

Московский авиационный институт
(Национальный исследовательский университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа № 3
по курсу «Компьютерная графика»
Тема: Основы построения фотореалистичных
изображений.

Студент: Сорокин Н.Э.
Группа: М8О-303Б-20
Преподаватель: Филиппов Г.С.
Оценка:

Москва, 2022

1. Постановка задачи

Написать программу, аппроксимирующую тело, реализованное в предыдущей работе.

2. Реализация

```
double* Operations::phong_lighting_model(double** perspective_points, double** normals, double
distance, int n, int m) {
    int N = n * m + 1;

    double k_a = 1;
    double i_a = 75;

    double k_d = 1;
    double i_d = 75;

    double k_s = 1;
    double i_s = 75;
    double alpha = 2;

    double light_source[3] = {0, 0, distance};

    double* edges_lighting = new double[n * (m + 1)];

    double** l = new double*[n * (m + 1)];
    for(int i = 0; i < n * m; i++) {
        l[i] = new double[3];

        double norm = sqrt((light_source[0] - perspective_points[i]
[0]) * (light_source[0] - perspective_points[i][0]) +
            (light_source[1] - perspective_points[i]
[1]) * (light_source[1] - perspective_points[i][1]) +
            (light_source[2] - perspective_points[i]
[2]) * (light_source[2] - perspective_points[i][2]));

        for(int j = 0; j < 3; j++) {
            l[i][j] = (light_source[j] - perspective_points[i][j]) /
norm;
        }
    }

    for(int i = 0; i < n; i++) {
        l[i + (n * m)] = new double[3];
    }
}
```

```

        double norm = sqrt((light_source[0] - perspective_points[i]
[0]) * (light_source[0] - perspective_points[i][0]) +
                           (light_source[1] - perspective_points[i]
[1]) * (light_source[1] - perspective_points[i][1]) +
                           (light_source[2] - perspective_points[i]
[2]) * (light_source[2] - perspective_points[i][2]));

        for(int j = 0; j < 3; j++) {
            l[i + (n * m)][j] = (light_source[j] -
perspective_points[i][j]) / norm;
        }
    }

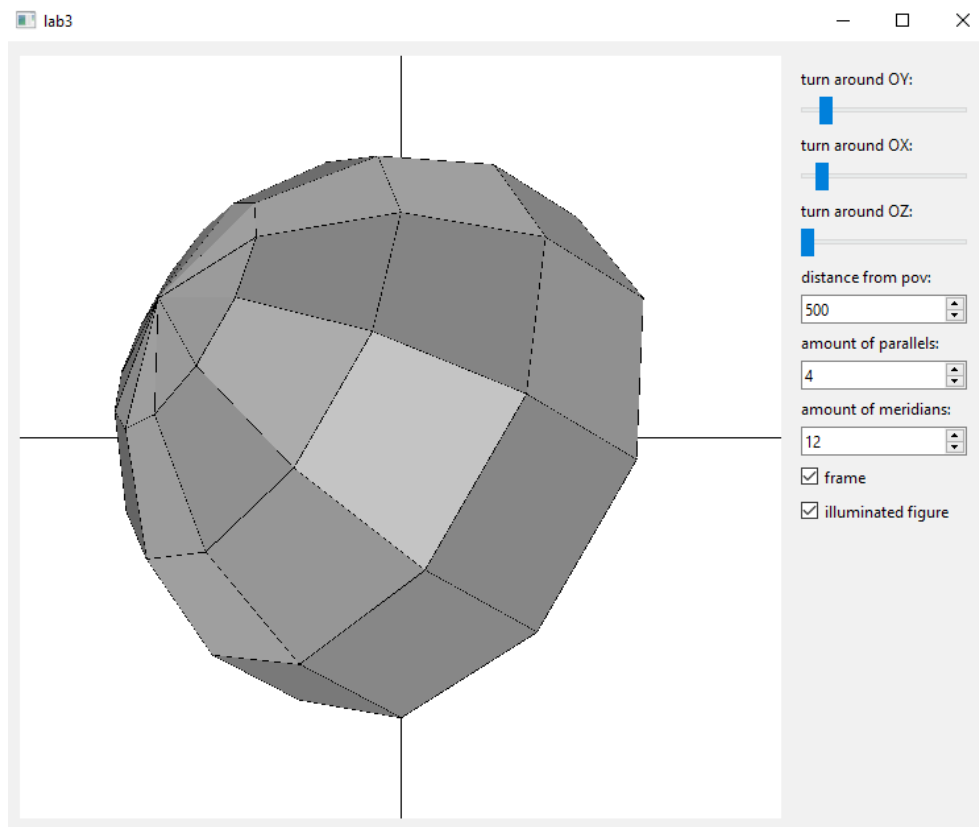
    for(int i = 0; i < n * (m + 1); i++) {
        if(i >= n * m) {
            double cos_NL = dot_product(normals[N], l[i]);
            edges_lighting[i] = k_a * i_a + k_d * cos_NL * i_d + k_s
* pow(2 * cos_NL * cos_NL - 1, alpha) * i_s;
        }
        if(i < n * m) {
            double cos_NL = dot_product(normals[i], l[i]);
            edges_lighting[i] = k_a * i_a + k_d * cos_NL * i_d + k_s
* pow(2 * cos_NL * cos_NL - 1, alpha) * i_s;
        }
    }

    return edges_lighting;
}

```

Функция реализующая модель освещения Фонга

3. Вывод программы



4. Вывод

В ходе данной лабораторной работы я освоил основы работы с инструментарием, предоставляемым QT для реализации модальных диалоговых окон и научился пользоваться процедурой аппроксимации фигур.