Министерство науки и образования Украины

Национальный технический университет «ХПИ»

Кафедра «Компьютерной математики и математического моделирования»

**ОТЧЕТ**

по «Вычислительной геометрии и компьютерной графике»

Лабораторная работа №5

# «OpenGL. Шейдеры»

Проверил:

Колбасин В.А.

Выполнила:

студентка ИФ 59а

Рощенко М.В.

Харьков, 2011

***Цель данной работы:***

Получение навыков придания реалистичности изображению и навыков работы с шейдерным языком GLSL.

***Постановка задачи:***

Необходимо нарисовать сцену (можно использовать из лабораторной № 4). При отрисовке получаемое изображение должно искажаться «эффектом старого кино». Искажение можно делать и как эффект постобработки, так и во время рендеринга сцены.

***Требования к программе:***

Обязательная часть:

* Сцена должна содержать несколько объектов (можно использовать уже построенный город)
* В сцене должно присутствовать освещение.
* Возможность ходить по сцене, осматривать её под разными углами
* Реализация «эффекта старого кино» с помощью шейдеров на языке GLSL:
  + Серость получаемого изображения (или сепия)
  + Точечный шум на изображении
  + Вертикальные полосы
  + Другие эффекты

***2. Иллюстрации, показывающие работу программы:***

|  |
| --- |
| Рисунок |
|  |
|  |

1. ***Шейдеры***

***PostShader.frag***

uniform sampler2D tex1;

uniform float randValue;

float rand(vec2 co){

return fract(sin(dot(co.xy ,vec2(12.9898,78.233+randValue))) \* 43758.5453);

}

void main()

{

float NoiseAmount = 0.03;

vec3 Color;

Color= vec3(texture2D(tex1,gl\_TexCoord[0].xy));

float gray = dot(Color, float4(0.3, 0.59, 0.11, 0));

Color = float4(gray \* float3(0.9, 0.8, 0.6) , 1);

float r = rand(gl\_TexCoord[0]);

if(r>0.97)

{

Color.r = 0.4;

Color.g = 0.4;

Color.b = 0.4;

}

int t = int(5000\*gl\_TexCoord[0].x);

int k1 = randValue\*randValue\*rand(vec2(gl\_TexCoord[0].x,randValue))+81;

if((t/k1)\*k1==t&&r>0.6)

{

Color.r = 0.6;

Color.g = 0.6;

Color.b = 0.6;

}

gl\_FragColor = vec4(Color, 1.0);

}

***PostShader.vert***

void main()

{

gl\_Position = ftransform();

gl\_TexCoord[0] = gl\_MultiTexCoord0;

}

1. ***Моделирование эффекта «старого кино»***

Для реализации postprocessing эффектов мы выполнили следующие действия:

1. Отрисовали сцену;
2. С помощью функции **glCopyTexImage2D** скопировали видеобуфер в активную текстуру. Затем сбросили настройки и очистили сцену:

postRender.tex.Bind();

glCopyTexImage2D(GL\_TEXTURE\_2D, 0, GL\_RGB, 0, 0, tw,th, 0);

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

glClearColor(0.9f,0.0f,0.0f,1.0f);

glDisable(GL\_LIGHTING);

glDisable(GL\_FOG);

postRender.tex.Bind();

glLoadIdentity();

1. Отрисовали на весь экран прямоугольник с наложенной на него текстурой. В этот момент выполняем обработку текстуры различными фильтрами в активном шейдере. Для отрисовки также передаем параметры в шейдер с помощью кода:

postRender.shader.Use();

glActiveTexture(GL\_TEXTURE1);

postRender.tex.Bind();

postRender.randValue = 2 + 4\*rand()/RAND\_MAX;

postRender.shader.SetUniformInt("tex1",1);

postRender.shader.SetUniformFloat("randValue",postRender.randValue);

glTranslatef(-tw/2/kw,-th/2/kh,-8.1);

glBegin(GL\_QUADS);

glTexCoord2f(1, 0); glVertex3f(tw/kw, 0,0);

glTexCoord2f(1, 1); glVertex3f(tw/kw,th/kh,0);

glTexCoord2f(0, 1); glVertex3f(0, th/kh,0);

glTexCoord2f(0, 0); glVertex3f(0,0,0);

glEnd();

postRender.shader.Close();

glActiveTexture(GL\_TEXTURE0);

1. ***Дополнительные возможности программы***

В данной работе для улучшения реалистичности сцены были добавлены такие эффекты, как:

1. Небо, представленное в виде сферы с наложенной на нее текстурой;
2. Перемещающийся корабль по круговой траектории, реализованный с помощью таймера.