과제4 16011024 주수헌

1. 좌하단 (은면, 후면 존재)

-후면과 은면을 제거하지 않은 상태이다

-시점 반대 방향을 향하는 주전자 바닥까지 보이게 된다

2. 우하단 (은면제거)

-주전자 뒷면을 이면만으로 대신하기 때문에 이면이 그래도 남게 된다

-은면 제거시 주전자 앞쪽의 표면과 뒤쪽이면의 깊이가 비교되어

뚜껑 가장자리 빈 부분이 앞쪽 또는 뚜껑 표면에 가려지지 않아서 주전자 뒤쪽 이면이 보이게 된다

3. 좌상단 (후면제거)

-바닥 면 자체는 보이지 않는다

-주전자 뚜껑, 손잡이 부분이 사실적이지 않다

-뚜껑 손잡이에 가려져 보이지 않아야 할 뚜껑의 뒷부분이 보인다

(손잡이와 뚜껑 뒷부분 둘 다 후면이 아니라서 먼저 손잡이가 그려지고 그 위에 뚜껑이 덧칠해진 것이다)

-주전자뚜껑은 닫을 수 있는 크기보다 작게 모델링 되는 것이여서 주전자가 뚫려 보이는 것이다

원래는 주전자 뒷면의 안쪽이 보여야 하는데 후면이 제거 돼서 배경이 보이는 것이다

4. 우상단 (은면제거, 후면제거)

-후면제거한 상태에서 은면 제거를 했기에 주전자 손잡이와 뚜껑사이 부분이 명확해 졌다.

-후면 뚜껑부분은 후면을 제거하고 은면제거를 하는데 후면제거를 한부분은 이미제거 되었기 때문에 은면제거가 입력되지 않는다

\*결론

-사람 눈으로 사실적 표현을 위해 적합한 방식은 은면제거만 하는 것이다.

사람의 시각에서는 은면이 아닌 부부만 보게 된다. 은면이란 다른면에 가려서 보이지 않는 면을 말한다. 후면도 사람 시각에 반대 방향이지만 후면도 제거하게 되면 후면의 이면도 제거하게 되어 위와 같이 주전자 가장자리 뚫린 부분에서 배경이 보이게 된다. 이럴 경우 우리의 시각에서는 후면부분의 이면이 보여야한다. 그러기에 후면과 은면 둘 다 제거하지 않고 은면부분만 제거하는 것이 사실적 표현을 위해 적합한 방식이다

\*전체코드

#include <GL/glut.h>

#include <GL/gl.h>

#include <GL/glu.h>

int Width, Height;

void InitLight() {

GLfloat mat\_diffuse[] = { 0.5, 0.4, 0.3, 1.0 };

GLfloat mat\_specular[] = { 1.0, 1.0, 1.0, 1.0 };

GLfloat mat\_ambient[] = { 0.5, 0.4, 0.3, 1.0 };

GLfloat mat\_shininess[] = { 50.0 };

GLfloat light\_specular[] = { 1.0, 1.0, 1.0, 1.0 };

GLfloat light\_diffuse[] = { 0.8, 0.8, 0.8, 1.0 };

GLfloat light\_ambient[] = { 0.3, 0.3, 0.3, 1.0 };

GLfloat light\_position[] = { -3, 2, 3.0, 0.0 };

glShadeModel(GL\_SMOOTH);

glEnable(GL\_LIGHTING);

glEnable(GL\_LIGHT0);

glLightfv(GL\_LIGHT0, GL\_POSITION, light\_position);

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, light\_diffuse);

glLightfv(GL\_LIGHT0, GL\_SPECULAR, light\_specular);

glLightfv(GL\_LIGHT0, GL\_AMBIENT, light\_ambient);

glMaterialfv(GL\_FRONT, GL\_DIFFUSE, mat\_diffuse);

glMaterialfv(GL\_FRONT, GL\_SPECULAR, mat\_specular);

glMaterialfv(GL\_FRONT, GL\_AMBIENT, mat\_ambient);

glMaterialfv(GL\_FRONT, GL\_SHININESS, mat\_shininess);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

glOrtho(-1.0, 1.0, -1.0, 1.0, -1.0, 2.0);

}

void MyDisplay() {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glMatrixMode(GL\_MODELVIEW);

glLoadIdentity();

gluLookAt(0.0, 0.4, 0.5, 0.0, -0.5, -1.0, 0.0, 1.0, 0.0);

//은면, 후면 존재

glViewport(0, 0, (GLsizei)Width / 2, (GLsizei)Height / 2); //좌하단

glutSolidTeapot(0.58);

//은면제거

glEnable(GL\_DEPTH\_TEST); //은면제거모드 실행

glViewport((GLsizei)Width / 2, 0, (GLsizei)Width / 2, (GLsizei)Height / 2);//우하단

glutSolidTeapot(0.58);

//후면제거

glEnable(GL\_CULL\_FACE); //후면제거 모드 활성화

glDisable(GL\_DEPTH\_TEST); //은면제거모드 비활성화

glFrontFace(GL\_CW); //'정점을 시계방향으로 제시한 면'을 표면을 설정

glCullFace(GL\_BACK); //최종적 후면제거 실행함수

glViewport(0, (GLsizei)Height / 2, (GLsizei)Width / 2, (GLsizei)Height / 2);//좌상단

glutSolidTeapot(0.58);

//은면제거 후면제거

glEnable(GL\_DEPTH\_TEST); //은면제거모드 다시 활성화

glViewport((GLsizei)Width / 2, (GLsizei)Height / 2, (GLsizei)Width / 2, (GLsizei)Height / 2);//우상단

glutSolidTeapot(0.58);

glFlush();

}

void MyReshape(int w, int h) {

Width = w;

Height = h;

}

int main(int argc, char\*\* argv) {

Width = 400;

Height = 400;

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGBA | GLUT\_DEPTH);

glutInitWindowSize(Width, Height);

glutInitWindowPosition(0, 0);

glutCreateWindow("과제4 16011024 주수헌"); //윈도우 생성

glClearColor(0.5, 0.5, 0.5, 0.0);

InitLight();

glutDisplayFunc(MyDisplay);

glutReshapeFunc(MyReshape);

glutMainLoop();

return 0;

}

\*실행결과

1. 좌하단 (은면, 후면 존재)

2. 우하단 (은면제거)

3. 좌상단 (후면제거)

4. 우상단 (은면제거, 후면제거)

