

**Московский Авиационный Институт
(Национальный Исследовательский Университет)**

Факультет: “Информационные технологии и прикладная математика”
Кафедра: 806 “Вычислительная математика и программирование”

Лабораторная работа №6.

Создание шейдерных анимационных эффектов в OpenGL 2.1.

Студент:	Ефимов А. В.
Группа:	М8О-307Б-18
Преподаватель:	Филиппов Г. С.
Оценка:	_____
Дата:	_____
Подпись:	_____

Москва, 2020

1. Постановка задачи

Для поверхности, созданной в лабораторной работе №5, обеспечить выполнение следующего шейдерного эффекта: Анимация. Вращение относительно направления на источник света.

2. Решение задачи

Для начала, сделаем предположение, что источник света остается на одной и той же оси, например, Y (в координатах OpenGL). Тогда задание сводится к смещению всей фигуры на заданный программистом (а позже, возможно, и пользователем) радиус.

Этот радиус в течении времени будет изменяться по формулам:

$$\begin{cases} x = r * \cos(t * \frac{180}{\pi}) \\ y = r * \sin(t * \frac{180}{\pi}) \end{cases}$$

Если считать, что \sin и \cos считают радианы, переводим их в углы, иначе $\frac{180}{\pi}$ можно опустить. t может быть получен любым методом подсчета времени.

Теперь, ослабим предположение, что источник света находится на оси. Тогда плоскость, в которой находится радиус, необходимо повернуть так, чтобы вектор от центра фигуры до этой плоскости был перпендикулярен. Сделать это можно с помощью матрицы

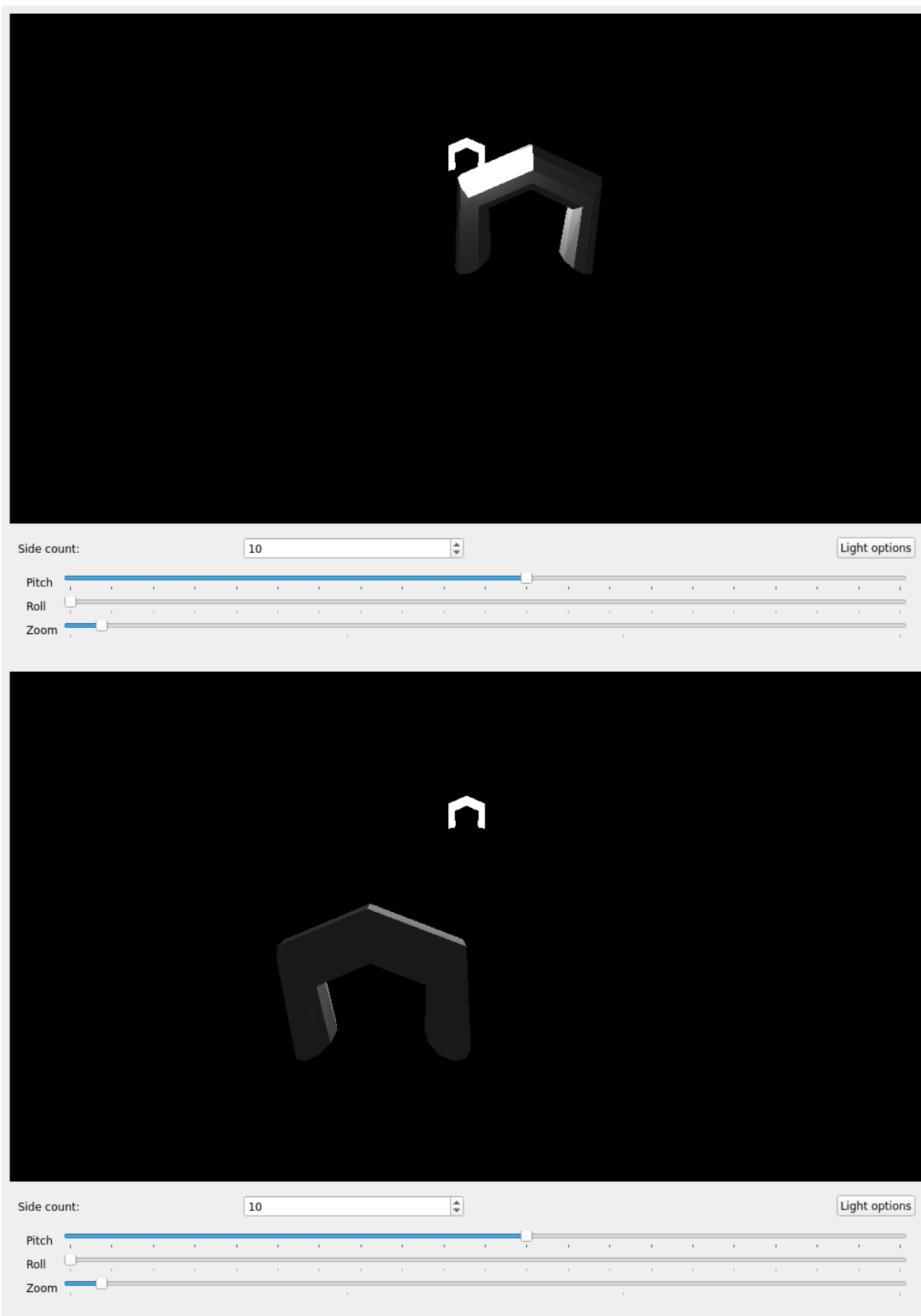
$$\begin{aligned} m(1,1) &= n_x * n_x * (1 - \cos(\alpha)) + \cos(\alpha) \\ m(1,2) &= n_x * n_y * (1 - \cos(\alpha)) - \sin(\alpha) * n_z \\ m(1,3) &= n_x * n_z * (1 - \cos(\alpha)) + \sin(\alpha) * n_y \\ m(2,1) &= n_y * n_x * (1 - \cos(\alpha)) + \sin(\alpha) * n_z \\ m(2,2) &= n_y * n_y * (1 - \cos(\alpha)) + \cos(\alpha) \\ m(2,3) &= n_y * n_z * (1 - \cos(\alpha)) - \sin(\alpha) * n_x \\ m(3,1) &= n_z * n_x * (1 - \cos(\alpha)) - \sin(\alpha) * n_y \\ m(3,2) &= n_z * n_y * (1 - \cos(\alpha)) + \sin(\alpha) * n_x \\ m(3,3) &= n_z * n_z * (1 - \cos(\alpha)) + \cos(\alpha) \end{aligned}$$

Где n – нормаль к плоскости, α – угол, на который нужно повернуть (фактически, угол между n и направлением на источник света). Полученную 3*3 матрицу можно умножит на радиус для получения поворота к нормали в направлении источника света. Полученный радиус будет корректно смещать в 3D пространстве.

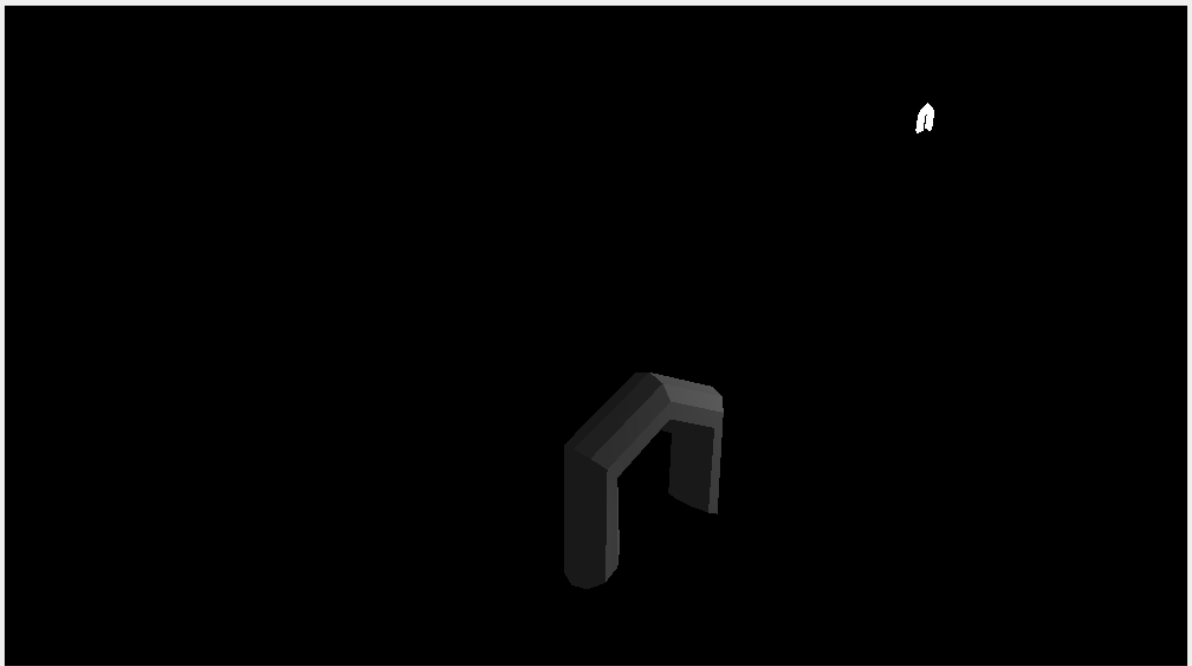
В программе будет использована оптимизированная версия, про которую можно прочесть по ссылке в конце.

3. Программа

Программа визуально ничем не отличается от той, что была ранее. Вращение начинается при запуске:



При изменении положения источника света ось вращения также меняется:



Side count: Light options

Pitch

Roll

Zoom

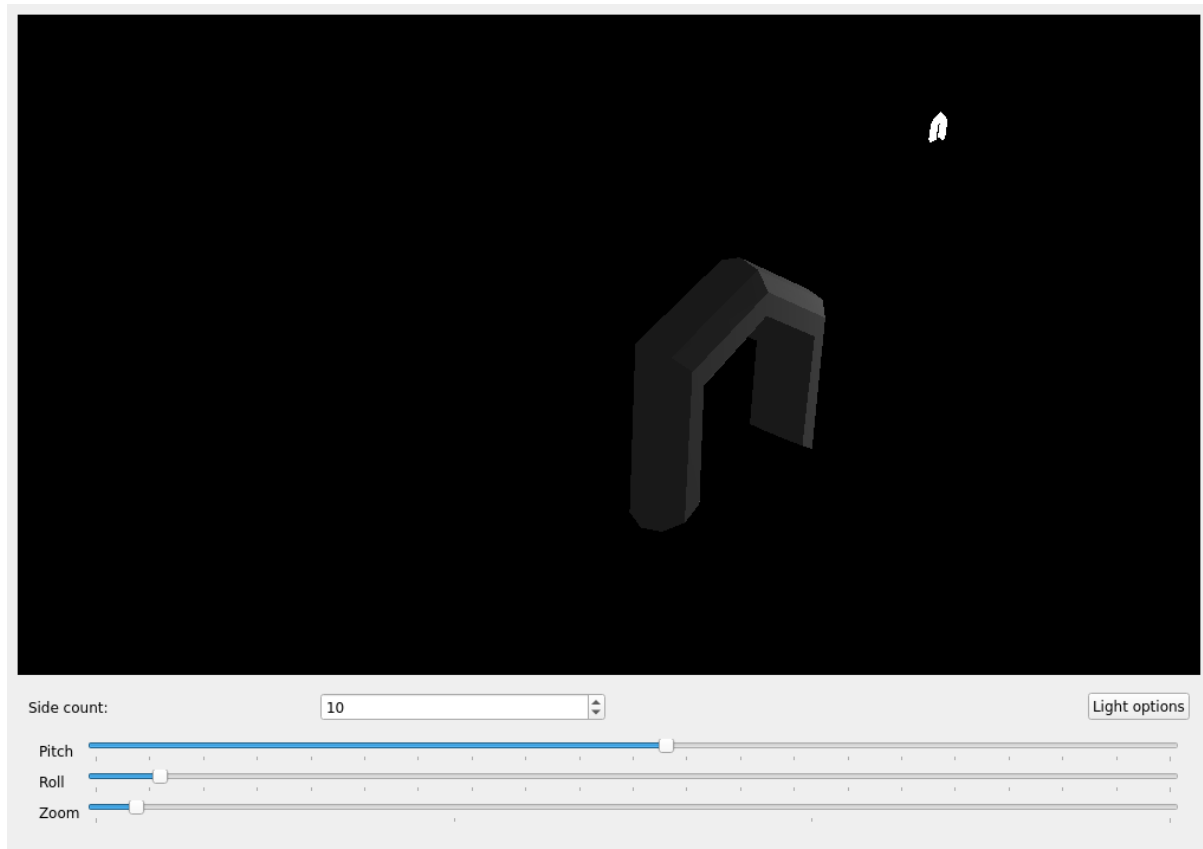


Side count: Light options

Pitch

Roll

Zoom



4. Выводы

Шейдеры являются великолепным способом придания объектам каких-либо анимации или спец. эффектов, таких как за цикленными движений пространстве (например, симуляции движение травы под ветром или волн моря) или меняющегося цвета самого объекта (динамическое освещение).

5. Источники

- <https://gist.github.com/kevinmoran/b45980723e53edeb8a5a43c49f134724>