pwr**POLITECHNIKA WROCŁAWSKA**

**Instytut Informatyki, Automatyki i Robotyki**

**Zakład Systemów Komputerowych**

**Wprowadzenie do grafiki komputerowej**

**Kurs: INE4234L**

**Sprawozdanie z ćwiczenia nr 3**

**Open GL - modelowanie obiektów 3-D**

|  |  |
| --- | --- |
| **Wykonał:** | **Wojciech Wójcik, 235621** |
| **Termin:** | **PT/TP 8:00-11:00** |
| **Data wykonania ćwiczenia:** | **16.11.2018** |
| **Data oddania sprawozdania:** | **2.01.2019** |
| **Ocena:** |  |

|  |
| --- |
| **Uwagi prowadzącego:** |

1. **Cel ćwiczenia**

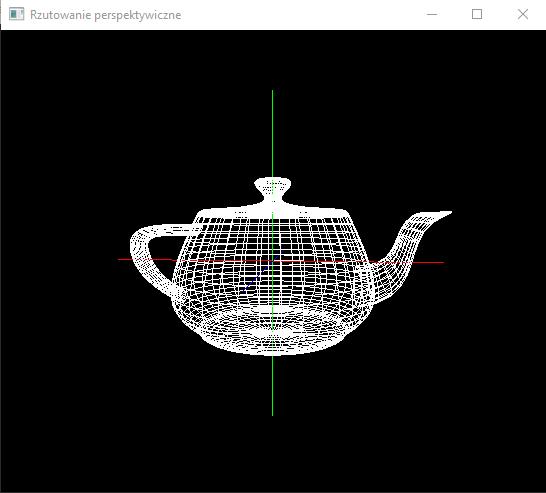
Ćwiczenie miało za zadanie pokazać jak można zrealizować prostą interakcję polegającą na sterowaniu ruchem obiektu lub obserwatora przy pomocy OpenGL oraz GLUT. Zostały wyjaśnione różnice między rzutem ortogonalnym i perspektywicznym .

1. **Zadanie do wykonania**

Należało stworzyć program w którym można modyfikować położenie czajnika w osiach w osiach y i x oraz wykonywać zbliżanie i oddalanie

W kolejnym zadaniu interakcje i ruch należało realizować przez obrót kamery obserwatora wokół obiektu.

1. **Realizacja zadania**
   1. **Obrót czajnika**

****

Pierwszą częścią zadania było wykrywanie naciśnietego klawisza myszy. Została użyta do tego funkcja *Mouse*. W niej jest porównywana zmienna *btn* do GLUT\_RIGHT\_BUTTON oraz GLUT\_LEFT\_BUTTON. Aby zapamiętać który przycisk został wciśnięty została użyta zmienna status, która jest wykorzystywana w funkcji renderowania sceny. Funkcja wygląda następująco:

void Mouse(int btn, int state, int x, int y)

{

if (btn == GLUT\_LEFT\_BUTTON && state == GLUT\_DOWN)

{

x\_pos\_old = x;

status = 1;

y\_pos\_old = y;

}

else if (btn == GLUT\_RIGHT\_BUTTON && state == GLUT\_DOWN)

{

x\_pos\_old = x;

y\_pos\_old = y;

status = 2;

}

else

status = 0; // nie został wciśnięty żaden klawisz

}

Dodatkowo, aby wykonać zadanie należało dodać do funkcji *Motion* zadeklarowaną wcześniej zmienną *delta\_y* w której zapamiętywana jest zmiana współrzędnych osi Y.

void Motion(GLsizei x, GLsizei y)

{

delta\_x = x - x\_pos\_old; // obliczenie różnicy położenia kursora myszy

x\_pos\_old = x; // podstawienie bieżącego położenia jako poprzednie

delta\_y = y - y\_pos\_old;

y\_pos\_old = y;

glutPostRedisplay(); // przerysowanie obrazu sceny

}

Końcową częścią zadania była modyfikacja kodu funkcji *RenderScene.* Wymagane było sprawdzanie czy lewy lub prawy klawisz został wciśnięty i odpowiednia modyfikacja parametrów.

if (status == 1) // jeśli lewy klawisz myszy wciśnięty

{

theta += delta\_x \* pix2angle;

thetaY += delta\_y \* pix2angle;

}

if (status == 2) // jeśli prawy klawisz myszy wciśnięty

{

viewer[2] += delta\_x \* pix2angle;

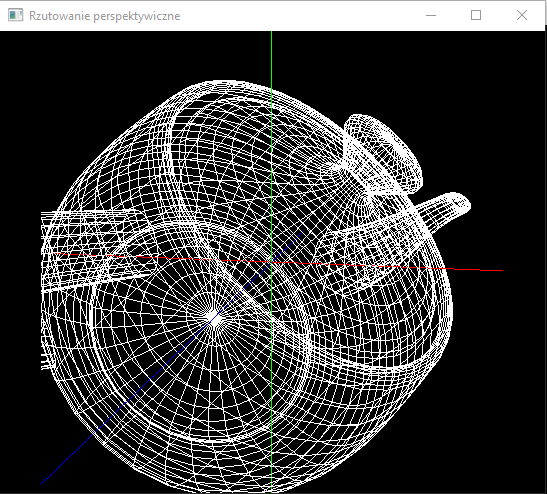
if (viewer[2] > 100)

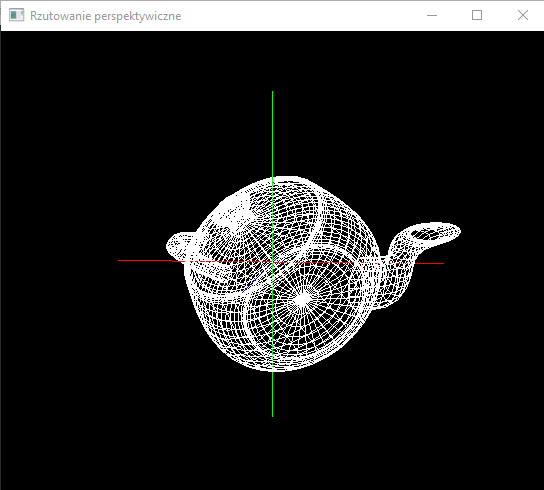
viewer[2] = 100;

else if (viewer[2] < 5)

viewer[2] = 5;

}

Efekt dokonanych modyfikacji:



* 1. **Obracanie obserwatora na przykładzie jajka**

1. **Wnioski**

Ćwiczenie pozwoliło zapoznać się z podstawami rysowania obiektów 3D. pozwoliło zrozumieć na czym polega mapowanie UV. Ćwiczenie również wprowadziło w podstawy manipulacją obiektu. Przedstawiło różnicę między wierzchołkami obiektu 3D, a liniami i ścianami. Każde z zadań pozwoliło za stopniowe zrozumienie problematyki.

1. **Zadanie domowe – układ odwzorowań iterowanych**

Zadanie polegało na pobraniu pozycji myszy przy naciśnięciu klawisza, począwszy od tego punktu trzeba było narysować drzewko zgodnie z dostarczoną tablicą, poprzez losowanie. Wynikiem tych działań jest powstałe drzewko, ta technika może znaleźć zastosowanie na przykład przy generowaniu losowej roślinności.

