

ИШИТР ОИТ

09.03.01 «Информатика и вычислительная техника»

ОТЧЁТ

по Лабораторной работе № 11

**Построение трехмерных сцен на XAML**

дисциплина:

**Компьютерная графика**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Исполнитель:** |  | | | | |
| студент группы 8В14 | | | В.О.Маслюк |  | 06.11.2023 |
|  |  |  |  |  | Дата сдачи |
| **Руководитель:** |  | | | | |
| к.т.н. доцент ОИТ ИШИТР А.Ю.Демин | | | | | |
|  |  |  |  |  |  |

Томск – 2023

**Цель работы**: изучить принципы построения статических 3D сцен с помощью XAML (eXtensible Application Markup Language) в WPF (windows presentation foundation). Написать и протестировать программу для получения трехмерной сцены.

**Задание:**

Создать с помощью описания XAML трехмерную сцену, состоящую из нескольких разноцветных объектов. Добавьте к этой сцене несколько источников света. Поместите в отчет изображение этой сцены, полученное при различных трехмерных преобразованиях. Получите разные типы проекций, результат также поместите в отчет. Самостоятельно изучите возможности работы с текстурами. Добавьте в трехмерную сцену объекты с текстурой.

**Ход работы:**

1. Опишем с помощью XAML трехмерную сцену с кубом, октаэдром и полигоном:

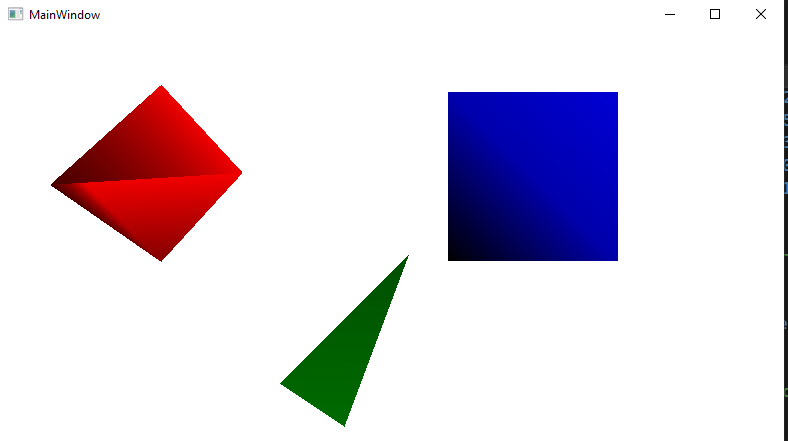


Рисунок 1 Начальная трехмерная сцена

1. Произведем преобразования над фигурами:

* Сдвинем синий куб по оси Х
* Увеличим зеленый полигон по оси Х
* Повернем октаэдр на 45 градусов по оси Y

Результат преобразований:

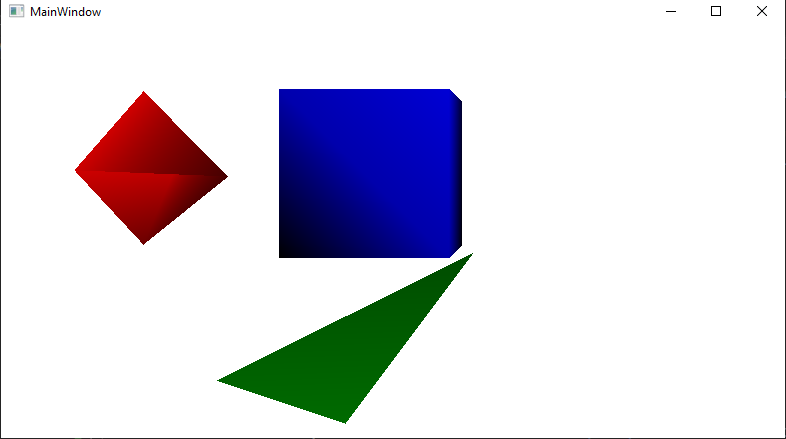


Рисунок 2 Сцена с преобразованными элементами

1. Получим ортогональные проекции октаэдра и полигона, сменив PersperctiveCamera на OrtographicCamera:

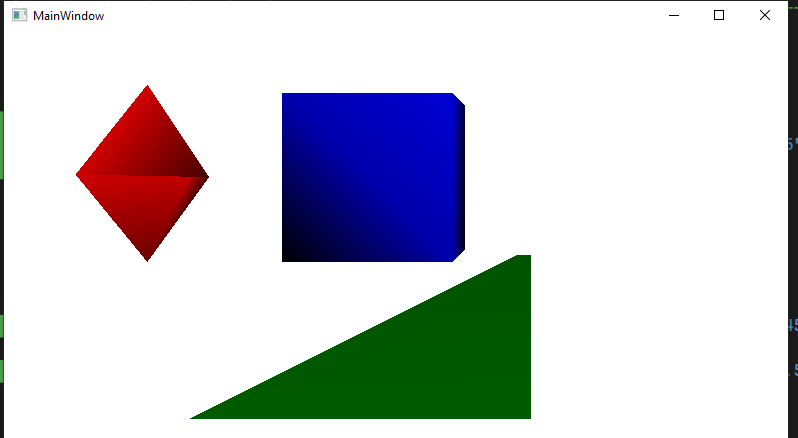


Рисунок 3 Сцена с полученными проекциями

1. Наложим текстуру на наш куб и повернем его:

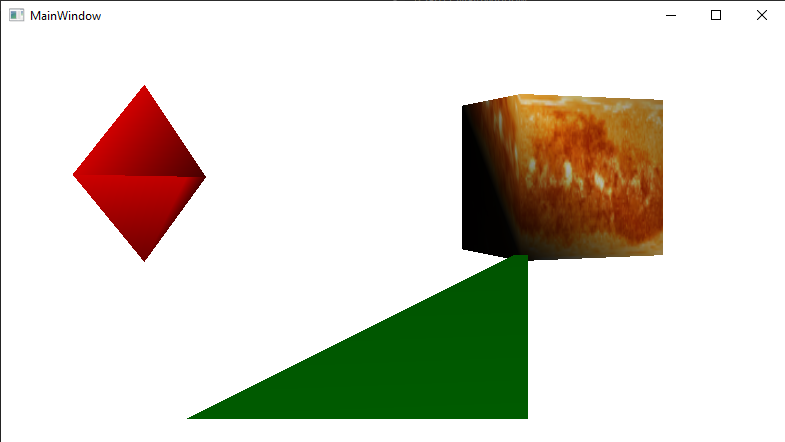


Рисунок 4 Куб с текстурой

**Текст программы:**

<Window x:Class="lab11.MainWindow"

xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"

xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"

xmlns:d="http://schemas.microsoft.com/expression/blend/2008"

xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"

xmlns:local="clr-namespace:lab11"

mc:Ignorable="d"

Title="MainWindow" Height="450" Width="800">

<Grid Margin="0,0,10,-1">

<Grid.ColumnDefinitions>

<ColumnDefinition Width="45\*"/>

<ColumnDefinition Width="113\*"/>

</Grid.ColumnDefinitions>

<Viewport3D Margin="146,-60,-146,59" Grid.ColumnSpan="2">

<Viewport3D.Camera>

<PerspectiveCamera Position="0.5,0.5,6.5" LookDirection="0,0,-3.5" />

</Viewport3D.Camera>

<Viewport3D.Children>

<ModelVisual3D>

<ModelVisual3D.Content>

<DirectionalLight Color="#FFFFFF" Direction="-1,-1,-2" />

</ModelVisual3D.Content>

</ModelVisual3D>

<ModelVisual3D>

<ModelVisual3D.Content>

<GeometryModel3D>

<GeometryModel3D.Geometry>

<MeshGeometry3D

TriangleIndices="0,2,1 1,2,3 0,4,2 2,4,6

0,1,4 1,5,4 1,7,5 1,3,7

4,5,6 7,6,5 2,6,3 3,6,7 "

Positions="0,0,0 1,0,0 0,1,0 1,1,0

0,0,1 1,0,1 0,1,1 1,1,1 "

TextureCoordinates=

"0,0 0,1 0,1 1,1

1,1 0,1 1,0 0,0

0,0 1,0 0,1 1,1

1,1 0,1 1,0 0,0

1,1 0,1 1,0 0,1

"/>

</GeometryModel3D.Geometry>

<GeometryModel3D.Material>

<DiffuseMaterial>

<DiffuseMaterial.Brush>

<ImageBrush ImageSource ="2k\_sun.jpg"/>

</DiffuseMaterial.Brush>

</DiffuseMaterial>

</GeometryModel3D.Material>

</GeometryModel3D>

</ModelVisual3D.Content>

<ModelVisual3D.Transform>

<RotateTransform3D>

<RotateTransform3D.Rotation>

<AxisAngleRotation3D Axis="0 1 0" Angle="25" />

</RotateTransform3D.Rotation>

</RotateTransform3D>

</ModelVisual3D.Transform>

</ModelVisual3D>

</Viewport3D.Children>

</Viewport3D>

<Viewport3D Margin="-41,-38,460,103" RenderTransformOrigin="0.538,0.458" Grid.ColumnSpan="2">

<Viewport3D.Camera>

<OrthographicCamera Position="0.2,0.1,2.5" LookDirection="0.3,-0.1,-4.5" />

</Viewport3D.Camera>

<Viewport3D.Children>

<ModelVisual3D>

<ModelVisual3D.Content>

<DirectionalLight Color="White" Direction="-1,-2,-7" />

</ModelVisual3D.Content>

</ModelVisual3D>

<ModelVisual3D>

<ModelVisual3D.Content>

<GeometryModel3D>

<GeometryModel3D.Geometry>

<MeshGeometry3D

TriangleIndices="0,1,2 1,3,0 3,2,0 2,3,1 1,0,4 0,3,4 4,3,1"

Positions="0,0,0 1,0,0 0.5,0.5,0.1 0,0,1 0.5,-0.5,0.1">

</MeshGeometry3D>

</GeometryModel3D.Geometry>

<GeometryModel3D.Material>

<MaterialGroup>

<DiffuseMaterial Brush="Red" />

</MaterialGroup>

</GeometryModel3D.Material>

</GeometryModel3D>

</ModelVisual3D.Content>

<ModelVisual3D.Transform>

<RotateTransform3D>

<RotateTransform3D.Rotation>

<AxisAngleRotation3D Axis="0 1 0" Angle="45" />

</RotateTransform3D.Rotation>

</RotateTransform3D>

</ModelVisual3D.Transform>

</ModelVisual3D>

</Viewport3D.Children>

</Viewport3D>

<Viewport3D Margin="172,224,247,24" RenderTransformOrigin="0.538,0.458" Grid.ColumnSpan="2">

<Viewport3D.Camera>

<OrthographicCamera Position="0,0,10" LookDirection="0,0,-4.5" />

</Viewport3D.Camera>

<Viewport3D.Children>

<ModelVisual3D>

<ModelVisual3D.Content>

<PointLight Color="White" Position="-1,-7,7" />

</ModelVisual3D.Content>

</ModelVisual3D>

<ModelVisual3D>

<ModelVisual3D.Content>

<GeometryModel3D>

<GeometryModel3D.Geometry>

<MeshGeometry3D

TriangleIndices="0,1,2 "

Positions="0,-2.5,0 1.5,1.5,0 -1.5,-1.5,0">

</MeshGeometry3D>

</GeometryModel3D.Geometry>

<GeometryModel3D.Material>

<MaterialGroup>

<DiffuseMaterial Brush="Green" />

</MaterialGroup>

</GeometryModel3D.Material>

</GeometryModel3D>

</ModelVisual3D.Content>

<ModelVisual3D.Transform>

<ScaleTransform3D ScaleX="2"/>

</ModelVisual3D.Transform>

</ModelVisual3D>

</Viewport3D.Children>

</Viewport3D>

</Grid>

</Window>

**Вывод:** в ходе лабораторной работы изучили принципы построения статических 3D сцен с помощью XAML (eXtensible Application Markup Language) в WPF (windows presentation foundation). Написали и протестировали программу для получения трехмерной сцены.