ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี

AUGMENTED REALITY FOR MARKETING PROMOTION OF DVD

นายธีรวุฒิ เลี้ยงวณิชย์ รหัส 51364811

นางสาวสุปราณี โล่งกระโทก รหัส 51365016

ห้องสมุดคณะวิศวกรรมศาสตร์
วันที่รับ 2 ก.ค. 2556
เลขทะเบียน 16280043
เลขเรียกหนังสือ ปร
มหาวิทยาลัยนเรศวร ธ 649 ส 2554

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ ภาควิชาวิศวกรรมไฟฟ้าและคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

ปีการศึกษา 2554

ใบรับรองปริญญานิพนธ์

ชื่อหัวข้อโครงงาน	**ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี**
ผู้ดำเนินโครงงาน	**นายธีรวุฒิ เลี้ยงวณิชย์ รหัส 51364811**
	นางสาวสุปราณี โล่งกระโทก รหัส 51365016
ที่ปรึกษาโครงงาน	นายรัฐภูมิ วรานุสาสน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2554

.....

คณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร อนุมัติให้ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิศวกรรมศาสตรบัณฑิต สาขาวิชาวิศวกรรมคอมพิวเตอร์

.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. พนมขวัญ ริยะมงคล)

.....กรรมการ
(ดร. พงศ์พันธ์ กิจสนาโยธิน)

.....กรรมการ
(อาจารย์เศรษฐา ตั้งค้าวานิช)

.....กรรมการ
(อาจารย์รัฐภูมิ วรานุสาสน์)

ชื่อหัวข้อโครงงาน	ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี
ผู้ดำเนินโครงงาน	ธีรวุฒิ เลี้ยงวณิชย์ รหัส 51364811
	สุปราณี โล่งกระโทก รหัส 51365016
ที่ปรึกษาโครงงาน	อ.รัฐภูมิวรานุสาสน์
สาขาวิชา	วิศวกรรมคอมพิวเตอร์
ภาควิชา	วิศวกรรมไฟฟ้าและคอมพิวเตอร์
ปีการศึกษา	2554

บทคัดย่อ

โครงงานนี้เป็นการศึกษาและประยุกต์ใช้เทคโนโลยีความจริงเสริมเพื่อนำมาพัฒนาระบบความจริงเสริมในการผลิตสื่อโฆษณาเพื่อส่งเสริมการตลาดของสินค้าประเภทดีวีดี เพื่อสร้างความดึงดูดใจและประสบการณ์ใหม่ให้กับผู้บริโภค โดยการทำงานหลักแบ่งได้เป็น 3 ส่วนได้แก่ ส่วนแรก คือ การตรวจจับและการประมาณตำแหน่งและทิศทางของวัตถุสัญลักษณ์ ซึ่งติดอยู่ด้านบนของหน้าปกแผ่นดีวีดี ผ่านทางกล้องเว็บแคม ส่วนที่สอง คือ การแสดงผลโมเดลสามมิติและวิดีโอโดยอาศัยข้อมูลตำแหน่งและทิศทางของวัตถุสัญลักษณ์ในการเลือกระหว่างที่แสดง และส่วนที่สาม คือ การแสดงภาพกราฟิกให้สอดคล้องกับตำแหน่งและทิศทางของวัตถุสัญลักษณ์ที่ตรวจจับได้ โครงงานนี้แสดงให้เห็นถึงความเป็นไปได้ ในการประยุกต์ใช้งานทางด้านเทคโนโลยีความจริงเสริมเพื่อความบันเทิง และความสะดวกสบายให้กับผู้บริโภค ซึ่งมีความแปลกใหม่น่าสนใจ

Project title	Augmented reality for marketing promotion of DVD	
Name	Mr. TheerawutLeangwanit	ID. 51364811
	Miss. SupraneeLongkratok	ID. 51365016
Project advision	Mr. RattapoomWaranusast	
Major	Computer Engineering	
Department	Electrical and Computer Engineering	
Academic year	2011	

Abstract

This project studies the augmented reality technology and its applications in order to support marketing and advertisement of DVD sales. This technology will be attracting and offering a new experience to the consumers. Theprocess can be divided into three major steps, first is to detect and estimate object positionsand orientationsvia a web camera based on AR markers attaching to the DVD box.The second is to create three-dimensional models and videos using poses of the detected markers. The last step is to render the generated models and videosonto the display screen. This study shows the possibilities of applying the augmented reality technology in daily life such as entertainment, recreation and convenience to serve consumers.

กิตติกรรมประกาศ

โครงงานนี้สำเร็จลุล่วงไปได้ด้วยความกรุณาเป็นอย่างยิ่งจากอาจารย์ที่ปรึกษา อาจารย์ รัฐภูมิ วรานุสาสน์ ที่คอยให้คำปรึกษาและคอยแนะนำวิธีการแก้ปัญหามากมาย และให้ความกรุณาในการตรวจทานปริญญานิพนธ์ คณะผู้จัดทำโครงงานขอกราบขอบพระคุณเป็นอย่างสูงและขอระลึกถึงความกรุณาของท่านไว้ตลอดไป

ผู้ช่วยศาสตราจารย์ ดร.พนมขวัญ ริยะมงคล,ดร.พงศ์พันธ์ กิจสนาโยธิน และ นายเศรษฐา ตั้งค้าวานิช ที่กรุณาสละเวลา เป็นอาจารย์กรรมการสอบโครงงาน พร้อมทั้งให้คำแนะนำที่เป็นประโยชน์

ขอบคุณมหาวิทยาลัยนเรศวรที่ทำให้ได้รับความอบอุ่นเหมือนเป็นบ้านหลังที่สองของพวกเราและให้เราได้พบกับชีวิตที่มีสีสันในมหาวิทยาลัย

ขอบคุณรุ่นพี่และเพื่อนทุกคนที่คอยให้คำปรึกษาและแนะแนวทางในการดำเนินโครงงานตั้งแต่ในระดับขั้นเริ่มต้นและคอยให้กำลังใจเรื่อยมา

สุดท้ายที่ขาดไม่ได้เลยคือขอขอบคุณครอบครัวที่ให้การสนับสนุนในหลายๆเรื่องเรื่อยมา อีกทั้งยังคอยให้กำลังใจและห่วงใยอยู่เสมอประโยชน์ความดีและคุณค่าอันพึงมีจากโครงงานนี้ขอมอบแด่ทุกท่านครับ

คณะผู้จัดทำ

นายธีรวุฒิ เลี้ยงวณิชย์

นางสาวสุปราณี โล่งกระโทก

สารบัญ

หน้า

บทคัดย่อภาษาไทย..... ก

บทคัดย่อภาษาอังกฤษ..... ข

กิตติกรรมประกาศ..... ค

สารบัญ..... ง

สารบัญตาราง..... ฉ

สารบัญรูป..... ช

บทที่ 1 บทนำ

1.1 ที่มาและความสำคัญของโครงงาน..... 1

1.2 วัตถุประสงค์ของโครงงาน..... 3

1.3 ขอบข่ายของโครงงาน..... 3

1.4 ขั้นตอนการดำเนินงาน..... 3

1.5 แผนผังการดำเนินโครงงาน..... 4

1.6 ผลที่คาดว่าจะได้รับ..... 5

1.7 งบประมาณของโครงงาน..... 5

บทที่ 2 หลักการและทฤษฎี

2.1 เทคโนโลยีความจริงเสริม (Augmented Reality)..... 6

2.2 ความรู้เบื้องต้นเกี่ยวกับ ARToolKit..... 12

2.3 การสร้างภาพสามมิติโดยใช้ OpenGL (3D Rendering with OpenGL)..... 17

2.4 ความรู้เบื้องต้นเกี่ยวกับ OpenCV..... 23

2.5 Marker Detection..... 24

2.6 Threshold..... 27

2.7 สี RGB..... 27

2.8 ความรู้เบื้องต้นเกี่ยวกับ irrKlang..... 28

สารบัญ(ต่อ)

หน้า

บทที่ 3 วิธีการดำเนินโครงงาน

3.1 ออกแบบการทำงาน..... 30

3.2 ออกแบบการสร้างวัตถุสัญลักษณ์..... 33

3.3 ออกแบบการสร้างวัตถุสามมิติ..... 38

3.4 ออกแบบการสร้าง Texture Mapping..... 42

3.5 ออกแบบการสร้างภาพเคลื่อนไหว (Animation)..... 50

บทที่ 4 ผลการทดลอง

4.1 ผลการทดลองการค้นหาวัตถุสัญลักษณ์ (Marker)..... 54

4.2 ผลการทดลองการติดตั้งอุปกรณ์..... 56

4.3 ผลการทดลองของโปรแกรม..... 57

บทที่ 5 บทสรุปและข้อเสนอแนะ

5.1 สรุปผลการทดลอง..... 62

5.2 ปัญหาที่พบ..... 62

5.3 ข้อเสนอแนะและวิธีการแก้ไขปัญหา..... 63

เอกสารอ้างอิง..... 64

ภาคผนวก..... 65

ประวัติผู้เขียนโครงงาน..... 77

สารบัญตาราง

ตารางที่ หน้า

1.1 แผนการดำเนินงาน..... 4

4.1 แสดงผลการทำงานของโปรแกรม..... 57

สารบัญรูป

รูปที่ หน้า

1.1 แสดงการประยุกต์ใช้เทคโนโลยีความจริงเสริมในการแสดงสินค้าตัวอย่างของ Lego 1

1.2 แสดงการประยุกต์ใช้เทคโนโลยีความจริงเสริมในหนังสือ ZooBurst..... 2

2.1 แสดงแผนภาพการทำงานของระบบความจริงเสริม (Augmented Reality)..... 8

2.2 แสดงส่วนของการจำลองสมอง..... 9

2.3 แสดงการประยุกต์ใช้เทคโนโลยีเสมือนจริงจำลองการผ่าตัดผ่านระบบ ARI*SER..... 10

2.4 แสดงการโปรโมทสินค้าของ adidas..... 10

2.5 แสดงการให้บริการทางด้าน GPS..... 10

2.6 แสดงการประยุกต์ใช้เทคโนโลยีเสมือนจริงในการทดลองนาฬิกา..... 11

2.7 แสดงตัวอย่างการสร้างภาพเสมือนจริง..... 12

2.8 แสดงรูปแบบที่เหมาะสมของ Marker..... 13

2.9 การทำ original image..... 13

2.10 การทำ Threshold image..... 13

2.11 การทำ Connected component..... 13

2.12 การทำ Contours..... 14

2.13 การทำ Extracted marker edges..... 14

2.14 ความสัมพันธ์ระหว่าง Camera Coordinate และ Marker Coordinated Frame..... 15

2.15 แสดงความสัมพันธ์ระหว่าง Ideal Screen Coordinates และ Observe Screen Coordinates... 15

2.16 กระบวนการคำนวณค่าตำแหน่งเชิง 3 มิติ..... 16

2.17 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ..... 17

2.18 แสดง library organization..... 20

2.19 แสดงเมตริกซ์การเลื่อนตำแหน่ง..... 20

2.20 แสดงเมตริกซ์การย่อขยาย..... 21

2.21 แสดงเมตริกซ์ในการหมุนภาพรอบแกน x..... 21

2.22 แสดงเมตริกซ์ในการหมุนภาพรอบแกน y..... 22

2.23 แสดงเมตริกซ์ในการหมุนภาพรอบแกน z..... 22

2.24 แสดงการใช้งานของไลบรารี OpenCV..... 23

2.25 แสดงการแบ่งพื้นที่ของภาพ..... 24

สารบัญรูป(ต่อ)

รูปที่ หน้า

2.26 แสดงการตรวจหาขอบในพื้นที่ 25

2.27 แสดงการค้นหากลุ่มในพื้นที่ 25

2.28 แสดงการรวมกลุ่มของเส้น 26

2.29 แสดงการขยายเส้นขอบ 26

2.30 แสดงการจัดเก็บเส้นบริเวณมุม 26

2.31 แสดงการค้นหา Marker 26

2.32 แสดงตัวอย่างการทำ Threshold 27

2.33 แสดงแบบแม่สีหลักและแม่สีรอง 28

2.34 แสดงการทำงานของ irrEdit 29

3.1 แผนผังการทำงานของระบบความจริงเสริม 30

3.2 หลักการประมวลผลของระบบความจริงเสริม 32

3.3 แสดงลำดับการทำงานของโปรแกรม 33

3.4 แสดงการใช้โปรแกรม Paint สร้างวัตถุสัญลักษณ์คำว่า "AR" 34

3.5 แสดงการใช้โปรแกรม Paint สร้างวัตถุสัญลักษณ์คำว่า "SHOW" 34

3.6 แสดงผลตัวอย่างการปริ้นวัตถุสัญลักษณ์ 35

3.7 แสดงผลตัวอย่างการติดตั้งอุปกรณ์ 35

3.8 แสดงผลหน้าต่าง cmd.exe 35

3.9 แสดงตัวอย่างการเลือกขนาดกล้อง 36

3.10 แสดงผลภาพของวัตถุสัญลักษณ์ 36

3.11 แสดงผลหน้าต่าง cmd.exe เพื่อให้ใส่ filename 36

3.12 แสดงผลหน้าต่าง cmd.exe เพื่อให้ใส่ชื่อของวัตถุสัญลักษณ์ 37

3.13 แสดงผลหน้าต่าง cmd.exe เพื่อให้บันทึกชื่อและข้อมูล 37

3.14 แสดงตัวอย่างการสร้างวัตถุสามมิติด้านหน้า 38

3.15 แสดงตัวอย่างการสร้างวัตถุสามมิติด้านข้าง 38

3.16 แสดงผลการเปิดโปรมแกรม MilkShape 3D 1.8.4 38

3.17 แสดงผลการวาดวัตถุ 39

สารบัญรูป(ต่อ)

รูปที่ หน้า

3.18 แสดงผลการปรับแต่งวัตถุ..... 39

3.19 แสดงผลการวาดกรอบวัตถุ..... 39

3.20 แสดงผลการเลื่อนกรอบวัตถุ..... 40

3.21 แสดงผลการสร้างส่วนประกอบขึ้นมาตกแต่ง..... 40

3.22 แสดงผลการสร้างบนวัตถุรูปสี่เหลี่ยม..... 40

3.23 แสดงผลการ Set สีต่างๆของวัตถุก่อนใส่สี..... 41

3.24 แสดงผลการ Set สีต่างๆของวัตถุหลังใส่สี..... 41

3.25 แสดงผลการใส่ภาพ texture เข้าในวัตถุสี่เหลี่ยม..... 41

3.26 แสดงผลการ Export Model..... 42

3.27 แสดงผลการเปิดโปรแกรม GNU Image Manipulation Program (Gimp)..... 43

3.28 แสดงผลการเปิดไฟล์รูปภาพขึ้นมา..... 43

3.29 แสดงผลการคลิกเลือก file > Save As..... 43

3.30 แสดงผลการเลือกที่เก็บไฟล์..... 44

3.31 แสดงผลการเลือก Save as TGA..... 44

3.32 แสดงผลการได้ไฟล์ texture.tga..... 44

3.33 แสดงผลการ Import ไฟล์..... 45

3.34 แสดงผลการเลือกไฟล์โมเดลสามมิติ..... 45

3.35 แสดงผลโมเดลสามมิติ..... 45

3.36 แสดงผลการเลือกส่วนที่จะใส่ภาพพื้นผิว..... 46

3.37 แสดงผลหน้าต่าง Materials..... 46

3.38 แสดงผลการตั้งค่าหน้าต่าง Materials..... 47

3.39 แสดงผลการเลือกไฟล์ภาพพื้นผิวที่ชนิด True Vision Targa (TGA)..... 47

3.40 แสดงผลการเลือกหัวข้อ Textured..... 48

3.41 แสดงผลการใส่ Textured..... 48

3.42 แสดงผลการใส่ Textured ทั้งหมด..... 48

สารบัญรูป(ต่อ)

รูปที่ หน้า

3.43 แสดงผลการทำ Export โมเดลสามมิติ..... 49

3.44 แสดงผลทั้งหมดสองไฟล์คือ .obj และ .mtl..... 49

3.45 แสดงตัวอย่างภาพเคลื่อนไหว..... 50

3.46 แสดงตัวอย่างการสร้างแท่งสี่เหลี่ยมผืนผ้า..... 50

3.47 แสดงตัวอย่างการสร้างโมเดลสามมิติรูปการ์ตูน 51

3.48 แสดงตัวอย่างการสร้างโมเดลสามมิติรูปการ์ตูนทั้งหมด..... 51

3.49 แสดงตัวอย่างของโค้ดทำให้โมเดลสามมิติเคลื่อนไหว..... 52

3.50 แสดงตัวอย่างโมเดลสามมิติ..... 52

3.51 แสดงตัวอย่างโมเดลสามมิติ..... 53

3.52 แสดงตัวอย่างโมเดลสามมิติ..... 53

4.1 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านบน..... 54

4.2 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านข้าง..... 54

4.3 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านหน้า..... 55

4.4 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านบน..... 55

4.5 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านข้าง..... 55

4.6 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านหน้า..... 56

4.7 แสดงผลการติดตั้งอุปกรณ์..... 56

4.8 แสดงผลการทดลอง..... 61

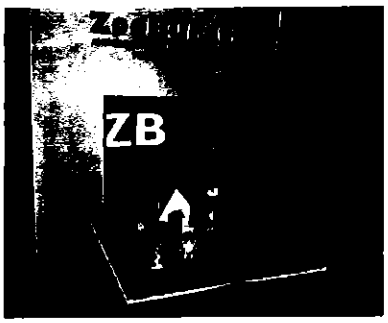
บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของโครงงาน

ปัจจุบันคอมพิวเตอร์ได้เข้ามามีบทบาทมากขึ้น ทำให้ยุคนี้กลายเป็นยุคแห่งข้อมูลข่าวสารหรือสังคมสารสนเทศ (Information Society) มีการใช้ข่าวสารในชีวิตประจำวันมากขึ้น การผลิตที่เคยใช้แรงงานและทรัพยากรธรรมชาติเป็นฐาน ก็ได้เปลี่ยนเป็นสังคมที่เศรษฐกิจ การผลิต การบริโภค มีการใช้ข้อมูลข่าวสารและเทคนิควิธีการเป็นฐานแทน การเปลี่ยนแปลงของเทคโนโลยี ทำให้มีสื่อใหม่ๆเกิด ขึ้นเป็นจำนวนมาก ซึ่งในแต่ละสื่อล้วนแข่งขันการพัฒนารูปแบบต่างๆ เพื่อให้ดึงดูดใจผู้บริโภคและผู้ ใช้บริการยุคใหม่ได้เป็นอย่างดี เนื่องจากแต่เดิมนั้นการผลิตสื่อโฆษณาสินค้าจะเป็นการโฆษณาผ่านทางโทรทัศน์ วิทยุ และสื่อสิ่งพิมพ์เป็นส่วนใหญ่ แต่เมื่อเทคโนโลยีมีการเปลี่ยนแปลงและเจริญเติบโตขึ้น ทำให้มีการนำคอมพิวเตอร์มาใช้ในการผลิตสื่อโฆษณาที่แปลกใหม่และทันสมัย ระบบความจริงเสริม (Augmented Reality) หรือเทคโนโลยีเสมือนจริงที่เป็นการนำเอาระบบความจริงเสมือนมาผนวกกับเทคโนโลยีภาพ เพื่อสร้างสิ่งที่เสมือนจริงให้กับผู้บริโภคได้เข้ามามีบทบาทกับการผลิตสื่อโฆษณามากขึ้น ซึ่งปัจจุบันนี้องค์กรธุรกิจต่างๆได้เริ่มนำเอาเทคโนโลยีระบบความจริงเสริมมาใช้ในการผลิตสื่อโฆษณาสินค้า ดังรูปที่ 1.1 – 1.2 แสดงตัวอย่างของการประยุกต์ใช้เทคโนโลยีระบบความจริงเสริมในการส่งเสริมการขาย

(ก) (ข)

รูปที่ 1.1 แสดงการประยุกต์ใช้เทคโนโลยีความจริงเสริมในการแสดงสินค้าตัวอย่างของ Lego (ก) ภาพ Lego รูปบ้าน (ข) ภาพ Lego รูปเรือ [1]



รูปที่ 1.2 แสดงการประยุกต์ใช้เทคโนโลยีความจริงเสริมในหนังสือ ZooBurst [2]

เมื่อเปรียบเทียบระหว่างการประยุกต์ใช้เทคโนโลยีความเป็นจริงเสมือน (virtual reality) และเทคโนโลยีความจริงเสริม (Augmented Reality) พบว่าแตกต่างกันในเรื่องของการปฏิสัมพันธ์ของมนุษย์ (Human Interaction) กับสิ่งแวดล้อมเสมือน (Virtual Environment) โดยการประยุกต์ใช้เทคโนโลยีความเป็นจริงเสมือนจะใช้อุปกรณ์ที่มีความซับซ้อนเพื่อระบุตำแหน่งของส่วนที่ปฏิสัมพันธ์กับมนุษย์ เช่น การใช้ถุงมือเพื่อระบุตำแหน่ง โดยใช้สัญญาณแม่เหล็กไฟฟ้า (Magnetic - Gloves) แต่ในการประยุกต์ใช้เทคโนโลยีความจริงเสริมจะสามารถพัฒนาส่วนของการปฏิสัมพันธ์กับสิ่งแวดล้อมได้ง่ายกว่า โดยใช้เพียงกล้องที่ติดกับอุปกรณ์ต่างๆ เช่น กล้องวิดีโอ เว็บแคม และวัตถุสัญลักษณ์ (Marker) ซึ่งทำให้ประหยัดต้นทุนในการพัฒนาระบบได้มากกว่าภายใต้สิ่งแวดล้อมเสมือนที่คล้ายกัน

ดังนั้นคณะผู้จัดทำโครงงานจึงมีความคิดที่จะพัฒนาเทคโนโลยีระบบความจริงเสริมในการผลิตสื่อโฆษณาเพื่อส่งเสริมการตลาดของสินค้าประเภทดีวีดี เพื่อสร้างความดึงดูดใจและประสบการณ์ใหม่ให้กับผู้บริโภค

1.2 วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาเทคโนโลยี และเทคนิควิธีในการพัฒนาระบบความจริงเสริม (Augmented - Reality)
2. เพื่อศึกษาเทคนิควิธีการที่เกี่ยวข้องเช่น คอมพิวเตอร์กราฟิกและทัศนศาสตร์คอมพิวเตอร์
3. เพื่อพัฒนาระบบส่งเสริมการขายด้วยการใช้เทคโนโลยีระบบความจริงเสริม (Augmented - Reality)

1.3 ขอบข่ายของโครงงาน

1. สามารถพัฒนาโปรแกรมเพื่อส่งเสริมการขายสินค้าประเภทดีวีดีโดยใช้ระบบความจริงเสริมได้
2. สามารถแสดงภาพ 3 มิติ ภาพเคลื่อนไหว และเสียงประกอบบนวัตถุสัญลักษณ์ (Marker) ได้
3. สามารถตอบโต้กับผู้ใช้ได้ โดยภาพที่แสดงจะเปลี่ยนไปตามการเลือกวัตถุสัญลักษณ์

1.4 ขั้นตอนการดำเนินงาน

1. ศึกษาความรู้ทั่วไปเกี่ยวกับระบบความจริงเสริม
2. ศึกษาการทำงานของไลบรารี ARToolKit เพื่อทำการคำนวณตำแหน่งที่จะนำภาพวัตถุสามมิติมาลงยังตำแหน่งที่ภาพสัญลักษณ์อยู่
3. ศึกษาการทำงานของไลบรารี OpenGL เพื่อสร้างภาพกราฟิก
4. ศึกษาการทำงานของไลบรารี OpenCV
5. ศึกษาการสร้างภาพเคลื่อนไหว ภาพสามมิติและการใส่เสียงประกอบ
6. สร้างโปรแกรมและทดสอบให้ได้ตามวัตถุประสงค์
7. จัดทำเอกสารและคู่มือการใช้

1.5 แผนผังการดำเนินโครงงาน

ตารางที่ 1.1 แผนการดำเนินงาน

กิจกรรม	ปี 2554							ปี 2555				
	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. ศึกษาความรู้ทั่วไปเกี่ยวกับความจริงเสริม (Augmented Reality)												
2. ศึกษาการทำงานของไลบรารี ARToolKit												
3. ศึกษาการทำงานของไลบรารี OpenGL												
4. ศึกษาการทำงานของไลบรารี OpenCV												
5. ศึกษาการสร้างภาพเคลื่อนไหวภาพสามมิติและเสียงประกอบ												

6. สร้างโปรแกรมและทดสอบให้ได้ตามวัตถุประสงค์											
7. จัดทำเอกสารและคู่มือการใช้											

1.6 ผลที่คาดว่าจะได้รับ

1. สามารถนำเทคโนโลยีระบบความจริงเสริมมาประยุกต์ใช้กับการโฆษณาสินค้าได้

2. สามารถนำเทคโนโลยีคอมพิวเตอร์กราฟิกและทัศนศาสตร์คอมพิวเตอร์มาใช้ในการพัฒนาระบบความจริงเสริมได้

3. สามารถนำเทคโนโลยีระบบความจริงเสริมมาช่วยในการพัฒนาระบบส่งเสริมการขายและสร้างความประทับใจให้กับผู้ใช้ได้

1.7 งบประมาณของโครงงาน

1. หนังสือประกอบการทำโครงงาน	1,000	บาท
2. ค่าเอกสาร	500	บาท
3. อื่นๆ	500	บาท
รวมเป็นเงินทั้งสิ้น	2,000	บาท (สองพันบาทถ้วน)

บทที่ 2
ทฤษฎีและหลักการที่เกี่ยวข้อง

บทนี้จะกล่าวถึงทฤษฎีและหลักการในการประยุกต์ทางด้านกราฟิกสามมิติ อย่างที่ได้กล่าวไปในบทที่ 1 แล้วว่าในโครงงานนี้เราเน้นศึกษาเทคโนโลยีที่เกี่ยวข้องกับเทคโนโลยีความจริงเสริม (Augmented Reality : AR) ในบทนี้เราจะกล่าวถึงองค์ความรู้ของเทคโนโลยีความจริงเสริม (AR) รวมทั้งเครื่องมือที่เกี่ยวข้อง อันได้แก่หลักการของเทคโนโลยีความจริงเสริม ซึ่งจะกล่าวถึงในหัวข้อ 2.1 ความรู้เบื้องต้นของ ARToolKit ในหัวข้อ 2.2 การสร้างภาพสามมิติโดยใช้ OpenGL ในหัวข้อ 2.3 ความรู้เบื้องต้น OpenCV ในหัวข้อ 2.4 Marker Detection ในหัวข้อ 2.5 Threshold ในหัวข้อ 2.6 รวมไปถึงสี RGB ในหัวข้อ 2.7 และความรู้เบื้องต้นเกี่ยวกับ irrKlang ในหัวข้อ 2.8

2.1 เทคโนโลยีความจริงเสริม (Augmented Reality) [1]

เทคโนโลยีความจริงเสริม (Augmented Reality: AR) เป็นประเภทหนึ่งของเทคโนโลยีความจริงเสริมที่มีการนำระบบความจริงเสริมมาผนวกเข้ากับเทคโนโลยีภาพเพื่อสร้างสิ่งที่เสมือนจริงให้กับผู้ใช้ และเป็นนวัตกรรมหรือเทคโนโลยีที่มีมาตั้งแต่ปี ค.ศ. 2004 เทคโนโลยีความจริงเสริมนั้นจัดเป็นแขนงหนึ่งของงานวิจัยทางด้านวิทยาการคอมพิวเตอร์ ว่าด้วยการเพิ่มภาพเสมือนของโมเดลสามมิติที่สร้างจากคอมพิวเตอร์ลงไปในภาพที่ถ่ายมาจากกล้องวิดีโอ กล้องเว็บแคม หรือกล้องในโทรศัพท์มือถือ แบบเฟรมต่อเฟรมด้วยเทคนิคทางด้านคอมพิวเตอร์กราฟิก

ปัจจุบันเทคโนโลยีความจริงเสริมได้ถูกนำมาประยุกต์ใช้กับธุรกิจต่างๆ ไม่ว่าจะเป็นด้านอุตสาหกรรม สถาปัตยกรรม การแพทย์ การบันเทิง การสื่อสาร การตลาดและธุรกิจ โดยใช้เทคโนโลยีความจริงเสริมมาผนวกเข้ากับเทคโนโลยีภาพผ่านทางซอฟต์แวร์และอุปกรณ์เชื่อมต่อต่างๆ และแสดงผลผ่านหน้าจอคอมพิวเตอร์หรือบนหน้าจอโทรศัพท์มือถือ ทำให้ผู้ใช้สามารถนำเทคโนโลยีความจริงเสริมมาใช้กับการทำงานแบบออนไลน์ที่สามารถโต้ตอบได้ทันทีระหว่างผู้ใช้กับสินค้าหรืออุปกรณ์ต่อเชื่อมแบบเสมือนจริงของโมเดลแบบสามมิติ ซึ่งภาพที่เห็นในจอภาพจะเป็นวัตถุ (คน, สัตว์, สิ่งของ) 3 มิติที่มีมุมมองถึง 360 องศา โดยผู้ใช้ไม่จำเป็นต้องไปสถานที่จริง

เมื่อเปรียบเทียบระหว่างการประยุกต์ใช้เทคโนโลยีความเป็นจริงเสมือน (Virtual Reality : VR) และเทคโนโลยีความจริงเสริม (AR) พบว่าแตกต่างกันในการใช้อุปกรณ์ระบุตำแหน่ง โดยการประยุกต์ใช้เทคโนโลยีความเป็นจริงเสมือนจะใช้อุปกรณ์ที่มีความซับซ้อนเพื่อระบุตำแหน่งของส่วนที่ปฏิสัมพันธ์กับมนุษย์ เช่น การใช้ถุงมือเพื่อระบุตำแหน่งโดยใช้สัญญาณแม่เหล็กไฟฟ้า แต่ในการประยุกต์ใช้เทคโนโลยีความจริงเสริมจะใช้เพียงกล้องที่ติดกับอุปกรณ์ต่างๆ เช่น กล้องวิดีโอ

กล้องเว็บแคม และวัตถุสัญลักษณ์ (Marker) ทำให้สามารถพัฒนาส่วนของการปฏิสัมพันธ์กับสิ่งแวดล้อมได้ง่ายกว่า และประหยัดต้นทุนในการพัฒนาระบบได้มากกว่าภายใต้สิ่งแวดล้อมเสมือน (Virtual Environments : VE) ที่คล้ายกัน

2.1.1 หลักการของความจริงเสริม (Augmented Reality Technique)

แนวคิดหลักของเทคโนโลยีความจริงเสริม คือการพัฒนาเทคโนโลยีที่ผสานเอาโลกแห่งความเป็นจริงและความเสมือนจริงเข้าด้วยกันผ่านทางซอฟต์แวร์และอุปกรณ์เชื่อมต่อต่างๆ เช่น กล้องวิดีโอ กล้องเว็บแคม หรืออุปกรณ์อื่นๆที่เกี่ยวข้อง ซึ่งภาพเสมือนจริงนั้นจะแสดงผลผ่านหน้าจอคอมพิวเตอร์ หน้าจอโทรศัพท์มือถือ บนเครื่องฉายภาพ หรือบนอุปกรณ์แสดงผลอื่นๆ โดยภาพเสมือนจริงที่ปรากฏขึ้นจะมีปฏิสัมพันธ์กับผู้ใช้ได้ทันที ทั้งในลักษณะที่เป็นภาพเคลื่อนไหว ภาพนิ่งสามมิติ หรืออาจจะเป็นสื่อที่มีเสียงประกอบ ขึ้นกับการออกแบบสื่อแต่ละรูปแบบว่าให้ออกมาแบบใด

กระบวนการหลักการทำงานของเทคโนโลยีความจริงเสริม (AR) โดยรวมแล้วจะประกอบด้วย 3 กระบวนการ ได้แก่

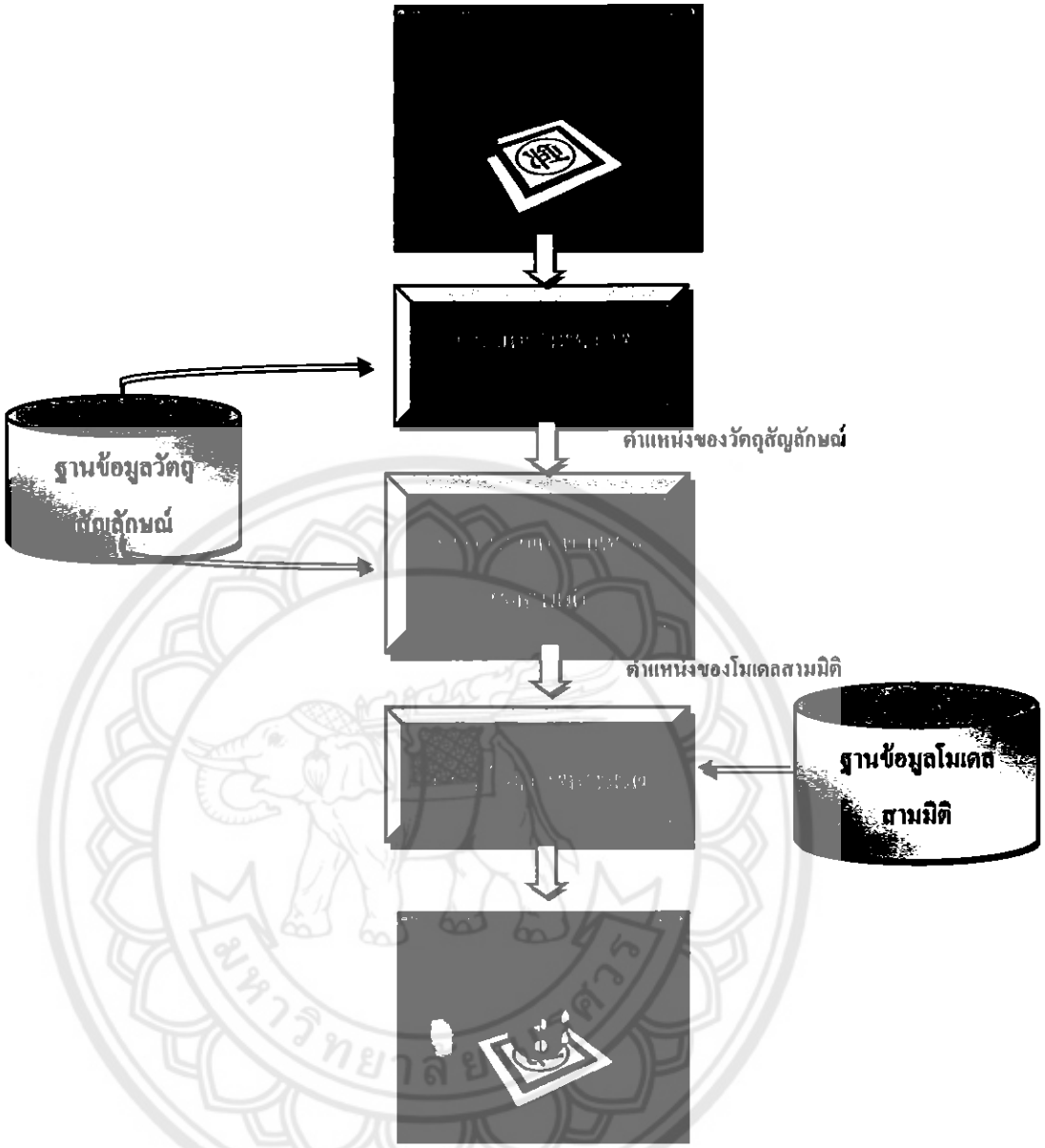
• การวิเคราะห์ภาพ (Image Analysis) โดยส่วนนี้เป็นขั้นตอนการวิเคราะห์ภาพโดยอาศัยวัตถุสัญลักษณ์เป็นหลักในการทำงาน ส่วนจากภาพที่ได้จากกล้องแล้วสืบค้นจากฐานข้อมูล ที่มีการเก็บข้อมูลขนาดและรูปแบบของวัตถุสัญลักษณ์ เพื่อนำมาวิเคราะห์รูปแบบของวัตถุสัญลักษณ์

• การคำนวณค่าตำแหน่งเชิง 3 มิติ (Pose Estimation) เป็นการวิเคราะห์ตำแหน่งและทิศทางของวัตถุสัญลักษณ์ เพื่อนำมาวิเคราะห์รูปแบบของวัตถุสัญลักษณ์แล้วทำการเปรียบเทียบกับกล้อง

• กระบวนการสร้างภาพสองมิติ จากโมเดลสามมิติ (3D Rendering) เป็นการเพิ่มข้อมูลเข้าไปในภาพ โดยใช้ค่าตำแหน่งเชิง 3 มิติ ที่คำนวณได้จนได้ภาพเสมือนจริง

หลักการของระบบความจริงเสริม ประกอบด้วย

1. ตัววัตถุสัญลักษณ์
2. กล้องวิดีโอ กล้องเว็บแคม กล้องโทรศัพท์มือถือ หรือตัวจับเซ็นเซอร์อื่นๆ
3. ส่วนแสดงผล ได้แก่ จอภาพ
4. ซอฟต์แวร์หรือส่วนประมวลผลเพื่อสร้างภาพหรือวัตถุแบบสามมิติ



รูปที่ 2.1 แสดงแผนภาพการทำงานของระบบความจริงเสริม [3]

2.1.2 เทคนิคความจริงเสริม

สามารถแบ่งชนิดของเทคนิคความจริงเสริมออกเป็น 4 ชนิดดังนี้

1. Optical See-Through AR

เป็นระบบความจริงเสริมสำหรับผู้ใช้ส่วนบุคคล โดยผู้ใช้นำอุปกรณ์ประเภทจอภาพสวมที่ศีรษะ (Head-mounted Display : HMD) ได้แก่หมวกเหล็กหรือหน้ากากมาใช้จำลองภาพและการได้ยิน

2. Projector Based AR

ใช้วัตถุโลกจริงเช่นเดียวกับการออกแบบพื้นผิวสำหรับสิ่งแวดล้อมเสมือน (Virtual Environments : VE)

3. Video See-Through AR

ใช้ HMD แบบทึบแสงในการแสดงผลซึ่งผสมผสานกับวิดีโอของ VE และมองจากกล้องถ่ายรูปบน HMD

4. Monitor-Based AR

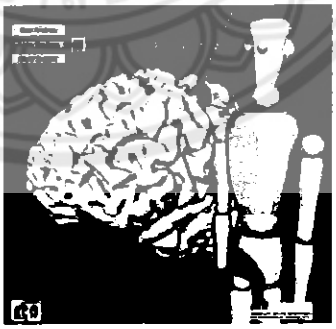
ใช้ผสมผสานกับวิดีโอสตรีมแต่การแสดงผลน่าติดตามมากกว่า โดยที่ Monitor-Based AR คือความเป็นไปได้ยากเล็กน้อยที่จะติดตั้งเพราะมันจำกัดเนื้อหาของ HMD

2.1.3 การประยุกต์ใช้เทคโนโลยีความจริงเสริม [4]

มีตัวอย่างการนำไปประยุกต์ใช้ดังนี้

1. เทคโนโลยีความจริงเสริมทางด้านการแพทย์ (Medical Application of Virtual Reality (VR) /Augmented Reality (AR)

ถึงแม้ว่าเทคโนโลยีความจริงเสริมจะได้มีการนำมาใช้ทางการแพทย์นับตั้งแต่การวางแผนผ่าตัดเสมือนจริงเมื่อต้นปี 1996 และการเรียบเรียงหลักการประยุกต์ใช้ภาพเสมือนจริงทางการแพทย์ ในปี 1997 อาทิเช่น การส่องกล้องดูลำไส้ใหญ่เชิงโต้ตอบในปี 1999 แต่การพัฒนาเทคโนโลยีความจริงเสริมนั้นโดยเพิ่มตัวต่อประสานระบบสัมผัสภาพ 3 มิติเพื่อเพิ่มความสมจริง (Realistic Sense) ในการรักษาโดยอาศัยการสร้างแรงปฏิกิริยาที่สะท้อนกลับ (Haptic Forces) ไปหาเครื่องมือที่ใช้ เมื่อเครื่องมือที่ใช้สัมผัสกับพื้นที่เป้าหมาย ในปี 2007 เป็นการอำนวยความสะดวกให้นักศึกษาแพทย์ได้ใช้เครื่องมือรักษา หรือผ่าตัดผู้ป่วยได้โดยไม่จำเป็นต้องทดลองกับคนไข้จริง ซึ่งเสี่ยงต่อความผิดพลาดที่มีอันตรายถึงชีวิตต่อผู้ป่วยได้



รูปที่ 2.2 แสดงส่วนของการจำลองสมอง [1]

ขณะนี้ได้มีการนำเทคโนโลยีความจริงเสริมจำลองการผ่าตัดได้สมจริงคือระบบ ARI*SER เพื่อการผ่าตัดเสมือนจริงซึ่งทางมหาวิทยาลัยแพทยศาสตร์ Ganz ได้แปลงให้เป็นระบบจำลองการผ่าตัดตับเสมือนจริง (Liver Surgery Planning System) เนื่องจากการผ่าตัดตับเป็นงานผ่าตัดที่ยากมาก ต้องมีความชำนาญเฉพาะทางจึงสำเร็จ

รูปที่ 2.3 แสดงการประยุกต์ใช้เทคโนโลยีเสมือนจริงจำลองการผ่าตัดผ่านระบบ ARI*SER [1]

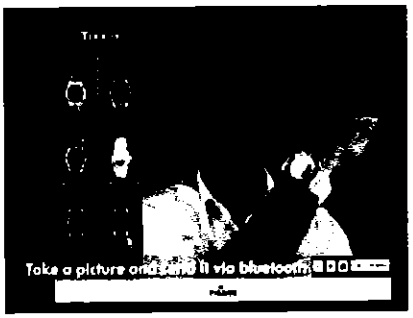
2. **เทคโนโลยีความจริงเสริมทางด้านธุรกิจ** เช่น ผลิตภัณฑ์ชนิดหนึ่งได้มีการนำเทคโนโลยีความจริงเสริมไปออกแบบเพื่อโปรโมทสินค้า

รูปที่ 2.4 แสดงการโปรโมทสินค้าของ adidas [5]

3. **เทคโนโลยีความจริงเสริมทางด้านท่องเที่ยว** เช่น การให้บริการบอกเส้นทางระยะทางในการเดินทาง ตำแหน่งของสถานที่ต่างๆ

รูปที่ 2.5 แสดงการให้บริการทางด้าน GPS [6]

4. **เทคโนโลยีความจริงเสริมทางด้านการสั่งสินค้าออนไลน์ เช่น** บริษัท Tissot ให้ลูกค้าสามารถลองสินค้าผ่านหน้าจอคอมพิวเตอร์ที่มีเว็บแคม โดยลูกค้าจะเลือกรหัสสินค้า หรือรุ่นที่ลูกค้าต้องการ ทำให้ลูกค้าได้ลองสินค้าเสมือนจริงผ่านเทคโนโลยี AR จนได้สินค้าที่ถูกใจก่อนสั่งซื้อสินค้า



รูปที่ 2.6 แสดงการประยุกต์ใช้เทคโนโลยีเสมือนจริงในการทดลองนาฬิกา [5]

2.1.4 ข้อดีและข้อเสียของการใช้เทคโนโลยีความจริงเสริม

2.1.4.1 ข้อดีจากการนำระบบเทคโนโลยีความจริงเสริมมาใช้

1. เป็นการสร้างประสบการณ์ที่แปลกใหม่ให้แก่ผู้บริโภค ถือเป็นอีกทางเลือกหนึ่งในการเข้าถึงกลุ่มลูกค้าที่เป็นคนรุ่นใหม่ที่ชอบและสนใจเทคโนโลยี

2. ผู้ใช้บริการสามารถค้นหาตำแหน่งและรายละเอียดของสินค้าที่ตนต้องการได้อย่างถูกต้องชัดเจน

3. บริษัทสามารถสร้างแคมเปญจน์ต่างๆ เพื่อสร้างความสนใจในตัวสินค้า จึงสามารถดึงดูดลูกค้าและเพิ่มยอดขายได้เพิ่มมากขึ้น

4. เพิ่มโอกาสของการค้าทางอินเทอร์เน็ท (E-commerce) เนื่องจากการที่ผู้ซื้อสามารถเห็นภาพจำลองของตนและสินค้าก่อนทำการสั่งซื้อสินค้า จึงเป็นการเปิดตลาดให้มีผู้ใช้บริการช่องทางนี้เพิ่มมากขึ้น ซึ่งทั้งนี้ยังส่งผลต่อไปยังผู้ที่ต้องการลงทุนทางธุรกิจ โดยช่วยลดค่าใช้จ่ายในการลงทุน เนื่องจากไม่จำเป็นต้องมีหน้าร้านเพื่อให้บริการ จึงไม่ต้องเสียค่าเช่าสถานที่ ฯลฯ

2.1.4.2 ข้อเสียจากการนำระบบเทคโนโลยีความจริงเสริมมาใช้

1. ไม่เหมาะกับกลุ่มคนที่มีความรู้ทางด้านเทค โนโลยีค่อนข้างน้อยมากนัก เนื่องจากว่าการนำเสนอด้วยรูปแบบนี้ ผู้ใช้จำเป็นต้องมีกล้องเว็บแคมและเครื่องพิมพ์ในกรณีที่เป็นการปริ้นตัววัตถุสัญลักษณ์ (Marker) ผ่านหน้าเว็บไซต์

2. เข้าถึงผู้บริโภคในกลุ่มที่จำกัด โดยผู้ใช้บริการต้องมีการใช้จ่ายที่ค่อนข้างดี เนื่องจากการใช้เทคโนโลยี AR ต้องอาศัยอุปกรณ์อิเลคทรอนิกส์หลายอย่าง

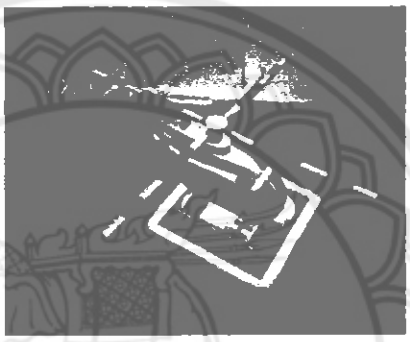
3. การที่มีกลุ่มผู้บริโภคจำกัด ทำให้อาจไม่คุ้มกับการลงทุนของบริษัทในการวางระบบเครือข่ายรวมทั้งการทำฐานข้อมูลต่างๆ เช่น การทำฐานข้อมูลของร้านค้า หรือสถานที่

4. ยังขาดการสนับสนุนจากภาครัฐบาล เนื่องจากในการใช้งานอย่างเช่น โทรศัพท์มือถือต้องใช้ระบบ 3G ซึ่งระบบดังกล่าวยังไม่ครอบคลุมพื้นที่ให้บริการส่วนใหญ่ของประเทศ ทำให้การใช้งานเทคโนโลยีความจริงเสริม (AR) ยังอยู่ในวงที่จำกัด

2.2 ความรู้เบื้องต้นเกี่ยวกับ ARToolKit

ARToolKit เป็นซอฟต์แวร์ไลบรารีสำหรับการประยุกต์ใช้สร้างระบบความจริงเสริม (Augmented Reality : AR) ซึ่งเป็นลักษณะการจำลองภาพเสมือนจริง (Visual Computer Graphic Image) ให้เกิดการซ้อนทับกันของภาพเสมือนในโลกแห่งความเป็นจริง

ความสามารถของไลบรารีของ ARToolKit ที่สำคัญคือการคำนวณจุดที่มองผ่านกล้องไปยังตำแหน่งภาพสัญลักษณ์เพื่อใช้ในการคำนวณตำแหน่งที่จะนำภาพวัตถุสามมิติมาลงยังตำแหน่งที่ภาพสัญลักษณ์อยู่ และคุณสมบัติที่สำคัญอีกอย่างคือวัตถุสามมิติเหล่านั้นจะสามารถเคลื่อนย้ายตำแหน่งได้ตามภาพสัญลักษณ์ได้

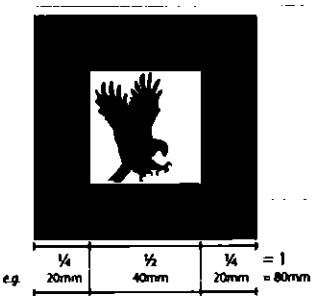


รูปที่ 2.7 แสดงตัวอย่างการสร้างภาพเหมือนจริง [7]

การทำงานของ ARToolKit เป็นเหมือนเครื่องมือที่ช่วยวาดวัตถุสามมิติประกอบฉาก โดยภาพที่ได้นั้นเกิดจากการรวมภาพที่ถ่ายจากกล้องเข้ากับโมเดลสามมิติที่ต้องการ โดยมีการจัดทิศทางของวัตถุให้สอดคล้องกับมุมของกล้องทำให้ดูเสมือนราวกับว่ามีวัตถุอยู่ในบริเวณนั้นจริงๆ ซึ่ง ARToolKit เป็น SDK ที่พัฒนาขึ้นโดยภาษา C และ C++ เพื่อใช้เป็นแก่นในการพัฒนาโปรแกรมประยุกต์ทางด้าน AR โดยอ้างอิงเนื้อหาจากหัวข้อ 2.1.1 (หลักการของความจริงเสริม) เครื่องมือ ARToolKit จะมีฟังก์ชันสำหรับส่วนของการวิเคราะห์ภาพ (Image Analysis) และ ส่วนคำนวณตำแหน่งเชิง 3 มิติ (Pose Estimation) ที่ค่อนข้างสมบูรณ์ อย่างไรก็ตาม ARToolKit ยังมีฟังก์ชันที่ใช้สำหรับการสร้างภาพจากโมเดล 3 มิติ (3D Rendering) ในระดับพื้นฐานซึ่งอิงอยู่กับ OpenGL SDK [8]

2.2.1 Image Analysis [9]

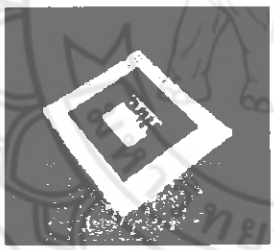
ในขั้นตอนนี้เป็นขั้นตอนการค้นหาวัตถุสัญลักษณ์จากภาพที่ได้จากกล้องวีดีโอ โดยก่อนอื่นเราจะต้องทำการดึงข้อมูลที่จำเป็นของวัตถุสัญลักษณ์หนึ่งๆที่จะใช้ มาสร้างเป็นฐานข้อมูลเก็บไว้ก่อน ซึ่งข้อมูลที่จำเป็นเหล่านี้ เช่น ขนาดของวัตถุสัญลักษณ์(ในหน่วย Cm.) และรูปแบบของวัตถุสัญลักษณ์เป็นต้น ซึ่งโดยทั่วไปแล้ว (ARToolKit) รูปแบบของ Marker จะต้องเป็นรูปสี่เหลี่ยมจัตุรัสที่มีขอบสีดำ พื้นหลังด้านในสีขาว และรูปแบบของวัตถุสัญลักษณ์เป็นสีดำ



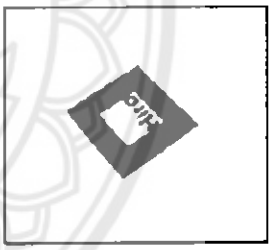
รูปที่ 2.8 แสดงรูปแบบที่เหมาะสมของวัตถุสัญลักษณ์ [9]

ขั้นตอนภายในของกระบวนการ Image Analysis เพื่อที่จะให้ได้มาซึ่งข้อมูลของ Marker ที่เราต้องการ โดยที่เมื่อระบบทำการค้นหา Marker หนึ่งๆในเฟรมภาพใดๆ จากกล้องวิดีโอ สามารถอธิบายได้ดังนี้ [9]

1. ทำการแปลงภาพที่ได้มาจากกล้องวิดีโอ (แสดงดังรูปที่ 2.9) ที่เป็นภาพสีนั้นให้กลายเป็น Binary Image (แสดงดังรูปที่ 2.10) โดยการกำหนดให้แต่ละพิกเซลในภาพ Binary Image มีค่าเป็น 0 หรือ 1 โดยที่จะมีค่าเป็น 0 เมื่อค่าระดับความสว่าง (Intensity) ของพิกเซลนั้นมีต่ำกว่าค่า Threshold มิฉะนั้นแล้วจะมีค่าเป็น 1

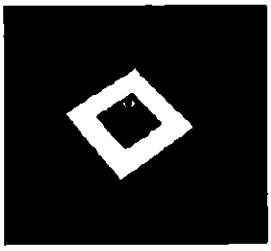


รูปที่ 2.9 การทำ Original image [9]



รูปที่ 2.10 การทำ Thresholded image [9]

2. ทำการหาพื้นที่ติดต่อกันโดยใช้เทคนิคทางการวิเคราะห์ภาพที่เรียกว่า Connected component labeling แสดงดังรูปที่ 2.11



รูปที่ 2.11 การทำ Connected component [9]

3. จากนั้นระบบจะทำการหาเส้นรอบรูป (Contours) ของพื้นที่ที่ได้มาจากผลลัพธ์ในขั้นตอนที่แล้ว แสดงดังรูปที่ 2.12

รูปที่ 2.12 การทำเส้นรอบรูป [9]

4. จากผลลัพธ์ของขั้นตอนที่แล้วระบบจะทำการประมาณหาค่าพารามิเตอร์ของสมการเส้นตรงที่แทนเส้นรอบรูปซึ่งเป็นรูปสี่เหลี่ยมทั้ง 4 เส้น หลังจากนั้นระบบจะหาจุดมุมทั้งสี่จุดของวัตถุสัญลักษณ์จากจุดตัดของเส้นตรงทั้งสี่ที่หาได้ (แสดงดังรูปที่ 2.13) ซึ่งจุดมุม 4 จุดนี้จะถูกนำไปใช้ในกระบวนการ Pose Estimation ซึ่งจะอธิบายต่อไป

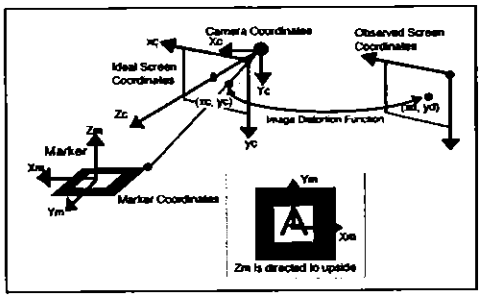
รูปที่ 2.13 การทำ Extracted marker edges [9]

2.2.2 Pose Estimation [9]

เป็นขั้นตอนของการคำนวณค่าตำแหน่งเชิง 3 มิติของวัตถุสัญลักษณ์ เมื่อเทียบกับกล้องวิดีโอ ค่านี้จะถูกแสดงในรูปเมตริกซ์ขนาด 4x4 (T_{CM}) ที่ระบุความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame ดังสมการที่ 1

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

ซึ่ง Camera Coordinated Frame ก็คือ Coordinated Frame ที่ใช้อ้างอิงตำแหน่งใดๆของกล้องวีดีโอ และ Marker Coordinated Frame ก็คือ Coordinated Frame ที่ใช้อ้างอิงตำแหน่งใดๆของวัตถุสัญลักษณ์ซึ่งสามารถแสดงดังรูปที่ 2.14



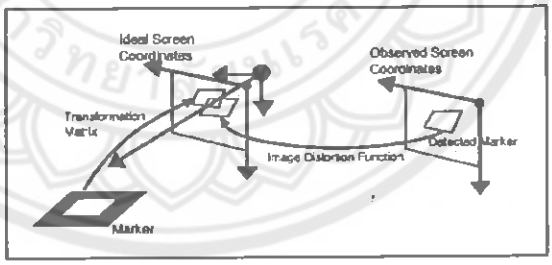
รูปที่ 2.14 ความสัมพันธ์ระหว่าง Camera Coordinated Frame และ Marker Coordinated Frame [9]

จากรูปที่ 2.14 ความสัมพันธ์ระหว่างจุดใดๆ (X_c Y_c Z_c) บนCamera Coordinated Frame กับจุดที่ตรงกัน (x_I y_I) ใน Ideal Screen Coordinated Frame เป็นไปตาม Perspective Projection ดังสมการที่ 2

$$\begin{bmatrix} hx_I \\ hy_I \\ h \end{bmatrix} = \begin{bmatrix} sf_x & 0 & x_c & 0 \\ 0 & sf_y & y_c & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = C \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2)$$

โดยที่ C ซึ่งเป็นเมตริกซ์ขนาด 3x4 ซึ่งประกอบไปด้วยค่า s, f_x, f_y, x_c, y_c โดยทั่วไปค่าเหล่านี้รวมกันเรียกว่า ค่าพารามิเตอร์ของกล้อง ซึ่งจะคำนวณได้มาจากขั้นตอน Camera Calibration

ส่วนค่าความสัมพันธ์ระหว่างจุดใดๆ บน Ideal Screen Coordinated Frame (x_I y_I) กับ Observe Screen Coordinated Frame (x_O y_O) ซึ่งเป็นจุดที่เราเห็นจริงๆในภาพ สามารถแสดงดังรูปที่ 2.15

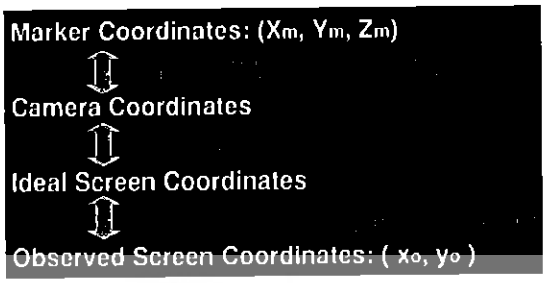


รูปที่ 2.15 แสดงความสัมพันธ์ระหว่าง Ideal Screen Coordinates และ Observe Screen Coordinates [9]

สามารถอธิบายได้ดังสมการที่ 3

$$d^2 = (x_1 - x_0)^2 + (y_1 - y_0)^2$$
$$p = \{1 - fd^2\}$$
$$x_0 = p(x_1 - x_0) + x_0, y_0 = p(y_1 - y_0) + y_0 \quad (3)$$

โดยที่ x_0 y_0 คือจุดศูนย์กลางของการ Distortion (Center Coordinates of Distortion) และ f คือ Distortion Factor ซึ่งค่าทั้ง 2 จะได้มาจากกระบวนการ Camera Calibration



รูปที่ 2.16 กระบวนการคำนวณค่าตำแหน่งเชิง 3 มิติ [9]

จากรูปที่ 2.16 จะแสดงกระบวนการที่จะได้มาของค่า T_{CM} เมื่อเรารู้ค่าตำแหน่งของ Marker ทั้ง 4 จุดบน Observed Screen Coordinates ในภาพที่ถ่ายจากกล้องวิดีโอ ซึ่งกล่าวโดยเฉพาะค่านี้สามารถหาได้จากการคำนวณหาคำตอบของฟังก์ชันค่าผิดพลาด ดังสมการที่ 4 ซึ่งโดยทั่วไปแล้วเราจะใช้เทคนิคทางด้านการหาค่าที่เหมาะสม ซึ่งเป็นกระบวนการแบบ Iterative

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

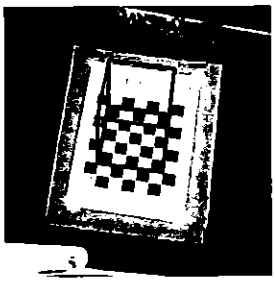
โดยที่ $\hat{x}_i$ $\hat{y}_i$ แสดงได้โดย

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

2.2.3 3D rendering [9]

ส่วนนี้เป็นส่วนสุดท้ายที่จะทำให้กระบวนการระบบความจริงเสริมครบถ้วนสมบูรณ์ ซึ่งก็คือ การเพิ่มข้อมูลที่เราต้องการซึ่งโดยทั่วไปแล้วจะเป็นโมเดล 3 มิติ ลงไปในภาพที่ได้จากกล้องวีดีโอ ณ ตำแหน่งของวัตถุสัญลักษณ์ ที่ตรวจพบจากขั้นตอน Image Analysis โดยใช้ค่าตำแหน่งเชิง 3 มิติที่คำนวณได้จากขั้นตอนการคำนวณหาตำแหน่งเชิง 3 มิติ

กล่าวโดยทั่วไปแล้ว 3D Rendering หมายถึง กระบวนการที่ทำการสร้างภาพ 2 มิติ จากโมเดล 3 มิติ ซึ่งโมเดล 3 มิตินี้ จะอธิบายวัตถุหรือสิ่งแวดล้อมหนึ่ง ๆ ที่ต้องการสร้างภาพนั้น



รูปที่ 2.17 แสดงการคำนวณหาโมเดล 3 มิติจากตำแหน่งวัตถุ [9]

2.3 การสร้างภาพสามมิติโดยใช้ OpenGL (3D Rendering with OpenGL) [10]

ในสมัยแรกๆ การเขียนโปรแกรมให้แสดงภาพกราฟิกบนจอภาพ ถ้าต้องการจะเขียนภาพเส้นตรง วงกลม วงรี การระบายสีโพลีกอน (รูปหลายเหลี่ยม) การย้ายวัตถุ หมุนวัตถุ ย่อหรือขยายวัตถุและการสะท้อนวัตถุจะต้องเขียนโปรแกรมติดต่อกับฮาร์ดแวร์โดยตรง ดังนั้นจึงได้มีการสร้างกราฟิกไลบรารีเพื่อเป็นคำสั่งในการเขียนโปรแกรมด้านกราฟิก เช่น การวาดเส้นตรง การวาดวงกลมและการวาดวงรี ต่อมาบริษัท Silicon Graphics ได้สร้าง OpenGL (Open Graphics Library) เป็นระบบเปิดและไมโครซอฟต์ก็ได้สร้าง DirectX ซึ่งเป็นกราฟิกไลบรารีเช่นเดียวกันแต่มีมาตรฐานที่ต่างกัน ปัจจุบันการ์ดแสดงผลแบบสามมิติรุ่นใหม่ๆ ได้นำชุดคำสั่งของ OpenGL ไปเก็บไว้ในตัวการ์ดแสดงผลเพื่อให้การทำงานมีความเร็วเพิ่มขึ้น

ดังนั้น OpenGL คือซอฟต์แวร์ที่ใช้ในการเชื่อมต่อกับฮาร์ดแวร์ที่แสดงผลทางด้านกราฟิก ซึ่งประกอบด้วยคำสั่งประมาณ 120 คำสั่ง ที่สามารถใช้กำหนดคุณลักษณะและควบคุมการทำงานของแอปพลิเคชัน 3 มิติ ซึ่งผู้พัฒนาโปรแกรมสามารถใช้ไลบรารี OpenGL ได้โดยไม่มีค่าลิขสิทธิ์ ทำให้มีการนำไลบรารีของ OpenGL ไปใช้งานอย่างแพร่หลายในงานกราฟิก

2.3.1 ภาษาที่สามารถใช้กับ OpenGL

- C/C++ (VC++, Borland C++, C++ Builder, C compiler on UNIX)
- Delphi
- Visual Basic
- Java
- Pert
- Python
- Fortran

2.3.2 หลักการทำงานของ OpenGL

OpenGL เป็น Graphic API (Application Programming Interface) ที่เป็นทั้งแบบ 2 มิติและแบบ 3 มิติโดยการทำงานจะแยกเป็นสองส่วนอันดับแรกคือ ทางฮาร์ดแวร์ที่ต้องรองรับการทำงานของ OpenGL และฮาร์ดแวร์จะทำงานได้ต้องทำงานผ่าน OpenGL ICD (Installable Client Driver) ซึ่งจะมีมากับไดร์เวอร์ของการ์ดแสดงผลทุกตัวซึ่งบางตัวก็จะเป็นแค่ MCD (Mini Client Driver) อีกส่วนก็คือ WGL (OpenGL Extension for Microsoft Window) หรือ Environment ของ Window ที่จะเป็นตัวกลางระหว่าง OpenGL และฮาร์ดแวร์เพื่อให้ระบบสามารถใช้ OpenGL ได้อย่างถูกต้อง ดังนั้นOpenGL คือ API แบบ 2 มิติและ 3 มิติที่สามารถย้ายโอนระบบได้อย่างอิสระซึ่งจะหมายความว่าระบบ API ตัวนี้จะไม่อิงกับระบบปฏิบัติการ Windows, Mac หรือระบบใดๆ

OpenGL ถูกออกแบบมาโดยไม่ยึดติดกับระบบใดๆ สามารถทำงานได้บนทุกๆ แพล็ตฟอร์ม (Portability) เพื่อที่จะบรรลุเป้าหมายนี้ OpenGL จะไม่มีคำสั่งที่จัดการกับระบบปฏิบัติการเลย อีกทั้งยังไม่มีคำสั่งเพื่อรับอินพุทจากผู้ใช้อีกด้วย หน้าที่ทั้งสองอย่างนี้จึงตกอยู่กับผู้เขียนโปรแกรม อย่างไรก็ตามยังมีซอฟต์แวร์เพิ่มเติมที่ช่วยจัดการงานทั้งสองนี้หากพัฒนาโปรแกรมบนระบบปฏิบัติการแบบ Windows (ดู GLUT: OpenGL Utility Toolkit) นอกจากนี้ OpenGL ยังไม่มีคำสั่งระดับสูงที่จะใช้วาดวัตถุสามมิติแบบซับซ้อน อย่างเช่นรถยนต์ อวัยวะ หรือโมเลกุล สิ่งที่ OpenGL เตรียมไว้ให้สำหรับสร้างรูปจำลองสามมิติคือรูปทรงเรขาคณิตอย่างง่ายได้แก่ จุด เส้น และรูปหลายเหลี่ยมซึ่งผู้ใช้งานจะต้องนำรูปทรงเหล่านี้มาประกอบกัน

คณะกรรมการที่ร่วมกันกำหนดมาตรฐานของ OpenGL หรือ OpenGL Architecture Review Board (ARB) ประกอบด้วยบริษัท 3DLabs, Apple, ATI, Dell, Evans & Sutherland, HP, IBM, Intel, Matrox, NVIDIA, SGI, และ Sun Microsystems ปัจจุบัน OpenGL ได้ถูกพัฒนามาจนถึงเวอร์ชัน 1.5

2.3.3 เหตุผลที่นิยมใช้ OpenGL ในระบบกราฟิก

ในการออกแบบและการทำงานในระบบกราฟิกมีเครื่องมือที่ช่วยงานหลายเครื่องมือ แต่ละเครื่องมีคุณสมบัติ, ฟังก์ชัน หรือเครื่องมือที่แตกต่างกันไป เครื่องมือเหล่านั้นอาจจะมีจุดเด่น-จุดด้อยที่แตกต่างกันไป แต่ปัจจุบันเครื่องมือที่นิยมนำมาใช้ในการสร้างระบบกราฟิกก็คือ OpenGL ซึ่งเหตุผลสำคัญที่นิยมนำ OpenGL มาใช้งานมีดังนี้

- มีประสิทธิภาพสูงในการเร่งความเร็วแอปพลิเคชัน 3 มิติ และเกมต่างๆ ในปัจจุบัน
- สามารถใช้ข้อมูลจำนวนมากและสร้างเอ็ฟเฟ็กต์ 3 มิติ ในแบบเรียลไทม์ได้อย่างมีประสิทธิภาพ
- เพิ่มการสนับสนุนอุปกรณ์ใหม่ๆ ลงไปใน OpenGL ที่ทำได้ง่ายและรวดเร็ว

- ทำงานได้บนหลายแพล็ตฟอร์ม ทำให้การย้ายโปรแกรมประยุกต์ระหว่างแต่ละแพล็ตฟอร์มนั้นทำได้ง่าย และประหยัด
- มีเสถียรภาพในการทำงานสูงสามารถใช้งานกับเครื่องเวิร์คสเตชันแบบ High End 3D และ ซุปเปอร์คอมพิวเตอร์ได้

2.3.4 ไลบรารีที่เกี่ยวข้องกับ OpenGL

ฟังก์ชัน OpenGL จะขึ้นต้นด้วย gl และถูกเก็บไว้ที่ library ที่เรียกว่า GL นอกจากไลบรารีหลักของ OpenGL แล้ว ยังมีไลบรารีที่เกี่ยวข้องอีกเป็นจำนวนมากเพื่อจัดการกับงานเฉพาะอย่างได้อย่างมีประสิทธิภาพ ไลบรารีที่เกี่ยวข้องกับ OpenGL และเป็นไลบรารีเฉพาะมีดังนี้

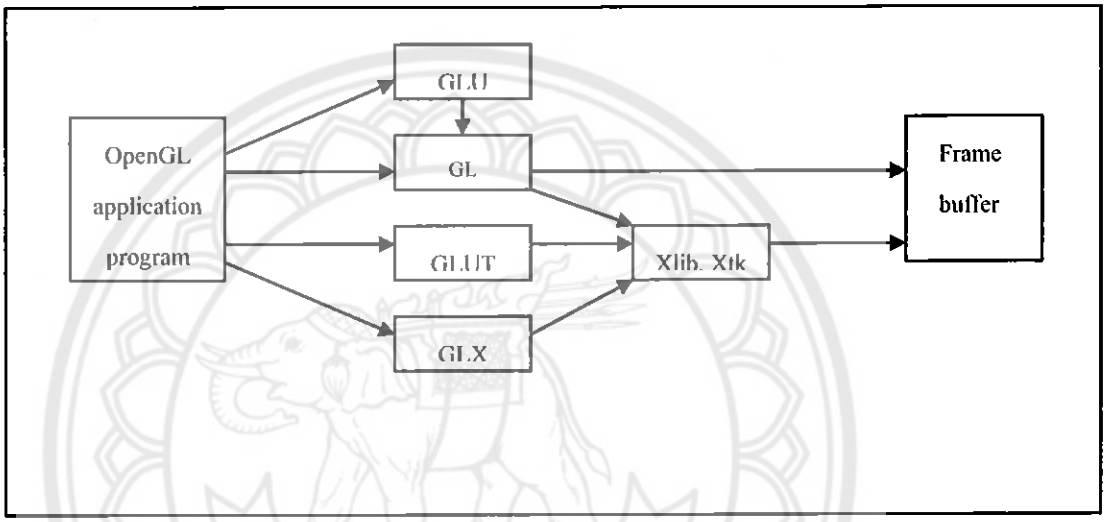
2.3.4.1 OpenGL Utilities (GLU)

OpenGL Utilities (GLU) เป็นไลบรารีที่ประกอบด้วยรูทีนมากมายในการจัดการมุมมองเพื่อแสดงรูปพื้นฐานและออปเจ็กต์ที่ซับซ้อนที่ประกอบขึ้นจากเส้นและรูปหลายเหลี่ยม, แสดงรูปลูกบาศก์, การเรนเดอร์พื้นผิว (surface-rendering) และงานที่ซับซ้อน ตัวอย่างการสร้างวัตถุพื้นฐาน เช่น ฟังก์ชันในการสร้างทรงกลม ผู้ใช้ไม่จำเป็นต้องทราบว่าการสร้างทรงกลมสร้างอย่างไรแต่สามารถเรียกใช้ฟังก์ชันได้เลย GLU นี้เป็นไลบรารีที่มีใน OpenGL อยู่แล้ว ซึ่งฟังก์ชันของ GLU จะขึ้นต้นด้วยคำว่า glu เสมอ การเรียกใช้จะต้อง include ไฟล์ header ที่ชื่อ glu.h ในตอนต้นของโค้ดโปรแกรม นอกจากนี้ยังมีชุดเครื่องมือสำหรับการพัฒนาเชิงวัตถุ (object-oriented) ที่อ้างอิงกับ OpenGL อีก เรียกว่า Open Inventor ซึ่งมีรูทีนและรูปทรงออปเจ็กต์สำหรับแอปพลิเคชัน 3 มิติจำนวนมาก ชุดเครื่องมือนี้เขียนด้วย C++

2.3.4.2 OpenGL Utility Toolkit (GLUT)

OpenGL Utility Toolkit (GLUT) คือ ไลบรารีของระบบกราฟิกที่ช่วยในการติดต่อกับการแสดงผลทางจอภาพ ทั้งนี้เนื่องจากในการใช้งาน OpenGL เพื่อใช้งานด้านกราฟิก สิ่งแรกที่มีความจำเป็นต้องทำก็คือการกำหนดวินโดว์สำหรับการแสดงผล (display window) บนจอภาพ ซึ่งวินโดว์ดังกล่าวนี้ก็คือพื้นที่สี่เหลี่ยมผืนผ้าของจอภาพที่ใช้แสดงกราฟิก โดยไม่สามารถสร้างวินโดว์นี้ได้โดยตรงจากฟังก์ชันของ OpenGL เนื่องจากไลบรารีนี้ประกอบเพียงฟังก์ชันทางด้านกราฟิกที่ไม่ขึ้นกับอุปกรณ์ใดๆ นอกจากนี้การจัดการเกี่ยวกับวินโดว์ก็ขึ้นอยู่กับคอมพิวเตอร์ที่ใช้งานอยู่อีกด้วย อย่างไรก็ตามยังคงมีไลบรารีจำนวนหนึ่งที่สนับสนุนฟังก์ชันของ OpenGL สำหรับเครื่องที่หลากหลายกันไป ซึ่ง GLUT เป็นชุดเครื่องมือที่มีไลบรารีของฟังก์ชันสำหรับการใช้งานกับระบบวินโดว์ของจอภาพทั่วๆไป (ไม่ขึ้นกับระบบใดระบบหนึ่ง) GLUT นี้เขียนขึ้นโดย Mark Kilgard ทำให้การติดต่อระหว่างแอปพลิเคชันของ OpenGL กับระบบปฏิบัติการ

แบบ Windows หรือเรียกว่าเป็น API (Application Programming Interface) ทำได้ง่ายขึ้น เนื่องจากการเขียนโปรแกรมโดยใช้ OpenGL มีความซับซ้อนน้อยลง เหมาะสำหรับการพัฒนาโปรแกรมขนาดเล็กถึงขนาดกลาง คำสั่งของ GLUT จะขึ้นต้นด้วยคำว่า glut เสมอ การเรียกใช้จะต้อง include ไฟล์ header ที่ชื่อ glut.h ในตอนต้นของโค้ดโปรแกรมเช่นกัน ถ้าคุณใช้ glut.h แล้ว ก็ไม่จำเป็นต้องใช้ gl.h และ glu.h อีก ไลบรารีเหล่านี้ยังประกอบด้วยเมธอดสำหรับการสร้างและเรนเดอร์ quadric curves และพื้นผิว



รูปที่ 2.18 แสดง library organization [10]

2.3.5 การแปลงภาพ 3 มิติ [10]

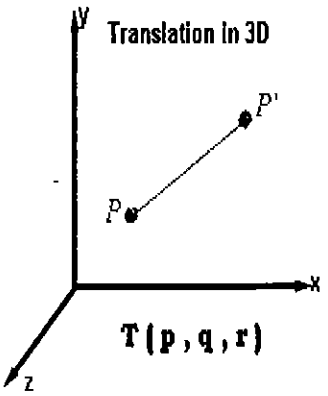
ในระบบ 3 มิติการแปลงเบื้องต้นประกอบไปด้วยการแปลงพื้นฐาน 3 แบบคือ

2.3.5.1 การเลื่อนตำแหน่ง (Translation)

การเลื่อนตำแหน่งหรือก็คือการย้ายจุดใดๆของวัตถุไปยังจุดใหม่ โดยเวกเตอร์ในการย้ายคือ (p , q , r) เมตริกซ์ในการเลื่อนตำแหน่งดังแสดงในภาพที่ 2.19

$$Tr = \begin{bmatrix} 1 & 0 & 0 & p \\ 0 & 1 & 0 & q \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

where Tr is the translation matrix



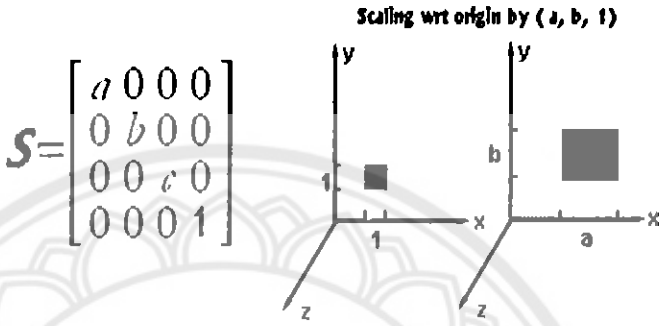
รูปที่ 2.19 แสดงเมตริกซ์การเลื่อนตำแหน่ง

[10]

ฟังก์ชันที่ใช้ในการเลื่อนตำแหน่งคือ glTranslatef (Glfloat x, Glfloat y, Glfloat z) สำหรับการย้ายเวกเตอร์ V(x , y , z)

2.3.5.2 การย่อ/ขยายภาพ (Scaling)

การย่อขยายภาพเป็นการเพิ่มหรือลดขนาดของวัตถุโดยมีเวกเตอร์ (a ,b ,c) โดยมีเมตริกซ์ในการย่อขยายดังแสดงในภาพที่ 2.20



รูปที่ 2.20 แสดงเมตริกซ์การย่อขยาย [10]

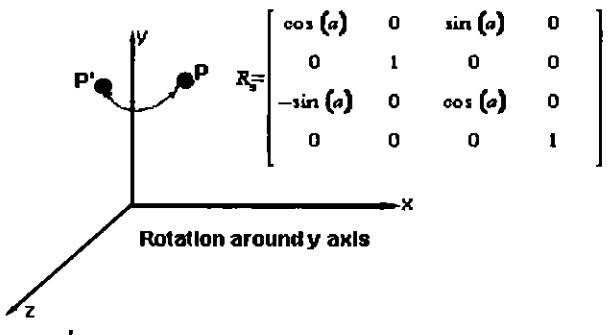
การย่อขยายภาพมีฟังก์ชันของ OpenGL คือ glScalef (a ,b ,c) ในการย่อขยาย (a ,b ,c) ตัวอย่าง หากต้องการขยายภาพเป็น 2 เท่าของภาพปกติเขียนแทนโดย glScalef (2 ,2 ,2)

2.3.5.3 การหมุนภาพ (Rotation)

ใน OpenGL สามารถทำการหมุนภาพรอบแกนได้ง่าย โดยมีฟังก์ชันคือ Rotatef (Glfloat angle, Glfloat x, Glfloat y, Glfloat z) เมื่อ (x , y , z) คือแกนที่จะให้วัตถุหมุน เช่น x(1 , 0 , 0) , y(0 , 1 , 0) , z (0 , 0 , 1) และกำหนดทิศทางในการหมุนทวนเข็มนาฬิการอบแกนที่ทำการหมุน

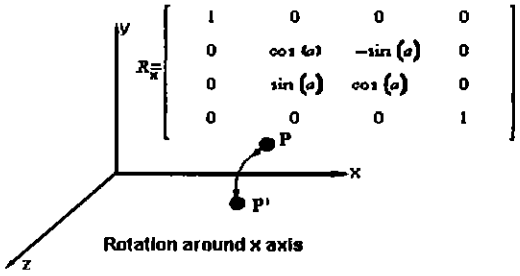
ตัวอย่าง หากต้องการหมุนภาพไปในทิศทาง 90 องศา รอบแกน y โดยมีทิศทางในการหมุนตามเข็มนาฬิกากับแกนที่ทำการหมุน สามารถแทนฟังก์ชันได้เป็น glRotatef (-90 , 0 , 1 , 0)

คุณลักษณะที่สำคัญของการหมุนภาพคือ มุมในการหมุนภาพและแกนหมุน

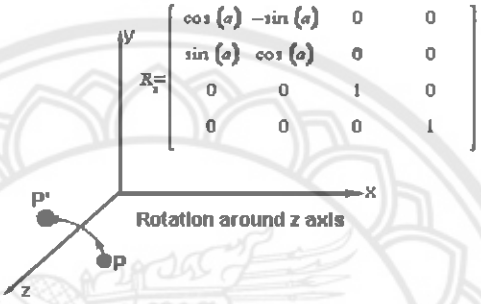


$$R_y = \begin{bmatrix} \cos(a) & 0 & \sin(a) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(a) & 0 & \cos(a) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

รูปที่ 2.21 แสดงเมตริกซ์ในการหมุนภาพรอบแกน x



รูปที่ 2.22 แสดงเมตริกซ์ในการหมุนภาพรอบแกน y



รูปที่ 2.23 แสดงเมตริกซ์ในการหมุนภาพรอบแกน z

2.3.6 ฟังก์ชันภาพเคลื่อนไหวของ OpenGL

ไลบรารีของ GLUT ได้เตรียมฟังก์ชันสำหรับการแสดงผลบนจอภาพระบบแรสเตอร์ในระบบฮาร์ดแวร์บางระบบอาจจะไม่มีฟีเจอร์ของการแสดงภาพเคลื่อนไหวบนจอคอมพิวเตอร์ เช่น บัพเฟอร์คู่ (Double buffer) แต่ถ้าฮาร์ดแวร์สนับสนุนฟีเจอร์นี้ เราสามารถกระตุ้นให้ทำงานได้ด้วยคำสั่ง GLUT ดังนี้

glutInitDisplayMode (GLUT_DOUBLE);

คำสั่งนี้ทำให้ใช้บัพเฟอร์ได้ 2 บัพเฟอร์ที่เรียกว่า บัพเฟอร์หน้า (front buffer) และบัพเฟอร์หลัง (back buffer) ทำให้เราเพื่อรีเฟรชจอแสดงผล ในขณะที่บัพเฟอร์หนึ่งกำลังทำหน้าที่รีเฟรชจอแสดงผลนั้น เฟรมถัดไปของภาพเคลื่อนไหวจะสร้างได้บนอีกบัพเฟอร์หนึ่ง เรากำหนดการเปลี่ยนหน้าที่ระหว่างบัพเฟอร์ทั้งสองด้วยคำสั่ง **glutSwapBuffers ();**

การพิจารณาว่าระบบบัพเฟอร์คู่สามารถทำงานในระบบได้หรือไม่ สามารถตรวจสอบได้ด้วยคำสั่ง **glGetBooleanv (GL_DOUBLEBUFFE, status) ;**

ค่า GL_TRUE ถูกส่งค่ากลับมาในพารามิเตอร์แบบอาร์เรย์ status ถ้าบัพเฟอร์ทั้งสองมีในระบบ ถ้าไม่มีจะส่งค่า GL_FALSE กลับมา

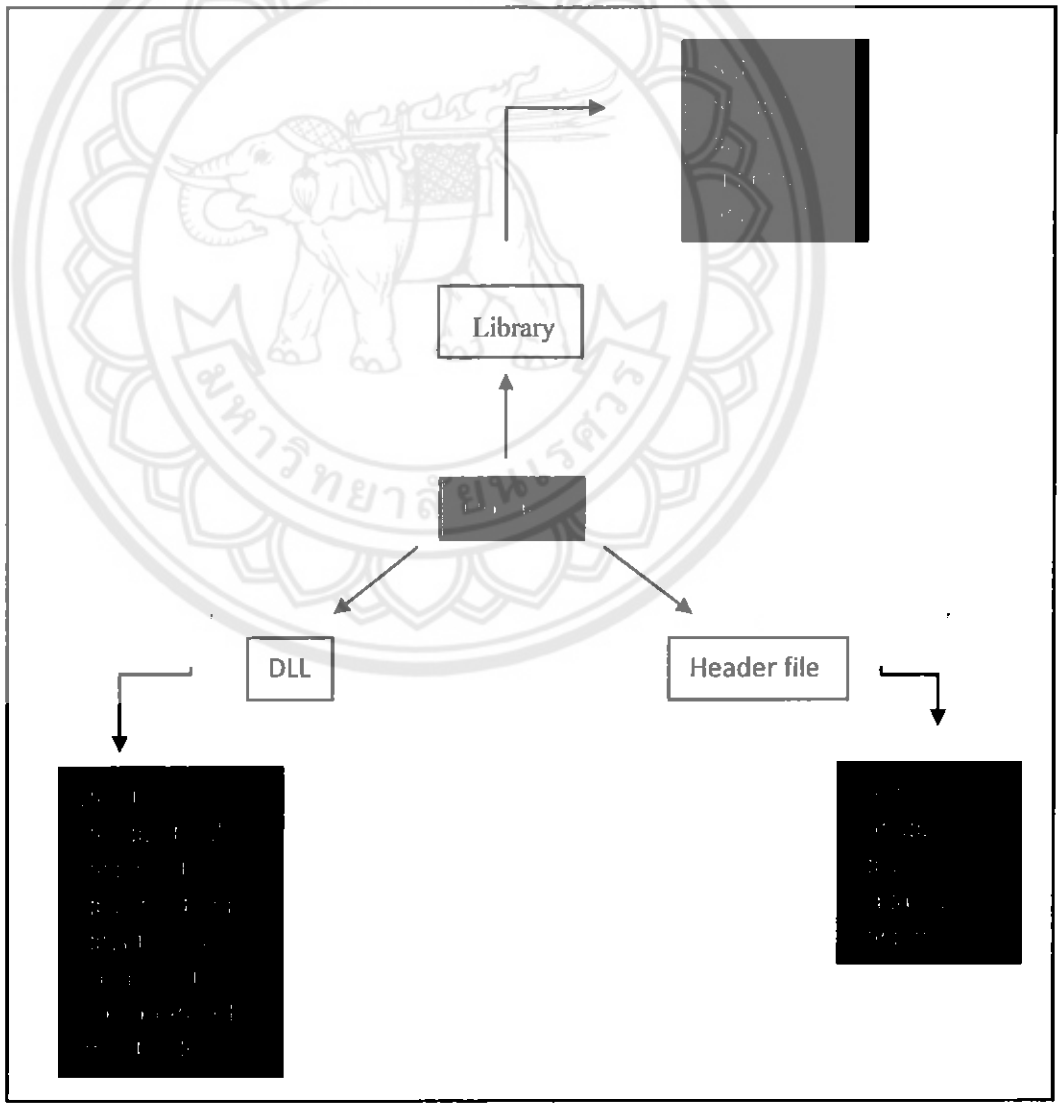
ในการแสดงภาพเคลื่อนไหวอย่างต่อเนื่อง เรายังสามารถใช้คำสั่ง

glutIdleFunc (animationFcn) ;

2.4 ความรู้เบื้องต้นเกี่ยวกับ OpenCV

OpenCV เป็นไลบรารีสำหรับใช้ในการประมวลผลภาพ ซึ่งเป็นไลบรารีโอเพนซอร์ส สามารถดาวน์โหลดใช้งานได้ฟรี ไลบรารีต่างๆของ OpenCV ได้พัฒนาขึ้นด้วย บริษัทอินเทล จุดเด่นในด้านความสามารถของไลบรารี

OpenCV คือสามารถประมวลผลภาพดิจิตอลได้ทั้งภาพนิ่งและภาพเคลื่อนไหวเช่น ภาพจากกล้องวิดีโอ หรือไฟล์วิดีโอ เป็นต้น โดยไม่ยึดติดทางด้านฮาร์ดแวร์ทำให้ OpenCV สามารถพัฒนาโปรแกรมได้หลากหลายภาษา รวมถึงมีฟังก์ชันสำเร็จรูปสำหรับจัดการข้อมูลภาพ และการประมวลผลภาพพื้นฐานเช่น การหาขอบภาพ การกรองข้อมูลภาพ โดยฟังก์ชันต่างๆของ OpenCV จะสามารถเรียกใช้งานได้จะต้องมีการเรียก ไฟล์ส่วนหัว และลิงค์ไลบรารีต่างๆ รวมถึง DLL (Dynamic Link Library) โดยมีความสัมพันธ์ดังรูปที่ 2.24



รูปที่ 2.24 แสดงการใช้งานของไลบรารี OpenCV [11]

2.4.1 การทำ Camera calibration ด้วย OpenCV

การทดสอบกล้องหรือวิดีโอนั้นเป็นสิ่งสำคัญสำหรับการประมวลผลภาพใดๆ ที่จะทำให้มีความถูกต้องอย่างมากเมื่อเทียบกับภาพจริง ข้อมูลเบื้องต้นสำหรับการทำ calibration นั้นกล่าวรวมถึงการสร้างภาพเรขาคณิตและวิธีการที่แยกพารามิเตอร์ของกล้องที่เกี่ยวข้องจากลำดับของภาพ โดยการพิจารณาจากวัตถุที่รู้มิติแน่นอนจากตำแหน่งในสามมิติและใน pixel coordinate

ฟังก์ชันที่ใช้สำหรับ Video I/O

- **CvCapture** เป็นฟังก์ชันโครงสร้างที่ไม่ได้มีการอินเทอร์เฟซทั่วไปและจะใช้เฉพาะพารามิเตอร์สำหรับฟังก์ชันจับภาพวิดีโอ ชนิดที่ใช้คือ CvCapture CvCapture;
- **cvCreateFileCapture** เป็นฟังก์ชันค่าเริ่มต้นสำหรับการจับภาพวิดีโอจากไฟล์ซึ่งจะเริ่มทำงานได้จากโครงสร้างของ CvCapture สำหรับการอ่านสตรีมไฟล์วิดีโอที่ระบุ ชนิดที่ใช้คือ CvCapture* cvCreateFileCapture(const char* filename);
- **cvQueryFrame** เป็นฟังก์ชันคว้ากรอบจากไฟล์กล้องหรือวิดีโอ ซึ่งฟังก์ชันนี้จะเป็นเพียงแค่การรวมกันของ cvGrabFrame และ cvRetrieveFramre ในหนึ่งภาพ ดังนั้นผลลัพธ์ที่ออกมาไม่ควรมีการปรับเปลี่ยนโดยผู้ใช้ ชนิดที่ใช้คือ IplImage* cvQueryFrame(CvCapture* capture);
- **cvGetCaptureProperty** เป็นฟังก์ชันที่ดึงคุณสมบัติที่ระบุจากไฟล์กล้องหรือวิดีโอ ชนิดที่ใช้คือ double cvGetCaptureProperty(CvCapture* capture, int property_id);

2.5 Marker Detection [12]

ขั้นตอนการทำ Marker Detection

2.5.1 ขั้นตอนที่ 1 การแบ่งพื้นที่ของภาพ

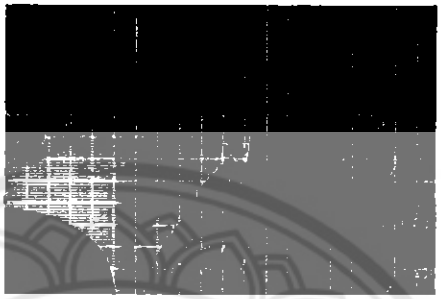
ก่อนที่ภาพจะถูกแบ่งออกเป็นพื้นที่ขนาดเล็ก 40x40 พิกเซลและแต่ละพื้นที่จะแบ่งออกเป็นแนวนอนและแนวตั้ง และขั้นตอนถัดไปจะดำเนินการนอกเหนือพื้นที่ซึ่งช่วยเพิ่มประสิทธิภาพการทำงาน

รูปที่ 2.25 แสดงการแบ่งพื้นที่ของภาพ [12]

ห้องสมุดคณะวิศวกรรมศาสตร์ มหาวิทยาลัยนเรศวร

2.5.2 ขั้นตอนที่ 2 การตรวจหาขอบในพื้นที่

อนุพันธ์เกาส์ (Gaussian derivative) **ใช้ในการตรวจจับขอบสีดำ/สีขาวในภาพ อนุพันธ์เกาส์ขึ้นอยู่กับ** convolving **ภาพที่มีตัวกรองขนาดเล็กแยกกันไม่ออก การวางแนวขอบนั้นได้ถูกบันทึกค่าไว้ซึ่งในภาพข้างต้นเป็นจุดสีเขียวบริเวณ** edgels orientated **ในแนวนอนและจุดสีฟ้าบริเวณ** edgels orientated **ในแนวตั้ง**



รูปที่ 2.26 แสดงการตรวจหาขอบในพื้นที่ [12]

2.5.3 ขั้นตอนที่ 3 การค้นหากลุ่มในพื้นที่

หลังจากการตรวจสอบจากขอบภาพ อัลกอริทึม RANSAC จะ**ใช้ในการสร้างส่วน**ของแต่ละพื้นที่ เพราะเรารู้ว่าทิศทางของการหาขอบภาพจากขั้นตอนที่ 2 เรา**สามารถวางทิศทางในภาพข้างต้นให้เป็นส่วนที่สร้างขึ้นจะแสดงเป็นลูกศรสีแดงและสีเขียว**

รูปที่ 2.27 แสดงการค้นหากลุ่มในพื้นที่ [12]

2.5.4 ขั้นตอนที่ 4 การรวมกลุ่มของเส้น

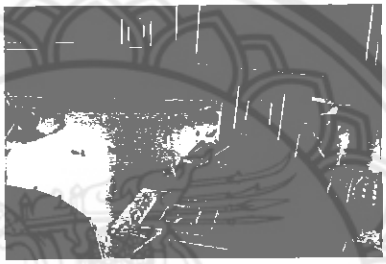
การรวมสองส่วนจะถูกรวมกันถ้า

1. **ทิศทางเข้ากันได้**
2. **ส่วนที่ใกล้กันถูกปิด**
3. **หากพิกเซลตามเส้นระหว่างสองทิศทางบนขอบโดยใช้เกณฑ์เดียวกับการตรวจสอบขอบภาพในขั้นตอนที่** 2

รูปที่ 2.28 แสดงการรวมกลุ่มของเส้น [12]

2.5.5 ขั้นตอนที่ 5 การขยายเส้นขอบ

บริเวณเส้นเส้นที่ขยายจะปรากฏเป็นลูกศรสีเหลืองในภาพข้างต้น ในขั้นตอนที่ 6 และ 7 เป็นเฉพาะเส้นเท่านั้นที่ต้องมีมุมอย่างน้อยหนึ่งมุมที่ใช้



รูปที่ 2.29 แสดงการขยายเส้นขอบ [12]

2.5.6 ขั้นตอนที่ 6 การจัดเก็บเส้นบริเวณมุม

รูปที่ 2.30 แสดงการจัดเก็บเส้นบริเวณมุม [12]

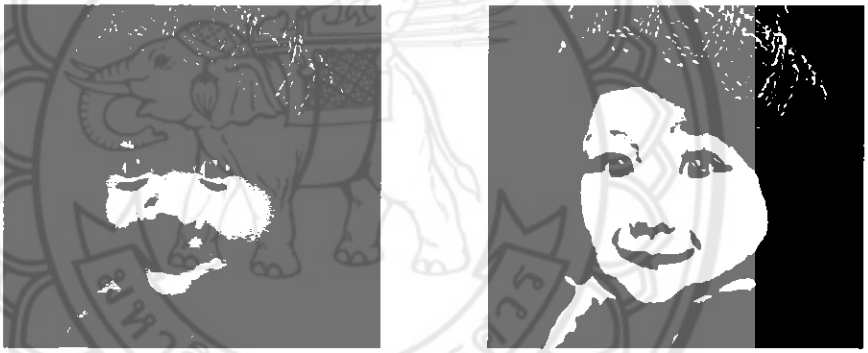
2.5.7 ขั้นตอนที่ 7 การค้นหา Marker

สุดท้ายเราพยายามที่จะหา marker ทั้งหมดภายในภาพ

รูปที่ 2.31 แสดงการค้นหา Marker [12]

2.6 Threshold

การแยกบริเวณของภาพโดยการใช้ค่า Threshold ซึ่งค่า Threshold เป็นค่าที่เป็นจำนวนเต็มที่มีค่าระหว่าง 0 ถึง 255 เช่นเดียวกับค่าความเข้มแสงของพิกเซลที่อยู่ในภาพ Gray scale สำหรับการแยกบริเวณโดยการใช้ค่า Threshold นั้น จะเป็นการแปลงภาพ Gray scale ให้เปลี่ยนเป็นภาพที่มีเพียงสองระดับ (binary image) โดยได้มีการกำหนดค่า Threshold ขึ้นมา 2 ค่า คือ high threshold (T1) และ low threshold (T2) โดยพิกเซลที่มีค่ามากกว่า T1 จะถูกปรับเป็น 1 (เป็นพิกเซลที่เป็นขอบ) แต่ถ้าน้อยกว่า T2 จะถูกปรับเป็น 0 ส่วนค่าที่อยู่ระหว่างค่า threshold ทั้งสอง การปรับเป็นค่า 0 หรือ 1 นั้นขึ้นอยู่กับพิกเซลที่อยู่รอบข้าง หากพบว่าพิกเซลที่อยู่รอบข้างของพิกเซลที่เป็นขอบ (ค่า >T1) มีค่ามากกว่า T2 แล้ว จะปรับค่าพิกเซลดังกล่าวให้มีค่าเป็น 1 และถือเป็นสมาชิกหนึ่งในภาพขอบด้วยเช่นกัน



รูปที่ 2.32 แสดงตัวอย่างการทำ Threshold [13]

2.7 สี RGB

ระบบสี RGB เป็นระบบสีของแสง ซึ่งเกิดจากการหักเหของแสงผ่านแท่งแก้วปริซึม ซึ่งจะเกิดแถบสีที่เรียกว่า สีรุ้ง (Spectrum) ซึ่งแยกสีตามที่สายตามองเห็นได้ 7 สี คือ ม่วง คราม น้ำเงิน เขียว เหลือง แสด แดง

ซึ่งเป็นพลังงานที่อยู่ในรูปของรังสี ที่มีช่วงคลื่นที่สายตาสามารถมองเห็นได้ แสงสีม่วงมีความถี่คลื่นสูงที่สุด คลื่นแสงที่มีความถี่สูงกว่าแสงสีม่วง เรียกว่า อุลตราไวโอเลต (Ultra violet) และคลื่นแสงสีแดง มีความถี่คลื่นต่ำที่สุด คลื่นแสงที่ต่ำกว่าแสงสีแดงเรียกว่า อินฟราเรด (Infrared) คลื่นแสงที่มีความถี่สูงกว่าสีม่วง และต่ำกว่าสีแดงนั้น สายตาของมนุษย์ไม่สามารถรับได้

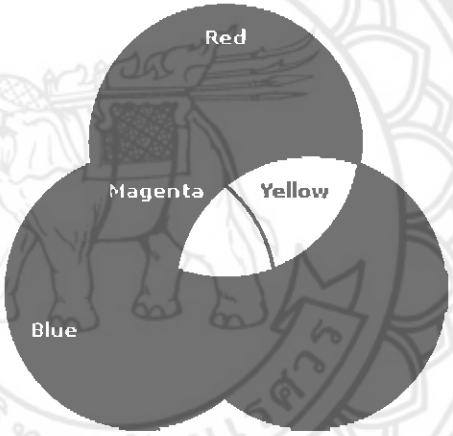
แสงสีทั้งหมดเกิดจาก แสงสี 3 สี คือ สีแดง (Red) สีน้ำเงิน (Blue) และสีเขียว (Green) ทั้งสามสีถือเป็นแม่สีของแสง แม่สีของแสงมีด้วยกัน 3 สี คือ สีแดง, สีเขียว, สีน้ำเงิน และแต่ละแม่สีเมื่อรวมกันก็จะได้สีดังนี้

สีแดง+สีเขียว ได้ สีเหลือง Yellow

สีเขียว+น้ำเงิน ได้ สีฟ้า Cyan

สีแดง+สีน้ำเงิน ได้ สีแดงอมชมพู่ Magenta

เมื่อนำแม่สีของแสงทั้ง 3 มาผสมกัน ในปริมาณแสงสว่างเท่ากันก็จะได้เป็นแสงที่สีขาว จากคุณสมบัติของแสงนี้เราได้นำมาใช้ประโยชน์ทั่วไป ในการฉายภาพยนตร์ การบันทึกภาพวิดีโอ ภาพโทรทัศน์ การสร้างภาพเพื่อการนำเสนอทางจอคอมพิวเตอร์ และการจัดแสงสีในการแสดง เป็นต้น



รูปที่ 2.33 แสดงแบบแม่สีหลักและแม่สีรอง [14]

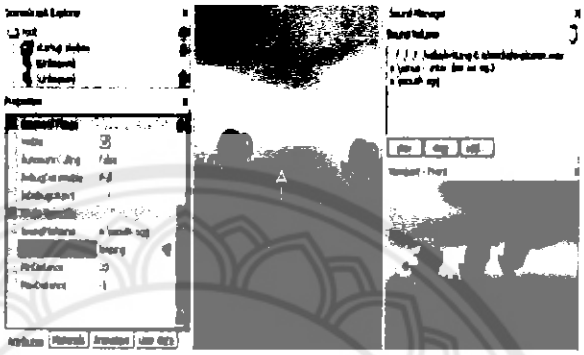
2.8 ความรู้เบื้องต้นเกี่ยวกับ irrKlang

คุณสมบัติสำคัญของ irrKlang [15]

irrKlang เป็น API (Application Programming Interface) ระดับที่มีประสิทธิภาพสูงสำหรับการเล่นกับเสียงในแอพพลิเคชัน 3 มิติ และ 2 มิติ เช่นในเกม ภาพทางวิทยาศาสตร์และการใช้งานมัลติมีเดียนั้น irrKlang จะให้ใช้ฟรีสำหรับที่ไม่ใช่ในเชิงพาณิชย์ นอกจากนี้ยังมีรุ่นสูงของ irrKlang ที่มีชื่อว่า irrKlang Pro ซึ่งสามารถซื้อเพื่ออนุญาตใช้งานในเชิงพาณิชย์ได้

2.8.1 3D Sound

irrKlang นั้นมีตัวสนับสนุนสำหรับเสียง 3 มิติบนแพลตฟอร์มทั้งหมดและไดร์เวอร์เสียง มันถูกออกแบบมาเพื่อให้ใช้ในการเล่นเกม 3 มิติ ดังนั้นจึงมีประสิทธิภาพมากและไม่ได้ใช้จำนวนของ CPU มาก irrKlang จะแสดงการทำงานภายใน irrEdit สำหรับการแก้ไขในโปรแกรม ดังรูปที่ 2.34



รูปที่ 2.34 แสดงการทำงานของ irrEdit [15]

irrKlang สามารถเล่นเสียงบัพเฟอร์สั้นเช่น เสียงยิงปืน รวมทั้งเสียงสตรีมขนาดใหญ่ เช่น เสียงวิทยุ เสียงmp3 ซึ่งใน 3 มิตินั้นการแปลงเสียงแบบโมโนไม่จำเป็นเพราะ irrKlang สามารถทำเช่นนี้ได้ทันที

2.8.2 Compilers ที่สนับสนุนและภาษาที่ใช้สำหรับ irrKlang

- Microsoft Visual C/C++ 6.0 2010
- GCC 3-4
- All .NET languages (C#, VisualBasic.NET, Delphi.NET, IronPhyton etc.)

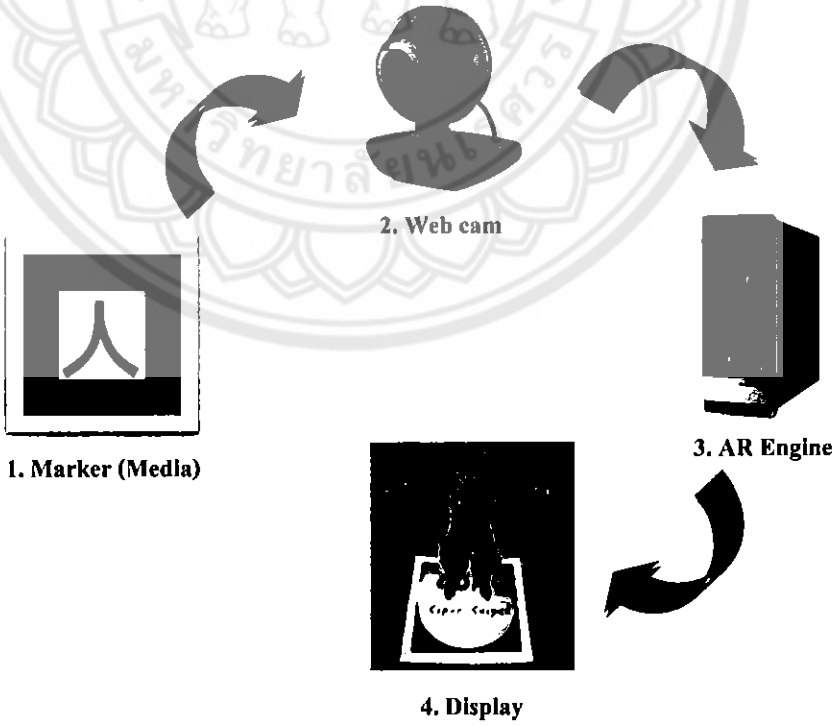
บทที่ 3

วิธีการดำเนินโครงงาน

ในบทนี้จะเป็นการกล่าวถึงการออกแบบการทำงานของการพัฒนาระบบความจริงเสริม (Augmented Reality) เพื่อส่งเสริมการตลาดของดีวีดี โดยรูปแบบการทำงานของระบบความจริงเสริมจะใช้หลักการของการตรวจจับวัตถุสัญลักษณ์ (Marker) โดยผ่านกล้องเว็บแคม แล้วแสดงภาพออกมา โดยภาพที่ปรากฏจะมีลักษณะเป็นสามมิติ หรือภาพเคลื่อนไหว (Animation) ซึ่งมีมุมมองถึง 360 องศา

3.1 ออกแบบการทำงาน

การทำงานของเทคโนโลยีระบบความจริงเสริม จะประกอบด้วยวัตถุสัญลักษณ์ (Marker) กล้องเว็บแคม โปรแกรมประมวลผล และจอแสดงผล ดังแสดงในรูปที่ 3.1



รูปที่ 3.1 แผนผังการทำงานของระบบความจริงเสริม (Augmented Reality)

อธิบายแผนผังการทำงานของระบบความจริงเสริม (Augmented Reality)

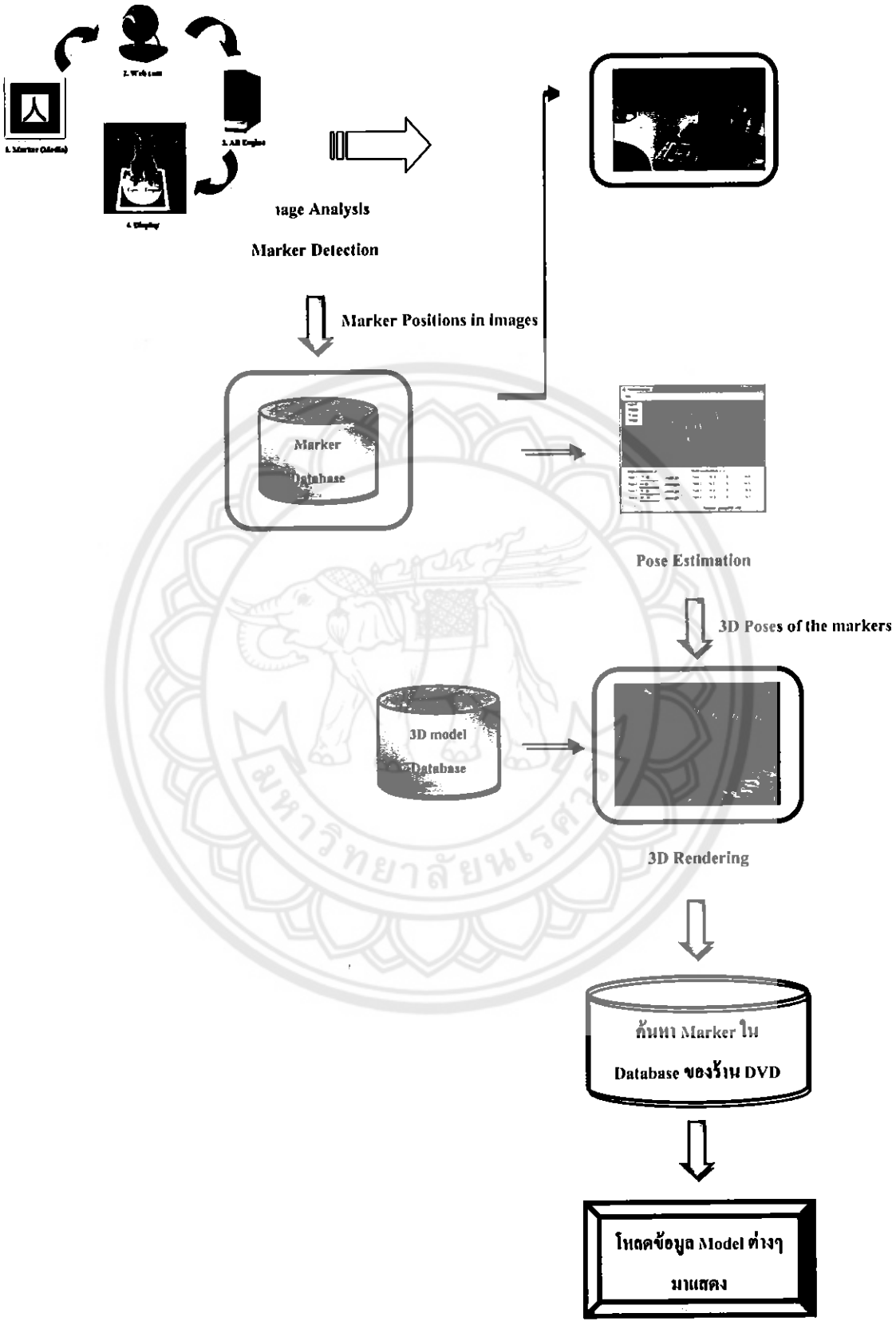
1. วัตถุสัญลักษณ์ (Marker) ใช้ในการกำหนดตำแหน่งของวัตถุ
2. กล้องเว็บแคม (Web cam) ใช้มองตำแหน่งจากวัตถุสัญลักษณ์ แล้วส่งเข้ามาที่โปรแกรมประมวลผล (AR Engine)
3. โปรแกรมประมวลผล (AR Engine) ใช้ในการอ่านข้อมูลจากกล้องเข้ามาประมวลผล
4. จอแสดงผล (Display) แสดงข้อมูลที่โปรแกรมประมวลผลส่งมาในรูปแบบของภาพต่างๆ เช่น ภาพ 3 มิติ หรือภาพเคลื่อนไหว พร้อมเสียงประกอบ

หลักการทำงานของเทคโนโลยีระบบความจริงเสริมนั้น คอมพิวเตอร์จะรับภาพจริงจากกล้องและคำนวณหาตำแหน่งของภาพโดยใช้กระดาษที่เป็นวัตถุสัญลักษณ์ และทำการสร้างภาพ 3 มิติ หรือภาพเคลื่อนไหวขึ้นมาซ้อนทับ โดยภาพที่ได้จะสามารถมองเห็นผ่านจอมอนิเตอร์ ซึ่งลักษณะของภาพจะมีมุมมองถึง 360 องศา

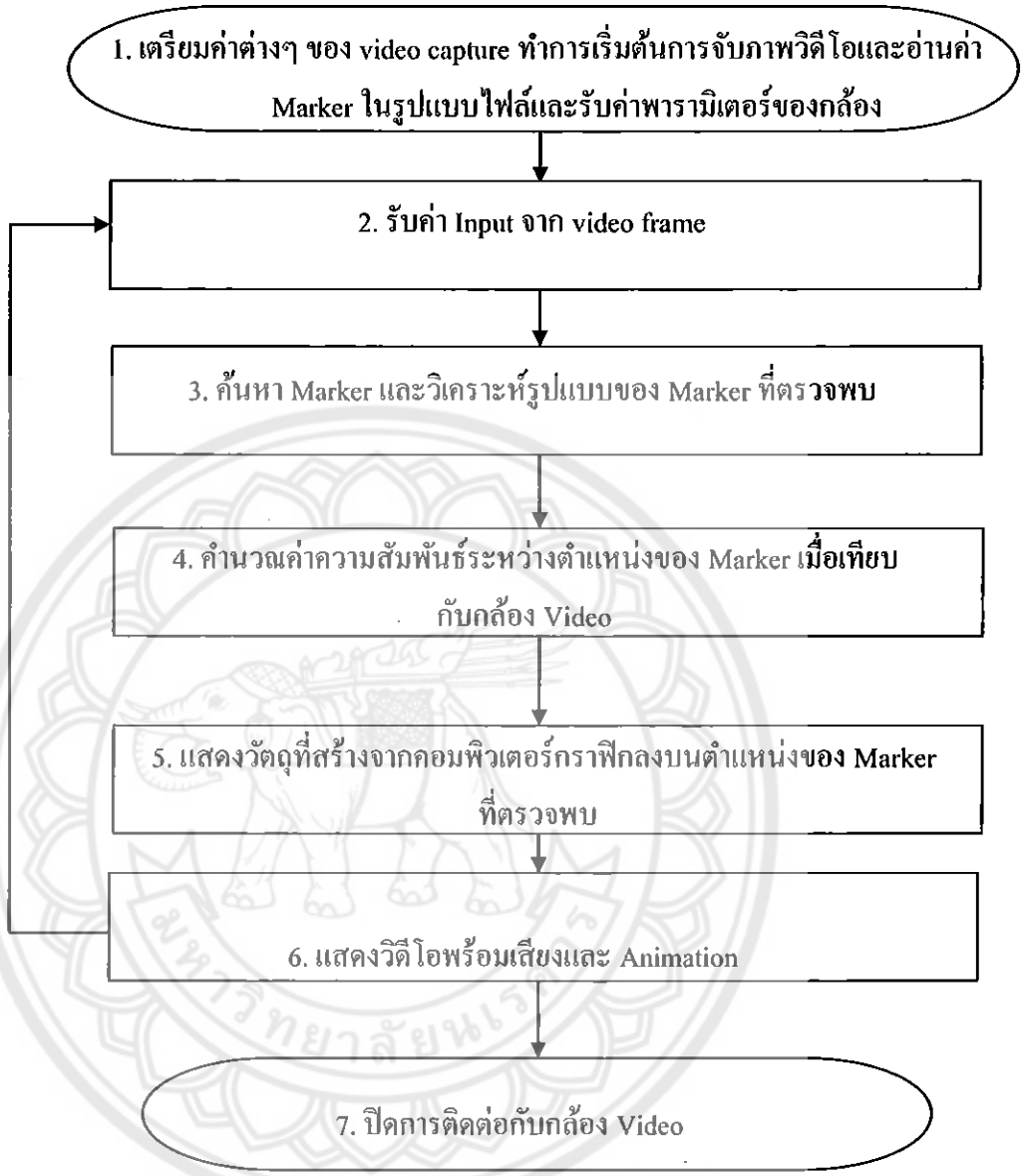
ในส่วนการประมวลผล (AR Engine) นั้น จะต้องใช้หลักการของการวิเคราะห์ภาพ (Image - Analysis) การคำนวณค่าตำแหน่งเชิง 3 มิติ (Pose Estimation) และกระบวนการสร้างภาพ 2 มิติ จากโมเดล 3 มิติ (3D Rendering)

ระบบความจริงเสริมสามารถแบ่งประเภทของการวิเคราะห์ภาพ (Image Analysis) ออกเป็น 2 ประเภทได้แก่ การวิเคราะห์ภาพโดยอาศัยวัตถุสัญลักษณ์ เป็นหลักในการทำงาน (Marker based AR) และการวิเคราะห์ภาพโดยใช้ลักษณะต่างๆที่อยู่ในภาพมาวิเคราะห์ (Marker-less based AR) ในโครงงานนี้เราจะเน้นศึกษาเฉพาะ การวิเคราะห์ภาพโดยอาศัยวัตถุสัญลักษณ์เป็นหลัก (Marker based AR)

- การวิเคราะห์ภาพ (Image Analysis) เป็นขั้นตอนการค้นหาวัตถุสัญลักษณ์จากภาพที่ได้จากกล้องแล้วนำมาสืบค้นจากฐานข้อมูล (Marker Database) ที่มีการเก็บข้อมูลขนาดและรูปแบบของวัตถุสัญลักษณ์
- การคำนวณค่าตำแหน่งเชิง 3 มิติ (Pose Estimation) เป็นการวิเคราะห์ตำแหน่งและทิศทางของวัตถุสัญลักษณ์ เพื่อนำมาวิเคราะห์รูปแบบของวัตถุสัญลักษณ์เทียบกับกล้อง
- กระบวนการสร้างภาพ 2 มิติ จากโมเดล 3 มิติ (3D Rendering) เป็นการเพิ่มข้อมูลเข้าไปในภาพ โดยใช้ค่าตำแหน่งเชิง 3 มิติ ที่คำนวณได้จนได้เป็นภาพเสมือนจริง ดังแสดงในรูปที่ 3.2



รูปที่ 3.2 หลักการประมวลผลของระบบความจริงเสริม (Augmented Reality)



รูปที่ 3.3 แสดงลำดับการทำงานของโปรแกรม

3.2 ออกแบบการสร้างวัตถุสัญลักษณ์

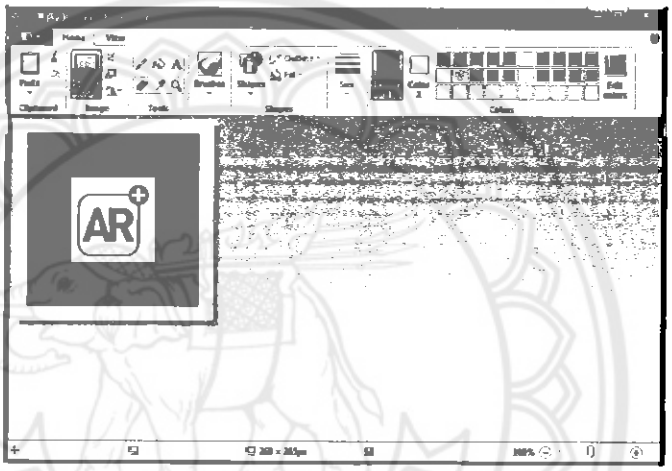
ในขั้นตอนนี้เป็นการค้นหาวัตถุสัญลักษณ์ (Marker) จากภาพที่ได้จากกล้องวีดีโอ หรือกล้องเว็บแคม โดยก่อนอื่นต้องทำการดึงข้อมูลที่จำเป็นของ Marker ที่จะใช้ มาสร้างเป็นฐานข้อมูลเก็บไว้ก่อน ซึ่งข้อมูลที่จำเป็น เช่น ขนาดของ Marker (cm.) และรูปแบบของ Marker เป็นต้น โดยทั่วไปแล้ว ARToolKit รูปแบบของ Marker จะต้องเป็นรูปสี่เหลี่ยมจัตุรัสที่มีขอบสีดำ พื้นหลังด้านในสี

ขาว และรูปแบบของ Marker เป็นสีดำ จะถูกจัดเก็บค่าพิกเซล (pixel) ให้อยู่ในรูปแบบของเมทริกซ์ (Matrix) และถูกจัดเก็บให้อยู่ให้ไฟล์นามสกุล ".patt"

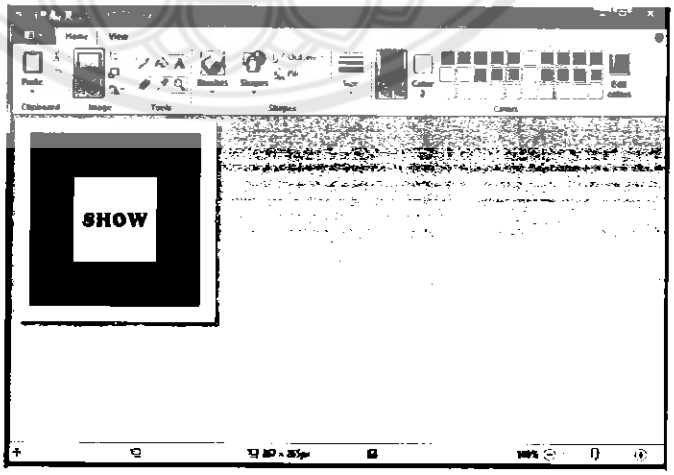
ขั้นตอนการออกแบบและการสร้างวัตถุสัญลักษณ์

1. การออกแบบวัตถุสัญลักษณ์

การออกแบบวัตถุสัญลักษณ์สามารถออกแบบได้ในโปรแกรมวาดภาพทั่วไป ในที่นี้ใช้โปรแกรม Paint ซึ่งสามารถสร้างเป็นภาพหรือตัวอักษรก็ได้ ดังรูปที่ 3.4 และรูปที่ 3.5



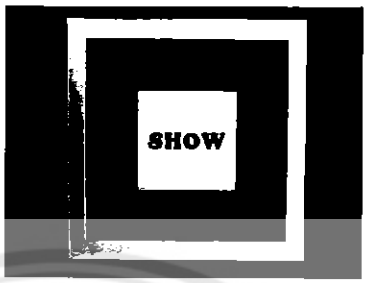
รูปที่ 3.4 แสดงการใช้โปรแกรม Paint สร้างวัตถุสัญลักษณ์คำว่า "AR"



รูปที่ 3.5 แสดงการใช้โปรแกรม Paint สร้างวัตถุสัญลักษณ์คำว่า "SHOW"

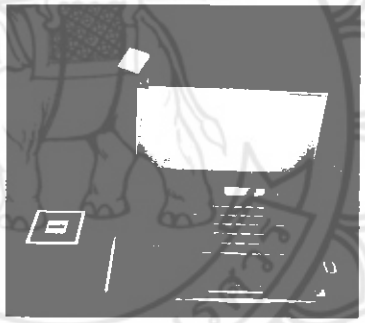
2. การสร้างวัตถุสัญลักษณ์

2.1 เมื่อทำการออกแบบวัตถุสัญลักษณ์เรียบร้อยแล้ว ให้ปริ้นวัตถุสัญลักษณ์ลงในกระดาษสีขาว ดังรูปที่ 3.6



รูปที่ 3.6 แสดงผลตัวอย่างการปริ้นวัตถุสัญลักษณ์

2.2 ขั้นตอนต่อไปเป็นการเก็บข้อมูลของวัตถุสัญลักษณ์ เพื่อนำข้อมูลนี้ไปใช้ในโปรแกรม โดยทำการติดตั้งอุปกรณ์ ดังรูปที่ 3.7

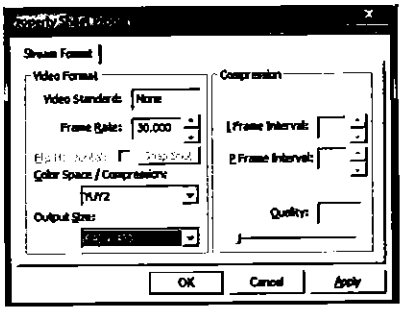


รูปที่ 3.7 แสดงผลตัวอย่างการติดตั้งอุปกรณ์

2.3 ไลบรารี ARToolKit มีเครื่องมือสำหรับการเก็บข้อมูลของวัตถุสัญลักษณ์นี้มาให้เรียบร้อยแล้ว โดยเข้าไปที่ C:\Program Files\ARToolKit\bin\mkpatt.exe เมื่อดับเบิ้ลคลิกจะแสดงหน้าต่างขึ้นมา ดังรูปที่ 3.8

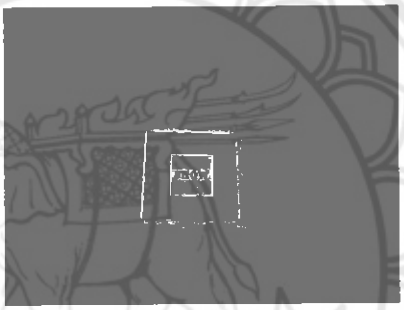
รูปที่ 3.8 แสดงผลหน้า cmd.exe

2.4 กดปุ่ม Enter ที่คีย์บอร์ด จะแสดงหน้าต่าง ดังรูปที่ 3.9



รูปที่ 3.9 แสดงตัวอย่างการเลือกขนาดกล้อง

2.5 คลิกปุ่ม OK จะแสดงหน้าต่าง ดังรูปที่ 3.10

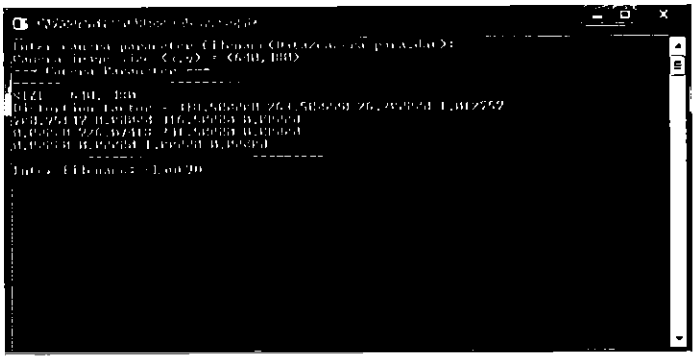


รูปที่ 3.10 แสดงผลภาพของวัตถุสัญลักษณ์

คลิกเมาส์ที่หน้าต่างแสดงรูปวัตถุสัญลักษณ์ 1 ครั้ง โปรแกรมจะทำการประมวลผลโดยอัตโนมัติ จากนั้นโปรแกรมจะแสดงหน้าต่างเพื่อให้ใส่ filename ดังรูปที่ 3.11

รูปที่ 3.11 แสดงผลหน้าต่าง cmd.exe เพื่อให้ใส่ filename

2.6 ใส่ชื่อของวัตถุสัญลักษณ์ ดังรูปที่ 3.12



รูปที่ 3.12 แสดงผลหน้าต่าง cmd.exe เพื่อให้ใส่ชื่อของวัตถุสัญลักษณ์

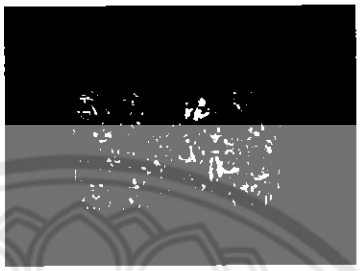
2.7 กดปุ่ม Enter ที่คีย์บอร์ด โปรแกรมจะทำการบันทึกชื่อและข้อมูลของวัตถุสัญลักษณ์ ดังรูปที่ 3.13

รูปที่ 3.13 แสดงผลหน้าต่าง cmd.exe เพื่อให้บันทึกชื่อและข้อมูล

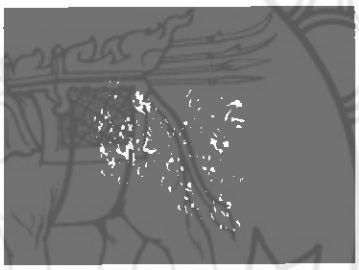
2.8 โปรแกรมจะบันทึกไฟล์ข้อมูลของวัตถุสัญลักษณ์ไว้ใน C:\Users\jumjim\AppData\Local\VirtualStore\Program Files\ARToolKit\bin ซึ่งสามารถนำไฟล์ข้อมูลนี้ไปใช้งานต่อไปได้

3.3 ออกแบบการสร้างวัตถุสามมิติ

การสร้างวัตถุสามมิติในโครงงานนี้เราจะใช้โปรแกรม MilkShape 3D 1.8.4 ในการออกแบบและ Export object เพื่อนำไปใช้แสดงผลในโปรแกรม ในหัวข้อนี้เราจะแสดงขั้นตอนการสร้างวัตถุสามมิติอย่างง่ายโดยใช้โปรแกรม MilkShape 3D 1.8.4 ดังรูปที่ 3.14 และรูปที่ 3.15



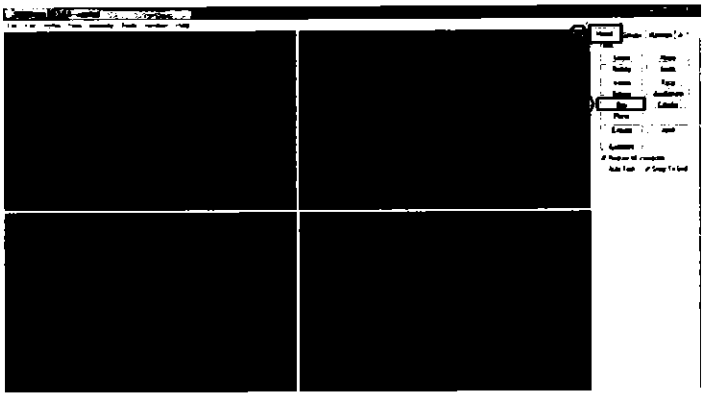
รูปที่ 3.14 แสดงตัวอย่างการสร้างวัตถุสามมิติด้านหน้า



รูปที่ 3.15 แสดงตัวอย่างการสร้างวัตถุสามมิติด้านข้าง

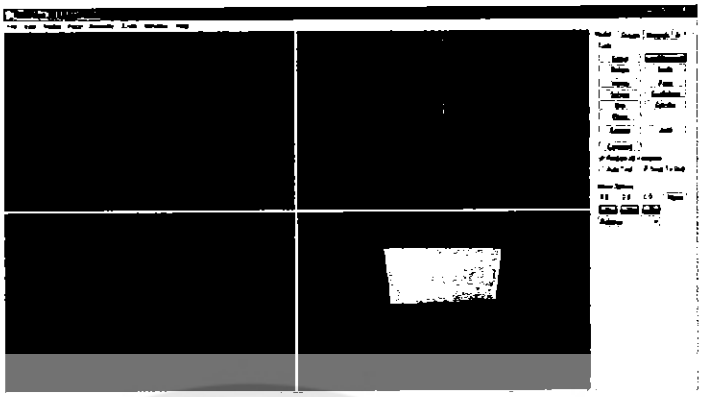
ขั้นตอนการสร้างวัตถุสามมิติ

1. เปิดโปรมแกรม MilkShape 3D 1.8.4 ขึ้นมา แล้วกดเลือกที่โหมด Model ดังหมายเลข 1 จากนั้นคลิกเลือกที่ปุ่ม Box ดังหมายเลข 2 เพื่อสร้างวัตถุรูปสี่เหลี่ยมขึ้นมา ดังรูปที่ 3.16



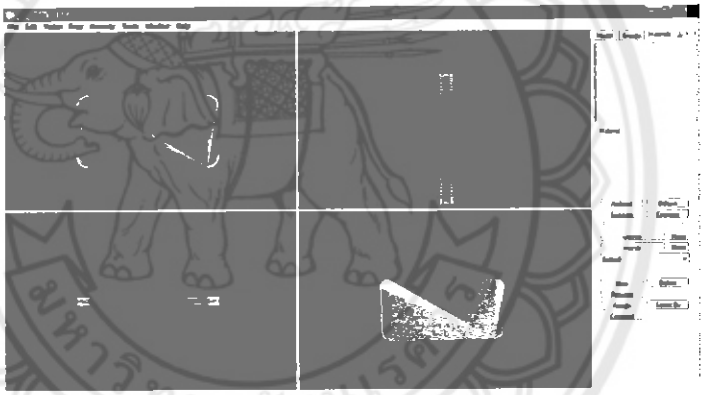
รูปที่ 3.16 แสดงผลการเปิดโปรมแกรม MilkShape 3D 1.8.4

2. ทำการลากเมาส์เพื่อวาดวัตถุลงในพื้นที่สร้างงานในโปรแกรม ดังแสดงในรูปที่ 3.17



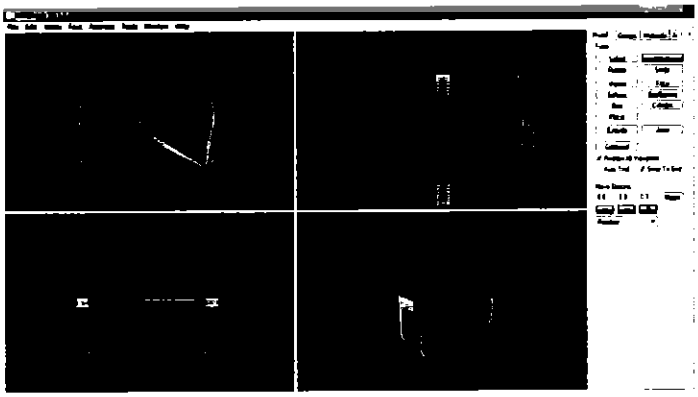
รูปที่ 3.17 แสดงผลการวาดวัตถุ

3. ปรับแต่งวัตถุให้มีส่วนโค้งมนตามต้องการ ดังรูปที่ 3.18



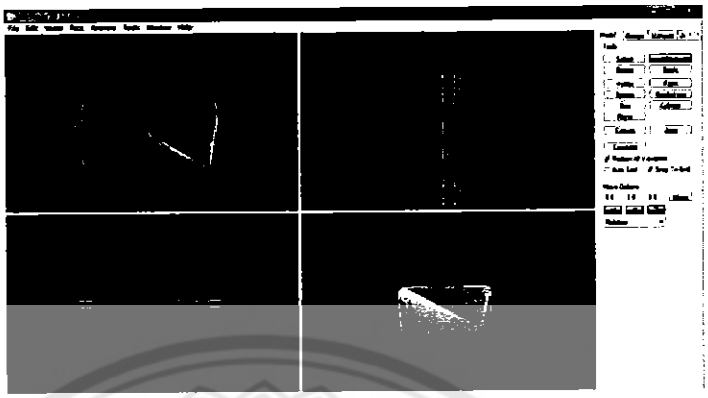
รูปที่ 3.18 แสดงผลการปรับแต่งวัตถุ

4. วาดกรอบวัตถุที่เป็นลักษณะสี่เหลี่ยมผืนผ้าขึ้นมาเพื่อสร้างเป็นกรอบของภาพ ดังรูปที่ 3.19



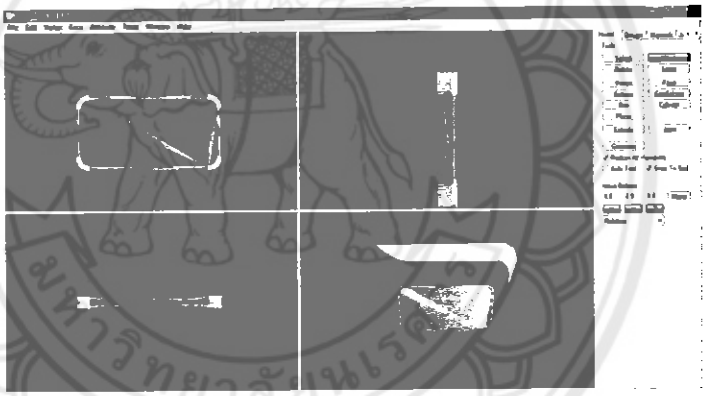
รูปที่ 3.19 แสดงผลการวาดกรอบวัตถุ

5. เลื่อนกรอบรูปสี่เหลียมผืนผ้าที่สร้างในข้อ 4 ไปรวมกับวัตถุที่สร้างขึ้นในครั้งแรก ดังรูปที่ 3.20



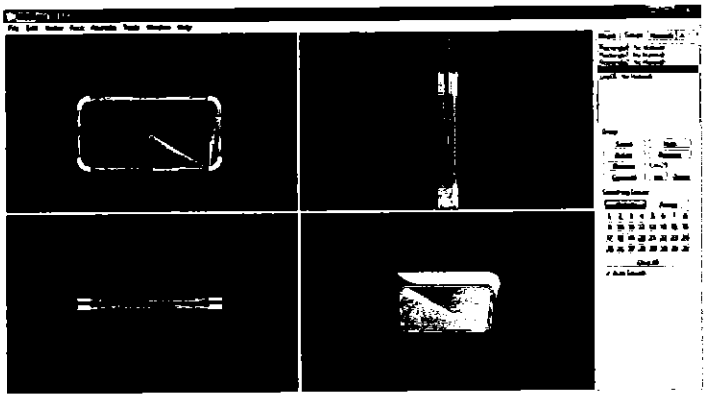
รูปที่ 3.20 แสดงผลการเลื่อนกรอบวัตถุ

6. สร้างส่วนประกอบที่มีลักษณะเป็นส่วนโค้งขึ้นมาเพื่อตกแต่งวัตถุรูปสี่เหลี่ยม ดังรูปที่ 3.21



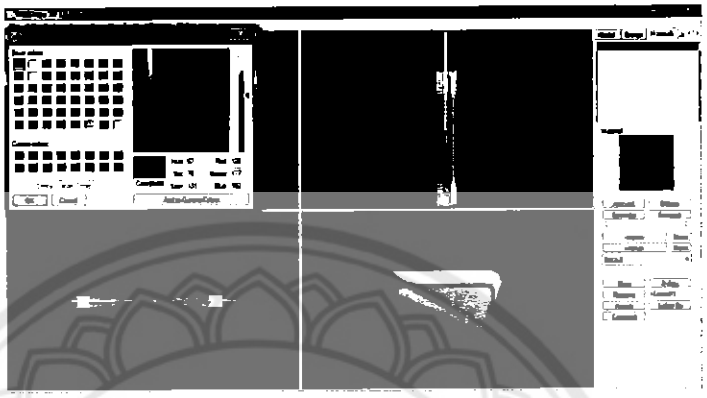
รูปที่ 3.21 แสดงผลการสร้างส่วนประกอบขึ้นมาตกแต่ง

7. นำวัตถุที่สร้างขึ้นในข้อ 6 ไปไว้ด้านบนวัตถุรูปสี่เหลี่ยม ดังรูปที่ 3.22

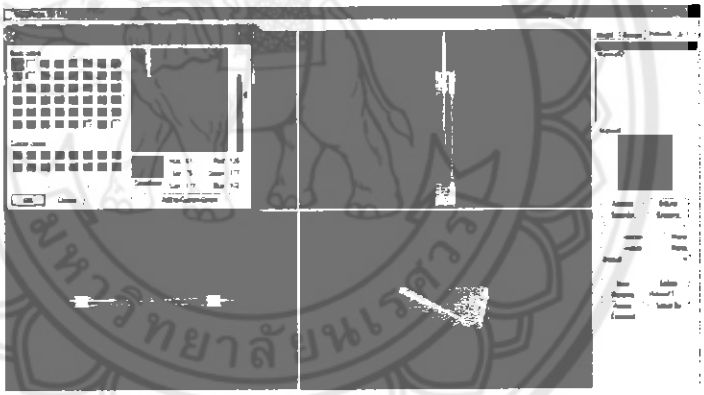


รูปที่ 3.22 แสดงผลการสร้างบนวัตถุรูปสี่เหลี่ยม

8. Set สีต่างๆของวัตถุ โดยเลือกจากโหมด Ambient, Diffuse, Specular และ Emissive ที่อยู่ทางด้านขวาของโปรแกรม ดังรูปที่ 3.23 และรูปที่ 3.24

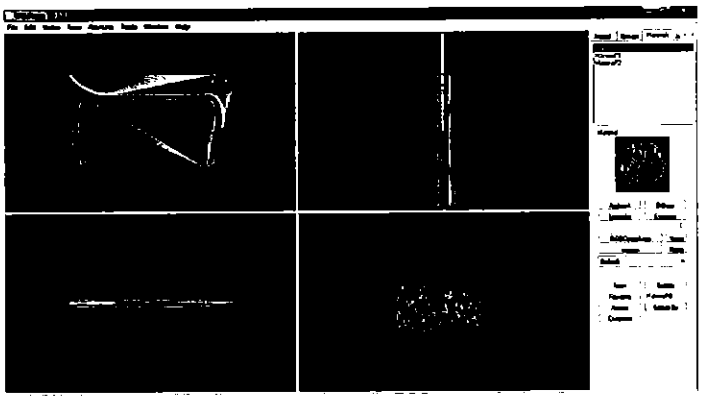


รูปที่ 3.23 แสดงผลการ Set สีต่างๆของวัตถุก่อนใส่สี



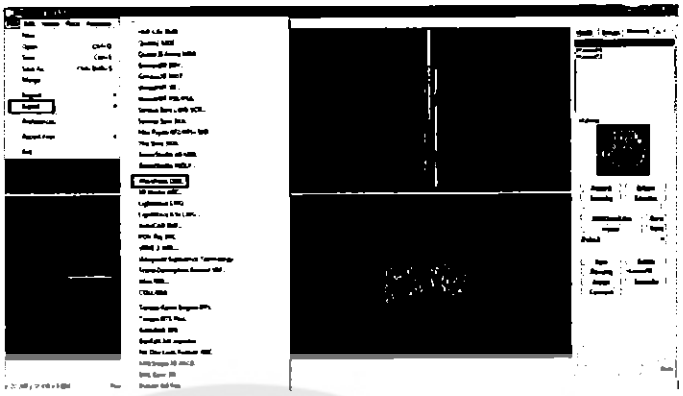
รูปที่ 3.24 แสดงผลการ Set สีต่างๆของวัตถุหลังใส่สี

9. ใส่ภาพ Texture เข้าในวัตถุสี่เหลี่ยม ดังรูปที่ 3.25



รูปที่ 3.25 แสดงผลการใส่ภาพ texture เข้าในวัตถุสี่เหลี่ยม

10. เมื่อสร้างเสร็จเรียบร้อยแล้วก็ Export Model ไปใช้งานในโปรแกรมต่อไป ดังรูปที่ 3.26



รูปที่ 3.26 แสดงผลการ Export Model

3.4 ออกแบบการสร้าง Texture Mapping

Texture mapping เป็นเทคนิคในงานกราฟิกที่นิยมใช้ในการทำให้วัตถุดูเสมือนจริง อีกทั้งยังแสดงผลได้อย่างรวดเร็วอีกด้วย โดยทั่วไปแล้วรูปถ่ายหรือภาพที่สร้างขึ้นจะถูกใช้เพื่อแม็พลงบนวัตถุหรือพื้นผิวที่อยู่ในปริภูมิ 3 มิติ จุด vertex บนวัตถุแต่ละจุดจะมีการกำหนดตำแหน่งพิกัดบนรูปภาพหรือที่เรียกว่า texture coordinate เพื่อให้รู้ว่าบริเวณไหนบนรูปจะถูกนำไปแสดง ณ บริเวณใดบนวัตถุ

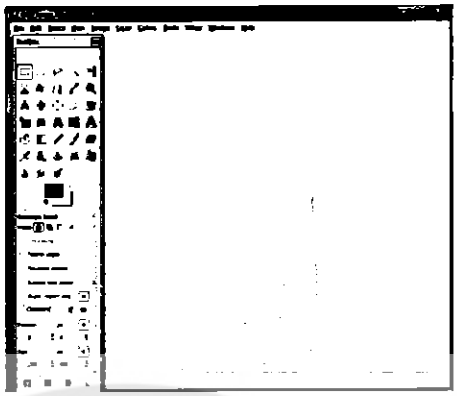
ขั้นตอนการสร้างภาพพื้นผิว (Texture)

การสร้างภาพพื้นผิวในโครงงานนี้เราจะสร้างพร้อมกับการสร้างวัตถุสามมิติในโปรแกรม MilkShape 3D 1.8.4 เพื่อความสะดวกและรวดเร็วในการนำไปแสดงผลในโปรแกรม ซึ่งไฟล์รูปภาพที่จะนำไปทำภาพพื้นผิวนั้น ต้องอยู่ในรูปแบบของ True Vision Targa (TGA) ซึ่งเป็นรูปแบบข้อมูลแบบบิตแมป ขนาด 24 บิต หรือ 32 บิต ทำให้ได้ภาพที่มีความละเอียดและความคมชัดสูง เหมาะสำหรับการนำมาใช้งานกราฟิกระดับสูง

1. **ขั้นตอนการสร้างไฟล์ True Vision Targa (TGA)**

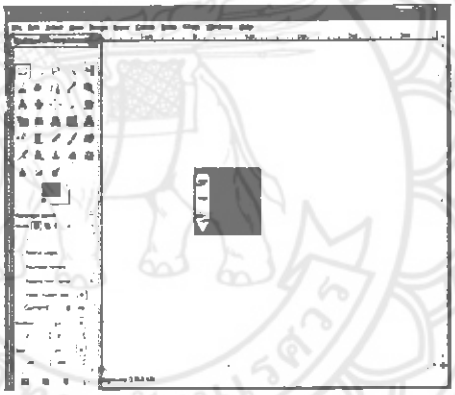
ในโครงงานนี้เราจะใช้โปรแกรม GNU Image Manipulation Program (Gimp) เพื่อสร้างไฟล์ True Vision Targa (TGA) โดยมีขั้นตอนดังนี้

1.1 เปิดโปรแกรม GNU Image Manipulation Program (Gimp) ขึ้นมา ดังรูปที่ 3.27



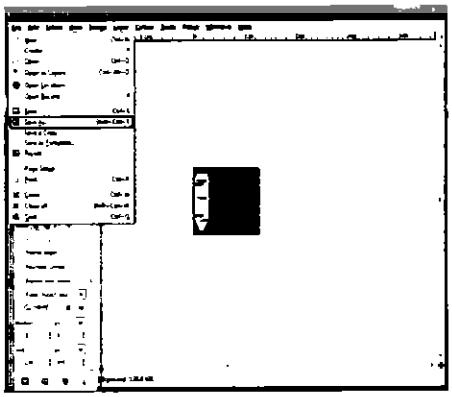
รูปที่ 3.27 แสดงผลการเปิดโปรแกรม GNU Image Manipulation Program (Gimp)

1.2 เปิดไฟล์รูปภาพขึ้นมา ดังรูปที่ 3.28



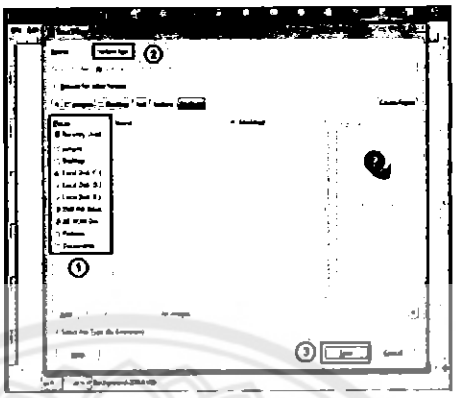
รูปที่ 3.28 แสดงผลการเปิดไฟล์รูปภาพขึ้นมา

1.3 คลิกเลือก file > Save As.. ดังรูปที่ 3.29



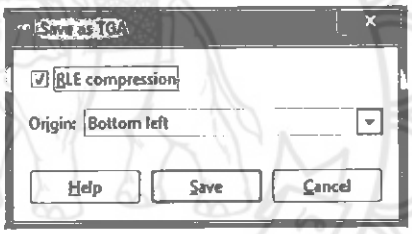
รูปที่ 3.29 แสดงผลการคลิกเลือก file > Save As..

1.4 เลือกที่เก็บไฟล์ดังหมายเลข 1 และใส่ชื่อไฟล์โดยให้นามสกุลเป็น .tga เช่น texture.tga ดังหมายเลข 2 จากนั้นคลิกเลือก Save ดังรูปที่ 3.30



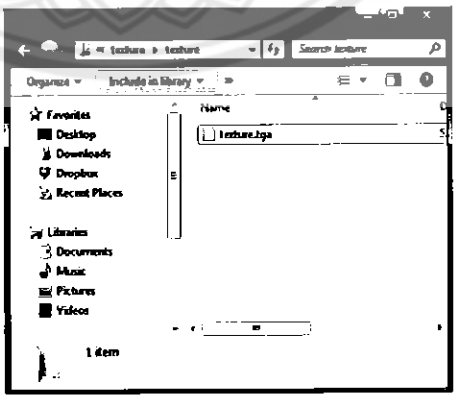
รูปที่ 3.30 แสดงผลการเลือกที่เก็บไฟล์

1.5 เลือก Bottom left แล้วเลือก Save ดังรูปที่ 3.31



รูปที่ 3.31 แสดงผลการเลือก Save as TGA

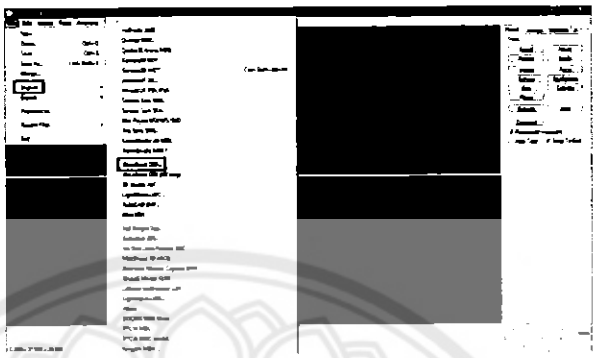
1.6 เสร็จแล้วจะได้ไฟล์ texture.tga มาเพื่อนำไปใช้ต่อไป ดังรูปที่ 3.32



รูปที่ 3.32 แสดงผลการได้ไฟล์ texture.tga

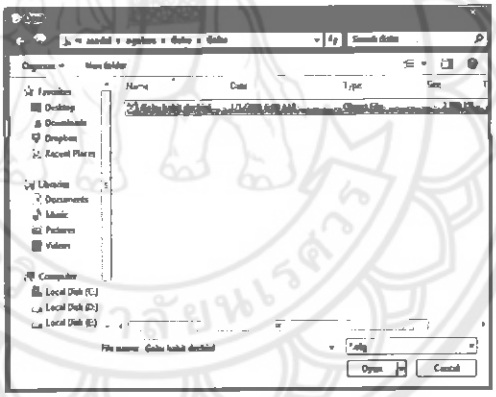
2. ขั้นตอนการสร้างภาพพื้นผิว

2.1 เปิดโปรแกรม MilkShape 3D 1.8.4 แล้ว Import ไฟล์ชนิด Wavefront OBJ (.obj) เข้ามาในโปรแกรม ดังรูปที่ 3.33



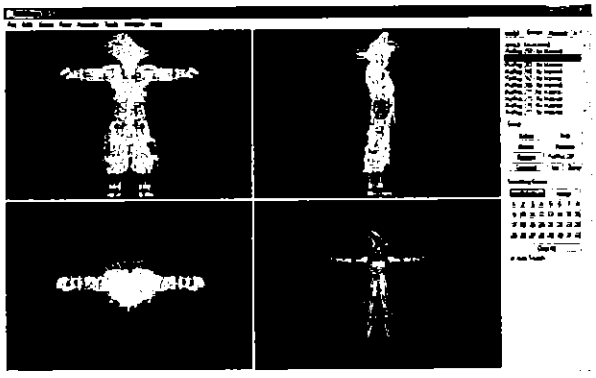
รูปที่ 3.33 แสดงผลการ Import ไฟล์

2.2 เลือกไฟล์โมเดลสามมิติที่ต้องการเข้ามา ดังรูปที่ 3.34



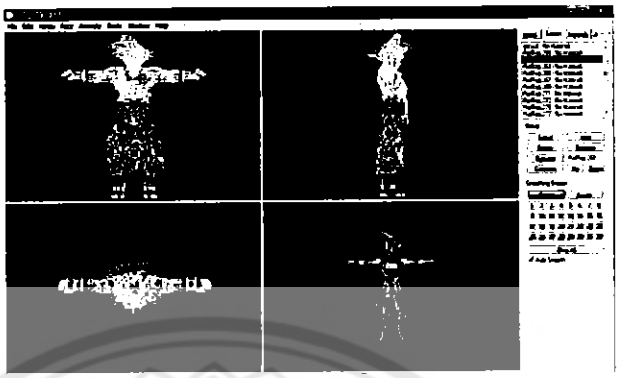
รูปที่ 3.34 แสดงผลการเลือกไฟล์โมเดลสามมิติ

2.3 โมเดลสามมิติจะแสดง ดังรูปที่ 3.35



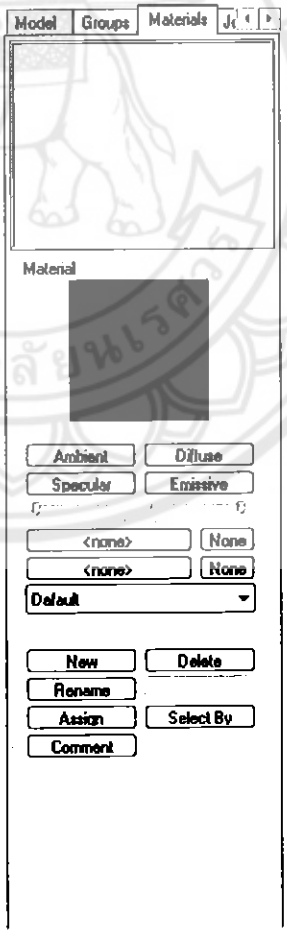
รูปที่ 3.35 แสดงผลโมเดลสามมิติ

2.4 เลือกส่วนที่จะใส่ภาพพื้นผิวในหน้าต่าง Groups ทางด้านขวาของโปรแกรม ดังรูปที่ 3.36



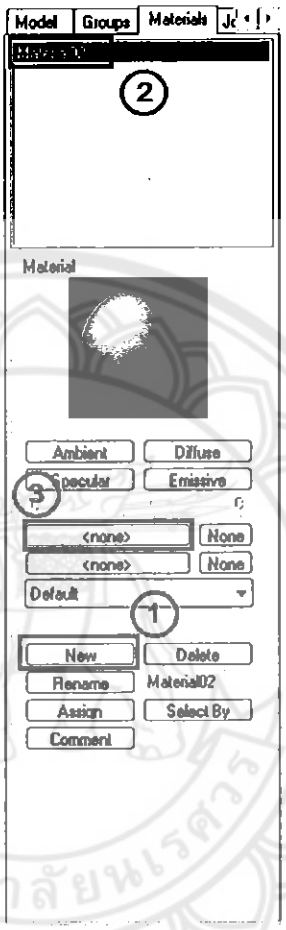
รูปที่ 3.36 แสดงผลการเลือกส่วนที่จะใส่ภาพพื้นผิว

2.5 จากนั้นไปที่หน้าต่าง Materials เพื่อใส่ภาพพื้นผิว ดังรูปที่ 3.37



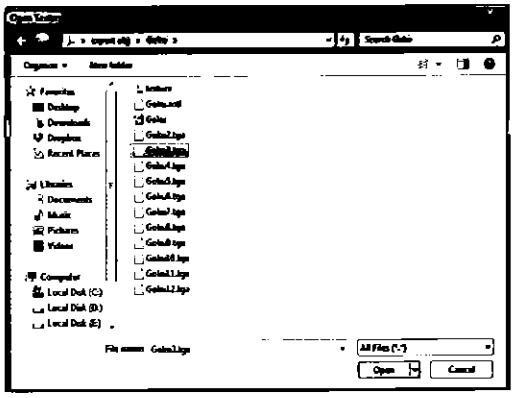
รูปที่ 3.37 แสดงผลหน้าต่าง Materials

2.6 ที่หน้าต่าง Materials เลือกปุ่ม New ดังหมายเลข 1 จะขึ้นชื่อ Matrrial02 ในหน้าต่างดังหมายเลข 2 จากนั้น คลิกเลือกปุ่ม <none> ดังหมายเลข 3 เพื่อใส่ภาพพื้นผิว ดังรูปที่ 3.38



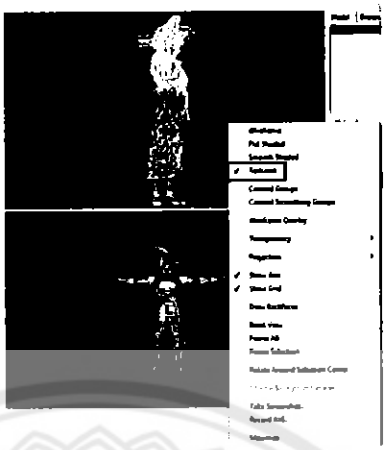
รูปที่ 3.38 แสดงผลการตั้งค่าหน้าต่าง Materials

2.7 เลือกไฟล์ภาพพื้นผิวที่ชนิด True Vision Targa (TGA) แล้วเลือก Open ดังรูปที่ 3.39



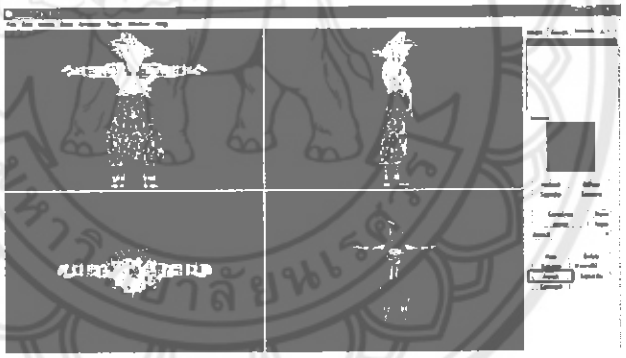
รูปที่ 3.39 แสดงผลการเลือกไฟล์ภาพพื้นผิวที่ชนิด True Vision Targa (TGA)

2.8 คลิกขวาที่พื้นที่สร้างงานในโปรแกรม แล้วคลิกเลือกในหัวข้อ Textured ดังรูปที่ 3.40



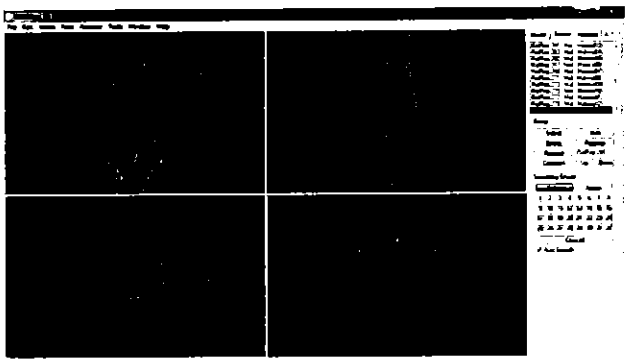
รูปที่ 3.40 แสดงผลการเลือกหัวข้อ Textured

2.9 จากนั้นคลิกปุ่ม Assign ในหน้าต่างด้านขวา โปรแกรมจะแสดงภาพพื้นผิวที่ได้ใส่เข้าไป ดังรูปที่ 3.41



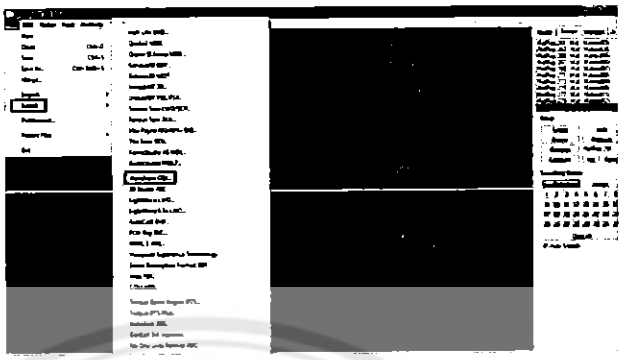
รูปที่ 3.41 แสดงผลการใส่ Textured

2.10 ทำการใส่ภาพพื้นผิวให้หมดทั้งวัตถุสามมิติ เมื่อใส่เสร็จแล้วจะแสดง ดังรูปที่ 3.42



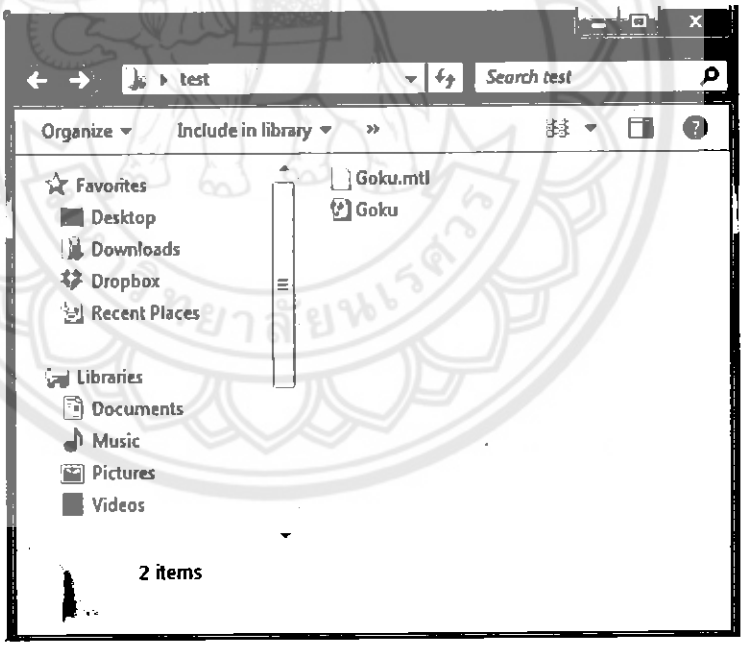
รูปที่ 3.42 แสดงผลการใส่ Textured ทั้งหมด

2.11 ทำการ Export โมเดลสามมิติให้อยู่ในรูปแบบ Wavefront OBJ (.obj) เพื่อนำไปใช้งาน ดังรูปที่ 3.43



รูปที่ 3.43 แสดงผลการทำ Export โมเดลสามมิติ

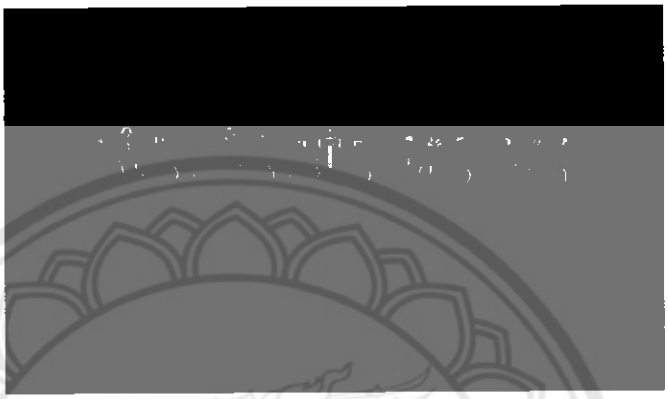
2.12 เมื่อ Export เสร็จเรียบร้อยจะได้ทั้งหมดสองไฟล์คือ .obj และ .mtl ในการนำไปใช้งานต้องเอาทั้งสองไฟล์ไปด้วย ดังรูปที่ 3.44



รูปที่ 3.44 แสดงผลทั้งหมดสองไฟล์คือ .obj และ .mtl

3.5 ออกแบบการสร้างภาพเคลื่อนไหว (Animation)

การสร้างภาพเคลื่อนไหว (Animation) ในโครงงานนี้จะใช้วิธีการสร้างโมเดลสามมิติขึ้นมาก่อนแล้วเขียนโปรแกรม เพื่อสร้างการเคลื่อนไหวให้กับโมเดลสามมิติ ตัวอย่างภาพเคลื่อนไหวแสดง ดังรูปที่ 3.45



รูปที่ 3.45 แสดงตัวอย่างภาพเคลื่อนไหว

ขั้นตอนการสร้างภาพเคลื่อนไหว (Animation)

1. สร้างโมเดลสามมิติขึ้นมาโดยเริ่มจาก แท่งสี่เหลี่ยมผืนผ้าที่มีลักษณะเป็นช่องวงกลมอยู่ด้านบน ดังรูปที่ 3.46

รูปที่ 3.46 แสดงตัวอย่างการสร้างแท่งสี่เหลี่ยมผืนผ้า

2. สร้างโมเดลสามมิติรูปการ์ตูนขึ้นมา แล้วนำไปวางไว้บนแท่งสี่เหลี่ยมผืนผ้า โดยให้มีตำแหน่งตรงกับวงกลมช่องแรก ดังรูปที่ 3.47

รูปที่ 3.47 แสดงตัวอย่างการสร้างโมเดลสามมิติรูปการ์ตูน

3. จากนั้นสร้างโมเดลสามมิติรูปการ์ตูนขึ้นมาวางไว้จนครบทั้ง 5 ช่อง **ดังรูปที่** 3.48

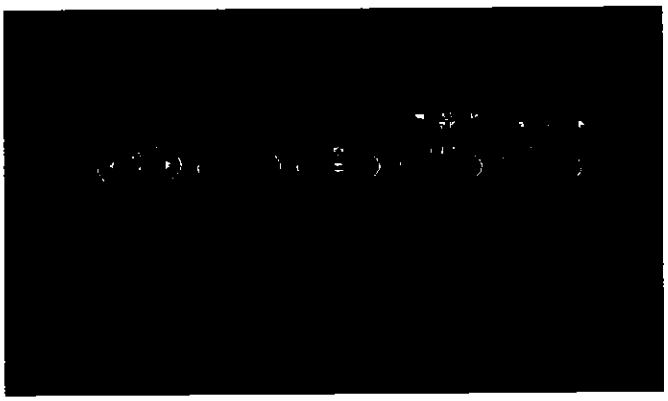
รูปที่ 3.48 แสดงตัวอย่างการสร้างโมเดลสามมิติรูปการ์ตูนทั้งหมด

4. ต่อไปจะเป็นในส่วนของการเขียนโปรแกรม เพื่อทำให้โมเดลสามมิติมีการเคลื่อนไหว ดังรูปที่ 3.49

```
static void animateDrawSelect()
{
    chkAni1 += 0.2;
    chkAni2 += 0.2;
    chkAni3 += 0.2;
    chkAni4 += 0.2;
    chkAni5 += 0.2;


    // model 1
    if(chkAni1 > 0)
    {
        model1 -= 0.1;
    }
    if(chkAni1 > 15)
    {
        model1 += 0.3;
    }
    if(chkAni1 > 25)
    {
        model1 -= 0.3;
    }
    if(chkAni1 > 30)
    {
        chkAni1 = 0;
        model1 = 0.0;
    }

    void animate()
    {

        // drawSelectFunction
        animateDrawSelect();


        glutPostRedisplay();
    }
```

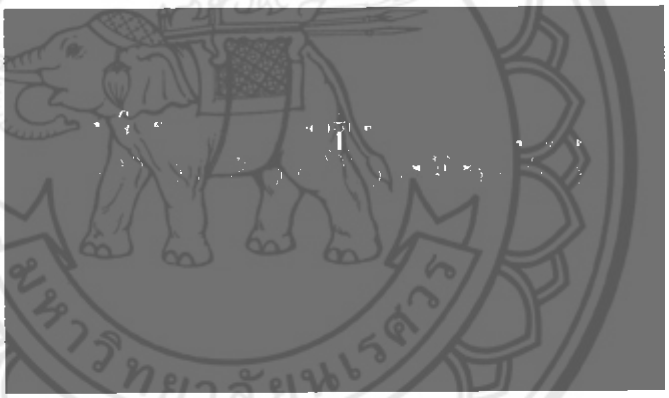
รูปที่ 3.49 แสดงตัวอย่างของโค้ดทำให้โมเดลสามมิติเคลื่อนไหว

5. จากนั้นรันโปรแกรมเพื่อดูผลลัพธ์ ซึ่งโมเดลสามมิติจะเลื่อนขึ้นและเลื่อนลงในช่องวงกลมโดยอัตโนมัติเป็นแบบนี้ไปเรื่อยๆแบบต่อเนื่อง ดังรูปที่ 3.50 – **รูปที่** 3.52



รูปที่ 3.50 แสดงตัวอย่างโมเดลสามมิติ

รูปที่ 3.51 แสดงตัวอย่างโมเดลสามมิติ



รูปที่ 3.52 แสดงตัวอย่างโมเดลสามมิติ

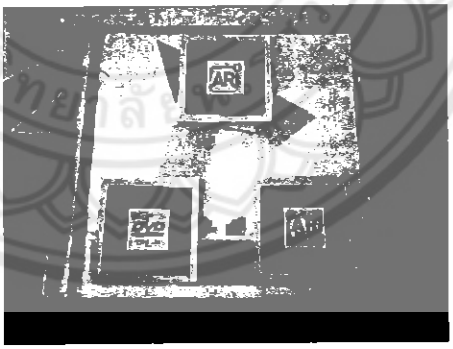
บทที่ 4
ผลการทดลอง

โครงงาน "ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี" เป็นการสร้างกราฟิกสามมิติและวิดีโอขึ้นมาเพื่อนำมาพัฒนาระบบความจริงเสริมในการผลิตสื่อโฆษณาเพื่อส่งเสริมการตลาดของสินค้าประเภทดีวีดี ซึ่งผลของการทดลองในโครงงานนั้นจะมีลักษณะแสดงผลกราฟิกสามมิติและวิดีโอพร้อมเสียง

4.1 ผลการทดลองการค้นหาวัตถุสัญลักษณ์ (Marker)

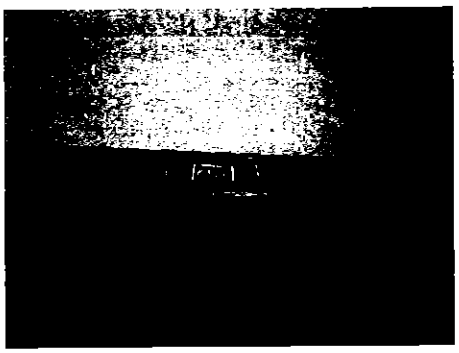
1. ผลการตรวจพบวัตถุสัญลักษณ์

โปรแกรมจะค้นหาวัตถุสัญลักษณ์โดยทำการวาดกรอบสี่เหลี่ยมล้อมรอบกรอบสีดำของวัตถุสัญลักษณ์ ซึ่งเมื่อตรวจพบวัตถุสัญลักษณ์โปรแกรมจะวาดกรอบสี่เหลี่ยมสีเขียวล้อมรอบ ซึ่งกล้องสามารถตรวจพบวัตถุสัญลักษณ์ได้ทั้งด้านบน ด้านข้าง และด้านหน้า ดังรูปที่ 4.1 – รูปที่ 4.3



รูปที่ 4.1 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านบน

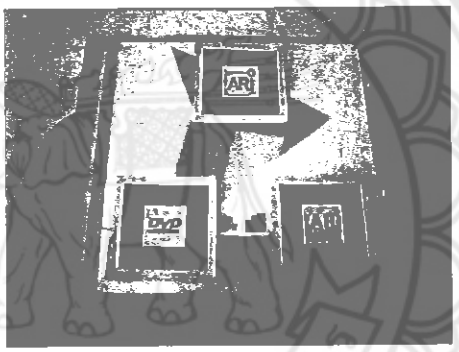
รูปที่ 4.2 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านข้าง



รูปที่ 4.3 แสดงผลการตรวจพบวัตถุสัญลักษณ์ด้านหน้า

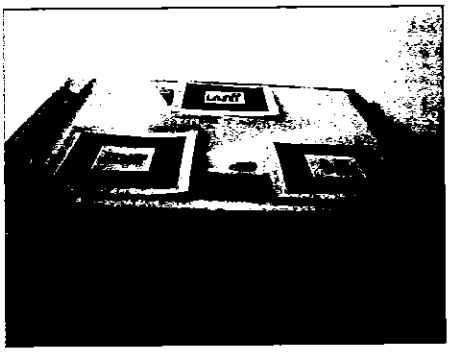
2. ผลการตรวจไม่พบวัตถุสัญลักษณ์

เมื่อโปรแกรมตรวจไม่พบวัตถุสัญลักษณ์จะแสดง ดังรูปที่ 4.4 – **รูปที่ 4.6**



รูปที่ 4.4 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านบน

รูปที่ 4.5 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านข้าง

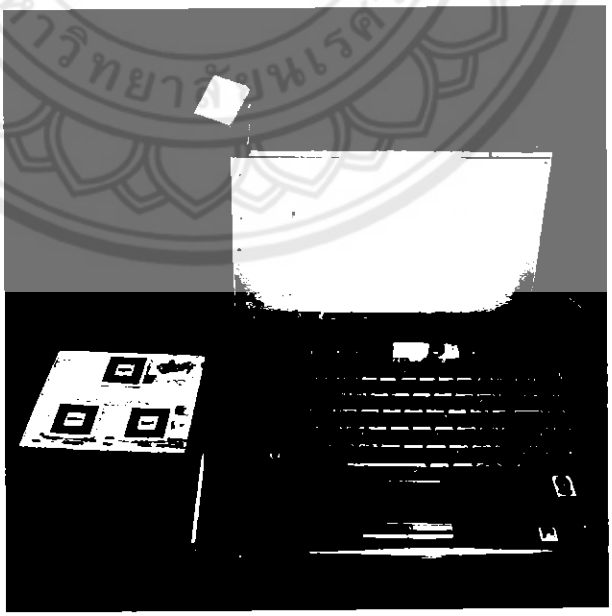


รูปที่ 4.6 แสดงผลการตรวจไม่พบวัตถุสัญลักษณ์ด้านหน้า

4.2 ผลการทดลองการติดตั้งอุปกรณ์

การติดตั้งอุปกรณ์ในการทำงานและแสดงผลของโปรแกรมประกอบด้วย

1. คอมพิวเตอร์ที่ใช้ในการประมวลผลโปรแกรม
2. กล้องเว็บแคม ซึ่งต้องติดตั้งให้สามารถมองเห็นวัตถุสัญลักษณ์บนแผ่นดีวีดีได้ชัดเจน
3. แผ่นดีวีดี ที่ทำการติดวัตถุสัญลักษณ์เรียบร้อยแล้ว

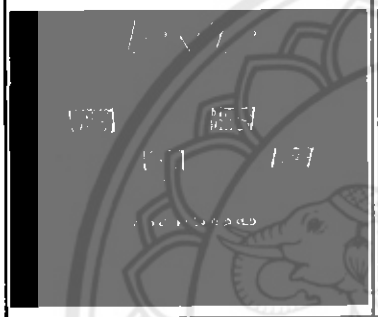
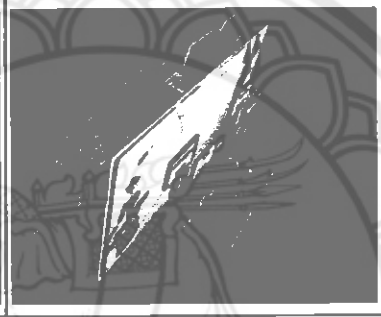
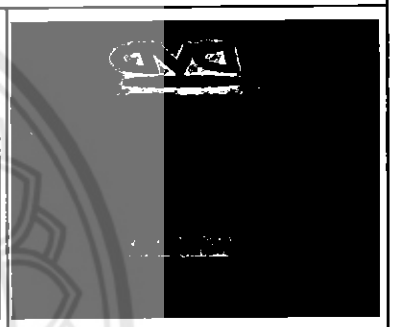
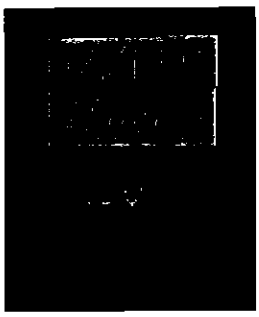


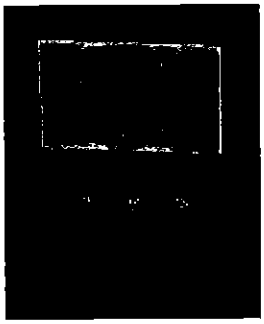
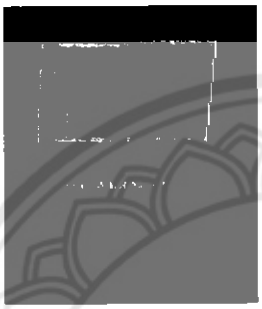
รูปที่ 4.7 แสดงผลการติดตั้งอุปกรณ์

4.3 ผลการทดลองของโปรแกรม

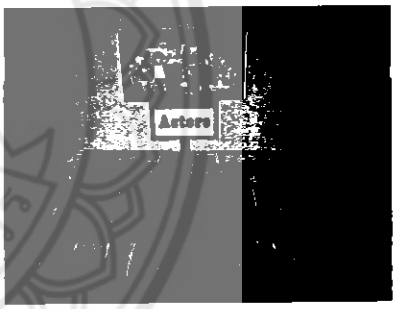
โปรแกรมจะทำงานเมื่อเราทำการเชื่อมต่อกล้องและค้นหาวัตถุสัญลักษณ์ที่สร้างไว้ เมื่อเจอแล้วจะตีกรอบสี่เหลี่ยมสีเขียวล้อมรอบรูป จากนั้นโปรแกรมจะแสดงผลผ่านทางหน้าจอ ในหน้าต่างแรกนั้นจะเป็นส่วนของการทำงานหลัก ประกอบไปด้วยรายละเอียดต่างๆในการแสดงการทำงานดังตารางที่ 4.1

ตารางที่ 4.1 แสดงผลการทำงานของโปรแกรม

1. แสดงผลการทำงานของโปรแกรม		
		
ด้านหน้า	ด้านข้าง	ด้านหลัง
2. แสดงผลการทำงานของการเลือกโหมดต่างๆ		
	Introduce Actor เป็นโหมดที่ไว้สำหรับแสดงรายละเอียดของการแนะนำนักแสดงภายในเรื่อง	
	Trailer Movie เป็นโหมดสำหรับแสดงรายละเอียดของการแนะนำตัวอย่างของภาพยนตร์ภายในเรื่อง	

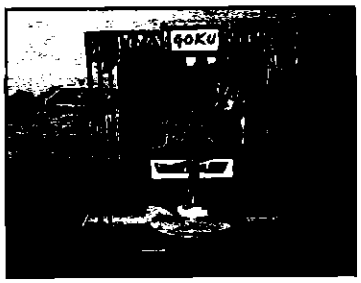
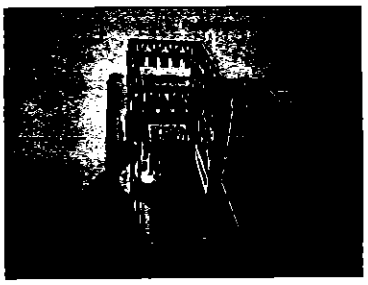
	Song and Gallery เป็นโหมดสำหรับแสดงรายละเอียดของการแนะนำเสียงประกอบและรูปภาพของตัวละครภายในเรื่อง
	Movies Detail เป็นโหมดสำหรับแสดงรายละเอียดของการแนะนำรายละเอียดของภาพยนตร์ภายในเรื่อง

3. แสดงผลการทำงานโหมด Introduce Actor

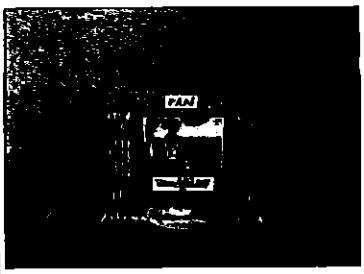
	
ด้านหน้า	ด้านหลัง

4. แสดงผลการทำงานโหมด Introduce Actor ของแต่ละตัวละคร

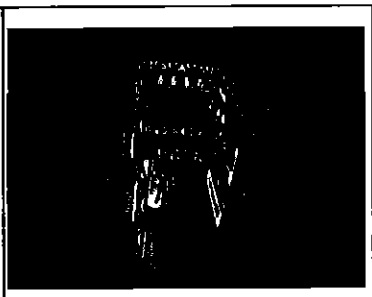
4.1 GOKU

		
ด้านหน้า	ด้านข้าง	ด้านหลัง

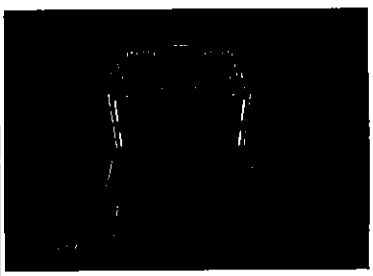
4.2 PAN



ด้านหน้า



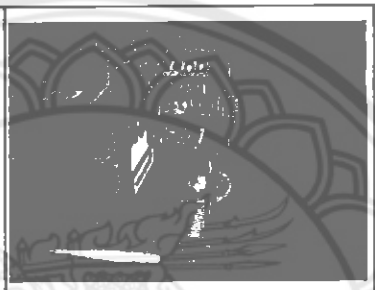
ด้านข้าง



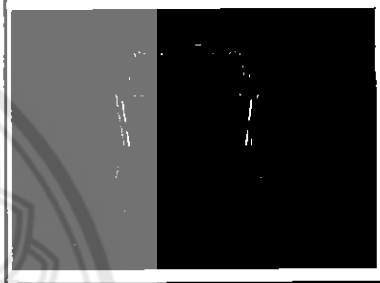
ด้านหลัง

4.3 TRUNKS

ด้านหน้า



ด้านข้าง

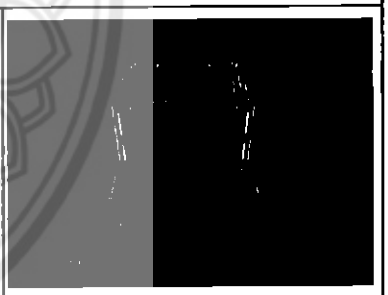


ด้านหลัง

4.4 GOHAN

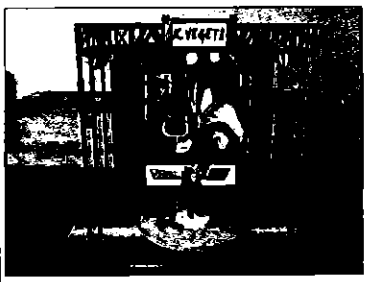
ด้านหน้า

ด้านข้าง

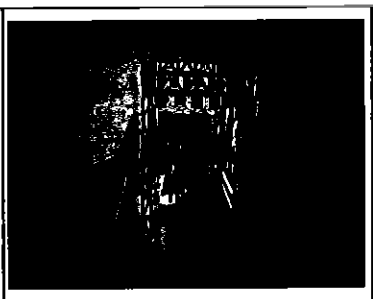


ด้านหลัง

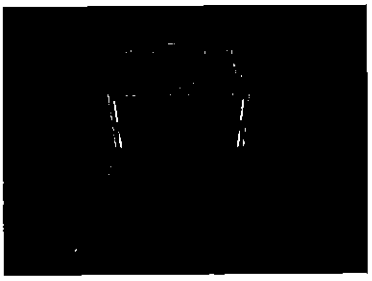
4.5 KING VEGETA



ด้านหน้า

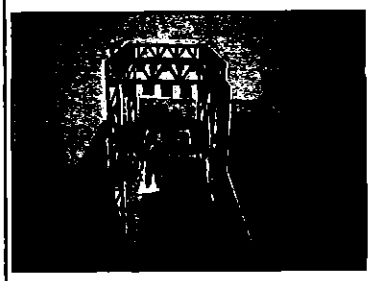


ด้านข้าง



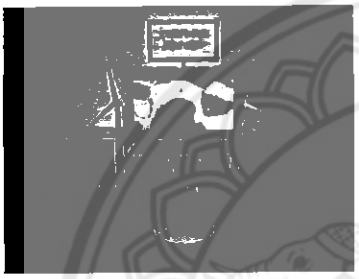
ด้านหลัง

4.6 MUTEN ROSHI



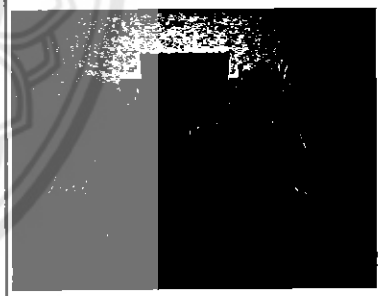
ด้านหน้า		ด้านข้าง

5. แสดงผลการทำงานโหมด Trailer Movie



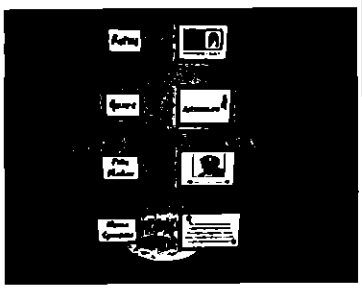
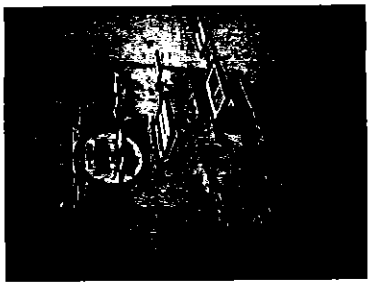
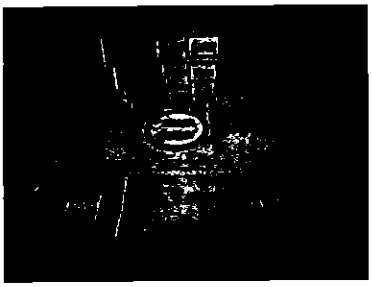
ด้านหน้า	ด้านข้าง	ด้านหลัง

6. แสดงผลการทำงานโหมด Song and Gallery



ด้านหน้า	ด้านข้าง	ด้านหลัง

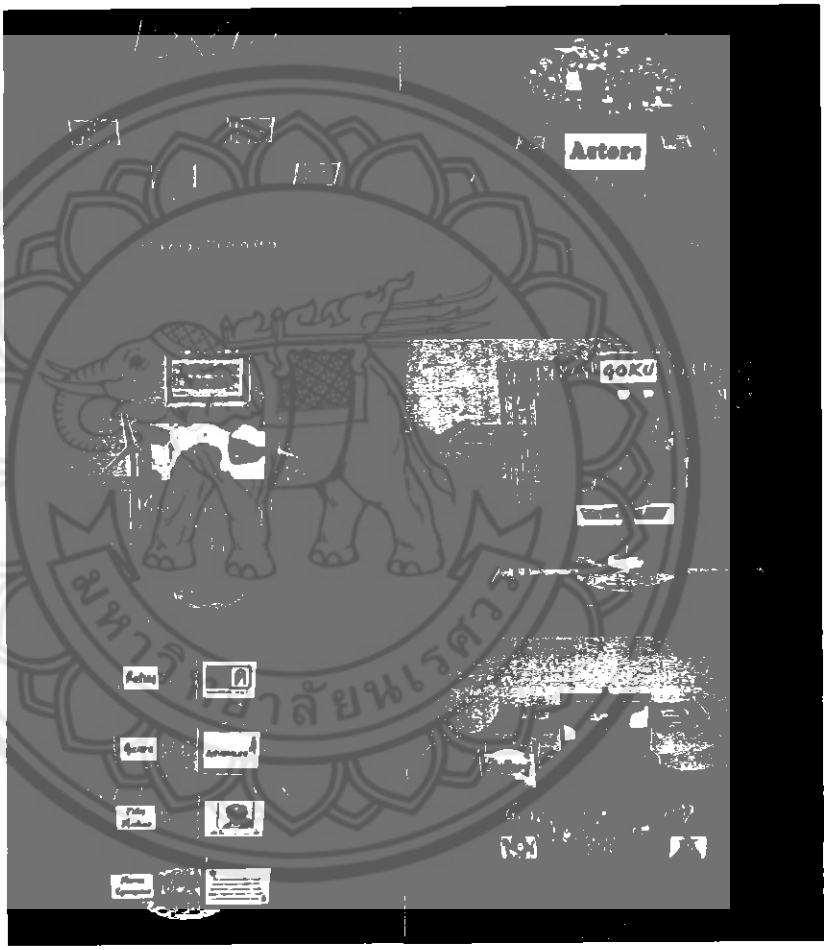
7. แสดงผลการทำงานโหมด Movies Detail

ด้านหน้า	ด้านข้าง	ด้านหลัง

สรุปผลการทดลอง

จากผลการทดลอง "ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี" สามารถแสดงผลภาพกราฟิกสามมิติ ภาพเคลื่อนไหว ภาพวิดีโอและเสียงประกอบได้ ซึ่งจะแสดงผลตามโหมดที่เลือกโดยจะแสดงบนวัตถุสัญลักษณ์ ซึ่งสามารถแสดงผลได้โดยมีมุมมองถึง 360 องศา ดังแสดงในรูปที่ 4.8



รูปที่ 4.8 แสดงผลการทดลอง

บทที่ 5
สรุปผลและข้อเสนอแนะ

บทนี้กล่าวถึงบทสรุป ปัญหาที่พบและข้อเสนอแนะของโครงงาน "ระบบความจริงเสริมเพื่อการส่งเสริมการตลาดของดีวีดี" (Augmented Reality for marketing promotion of DVD) เพื่อให้เกิดความเข้าใจและสามารถนำโครงงานนี้ไปพัฒนาต่อไป

5.1 สรุปผลการทดลอง

โครงงานนี้จัดทำขึ้นเพื่อนำเทคโนโลยีความจริงเสริม (Augmented Reality) มารวมกับโมเดลสามมิติและการแสดงวิดีโอพร้อมเสียงเพื่อให้ความบันเทิง นันทนาการ และให้ผู้ใช้ได้รับประสบการณ์ใหม่ ๆ เกี่ยวกับการใช้งานเพื่อการส่งเสริมการขาย ความสะดวกสบาย ซึ่งผู้ใช้สามารถควบคุมการใช้งานของการแสดงรายละเอียดของภาพยนตร์ ตัวละคร รวมถึงตัวอย่างภาพยนตร์ ผ่านทางวัตถุสัญลักษณ์ (Marker) และดูภาพเสมือนจริงที่ผสมผสานภาพในโลกแบบ Real Time ทำให้ผู้ใช้ได้มีปฏิสัมพันธ์โดยตรงกับภาพเสมือนจริงและผลลัพธ์ของการแสดงภาพกราฟิก ทำให้ผู้ใช้เกิดประสบการณ์ใหม่ ๆ และมีความสนใจในเทคโนโลยีความจริงเสริมมากขึ้น

5.2 ปัญหาที่พบ

จากการทดลองโครงงานนี้ทำให้พบปัญหาใหญ่ๆ 2 ส่วนได้แก่การศึกษาและใช้งานไลบรารีต่างๆที่ใช้ในโครงงานนี้ และขั้นตอนการทดลอง

1. ปัญหาการใช้งานไลบรารี ARToolKit, OpenGL, irrKlang และ OpenCV ที่ต้องใช้เวลาศึกษาพอสมควร เนื่องจากไม่เคยใช้ไลบรารีเหล่านี้มาก่อน

2. อุปกรณ์ที่ใช้ในการทดลองของโครงงานไม่ค่อยอำนวยความสะดวก จึงทำให้ขั้นตอนการทดลองเกิดการผิดพลาดบ้าง

3. สภาพแวดล้อมและสถานที่ในการทดลองไม่เอื้ออำนวย เช่น เรื่องแสง เป็นต้น ทำให้เกิดการติดขัดในการทดลองของโครงงาน

5.3 ข้อเสนอแนะและวิธีการแก้ไขปัญหา

จากปัญหาที่พบในโครงงานนี้ ทางผู้จัดทำขอเสนอข้อเสนอแนะ และวิธีการแก้ไขปัญหา ดังนี้

1. วางแผนการศึกษาไลบรารีแต่ละแบบให้เข้าใจ และศึกษาตามเว็บไซต์ต่างประเทศเพราะมีข้อมูลและวิธีการใช้งานไลบรารีมากกว่าเว็บไซต์ในประเทศไทย

2. จัดหาอุปกรณ์ต่างๆ ดังนี้

2.1 กล้อง Webcam

2.2 ขาตั้งกล้อง Webcam

2.3 ฟิวเจอร์บอร์ดเพื่อนำมาเป็นพื้นในการทดลอง เพื่อลดปัญหาการค้นหาวัตถุสัญลักษณ์ (Marker)

2.4 วัตถุสัญลักษณ์ (Marker) ให้ออกแบบเองให้มีความแตกต่างกันพอสมควร

3. หาสถานที่เพื่อทำการทดลองที่มีแสงไม่มากและไม่น้อยเกินไป เพื่อลดปัญหาการค้นหาวัตถุสัญลักษณ์ (Marker)

เอกสารอ้างอิง

[1] Srinakharinwirot University. "A Study of Augmented Reality Technologies: Case Study Developing "MemCards" Game." [Online]. เข้าถึงได้จาก : http://facstaff.swu.ac.th/praditm/CP499_2552_AR.pdf

[2] "**Augmented Reality เทคนิคเพิ่มความเสมือนจริง.**" [Online]. เข้าถึงได้จาก : http://plakim69.wordpress.com/2009/09/04/augmented-reality/

[3] "To Be Launched: Adidas Augmented Reality." [Online]. เข้าถึงได้จาก : http://www.viralblog.com/widgets-social-apps/to-be-launched-adidas-augmented-reality/

[4] National Electronics and Computer Technology Center. "**เทคโนโลยีการประมวลผลภาพ Image Processing.**" [Online]. เข้าถึงได้จาก : http://www.nectec.or.th/index.php?option=com_content&view=article&id=289&catid=40&Itemid=165

[5] "**Augmented Reality แนวทางการประยุกต์ใช้ในธุรกิจ.**" [Online]. เข้าถึงได้จาก : http://kafaak.wordpress.com/2010/05/22/ar-and-daily-life/

[6] "Augmented Reality : Travel Is Coming Very, Very Soon." [Online]. เข้าถึงได้จาก : http://www.vagabondish.com/augmented-reality-travel/

[7] "**เทคโนโลยี ARToolKit เครื่องมือช่วยวาดวัตถุ 3D ประกอบฉาก.**" [Online]. เข้าถึงได้จาก : http://www.hitl.washington.edu/artoolkit/documentation/cs.htm

[8] "ARToolKit Tutorial 1: Tracking Stability." เข้าถึงได้จาก : www.hitl.washington.edu/artoolKit/documentation/tutorialhistory.htm

[9] "Three dimensional geometry transformations with OpenGL." เข้าถึงได้จาก : 202.28.94.55/web/320491/2546/seminar/g24/seminar.doc

[10] น.ท.ไพศาล โมลิสกุลมงคล. (2550). **คอมพิวเตอร์กราฟิกส์ (Computer Graphics using OpenGL).** กรุงเทพฯ: สำนักพิมพ์ดวงกมลสมัย.

[11] "OpenCV." เข้าถึงได้จาก : http://opencv.willowgarage.com/wiki/

[12] "Marker Detection for Augmented Reality Application." เข้าถึงได้จาก : http://infi.nl/blog/view/id/56/

[13] "**การหาค่า Threshold.**" เข้าถึงได้จาก : http://www.narisa.com/forums/index.php?showtopic=23541

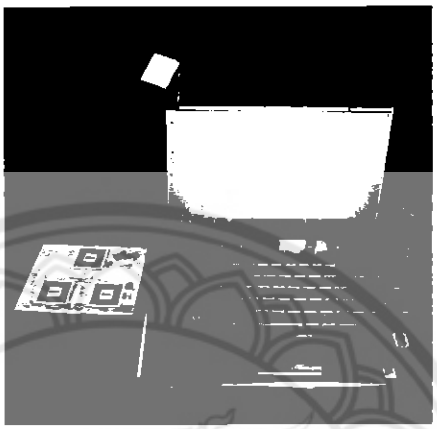
[14] "**ระบบสี RGB.**" เข้าถึงได้จาก : http://web.ofebia.com/contents/view/50.htm

[15] "**Ambiera IrrKlang.**" เข้าถึงได้จาก : http://www.ambiera.com/irrKlang

ภาคผนวก

ขั้นตอนการทำงานของโปรแกรม

1. ติดตั้งอุปกรณ์การแสดงผล



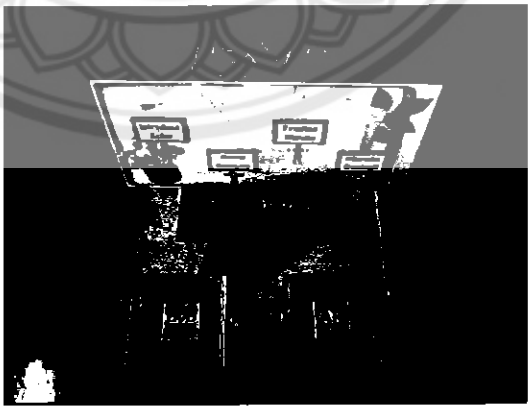
2. เมื่อรันโปรแกรมขึ้นมา หน้าแรกจะเป็นการแสดงโหมดต่างๆ ให้เลือกดู ซึ่งประกอบไปด้วย

   2.1 Introduce Actor

   2.2 Trailer Movie

   2.3 Song & Gallery

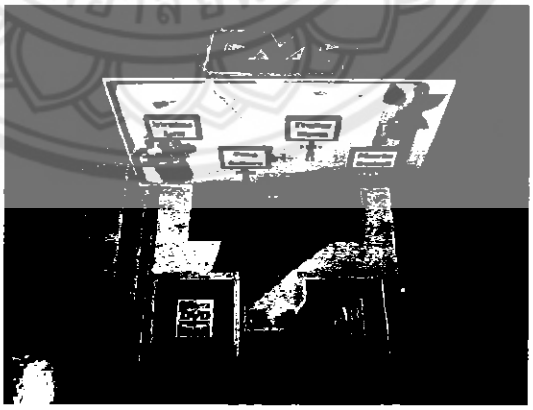
   2.4 Movie Detail



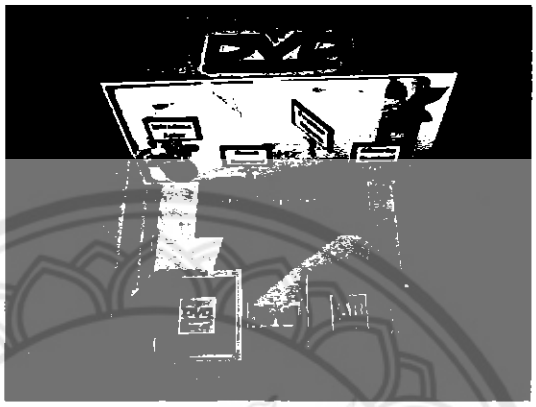
3. ในการเข้าไปดูการแสดงผลในโหมดต่างๆ สามารถทำได้โดยการใช้มือปิดวัตถุสัญลักษณ์ทางด้านซ้ายมือของแผ่นดีวีดี จะเห็นว่าโมเดลสามมิติของโหมด Introduce Actor หมุนรอบตัวเอง เพื่อแสดงสถานะว่ากำลังจะแสดงผลในโหมดนี้

4. แสดงผลการทำงานของโหมด Introduce Actor ซึ่งจะเป็นการแนะนำตัวละครต่างๆ

5. เมื่อต้องการกลับไปยังหน้าแรก สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านขวามือของแผ่นดีวีดี โปรแกรมก็จะกลับสู่หน้าหลักของการเลือกโหมด

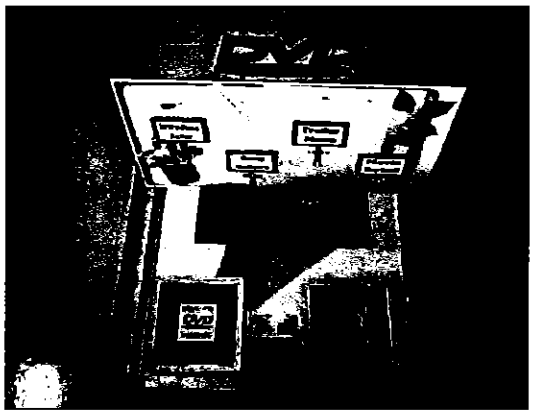


6. เมื่อต้องการดูการแสดงผลในโหมดต่อไป สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านซ้ายมือของแผ่นดีวีดีในลักษณะเดียวกับการเข้าไปดูโหมดแรก ซึ่งจะเห็นว่าโมเดลสามมิติของโหมด Trailer Movie หมุนรอบตัวเอง เพื่อแสดงสถานะว่ากำลังจะแสดงผลในโหมดนี้

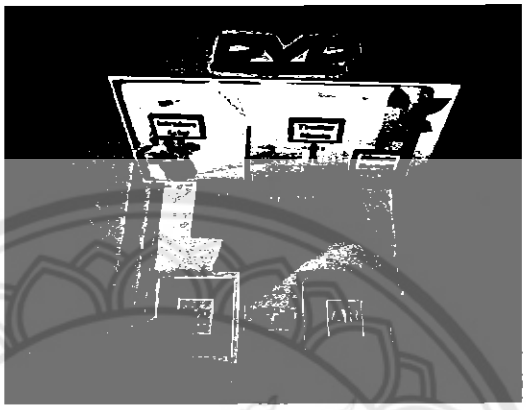


7. แสดงผลการทำงานของโหมด Trailer Movie ซึ่งจะเป็นการแสดงตัวอย่างภาพยนตร์

8. เมื่อต้องการกลับไปยังหน้าแรก สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านขวามือของแผ่นดีวีดี โปรแกรมก็จะกลับสู่หน้าหลักของการเลือกโหมด

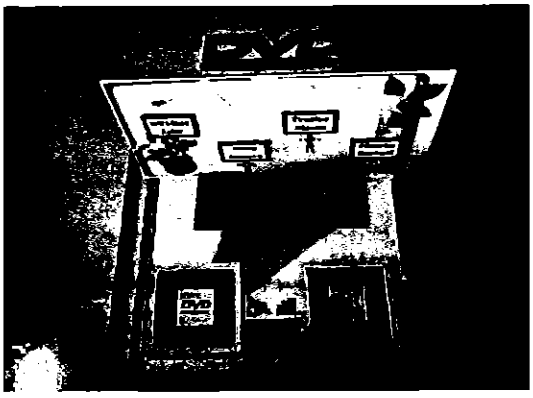


9. เมื่อต้องการดูการแสดงผลในโหมดต่อไป สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านซ้ายมือของแผ่นดีวีดีในลักษณะเดียวกับการเข้าไปดูโหมดแรก ซึ่งจะเห็นว่าโมเดลสามมิติของโหมด Song & Gallery หมุนรอบตัวเอง เพื่อแสดงสถานะว่ากำลังจะแสดงผลในโหมดนี้

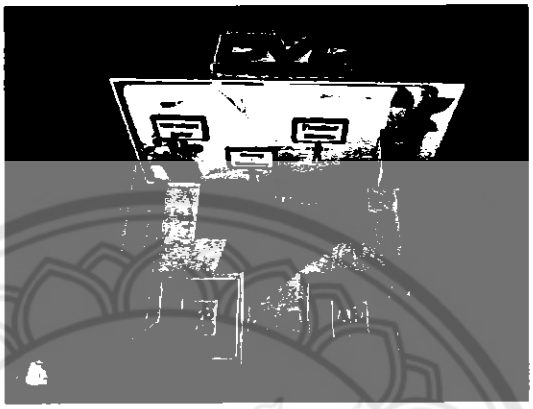


10. แสดงผลการทำงานของโหมด Song & Gallery ซึ่งจะเป็นการแสดงเพลงประกอบภาพยนตร์ และรูปภาพนักแสดงในเรื่อง

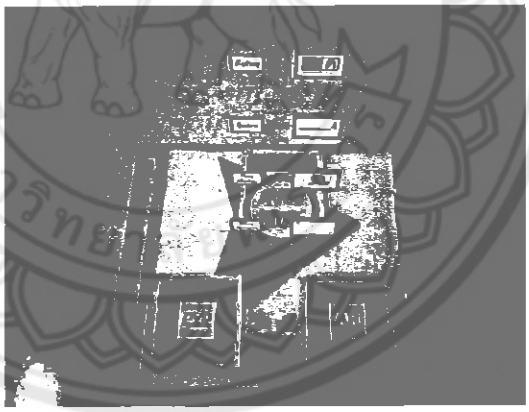
11. เมื่อต้องการกลับไปยังหน้าแรก สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านขวามือของแผ่นดีวีดี โปรแกรมก็จะกลับสู่หน้าหลักของการเลือกโหมด



12. เมื่อต้องการดูการแสดงผลในโหมดต่อไป สามารถทำได้โดยใช้มือปัดวัตถุสัญลักษณ์ทางด้านซ้ายมือของแผ่นดีวีดีในลักษณะเดียวกับการเข้าไปดูโหมดแรก ซึ่งจะเห็นว่าโมเดลสามมิติของโหมด Movie Detail หมุนรอบตัวเอง เพื่อแสดงสถานะว่ากำลังจะแสดงผลในโหมดนี้



13. แสดงผลการทำงานของโหมด Movie Detail ซึ่งจะเป็นการแสดงรายละเอียดของภาพยนตร์ เช่น เรื่องย่อ ผู้สร้าง ประเภทของภาพยนตร์ เป็นต้น

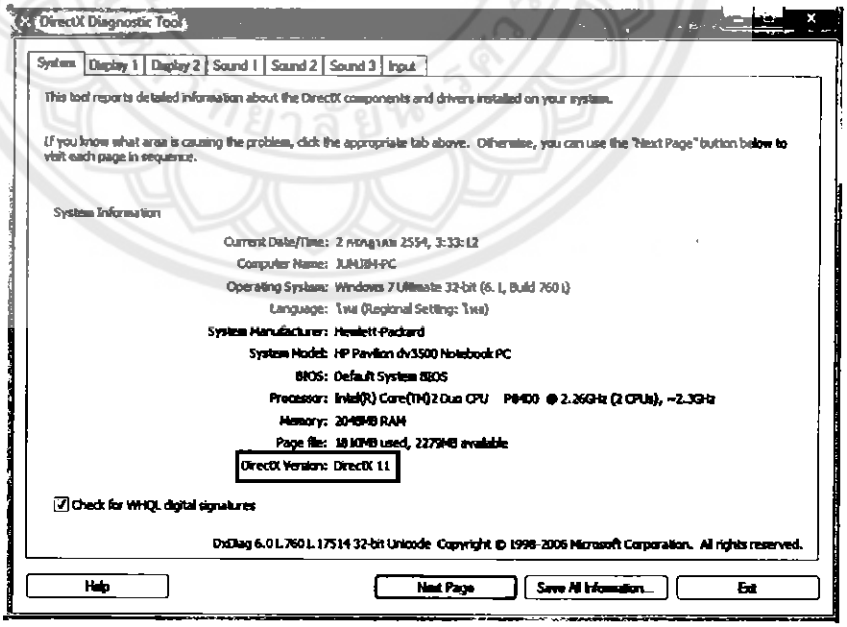


14. เมื่อต้องการกลับไปยังหน้าแรก สามารถทำได้โดยใช้มือปิดวัตถุสัญลักษณ์ทางด้านขวามือของแผ่นดีวีดี โปรแกรมก็จะกลับสู่หน้าหลักของการเลือกโหมด

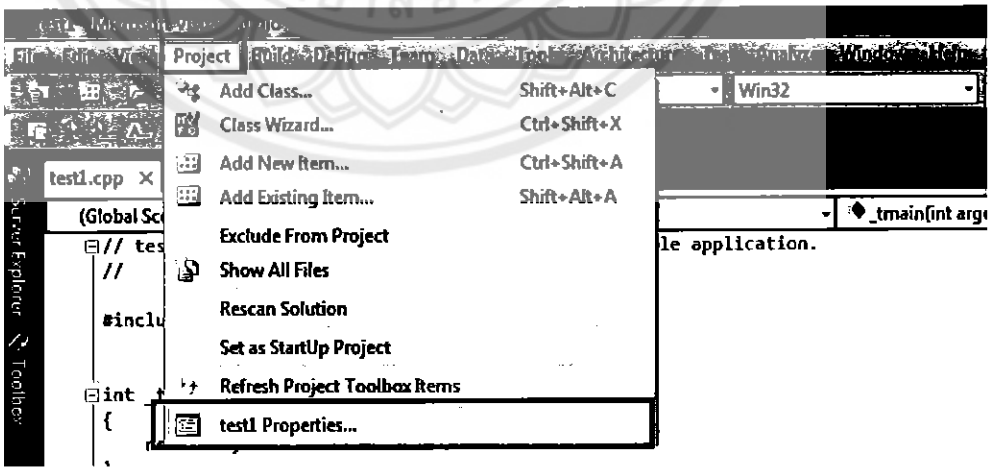
15. เมื่อต้องการกลับไปดูในโหมดแรกอีกครั้ง สามารถทำตามลำดับขั้นตอนตั้งแต่ข้อ 3 เป็นต้นมาได้

การติดตั้ง ARToolKit และการ Test กล้อง

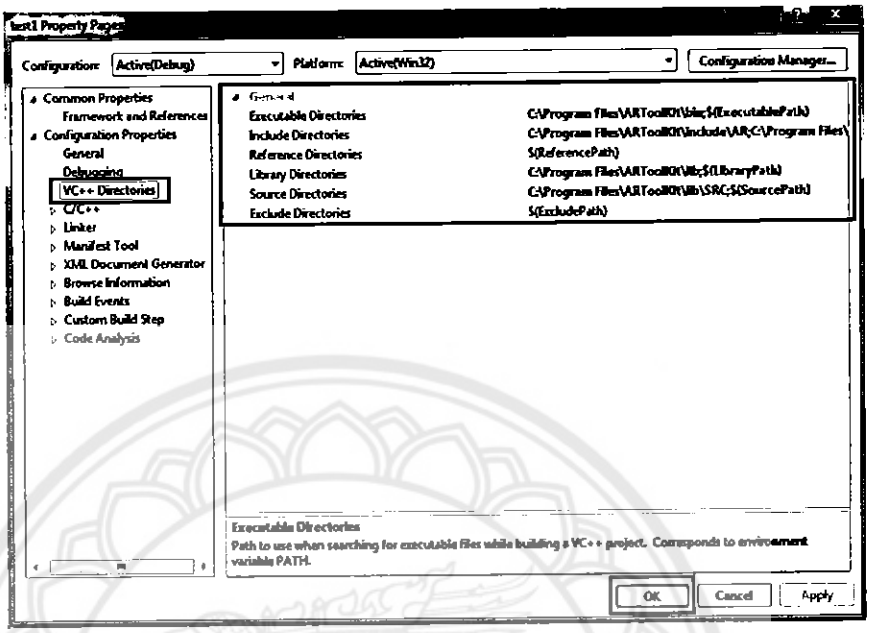
1. download Microsoft Visual Studio 2010 หรือ version ไหนก็ได้ และ installให้เรียบร้อย
2. download DSVL-0.0.8b.zip จาก http://sourceforge.net/projects/artoolkit/files/artoolkit/ (download มาเตรียมไว้ก่อน)
3. download GLUT จาก http://www.xmission.com/~nate/glut.html เลือก glut-3.7.6-bin.zip (117 KB) กด 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
4. เอาไฟล์ MSVCP71D.dll และ MSVCR71D.dll ที่ส่งไปให้ ไปไว้ที่ c:\windows\system32
5. ตรวจสอบเวอร์ชั่น ของ directX ที่มีในเครื่องของเราโดย start >> run >>dxdiag >>OK directX version จะต้องมากกว่า 9.0b ขึ้นไป



6. ต่อสายกล้อง ทดสอบกล้องว่าใช้งานได้ป่าว
7. download ARToolKit แตกไฟล์ ไว้ที่สำหรับใช้งาน เช่น C:\Program Files\ARToolKit
8. แตกไฟล์ DSVL-0.0.8b.zip ที่โหลดมา ใส่ไว้ใน C:\Program Files\ARToolKit ที่เราเพิ่งแตกเมื่อกี้ (ข้อ7) (Make sure that the directory is named "DSVL".)
9. Copy the files DSVL.dll and DSVLd.dll จาก C:\Program Files\ARToolKit\DSVL\bin ไปไว้ที่ C:\Program Files\ARToolKit\bin
10. รันสคริปต์ C:\Program Files\ARToolKit\Configure.win32.bat
11. เปิด microsoft visual studio เลือก open project เลือก ARToolKit (type= Microsoft Visual Studio Solution Object) จาก C:\Program Files\ARToolKit
12. Build >> Build Solution เสร็จต้อง build succeeded ,0 failed
13. หมายเหตุ ถ้า Build ไม่สำเร็จ ให้ทำดังนี้
    13.1 เปิด microsoft visual studio ขึ้นมา เลือก new project ขึ้นมา
    13.2 Copy Code จากไฟล์ simpleLite.c ใน ARToolKit (type= Microsoft Visual Studio Solution Object) มาวางที่ project ของเรา
    13.3 Set ค่าใน microsoft visual studio ดังนี้
        13.3.1 ไปที่ Project >> ชื่อไฟล์ Properties...

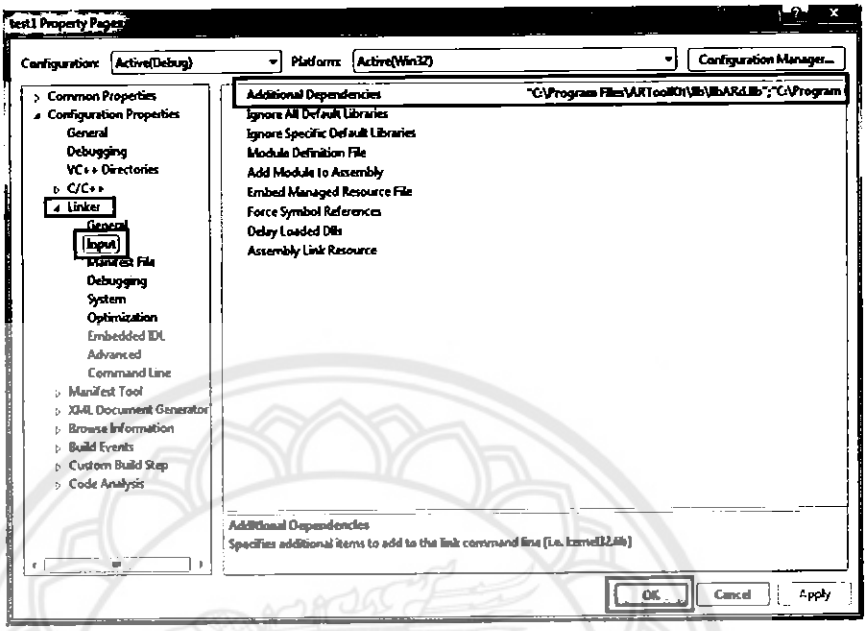


13.3.2 เลือก VC++ Directories และทำการเพิ่มข้อมูลในหน้าต่างด้านขวาดังนี้



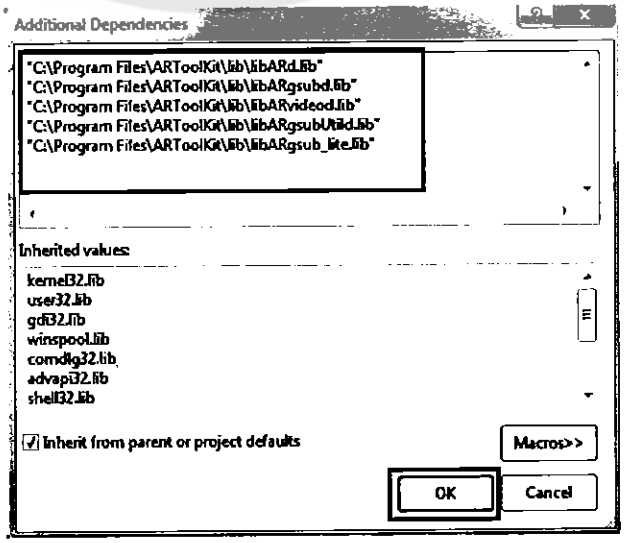
- Executable Directories : C:\Program Files\ARToolKit\bin
- Include Directories : C:\Program Files\ARToolKit\include\AR และ C:\Program Files\ARToolKit\include
- Reference Directories : เหมือนเดิม
- Library Directories : C:\Program Files\ARToolKit\lib
- Source Directories : C:\Program Files\ARToolKit\lib\SRC
- Exclude Directories : เหมือนเดิม

13.3.3 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



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

15. จากนั้นก็ Build >> Build Solution โปรเจ็คของเรา แล้วก็รัน ทดลองเล่น AR ได้

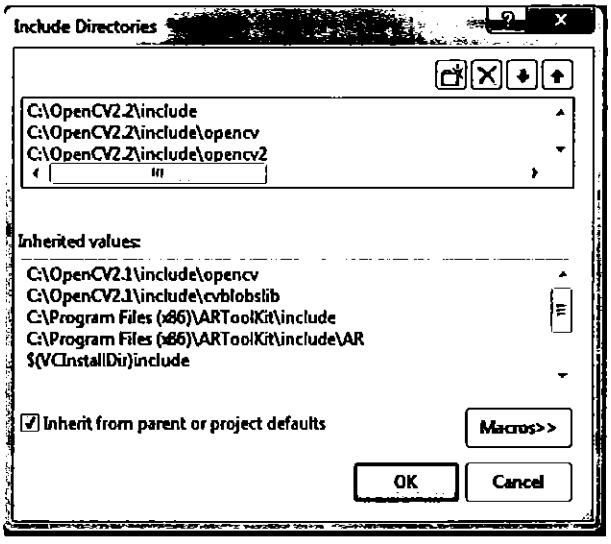
16. สามารถปริ้นตัว marker ได้ที่ C:\Program Files\ARToolKit\patterns

การติดตั้ง irrKlang

1. เข้าไปที่ C/C++ -> General -> Additional Iclude Directories แล้ว add
   - C:\irrKlang-1.3.0b\irrKlang-1.3.0\include
   - C:\irrKlang-1.3.0b\irrKlang-1.3.0\plugins\ikpMP3
   - C:\irrKlang-1.3.0b\irrKlang-1.3.0\plugins\ikpMP3\decoder
2. เข้าไปที่ Linker -> General -> Additional Library Directories แล้ว add
   - C:\irrKlang-1.3.0b\irrKlang-1.3.0\lib\Win32-visualStudio

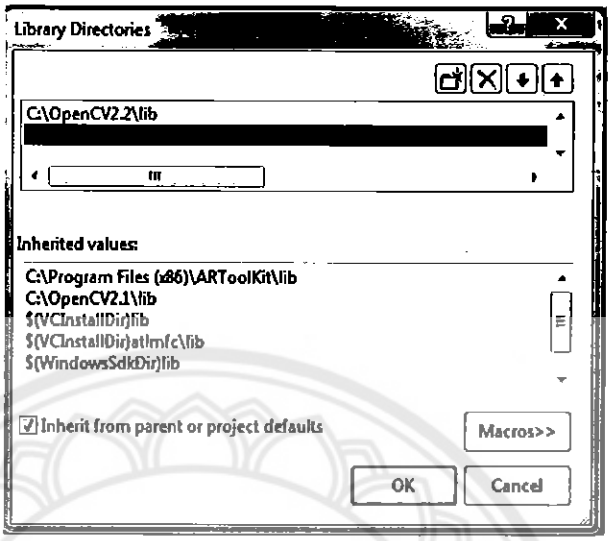
การติดตั้ง OpenCV

1. เข้าไปที่ Project -> Property -> VC++ Directories -> Include Directories แล้ว add
   - C:\OpenCV2.2\include
   - C:\OpenCV2.2\include\opencv
   - C:\OpenCV2.2\include\opencv2



2. เข้าไปที่ Project -> Property -> VC++ Directories -> Library Directories แล้ว add

   - C:\OpenCV2.2\lib



3. เข้าไปที่ Project -> Property -> Linker -> Input -> Additional Dependencies แล้ว add

   - C:\OpenCV2.2\lib\opencv_core220d.lib
   - C:\OpenCV2.2\lib\opencv_highgui220d.lib
   - C:\OpenCV2.2\lib\opencv_video220d.lib
   - C:\OpenCV2.2\lib\opencv_ml220d.lib
   - C:\OpenCV2.2\lib\opencv_legacy220d.lib
   - C:\OpenCV2.2\lib\opencv_imgproc220d.lib