Методические указания к лабораторной работе №3 «Наложение текстур в OpenGL» по курсу «Компьютерная графика» для специальностей:

Цель лабораторной работы: Понять принципы работы с двумерными текстурами в OpenGL

Лабораторная работа основывается на лекциях: «Текстуры».

Контрольные вопросы по лекционному курсу:

Для чего используются текстуры?

Какие текстуры реализованы в OpenGL?

К какому диапазону приведены текстурные карты?

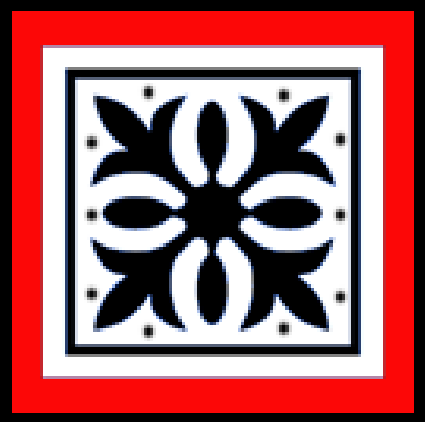
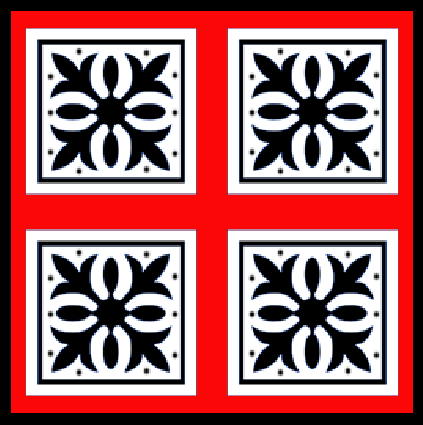
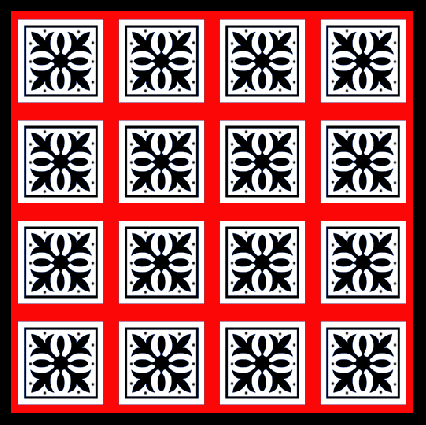
Каковы должны быть размеры текстурной карты?

Для чего нужны фильтры при текстурировании?

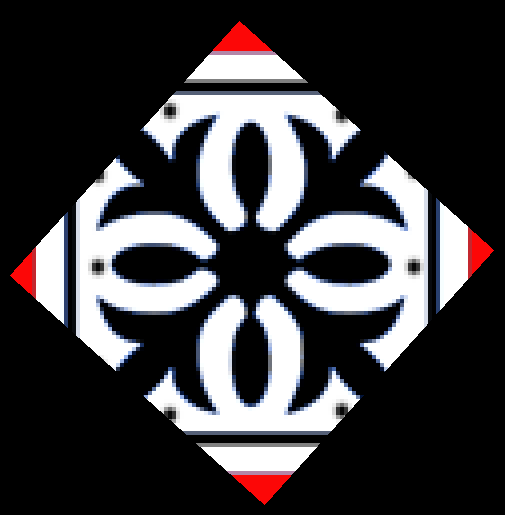
Какие виды фильтров используются в OpenGL?

**Тестовые задания:**

1. Наложить текстуру на плоский многоугольник (текстура находится в папке BMP). Привязку текстурных координат сделать вручную. Добиться следующих вариантов наложения:

a)  b)  c) 

Обеспечив вращение прямоугольника вокруг оси Z, получить еще два варианта наложения:

d)  e) 

Получить те же варианты наложения при автоматической генерации текстурных координат.

1. Создать куб из плоских многоугольников. На каждую сторону куба наложить оригинальную текстуру (текстура находится в папке CUBE), используя автоматический расчет текстурных координат. Обеспечить вращение куба вокруг осей X и Y его координатной системы.
2. Наложить текстуру на плоский многоугольник, используя MIPMAP отображение. Текстурные карты разного размера загрузить вручную из папки BMP. Обеспечить масштабирование многоугольника, изменяя коэффициенты масштабирования с клавиатуры. Установив дополнительные для MIPMAP фильтры пронаблюдать изменение используемых системой уровней текстуры.

**Подготовка данные для текстуры.**

Для загрузки изображения из файла можно использовать любую стороннюю бибилиотеку.

Например, Developer's Image Library (DevIL) . Библиотека доступна по адресу: <http://openil.sourceforge.net/download.php>. Скачивать надо файл архива DevIL 1.8.0 SDK for Windows (3.01 MB), содержащий собственно файлы библиотеки, заголовочные файлы и пр. При установке надо использовать библиотеки из папки Unicode. Установка библиотеки в Visual Studio аналогична установке OpenGL.

#include <IL\il.h>

#pragma comment (lib,"DevIL.lib")

void loadTexture..

ilInit();

ilLoadImage(L"fileName.bmp");

int width = ilGetInteger(IL\_IMAGE\_WIDTH);

int height = ilGetInteger(IL\_IMAGE\_HEIGHT);

int type = ilGetInteger(IL\_IMAGE\_TYPE);

int fmt = ilGetInteger(IL\_IMAGE\_FORMAT);

glTexImage2D(GL\_TEXTURE\_2D, 0, GL\_RGB, width, height, 0, fmt, type, ilGetData());

…

**Создать текстурный объект (если необходимо);**

GLuint texName[4]; - массив имен текстур

glGenTextures( 4, texName ); - генерация имен

glBindTexture(GL\_TEXTURE\_2D, texName[0]); - Связывание имени с текстурным объектом. Если объект с таким именем уже существует он становится текущим.

**Загрузить данные в текстурный объект;**

Смотри выше glTexImage2D

**Установить необходимые фильтры уменьшения и увеличения текстуры;**

Пример фильтра уменьшения текстуры:

glTexParameteri(GL\_TEXTURE\_2D, GL\_TEXTURE\_MIN\_FILTER, GL\_NEAREST);

**Определить поведение текстуры при выходе координат за диапазон;**

Пример фиксации параметра S текстуры:

glTexParameterf(GL\_TEXTURE\_2D, GL\_TEXTURE\_WRAP\_S, GL\_CLAMP); Поведение необходимо описывать для каждого параметра отдельно. Для двумерной текстуры для S и для T.

**Задать функцию наложения текстуры;**

Пример. Текстурная функция заменяющая цвет пикселя на информацию о цвете хранимую в текселе:

glTexEnvi(GL\_TEXTURE\_ENV, GL\_TEXTURE\_ENV\_MODE, GL\_REPLACE)

**Привязать координаты текстуры к вершинам текстурируемого объекта;**

glBegin(GL\_POLYGON);

glTexCoord2f(0, 1);

glVertex3f(-2, 2, 0);

glTexCoord2f(0, 0);

glVertex3f(-2, -2, 0);

glTexCoord2f(1, 0);

glVertex3f(2, -2, 0);

glTexCoord2f(1, 1);

glVertex3f(2, 2, 0);

glEnd();

**(Если используется автоматическая генерация текстурных координат, задать метод генерации);**

GLdouble ss[4] = { 0.25, 0, 0, 0.5 };

glEnable(GL\_TEXTURE\_GEN\_S);

glTexGeni(GL\_T, GL\_TEXTURE\_GEN\_MODE, GL\_OBJECT\_LINEAR);

glTexGendv(GL\_S, GL\_OBJECT\_PLANE, ss);

**Разрешить использование текстуры.**

glEnable(GL\_TEXTURE\_2D);

glGenerateMipmap(GL\_TEXTURE\_2D);

https://docs.gl/gl3/glBindTexture