**计算机游戏设计期末作业**

课程名称：计算机游戏设计

任课教师：王 欣

学 号：2017542007

姓 名：杜婉莹

2017年11月21日

* 第一次作业

1. 将窗口改为400\*400

int mClientWidth = 400;

int mClientHeight = 400;

1. 将窗口标题改为“姓名-学号”

std::wstring mMainWndCaption = L"杜婉莹+2017542007";

1. 将窗口内部颜色改为红色

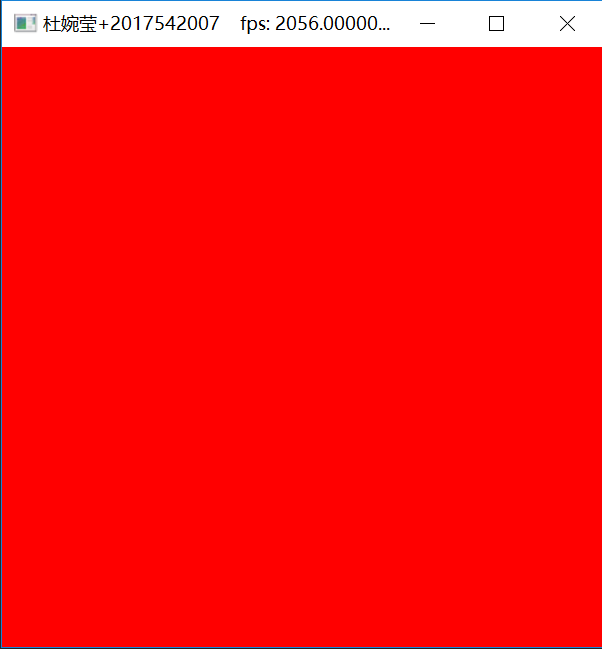
mCommandList->ClearRenderTargetView(CurrentBackBufferView(), Colors::Red, 0, nullptr); //把后台缓存清掉，重新覆盖掉

1. 解释mspf的含义：（mspf=1000.0/fps）多少毫秒每帧

float fps = (float)frameCnt; // fps = frameCnt / 1

float mspf = 1000.0f / fps;

1. 运行结果



* 第二次作业

1. 构造四棱锥（Pyramid）

std::array<Vertex, 8> vertices =

{

Vertex({ XMFLOAT3(0.0f, +1.414f, 0.0f), XMFLOAT4(Colors::White) }),

Vertex({ XMFLOAT3(+1.0f, 0.0f, -1.0f), XMFLOAT4(Colors::Red) }),

Vertex({ XMFLOAT3(+1.0f, 0.0f, +1.0f), XMFLOAT4(Colors::Green) }),

Vertex({ XMFLOAT3(-1.0f, 0.0f, +1.0f), XMFLOAT4(Colors::Blue) }),

Vertex({ XMFLOAT3(-1.0f, 0.0f, -1.0f), XMFLOAT4(Colors::Yellow) }),

Vertex({ XMFLOAT3(0.0f, -1.414f, 0.0f), XMFLOAT4(Colors::Black) })

};

std::array<std::uint16\_t, 36> indices =

{

//顺时针！！！

//up

0,2,1,

0,3,2,

0,4,3,

0,1,4,

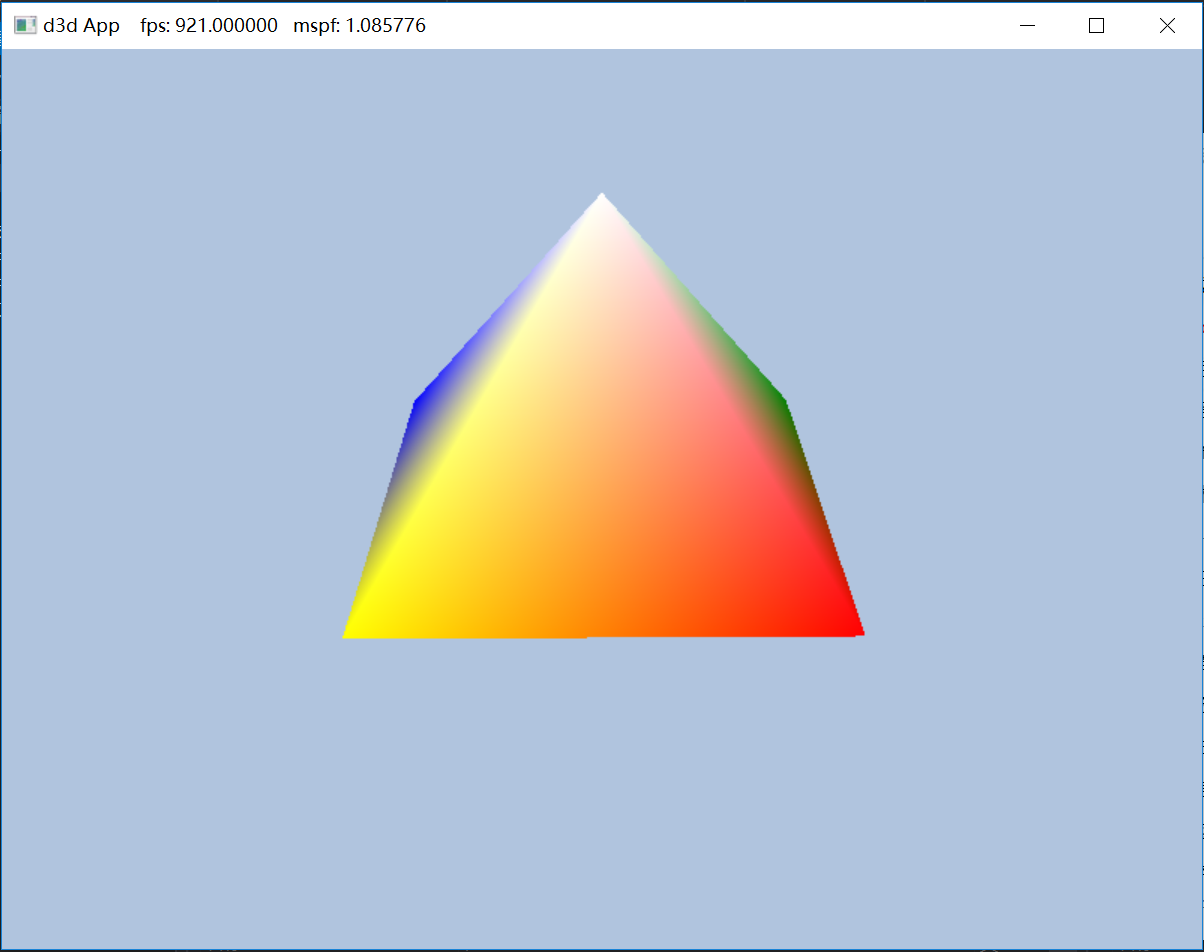
//middle

4,2,3,

2,4,1

};

1. 运行结果



* 第三次作业

1. 构造正八面体（Pyramid\_Eight）

std::array<Vertex, 8> vertices =

{

Vertex({ XMFLOAT3(0.0f, +1.414f, 0.0f), XMFLOAT4(Colors::White) }),

Vertex({ XMFLOAT3(+1.0f, 0.0f, -1.0f), XMFLOAT4(Colors::Red) }),

Vertex({ XMFLOAT3(+1.0f, 0.0f, +1.0f), XMFLOAT4(Colors::Green) }),

Vertex({ XMFLOAT3(-1.0f, 0.0f, +1.0f), XMFLOAT4(Colors::Blue) }),

Vertex({ XMFLOAT3(-1.0f, 0.0f, -1.0f), XMFLOAT4(Colors::Yellow) }),

Vertex({ XMFLOAT3(0.0f, -1.414f, 0.0f), XMFLOAT4(Colors::Black) })

};

std::array<std::uint16\_t, 36> indices =

{

//顺时针！！！

//up

0,2,1,

0,3,2,

0,4,3,

0,1,4,

//middle

/\*4,2,3,

2,4,1\*/

//down

5,1,2,

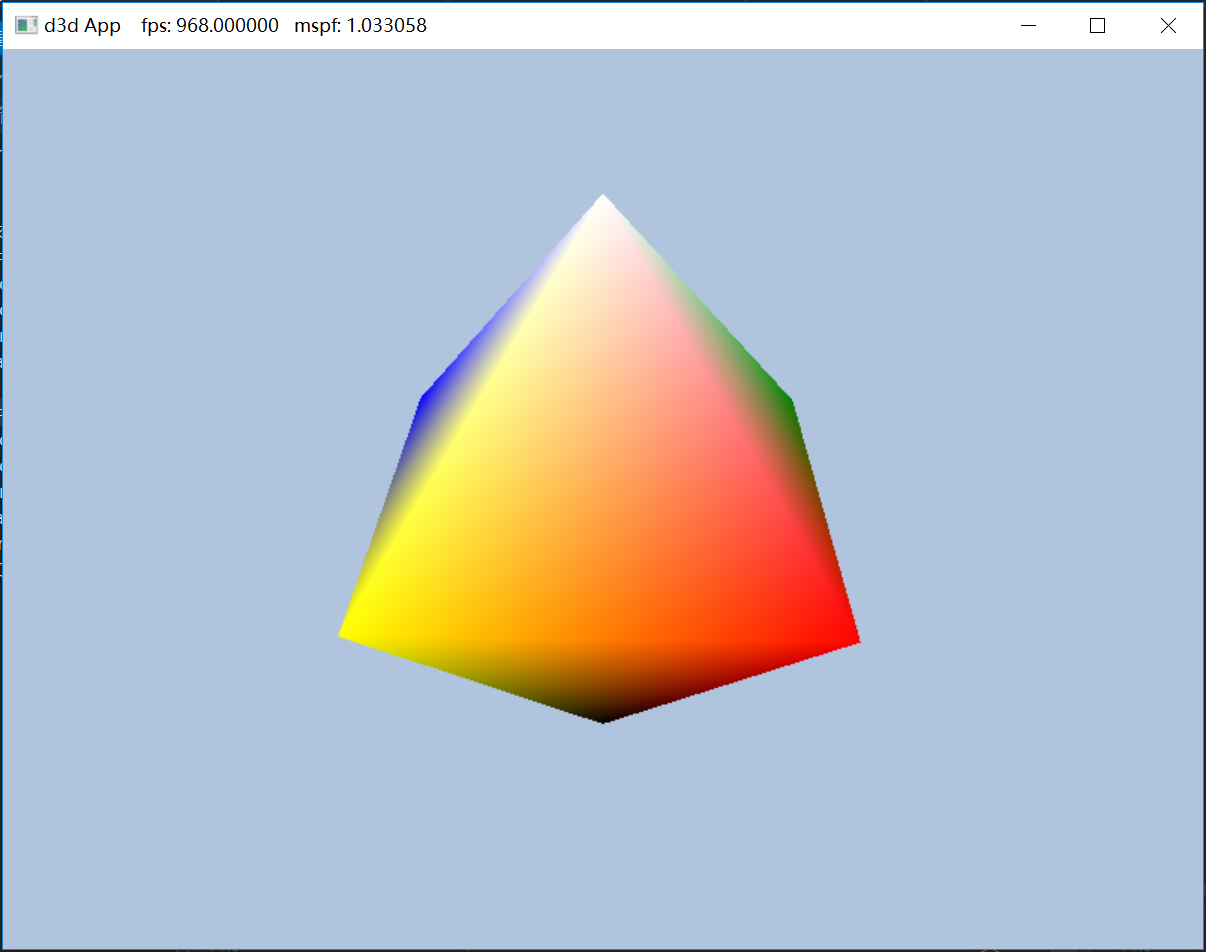
5,2,3,

5,3,4,

5,4,1

};

1. 运行结果



* 第四次作业

1. 在原始场景中打红光

mMainPassCB.Lights[0].Direction = { 0.57735f, -0.57735f, 0.57735f };//{ 0.57735f, -0.57735f, 0.57735f }

mMainPassCB.Lights[0].Strength = { 0.9f, 0.05f, 0.05f };//{ 0.6f, 0.6f, 0.6f }

mMainPassCB.Lights[1].Direction = { -0.57735f, -0.57735f, 0.57735f };//{ -0.57735f, -0.57735f, 0.57735f }

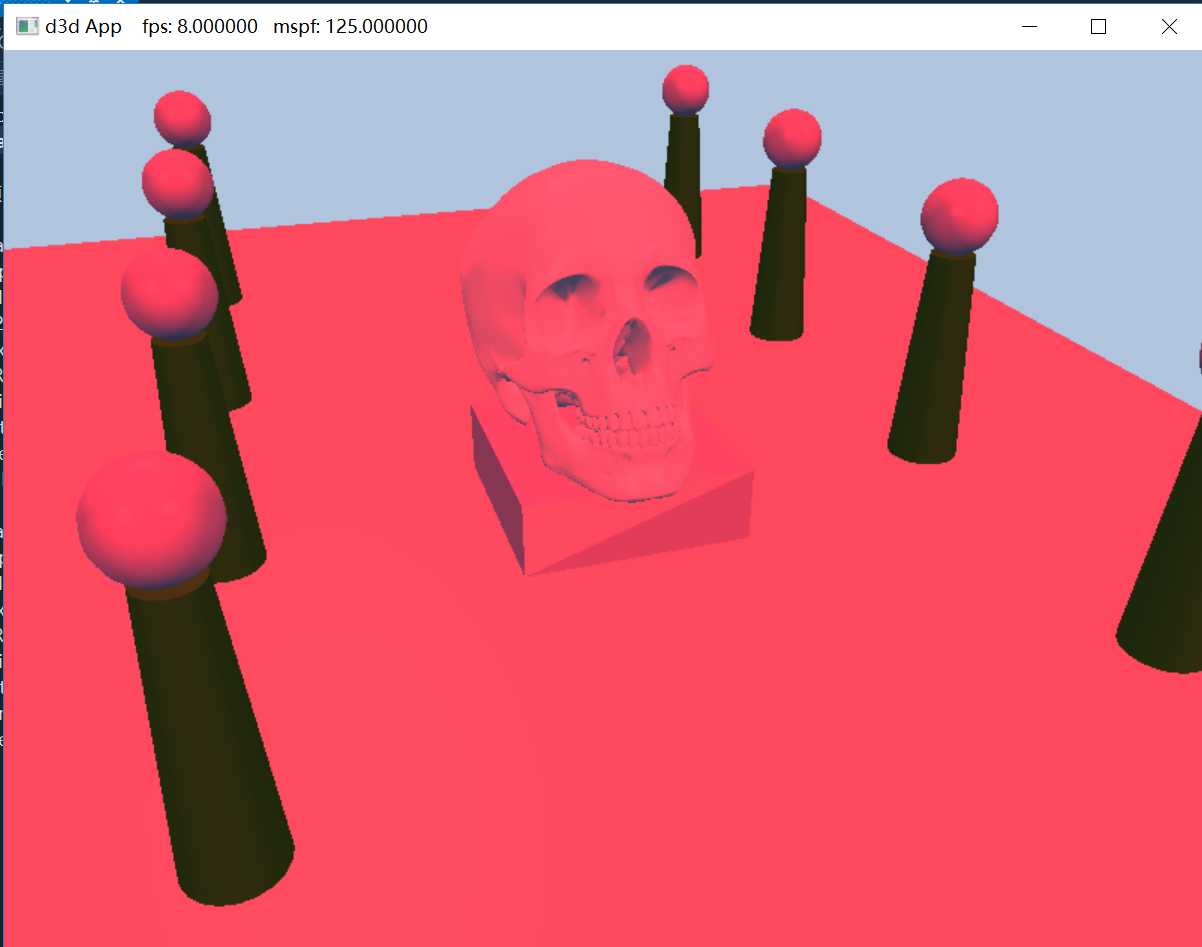
mMainPassCB.Lights[1].Strength = { 0.9f, 0.05f, 0.05f };//{ 0.3f, 0.3f, 0.3f }

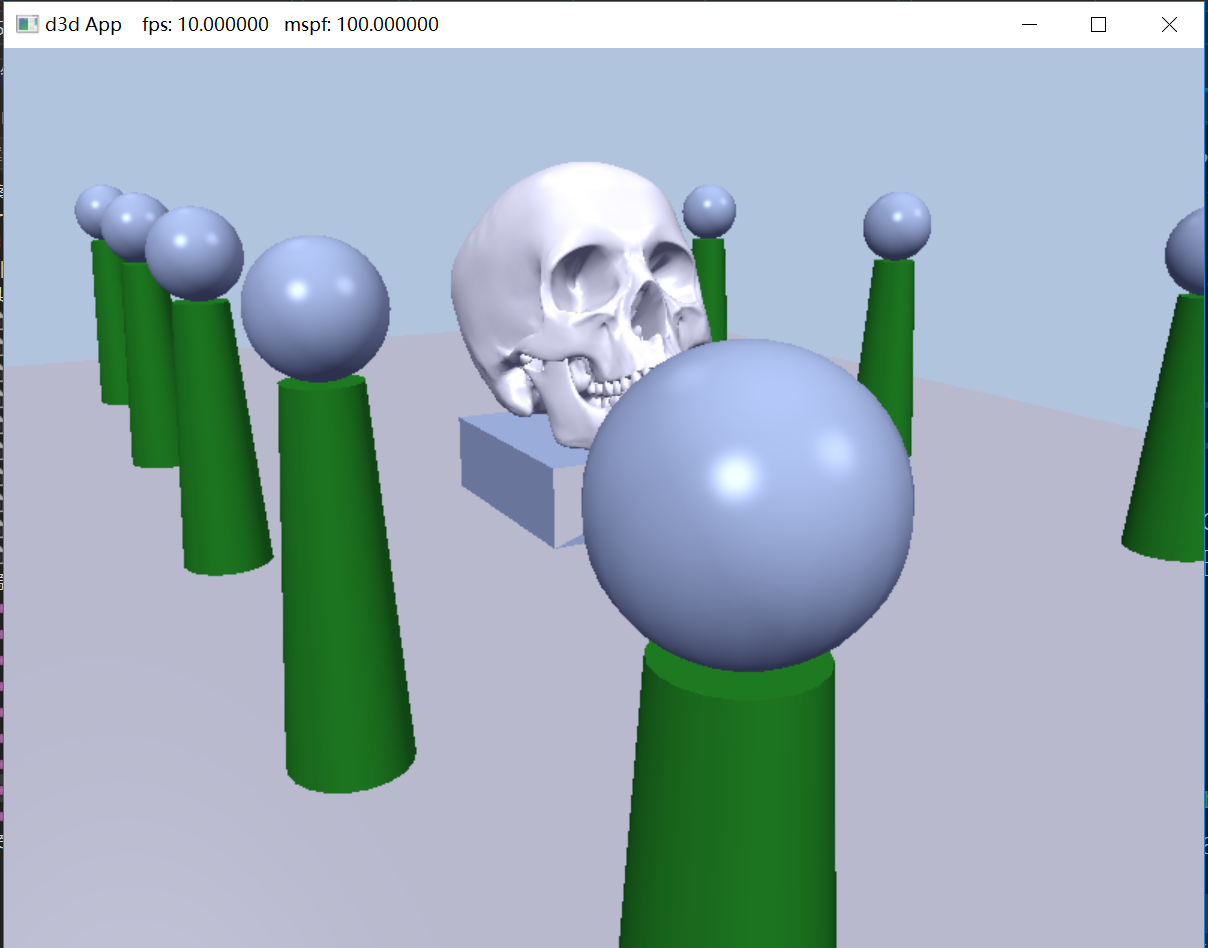
mMainPassCB.Lights[2].Direction = { 0.0f, -0.707f, -0.707f };//{ 0.0f, -0.707f, -0.707f }

mMainPassCB.Lights[2].Strength = { 0.9f, 0.05f, 0.05f };//{ 0.15f, 0.15f, 0.15f }

1. 调整粗糙度

stone0->Roughness = 0.3f;//0.9f

1. 运行结果



* 第五次作业

1. 给正方体贴5\*5重复纹理

// Fill in the back face vertex data.

v[4] = Vertex(-w2, -h2, +d2, 0.0f, 0.0f, 1.0f, -1.0f, 0.0f, 0.0f, 5.0f, 5.0f);

v[5] = Vertex(+w2, -h2, +d2, 0.0f, 0.0f, 1.0f, -1.0f, 0.0f, 0.0f, 0.0f, 5.0f);

v[6] = Vertex(+w2, +h2, +d2, 0.0f, 0.0f, 1.0f, -1.0f, 0.0f, 0.0f, 0.0f, 0.0f);

v[7] = Vertex(-w2, +h2, +d2, 0.0f, 0.0f, 1.0f, -1.0f, 0.0f, 0.0f, 5.0f, 0.0f);

1. 镜像纹理

const CD3DX12\_STATIC\_SAMPLER\_DESC pointWrap(

0, // shaderRegister

D3D12\_FILTER\_MIN\_MAG\_MIP\_POINT, // filter

D3D12\_TEXTURE\_ADDRESS\_MODE\_MIRROR, // addressU

D3D12\_TEXTURE\_ADDRESS\_MODE\_MIRROR, // addressV

D3D12\_TEXTURE\_ADDRESS\_MODE\_MIRROR); // addressW

1. 运行结果

