฟังก์ชันสำหรับสร้างหน่วยพื้นฐานเหล่านี้ได้ โดยแต่ละฟังก์ชันจะทำงานเป็นอิสระต่อกันหน่วยพื้นฐานต่างๆจะถูกกำหนดอยู่ในรูปของค่าตำแหน่งจุดซึ่งอาจจะมีเพียงค่าเดียวหรือประกอบกันหลายๆค่าก็ได้ เช่น อาจจะกำหนดตำแหน่งของจุด หรือจุดสิ้นสุดของเส้น หรืออาจจะเป็นมุมของรูปหลายเหลี่ยม (polygon) เป็นต้น นอกจากตำแหน่งต่างๆแล้ว ข้อมูลของหน่วยพื้นฐานก็ยังประกอบไปด้วย สี นอร์มอล (Normal) เท็กเจอร์ (Texture) และเงื่อนไขกำหนดขอบ (Edge Flags) โดยข้อมูลของหน่วยพื้นฐานจะถูกประมวลผลเป็นอิสระต่อกัน แต่จะประมวลผลด้วยวิธีเดียวกันอย่างเป็นลำดับ การนำข้อมูลเหล่านี้ไปวาดเป็นวัตถุและบางครั้งอาจจะมีการตัดแต่งวัตถุ (Clip) เพื่อให้เหมาะกับพื้นที่ที่แสดงผล ซึ่งในกรณีนี้ข้อมูลของจุดอาจจะถูกดัดแปลงเพื่อสร้างจุดขึ้นมาใหม่ สำหรับคำสั่งต่างๆจะถูกประมวลผลเป็นลำดับเสมอ แม้ว่าจะมีการหน่วงเวลาก่อนคำสั่งแสดงผล ซึ่งหมายความว่าหน่วยพื้นฐานต่างๆจะถูกวาดให้เสร็จก่อนที่จะมีการแสดงผล OpenGL ให้เราสามารถควบคุมการทำงานต่างๆได้โดยตรงไม่ว่าจะเป็นการทำงานที่เป็นกราฟิกสองมิติหรือสามมิติ ซึ่งสิ่งที่เราสามารถควบคุม หรือจัดการได้จะอยู่ในรูปของตัวแปรต่างๆ เช่น การแปลงเมตริกซ์สัมประสิทธ์ของสมการส่องสว่างและตัวดำเนินการปรับปรุงจุดภาพ แต่อย่างไรก็ตาม OpenGL จะไม่ให้เราควบคุมวัตถุทางเรขาคณิตที่มีความซับซ้อน นั่นคือ OpenGL จะประกอบไปด้วยคำสั่งพื้นฐานมากกว่าที่จะเป็นฟังก์ชันที่ใช้สร้างวัตถุที่มีรูปร่างเฉพาะบางอย่าง รูปแบบสำหรับการแปลความหมายคำสั่ง OpenGL คือ ไคลเอนต์เซิร์ฟเวอร์ (Client/Server) โดยโค้ดคำสั่งที่ใช้งานจะอยู่ในส่วนของไคลเอนต์ ซึ่งจะถูกแปลและประมวลผลด้วย OpenGL ที่ทำหน้าที่เป็นเซิร์ฟเวอร์ เซิร์ฟเวอร์อาจจะอยู่ในเครื่องเดียวกันกับไคลเอนต์ หรืออาจจะอยู่กันคนละเครื่องก็ได้ ดังนั้น

OpenGL จึงเป็นโปรแกรมที่ทำงานผ่านระบบเครือข่าย โดยเซิร์ฟเวอร์สามารถจัดการกับคำสั่งต่างๆ ในรูปของแคปซูลคำสั่งของ OpenGL ที่มีการเรียกใช้จากไคลเอนต์ โดยข้อกำหนดของโปรโตคอลที่มีอยู่แล้วหรืออาจจะใช้โปรโตคอลอิสระขั้นตอนการประมวลผลข้อมูลของ OpenGL คำสั่งจะถูกส่งผ่านจากทางด้านซ้ายโดยที่แต่ละบล็อกจะประมวลผล และส่งข้อมูลไปยังบล็อกถัดไปตามลำดับ ซึ่งการประมวลผลในลักษณะเช่นนี้จะเรียกว่าการประมวลผลแบบไปป์ไลน์ (Pipeline) คำสั่งส่วนหนึ่งที่ส่งมาเป็นคำสั่งที่ใช้ในการวาดรูปทรงเรขาคณิต ส่วนคำสั่งอีกส่วนหนึ่งจะใช้ควบคุมวัตถุที่วาดขึ้นมา

2.2.1 คุณลักษณะเด่นของ OpenGL [2]

OpenGL เป็นสภาพแวดล้อมที่ดีเยี่ยมในการพัฒนาโปรแกรมประยุกต์กราฟิก 2 มิติ และ 3 มิติ ตั้งแต่ที่ได้มีการแนะนำ OpenGL ในปี พ.ศ.2534 OpenGL ก็กลายมาเป็นอุตสาหกรรมและได้รับการยอมรับทั่วไปในการสร้างกราฟิกสามมิติ โดยเฉพาะในงานด้านวิศวกรรมงาน ด้านการแพทย์ และงานที่ต้องการคุณภาพกราฟิกระดับสูง

สำหรับลักษณะเด่นของ OpenGL สามารถสรุปได้ดังนี้

1. มาตรฐานอุตสาหกรรมด้านกราฟิกรายละเอียด OpenGL ถูกจัดการหรือบริหารโดยสมาคมอิสระซึ่งเป็น ARB (Architecture Review Board) โดยมีคณะกรรมการส่วนหนึ่งมาจาก SGI หรือ Siligon Graphics และจาก ไมโครซอฟท์ ARP มีหน้าที่แนะนำสเปค OpenGL ต่างๆด้วยการสนับสนุนของคณะกรรมการ OpenGL จึงเป็นระบบเปิดอย่างแท้จริง (มาตรฐานกราฟิกที่ทำงานได้หลายๆ แพลตฟอร์ม)

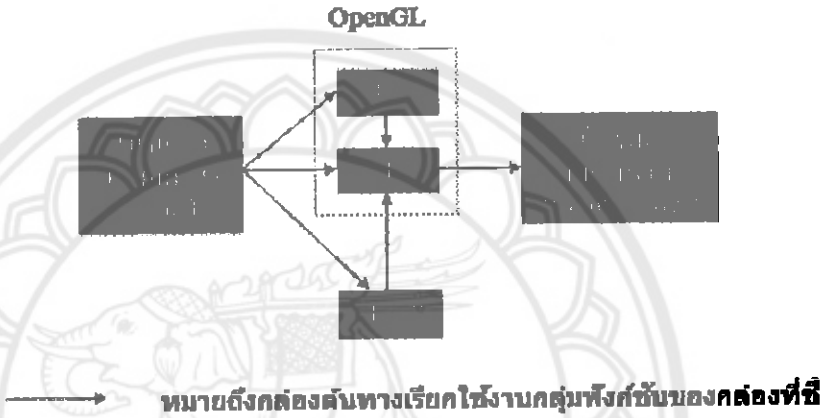
2. มีความเสถียรภาพเนื่องจาก OpenGL มีการใช้มานานประมาณ 9 ปี รายละเอียดของ OpenGL ถูกควบคุมโดยสมาคมอิสระ และมีการอัพเดทข้อมูลพร้อมทั้งการประกาศสู่สาธารณชนสำหรับนักพัฒนาซอฟท์แวร์เพื่อรับทราบสำหรับการเปลี่ยนแปลง รายละเอียด หรือสเปค

3. มีความเชื่อถือได้และย้ายไปไหนก็ได้ ทุกแอพพริเคชั่นที่ใช้ OpenGL จะสร้างภาพที่ยังคงเหมือนเดิมซึ่งเป็นผลมาจากฮาร์ดแวร์กราฟิกที่เข้ากันได้กับ OpenGL โดยไม่คำนึงถึงระบบปฏิบัติการ แอพพริเคชั่นที่ใช้ OpenGL สามารถทำงานได้ตั้งแต่จากคอมพิวเตอร์เครื่องเล็ก PC

เวิร์คสเตชั่น จนถึงซุปเปอร์คอมพิวเตอร์ ผลลัพธ์ก็คือ แอพพริเคชันสามารถขยายไปยังคอมพิวเตอร์ระดับใดๆก็ได้

4.ความง่ายต่อการใช้งาน OpenGLมีคำสั่งต่างๆมากมายที่ได้ถูกจัดโครงสร้างอย่างดี และถูกหลักตรรกศาสตร์ทำให้มีความง่ายดายต่อการเข้าใจและการนำมาใช้งาน

2.2.2 การทำงานของ OpenGL

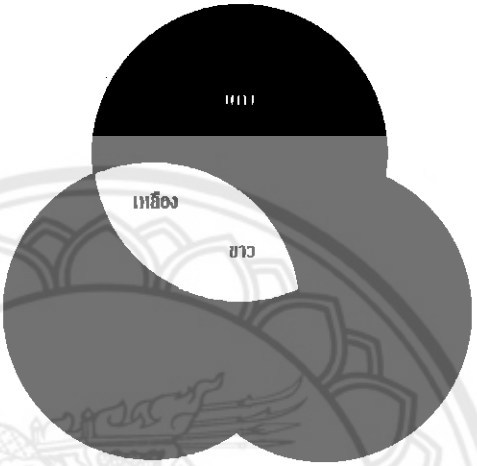


รูปที่ 2.7 การทำงานของ OpenGL [2]

2.2.3 ระบบสี RGB [3]

ระบบสี RGB เป็นระบบสีของแสง ซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึม จะเกิดแถบสีที่เรียกว่า สีรุ้ง (Spectrum) ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ แดง แสด เหลือง เขียว น้ำเงิน คราม ม่วง ซึ่งเป็นพลังงานอยู่ในรูปของรังสี ที่มีช่วงคลื่นที่สายตา สามารถมองเห็นได้ แสงสีม่วงมีความถี่คลื่นสูงที่สุด คลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วง เรียกว่า อัลตราไวโอเลต (Ultra Violet) และคลื่นแสงสีแดง มีความถี่คลื่นต่ำที่สุด คลื่นแสง ที่ต่ำกว่าแสงสีแดงเรียกว่า อินฟราเรด (InfraRed) คลื่นแสงที่มีความถี่สูงกว่าสีม่วง และต่ำ กว่าสีแดงนั้น สายตาของมนุษย์ไม่สามารถรับได้ และเมื่อศึกษาดูแล้วแสงสีทั้งหมดเกิดจาก แสงสี 3 สี คือสีแดง (Red) สีน้ำเงิน (Blue) และสีเขียว (Green) ทั้งสามสีถือเป็นแม่สีของแสง เมื่อนำมาฉายรวมกันจะทำให้เกิดสีใหม่อีก 3 สี คือ สีแดงมาเจนต้า สีฟ้าไซแอน และสีเหลือง และถ้าฉายแสงสีทั้งหมดรวมกันจะได้แสง

สีขาว จากคุณสมบัติของแสงนี้เรา ได้นำมาใช้ประโยชน์ทั่วไป ในการฉายภาพยนตร์ การบันทึกภาพวิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์ และการจัดแสงสีในการแสดง เป็นต้น [3]



รูปที่ 2.8 รูปของระบบสี RGB [3]

2.3 ความรู้เบื้องต้นเกี่ยวกับไลบรารี ARToolKit [4]

ARToolKit เป็นไลบรารี (library) ที่ช่วยในการสร้างเทคโนโลยีความจริงเสริมและเครื่องมือช่วยวาดวัตถุสามมิติ ประกอบฉากภาพที่ได้เกิดจากการรวมภาพที่ถ่ายจากกล้องเข้ากับโมเดลสามมิติ ที่ต้องการ โดยมีการจัดทิศทางของวัตถุให้สอดคล้องกับมุมของกล้องทำให้ดูเสมือนราวกับว่ามีวัตถุอยู่ในบริเวณนั้นจริงๆ อุปกรณ์ที่จำเป็นคือ 1. กล้อง 2. โปรแกรม 3.เครื่องพิมพ์เอาไว้พิมพ์ลวดลายลงบนกระดาษให้กล้องตรวจจับหลักการทำงานของโปรแกรมคือเมื่อภาพลวดลายเกิดการขยับโปรแกรมจะคำนวณหามุมที่เปลี่ยนไปแล้วนำมาใช้อ้างอิงซ้อนวัตถุสามมิติลงในภาพอีกที AR-ToolKit เป็นชุดพัฒนาซอฟแวร์ (SDK) ที่พัฒนาขึ้นโดยภาษา C และ C++ เพื่อใช้เป็นแก่นในการพัฒนาโปรแกรมประยุกต์ทางด้านเทคโนโลยีความจริงเสริม โดยอ้างอิงเนื้อหาจากหัวข้อ 2.1.1 (หลักการของเทคโนโลยีความจริงเสริม) เครื่องมือ ARToolKit จะมีฟังก์ชั่นสำหรับส่วนของการวิเคราะห์ภาพ (Image Analysis) และ ส่วนคำนวณตำแหน่งเชิง 3 มิติ (Pose Estimation) ที่ค่อนข้างสมบูรณ์ อย่างไรก็ตาม ARToolKit ยังมีฟังก์ชั่นที่ใช้สำหรับการสร้างภาพจากโมเดล 3 มิติ (3D Rendering) ในระดับพื้นฐานซึ่งอิงอยู่กับไลบรารี OpenGL ส่วนที่ยากของการพัฒนาโปรแกรมประยุกต์เทคโนโลยีความจริงเสริมคือความแม่นยำในการคำนวณตำแหน่งของวัตถุในสิ่งแวดล้อมจริง (Image Analysis และ Pose Estimation) เพื่อเพิ่มภาพที่สร้างจาก

คอมพิวเตอร์กราฟิกลงไปในตำแหน่งเดียวกันตามมุมมองของผู้ใช้ในแบบเวลาจริง (Real time) ซึ่งใน ARToolKit นี้ได้ใช้เทคนิคทางทัศนศาสตร์คอมพิวเตอร์ (Computer Vistion) ในการคำนวณตำแหน่งของกล้องเทียบกับตำแหน่งของสัญลักษณ์ เพื่อช่วยให้โปรแกรมเมอร์ง่ายต่อการนำภาพที่สร้างจากคอมพิวเตอร์กราฟิกใส่ลงไปยังตำแหน่งที่คำนวณได้ นอกจากนี้ ARToolKit มีส่วนของการติดตามสัญลักษณ์ (Marker Tracking) ที่รวดเร็วและแม่นยำทำให้เกิดการพัฒนาโปรแกรมประยุกต์เทคโนโลยีความจริงเสริมใหม่ๆที่น่าสนใจอย่างแพร่หลาย [4]

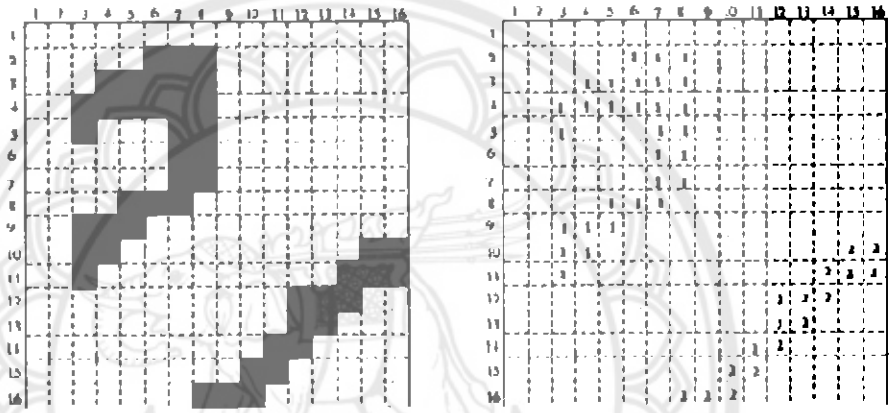
2.4 การหาค่าขีดแบ่ง [5]

การแยกบริเวณโดยการใช้ค่าขีดแบ่ง (Threshold) ค่าขีดแบ่งเป็นค่าที่เป็นจำนวนเต็มที่มีค่าระหว่าง 0 ถึง 255 เช่นเดียวกับค่าความเข้มแสงของพิกเซลที่อยู่ในภาพระดับเทา (Gray scale) สำหรับการแยกบริเวณโดยการใช้ค่าขีดแบ่งนั้น จะเป็นการแปลงภาพระดับเทาให้เปลี่ยนเป็นภาพที่มีเพียงสองระดับ (binary image) โดยการใช้เงื่อนไขว่าถ้าค่าความเข้มแสงที่พิกเซลตำแหน่งใดมีค่าต่ำกว่าหรือเท่ากับค่าขีดแบ่งให้ค่าพิกเซลในตำแหน่งนั้นมีค่าเป็น 0 หรือเปลี่ยนเป็นด้านมืดไป และถ้าพิกเซลใดมีค่าสูงกว่าค่าขีดแบ่งแล้วให้พิกเซลนั้นมีค่าเป็นค่า 255 หรือเปลี่ยนเป็นด้านสว่างไป ตัวอย่างดังรูปที่ 2.9

รูปที่ 2.9 แสดงภาพการแยกภาพค่าขีดแบ่ง [5]

2.5 การติดป้ายให้ส่วนประกอบที่ติดกัน [6]

การพิจารณาว่าพิกเซลใดบ้างที่มีการ "เชื่อมต่อ (connect)" กัน เพื่อที่จะจัดให้พิกเซลเหล่านั้นให้อยู่ในบริเวณหรือวัตถุเดียวกัน กระบวนการย่อยนี้เรียกว่าการติดป้ายให้ส่วนประกอบที่ติดกัน (Connected components labeling) ซึ่งผลที่ได้จากการทำงานของกระบวนการย่อยนี้ คือ จะทำให้รู้ว่าพิกเซลในแต่ละตำแหน่งนั้น จัดเป็นของวัตถุชิ้นใด ดังแสดงไว้ในรูปที่ ซึ่งแสดงหมายเลขของชิ้นวัตถุที่พิกเซลตำแหน่งนั้น ๆ เป็น "สมาชิก"อยู่ ตัวอย่างดังรูปที่ 2.10



รูปที่ 2.10 แสดงการทำงานของกระบวนการการติดป้ายให้ส่วนประกอบที่ติดกัน [6]

2.6 การตรวจจับจุดตัดผ่านศูนย์ [6]

การตรวจจับจุดผ่านศูนย์ (Zero crossing Detection) คือ ค่าที่ลาปาเซียนผ่านศูนย์ เป็นการหาอนุพันธ์อันดับสองของภาพเพื่อให้ได้ขอบของภาพ โดยภาพที่ผ่านการหาอนุพันธ์อันดับสอง (Laplacian Operator($\nabla^2 P$)) บริเวณที่เป็น ส่วนขอบจะเด่นชัดขึ้น การตรวจจับจุดตัดผ่านศูนย์ทำได้โดยแบ่งด้วยค่าขีดแบ่ง (thresholding) กับ ของ $\nabla^2 P$ (ตำแหน่งที่พิกเซลของ $\nabla^2 P$ เปลี่ยนจาก + เป็น – หรือจาก – เป็น +)

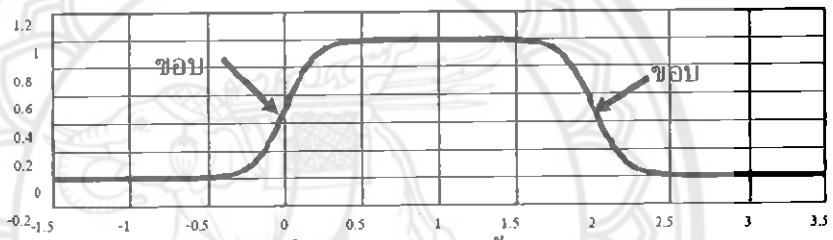
$$Mask(E_x) = \begin{bmatrix} Z_{x1} & Z_{x2} & Z_{x3} \\ Z_{x4} & Z_{x5} & Z_{x6} \\ Z_{x7} & Z_{x8} & Z_{x9} \end{bmatrix} = E_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (1)$$

$$Mask(E_y) = \begin{bmatrix} Z_{y1} & Z_{y2} & Z_{y3} \\ Z_{y4} & Z_{y5} & Z_{y6} \\ Z_{y7} & Z_{y8} & Z_{y9} \end{bmatrix} = E_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (2)$$

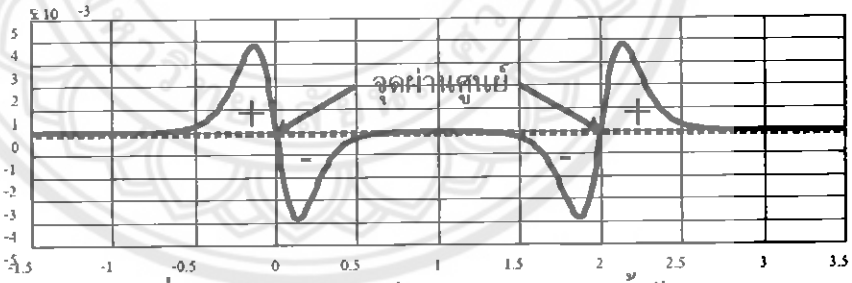
ทำให้ตำแหน่งของบริเวณขอบของวัตถุในภาพคือค่าจุดผ่านศูนย์ (Zero Crossing) ของ $\nabla^2 P$ การหาอนุพันธ์อันดับสอง สามารถหาได้จากสมการ

$$\nabla^2 P = \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial y^2} \tag{3}$$

การค้นหาขอบภาพอนุพันธ์อันดับสอง เมื่อทำ $\nabla^2 P$ บริเวณขอบจะมีตำแหน่งเดียวกับค่าจุดผ่านศูนย์ของค่าจาก $\nabla^2 P$ พิจารณาได้จากตำแหน่งที่พิกเซลเปลี่ยนแปลงจากค่าที่เป็นบวกเป็นค่าที่ เป็นลบหรือจากค่าที่เป็นลบไปเป็นค่าที่เป็นบวก ดังรูปที่ 2.11 และรูปที่ 2.12



รูปที่ 2.11 แสดงภาพตั้งต้น [6]



รูปที่ 2.12 แสดงอนุพันธ์อันดับสองของภาพตั้งต้น [6]

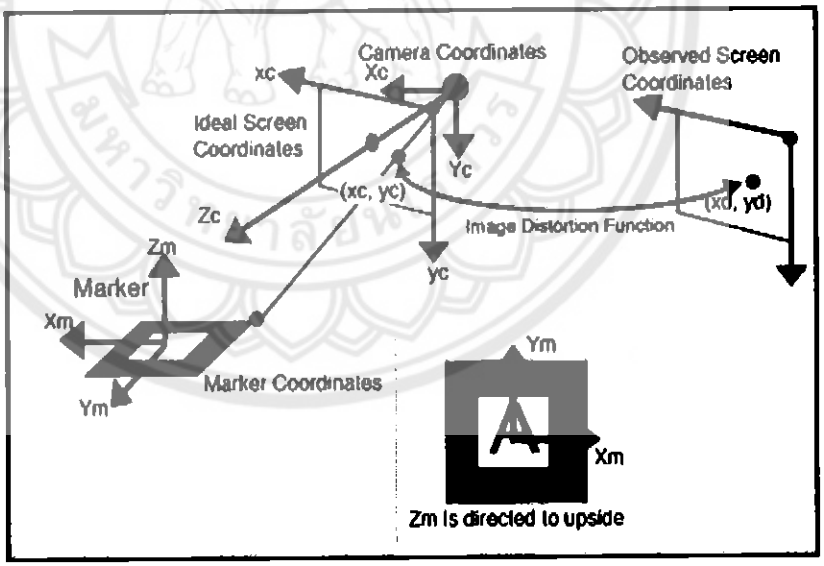
การหาขอบโดยใช้อนุพันธ์อันดับสอง ไม่สนใจทิศทางของภาพในแนวแกน x และ แกน y กำหนดจุดที่ค่า y เป็นจุดผ่านศูนย์ วิธีนี้ใช้เวลาในการคำนวณมากกว่าการค้นหาขอบโดยใช้ อนุพันธ์อันดับหนึ่ง กล่าวคือเราสามารถตรวจจับความไม่ต่อเนื่องของพิกเซลในโดเมนรูปภาพ ได้โดยใช้อนุพันธ์ของภาพนั้นเอง

2.7 การประมาณตำแหน่ง [6]

การประมาณตำแหน่ง (Pose Estimation) เป็นขั้นตอนของการคำนวณค่าตำแหน่งเชิง 3 มิติ (3D Pose) ของ Marker เมื่อเทียบกับกล้องวีดีโอ ค่านี้จะถูกแสดงในรูปเมตริกซ์ขนาด 4 x 4 (TCM) ที่ระบุความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame แสดงดังสมการที่ (1)

$$\begin{bmatrix} X_C \\ Y_C \\ Z_X \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_1 \\ R_{21} & R_{22} & R_{23} & T_2 \\ R_{31} & R_{32} & R_{33} & T_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_M \\ Y_M \\ Z_M \\ 1 \end{bmatrix} = T_{CM} \begin{bmatrix} X_M \\ Y_M \\ Z_M \\ 1 \end{bmatrix} \quad (4)$$

ซึ่ง Camera Coordinated Frame ก็คือ Coordinated Frame ที่ใช้อ้างอิงตำแหน่งใด ๆ ของกล้องวีดีโอ และ Marker Coordinated Frame ก็คือ Coordinated Frame ที่ใช้อ้างอิงตำแหน่งใด ๆ ของ Marker ซึ่งสามารถแสดงดังรูปที่ 2.13

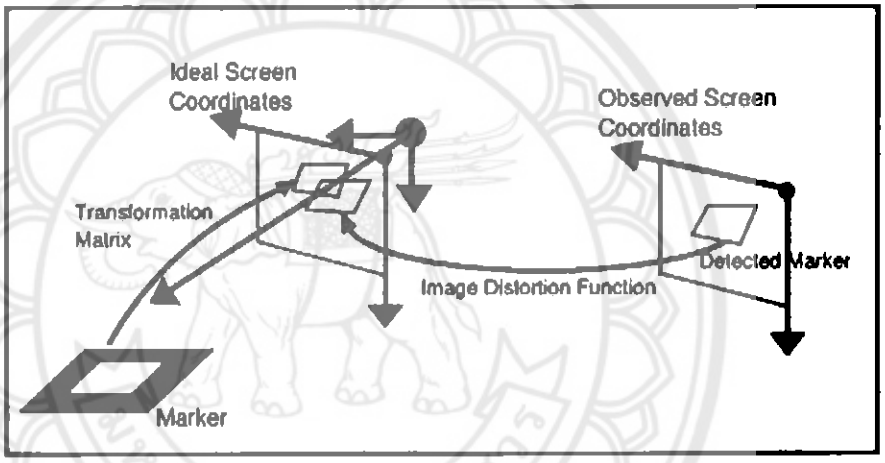


รูปที่ 2.13 ความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame [6]

จากรูปที่ 2.13 ความสัมพันธ์ระหว่างจุดใดๆ (Xc Yc Zc) บนCamera Coordinated Frame กับจุดที่ตรงกัน ใน Ideal Screen Coordinated Frame เป็นไปตาม Perspective Projection แสดงดังสมการที่ (2)

$$\begin{bmatrix} hx_I \\ hy_I \\ h \end{bmatrix} = \begin{bmatrix} sf_x & 0 & x_c & 0 \\ 0 & sf_y & y_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = C \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (5)$$

โดยที่ C ซึ่งเป็นเมตริกซ์ขนาด 3x4 ซึ่งประกอบไปด้วยค่า s, Fx, Fy, Xc, Yc โดยทั่วไปค่าเหล่านี้รวมกันเรียกว่า Camera Parameters ซึ่งจะคำนวณได้มาจากขั้นตอน Camera Calibration ส่วนค่าความสัมพันธ์ระหว่างจุดใด ๆ บน Ideal Screen Coordinated Frame (X1 Y1) กับ Observe Screen Coordinated Frame (X0 Y0) ซึ่งเป็นจุดที่เราเห็นจริง ๆ ในภาพ สามารถแสดงดังรูปที่ 2.14 และสามารถอธิบายได้แสดงดังสมการที่ (3)



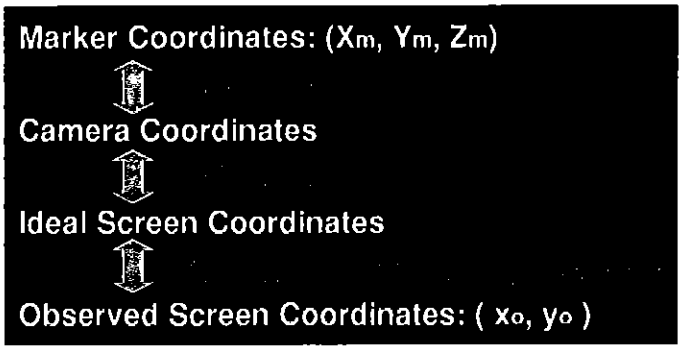
รูปที่ 2.14 ความสัมพันธ์ระหว่าง Ideal Screen Coordinates และ Observe Screen Coordinates [6]

$$d^2 = (x_1 - x_0)^2 + (y_1 - y_0)^2$$

$$p = \{1 - fd^2\}$$

$$x_0 = p(x_1 - x_0) + x_0, y_0 = p(y_1 - y_0) + y_0 \quad (6)$$

โดยที่ X0 Y0 คือจุดศูนย์กลางของการ Distortion (Center Coordinates of Distortion) และ f คือ Distortion Factor ซึ่งค่าทั้ง 2 จะได้มาจากกระบวนการ Camera Calibration



รูปที่ 2.15 กระบวนการคำนวณค่า 3D Poses [6]

จากรูปที่ 4 จะแสดงกระบวนการที่จะได้มาของค่า TCM เมื่อเรารู้ค่าตำแหน่งของ Marker ทั้ง 4 จุด บน Observed Screen Coordinates ในภาพที่ถ่ายจากกล้องวีดีโอ ซึ่งกล่าวโดยเฉพาะค่านี้สามารถหาได้จากการคำนวณหาคำตอบของฟังก์ชั่นค่าผิดพลาด (Error Function) แสดงดังสมการที่ (4) ซึ่งโดยทั่วไปแล้วเราจะใช้เทคนิคทางด้านการหาค่าที่เหมาะสม (Optimization) ซึ่งเป็นกระบวนแบบ Iterative

$$err = \frac{1}{4}\sum_{i=1,2,3,4}\{(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2\} \tag{7}$$

โดยที่ $\hat{x}_i$ $\hat{y}_i$ แสดงได้โดย

$$\begin{bmatrix} h\hat{x}_i \\ h\hat{y}_i \\ h \end{bmatrix} = \mathbf{C}.\mathbf{T}_{CM}\begin{bmatrix} X_{Mi} \\ Y_{Mi} \\ Z_{Mi} \\ 1 \end{bmatrix}, i = 1,2,3,4 \tag{8}$$

2.8 แมทริกซ์การแปลง (Tranformation Matrix) [7]

จากการประมาณตำแหน่งจะทำให้ได้ความสัมพันธ์ของแมทริกซ์ดังนี้ คือ แมทริกซ์ขนาด 3x4 โดยที่มีความสัมพันธ์เกี่ยวกับการปรับหมุน (rotation) เป็นขนาด 3x3 และ ความสัมพันธ์เกี่ยวกับทิศทาง (vector) เป็นขนาด 3x1 ดังรูปที่ 2.16

ความสัมพันธ์เกี่ยวกับทิศทาง (vector)

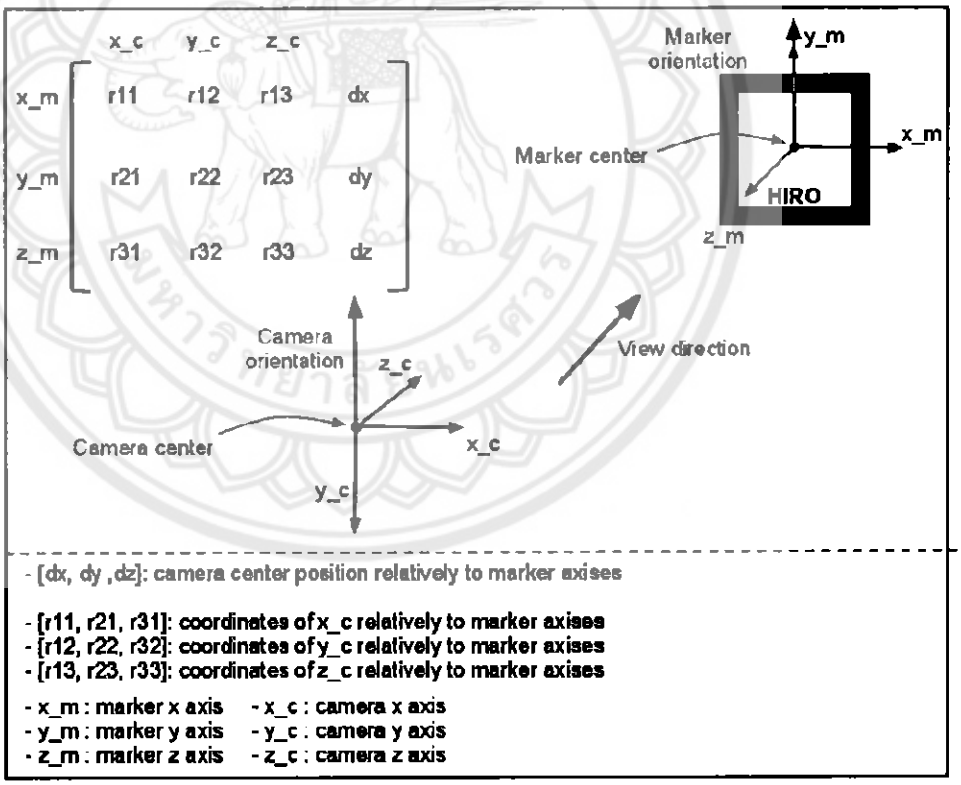
[dx ,dy, dz] คือ จุดศูนย์กลางของความสัมพันธ์ระหว่างกล้องกับสัญลักษณ์

ความสัมพันธ์เกี่ยวกับการปรับหมุน (rotation)

[r11, r12, r13] คือ กรอบพิกัดอ้างอิงแกน x ของกล้องกับสัญลักษณ์

[r21, r22, r23] คือ กรอบพิกัดอ้างอิงแกน y ของกล้องกับสัญลักษณ์

[r31, r32, r33] คือ กรอบพิกัดอ้างอิงแกน z ของกล้องกับสัญลักษณ์



รูปที่ 2.16 แสดงความสัมพันธ์แมทริกซ์การแปลง [7]

กรอบพิกัดอ้างอิงของสัญลักษณ์และกล้อง

x_m คือ แกน x ของสัญลักษณ์

y_m คือ แกน y ของสัญลักษณ์

z_m คือ แกน z ของสัญลักษณ์

x_c คือ แกน x ของกล้อง

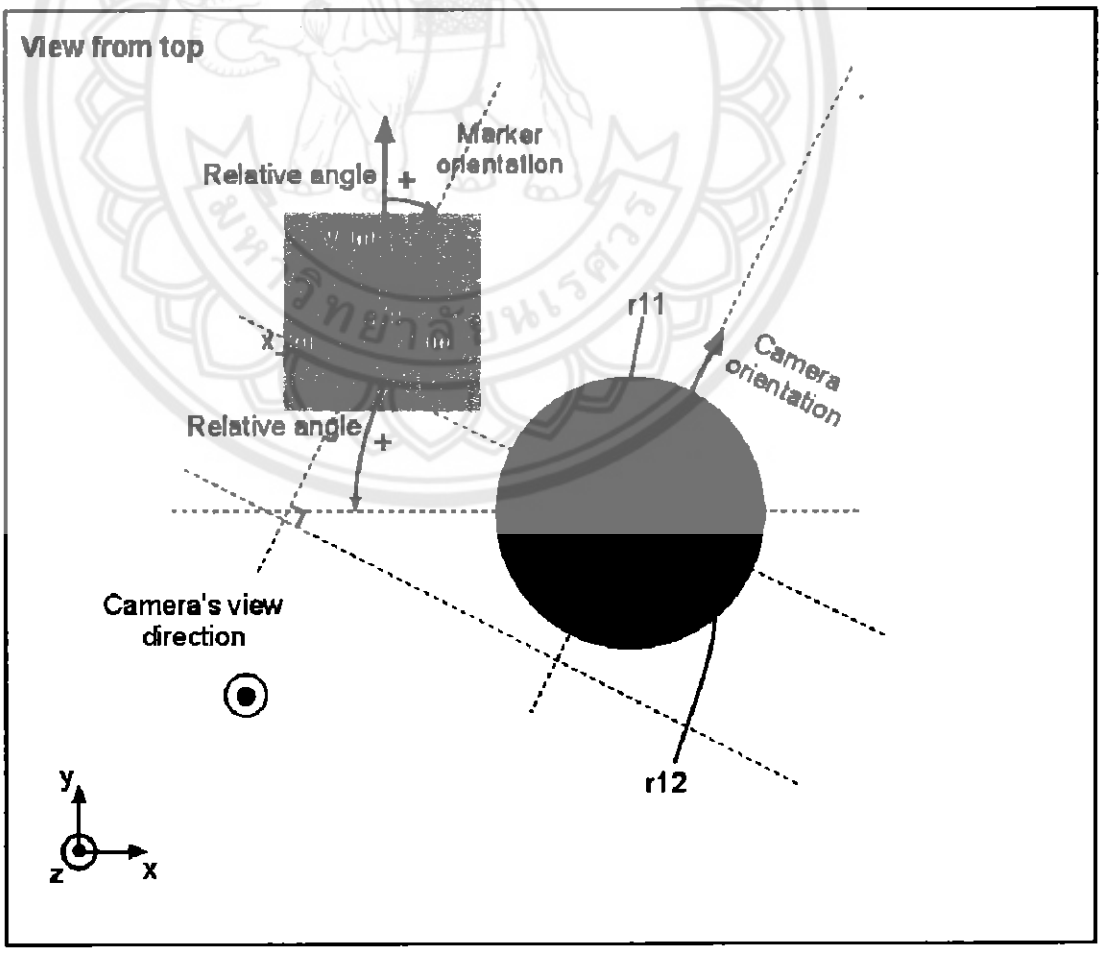
y_c คือ แกน y ของกล้อง

z_c คือ แกน z ของกล้อง

สามารถหาระยะทางระหว่างกล้องกับสัญลักษณ์ได้จากสมการนี้

ระยะทางระหว่างกล้องกับสัญลักษณ์ = $\sqrt{dx^2 + dy^2 + dz^2}$

ความสัมพันธ์มุม (Relative angle) เกิดจากการเปลี่ยนองศาระหว่างกล้องกับ**สัญลักษณ์ซึ่งเกิดจาก**ความสัมพันธ์บางส่วน ดังรูปที่ 2.17



รูปที่ 2.17 แสดงความสัมพันธ์ของมุมระหว่างกล้องกับสัญลักษณ์ [7]

ซึ่งสามารถคำนวณหาขนาดของมุมได้จากสมการนี้

$$\text{relative angle} = 2\tan^{-1}\frac{r_{12}}{r_{11}}$$

มุมที่ส่งกลับจะอยู่ในรูปมุมเรเดียนอยู่ระหว่าง $-\pi$ และ π โดยไม่นับมุม $-\pi$

2.9 การเรนเดอร์ภาพสามมิติ [8]

ส่วนนี้เป็นส่วนสุดท้ายที่จะทำให้กระบวนการสร้างความจริงเสริม ครบถ้วนสมบูรณ์ ซึ่งก็คือ การเพิ่ม (Augment) ข้อมูลที่เราต้องการซึ่งโดยทั่วไปแล้วจะเป็นโมเดล 3 มิติ (3D Model) ลงไปในภาพที่ได้จากกล้องวีดีโอ ณ ตำแหน่งของสัญลักษณ์ ที่ตรวจพบจากขั้นตอนการวิเคราะห์ภาพโดยใช้ค่าตำแหน่งเชิง 3 มิติที่คำนวณได้จากขั้นตอนประมาณตำแหน่ง กล่าวโดยทั่วไปแล้วการเรนเดอร์ภาพสามมิติ (3D Rendering) หมายถึง กระบวนการที่ทำการสร้างภาพ 2 มิติ จากโมเดล 3 มิติ ซึ่งโมเดล 3 มิตินี้ จะอธิบายวัตถุหรือสิ่งแวดล้อมหนึ่ง ๆ ที่ต้องการสร้างภาพนั้น ดังรูปที่ 2.18

รูปที่ 2.18 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ [8]

2.10 ความรู้เกี่ยวกับ openFrameWork [9]

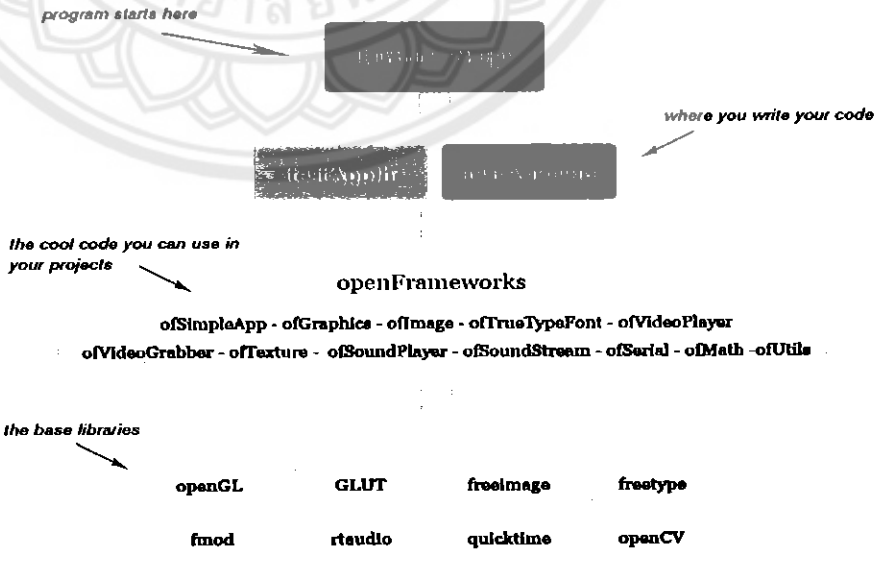
openFrameWork เป็นไลบรารี่อีกไลบรารี่หนึ่งที่มีการทำงาน คล้ายกับไลบรารี่ ARToolKit ซึ่ง openFrameWork มีการทำงาน 3 ส่วนหลักๆดังต่อไปนี้ [6]

- main.cpp เป็นส่วนที่ใช้สร้างตัวหน้าต่างของโปรแกรมขึ้นมากำหนดขนาด 800x600
- testApp.h เป็นส่วนที่ใช้ในการกำหนดตัวแปรขึ้นมาใช้
- testApp.cpp ส่วนนี้จะแบ่งออกเป็น 3 ส่วน

1. ฟังก์ชั่น setup() เป็นส่วนที่ใช้ตั้งค่าที่จะนำไปใช้ในโปรแกรมเช่นการตั้งค่าหน้าต่างของโปรแกรม การโหลดโมเดลของหน้ากากแฟนซีการที่จะเรียกไปวาดลงบนภาพสัญลักษณ์ การโหลดภาพ ฯลฯ

2. ฟังก์ชั่น update() เป็นส่วนที่ใช้ในการกำหนดค่าต่างๆของการแสดงภาพลงบนหน้าต่างของโปรแกรม เช่น การกำหนดค่าที่ภาพให้เป็นภาพสี ขาวดำ การกลับภาพวิดีโอ ฯลฯ

3. ฟังก์ชัน draw() ส่วนนี้เป็นส่วนให้เรียกโมเดลหน้ากากแฟนซี ภาพ ตัวอักษรสีลงในโปรแกรม อีกทั้งยังเป็นส่วนที่สร้างกระบวนการทำงานของโปรแกรมขึ้นมา เช่น การวาดโมเดลหน้ากากแฟนซี การเปลี่ยนหน้ากากแฟนซี การแสดงสถานะของหน้ากากแฟนซี การเรียกใช้ตัวอักษร ฯลฯ



รูปที่ 2.19 ขั้นตอนการทำงานของ openFrameWork [9]

บทที่ 3

วิธีการดำเนินโครงการ

จุดประสงค์ของโครงงานก็เพื่อที่จะสร้างโปรแกรมที่สามารถรับข้อมูลทางกล้องแล้วนำไปประมวลผลให้แสดงภาพหน้ากาก 3 มิติ ดังนั้นจึงต้องมีการทำงานหลาย ๆ ส่วนร่วมกันระหว่างฮาร์ดแวร์และซอฟต์แวร์ โดยมีขั้นตอนคร่าว ๆ ดังนี้

- ศึกษารายละเอียดและหลักการ ความรู้ ทฤษฎีต่าง ๆ ที่เกี่ยวข้องกับโครงงาน
- ออกแบบการทำงานทั้งหมด
- ออกแบบหน้ากาก 3 มิติ และการใช้งานภาพสัญลักษณ์
- ออกแบบตัวเลือกการใช้งานในโปรแกรม

3.1 ศึกษารายละเอียดและหลักการ ความรู้ ทฤษฎีต่าง ๆ ที่เกี่ยวข้องกับโครงงาน

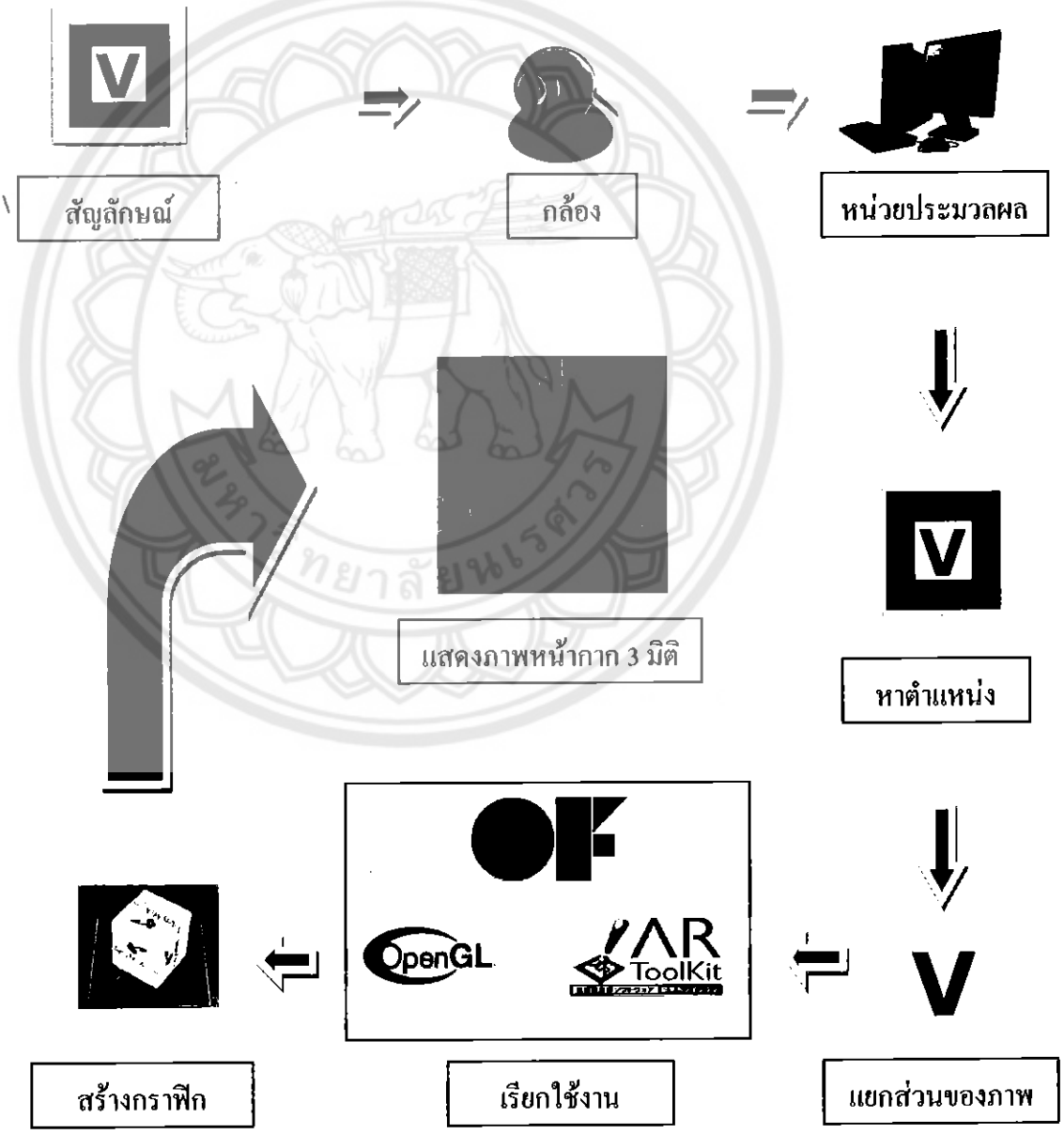
โครงงาน "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" แบ่งการทำงานหลักได้เป็น 2 ส่วน ได้แก่ ส่วนแรก คือ การตรวจจับและประมาณตำแหน่งและทิศทางตัวของสัญลักษณ์ (Marker) ซึ่งเครื่องมือที่ใช้ในการทำงานส่วนนี้คือไลบรารี่ARToolKit ส่วนที่สองการแสดงภาพกราฟิกให้สอดคล้องกับตำแหน่งและทิศทางของสัญลักษณ์ที่ตรวจจับได้ เครื่องมือที่ใช้ในการทำงานส่วนนี้ก็คือไลบรารี่ OpenGL และไลบรารี่ openFramework ซึ่งเครื่องมือทั้ง 3 อย่างนี้จะใช้หลักการเขียนโปรแกรมภาษา C++ เครื่องมือต่าง ๆ ที่ใช้แสดงดังรูปที่ 3.1

(ก) (ข) (ค)

รูปที่ 3.1 แสดงเครื่องมือที่ใช้

3.2 ออกแบบการทำงานทั้งหมด

หลักการทำงานของ "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" โดยจะนำภาพสัญลักษณ์มาส่องผ่านกล้อง จากนั้นคอมพิวเตอร์จะนำภาพที่ได้ไปประมวลผล แล้วหาตำแหน่งของภาพสัญลักษณ์เพื่อตรวจสอบว่ามีภาพนี้ในโปรแกรมหรือไม่ เมื่อพบภาพสัญลักษณ์โปรแกรมจะทำงานร่วมกับไลบรารี่ openGL, openFrameWork, ARToolkit เพื่อสร้างโมเดลหน้ากากขึ้นมา มีขั้นตอนต่างๆดังรูปที่ 3.2



รูปที่ 3.2 แสดงขั้นตอนการทำงานของ "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี"

โดยอธิบายหลักการทำงานของ รูปที่ 3.2 ดังนี้

1. อ่านสัญลักษณ์ผ่านเข้าทางกล้อง

2. กล้องจะส่งภาพสัญลักษณ์ที่เป็นกรอบสี่เหลี่ยมไปยังคอมพิวเตอร์เพื่อประมวลผล

3. ซอฟต์แวร์ทำการค้นหากรอบสี่เหลี่ยมซึ่งเรียกว่า "ภาพสัญลักษณ์" เพื่อหาตำแหน่งของภาพสัญลักษณ์ในโปรแกรม

4. จากนั้นซอฟต์แวร์จะทำการคำนวณเพื่อแยกแยะรูปร่างของ ภาพสัญลักษณ์ เพื่อที่จะเปรียบเทียบว่าตรงกันกับที่กำหนดไว้หรือไม่

5. หากถูกต้อง ก็จะสร้างรูปภาพวัตถุเสมือน (virtual object) ที่จะเป็นรูปแบบสอง หรือ3 มิติก็ได้ตามที่กำหนด

6. สุดท้ายเป็นขั้นตอนของการแสดงผลออกมาทางหน้าจอคอมพิวเตอร์เป็นภาพกราฟฟิกของหน้ากากแฟนซี

3.3 การทำงานของระบบความจริงสำหรับการลองหน้ากากแฟนซี โดยใช้ไลบรารี่ openFrameWork

- main.cpp เป็นส่วนที่ใช้สร้างตัวหน้าต่างของโปรแกรมขึ้นมากำหนดขนาด 800x600

- testApp.h เป็นส่วนที่ใช้ในการกำหนดตัวแปรขึ้นมาใช้

- testApp.cpp ส่วนนี้จะแบ่งออกเป็น 3 ส่วน

1. ฟังก์ชั่น setup() เป็นส่วนที่ใช้ตั้งค่าที่จะนำไปใช้ในโปรแกรม เช่น การตั้งค่าหน้าต่างของโปรแกรม การโหลดโมเดลของหน้ากากแฟนซีการที่จะเรียกไปวาดลงบนภาพสัญลักษณ์ การโหลดภาพ ฯลฯ

2. ฟังก์ชั่น update() เป็นส่วนที่ใช้ในการกำหนดค่าต่างๆของการแสดงภาพลงบนหน้าต่างของโปรแกรม เช่น การกำหนดค่าที่ภาพให้เป็นภาพสี ขาวดำ การกลับภาพวิดีโอ ฯลฯ

3. ฟังก์ชัน draw() ส่วนนี้เป็นส่วนให้เรียกโมเดลหน้ากากแฟนซี ภาพ ตัวอักษร สี ลงในโปรแกรม อีกทั้งยังเป็นส่วนที่สร้างกระบวนการทำงานของโปรแกรมขึ้นมา เช่น การวาดโมเดล

หน้ากากแฟนซี การเปลี่ยนหน้ากากแฟนซี การแสดงสถานะของหน้ากากแฟนซี การเรียกใช้ตัวอักษร ฯลฯ

3.4 การออกแบบหน้ากาก 3 มิติ

สำหรับการออกแบบนั้นเราใช้โปรแกรม 3Dmax studio โดยบันทึกไฟล์เป็นนามสกุล ".3ds" ในการสร้างหน้ากาก 3 มิติขึ้นมา โดยในโปรแกรมจะมีทั้งหมด 6 แบบให้เลือก

โมเดลต้นแบบ ภาพพื้นหลังลายหน้ากาก โมเดลหน้ากาก

รูปที่ 3.3 การสร้างโมเดลหน้าจากโมเดลต้นแบบโดยการใส่ภาพพื้นหลังลงบนโมเดลหน้ากาก

หน้ากากที่ออกแบบเพื่อใช้ในโครงงานนี้มีทั้งหมด 6 แบบ ดังแสดงในรูปที่ 3.5 ถึงรูปที่ 3.10

รูปที่ 3.4 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 1

รูปที่ 3.5 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 2

รูปที่ 3.6 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 3

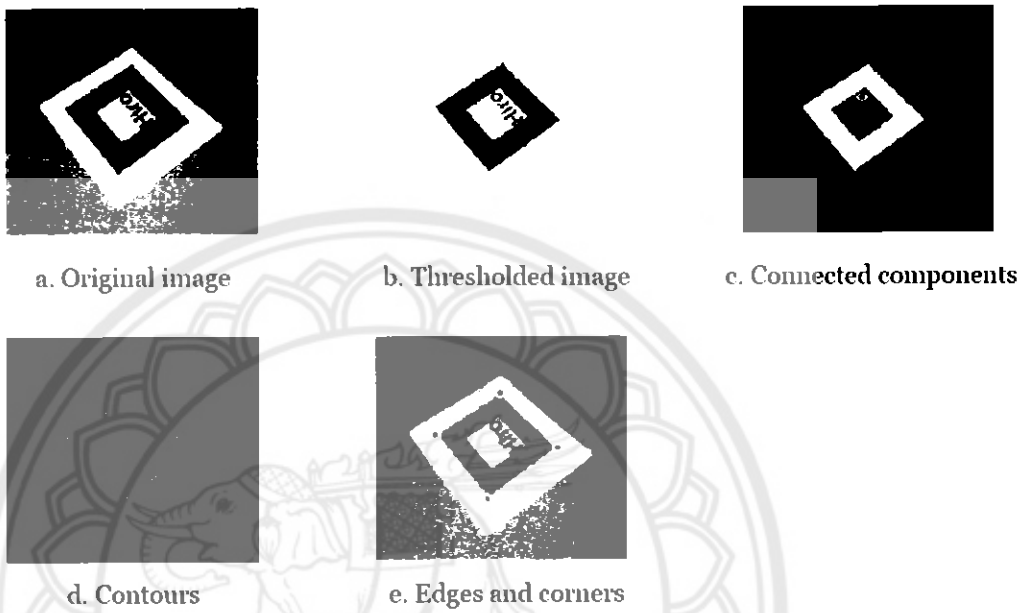
รูปที่ 3.7 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 4

รูปที่ 3.8 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 5

รูปที่ 3.9 ภาพตัวอย่างหน้ากาก 3 มิติ แบบที่ 6

3.5 ออกแบบการใช้งานและค้นหาสัญลักษณ์ [5]

การค้นหากรอบสี่เหลี่ยมจะใช้หลักการประมาณตำแหน่ง (pose estimation algorithm) โดยมีหลักการดังรูปที่ 3.10



a. Original image b. Thresholded image c. Connected components

d. Contours e. Edges and corners

รูปที่ 3.10 แสดงหลักการอ่านและค้นหาสัญลักษณ์ (Marker) [5]

a. รับภาพจากกล้อง

b. ทำการแปลงภาพให้กลายเป็นระดับไบนารีโดยการกำหนดให้แต่ละพิกเซลในภาพมีค่าเป็น 2 ระดับ คือ 0 หรือ 1 โดยที่จะมีค่าเป็น 0 เมื่อค่าระดับความสว่าง (Intensity) ของพิกเซล นั้นมีต่ำกว่าค่าขีดแบ่ง (Threshold) มิฉะนั้นแล้วจะมีค่าเป็น 1

c. ทำการหาพื้นที่ติดต่อกันโดยใช้เทคนิคทางการวิเคราะห์ภาพที่เรียกว่าติดป้ายองค์ประกอบที่ติดกัน (Connected component labeling)

d. เมื่อทำขั้นตอน c. เสร็จแล้วก็จะได้ผลลัพธ์เป็นเส้นรอบรูป (Contours)

e. เมื่อหาเส้นรอบรูปได้แล้ว ระบบจะทำการประมาณหาค่าพารามิเตอร์ของสมการเส้นตรงที่แทนเส้นรอบรูปซึ่งเป็นรูปสี่เหลี่ยมทั้ง 4 เส้น หลังจากนั้นระบบจะหาจุดมุม (Corners) ทั้ง 4 จุดของสัญลักษณ์จากจุดตัดของเส้นตรงทั้งสี่ที่หาได้ ซึ่งจุดมุม 4 จุดนี้จะถูกนำไปใช้ในกระบวนการประมาณตำแหน่ง

3.5.1 การประมาณตำแหน่ง [6]

เป็นขั้นตอนของการคำนวณค่าตำแหน่งเชิง 3 มิติ (3D Pose) ของสัญลักษณ์เมื่อเทียบกับกล้องวีดีโอ ค่านี้จะถูกแสดงในรูปเมตริกซ์ขนาด 4x4 ที่ระบุความสัมพันธ์ระหว่างกรอบพิกัดอ้างอิงของกล้องและกรอบพิกัดอ้างอิงของสัญลักษณ์ ซึ่งกรอบพิกัดอ้างอิงของกล้องก็คือกรอบพิกัดอ้างอิงที่ใช้อ้างอิงตำแหน่งใดๆของกล้องวีดีโอ และกรอบพิกัดอ้างอิงสัญลักษณ์ก็คือกรอบพิกัดอ้างอิงที่ใช้อ้างอิงตำแหน่งใดๆของสัญลักษณ์

3.5.2 3D rendering [8]

ส่วนนี้เป็นส่วนสุดท้ายที่จะทำให้กระบวนการ AR ครบถ้วนสมบูรณ์ ซึ่งก็คือ การเพิ่ม (Augment) ข้อมูลที่เราต้องการซึ่งโดยทั่วไปแล้วจะเป็นโมเดล 3 มิติ (3D Model) ลงไปในภาพที่ได้จากกล้องวีดีโอ ณ ตำแหน่งของ Marker ที่ตรวจพบจากขั้นตอน Image Analysis โดยใช้ค่าตำแหน่งเชิง 3 มิติที่คำนวณได้จากขั้นตอน Pose Estimation กล่าวโดยทั่วไปแล้ว 3D Rendering หมายถึงกระบวนการที่ทำการสร้างภาพ 2 มิติ จากโมเดล 3 มิติ ซึ่งโมเดล 3 มิตินี้ จะอธิบายวัตถุหรือสิ่งแวดล้อมหนึ่ง ๆ ที่ต้องการสร้างภาพนั้น ตัวอย่างดังรูปที่ 3.11

รูปที่ 3.11 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ [8]

3.6 การออกแบบส่วนติดต่อกับผู้ใช้

3.6.1 การแสดงโมเดลหน้ากาก

โปรแกรมจะทำการวาดโมเดลหน้ากากแฟนซีลงบนสัญลักษณ์ที่กำหนดไว้ในโปรแกรม เมื่อส่องผ่านกล้อง โดยกำหนดให้สัญลักษณ์ที่ใช้สร้างโมเดลหน้ากากคือภาพสัญลักษณ์ตัวอย่างดังในรูปที่ 3.12

รูปที่ 3.12 ตัวอย่างการวาดโมเดลลงบนสัญลักษณ์

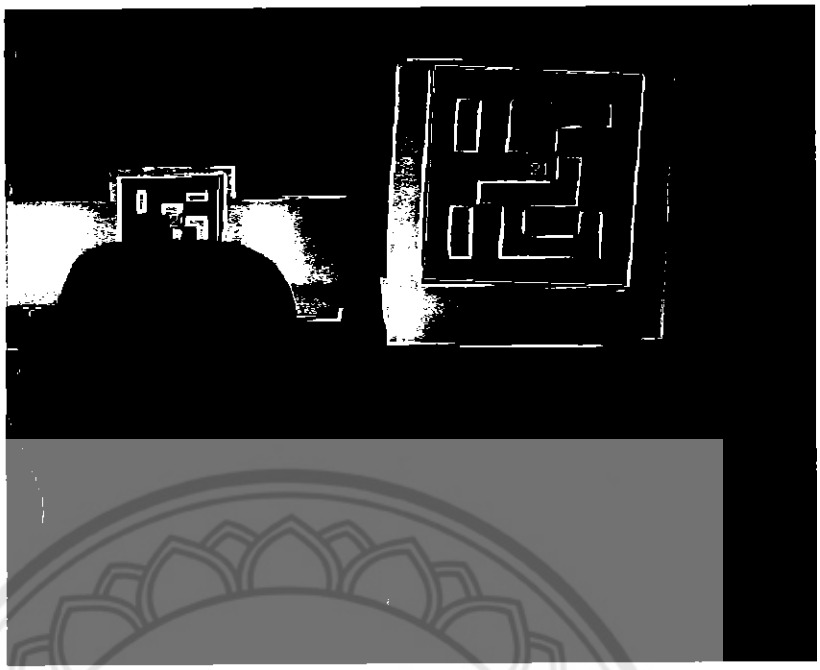
3.6.2 การออกแบบสถานะของโมเดลหน้ากากขณะแสดงผล

โดยจะทำการโหลดภาพตัวอย่างหน้ากากลงไปบนจุดๆ หนึ่งที่เราต้องการในหน้าต่างของโปรแกรมเมื่อเกิดการเปลี่ยนแปลงโมเดลหน้ากากขึ้น โปรแกรมจะทำการวาดรูปกรอบสี่เหลี่ยมขึ้นรอบรูปตัวอย่างหน้ากากที่ตรงกับโมเดลหน้ากากที่กำลังแสดงผลอยู่ขณะนั้น

รูปที่ 3.13 ตัวอย่างภาพสถานะของโมเดลหน้ากาก

3.6.3 การเปลี่ยนโมเดลหน้ากาก

โปรแกรมจะทำการเปลี่ยนหน้ากากเมื่อพบสัญลักษณ์ A และสัญลักษณ์ B โดยสัญลักษณ์ A จะทำการเพิ่ม Model Index ของโมเดลหมวกในโปรแกรมทำให้โมเดลหน้ากากเปลี่ยนไปในตำแหน่งต่อจากตำแหน่งเดิมไปเรื่อยๆ และให้สัญลักษณ์ B ทำการลด Model Index ทำให้โมเดลหน้ากากเลื่อนกลับมายังโมเดลแรกเรื่อยๆ(โมเดลหน้ากากทำการโหลดเก็บไว้ในโปรแกรมในรูปของ Array)



รูปที่ 3.14 เมื่อตรวจพบภาพสัญลักษณ์ A จะทำให้ Model Index เพิ่มและทำการเปลี่ยนโมเดลหน้ากากไปทางขวาเรื่อยๆ

รูปที่ 3.15 เมื่อตรวจพบภาพสัญลักษณ์ B จะทำการลด Model Index ทำให้เลื่อนโมเดลหน้ากากไปยังด้านซ้ายเรื่อยๆ

บทที่ 4

ผลการทดลอง

ส่วนนี้จะเป็นส่วนที่รวบรวมผลการทดลองทั้งหมดที่มีผลกับการใช้งานของ "ระบบความจริงสำหรับการลองหน้ากากแฟนซี" ไม่ว่าจะเป็นการทดลองที่เกี่ยวเนื่องกับสร้างโมเดลหน้ากาก การตรวจจับภาพสัญลักษณ์ และปัจจัยที่มีผลต่อการทดลอง

4.1 ผลการทดลองการตรวจจับภาพสัญลักษณ์ (Marker)

เมื่อนำภาพสัญลักษณ์ส่องผ่านกล้องโปรแกรมจะทำการตรวจหาว่าภาพสัญลักษณ์ ตรงกับค่าหมายเลขภาพสัญลักษณ์ในโปรแกรมเพื่อที่จะดำเนินการขั้นต่อไป ดังรูปที่ 4.1

รูปที่ 4.1 หมายเลขภาพสัญลักษณ์ X

จาก รูปที่ 4.1 จะกำหนดให้โปรแกรมสร้างโมเดลหน้ากากเมื่อตรวจพบภาพสัญลักษณ์ X

รูปที่ 4.2 หมายเลขภาพสัญลักษณ์เท่ากับหมายเลข A ที่จะใช้เลื่อนโมเดลไปทางขวามือ

รูปที่ 4.3 หมายเลขภาพสัญลักษณ์เท่ากับหมายเลข B ที่จะใช้เลื่อนโมเดลไปทางซ้ายมือ

จาก รูปที่ 4.2 และ รูปที่ 4.3 เป็นภาพที่กำหนดให้โปรแกรมทำงานเมื่อพบภาพสัญลักษณ์ A,B โดยกำหนดว่าเมื่อพบ ภาพสัญลักษณ์ A โมเดลจะเลื่อนไปทางขวา และภาพสัญลักษณ์ B โมเดลจะเลื่อนไปทางซ้าย

4.2 ผลการทดลองการวาดโมเดลหน้ากากลงบนภาพสัญลักษณ์

เมื่อโปรแกรมตรวจจับภาพสัญลักษณ์แล้วตรวจสอบว่าหมายเลขภาพสัญลักษณ์คือหมายเลขที่ "X" จากนั้นโปรแกรมจะทำการวาดโมเดลหน้ากากลงบนภาพสัญลักษณ์(กำหนดภายในโปรแกรมให้หมายเลขภาพสัญลักษณ์ที่ X ใช้ในการวาดโมเดลหน้ากาก) ดังรูปที่ 4.4

รูปที่ 4.4 จากรูปที่ 4.1 โปรแกรมจะทำการวาดโมเดลหน้ากากลงบนภาพสัญลักษณ์หมายเลขที่ X

4.3 ผลการทดลองการใช้ภาพสัญลักษณ์ในการเปลี่ยนโมเดลหน้ากาก

เมื่อโปรแกรมทำการวาดโมเดลหน้ากากขึ้นมาแล้ว ผู้ใช้สามารถเปลี่ยนหน้ากากตามแบบที่ต้องการโดยการใช้ภาพสัญลักษณ์ A และ B ดังรูปต่อไปนี้

รูปที่ 4.5 โปรแกรมตรวจพบภาพสัญลักษณ์ A

จาก รูปที่ 4.4 เมื่อโปรแกรมสร้างโมเดลหน้าขึ้นมาแล้ว จากนั้นตรวจพบภาพสัญลักษณ์ A จะทำให้เลื่อนโมเดลไปทางขวา

รูปที่ 4.6 โปรแกรมตรวจพบภาพสัญลักษณ์ B

จาก รูปที่ 4.4 เมื่อโปรแกรมสร้างโมเดลหน้าขึ้นมาแล้ว จากนั้นตรวจพบภาพสัญลักษณ์ B จะทำให้เลื่อนโมเดลไปทางซ้าย

4.4 ปัจจัยที่มีผลต่อผลการทดลอง

4.4.1 มุมของภาพสัญลักษณ์

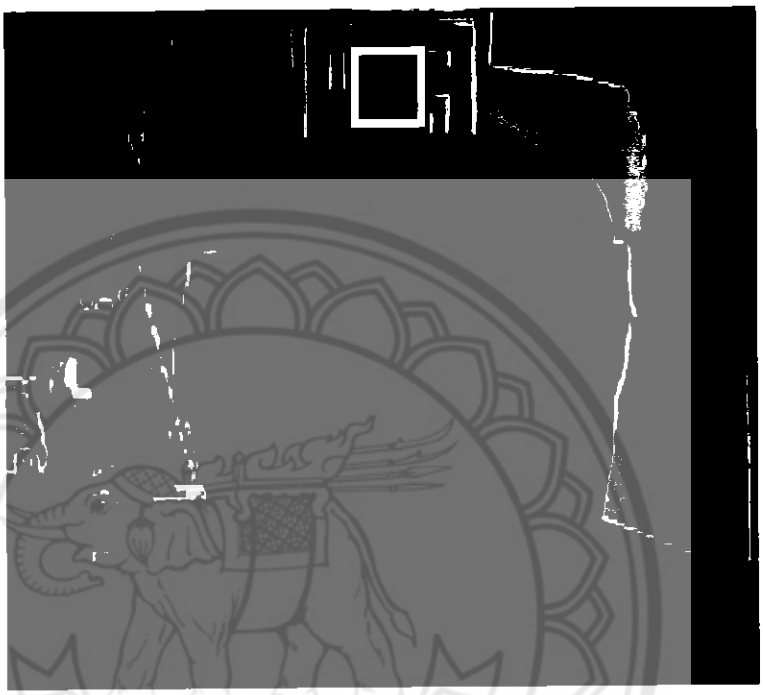
การแสดงผลของโมเดลหน้ากากจะเปลี่ยนแปลงตามมุมของภาพสัญลักษณ์ที่เปลี่ยนแปลงไปในทิศทางต่างๆ ซึ่งมุมที่สามารถตรวจจับภาพสัญลักษณ์มากที่สุดประมาณ 70 – 75 องศา โดยมุม 0 องศา จะสามารถตรวจจับภาพสัญลักษณ์ได้ดีที่สุดดังรูปที่ 4.7

รูปที่ 4.7 การแสดงผลของโมเดลหน้ากากเมื่อมุมของภาพสัญลักษณ์เปลี่ยนไป

เป็นภาพที่แสดงการทดสอบมุมของการตรวจจับภาพสัญลักษณ์ของกล้องที่ใช้ในการทดลอง โดยมุมที่สามารถตรวจจับได้มีมุมตั้งแต่ 0 – 75 องศาโดยประมาณ

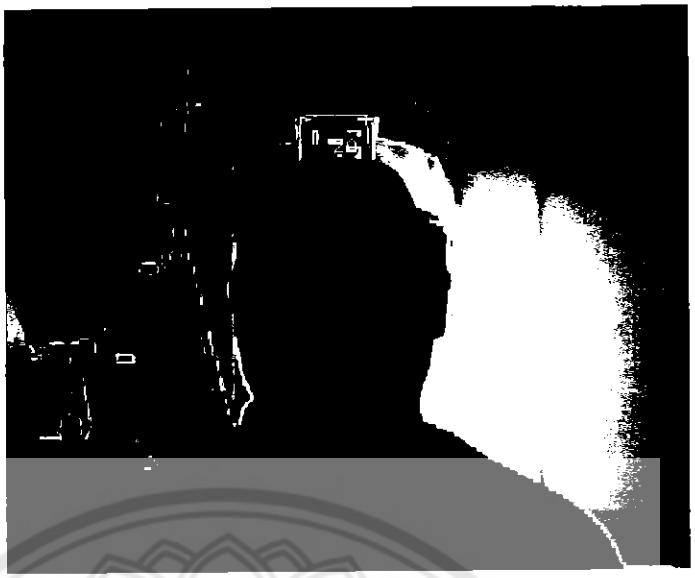
4.4.2 ระยะห่างของกล้องกับภาพสัญลักษณ์

ระยะทางระหว่างภาพสัญลักษณ์กับกล้องมีผลอย่างชัดเจนเมื่อมีระยะห่างที่แตกต่างกัน ดังรูปต่อไปนี้



รูปที่ 4.8 ภาพสัญลักษณ์มีระยะห่างจากกล้องประมาณ 30-50 เซนติเมตร

จากรูปที่ 4.8 เมื่อระยะห่างของกล้องกับภาพสัญลักษณ์ประมาณ 30-50 เซนติเมตร จะเป็นระยะที่โปรแกรมสามารถตรวจจับภาพสัญลักษณ์ได้ดีที่สุด เพราะเป็นระยะที่โปรแกรมสามารถนำภาพสัญลักษณ์มาประมวลผลชัดเจนที่สุด ทำโปรแกรมแสดงผลโมเดลหน้ากากได้ดี



รูปที่ 4.9 ภาพสัญลักษณ์มีระยะจากกล้องระยะประมาณ 80 - 100 เซนติเมตร

จาก รูปที่ 4.9 เมื่อภาพสัญลักษณ์มีระยะห่างจากกล้องเพิ่มขึ้นเป็น 80-100 เซนติเมตร แต่โปรแกรมก็ยังสามารถตรวจจับภาพสัญลักษณ์ได้ชัดเจนอยู่

รูปที่ 4.10 ภาพสัญลักษณ์มีระยะห่างจากกล้อง 120 เซนติเมตร

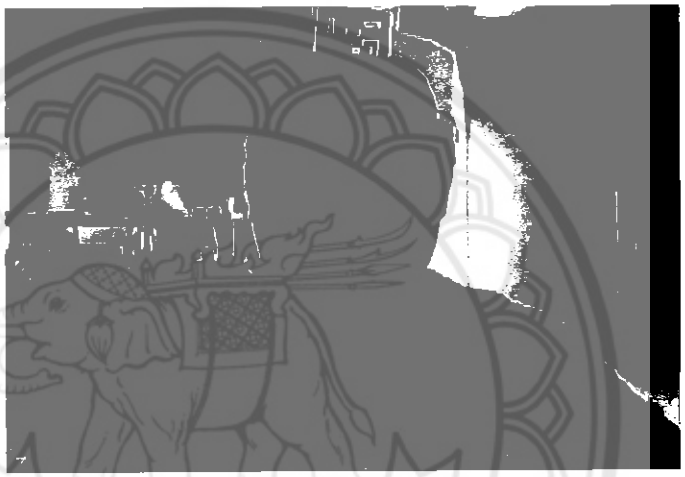
จาก รูปที่ 4.10 เมื่อระห่างมากขึ้นทำให้การตรวจจับเป็นไปได้ยากมากยิ่งขึ้น จากรูปจะเห็นว่าโปรแกรมยังสามารถตรวจจับภาพสัญลักษณ์ได้ แต่โปรแกรมไม่สารถแสดงโมเดลหน้ากากขึ้นมาได้เพราะระยะห่างที่มากเกินไปทำให้ภาพสัญลักษณ์ที่ตรวจจับไม่สามารถนำไปประมวลผลได้

รูปที่ 4.11 ระยะห่างระหว่างกล้องกับภาพสัญลักษณ์มีระยะ 180 เซนติเมตรขึ้นไป

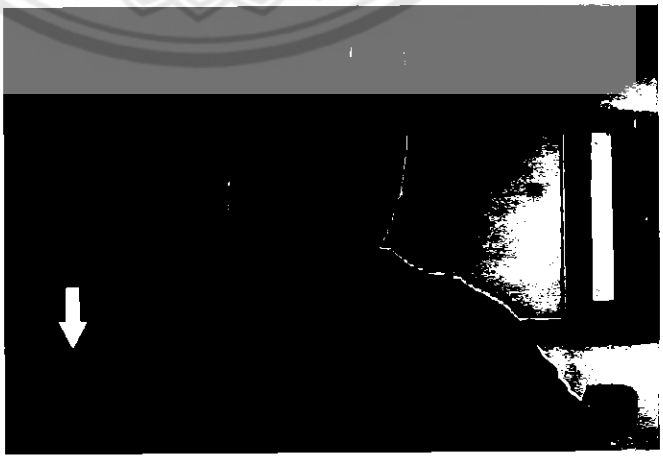
จากรูปที่ 4.11 เมื่อระยะห่างระหว่างกล้องกับภาพสัญลักษณ์มีมากกว่า 180 เซนติเมตรขึ้นไป จะทำให้การตรวจจับภาพสัญลักษณ์เพื่อไปประมวลผลในโปรแกรมไม่สามารถทำได้เพราะเกินขีดจำกัดของกล้องที่จะตรวจจับภาพสัญลักษณ์ได้

4.4.3 แสง

แสงมีผลอย่างมากกับการจับภาพสัญลักษณ์หากมีแสงสว่างไม่เหมาะสม คือเราทำการทดลองในบริเวณทดลองที่มีแสงน้อยมากเกินไป แต่ถ้าเราทดลองในที่บริเวณที่มีแสงสว่างพอเหมาะโดยประมาณจากแสงจากหลอดฟลูออเรสเซนส์ 2-3 หลอด หลอดละ 36 watt ดังรูปดังต่อไปนี้



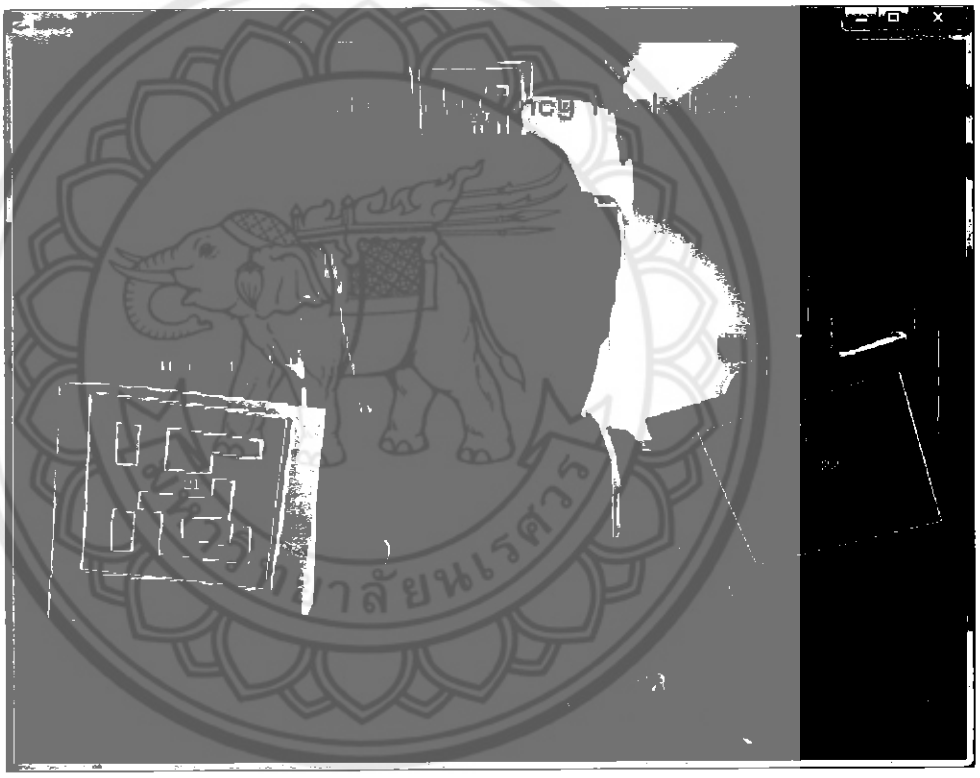
รูปที่ 4.12 ทดลองในที่ ที่แสงเหมาะสมทำให้การแสดงผลออกตามปกติ



รูปที่ 4.13 ทดลองในที่ ที่มีแสงน้อยทำให้ตรวจหาภาพสัญลักษณ์ไม่ได้

4.5 สรุปผลการทดลอง

จากการทดลอง "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" สามารถสร้างโมเดลหน้ากากแฟนซีลงบนภาพสัญลักษณ์ที่ต้องการได้ และสามารถใช้งานได้อย่างดี สามารถเปลี่ยนหน้ากากด้วยการใช้ภาพสัญลักษณ์ A และ B อีกทั้งยังมีแถบแสดงสถานะโมเดลของหน้ากากว่าขณะที่โมเดลหน้ากากแสดงผลอยู่เป็นโมเดลหน้ากากใด และยังออกแบบอุปกรณ์ที่ใช้เป็นแบบสวมทำให้สามารถเพิ่มมิติและความสะดวกสบายในการใช้งานได้ง่ายมากยิ่งขึ้น



รูปที่ 4.14 ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี

บทที่ 5

สรุปผลและข้อเสนอแนะ

บทนี้กล่าวถึงผลการทดลองของโครงงาน ปัญหาที่พบขณะทำโครงงาน และข้อเสนอแนะของโครงงาน "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี" (Augmented Reality for Masks Fitting) เพื่อเพิ่มความเข้าใจในการใช้งานการทำงานของโปรแกรมมากยิ่งขึ้น

5.1 สรุปผลการทดลอง

โครงงานนี้จัดทำขึ้นเพื่อศึกษาการทำงานของระบบความจริงเสริม (Augmented Reality) เพื่อสร้างโปรแกรมที่สามารถลองหน้ากากแฟนซี เพื่อให้เกิดรู้ความเข้าใจเพิ่มมากขึ้นอีกทั้งยังจะเป็นประโยชน์กับผู้ที่สนใจ โดยโปรแกรมจะทำงานเมื่อมีภาพสัญลักษณ์ที่กำหนดส่องผ่านกล้องแล้วแสดงผลโมเดลหน้ากากแฟนซีขึ้นบนหน้าต่างของโปรแกรม โดยจะมีภาพสัญลักษณ์อีก 2 ภาพ ที่ใช้ในการเปลี่ยนโมเดลหน้ากากตามแบบที่ต้องการได้ อีกทั้งยังมีแถบแสดงสถานะการแสดงผลในขณะนั้นๆ ของโมเดลหน้ากากด้วย ทำให้เกิดแปลกใหม่กับผู้ใช้ อีกทั้งยังได้ประสบการณ์ใหม่ๆอีกด้วย และยังมีความรู้ความเข้าใจในระบบความจริงเสริมเพิ่มมากขึ้น

5.2 ปัญหาที่พบระหว่างการทำโครงงาน

จากการทดลองจะพบว่าจะมีปัญหาหลักๆเลยก็คือ การศึกษาหาความรู้เกี่ยวกับ ไลบรารี่ต่างๆที่ใช้ในโครงงาน และสภาพแวดล้อมในการทดลอง อุปกรณ์ที่ใช้ในการทดลอง

1. ปัญหาเกี่ยวกับการศึกษา ไลบรารี่ ARToolKit ไลบรารี่ openFrameWork ที่ต้องใช้เวลาการศึกษาค่อนข้างมากเพราะยังไม่มีความรู้และยังไม่เคยได้ใช้งานมาก่อน
2. ปัญหาเกี่ยวกับการทดลอง เช่น เรื่อง แสงไม่พอ หรือมากเกินไป ทำให้การตรวจจับภาพสัญลักษณ์มีปัญหา ทำให้โมเดลหน้ากากแสดงผลออกมาค่อยข้างมีปัญหา

5.3 ข้อเสนอแนะและวิธีการแก้ไขปัญหา

จากปัญหาที่พบในโครงงานนี้ ทางผู้จัดทำขอเสนอข้อเสนอแนะ และวิธีการแก้ไขปัญหาดังนี้

1. ต้องมีการศึกษาการใช้งานไลบรารี่ก่อนที่จะทำโครงงาน โดยศึกษาจากเอกสารของผู้มีความรู้ที่มีสอนอยู่ และวางแผนศึกษาเป็นขั้นตอน จนพอมีความรู้ ความเข้าใจพอสมควร จึงเริ่มทำโครงงาน

2. ปัญหาเกี่ยวกับแสงไม่เพียงพอ ทำให้การตรวจจับภาพสัญลักษณ์มีปัญหา แก้ปัญหาโดยการใช้ภาพสัญลักษณ์ที่ชัดเจน บริเวณที่ทำการทดลองต้องมีแสงสว่างเพียงพอ

5.4 ข้อเสนอแนะสำหรับงานในอนาคต

1. ทำการพัฒนารูปแบบของการตรวจภาพสัญลักษณ์ให้มีคุณภาพมากยิ่งขึ้น ในเรื่องของมุมและแสงของภาพสัญลักษณ์

2. ออกแบบโมเดลหน้ากากที่จะใช้ในโปรแกรมให้เพิ่มมากยิ่งขึ้น มีความหลากหลายมากขึ้น

3. นำไปประยุกต์ใช้กับสิ่งของอื่นๆ เช่น แว่นตา นาฬิกา แหวน เสื้อผ้า ฯลฯ

เอกสารอ้างอิง

[1] ARToolKit Development Principles. สืบค้นเมื่อ 11 สิงหาคม 2555, จาก http://www.hitl.washington.edu/artoolkit/documentation/devprinciple.htm.

[2] "OpenGL คืออะไร".สืบค้นเมื่อ 24 กรกฎาคม 2555, จากhttp://202.44.14.219/~isurnich/ftp/CG/html/OpenGL.htm.

[3] "RGB คืออะไร อาร์จีบี คือระบบสีของแสง". สืบค้นเมื่อ 26 กรกฎาคม2555, จาก http://www.mindphp.com/คู่มือ/73-คืออะไร/2172-rgb-คืออะไร.html.

[4] "เทคโนโลยีความจริงเสริม" .สืบค้นเมื่อ 25 กรกฎาคม 2555, จากhttp://www.artymix.com/flartoolkit_chap2.php.

[5] การศึกษาเทคโนโลยีออคเมนต์เตดเรียลริตี้ กรณีศึกษาพัฒนาเกมส์ "เมมการ์ด". ผู้จัดทำ วสันต์ เกียรติแสงทอง,พรรษพล พรหมมาศ และอนุวัตร เฉลิมสกุลกิจ. สืบค้นเมื่อ 17 พฤศจิกายน 2555, จาก *facstaff.swu.ac.th/praditm/CP499_2552_AR.pdf.*

[6] "Introduction to Machine Vision". Retrieved march .สืบค้นเมื่อ 7 พฤษภาคม 2556, จาก cdn.researchers.in.th/assets/media/files/000/.../original_chapter01.pdf.

[7] แสดงความสัมพันธ์แมทริกซ์การแปลง. สืบค้นเมื่อ 3 พฤษภาคม 2555, จาก http://www.omf.mytestbed.net/projects/robot/wiki/RobotControlSoftware?version=51.

[8] กระบวนการเรนเดอร์ภาพสามมิติ .สืบค้นเมื่อ 17 ธันวาคม 2555, จาก http://www.artymix.com/flartoolkit_chap2.php.

[9] "openFrameWork".สืบค้นเมื่อ 15 มกราคม 2556, จาก http://www.openframeworks.cc/documentation/.

[10] แสดงความสัมพันธ์ของมุมระหว่างกล้องกับสัญลักษณ์. สืบค้นเมื่อ 3 พฤษภาคม 2555, จาก http://www.omf.mytestbed.net/projects/robot/wiki/RobotControlSoftware?version=51.

[11] ดร.วิวัฒน์. "Augmented Reality." สืบค้นเมื่อ 25 เมษายน 2556, จาก http://www.edu.nu.ac.th/wiwatm.

[12] เกมเสมือนสามมิติ .สืบค้นเมื่อ 11 พฤษภาคม 2556, จาก http://sephirosuy.blogspot.com/2011/04/nintendo-3ds-star-pics-and-how-to.html

[13] รูปแสดงตัวอย่างโปรแกรมเทคโนโลยีความจริงเสริมกับการศึกษา .สืบค้นเมื่อ 9 พฤษภาคม 2556,จาก http://felipesequerra.com/tag/augmented-reality/

[14] รูปที่นำเทคโนโลยีความจริงเสริมไปใช้การเชิงการแพทย์ .สืบค้นเมื่อ 8 พฤษภาคม 2556, จาก http://endigy.com/2.7_AR_medicine.php

[15] ภาพแสดงตัวอย่างตึกอาคารในเมือง .สืบค้นเมื่อ 12 พฤษภาคม 2556, จาก http://www.cleartag.com/blog/category/augmented-reality/

[16] "3dMax studio".สืบค้นเมื่อ 27 มกราคม 2556, จาก http://www.webthaidd.com/3dmax/.

[17] "Mask Model".สืบค้นเมื่อ 1 กุมภาพันธ์ 2556, จาก http://www.turbosquid.com/.

[18] "วิธีการติดตั้ง ARToolKit".สืบค้นเมื่อ 30 กรกฎาคม 2555, จาก http://www.learners.in.th/blogs/posts/299487.

[19] การทำงานของ OpenGL.สืบค้นเมื่อ 11 พฤษภาคม 2556, จาก http://202.44.14.219/~isurnich/ftp/CG/html/OpenGL.htm

[20] รูปของระบบสี RGB.สืบค้นเมื่อ 7 พฤษภาคม 2556, จาก http://www.klongdigital.com/photoshop/photoshop12

[21] แสดงภาพการแยกภาพค่าขีดแบ่ง.สืบค้นเมื่อ 1 พฤษภาคม 2556, จาก http://matlab.exponenta.ru/imageprocess/book2/80.php.สืบค้นเมื่อ 10 พฤษภาคม 2556,

[22] การทำงานของกระบวนการการติดป้ายให้ส่วนประกอบที่ติดกัน .สืบค้นเมื่อ 2 พฤษภาคม 2556, จาก cdn.researchers.in.th/assets/media/files/000/.../original_chapter01.pdf

[23] ความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame. สืบค้นเมื่อ 3 พฤษภาคม 2556, จาก http://www.artymix.com/flartoolkit_chap2.php.

[24] Augmented Reality Model for Fungi Simulation การจำลองตัวแบบภาพเสมือนสำหรับการจำลองอาณาจักรเห็ดรา. สืบค้นเมื่อ 9 กันยายน 2554, จาก http://202.28.94.53/e-Project/ProjectWeb.php?id=74.

[25] "Introduction to Machine Vision". Retrieved march .สืบค้นเมื่อ 7 พฤษภาคม 2556, จาก cdn.researchers.in.th/assets/media/files/000/.../original_chapter01.pdf.

[26] *Salim Bensiali.*"Control Software for the OMF". สืบค้นเมื่อ 10 พฤษภาคม 2556, จาก http://omf.mytestbed.net/projects/robot/wiki/RobotControlSoftware?version=51.

[27] ความสัมพันธ์ระหว่าง Ideal Screen Coordinates และ Observe Screen Coordinates. สืบค้นเมื่อ 3 พฤษภาคม 2556, จาก http://www.artymix.com/flartoolkit_chap2.php.

ภาคผนวก

1. คู่มือการใช้งาน "ระบบความจริงเสริมสำหรับการลองหน้ากากแฟนซี"

1. สวมอุปกรณ์ที่มี

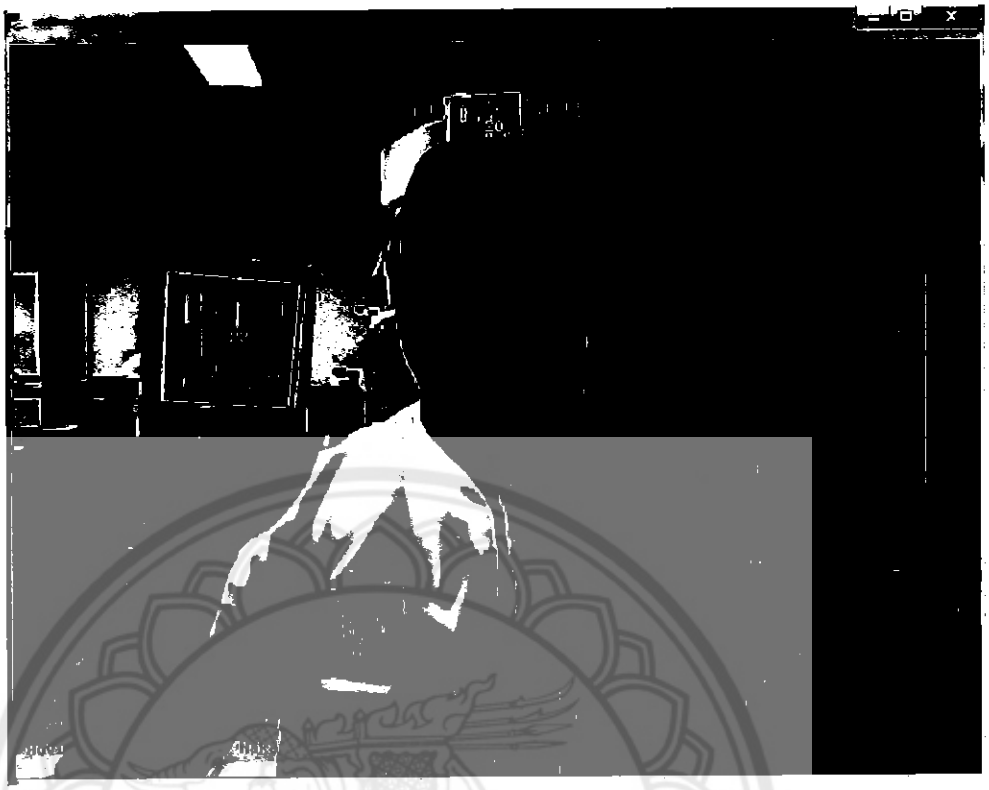
2. รันไฟล์ ARfancyMasks.exe

3. ใช้งานได้เลย

1.1 ฟังก์ชั่นในการใช้งาน

การเปลี่ยนโมเดลหน้ากาก

รูปตัวอย่างการเปลี่ยนโมเดลหน้ากาก



วิธีการ : ใช้ภาพสัญลักษณ์ ID=21,22 ในการเปลี่ยนโมเดลหน้ากากในแบบต่างๆ

การแสดงสถานะของโมเดลหมวก

วิธีการ : แถบแสดงสถานะของโมเดลหน้ากากแฟนซีที่แสดงผลในขณะนั้น

1.2 อุปกรณ์ที่ใช้

1. ภาพสัญลักษณ์ที่ผู้พัฒนาโครงงานกำหนดในโปรแกรม
2. กล้อง
3. คอมพิวเตอร์
4. โปรแกรม

2. ทดลองการใช้งานกล้อง และการติดตั้งไลบรารี่ ARToolKit

download Microsoft Visual Studio 2010 หรือ version ไหนก็ได้ และ installให้เรียบร้อย

download DSVL-0.0.8b.zip จาก http://sourceforge.net/projects/artoolkit/files/artoolkit/ (download มาเตรียมไว้ก่อน)

download GLUT จาก http://www.xmission.com/~nate/glut.html เลือก glut-3.7.6-bin.zip (117 KB) กด save แล้วแตกไฟล์ จากนั้น

3.1 เอาไฟล์ glut32.dll ไปไว้ใน c:\windows\system32

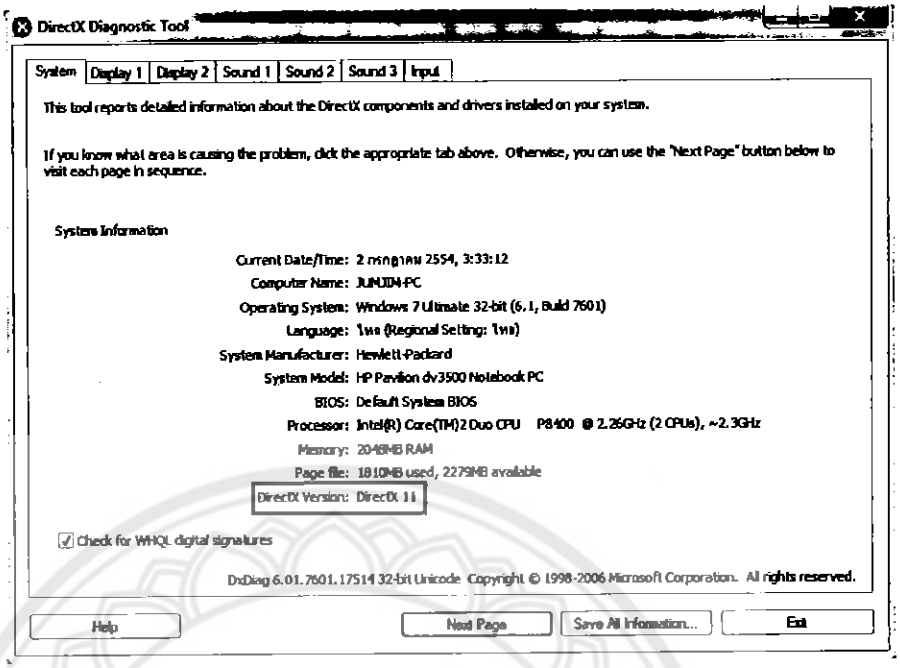
3.2 สร้างโฟลเดอร์ gl , เอาไฟล์ glut.h ไปไว้ในโฟลเดอร์ gl แล้ว เอาโฟลเดอร์ gl ไปไว้ใน C:\Program Files\Microsoft Visual Studio 10.0\VC\include

3.3 เอาไฟล์ glut32.lib ไปไว้ใน C:\Program Files\Microsoft Visual Studio 10.0\VC\lib

เอาไฟล์ MSVCP71D.dll และ MSVCR71D.dll ที่ส่งไปให้ ไปไว้ที่ c:\windows\system32

ตรวจสอบเวอร์ชั่น ของ directX ที่มีในเครื่องของเราโดย start >> run >>dxdiag >>OK

directX version จะต้องมากกว่า 9.0b ขึ้นไป



ต่อสายกล้อง ทดสอบกล้องว่าใช้งานได้ป่าว

download ARToolKit แตกไฟล์ ไว้ที่สำหรับใช้งาน เช่น C:\Program Files\ARToolKit

แตกไฟล์ DSVL-0.0.8b.zip ที่โหลดมา ใส่ไว้ใน C:\Program Files\ARToolKit ที่เราเพิ่งแตกเมื่อกี้ (ข้อ6) (Make sure that the directory is named "DSVL".)

Copy the files DSVL.dll and DSVLd.dll จาก C:\Program Files\ARToolKit\DSVL\bin ไปไว้ที่ C:\Program Files\ARToolKit\bin

รันสคริปต์ C:\Program Files\ARToolKit\Configure.win32.bat

เปิด microsoft visual studio เลือก open project เลือก ARToolKit (type= Microsoft Visual Studio Solution Object) จาก C:\Program Files\ARToolKit

Build >> Build Solution

เสร็จต้อง build succeeded ,0 failed

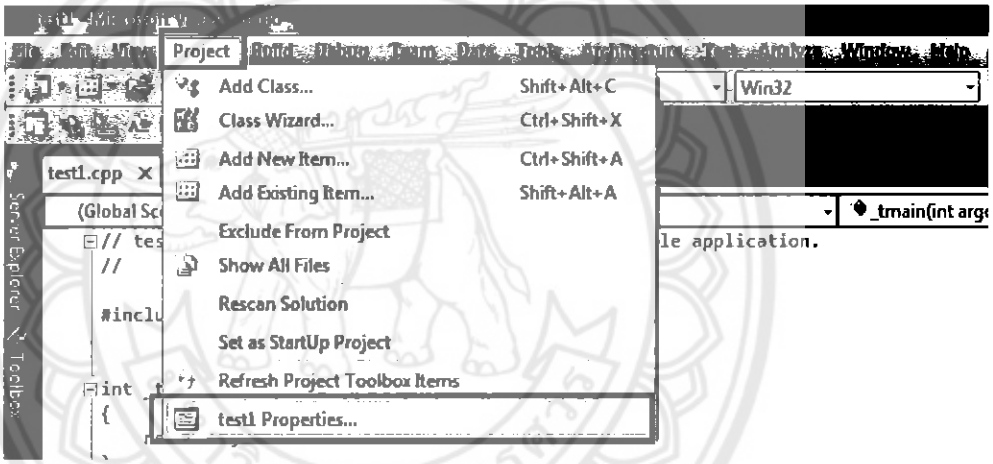
***หมายเหตุ ถ้า Build ไม่สำเร็จ ให้ทำดังนี้

เปิด microsoft visual studio ขึ้นมา เลือก new project ขึ้นมา

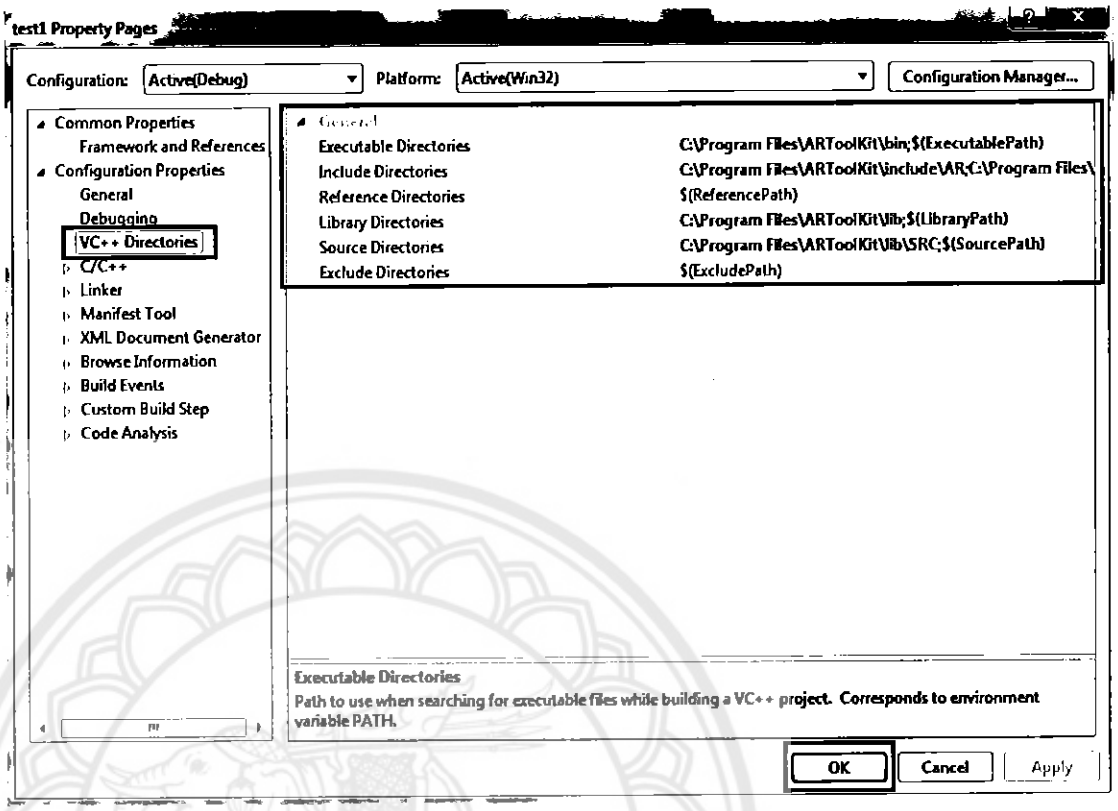
Copy Code จากไฟล์ simpleLite.c ใน ARToolKit (type= Microsoft Visual Studio Solution Object) มาวางที่ project ของเรา

Set ค่าใน microsoft visual studio ดังนี้

ไปที่ Project >> ชื่อไฟล์ Properties…



เลือก VC++ Directories และทำการเพิ่มข้อมูลในหน้าต่างด้านขวาดังนี้



Executable Directories : C:\Program Files\ARToolKit\bin

Include Directories : C:\Program Files\ARToolKit\include\AR และ C:\Program Files\ARToolKit\include

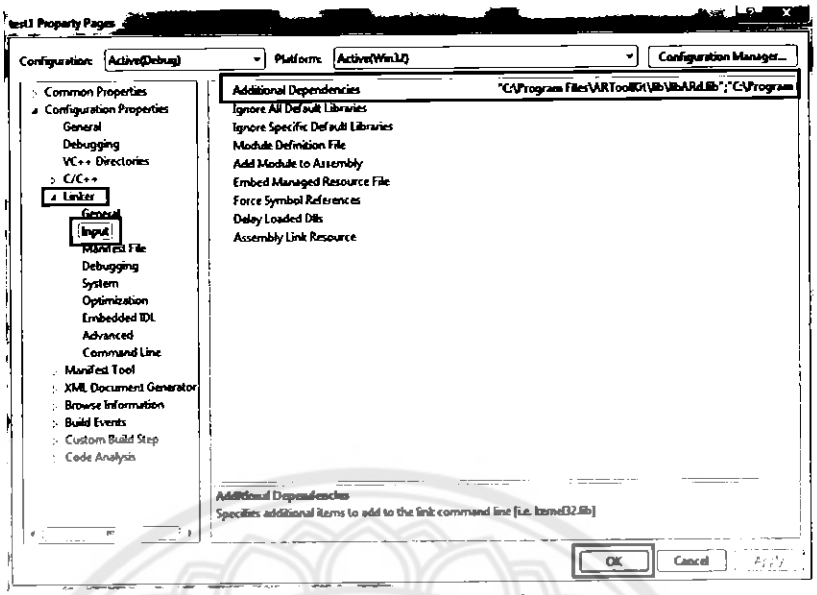
Reference Directories : เหมือนเดิม

Library Directories : C:\Program Files\ARToolKit\lib

Source Directories : C:\Program Files\ARToolKit\lib\SRC

Exclude Directories : เหมือนเดิม

จากนั้นไปเลือกที่ Linker >> Input



แล้วใส่ข้อมูลที่ Additional Dependencies ดังนี้

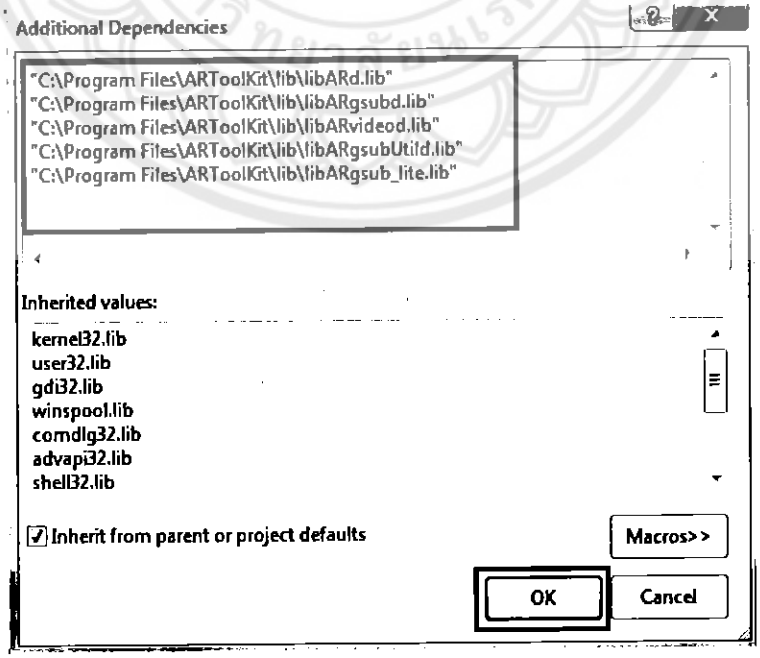
C:\Program Files\ARToolKit\lib\libARd.lib

C:\Program Files\ARToolKit\lib\libARgsubd.lib

C:\Program Files\ARToolKit\lib\libARvideod.lib

C:\Program Files\ARToolKit\lib\libARgsubUtild.lib

C:\Program Files\ARToolKit\lib\libARgsub_lite.lib



และสุดท้ายให้ copy เอา folder data , DSVL.dll, DSVLd.dll , js32.dll , libARvideo.dll และ libARvideod.dll จาก C:\Program Files\ARToolKit\bin ไปไว้ที่ไฟล์ .exe ของโปรเจ็คที่เราจะรัน เช่น C:\Users\jumjim\Documents\Visual Studio 2010\Projects\test1\Debug

เท่านี้ก็เรียบร้อย!!

จากนั้นก็ Build >> Build Solution โปรเจ็คของเรา แล้วก็รัน ทดลองเล่น AR ได้ตามสบาย

** สามารถปริ้นตัว marker ได้ที่ C:\Program Files\ARToolKit\patterns

3. การติดตั้ง OpenGL

Install Visual C++ 2010 Express

- Go to http://www.microsoft.com/express/
  and download the VC++ 2010 Express
- Install and Register the product

Install GLUT Library

- Go to http://www.xmission.com/~nate/glut.html
  and download glut-3.7.6-bin.zip (117 KB)
- Copy glut32.dll to "C:\Windows\System32"
- Copy glut32.lib to "C:\Program Files\Microsoft Visual Studio 10.0\VC\lib"
- Copy glut.h to "C:\Program Files\Microsoft Visual Studio 10.0\VC\Include\gl"
  Note that you may have to create the gl directory.
- Note that for newer and more active version of GLUT, try *freeglut* at http://freeglut.sourceforge.net and http://www.transmissionzero.co.uk/software/freeglut-devel/

เสร็จสิ้นการติดตั้ง