Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа № 10

по дисциплине "Программирование графических приложений"

ТЕМА РАБОТЫ:

Текстурирование

Выполнил:

студент гр. ПРИм-124

Парахин К.В.

Принял:

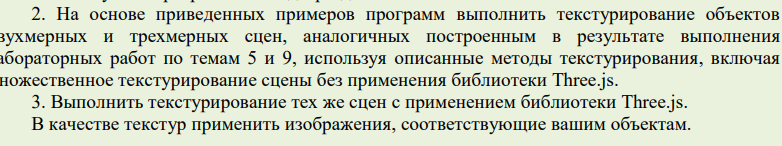
Жигалов И.Е.

Владимир 2024 г.

Цель работы:

Изучение методов текстурировнаия трехмерных графических объектов в WebGL.

Выполнение работы:



Листинг установки текстуры на двумерное изображение:

function setTextures(){

    texture = gl.createTexture();

    gl.bindTexture(gl.TEXTURE\_2D, texture);

    var image = new Image();

    image.onload = function() {

        handleTextureLoaded(image, texture);

  }

   image.src = "coord.png";

    shaderProgram.samplerUniform = gl.getUniformLocation(shaderProgram, "uSampler");

    gl.uniform1i(shaderProgram.samplerUniform, 0);

}



Рисунок 1. Текстурирование двухмерного изображения из темы 5

Создадим трехмерный аквариум и набросим на нее текстуру с соответствующим изображением:

function initAquariumBuffers() {

    var vertices = [

        // Передняя стена

        -0.5, -0.5, 0.5,

        -0.5, 0.5, 0.5,

         0.5, 0.5, 0.5,

         0.5, -0.5, 0.5,

        // Задняя стена

        -0.5, -0.5, -0.5,

        -0.5, 0.5, -0.5,

         0.5, 0.5, -0.5,

         0.5, -0.5, -0.5,

        // Левая стена

        -0.5, -0.5, -0.5,

        -0.5, 0.5, -0.5,

        -0.5, 0.5, 0.5,

        -0.5, -0.5, 0.5,

        // Правая стена

        0.5, -0.5, -0.5,

        0.5, 0.5, -0.5,

        0.5, 0.5, 0.5,

        0.5, -0.5, 0.5,

        // Дно

        -0.5, -0.5, -0.5,

        -0.5, -0.5, 0.5,

        0.5, -0.5, 0.5,

        0.5, -0.5, -0.5

    ];

    var indices = [

        // Передняя стена

        0, 1, 2,  2, 3, 0,

        // Задняя стена

        4, 5, 6,  6, 7, 4,

        // Левая стена

        8, 9, 10,  10, 11, 8,

        // Правая стена

        12, 13, 14,  14, 15, 12,

        // Дно

        16, 17, 18,  18, 19, 16

    ];

    aquariumVertexBuffer = gl.createBuffer();

    gl.bindBuffer(gl.ARRAY\_BUFFER, aquariumVertexBuffer);

    gl.bufferData(gl.ARRAY\_BUFFER, new Float32Array(vertices), gl.STATIC\_DRAW);

    aquariumVertexBuffer.itemSize = 3;

    aquariumIndexBuffer = gl.createBuffer();

    gl.bindBuffer(gl.ELEMENT\_ARRAY\_BUFFER, aquariumIndexBuffer);

    gl.bufferData(gl.ELEMENT\_ARRAY\_BUFFER, new Uint16Array(indices), gl.STATIC\_DRAW);

    aquariumIndexBuffer.numberOfItems = indices.length;

    var textureCoords = [];

    for (var i = 0; i < 5; i++) {

        textureCoords.push(0.0, 0.0, 0.0, 1.0, 1.0, 1.0, 1.0, 0.0);

    }

    aquariumTextureCoordsBuffer = gl.createBuffer();

    gl.bindBuffer(gl.ARRAY\_BUFFER, aquariumTextureCoordsBuffer);

    gl.bufferData(gl.ARRAY\_BUFFER, new Float32Array(textureCoords), gl.STATIC\_DRAW);

    aquariumTextureCoordsBuffer.itemSize = 2;

}

function aquariumDraw() {

    gl.bindBuffer(gl.ARRAY\_BUFFER, aquariumVertexBuffer);

    gl.vertexAttribPointer(shaderProgram.vertexPositionAttribute, aquariumVertexBuffer.itemSize, gl.FLOAT, false, 0, 0);

    gl.bindBuffer(gl.ARRAY\_BUFFER, aquariumTextureCoordsBuffer);

    gl.vertexAttribPointer(shaderProgram.vertexTextureAttributeLoc, aquariumTextureCoordsBuffer.itemSize, gl.FLOAT, false, 0, 0);

    gl.activeTexture(gl.TEXTURE0);

    gl.bindTexture(gl.TEXTURE\_2D, aquariumTexture);

    gl.enable(gl.DEPTH\_TEST);

    gl.drawElements(gl.TRIANGLES, aquariumIndexBuffer.numberOfItems, gl.UNSIGNED\_SHORT, 0);

}

function setupTextures() {

    aquariumTexture = gl.createTexture();

    setTexture("aquarium.png", aquariumTexture);

}

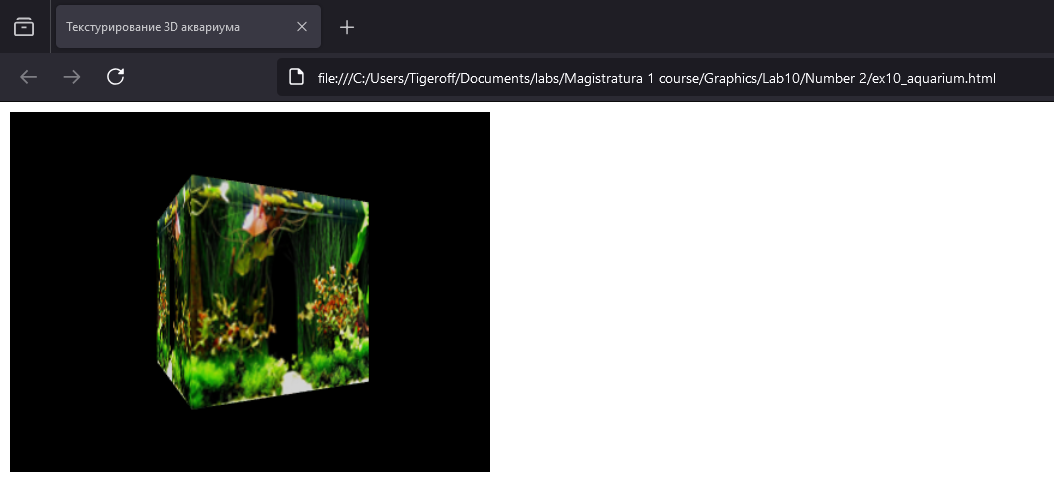


Рисунок 2. Трехмерное текстурирование

Сделаем тоже самое, но с множественным текстурированием – добавив свреху аквариума крышку:

function setupTextures() {

    aquariumWallsTexture = gl.createTexture();

    aquariumWallsTexture.image = new Image();

    aquariumWallsTexture.image.onload = function () {

        handleTextureLoaded(aquariumWallsTexture)

    }

    aquariumWallsTexture.image.src = "aquarium.png";

    aquariumTopTexture = gl.createTexture();

    aquariumTopTexture.image = new Image();

    aquariumTopTexture.image.onload = function () {

        handleTextureLoaded(aquariumTopTexture)

    }

    aquariumTopTexture.image.src = "stone.jpg";

}

Вывод

В результате выполнения работы я провел изучение методов текстурирования трехмерных графических объектов в WebGL.