

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง

ระบบสนับสนุนการเรียนรู้การสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล
LEARNING SUPPORT SYSTEM FOR 3D MODEL CREATION
USING OPENGL



T144242

โดย



วิฑูรย์ จันทร์สว่าง

WITHOON CHANSAWANG

อาจารย์ที่ปรึกษา

ดร.สุภวรรณ อ้นนันทน์

7574
2557

6-00264 260

เลขหมู่.....
เลขทะเบียน..... 144242
วัน,เดือน,ปี 10 9 2559

b. 12846966
i.....

รายงานนี้เป็นส่วนหนึ่งของวิชาโครงการพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 1 ปีการศึกษา 2557

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**LEARNING SUPPORT SYSTEM FOR 3D MODEL CREATION
USING OPENGL**



WITHOON CHANSAWANG

**A REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF THE COURSE
INDEPENDENT STUDY 2**

1/ 2014

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



COPYRIGHT 2014

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใบรับรองการศึกษาอิสระ 2 (Independent Study 2)

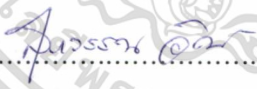
เรื่อง

ระบบสนับสนุนการเรียนการสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล
Learning Support System for 3D model Creation Using OpenGL


นายวิฑูรย์ จันทร์สว่าง

รหัสประจำตัว 53660564

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด
รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของการศึกษาวิชาการศึกษาอิสระ 2 หลักสูตรวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ) ภาควิชาการศึกษาอิสระ 2 หลักสูตรวิทยาศาสตรมหาบัณฑิต (เทคโนโลยีสารสนเทศ) ภาคเรียนที่ 1 ปีการศึกษา 2557

.....อาจารย์ที่ปรึกษา
(ดร.สุภวรรณ อ้นนันทน์)

.....กรรมการสอบ
(ดร.มานพ พันธุ์โลกกรวด)

.....กรรมการสอบ
(ดร.สิงหะ นวีสุข)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หัวข้อ	ระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล
นักศึกษา	นายวิฑูรย์ จันทร์สว่าง
รหัสนักศึกษา	53660564
ปริญญา	วิทยาศาสตรมหาบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
แขนงวิชา	วิทยาการสารสนเทศ
ปีการศึกษา	2557
อาจารย์ที่ปรึกษา	ดร.สุภวรรณ อันนันทน์

บทคัดย่อ

ในปัจจุบันการพัฒนางานจำลองสามมิติ (3D) กำลังได้รับความนิยมในด้านต่างๆ เช่น สื่อคอมพิวเตอร์ การออกแบบสินค้า หรือตลอดจนการก่อสร้างแบบจำลองต่างๆ การที่นักพัฒนาจะสร้างงานสามมิติให้คุณภาพได้นั้น มีความจำเป็นต้องเรียนรู้พื้นฐานเสียก่อน เพื่อที่จะได้นำไปประยุกต์ใช้กับงานขั้นสูงได้อย่างราบรื่น วิทยานิพนธ์ฉบับนี้จึงได้รวบรวมเอาเทคนิคพื้นฐานและทฤษฎีของการสร้างรูปทรงสามมิติเอาไว้เป็นสื่อการเรียนการสอน แบบปฏิสัมพันธ์กับผู้เรียน (Interactive) เพื่อให้ผู้เรียนได้ทำการเรียนรู้ไปพร้อมกับตัวอย่างงานสามมิติแต่ละรูปแบบ เช่น การสร้างรูปทรงพื้นฐานแต่ละ สีและการให้แสง ตลอดจนสมการที่เปลี่ยนแปลงไปเมื่อรูปทรงพื้นฐานมีการเปลี่ยนแปลงรูปแบบไปจากเดิม โดยผู้เขียนหวังว่าโครงการพัฒนาระบบฉบับนี้จะเป็นประโยชน์ต่อการเรียนการสอนรายวิชาคอมพิวเตอร์กราฟฟิกได้เป็นอย่างดี

Title	Learning support system for 3D model creation using openGL
Student	Mr. Withoon Chansawang
Student ID.	53660564
Degree	Master of Science
Program	Information Technology
Major	Information System Technology
Academic Year	2014
Advisor	Dr. Supawan Aunnunup

ABSTRACT

In currently time, the development in 3D are gaining popularity in social such as animation, design products by use 3D software or simulation. Before developer produces advance product in 3D. They necessary learns the basic of 3D for applied to advance process in computer graphic smoothly. This thesis included collection of basic technique and theories of 3D development such as basic 3D model, various object Transformation and lighting technique to 3D model. Student will be fast learning in computer graphic by use this thesis and can see the interactive lesson for quick to clear mores. I hope this reports of independent study will be full fill completed in computer graphic course.

สารบัญ

หน้า

บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
สารบัญ.....	III
สารบัญตาราง.....	V
สารบัญรูป.....	VI
บทที่ 1 บทนำ	
1.1 ความเป็นมาและความสำคัญของปัญหา.....	1
1.2 ความมุ่งหมายและวัตถุประสงค์ของการศึกษา.....	1
1.2 สมมติฐานของการศึกษา.....	1
1.3 ทฤษฎีหรือแนวคิดที่ใช้ในการศึกษา.....	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ.....	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	
2.1 ความรู้เกี่ยวกับโอเพนจีแอล.....	3
2.10 ชุดคำสั่งสำหรับสร้างโมเดลสามมิติใน โอเพนจีแอล	3
2.11 การแปลงรูปทรงแบบสามมิติและเมทริกซ์	9
2.12 GLUI (Graphic Library User Interface).....	13
2.13 แสง.....	13
2.20 การติดตั้ง OpenGL เพื่อทดสอบการทำงาน.....	18
บทที่ 3 การวิเคราะห์และออกแบบระบบ	
3.1 การออกแบบระบบ.....	19
3.2 การทำงานของระบบ.....	21
3.3 แบบจำลองเชิงแนวคิดของระบบ.....	26
บทที่ 4 การพัฒนาและทดสอบระบบงาน	
4.1 เครื่องมือที่ใช้ในการพัฒนาระบบ.....	44
4.2 รายละเอียดโปรแกรมที่ได้พัฒนาในเชิงเทคนิค.....	44

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ(ต่อ)

หน้า

บทที่ 5

5.1 ผลการทดลอง.....60

5.2 การประยุกต์ใช้งาน.....61

บรรณานุกรม.....62



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญตาราง

ตารางที่	หน้า
2.1 พารามิเตอร์ของแหล่งกำเนิดแสงและค่าเริ่มต้น.....	18
3.1 Use Case Diagram ของส่วน Create 3D Objects.....	27
3.2 Use Case Diagram ของส่วน Change Attribute.....	28
3.3 Use Case Diagram ของส่วน Translated Transformation.....	29
3.4 Use Case Diagram ของส่วน Rotated Transformation.....	30
3.5 Use Case Diagram ของส่วน Scaling Transformation.....	31
3.6 Use Case Diagram ของส่วน Shearing Transformation.....	32
3.7 Use Case Diagram ของส่วน Reflect Transformation.....	33
3.8 Use Case Diagram ของส่วน See Result of Matrix Transformation.....	34
3.9 Use Case Diagram ของส่วน Render Light to Object.....	35



สารบัญรูป

รูปที่	หน้า
2.1 รูปทรงกลมแบบทึบและแบบโปร่ง.....	4
2.2 ตัวอย่างรูปทรงกลม.....	4
2.3 ตัวอย่างรูปทรงลูกบาศก์.....	5
2.4 ตัวอย่างรูปทรงกรวย.....	5
2.5 ตัวอย่างรูปทรงโดนัท.....	6
2.6 ตัวอย่างรูปทรงสี่เหลี่ยมด้านไม่เท่า.....	7
2.7 ตัวอย่างรูปทรงแปดเหลี่ยม.....	7
2.8 ตัวอย่างรูปทรงสิบสองเหลี่ยม.....	8
2.9 ตัวอย่างรูปทรงยี่สิบเหลี่ยม.....	8
2.10 ตัวอย่างรูปทรงกาน้ำชา.....	9
2.11 ตัวควบคุมของ GLUI.....	13
2.12 ทรงกลมที่ถูกให้แสงตามทฤษฎีของฟงและทรงกลมปกติ.....	15
2.13 ประเภทของการสะท้อนตามทฤษฎีของฟง.....	15
2.14 การตั้งค่า Enable Light Source.....	16
2.15 การตั้งค่า Setup Light Position and Intensity.....	16
2.16 การตั้งค่า Setup Light Type for Object.....	17
2.18 การตั้งค่า Setup to Material.....	17
3.1 องค์ประกอบของของระบบกรณีศึกษาของการสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล.....	20
3.2 Context Flow Diagram ของระบบ.....	20
3.3 Dataflow Diagram ของระบบการสร้างอ็อบเจกต์สามมิติ.....	21
3.4 Dataflow Diagram ของระบบการแปลงรูปทรงสามมิติ.....	22
3.5 Dataflow Diagram ของระบบควบคุมวัตถุแบบจีแอลยูไอ.....	24
3.6 Dataflow Diagram ของระบบควบคุมการให้แสงต่อวัตถุ (Light&Lighting Control).....	25

สารบัญรูป(ต่อ)

รูปที่	หน้า
3.7 Use Case Diagram ของกรณีศึกษาการสร้าง โมเดลสามมิติโดยใช้โอเพนจีแอล.....	27
3.8 Class Diagram ของกรณีศึกษาการสร้าง โมเดลสามมิติโดยใช้โอเพนจีแอล.....	36
3.9 Sequence Diagram Create New Object.....	40
3.10 Sequence Diagram Select Lesson	40
3.11 Sequence Diagram Transformation Object.....	41
3.12 Sequence Diagram Render Light for Object.....	42
4.1 หน้าจอสำหรับเลือกบทเรียนของระบบกรณีศึกษา โมเดลสามมิติโดยใช้โอเพนจีแอล.....	45
4.2 หน้าจอสำหรับบทเรียนเกี่ยวกับรูปทรง (GLUT 3D Models).....	46
4.3 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเกี่ยวกับรูปทรง (GLUT 3D Models).....	47
4.4 หน้าจอสำหรับบทเรียนเกี่ยวกับการย่อหรือขยายรูปทรง (Scaling Transformation).....	48
4.5 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับย่อหรือขยายรูปทรง.....	49
4.6 หน้าจอสำหรับบทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติ.....	50
4.7 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติ.....	51
4.8 หน้าจอสำหรับบทเรียนเกี่ยวกับการหมุนรูปทรงสามมิติ (Rotated Transformation).....	52
4.9 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับการหมุนรูปทรงสามมิติ (Rotated Transformation).....	53
4.10 หน้าจอสำหรับบทเรียนเกี่ยวกับการสะท้อนรูปทรงสามมิติ (Reflect Transformation).....	54
4.11 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับการสะท้อนรูปทรงสามมิติ (Reflect Transformation).....	55
4.12 หน้าจอสำหรับบทเรียนเกี่ยวกับเฉือนของรูปทรงสามมิติ (Shearing Transformation).....	56
4.13 หน้าจอหน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับเฉือนของรูปทรงสามมิติ (Shearing Transformation).....	57

สารบัญรูป(ต่อ)

รูปที่	หน้า
4.14 หน้าจอสำหรับบทเรียนเกี่ยวกับแสงและการให้แสงของรูปทรงสามมิติ (Light and Lighting)	58
4.15 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับแสงและการให้แสงของรูปทรงสามมิติ.....	59



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

ในปัจจุบันการการพัฒนางานจำลองสามมิติ (3D) กำลังได้รับความนิยมในด้านต่างๆ เช่น สื่อคอมพิวเตอร์ แอนิเมชัน การออกแบบบรรจุภัณฑ์สินค้าด้วยซอฟต์แวร์ประเภทสามมิติ หรือ ตลอดจนการสร้างแบบจำลองต่างๆ การที่นักพัฒนาจะสร้างงานสามมิติให้คุณภาพได้นั้น มีความจำเป็นต้องเรียนรู้พื้นฐานของงานสามมิติเสียก่อน เพื่อที่จะได้นำไปประยุกต์ใช้กับงานขั้นสูงได้อย่างราบรื่น แต่การเรียนรู้พื้นฐานของสร้างงานสามมิติในปัจจุบันเป็นการยากที่ผู้เรียนจะเข้าใจหากเรียนรู้โดยที่ไม่มีตัวอย่างอธิบายที่ชัดเจน วิทยานิพนธ์ฉบับนี้จึงได้รวบรวมเอาเทคนิคพื้นฐานและทฤษฎีของการสร้างรูปทรงสามมิติเอาไว้เป็นสื่อการเรียนการสอน แบบปฏิสัมพันธ์กับผู้เรียน (Interactive) เพื่อให้ผู้เรียนได้ทำการเรียนรู้ไปพร้อมกับตัวอย่างในงานสามมิติแต่ละรูปแบบ เช่น การสร้างรูปทรงพื้นฐาน การแปลงรูปภาพแบบสามมิติ สีและการให้แสงแก่อบเจ็กต์ ตลอดจนผลลัพธ์ของรูปภาพที่แสดงออกมาทางหน้าจอแบบทันที

1.2 วัตถุประสงค์ของการศึกษา

- 1.2.1 เพื่อศึกษาการพัฒนาโปรแกรมแบบสามมิติ โดยประยุกต์ใช้ไลบรารีของโอเพนจีแอล
- 1.2.2 เพื่อศึกษาโมเดล การแปลงรูปทรงและการให้แสงกับวัตถุในพิกัดฉากแบบสามมิติ
- 1.2.3 เพื่อศึกษาการแปลงเมตริกซ์และสมการ จากการแปลงรูปทรง เช่น การเคลื่อนย้ายวัตถุ (Translation), การหมุนวัตถุ (Rotation) และการย่อขยายวัตถุ (Scaling) จากไลบรารีของโอเพนจีแอล
- 1.2.4 เพื่อศึกษาทฤษฎีการให้แสงแบบต่างๆแก่วัตถุ โดยใช้ทฤษฎีแสงของฟง (Phong Model)

1.3 ประโยชน์ที่คาดว่าจะได้รับ

- 1.3.1 ได้เรียนรู้และพัฒนาบทเรียนเพื่อไว้ใช้ศึกษาการสร้าง โมเดลประเภทสามมิติ
- 1.3.2 ผู้ใช้งานสามารถเรียนรู้พื้นฐานของการสร้างงานด้านสามมิติจากบทเรียนที่สร้างขึ้นมาได้อย่างรวดเร็ว

1.3.3 นำความรู้ที่ได้ไปใช้ในอนาคตเพื่อสนับสนุนด้านการพัฒนาโปรแกรมแบบ 3 มิติ

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.4 เครื่องมือที่ใช้ในการพัฒนา

1.4.1 ระบบปฏิบัติการ Window 7 Starter หรือ ระบบปฏิบัติการที่รองรับ OpenGL

1.4.2 Visual Studio C++ Express Edition 2010

1.4.3 OpenGL Library สำหรับ C++

1.5 ขั้นตอนการดำเนินงาน

1.5.1 ศึกษาเอกสารและตำราเกี่ยวกับการพัฒนา โปรแกรมแบบสามมิติ

1.5.2 ออกแบบบทเรียนสามมิติ โดยประยุกต์ใช้กราฟในพิกัดฉากสามมิติ เช่น การสร้างออบเจ็กต์ แอตทริบิวต์ของออบเจ็กต์ การเปลี่ยนแปลงของสมการเมื่อมีการแปลงรูปทรง การให้สีและแสงในกรณีต่างๆ

1.5.3 สร้างแอปพลิเคชันการทำงานแบบปฏิสัมพันธ์กับผู้ใช้บนพื้นฐานของโอเพนจีแอลและจีแอลยูไอ (GLUI)

1.5.4 สร้างแบบจำลองรูปทรงสามมิติ เช่น การสร้างรูปทรงพื้นฐาน แอตทริบิวต์ต่างๆของรูปทรง ตัวอย่างสมการที่มีการเปลี่ยนแปลงไปเมื่อมีการแปลงรูปทรง การให้แสงต่อวัตถุและผลลัพธ์ของโมเดลเมื่อมีการให้แสง

1.5.5 ตรวจสอบความถูกต้องของซอฟต์แวร์ที่สร้างและทฤษฎีที่เกี่ยวข้อง

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในปัจจุบันการศึกษาเกี่ยวกับมัลติมีเดียและคอมพิวเตอร์กราฟฟิกแบบสามมิติแพร่หลายเป็นอย่างมาก แต่ยังคงขาดสื่อการเรียนการสอนที่มีคุณภาพเนื่องจากการเรียนรู้บทเรียนที่เกี่ยวกับโมเดลสามมิตินั้น สื่อต่างๆไม่ว่าจะเป็นสไลด์ กระดานดำ หรือคอมพิวเตอร์สามารถแสดงผลได้เพียงสองมิติ ผู้เขียนจึงพัฒนาโปรแกรมสื่อการสอนที่ประกอบด้วยเนื้อหาเกี่ยวกับการสร้างรูปทรงสามมิติ การแปลงรูปทรงและการให้แสงกับวัตถุโดยใช้โอเพนจีแอล ซึ่งภายในจะประกอบด้วยเนื้อหาและสื่อแบบอินเตอร์แอคทีฟที่แสดงผลออกมาในรูปของโมเดลสามมิติ เพื่อให้ผู้เรียนได้ศึกษาไปพร้อมกับการทดลองผลด้วยตัวเอง ซึ่งในบทที่ 2 นี้จะกล่าวถึงทฤษฎีของการสร้างโมเดลสามมิติและการใช้งานโอเพนจีแอลที่ใช้ในการสร้างระบบดังกล่าว

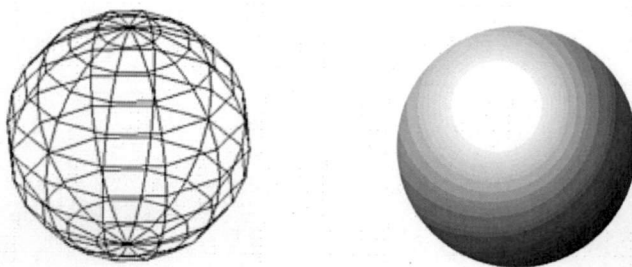
2.1 ความรู้เกี่ยวกับโอเพนจีแอล

โอเพนจีแอล หรือในภาษาอังกฤษ OpenGL ย่อมาจาก Open Graphic Library ซึ่งเป็นไลบรารีประกอบด้วยชุดคำสั่งมากมายในการสร้างโมเดลสามมิติ การแปลงรูปทรง ตลอดจนการให้แสงแก่โมเดลสามมิติ โดยในปัจจุบันโอเพนจีแอลเป็นที่นิยมมากในอุตสาหกรรมภาพยนตร์อนิเมชันแบบสามมิติ การประมวลผลภาพและวีดีโอเกมส์ นอกจากนี้โอเพนจีแอลยังไม่ยึดติดกับระบบปฏิบัติการและสามารถทำงานได้ในหลายแพลตฟอร์ม

โอเพนจีแอลถูกกำหนดด้วยมาตรฐานโดยมีคณะกรรมการร่วมกันกำหนด หรือ OpenGL Architecture Review Board(ARB) ประกอบด้วยบริษัท 3D Labs, Apple, ATI, Dell, Evans & Sutherland, HP, IBM, Intel, Matrox, NVIDIA, SGI, และ Sun Microsystems โดยจะมีหน้าที่ควบคุมมาตรฐานของชุดคำสั่งเพื่อให้ไลบรารีมีมาตรฐานสูงสุด

2.2 ชุดคำสั่งสำหรับสร้างโมเดลสามมิติในโอเพนจีแอล

โอเพนจีแอลจะไม่มีคำสั่งในการสร้างรูปทรงที่ซับซ้อน เช่น รถยนต์ หรือเครื่องบิน แต่จะมีคำสั่งสร้างรูปทรงพื้นฐานเพื่อให้ผู้ใช้งานไปประกอบกันหรือทำการแปลงเพื่อสร้างเป็นรูปทรงที่ซับซ้อน รูปทรงพื้นฐานในโอเพนจีแอลจะประกอบด้วยกัน 2 ชนิด คือรูปทรงแบบทึบและแบบโปร่งหรือแบบโครงลวดดังรูปภาพที่ 2.1 โดยเราสามารถสร้างรูปทรงได้ดังโดยใช้คำสั่งต่อไปนี้



รูปที่ 2.1 รูปทรงกลมแบบทึบและแบบโปร่ง

2.2.1 รูปทรงกลม (3D Sphere)



รูปที่ 2.2 ตัวอย่างรูปทรงกลม

คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- void glutSolidSphere(GLdouble radius, GLint slices, GLint stacks);

- void glutWireSphere(GLdouble radius, GLint slices, GLint stacks);

พารามิเตอร์

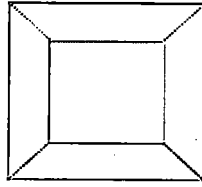
- GLdouble radius คือ ค่ารัศมีของรูปทรงกลม

- GLint slices คือ จำนวนเส้นรุ้งโดยรอบของรูปทรงกลม

- GLint stacks คือ จำนวนเส้นแวงของรูปทรงกลม

ตัวอย่างการนำไปใช้ เช่น glutWireSphere(5,10,10); หรือ glutSolidSphere(5,10,10); จะได้ผลลัพธ์รูปทรงกลมดังภาพ

2.2.2 รูปทรงลูกบาศก์(3D Cube)



รูปที่ 2.3 ตัวอย่างรูปทรงลูกบาศก์

คำสั่งที่ใช้ในการสร้างโมเดลสามมิติได้แก่

- void glutSolidCube(GLdouble size);
- void glutWireCube(GLdouble size);

พารามิเตอร์

- GLdouble size คือ ขนาดของรูปทรงลูกบาศก์

ตัวอย่างการนำไปใช้ เช่น glutWireCube(1.0); หรือ glutSolidCube(1.0); จะได้ผลลัพธ์รูปทรงลูกบาศก์ดังภาพ

2.2.3 รูปทรงกรวย(3D Cone)



รูปที่ 2.4 ตัวอย่างรูปทรงกรวย

คำสั่งที่ใช้ในการสร้างโมเดลสามมิติได้แก่

- void glutSolidCone(GLdouble base, GLdouble height, GLint slices, GLint stacks);
- void glutWireCone(GLdouble base, GLdouble height, GLint slices, GLint stacks);

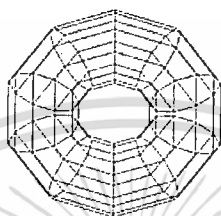
พารามิเตอร์

- GLdouble base คือ ขนาดความกว้างของฐานรูปทรงกรวย
- GLdouble height คือ ความสูงของรูปทรงกรวย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- GLint slices คือ จำนวนรอบวงของรูปทรงกรวย
 - GLint stacks คือ จำนวนเส้นแนวตั้งของรูปทรงกรวย
- ตัวอย่างการนำไปใช้ เช่น `glutWireCone(1,1,10,10)`; หรือ `glutSolidCone(1,1,10,10)`; จะได้ผลลัพธ์รูปทรงกรวยดังภาพ

2.2.4 รูปทรงโดนัท(3D Torus)



รูปที่ 2.5 ตัวอย่างรูปทรงโดนัท

คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- `void glutSolidTorus(GLdouble innerRadius, GLdouble outerRadius, GLint nsides, GLint rings);`
- `void glutWireTorus(GLdouble innerRadius, GLdouble outerRadius, GLint nsides, GLint rings);`

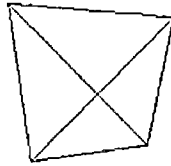
พารามิเตอร์

- `GLdouble innerRadius` คือ ระยะของเส้นรัศมีของวงกลมด้านในของรูปทรงโดนัท
- `GLdouble outerRadius` คือ ระยะเส้นรัศมีของวงกลมด้านนอกของรูปทรงโดนัท
- `GLint nsides` คือ จำนวนเส้นวงกลมแนวขวางของรูปทรงโดนัท
- `GLint rings` คือ จำนวนเส้นรอบวงจามแนวยาวของรูปทรงโดนัท

คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- ตัวอย่างการนำไปใช้ เช่น `glutWireTorus(0.5,1.5,10,10)`; หรือ `glutSolidTorus(0.5,1.5,10,10)` จะได้ผลลัพธ์รูปทรงโดนัทดังภาพ

2.2.5 รูปทรงสี่เหลี่ยมด้านไม่เท่า(3D Tetrahedron)



รูปที่ 2.6 ตัวอย่างรูปทรงสี่เหลี่ยมด้านไม่เท่า

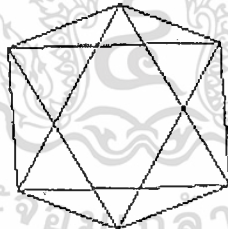
คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- void glutSolidTetrahedron(void);

- void glutWireTetrahedron(void);

ตัวอย่างการนำไปใช้ เช่น glutWireTetrahedron(); หรือ glutSolidTetrahedron(); จะได้ผลลัพธ์รูปทรงสี่เหลี่ยมด้านไม่เท่าดังภาพ

2.2.6 รูปทรงแปดเหลี่ยม(3D Octahedron)



รูปที่ 2.7 ตัวอย่างรูปทรงแปดเหลี่ยม

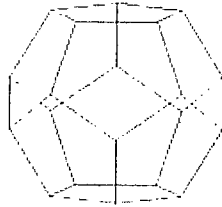
คำสั่งที่ใช้ในการสร้างโมเดลสามมิติได้แก่

- void glutSolidOctahedron(void);

- void glutWireOctahedron(void);

ตัวอย่างการนำไปใช้ เช่น glutWireOctahedron(); หรือ glutSolidOctahedron(); จะได้ผลลัพธ์รูปทรงแปดเหลี่ยมดังภาพ

2.2.7 รูปทรงสิบสองเหลี่ยม(3D Dodecahedron)



รูปที่ 2.8 ตัวอย่างรูปทรงสิบสองเหลี่ยม

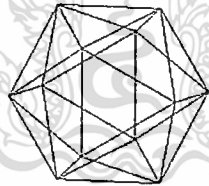
คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- void glutSolidDodecahedron(void);

- void glutWireDodecahedron(void);

ตัวอย่างการนำไปใช้ เช่น glutWireDodecahedron (); หรือ glutSolidDodecahedron (); จะได้ผลลัพธ์รูปทรงสิบสองเหลี่ยมดังภาพ

2.2.8 รูปทรงยี่สิบเหลี่ยม(3D Icosahedrons)



รูปที่ 2.9 ตัวอย่างรูปทรงยี่สิบเหลี่ยม

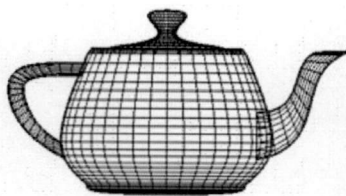
คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

void glutSolidIcosahedron(void);

void glutWireIcosahedron(void);

ตัวอย่างการนำไปใช้ เช่น glutWireIcosahedron (); หรือ glutSolidIcosahedron (); จะได้ผลลัพธ์รูปทรงยี่สิบเหลี่ยมดังภาพ

2.2.9 รูปทรงกาน้ำชา(3D Teapot)



รูปที่ 2.10 ตัวอย่างรูปทรงกาน้ำชา

คำสั่งที่ใช้ในการสร้าง โมเดลสามมิติได้แก่

- void glutSolidTeapot(GLdouble size);

- void glutWireTeapot(GLdouble size);

พารามิเตอร์

- GLdouble size คือ ขนาดของรูปทรงกาน้ำชา

ตัวอย่างการนำไปใช้ เช่น glutWireTeapot(1.0); หรือ glutSolidTeapot(1.0); จะได้ผลลัพธ์รูปทรงกาน้ำชาดังภาพ

2.3 การแปลงรูปทรงแบบสามมิติและเมทริกซ์ (3D Transformation and Matrix)

การแปลงรูปทรงในระบบสามมิติมีความสำคัญมากในการสร้างโมเดลที่มีความซับซ้อน โดยการแปลงโมเดลสามมิติในวิทยานิพนธ์เล่มนี้จะประกอบด้วยกรรมวิธีดังต่อไปนี้

2.3.1 การเคลื่อนย้ายอ็อบเจกต์ (Translate Transformation) คือ การเคลื่อนย้ายวัตถุไปตามแนวแกนสามมิติ X, Y และ Z เมื่อผู้ใช้ทำการสร้างอ็อบเจกต์ขึ้นมาในระบบ โอเพนจีแอลจะทำการสร้างอ็อบเจกต์ที่จุดออริจิน หรือจุดที่มีค่า $X=0, Y=0$ และ $Z=0$ ผู้ใช้จะต้องทำการเคลื่อนย้ายอ็อบเจกต์ไปไว้ในตำแหน่งที่ต้องการด้วยการเขียนคำสั่งเพื่อควบคุมอ็อบเจกต์ดังกล่าว โดยคำสั่งของโอเพนจีแอลที่ใช้ในการเคลื่อนย้ายอ็อบเจกต์ คือ

- glTranslatef (Gfloat x, Gfloat y, Gfloat z);

โดยที่ x คือทรานสเลชันเวกเตอร์ในแนวแกน x

y คือทรานสเลชันเวกเตอร์ในแนวแกน y

z คือทรานสเลชันเวกเตอร์ในแนวแกน z

โดยที่สามารถแสดงได้ดั่งเมทริกซ์ ของการเคลื่อนที่ได้ดังต่อไปนี้

กำหนดให้ x คือค่าของการเคลื่อนที่ที่เปลี่ยนแปลงไปตามแนวแกน X

y คือค่าของการเคลื่อนที่ที่เปลี่ยนแปลงไปตามแนวแกน Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

z คือค่าของการเคลื่อนที่ที่เปลี่ยนแปลงไปตามแนวแกน Z

$$\begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x+X.1 \\ y+Y.1 \\ z+Z.1 \\ 1 \end{bmatrix}$$

ตัวอย่างเช่น ทำการเคลื่อนที่อ็อบเจกต์ไปตามแนวแกน X 5 หน่วย เมทริกซ์ ๔x๔ ของอ็อบเจกต์จะมีการเปลี่ยนแปลงไปดังต่อไปนี้

$$\begin{bmatrix} 1 & 0 & 0 & 5 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 5 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} x+5 \\ y \\ z \\ 1 \end{bmatrix}$$

2.3.2 การย่อและขยายอ็อบเจกต์ (Scale Transformation) คือ เพิ่มขนาดหรือย่อขนาดของอ็อบเจกต์ไปตามแนวแกนสามมิติ ซึ่งผู้ใช้งานสามารถย่อขยายอ็อบเจกต์ตามแนวแกนใดๆ หรือขยายแบบขนานพร้อมกันทุกแกนก็ได้ โดยคำสั่งของ โอเพนจีแอลที่ใช้ในการย่อหรือขยายอ็อบเจกต์ คือ `glScalef (Gfloat x, Gfloat y, Gfloat z);`

โดยที่ x คือสเกลลิงแฟกเตอร์ในแนวแกน x

y คือสเกลลิงแฟกเตอร์ในแนวแกน y

z คือสเกลลิงแฟกเตอร์ในแนวแกน z

โดยที่สามารถแสดง ได้ดังเมทริกซ์ ของการย่อขยายได้ดังต่อไปนี้

กำหนดให้ SX คือ ค่าที่ย่อหรือขยายเปลี่ยนแปลงไปตามแนวแกน X

SY คือ ค่าที่ย่อหรือขยายเปลี่ยนแปลงไปตามแนวแกน Y

SZ คือ ค่าที่ย่อหรือขยายเปลี่ยนแปลงไปตามแนวแกน Z

$$\begin{bmatrix} SX & 0 & 0 & 0 \\ 0 & SY & 0 & 0 \\ 0 & 0 & SZ & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} SX.x \\ SY.y \\ SZ.z \\ 1 \end{bmatrix}$$

2.3.3 การหมุนอ็อบเจ็กต์ (Rotation Transformation) คือ การปรับตำแหน่งการวางอ็อบเจ็กต์ด้วยการหมุนไปตามแนวแกนสามมิติ ซึ่งผู้ใช้สามารถหมุนไปตามแกนที่ต้องการ หรือจะหมุนพร้อมกันในทุกแกนก็ได้ โดยคำสั่งของโอเพนจีแอลที่ใช้ในการย่อหรือขยายอ็อบเจ็กต์ คือ

glRotatef (GLfloat angle, GLfloat x, GLfloat y, GLfloat z);

โดยที่ angle คือมุมที่ต้องการให้หมุน (เป็นองศา)

x คือ กำหนดให้หมุนตามแนวแกน x

y คือ กำหนดให้หมุนตามแนวแกน y

z คือ กำหนดให้หมุนตามแนวแกน z

โดยที่สามารถแสดงได้ดังเมทริกซ์ของการหมุนได้ดังต่อไปนี้

เมทริกซ์ การหมุนตามแนวแกน X

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & \sin \theta & 0 \\ 0 & -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ \cos \theta \cdot y - \sin \theta \cdot z \\ \sin \theta \cdot y + \cos \theta \cdot z \\ 1 \end{bmatrix}$$

เมทริกซ์ การหมุนตามแนวแกน Y

$$\begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta \cdot x + \sin \theta \cdot z \\ y \\ -\sin \theta \cdot x + \cos \theta \cdot z \\ 1 \end{bmatrix}$$

เมทริกซ์ การหมุนตามแนวแกน Z

$$\begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta \cdot x - \sin \theta \cdot y \\ \sin \theta \cdot x + \cos \theta \cdot y \\ z \\ 1 \end{bmatrix}$$

2.3.4 การเฉือนอ็อบเจ็กต์ (Shear Transformation) คือ การเฉือนอ็อบเจ็กต์ไปให้มีรูปร่างบิดไปตามแนวแกนสามมิติ ซึ่งผู้ใช้สามารถบิดตามแนวแกนใดก็ได้ แต่โดยปกติแล้วมักจะบิดแกนใดแกนหนึ่ง ไม่นิยมบิดไปในทุกแกนเพราะจะควบคุมรูปทรงได้ยาก การแปลงด้วยการเฉือนจะไม่มีเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จุดคำสั่งใน โอเพนจีแอล แต่ผู้ใช้สามารถใช้เมทริกซ์ ๓ ในการแทนค่าของอ็อบเจ็กแทนจุดคำสั่งได้ดังต่อไปนี้

- เมทริกซ์ ของการเลื่อนอ็อบเจ็ก

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & a & 0 \\ 0 & 1 & b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

- เมทริกซ์ การคืนค่าดั้งเดิมของการเลื่อนอ็อบเจ็ก

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -a & 0 \\ 0 & 1 & -b & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

2.3.5 การสะท้อนอ็อบเจ็ก (Reflect Transformation) คือ การสร้างอ็อบเจ็กในระนาบด้านตรงข้ามกับอ็อบเจ็กที่สะท้อน คล้ายคลึงกับการสะท้อนบนกระจกเงา การแปลงด้วยการสะท้อนจะไม่มีจุดคำสั่งใน โอเพนจีแอล แต่ผู้ใช้สามารถใช้เมทริกซ์ ๓ ในการแทนค่าของอ็อบเจ็กแทนจุดคำสั่งได้ดังตัวอย่างต่อไปนี้

- เมทริกซ์ ของการสะท้อนอ็อบเจ็กระนาบแกน x-axis (Y=0)

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- เมทริกซ์ ของการสะท้อนอ็อบเจ็กระนาบแกน -xy-Plane

$$\begin{bmatrix} -1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- เมทริกซ์ ของการสะท้อนอ้อมเงิกระนาบแกน y-axis(x=0)

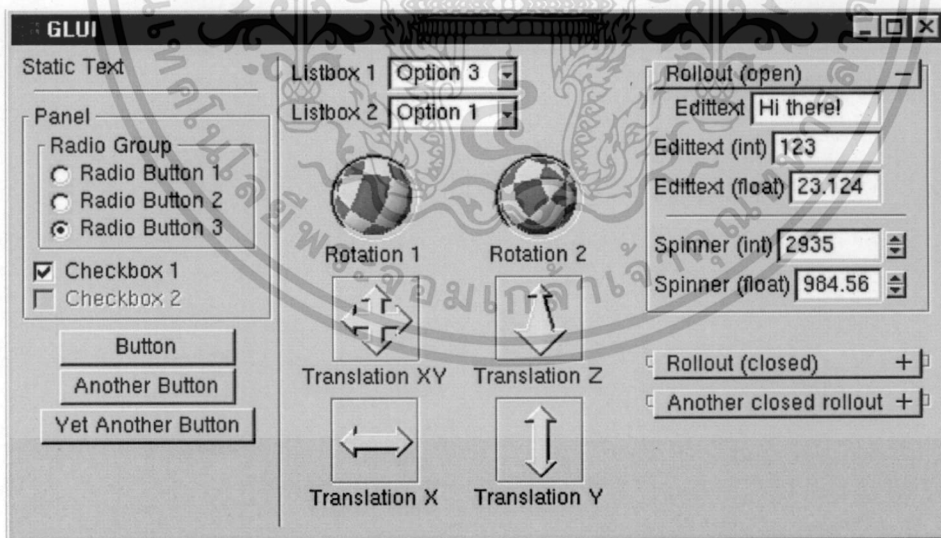
$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- เมทริกซ์ ของการสะท้อนอ้อมเงิกระนาบแกน xy-Plane

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

2.4 GLUT (Graphic Library User Interface)

GLUI เป็นส่วนติดต่อกับผู้ใช้ที่เราจะใช้ในการเชื่อมต่อกับ GLUT หรือส่วนประกอบของ วินโดว์ที่เราได้สร้างขึ้นมา เพื่อเอาไว้สำหรับควบคุมการแสดงผลโมเดลสามมิติที่สร้างขึ้นมา ถึงการ แปลงรูปทรง ตลอดจนการแสดงค่าของเมทริกซ์ ซึ่งที่เราใช้คำนวณ GLUT นอกจากนี้ยังสามารถ ติดต่อกับเมาส์ และคีย์บอร์ดได้อีกด้วย โดย GLUI จะประกอบด้วย Control ต่างดังภาพ



รูปที่ 2.11 ตัวอย่างควบคุมของ GLUI

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

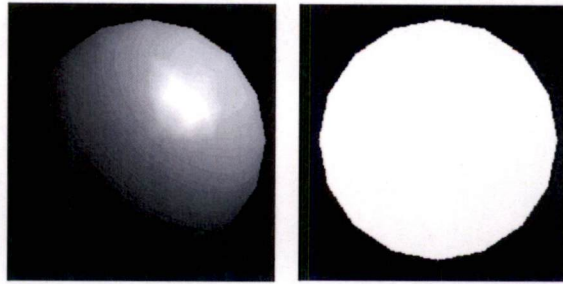
ตัวควบคุมแบบ GLUI จะประกอบด้วยตัวควบคุมดังต่อไปนี้

- ปุ่มกด(Button)
- กล่องกาช้บาค (Checkboxes) สำหรับตัวแปรแบบบูลีน (boolean variables)
- ปุ่มกดแบบวิทย์ (Radio Buttons) สำหรับตัวเลือกในลักษณะ mutually-exclusive
- กล่องข้อความแบบแก้ไขได้ (Editable text boxes)
- ตัวปั่น (Spinners) โดยมีการจัดการแบบตัวแปรจำนวนเต็มหรือทศนิยม
- บอลโค้ง (Arcball) ตัวควบคุมสำหรับการหมุน
- ตัวควบคุมการเลื่อน (Translation) ไปตามแนว x, y และ z
- กล่องรายการ (List boxes)
- สนามข้อความสถิต (Static text fields)
- แผ่นกระดาน (Panel) สำหรับรวมกลุ่มของตัวคอนโทรล
- หน้าปิดแบบม้วนกลับได้ (Rollouts (collapsible panels))
- เส้นแบ่ง (Separator lines)
- ตัวควบคุมที่ใช้ในการเรียกกลับเมื่อค่าของมันมีการเปลี่ยนแปลง

เมื่อเราทำการเชื่อมต่อกับวินโดว์ด้วย GLUI เมื่อค่าของวินโดว์มีการเปลี่ยนแปลงระบบจะส่งค่าดังกล่าวระหว่างกัน หรือเรียกว่า live Variable ซึ่งเราสามารถนำค่าที่ได้ไปทำการคำนวณ หรือสร้างคำสั่งในการแปรรูปทรงได้

2.5 แสง

แสงมีความสำคัญมากในการสร้างงานประเทภภาพสามมิติ เนื่องจากในระบบคอมพิวเตอร์ที่เราใช้แสดงผล สามารถแสดงได้เพียงสองมิติ แต่ในโลกแห่งความจริงจะมีสามมิติ ซึ่งมีความลึกของสีเพิ่มขึ้นมา ดังนั้น แสงที่กระทบกับอ็อบเจ็กต์จึงทำให้เกิดเงาและแสดงผลออกมาเป็นภาพสามมิติ จากรูปจะเห็นได้ว่า ถ้าระบบไม่มีการให้แสงแก่วัตถุ ระบบจะแสดงรูปทรงกลมเป็นลักษณะของแผ่นรองจานรูปวงกลมซึ่งไม่มีความลึกของรูปภาพ แต่ถ้าระบบมีการให้แสงแก่วัตถุทรงกลมแล้ว ระบบจะแสดงผลความลึกของสีออกมา ทำให้รูปทรงกลมแสดงผลเป็นรูปทรงสามมิติ



รูปที่ 2.12 ทรงกลมที่ถูกให้แสงตามทฤษฎีของฟงและทรงกลมปกติ

โดยปกติแล้วทฤษฎีเกี่ยวกับแสงที่เป็นนิยมนและใช้กันในงานสามมิติจะใช้ทฤษฎีแสงของฟง ซึ่งทฤษฎีดังกล่าวได้แบ่ง แสงออกเป็นหลายชนิด ได้แก่



รูปที่ 2.13 ประเภทของการสะท้อนตามทฤษฎีของฟง

แสงโดยรอบ (Ambient Light) แสงที่ไม่ได้มาจากทิศทางที่แน่นอน แสงสว่างที่เกิดขึ้นจะสว่างทั่วพื้นผิวและทุกทิศทาง

แสงแพร่ (Diffuse Light) มาจากทิศทางที่แน่นอนและสะท้อนออกจากพื้นผิว มีทั้งส่วนที่ดูดซับและสะท้อนออกไป การสะท้อนจะมีทุกทิศทาง

แสงกล้ำ (Specular Light) แสงที่มีทิศทางที่แน่นอนในการสะท้อน มีคุณภาพแสงสูง ทำให้เกิด ความสว่างเป็นจุดบนพื้นผิวออบเจ็กต์

แสงเปล่ง (Emissive Light) การปลดปล่อยแสงของอ็อบเจ็กต์ มีลักษณะคล้ายแสงที่มาจากหลอดไฟ ในโมเดลของฟงการกระจายของแสงเปล่งไม่ได้ถูกมาคำนวณการจัดแสง ดังนั้นพื้นผิวจะปรากฏ เช่นเดียวกับแสงสะท้อนอื่นๆ

เราสามารถกำหนดแสงให้แก่อ็อบเจ็กต์ตามหลักการทฤษฎีของฟงด้วย 4 ขั้นตอนดังต่อไปนี้

1.Enable Light Source เป็นการกำหนดแหล่งกำเนิดแสงต้นทางที่จะส่องไปที่อ็อบเจ็กต์ เราสามารถ กำหนดได้หลายตำแหน่งด้วยคำสั่งดังต่อไปนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
// Enable Lighting
glEnable(GL_LIGHTING);
glEnable(GL_LIGHT0);
glEnable(GL_LIGHT1);
glEnable(GL_LIGHT2);
glEnable(GL_LIGHT3);
```

รูปที่ 2.14 การตั้งค่า Enable Light Source

2.Setup Light Position and Intensity เป็นการกำหนดพิกัดของแหล่งกำเนิดแสง และความเข้มของแสงที่จะส่องสว่างไปที่อ็อบเจ็กต์ เราสามารถกำหนดโดยใช้ค่าดังต่อไปนี้

```
GLfloat lightIntensity[] = {1.0f,1.0f,1.0f,0.0f};
GLfloat light_position[] = {1.0f,2.0f,1.50f,1.0f};
glLightfv(GL_LIGHT0,GL_POSITION,light_position);
glLightfv(GL_LIGHT0,GL_DIFFUSE,lightIntensity);
```

รูปที่ 2.15 การตั้งค่า Setup Light Position and Intensity

3.Setup Light Type for Object เป็นการกำหนดค่าของแสงแต่ละชนิดตามทฤษฎีของฟง นอกจากนี้ โอปอเนนจีแอลยังสามารถกำหนดอ็อบเจ็กต์ให้มีคุณสมบัติความมันเงาได้อีกด้วย โดยค่าความมันเงา (Shiness) ยิ่งมีค่ามากแสงจะสะท้อนได้มาก หากมีค่าน้อยอ็อบเจ็กต์จะดูดซับแสงเอาไว้แทนการสะท้อน

```
float mat1_Ambient[] = {0.1,0.0,0.0,1.0};
float mat1_Diffuse[] = {0.8,0.0,0.0,1.0};
float mat1_Specular[] = {0.3,0.9,0.1,1.0};
float mat1_Emission[] = {0.0,0.0,0.0,1.0};
float mat1_Shininess[] = {10};
```

รูปที่ 2.16 การตั้งค่า Setup Light Type for Object

4.Setup to Material กำหนดค่าดังกล่าวให้แก่อ็อบเจ็กต์ด้วยชุดคำสั่งที่ตั้งค่าไว้ดังต่อไปนี้

```
float mat1_Ambient[] = {ambient_r,ambient_g,ambient_b,ambient_o};
float mat1_Diffuse[] = {diffuse_r,diffuse_g,diffuse_b,diffuse_o};
float mat1_Specular[] = {specula_r,specula_g,specula_b,specula_o};
float mat1_Emission[] = {emission_r,emission_g,emission_b,emission_o};
float mat1_Shininess[] = {shiness};
glMaterialfv(GL_FRONT,GL_EMISSION,mat1_Emission);
glMaterialfv(GL_FRONT,GL_AMBIENT,mat1_Ambient);
glMaterialfv(GL_FRONT,GL_DIFFUSE,mat1_Diffuse);
glMaterialfv(GL_FRONT,GL_SPECULAR,mat1_Specular);
glMaterialfv(GL_FRONT,GL_SHININESS,mat1_Shininess);
```

รูปที่ 2.18 การตั้งค่า Setup to Material

144242

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 2.1 พารามิเตอร์ของแหล่งกำเนิดแสงและค่าเริ่มต้น

Parameter Name (pname)	Default Value (param)	Meaing
GL_AMBIENT	{0.0, 0.0, 0.0, 1.0}	ambient RGBA intensity of light
GL_DIFFUSE	{1.0, 1.0, 1.0, 1.0}	diffuse RGBA intensity of light
GL_SPECULAR	{1.0, 1.0, 1.0, 1.0}	specular RGBA intensity of light
GL_POSITION	{0.0, 0.0, 1.0, 1.0}	x, y, z, w) position of light
GL_SPOT_DIRECTION	{0.0, 0.0, -1.0}	(x, y, z) direction of spotlight
GL_SPOT_EXPONENT	0.0	spotlight exponent
GL_SPOT_CUTOFF	180.0	spotlight cutoff angle
GL_CONSTANT_ATTENUATION	1.0	constant attenuation factor
GL_LINEAR_ATTENUATION	0.0	linear attenuation factor
GL_QUADRATIC_ATTENUATION	0.0	quadratic attenuation factor

2.6 การติดตั้ง OpenGL เพื่อทดสอบการทำงาน

การ Setup GLUT สำหรับ Visual C++

1. ดาวน์โหลดไฟล์ glutzip จาก <https://www.opengl.org/>
ไฟล์นี้ประกอบด้วยไฟล์ย่อย 3 ไฟล์คือ glut.h, glut32.lib, และ glut32.dll
2. คัดลอกไฟล์ glut.h ไปไว้ที่ C:\Program Files\Microsoft Visual Studio\VC98\Include\GL
3. คัดลอกไฟล์ glut32.lib ไปไว้ที่ C:\Program Files\Microsoft Visual Studio\VC98\Lib
4. คัดลอกไฟล์ glut32.dll ไปไว้ที่ C:\Windows\system32
5. ทดสอบการรันโปรแกรมด้วยไลบรารี OpenGL

บทที่ 3

การวิเคราะห์และออกแบบระบบ

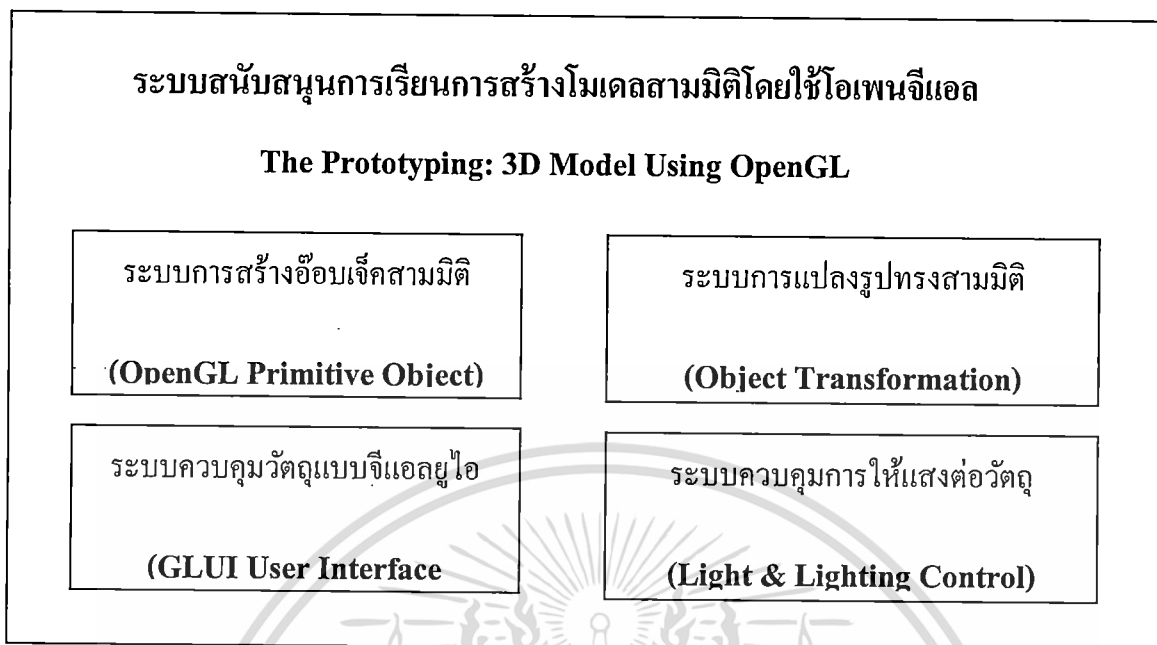
ในบทที่ 3 นี้จะกล่าวถึงการวิเคราะห์และออกแบบของระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล โดยตัวระบบสามารถอธิบายระบบการทำงาน โดยแยกตามส่วนต่างๆดังนี้

3.1 การออกแบบระบบ

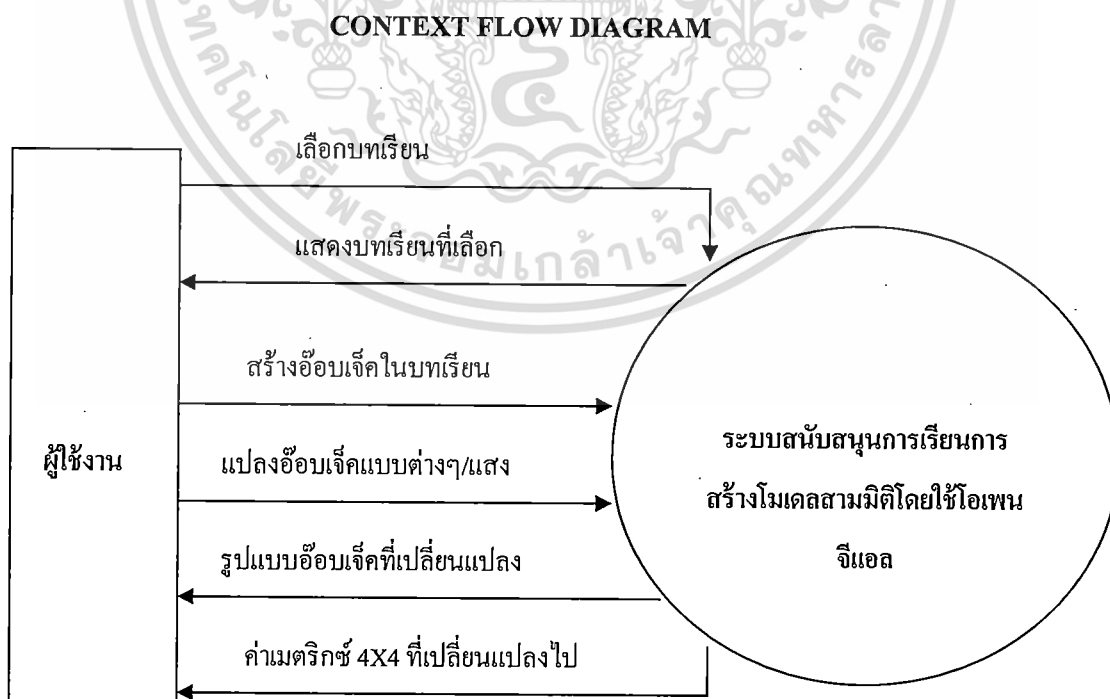
การพัฒนาบบระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล เป็นซอฟต์แวร์สำหรับช่วยให้ผู้เริ่มต้นศึกษาด้านโมเดลสามมิติเรียนรู้ได้อย่างรวดเร็ว โดยผู้เรียนได้ทราบถึงข้อมูลต่างๆแบบอินเตอร์แอคทีฟที่ตอบสนองต่อผู้ใช้แบบทันทีเมื่อมีการกระทำต่ออ็อบเจกต์ที่สร้างขึ้น

ระบบของระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอลประกอบด้วย 4 ส่วนคือ

- 1.ระบบการสร้างอ็อบเจ็กต์สามมิติ(OpenGL Primitive Object)
- 2.ระบบการแปลงรูปทรงสามมิติ(Object Transformation)
- 3.ระบบควบคุมวัตถุแบบจีแอลยูไอ(GLUI User Interface Control)
- 4.ระบบควบคุมการให้แสงต่อวัตถุ(Light & Lighting Control)



รูปที่ 3.1 องค์ประกอบของของระบบระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล



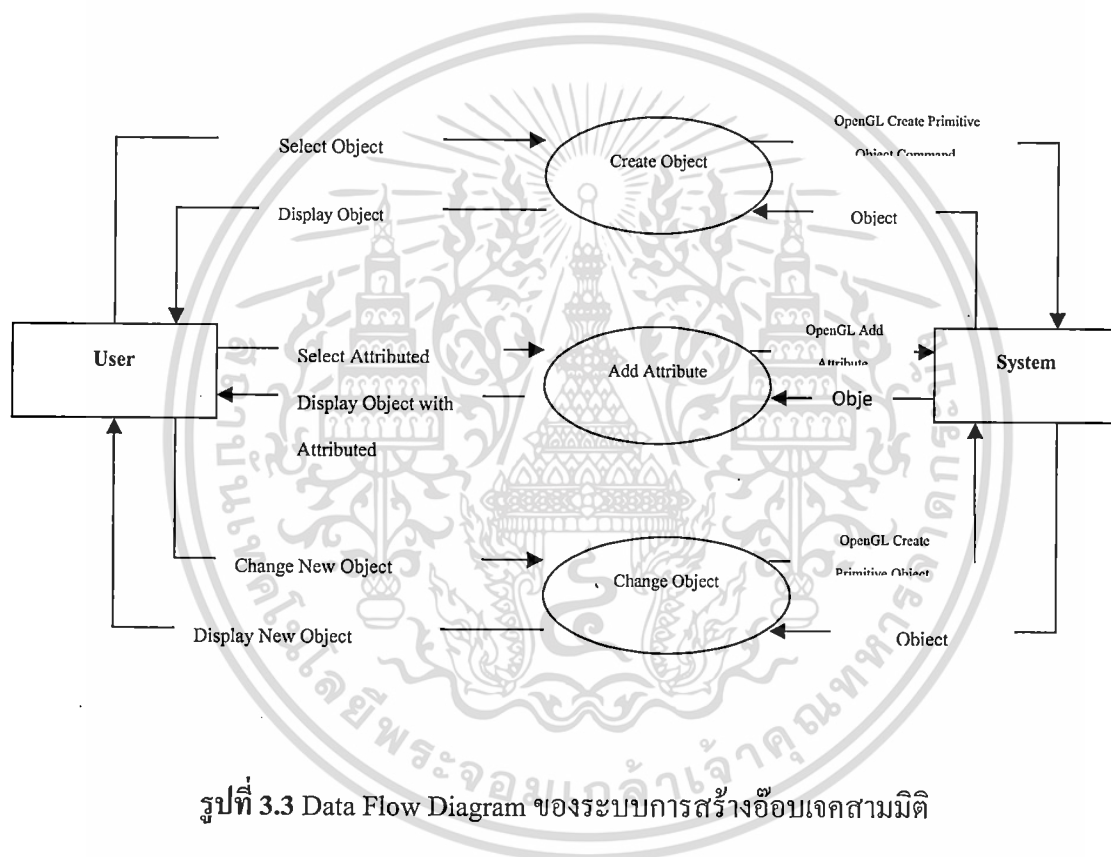
รูปที่ 3.2 Context Flow Diagram ของระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การทำงานของระบบ

3.2.1 ระบบการสร้างอ็อบเจกต์สามมิติ

เป็นระบบที่ช่วยในการสร้างอ็อบเจกต์สามมิติ ซึ่งเป็นพื้นฐานเริ่มต้นของระบบ โดยการทำงานของระบบจะทำการสร้างอ็อบเจกต์ตามที่ผู้ใช้งานได้เลือกผ่านส่วนติดต่อของผู้ใช้ที่เรียกว่า GLUTI ซึ่งเป็นส่วนติดต่อของผู้ใช้ที่ใช้สำหรับติดต่อกับโอเพนจีแอล จากนั้นระบบจะทำการส่งคำสั่งเรียกใช้ไลบรารีของโอเพนจีแอล จากนั้นระบบจะทำการแสดงผลอ็อบเจกต์ดังกล่าวออกมาบนหน้าจอ



รูปที่ 3.3 Data Flow Diagram ของระบบการสร้างอ็อบเจกต์สามมิติ

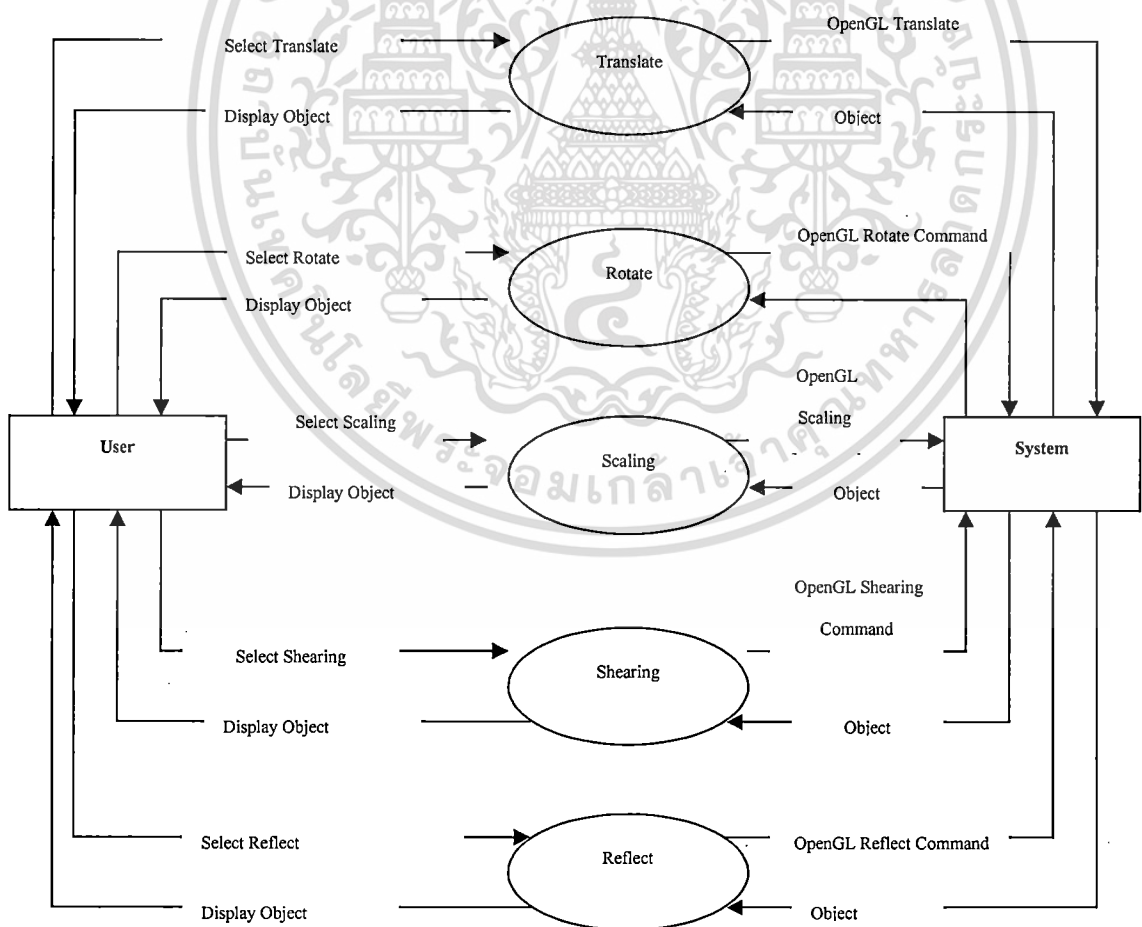
จากรูปที่ 3.4 แสดงการทำงานของระบบการสร้างอ็อบเจกต์สามมิติ ได้แก่ การสร้างอ็อบเจกต์สามมิติ การเพิ่มแอททริบิวต์ และการเปลี่ยนอ็อบเจกต์ โดยมีขั้นตอนดังต่อไปนี้

1. กระบวนการสร้างอ็อบเจกต์ เป็นกระบวนการที่ผู้ใช้เลือกอ็อบเจกต์ที่ต้องการศึกษาเพื่อให้ระบบสร้างรูปอ็อบเจกต์สามมิติดังกล่าวบนหน้าจอ โดยเมื่อผู้ใช้เลือกอ็อบเจกต์ระบบจะทำการส่งคำสั่งไปไลบรารีของโอเพนจีแอลเพื่อเรียกใช้งานชุดคำสั่ง จากนั้นระบบจะทำการสร้างและแสดงผลอ็อบเจกต์บนหน้าจอ

2. กระบวนการเพิ่มแอททริบิวต์ เป็นกระบวนการที่ผู้ใช้ใช้ปรับแต่งอ็อบเจกต์ที่จะศึกษาเพื่อเปลี่ยนแปลง อ็อบเจกต์ตามแอททริบิวต์ที่กำหนดไว้ เช่น สีของอ็อบเจกต์ เป็นต้น
3. การบวนการเปลี่ยนแปลงอ็อบเจกต์ เป็นกระบวนการที่ผู้ใช้ทำการเปลี่ยนอ็อบเจกต์ที่ต้องการศึกษา ผู้ใช้สามารถเปลี่ยนแปลงอ็อบเจกต์ที่สนใจได้ตลอดเวลา โดยเมื่อใช้เลือกระบบจะทำการส่งคำสั่ง ไปไลบรารีของ โอเพนจีแอลเพื่อเรียกใช้งานชุดคำสั่ง จากนั้นระบบจะทำการลบหน้าจอและส่ง อ็อบเจกต์ใหม่แสดงบนหน้าจอ

3.2.2 ระบบการแปลงรูปทรงสามมิติ

เป็นระบบที่จะทำการแปลงอ็อบเจกต์ที่ผู้ใช้เลือกเลือก เพื่อศึกษาการแปลงรูปทรงสามมิติ โดยการทำงานดังกล่าวจะประกอบไปด้วย การย่อขยายรูปทรง การเคลื่อนย้ายรูปทรง การหมุน การเฉือน และการสะท้อนรูปทรง



รูปที่ 3.4 Data Flow Diagram ของระบบการแปลงรูปทรงสามมิติ

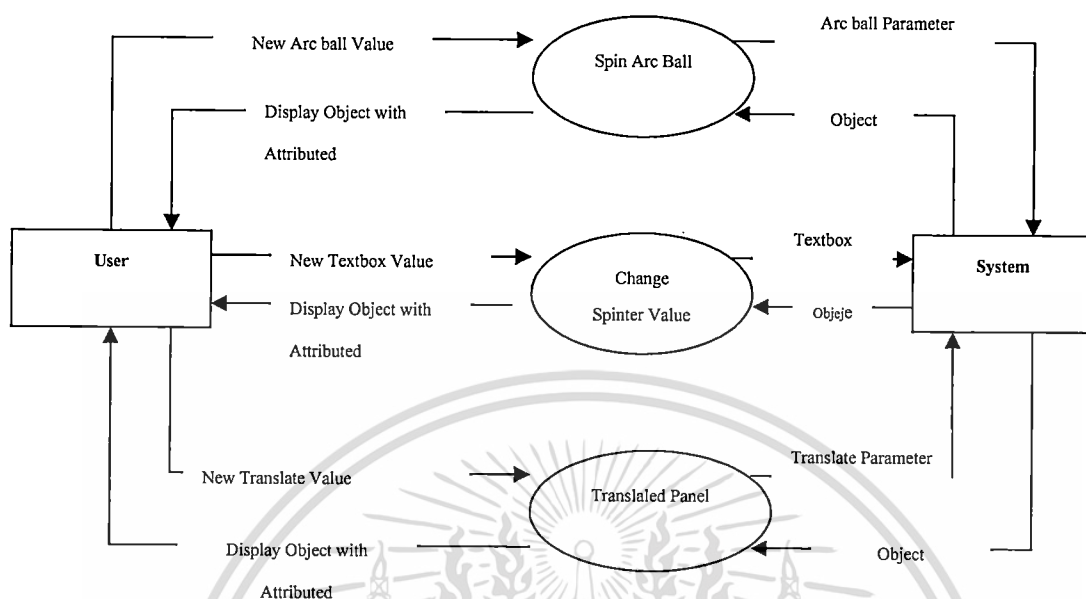
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.5 แสดงขั้นตอนของการแปลงรูปทรงสามมิติ โดยมีหลักการดังนี้

1. กระบวนการเคลื่อนย้ายวัตถุ (Translate) เป็นกระบวนการที่ระบบใช้เคลื่อนย้ายอ็อบเจกต์ไปในระนาบแนวแกนสามมิติ X Y และ Z
2. กระบวนการหมุนวัตถุ (Rotate) เป็นกระบวนการที่ระบบใช้หมุนอ็อบเจกต์ไปตามระนาบแนวแกนสามมิติ X Y และ Z
3. กระบวนการย่อขยายวัตถุ (Scaling) เป็นกระบวนการที่ระบบใช้ย่อหรือขยายอ็อบเจกต์ไปตามระนาบแนวแกนสามมิติ X Y และ Z
4. กระบวนการเฉือนวัตถุ (Shearing) เป็นกระบวนการที่ระบบใช้เฉือนอ็อบเจกต์ไปตามระนาบแนวแกนสามมิติ X Y และ Z
5. กระบวนการสะท้อนของวัตถุ (Reflect) เป็นกระบวนการที่ระบบใช้สะท้อนวัตถุไปตามระนาบตรงข้ามกับอ็อบเจกต์หลัก เพื่อให้ในการสร้างเงาสะท้อนที่เหมือนกับต้นแบบ เช่น เงาของอ็อบเจกต์ที่สะท้อนกับกระจกเงา เป็นต้น

3.2.3 ระบบควบคุมอ็อบเจกต์แบบจีแอลยูไอ (GLUI)

เป็นระบบที่จะทำการควบคุมการแปลงอ็อบเจกต์ด้วยส่วนติดต่อกับผู้ใช้ที่โอเพนจีแอลเตรียมไว้ให้ โดยหลักการทำงานผู้ใช้จะเป็นผู้ควบคุมเมนู โดยอาจมีการทดลองปรับค่าของพารามิเตอร์ต่างๆ ใส่งบในระบอบโดยผ่าน อาร์คบอล สปีนเนอร์ หรือ ปุ่มแบบเรดิโอ จากนั้นระบบจะส่งพารามิเตอร์ดังกล่าวไปทำการปรับปรุงอ็อบเจกต์เพื่อแสดงผลบนหน้าจอ



รูปที่ 3.5 Data Flow Diagram ของระบบควบคุมวัตถุแบบจีแอลยูไอ

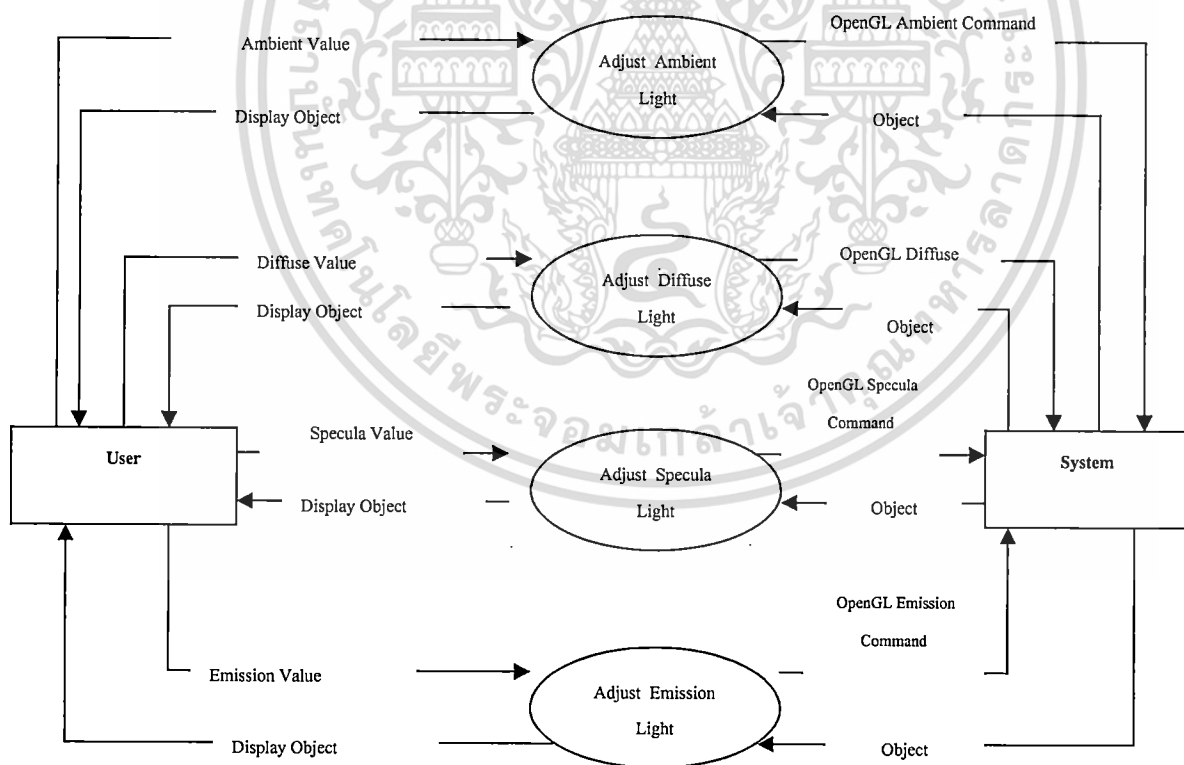
จากรูปที่ 3.6 แสดงขั้นตอนของการควบคุมวัตถุแบบจีแอลยูไอ โดยมีหลักการดังนี้

1. กระบวนการหมุนอาร์คบอล การหมุนอาร์คบอลจะเป็นส่วนติดต่อกับผู้ใช้ซึ่งโอเพนจีแอลเตรียมไว้ให้ใช้สำหรับหมุนอ็อบเจกต์ได้สะดวกมากขึ้น โดยลักษณะจะคล้ายกับแทรกบอลบนเมาส์ เมื่อผู้ใช้ทำการหมุนอาร์คบอล ค่าที่เปลี่ยนแปลงไปของ X Y และ Z จะถูกส่งไปที่ระบบและจะถูกทำการแปลงรูปภาพตามข้อมูลที่ได้รับมา จากนั้นจึงแสดงผลอ็อบเจกต์ออกมาทางจอภาพ
2. กระบวนการหมุนค่าสปินเตอร์ สปินเตอร์มีลักษณะเหมือน Drop Down List ทั่วไป แต่สำหรับส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอ สปินเตอร์จะมีค่าได้ทั้งบวก ลบ จำนวนเต็ม หรือทศนิยม นอกจากนี้ยังมีค่าลิมิตได้อีกด้วย เมื่อเมื่อผู้ใช้ทำการปรับสปินเตอร์บอล ค่าที่เปลี่ยนแปลงไปของ X Y และ Z จะถูกส่งไปที่ระบบและจะถูกทำการแปลงรูปภาพตามข้อมูลที่ได้รับมา จากนั้นจึงแสดงผลอ็อบเจกต์ออกมาทางจอภาพ สปินเตอร์มักจะถูกใช้กับคำสั่งการย่อขยายรูปภาพ การปรับค่าของแสงแบบต่างๆ

- กระบวนการปรับค่าเคลื่อนย้ายตามแนวแกน (Translated Panel) Translate Panel มีลักษณะเป็นปุ่มขนาดใหญ่สัญลักษณ์ถูกศรตามแนวแกนสามมิติ เมื่อผู้ใช้ทำการแดรกเมาส์ไปตามแนวแกน ค่าที่เปลี่ยนแปลงไปจะถูกส่งไปที่ระบบ จากนั้นระบบการแปลงรูปภาพตามข้อมูลที่รับมา จากนั้นจึงแสดงผลอ็อบเจกต์ออกมาทางจอภาพ Translated Panel โดยปกติมักจะถูกใช้กับคำสั่งการเคลื่อนย้ายรูปภาพเป็นหลัก

3.2.4 ระบบการให้แสงต่อวัตถุ (Light&Lighting)

เป็นระบบที่ใช้สำหรับให้แสงต่อวัตถุ ซึ่งเป็นส่วนที่สำคัญมากในการสร้างโมเดลสามมิติ เพราะหากไม่มีการปรับแต่งแสงที่ส่งไปกระทบกับวัตถุ เราจะเห็นภาพเป็นแค่สองมิติเท่านั้น เพราะในระบบคอมพิวเตอร์เป็นระบบสองมิติ ต้องมีการปรับค่าความสว่างแสงเพื่อให้วัตถุแสดงค่าความลึกของแสงและสีออกมา



รูปที่ 3.6 Data Flow Diagram ของระบบควบคุมการให้แสงต่อวัตถุ (Light&Lighting Control)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 3.7 แสดงขั้นตอนของการควบคุมการให้แสงต่อวัตถุ โดยมีหลักการดังนี้

1. การปรับค่าสภาวะแสงภายนอก (Adjust Ambient) เป็นการปรับค่าแสงภายนอกต่อวัตถุที่กระจายอยู่ในแวดล้อม ซึ่งแสงดังกล่าวจะเป็นแสงที่สลัวไม่สว่าง เป็นแสงที่สะท้อนกับวัตถุอื่นมากระทบกับตัวอ็อบเจกต์ที่ผู้เรียนรู้สร้างขึ้นมา เมื่อผู้ใช้ทำการปรับค่าของแสงภายนอก ระบบจะทำการสร้างอ็อบเจกต์ใหม่โดยผสมกับพารามิเตอร์ที่มีการปรับแต่งและส่งผลออกมาแสดงทางจอภาพ
2. การปรับค่าแสงแผ่ (Adjust Diffuse Light) แสงแผ่เป็นแสงที่มาจากแหล่งกำเนิดโดยตรงแต่เมื่อสะท้อนกับวัตถุจะกระจายไปในทุกทิศทาง เมื่อผู้ใช้ทำการปรับค่าของแสงแผ่ ระบบจะทำการสร้างอ็อบเจกต์ใหม่โดยผสมกับพารามิเตอร์ที่มีการปรับแต่งและส่งผลออกมาแสดงทางจอภาพ
3. การปรับค่าแสงกล้า (Adjust Specular Light) แสงกล้าเป็นแสงที่มาจากแหล่งที่เฉพาะเจาะจง และจะสะท้อนกับวัตถุไปทางที่เจาะจงเช่นกัน คือมุมสะท้อนที่ตกกระทบของแสงจากต้นกำเนิดที่ส่องมากระทบกับวัตถุ ตัวอย่างของแสงกล้าได้แก่ แสงจากกระบอกไฟฉาย แสงจากไฟแอลอีดี แสงจากไฟหน้าของรถยนต์ เป็นต้น เมื่อผู้ใช้ทำการปรับค่าของแสงกล้า ระบบจะทำการสร้างอ็อบเจกต์ใหม่โดยผสมกับพารามิเตอร์ที่มีการปรับแต่งและส่งผลออกมาแสดงทางจอภาพ
4. การปรับตั้งค่าแสงเปล่ง (Adjust Emission Light) แสงเปล่งคือแสงที่เปล่งออกมาตัวของวัตถุเอง เช่น แสงจากหลอดไฟฟ้า แสงจากดวงอาทิตย์ แต่โมเดลของฟงที่เราใช้ในโอเพนจีแอลนั้นยังไม่สามารถแสดงแสงในลักษณะนี้ได้ดี แต่ผู้ใช้ยังสามารถปรับแต่งค่าบางส่วนได้ เช่นเดียวกับแสงประเภทอื่นๆ

3.3 แบบจำลองเชิงแนวคิดของระบบ

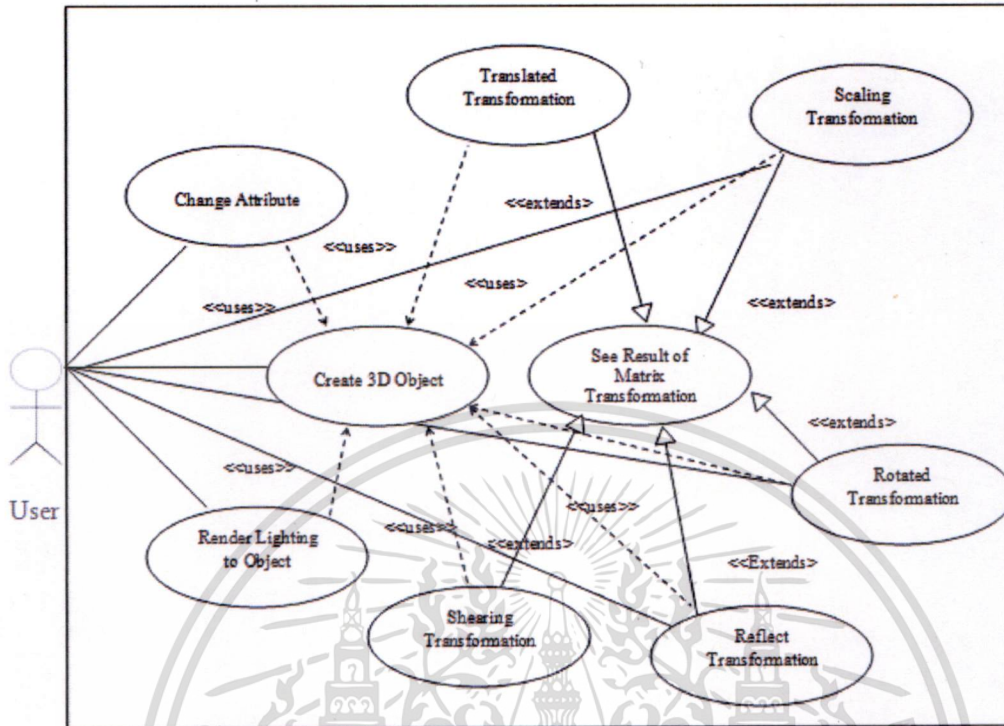
3.3.1 Use Case model

แบบจำลองเชิงแนวคิดของระบบสนับสนุนการเรียนการสอนโมเดลสามมิติโดยใช้โอเพนจี

แอล ส่วนการทำงานต่างๆของระบบสามารถอธิบายด้วย Use Case Diagram ดังรูป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

USECASE DIAGRAM
LEARNING SUPPORT SYSTEM FOR 3D MODEL CREATION USING OPENGL



รูปที่ 3.7 Use Case Diagrams ระบบสนับสนุนการเรียนการสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล

ตารางที่ 3.1 Use Case Diagram ของส่วน Create 3D Objects.

Use Case:	Create 3D Objects	
Triggering:		
Description:	สร้างอ็อบเจกต์สามมิติ	
Actor:	User	
Precondition:		
Post Condition:		
Flow Of Events:	Actor	System
	1. User ทำการเลือกอ็อบเจกต์พื้นฐานแบบสามมิติ	2. ระบบทำการสร้างโมเดลออกทางจอภาพบนจุดกำเนิด X=0,Y=0,Z=0 (Origin Coordinate)
Exception:		

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.2 Use Case Diagrams ของส่วน Change Attribute.

Use Case:	Change Attribute	
Triggering:		
Description:	ทำการปรับปรุงค่าของแอตทริบิวต์	
Actor:	User	
Precondition:	ผู้ใช้งานสร้างอ็อบเจกต์พื้นฐานในระบบ	
Post Condition:	ทำการเปลี่ยนแปลงค่าแอตทริบิวต์ของอ็อบเจกต์ได้	
Flow Of Events:	Actor	System
	1. User ทำการเลือกอ็อบเจกต์พื้นฐานแบบสามมิติ	2. ระบบทำการสร้างโมเดลออกทางจอภาพบนจุดกำเนิด $X=0, Y=0, Z=0$ (Origin Coordinate)
Exception:	1. ผู้ใช้งานปรับเปลี่ยนค่าของแอตทริบิวต์โดยที่ไม่มีอ็อบเจกต์	1. ไม่ทำการแอตทริบิวต์ตามกำหนดแสดงหน้าจอว่างๆเพื่อเตรียมพร้อมสร้างอ็อบเจกต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.3 Use Case Diagrams ของส่วน Translated Transformation

Use Case:	Translated Transformation	
Triggering:		
Description:	ทำการเคลื่อนย้ายอ็อบเจกต์ไปตามแนวแกนสามมิติ	
Actor:	User	
Precondition:	ผู้ใช้ทำการสร้างอ็อบเจกต์แบบสามมิติขึ้นมาในระบบ	
Post Condition:	ผู้ใช้สามารถทำการเคลื่อนย้ายอ็อบเจกต์ไปตามแนวแกนสามมิติ	
Flow Of Events:	Actor	System
	1. User ทำการแดรกเมาส์โดยให้ Translate Panel ในการเคลื่อนย้ายวัตถุตามแนวแกนสามมิติเพื่อส่งค่าของจุดหมายปลายทางให้ระบบ	1. ระบบทำการสร้างทำการเคลื่อนย้ายอ็อบเจกต์ตามแนวแกนสามมิติและแสดงภาพออกทางหน้าจอ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกต์ขึ้นมาในระบบ	1. สร้างอ็อบเจกต์ด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น

ตารางที่ 3.4 Use Case Diagrams ของส่วน Rotated Transformation

Use Case:	Rotate Transformation	
Triggering:		
Description:	ทำการหมุนอ็อบเจกไปตามแนวแกนสามมิติ	
Actor:	User	
Precondition:	ผู้ใช้ทำสร้างอ็อบเจกแบบสามมิติขึ้นมาในระบบ	
Post Condition:	ผู้ใช้สามารถทำการหมุนอ็อบเจกไปตามแนวแกนสามมิติ	
Flow Of Events:	Actor	System
	1. User ทำการหมุนอ็อบเจกได้โดยใช้อาร์คบอล ซึ่งเป็นส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอ สำหรับติดต่อกับไลบรารีโอเพนจีแอล	1. ระบบทำการสร้างทำการหมุนอ็อบเจกตามแนวแกนสามมิติและแสดงภาพออกทางหน้าจอ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกขึ้นมาในระบบ	1. สร้างอ็อบเจกด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.5 Use Case Diagrams ของส่วน Scaling Transformation

Use Case:	Scaling Transformation	
Triggering:		
Description:	ทำการย่อหรือขยายอ็อบเจกต์ไปตามแนวแกนสามมิติ	
Actor:	User	
Precondition:	ผู้ใช้ทำสร้างอ็อบเจกต์แบบสามมิติขึ้นมาในระบบ	
Post Condition:	ผู้ใช้สามารถทำการย่อหรือขยายอ็อบเจกต์ไปตามแนวแกนสามมิติ	
Flow Of Events:	Actor	System
	1. User ทำการย่อหรือขยายอ็อบเจกต์ได้โดยใช้สปีนเนอร์ ซึ่งเป็นส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับติดต่อกับไลบรารีโอเพนจีแอล	1. ระบบทำการย่อหรือขยายอ็อบเจกต์ตามแนวแกนสามมิติและแสดงภาพออกทางหน้าจอ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกต์ขึ้นมาในระบบ	1. สร้างอ็อบเจกต์ด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น

ตารางที่ 3.6 Use Case Diagrams ของส่วน Shearing Transformation

Use Case:	Shearing Transformation	
Triggering:		
Description:	ทำการเลื่อนอ็อบเจกไปตามแนวแกนสามมิติ	
Actor:	User	
Precondition:	ผู้ใช้ทำสร้างอ็อบเจกแบบสามมิติขึ้นมาในระบบ	
Post Condition:	ผู้ใช้สามารถทำการเลื่อนอ็อบเจกไปตามแนวแกนสามมิติ	
Flow Of Events:	Actor	System
	1.User ทำการเลื่อนอ็อบเจกได้โดยใช้สปีนเตอร์ ซึ่งเป็นส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอ สำหรับติดต่อกับไลบรารีโอเพนจีแอล	1.ระบบทำการเลื่อนอ็อบเจกตามแนวแกนสามมิติและแสดงภาพออกทางหน้าจอ
Exception:	1.ผู้ใช้ไม่ได้สร้างอ็อบเจกขึ้นมาในระบบ	1.สร้างอ็อบเจกด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 3.7 Use Case Diagrams ของส่วน Reflect Transformation

Use Case:	Reflect Transformation	
Triggering:		
Description:	ทำการสะท้อนอ็อบเจกไปตามแนวแกนสามมิติ	
Actor:	User	
Precondition:	ผู้ใช้งานสร้างอ็อบเจกแบบสามมิติขึ้นมาในระบบ	
Post Condition:	ผู้ใช้งานสามารถทำการสะท้อนอ็อบเจกไปตามแนวแกนสามมิติ	
Flow Of Events:	Actor	System
	1. User ทำการสะท้อนอ็อบเจกได้โดยใช้สปีนเตอร์ ซึ่งเป็นส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอ สำหรับติดต่อกับไลบรารีโอเพนจีแอล	1. ระบบทำการสะท้อนอ็อบเจกตามแนวแกนสามมิติและแสดงภาพออกทางหน้าจอ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกขึ้นมาในระบบ	1. สร้างอ็อบเจกด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

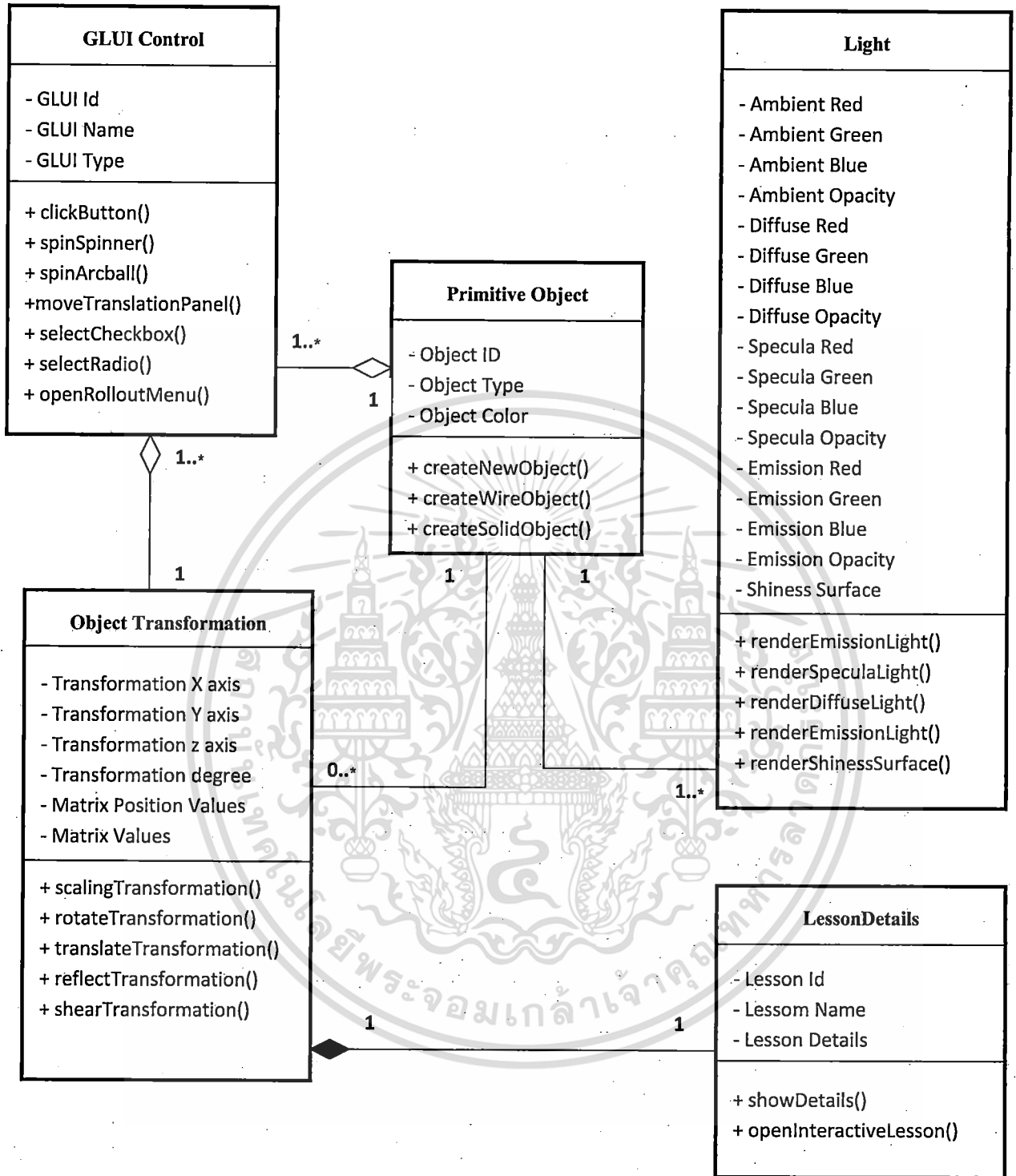
ตารางที่ 3.8 Use Case Diagrams ของส่วน See Result of Matrix Transformation

Use Case:	See Result of Matrix Transformation	
Triggering:	อ็อบเจกมีการเปลี่ยนแปลงไปตามแนวแกนสามมิติ	
Description:	ทำการแสดงค่าของเมตริกซ์ที่เปลี่ยนแปลงไปเมื่อผู้ใช้ส่ง, ค่าพารามิเตอร์ผ่านส่วนติดต่อกับผู้ใช้	
Actor:	User, System	
Precondition:	1. ผู้ใช้ทำสร้างอ็อบเจกแบบสามมิติขึ้นมาในระบบ 2. ผู้ใช้ต้องการแปลงรูปภาพโดยใช้ส่วนติดต่อกับผู้ใช้เพื่อส่งพารามิเตอร์ไปสู่ระบบ	
Post Condition:	1. ผู้ใช้สามารถตรวจสอบดูค่าของเมตริกซ์ที่เปลี่ยนแปลงไปเมื่อทำการแปลงรูปภาพ	
Flow Of Events:	Actor	System
	1. User ทำการแปลงอ็อบเจกด้วยการส่งค่าพารามิเตอร์ไปสู่ระบบโดยใช้ส่วนติดต่อกับผู้ใช้	1. ระบบทำการประมวลผลและแสดงค่าเมตริกซ์ที่เปลี่ยนแปลงไปออกมาทางจอภาพ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกขึ้นมาในระบบ	1. สร้างอ็อบเจกด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น

ตารางที่ 3.9 Use Case Diagrams ของส่วน Render Light to Object

Use Case:	Render Light to Object	
Triggering:		
Description:	ทำการให้กำหนดแสงให้อ็อบเจกต์ด้วยทฤษฎีของฟง	
Actor:	User, System	
Precondition:	1. ผู้ใช้ทำสร้างอ็อบเจกต์แบบสามมิติขึ้นมาในระบบ	
Post Condition:	1. ผู้ใช้กำหนดค่าแสงให้อ็อบเจกต์ที่สร้างขึ้นมาได้	
Flow Of Events:	Actor	System
	1. User ทำการปรับค่ากำหนดของแสงผ่านส่วนติดต่อของผู้ใช้ที่ออกแบบเอาไว้	1. ระบบทำการประมวลผลและแสดงค่ากำหนดของแสงที่ผู้ใช้ป้อนเข้ามาและแสดงภาพออกทางจอภาพ
Exception:	1. ผู้ใช้ไม่ได้สร้างอ็อบเจกต์ขึ้นมาในระบบ 2. ผู้ใช้ปรับค่าเกินขีดของฟังก์ชัน	1. สร้างอ็อบเจกต์ด้วยค่ากำหนดมาตรฐานขึ้นมาเตรียมพร้อมสำหรับการทำงานเบื้องต้น 2. ระบบกำหนดค่าขีดบนสปีนเตอร์เพื่อป้องกันค่าที่ป้อนจากผู้ใช้เกินมาตรฐานของฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 Class Diagram ของระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.3.2 Structure model

แบบจำลองเชิงโครงสร้างของระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้ไอโฟนจีแอล สามารถอธิบายโครงสร้างส่วนต่างๆของระบบด้วย Class Diagram ดังรูปที่ 3.9
Class Diagram ของระบบประกอบด้วย 5 คลาส ดังนี้

1.Primitive Object : เป็นคลาสที่ทำหน้าที่เกี่ยวกับการสร้างอ็อบเจกต์พื้นฐานขึ้นมาในระบบ ประกอบด้วย Attribute และ Operation ดังต่อไปนี้

1.1 Attribute - Object ID : เก็บรหัสของอ็อบเจกต์

1.2 Attribute - Object Type : เก็บรูปแบบของอ็อบเจกต์

1.3 Attribute - Object Color : เก็บสีของอ็อบเจกต์

1.4 Operation - createNewObject() : ทำหน้าที่สร้างอ็อบเจกต์ใหม่ขึ้นมาในระบบ

1.5 Operation - createWireObject() : ทำหน้าที่สร้างอ็อบเจกต์แบบโครงลวด

1.6 Operation - createSolidObject() : ทำหน้าที่สร้างอ็อบเจกต์แบบทึบ

2.Lesson Details : เป็นคลาสที่ทำหน้าที่เกี่ยวกับการสร้างบทเรียนของแต่ละการแปลงอ็อบเจกต์ในระบบ

2.1 Attribute - Lesson ID : เก็บรหัสของบทเรียน

2.2 Attribute - Lesson Name : เก็บชื่อของบทเรียน

2.3 Attribute - Lesson Details : เก็บรายละเอียดของบทเรียน

2.4 Operation - showDetails() : ทำหน้าที่แสดงบทเรียนและข้อมูลทั้งหมด

2.5 Operation - openInteractiveLesson() : ทำหน้าที่สร้างบทเรียนแบบอินเตอร์แอคทีฟ

3.Object Transformation : เป็นคลาสที่ทำหน้าที่เกี่ยวกับการแปลงอ็อบเจกต์ในระบบ ประกอบด้วย

3.1 Attribute - Transformation X axis : เก็บค่าการแปลงอ็อบเจกต์ของแกน X

3.2 Attribute - Transformation Y axis : เก็บค่าการแปลงอ็อบเจกต์ของแกน Y

3.3 Attribute - Transformation Z axis : เก็บค่าการแปลงอ็อบเจกต์ของแกน Z

3.4 Attribute - Transformation degree : เก็บค่าองศาการหมุน

3.5 Attribute - Matrix Position Values : เก็บตำแหน่งของเมตริกซ์

3.6 Attribute - Matrix Values : เก็บค่าของเมตริกซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.7 Operation – scalingTransformation() : ทำหน้าที่ย่อหรือขยายอ็อบเจกตามแนวแกนสามมิติ

3.8 Operation - rotateTransformation() : ทำหน้าที่หมุนอ็อบเจกตามแนวแกนสามมิติ

3.9 Operation - translateTransformation() : ทำหน้าที่เคลื่อนย้ายอ็อบเจกตามแนวแกนสามมิติ

3.10 Operation -reflectTransformation() : ทำหน้าที่สะท้อนอ็อบเจกตรงข้ามแนวแกนสามมิติ

3.11 Operation -shearTransformation() : ทำหน้าที่เฉือนอ็อบเจกตามแนวแกนสามมิติ

4.GLUI Control : เป็นคลาสที่ทำหน้าที่เกี่ยวกับการสร้างส่วนติดต่อกับผู้ใช้และอ็อบเจกในระบบ

4.1 Attribute - GLUT Id : เก็บรหัสของจีแอลยูไอ

4.2 Attribute - GLUT Name : เก็บชื่อจีแอลยูไอ

4.3 Attribute - GLUT Type : เก็บค่ารูปแบบของจีแอลยูไอ

4.4 Operation - clickButton() : ทำหน้าที่จับเหตุการณ์คลิกปุ่มของจีแอลยูไอ

4.5 Operation - spinSpinner() : ทำหน้าที่จับเหตุการณ์ปรับค่าค่าสปินเนอร์ของจีแอลยูไอ

4.6 Operation - spinArcball() : ทำหน้าที่จับเหตุการณ์หมุนอาร์คบอลของจีแอลยูไอ

4.7 Operation - moveTranslationPanel(): ทำหน้าที่จับเหตุการณ์เคลื่อนย้าย โดยการแทรกเม้าบน Translation Panel

4.8 Operation - selectCheckbox(): ทำหน้าที่จับเหตุการณ์เลือกเมนูแบบเรดิโอ

4.9 Operation - selectRadio() : ทำหน้าที่จับเหตุการณ์เลือกเมนูแบบเรดิโอ

4.10 Operation - openRolloutMenu() : ทำหน้าที่จับเหตุการณ์เปิดเมนูแบบยืดหดของจีแอลยูไอ

5.Light Control : เป็นคลาสที่ทำหน้าที่เกี่ยวกับการให้แสงแก่วัตถุ โดยใช้ทฤษฎีของฟง

5.1 Attribute - Ambient Red : เก็บค่าสีแดงของ Ambient

5.2 Attribute - Ambient Green : เก็บค่าสีเขียวของ Ambient

5.3 Attribute - Ambient Blue : เก็บค่าสีน้ำเงินของ Ambient

5.4 Attribute - Ambient Opacity : เก็บค่าความโปร่งใสของ Ambient

5.5 Attribute - Diffuse Red : เก็บค่าสีแดงของ Diffuse

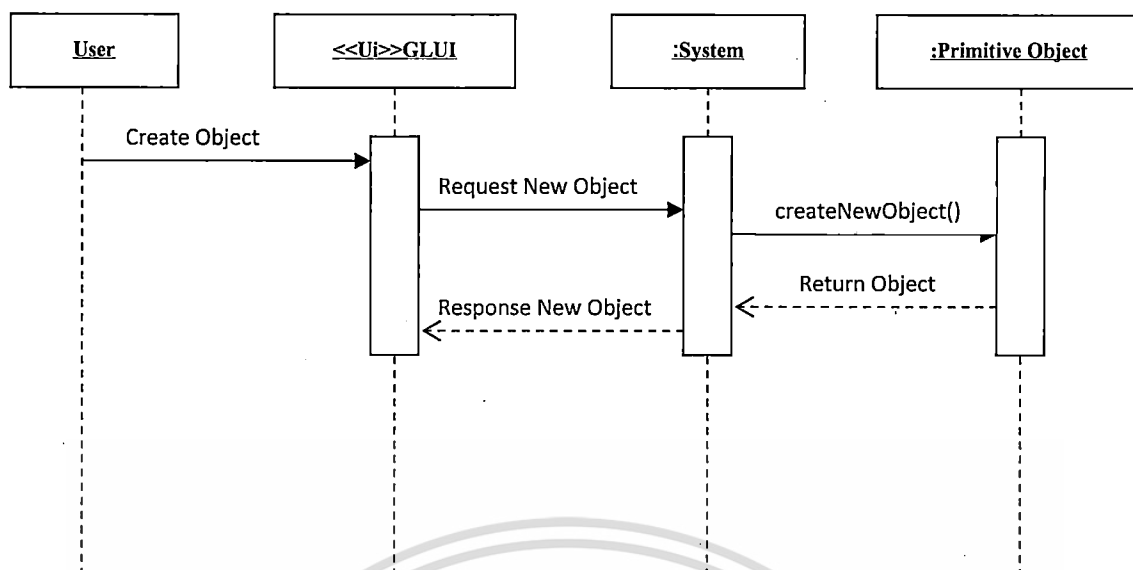
5.6 Attribute - Diffuse Green : เก็บค่าสีเขียวของ Diffuse

5.7 Attribute - Diffuse Blue : เก็บค่าสีน้ำเงินของ Diffuse

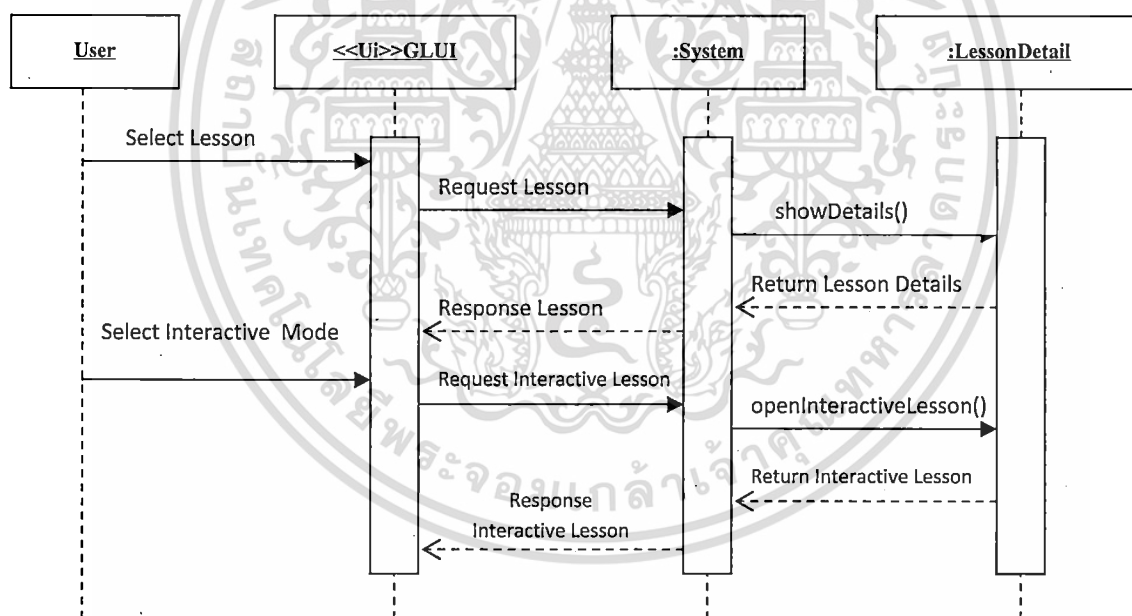
5.8 Attribute - Diffuse Opacity : เก็บค่าความโปร่งใสของ Diffuse

5.9 Attribute - Specular Red : เก็บค่าสีแดงของ Specular

- 5.10 Attribute - Specula Green : เก็บค่าสีเขียวของ Specula
- 5.11 Attribute - Specula Blue : เก็บค่าสีน้ำเงินของ Specula
- 5.12 Attribute - Specula Opacity : เก็บค่าความโปร่งใสของ Specula
- 5.13 Attribute - Emission Red : เก็บค่าสีแดงของ Emission
- 5.14 Attribute - Emission Green : เก็บค่าสีเขียวของ Emission
- 5.15 Attribute - Emission Blue : เก็บค่าสีน้ำเงินของ Emission
- 5.16 Attribute - Emission Opacity : เก็บค่าความโปร่งใสของ Emission
- 5.17 Attribute - Shiness Surface : เก็บค่าความมันเงาของพื้นผิว
- 5.18 Operation - renderEmissionLight() : ทำหน้าที่ให้แสงอ็อบเจกแบบ Emission
- 5.19 Operation - renderSpeculaLight() : ทำหน้าที่ให้แสงอ็อบเจกแบบ Specula
- 5.20 Operation - renderDiffuseLight() : ทำหน้าที่ให้แสงอ็อบเจกแบบ Diffuse
- 5.21 Operation - renderAmbientLight() : ทำหน้าที่ให้แสงอ็อบเจกแบบ Ambient
- 5.22 Operation - renderShinessSurface() : ทำหน้าที่ให้แสงอ็อบเจกแบบ Shiness

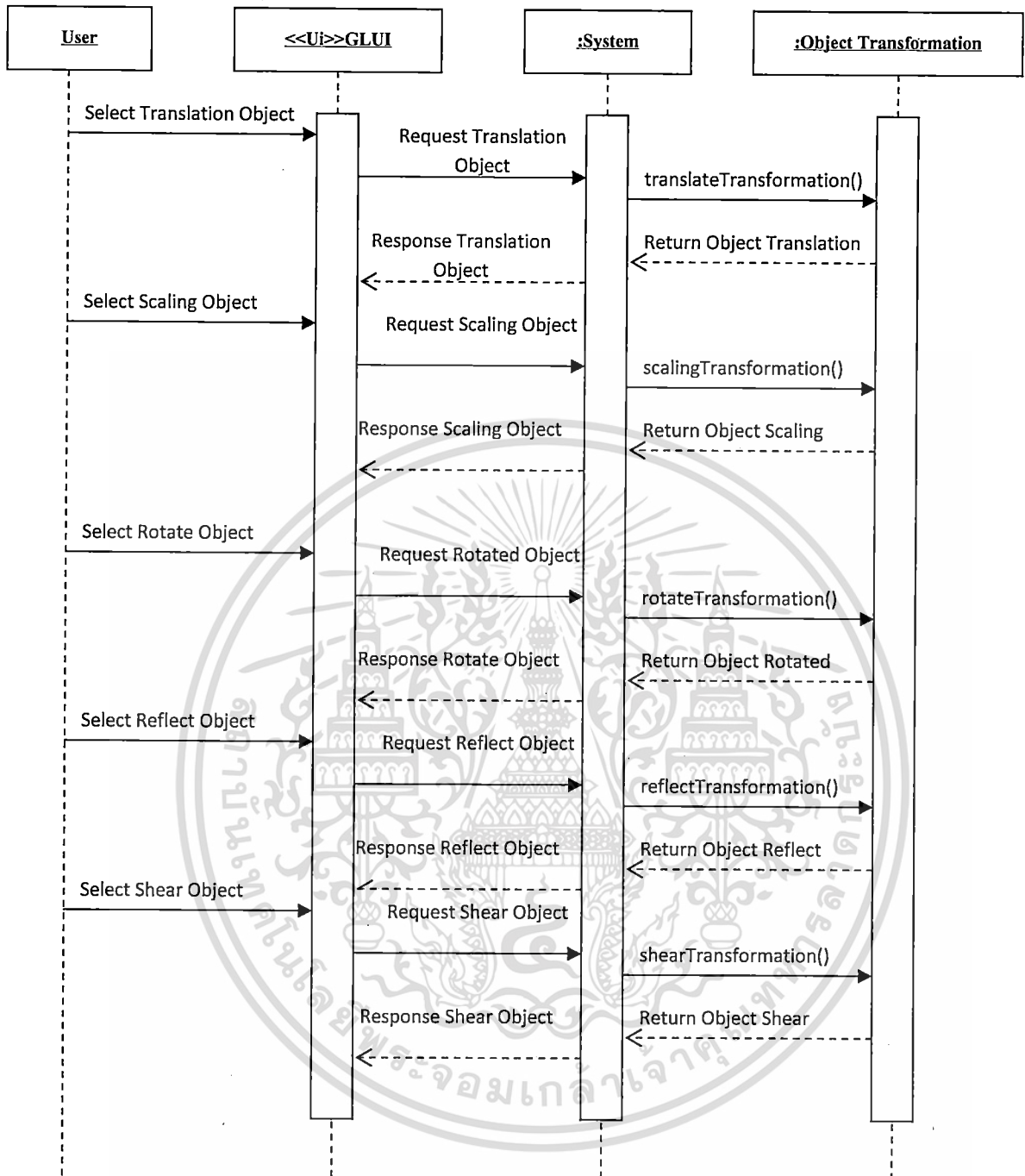


รูปที่ 3.9 Sequence Diagram Create New Object



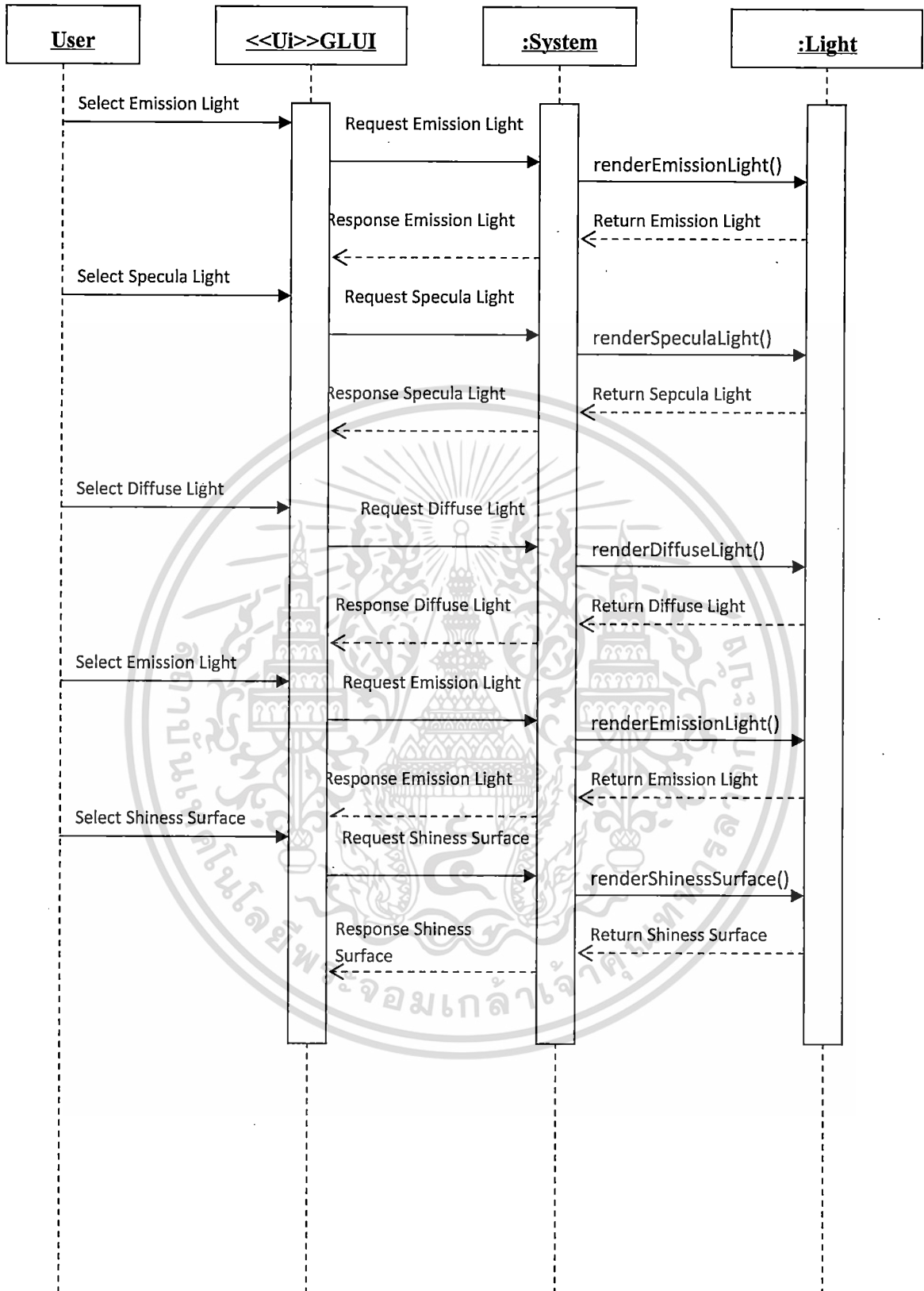
รูปที่ 3.10 Sequence Diagram Select Lesson

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.11 Sequence Diagram Transformation Object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.12 Sequence Diagram Render Light for Object

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

การพัฒนาและทดสอบระบบงาน

ในบทที่ 4 นี้จะกล่าวถึง การพัฒนาระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล ซึ่งจะอธิบายถึงสภาพแวดล้อมต่างๆที่ใช้ในการพัฒนาระบบ เช่น ระบบปฏิบัติการ ภาษาที่ใช้ในการพัฒนาระบบ และเครื่องมือที่ใช้ในการพัฒนาระบบการทดสอบและระบบงาน ภายใต้เงื่อนไขต่างๆ เพื่อจำลองการใช้งานจริงของระบบที่ได้ออกแบบเอาไว้ก่อนหน้านี้

4.1 เครื่องมือที่ใช้ในการพัฒนาระบบ

ระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอลอาศัยสภาพแวดล้อมในการพัฒนาระบบดังนี้

4.1.1 ฮาร์ดแวร์: เครื่องคอมพิวเตอร์ในการพัฒนาทดสอบระบบ

- Notebook: ที่มีหน่วยประมวลผลกลางรุ่น AMD Dual-Core Processorm E-350 (1.6 GHz)
- RAM: 2.0 GB
- OS: Window 7

4.1.2 ซอฟต์แวร์: ซอฟต์แวร์และภาษาโปรแกรมที่ใช้ในการพัฒนาและทดสอบระบบ

- Programming Languages: C/C++
- Software: Microsoft visual Studio 2010 Express Edition
- Library: OpenGL

4.2 รายละเอียดโปรแกรมที่ได้พัฒนาในเชิงเทคนิค

4.2.1 Input/ Output Specification

1. ผู้เรียนจะทำการเลือกบทเรียนที่ต้องการ จากนั้นระบบจะทำการเตรียมส่วนติดต่อเพื่อส่งค่าของพารามิเตอร์ไปยังฟังก์ชันที่เรียกใช้
2. ระบบจะทำการคำนวณเมตริกซ์ที่มีการเปลี่ยนแปลงไปและรูปทรงสุดท้ายเมื่อผู้ใช้ทำการปรับแต่งพารามิเตอร์ซึ่งเชื่อมต่อกับส่วนติดต่อกับผู้ใช้งาน

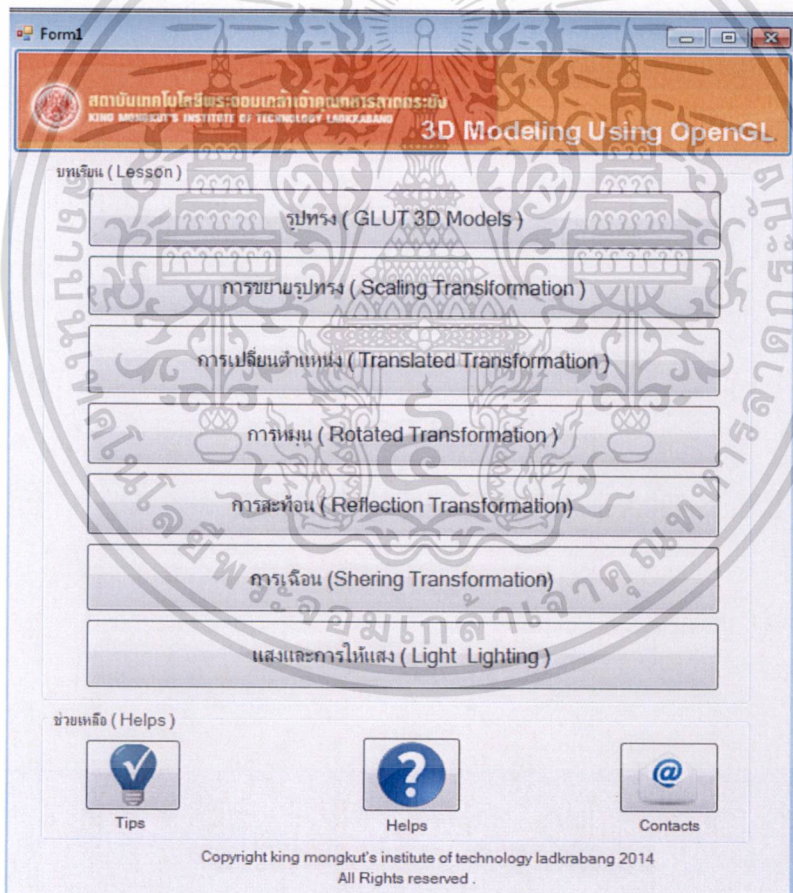
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. ระบบทำการแสดงผลรูปทรงที่มีลักษณะของวงกลมของสี่ตามทฤษฎีของฟง หากมีการเปิดบทเรียนเกี่ยวกับแสงและการให้แสง โดยจะคำนวณตามพารามิเตอร์ที่ผู้ใช้ป้อนเข้าสู่ระบบด้วยส่วนติดต่อกับผู้ใช้ที่ระบบเตรียมไว้ให้
4. เลือกบทเรียนที่ต้องเรียนรู้ถัดไป โดยกดปีกระบบและเลือกบทเรียนจากเมนูหลักที่แสดง

4.2.2 Function Specification

1. ฟังก์ชันหลักของระบบ

ฟังก์ชันนี้ทำหน้าที่แสดงบทเรียนพื้นฐานของระบบ โดยจะทำการแสดงบทเรียนทั้งหมดสำหรับผู้ใช้งานระบบดังกล่าว โดยผู้ใช้งานสามารถเลือกเรียนได้ตามลำดับ หรืออาจเลือกบางบทเรียนที่ต้องการเรียนหรือเลือกบทเรียนที่ต้องการกลับมาทบทวนความรู้

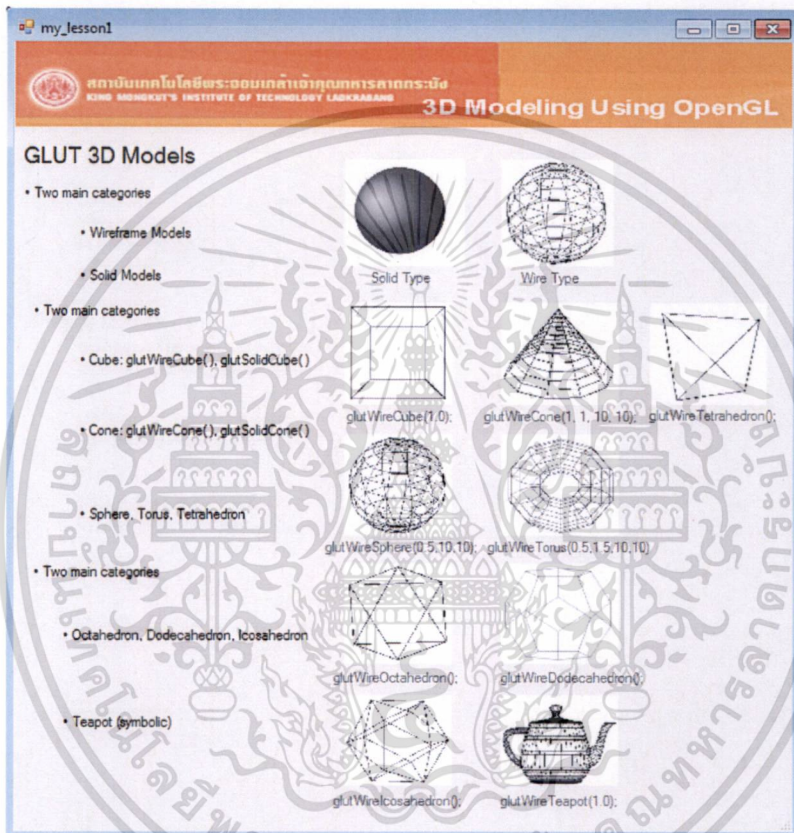


รูปที่ 4.1 หน้าจอสำหรับเลือกบทเรียนของระบบระบบสนับสนุนการเรียนการสร้างโมเดลสามมิติ โดยใช้โอเพนจีแอล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. บทเรียนเกี่ยวกับรูปทรง (GLUT 3 D Models)

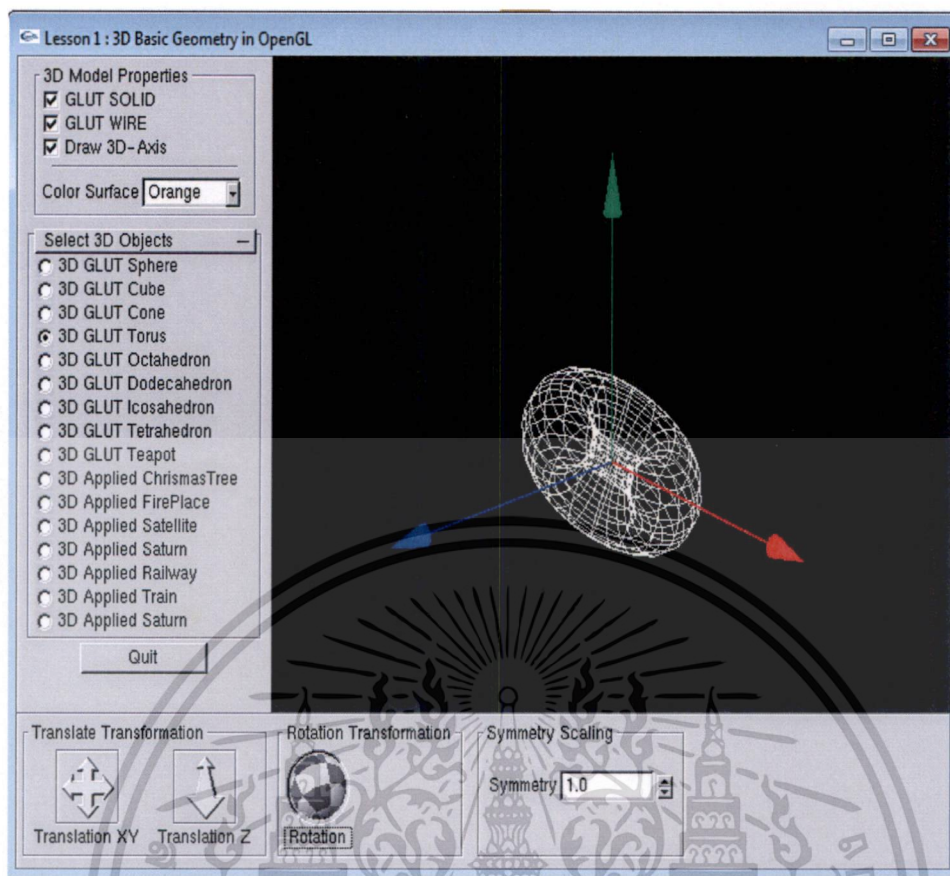
ใช้สำหรับเรียนรู้การสร้างโมเดลพื้นฐานแบบสามมิติซึ่งซึ่งโอเพนจีแอลเตรียมไว้ให้ บทเรียนดังกล่าวจะเป็นบทเรียนแบบอินเตอร์แอคทีฟ ซึ่งผู้เรียนสามารถเลือกสร้างรูปทรงต่างๆ พร้อมทั้งหมุนหรือเคลื่อนย้ายอ็อบเจกต์ที่สร้างได้แบบทันที



รูปที่ 4.2 หน้าจอสำหรับบทเรียนเกี่ยวกับรูปทรง (GLUT 3D Models)

3. บทเรียนเกี่ยวกับรูปทรงแบบอินเตอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการสร้างรูปทรงพื้นฐานได้ทันทีโดยเลือกจากตัวเลือกแบบเรดิโอที่ระบบได้เตรียมไว้ให้ จากนั้นระบบจะทำการสร้างรูปที่ต้องการผ่านหน้าจอ นอกจากนี้ผู้เรียนยังสามารถทำการสำรวจรูปทรงสามมิติที่ได้สร้างขึ้นด้วยการหมุนหรือเคลื่อนย้ายไปรอบๆ ได้ด้วย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยู โอเกี่ยวกับรูปทรง (GLUT 3D Models)

3.บทเรียนเกี่ยวกับการย่อหรือขยายรูปทรงสามมิติ (Scaling Transformation) ใช้สำหรับเรียนรู้เกี่ยวกับการทำการย่อหรือขยายรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

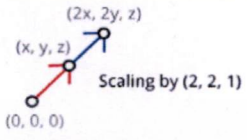
my_lesson2

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

3D Modeling Using OpenGL

Scaling Transformation

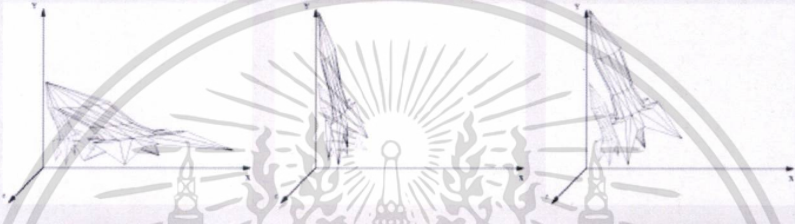
A scale transformation scales each of a vector's components by a (different) scalar. It is commonly used to shrink or stretch a vector and matrix that scales a given vector by (SX,SY,SZ) as demonstrated below.



Scaling Vector

$$\begin{bmatrix} SX & 0 & 0 & 0 \\ 0 & SY & 0 & 0 \\ 0 & 0 & SZ & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} SX \cdot x \\ SY \cdot y \\ SZ \cdot z \\ 1 \end{pmatrix}$$

Scaling Matrix Transformation



Scaling X-axis: `glScaled(2, 1, 1)`

Scaling Y-axis: `glScaled(1, 2, 1)`

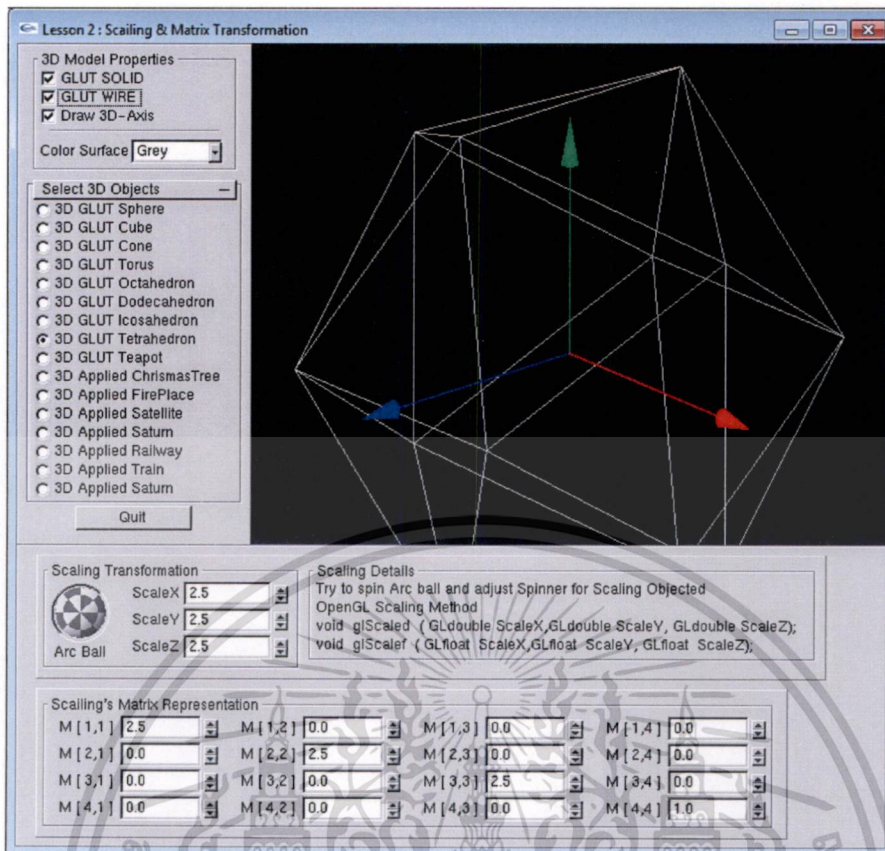
Scaling X and Y-axis: `glScaled(2, 2, 2)`

OpenGL Method : `void glScaled(GLdouble x, GLdouble y, GLdouble z)`
 Parameters : Specify scale factors along the x, y, and z axes, respectively.

รูปที่ 4.4 หน้าจอสำหรับบทเรียนเกี่ยวกับการย่อหรือขยายรูปทรง (Scaling Transformation)

4. บทเรียนเกี่ยวกับการย่อหรือขยายรูปทรงสามมิติแบบอินเตอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการย่อหรือขยายรูปทรงสามมิติ โดยใช้ตัวปรับค่าพารามิเตอร์ Scale X, Scale Y และ Scale Z และเมื่อทำการส่งค่าพารามิเตอร์ไปสู่ระบบเมตริกซ์ที่ได้จะเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.5 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับย่อหรือขยายรูปทรง

5. บทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติ (Translation Transformation)

ใช้สำหรับเรียนรู้เกี่ยวกับการทำการเปลี่ยนตำแหน่งของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

my_lesson3

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONSIEUR'S INSTITUTE OF TECHNOLOGY LADKRAKANG

3D Modeling Using OpenGL

Translation Transformation

Translation is a function that moves every point a constant distance in a specified direction. These are matrix representation and Translation moves a vector a certain distance in a certain direction.

Translation by (1, 2, 3)

Translation Vector

$$\begin{bmatrix} 1 & 0 & 0 & X \\ 0 & 1 & 0 & Y \\ 0 & 0 & 1 & Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x + X \cdot 1 \\ y + Y \cdot 1 \\ z + Z \cdot 1 \\ 1 \end{pmatrix}$$

Translation Matrix

Translate X and Y Axis
glTranslated(2,2,0)

Rotate Then Translated

Translate then Rotate

The Order Transformation : Scaling > Rotate > Translate

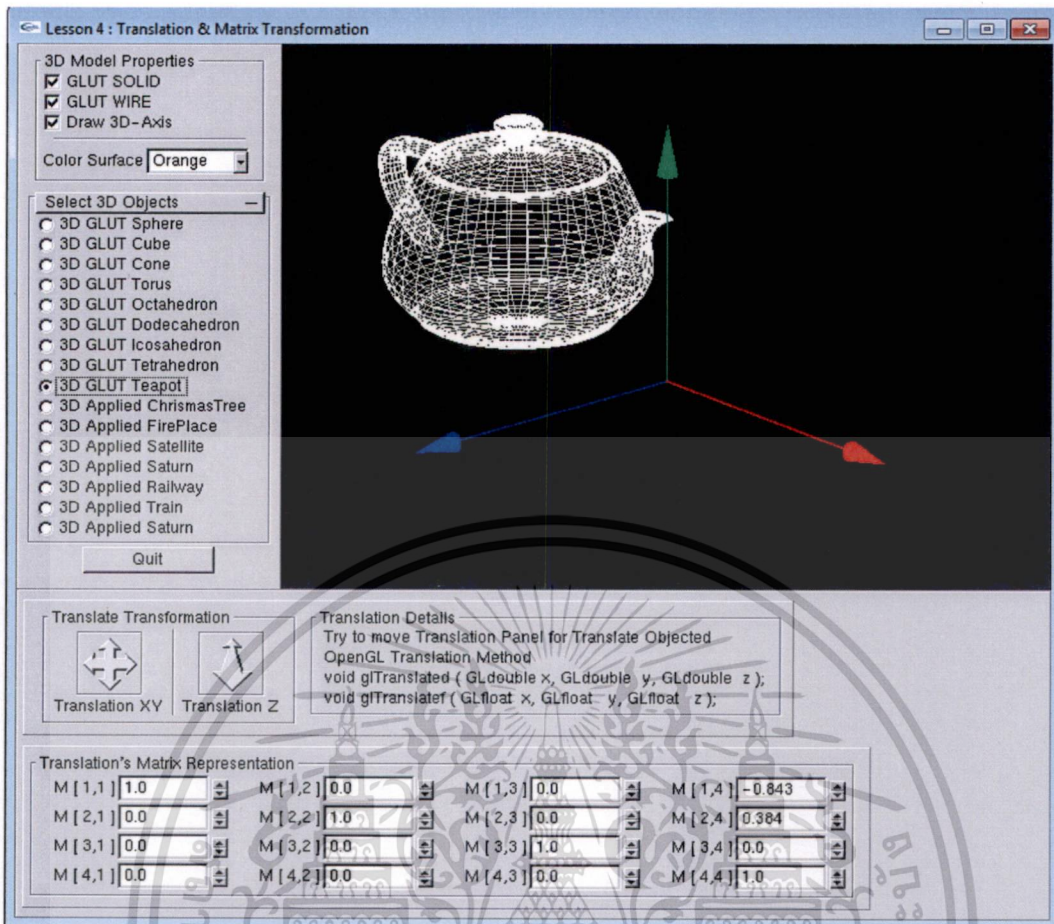
OpenGL Method : void glTranslated(GLdouble x, GLdouble y, GLdouble z);
void glTranslatef(GLfloat x, GLfloat y, GLfloat z);

Parameters : Specify the x, y, and z coordinates of a translation vector.

รูปที่ 4.6 หน้าจอสำหรับบทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติ

6. บทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติแบบอินเตอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการเปลี่ยนตำแหน่งรูปทรงสามมิติ โดยใช้ตัวรับ Translation Panel แครกเมาส์ไปตามแกนสามมิติ และเมื่อทำการส่งค่าพารามิเตอร์ไปสู่ระบบเมตริกซ์ที่ได้จะเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.7 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับการเปลี่ยนตำแหน่งของรูปทรงสามมิติ

7. บทเรียนเกี่ยวกับการหมุนตามแนวแกนของรูปทรงสามมิติ (Rotated Transformation)

ใช้สำหรับเรียนรู้เกี่ยวกับการทำการหมุนของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

my_lesson4


สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONKUT'S INSTITUTE OF TECHNOLOGY LADKRAKANG

3D Modeling Using OpenGL

Rotation Transformation

A rotation transformation rotates a vector around the origin (0,0,0) using a given axis and angle. Let's do a small experiment. Put your thumb up against your monitor and try rotating your hand around it. The object, your hand, is rotating around your thumb: the rotation axis. The further you rotate your hand away from its initial position, the higher the rotation angle. These are rotation vector and matrix representation.


Rotation around X-axis:



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta & 0 \\ 0 & \sin \theta & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} x \\ \cos \theta \cdot y - \sin \theta \cdot z \\ \sin \theta \cdot y + \cos \theta \cdot z \\ 1 \end{pmatrix}$$

Rotated X


Rotation around Y-axis:



$$\begin{bmatrix} \cos \theta & 0 & \sin \theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \theta & 0 & \cos \theta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \theta \cdot x + \sin \theta \cdot z \\ y \\ -\sin \theta \cdot x + \cos \theta \cdot z \\ 1 \end{pmatrix}$$

Rotated Y

Rotation around Z-axis:



$$\begin{bmatrix} \cos \theta & -\sin \theta & 0 & 0 \\ \sin \theta & \cos \theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} = \begin{pmatrix} \cos \theta \cdot x - \sin \theta \cdot y \\ \sin \theta \cdot x + \cos \theta \cdot y \\ z \\ 1 \end{pmatrix}$$

Rotated Z

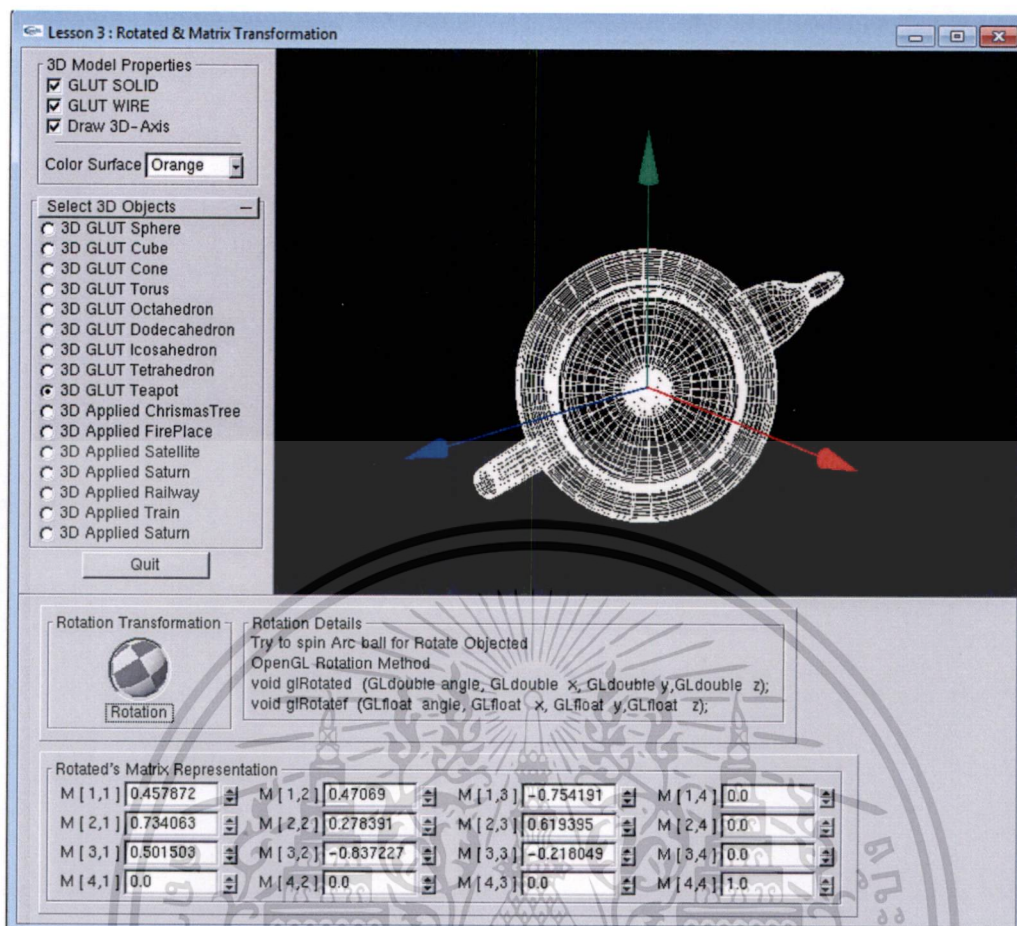
OpenGL Method : void glRotated(GLdouble angle, GLdouble x, GLdouble y, GLdouble z);
void glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat z);

Parameters : Angle Specifies the angle of rotation, in degrees.
: Specify the x, y, and z coordinates of a vector, respectively.

รูปที่ 4.8 หน้าจอสำหรับบทเรียนเกี่ยวกับการหมุนรูปทรงสามมิติ (Rotated Transformation)

8. บทเรียนเกี่ยวกับการหมุนสามมิติแบบอินเตอร์แอคทีฟ - ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการเปลี่ยนตำแหน่งรูปทรงสามมิติ โดยใช้อาร์บอลหมุนไปตามแกนสามมิติ และเมื่อทำการส่งค่าพารามิเตอร์ไปสู่ระบบเมตริกซ์ที่ได้จะเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.9 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับการหมุนรูปทรงสามมิติ (Rotated Transformation)

9. บทเรียนเกี่ยวกับการสะท้อนของรูปทรงสามมิติ (Reflection Transformation)

ใช้สำหรับเรียนรู้เกี่ยวกับการสะท้อนของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

my_lesson88

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONSIEUR'S INSTITUTE OF TECHNOLOGY LADKRAKANG

3D Modeling Using OpenGL

Reflection

In mathematics, a reflection (also spelled reflexion) is a mapping from a Euclidean space to itself that is an isometry with a hyperplane as a set of fixed points; this set is called the axis (in dimension 2) or plane (in dimension 3) of reflection.

x-axis ($y = 0$)

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Original
Reflected

xy-plane

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Reflections in X-axis

y-axis ($x = 0$)

$$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

xy-plane

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Reflections in Y-axis

Reflections in Y-axis

Mirror Reflection produces a mirror image of an object. In Matrix Form:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ -z \\ 1 \end{bmatrix}$$

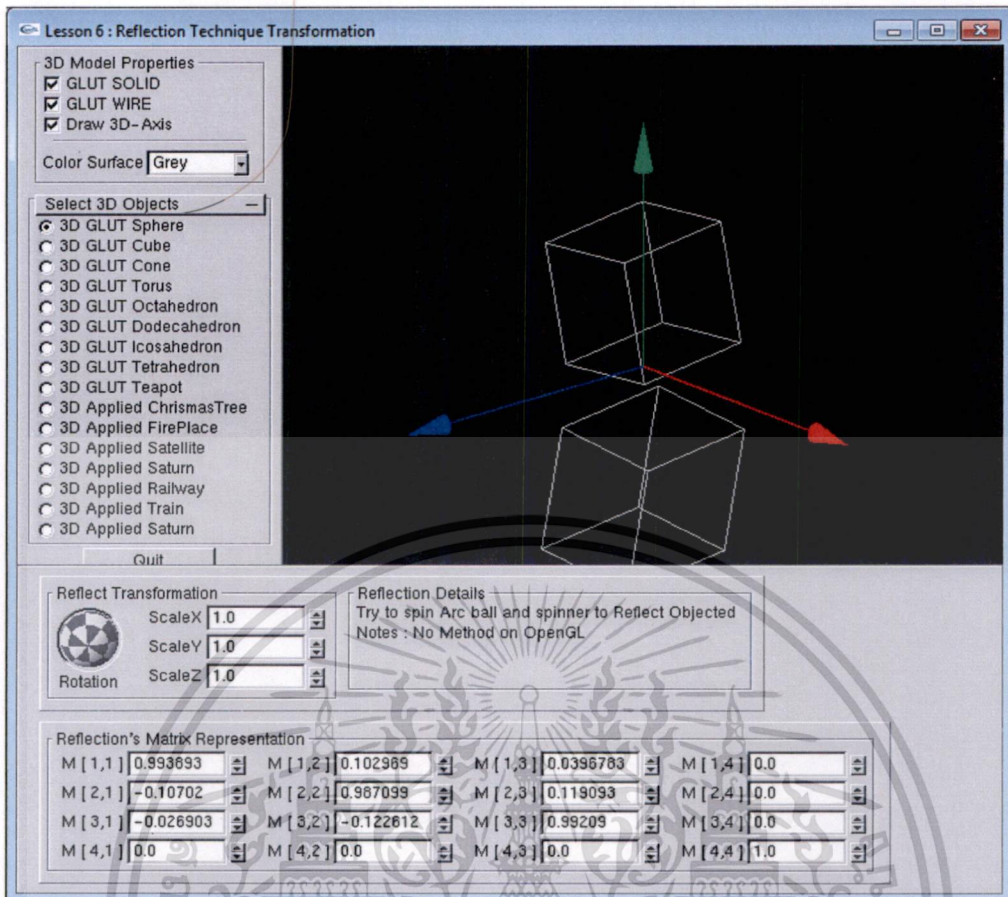
Reflections through the xz and the yz planes are defined similarly.

OpenGL Method : Use Shering Matrix

รูปที่ 4.10 หน้าจอสำหรับบทเรียนเกี่ยวกับการสะท้อนรูปทรงสามมิติ (Reflect Transformation)

10. บทเรียนเกี่ยวกับการสะท้อนของรูปทรงสามมิติแบบอินเตอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการสะท้อนรูปทรงสามมิติ โดยใช้อาร์คบอลหมุนไปตามแกนสามมิติ และเมื่อทำการส่งค่าพารามิเตอร์ ไปสู่ระบบเมตริกซ์ที่ได้จะเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.11 หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูโอสำหรับบทเรียนเกี่ยวกับการสะท้อนรูปทรงสามมิติ (Reflect Transformation)

11. บทเรียนเกี่ยวกับการเฉือนของรูปทรงสามมิติ (Shearing Transformation)

ใช้สำหรับเรียนรู้เกี่ยวกับการเฉือนของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

my_lesson89

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONKUT'S INSTITUTE OF TECHNOLOGY LADKRAKANG

3D Modeling Using OpenGL

Shearing

In plane geometry, a shear mapping is a linear map that displaces each point in fixed direction, by an amount proportional to its signed distance from a line that is parallel to that direction. An example is the mapping that takes any point with coordinates (x,y) to the point $(x+2y,y)$. In this case, the displacement is horizontal, the fixed line is the x-axis, and the signed distance is the y coordinate. Note that points on opposite sides of the reference line are displaced in opposite directions

original x - shear y - shear Shearing apply in 3D Objects

2D Matrix Shearing Matrix Shearing in X-axis Shearing in Y-axis Shearing in X and Y-axis

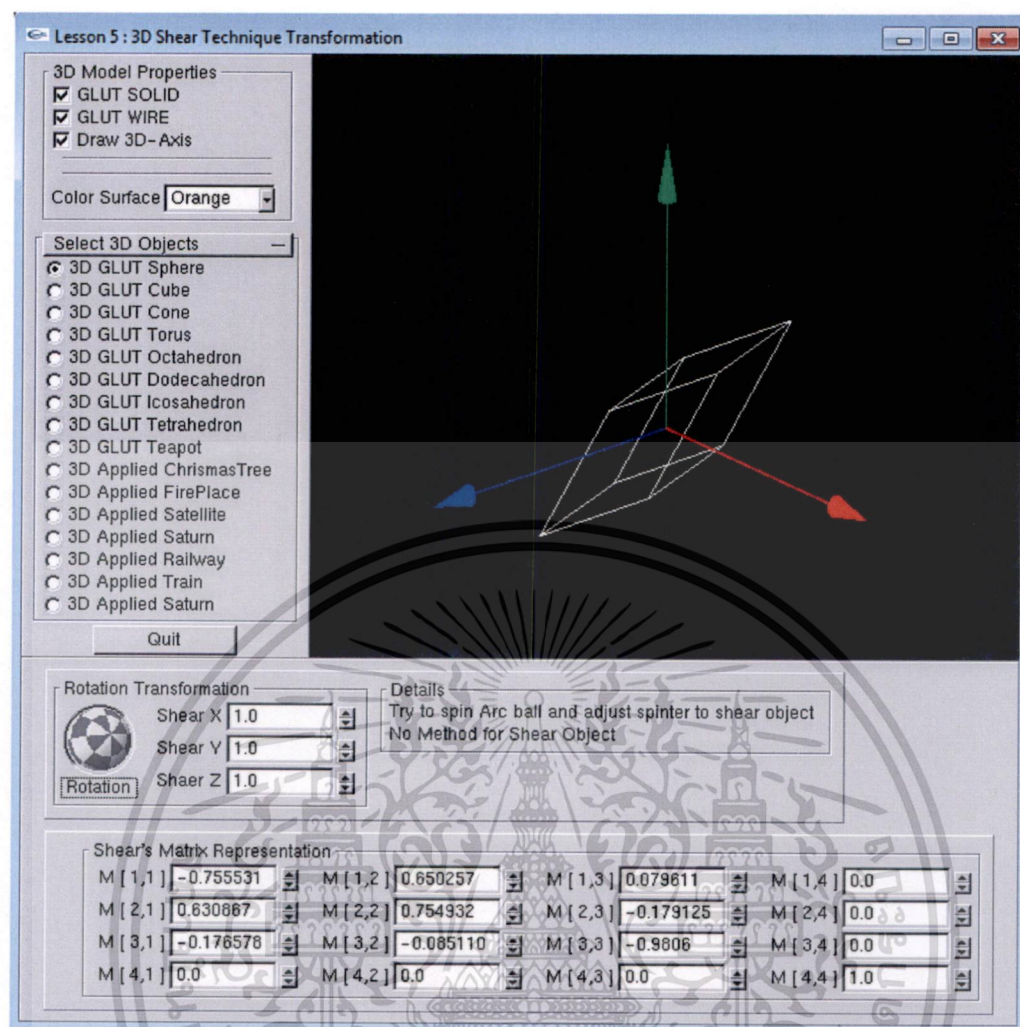
$$\begin{bmatrix} 1 & a & b & 0 \\ c & 1 & d & 0 \\ e & f & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x + ay + bz \\ cx + y + dz \\ ex + fy + z \\ 1 \end{bmatrix}$$

3D Matrix Shearing Matrix

รูปที่ 4.12 หน้าจอสำหรับบทเรียนเกี่ยวกับเงื่อนไขของรูปทรงสามมิติ (Shearing Transformation)

12. บทเรียนเกี่ยวกับการเงื่อนไขของรูปทรงสามมิติแบบอินเทอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำเงื่อนไขรูปทรงสามมิติ โดยใช้สปีนเนอร์ Shearing X, Shearing Y และ Shearing Z ทำการเงื่อนไขรูปทรงไปตามแกนสามมิติ และเมื่อทำการส่งค่าพารามิเตอร์ไปสู่ระบบเมตริกซ์ที่ได้จะเปลี่ยนแปลงไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 หน้าจอหน้าต่างส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับเงื่อนไขของรูปทรงสามมิติ (Shearing Transformation)

13. บทเรียนเกี่ยวกับการให้แสงของรูปทรงสามมิติ (Light and Lighting)
ใช้สำหรับเรียนรู้เกี่ยวกับการให้แสงของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

my_lesson6

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
KING MONSIEUR'S INSTITUTE OF TECHNOLOGY LADKRAKANG

3D Modeling Using OpenGL

Light and Lighting

OpenGL computes the color of each pixel in a final, displayed scene that's held in the frame buffer. Part of this computation depends on what lighting is used in the scene and on how objects in the scene reflect or absorb that light.

Phong shading refers to an interpolation technique for surface shading in 3D computer graphics. It is also called Phong interpolation or normal-vector interpolation shading.

FLAT SHADING PHONG SHADING

Ambient + Diffuse + Specular = Phong Reflection

Specular highlight is the bright spot of light that appears on shiny objects when illuminated (for example, see image at right). Specular highlights are important in 3D computer graphics, as they provide a strong visual cue for the shape of an object and its location with respect to light sources in the scene. (See images 3 Specular)

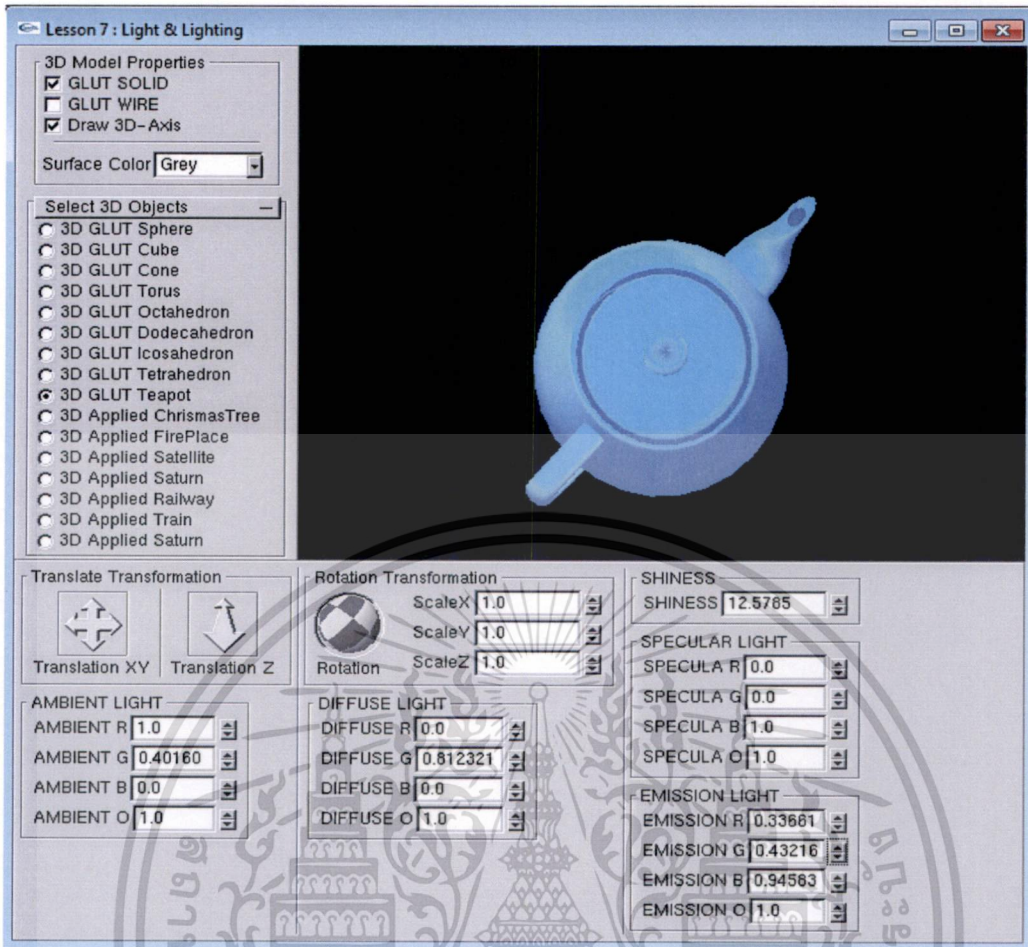
Diffuse reflection is the reflection of light from a surface such that an incident ray is reflected at many angles rather than at just one angle as in the case of specular reflection. An illuminated ideal diffuse reflecting surface will have equal luminance from all directions which lie in the half-space adjacent to the surface. (See images 2 Diffuse).

Ambient light doesn't come from a specific direction, the rays have light have bounced around so much that objects are evenly lit from all sides. Ambient lights are almost always used in combination with other types of lights. (See images 1 Ambient)

รูปที่ 4.14 หน้าจอสำหรับบทเรียนเกี่ยวกับแสงและการให้แสงของรูปทรงสามมิติ (Light and Lighting)

14. บทเรียนเกี่ยวกับการให้แสงของรูปทรงสามมิติแบบอินเทอร์แอคทีฟ – ผู้เรียนสามารถใช้งานส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอเพื่อทำการให้แสงแก่รูปทรงสามมิติ โดยใช้สปีนเตอร์ในกลุ่มของ Ambient, Diffuse, Specular, Emission และ Shiness ทำการให้แสงแต่ละรูปแบบแก่รูปทรงเพื่อทำการศึกษการให้แสงกับวัตถุตามทฤษฎีของฟง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.15 หน้าจอ หน้าจอส่วนติดต่อกับผู้ใช้แบบจีแอลยูไอสำหรับบทเรียนเกี่ยวกับแสงและการให้แสงของรูปทรงสามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

บทสรุป

การค้นคว้าแบบอิสระเรื่องระบบสนับสนุนการเรียนการสอนสร้างโมเดลสามมิติโดยใช้โอเพนจีแอล เป็นการทดลองโดยใช้ไลบรารีโอเพนจีแอลในการแสดงผลภาพสามมิติ โดยการส่งคำสั่งควบคุมการวาดภาพไปยังอุปกรณ์ฮาร์ดแวร์การประมวลผลภาพ โอเพนจีแอลสามารถใช้ได้ในหลายระบบคอมพิวเตอร์ในการเขียนโปรแกรมด้านคอมพิวเตอร์กราฟิกส์ โดยในคลังโปรแกรมที่ใช้มาตรฐานนี้จะมีชุดคำสั่งมีมากกว่า 250 ชุดคำสั่งเพื่อช่วยในการสร้างวัตถุ แปลงวัตถุ และสร้างภาพโดยให้แสงและเงา ทำโอเพนจีแอลได้รับความนิยมในอุตสาหกรรมผลิตแอนิเมชันและวิดีโอเกม ซึ่งโอเพนจีแอลนั้นเป็นไลบรารีที่มีมาตรฐานที่ถูกควบคุมโดยคณะกรรมการทบทวนสถาปัตยกรรมโอเพนจีแอล หรือ โอเพนจีแอล เออาร์บี (OpenGL Architecture Review Board:ARB) ซึ่งประกอบไปด้วยผู้ผลิตฮาร์ดแวร์และซอฟต์แวร์ รวมถึงองค์กร อย่าง เอทีไอ คอมแพค อีวาน ซูเธอร์แลนด์ ฮิวเลตต์-แพคการ์ด ไอบีเอ็ม อินเทล อินเทอร์เน็ต เอ็นวีเดีย ไมโครซอฟท์ และ ซิลิกอนกราฟิกส์ ซึ่งทำให้ไลบรารีโอเพนจีแอลนี้มีมาตรฐานสูงและมีความน่าเชื่อถือเป็นอย่างมาก นอกจากนี้ไลบรารีดังกล่าวยังเป็นโอเพนซอสที่เปิดโอกาสให้ผู้พัฒนานำไปใช้หรือพัฒนาต่อโดยไม่เสียค่าใช้จ่ายอีกด้วย

5.1 ผลการทดลอง

จากการทดลองนี้ผู้เขียนได้ทำการสร้างบทเรียนเกี่ยวกับ โมเดลสามมิติด้วยวัตถุพื้นฐานที่โอเพนจีแอลกำหนดขึ้นมา และนำไปแปลงและสร้างเป็นรูปต่างๆ โดยใช้กรรมวิธีการสร้างรูปทรงดังที่แสดงไว้ในบทที่ 4 นอกจากนี้ผู้เขียนประยุกต์ใช้ทฤษฎีแสงของฟงจำลองแสงในโลกของสามมิติซึ่งเลียนแบบในโลกของความเป็นจริงขึ้นมาใช้ในงานทดลองนี้อีกด้วย ซึ่งหลังจากการทดลองดังกล่าวเราสามารถสรุปได้ว่าไลบรารีโอเพนจีแอลสามารถนำไปประยุกต์ใช้ในการทำงานด้านต่างๆ ได้อย่างมีประสิทธิภาพและมีมาตรฐานที่สูงมาก ซึ่งผู้ใช้งานสามารถดาวน์โหลดไปใช้งานหรือพัฒนาต่อได้โดยไม่เสียค่าใช้จ่าย

5.2 การประยุกต์ใช้งาน

การประยุกต์ใช้งานไลบรารีของโอเพนจีแอลนอกจากที่ได้นำเสนอไว้ในบทที่ 3 แล้ว เรายังสามารถนำไปประยุกต์ใช้ในการทำงานต่างๆ ตั้งแต่งานกราฟฟิกทั่วไป เช่น การวาดภาพ 2 มิติ และ 3 มิติ การสร้างกราฟ 2 มิติและ 3 มิติ การสร้างงานอนิเมชัน การใช้ในการแสดงผลทางด้านวิทยาศาสตร์ หรืออาจใช้เป็นสื่อในการเรียนการสอนก็สามารถทำได้ ส่วนการใช้งานระดับสูงของโอเพนจีแอลนั้นอาจนำไปใช้ในการสร้างโมเดลจำลองต่างๆ หรือใช้สร้างแอปพลิเคชันประยุกต์เพื่อช่วยในการสร้างผลงานต่างๆ ทางด้านกราฟฟิกโดยสั่งให้โอเพนจีแอลทำงานอีกทีก็ได้ ดังเช่นโปรแกรมต่างๆ ที่มีในปัจจุบัน ได้แก่ MAYA 3D MAX หรือโปรแกรมออกแบบโครงสร้างจำพวก CAD เป็นต้น



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

ไพศาล โมลิสกุลมงคล. 2550. คอมพิวเตอร์กราฟิกส์. กรุงเทพฯ : ไทยเจริญการพิมพ์.

ไพศาล โมลิสกุลมงคล. มปป. คอมพิวเตอร์กราฟิกส์ใช้ OpenGL. [Online]. เข้าถึงได้จาก :

http://www.computer.cmru.ac.th/trid/CG/Slide/cg_ch03.ppt

วิวัฒน์ วิทยชานาญกุล. 2547. การสร้างภาพสามมิติโดยใช้ OpenGL. [Online]. เข้าถึงได้จาก :

<http://withawat.files.wordpress.com/2009/05/opengl2.pdf>

Ali BaderEddin 2008. **GLUI Sub Window Template**. [Online]. Available :

<http://mycodeblog.com/tag/subwindow/>

David Fleet 2005. **Basic Light and Reflection**. [Online]. Available :

<http://www.cs.toronto.edu/~kyros/courses/2503/Handouts/Phong418Ch8.pdf>

Doug James 2010. **3D Transformation**. [Online]. Available :

<http://www.cs.cornell.edu/courses/cs4620/2010fa/lectures/03transforms3d.pdf>

Alexander Overvoorde 2012. **OpenGL Transformation**. [Online]. Available :

<https://open.gl/transformations/>

Foley and Van Dam n.d. **2D Geometrical Transformations**. [Online]. Available :

http://www.cs.brandeis.edu/~cs155/Lecture_06.pdf

Paul Rademacher 2006. ส่วนเชื่อมต่อประสานแบบ GLUT-Based. [Online]. Available:

http://202.44.14.219/~isurnich/ftp/CG/GLUI/html/Siam%20glui_manual.htm/

Zeyang Li n.d. OpenGL Transformations [Online]. Available :

<http://www.cs.cmu.edu/afs/cs/academic/class/15462-s09/www/lec/03/lec03a.pdf>

The Khronos Group 1997. **Load OpenGL Function**. [Online]. Available :

https://www.opengl.org/wiki/Load_OpenGL_Functions/

The Khronos Group 1997. **OpenGL Development Setup**. [Online]. Available :

https://www.opengl.org/wiki/Getting_Started/

Wolfram Mathematica 2014. **Understand 3D Translation**. [Online]. Available :

<http://demonstrations.wolfram.com/Understanding3DTranslation/>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้