

北京林业大学 2023--2024 学年第 2 学期实验任务书

课程名称： 三维动画原理与制作 开课学院： 信息学院

考试班级： 数媒 21-1、2 命题人： 杨 猛

实验环境： Unity 3D 2019 或以上 实验学时： 2

实验题目(范围)： 实验 1 图片插值动画

请详细说明该设计的方案、内容、要求、进度等

严禁剽窃、抄袭等作弊行为！

实验目的：

1. 掌握 Unity 的 Build-In 实验环境。
2. 理解插值动画的基础思想。
3. 熟悉 Unity 中常见的插值计算。

实验内容：

1. Unity 3D 的动画编程方法，主要包括：
 - (1) 熟悉 Unity 3D，创建新项目的方法。
 - (2) 熟悉 Unity 3D 脚本编辑方法。
 - (3) 重点熟练掌握 Unity 3D 的动画编程方法。
2. 依照“**实验方法**”中步骤依次进行实验操作，并给所添加/修改的每一句代码添加注释。

实验环境要求：

1. 实验环境：建议使用 Unity 3D 2019 或更高版本，可在 <https://unity.cn/releases> 路径下载。

实验方法：**(注：以下演示步骤均在 Unity2019.4.17f1c1 环境中完成)**

1. 建立 Unity 项目，初始化实验环境

打开 UnityHub，选择，新建项目，新项目->3D 核心模板->创建项目 （图 1）。

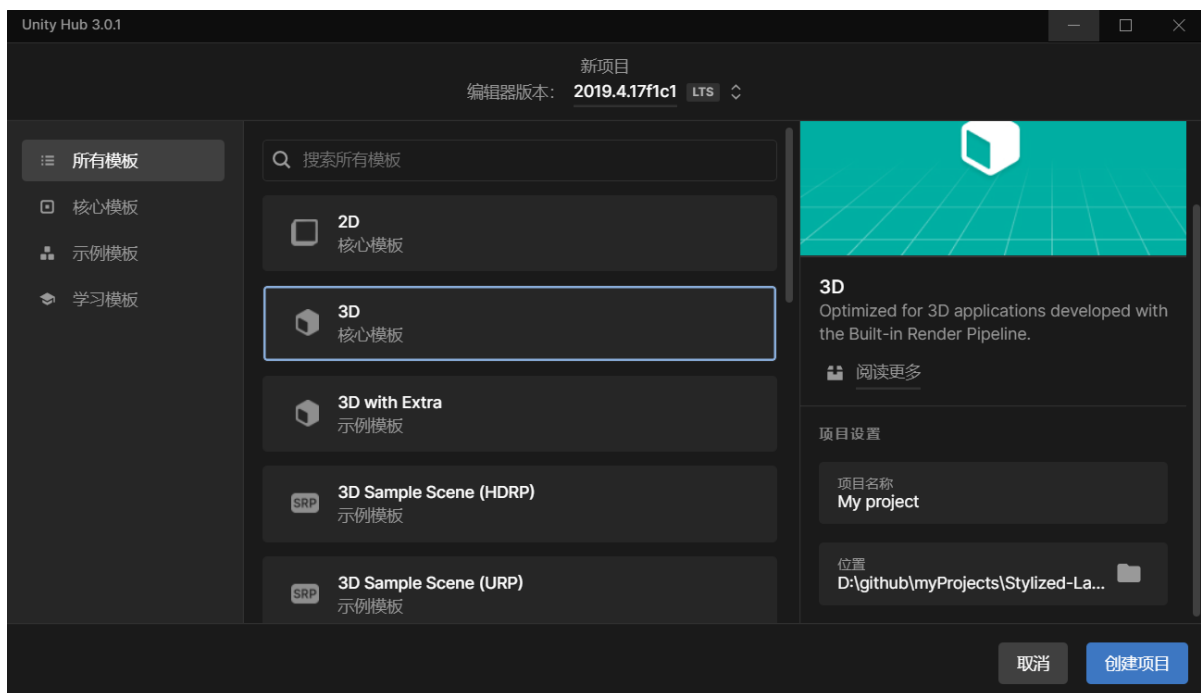


图 1 新建项目菜单

将 Assets 中的 Scenes 文件夹删除（如图 2 所示）。

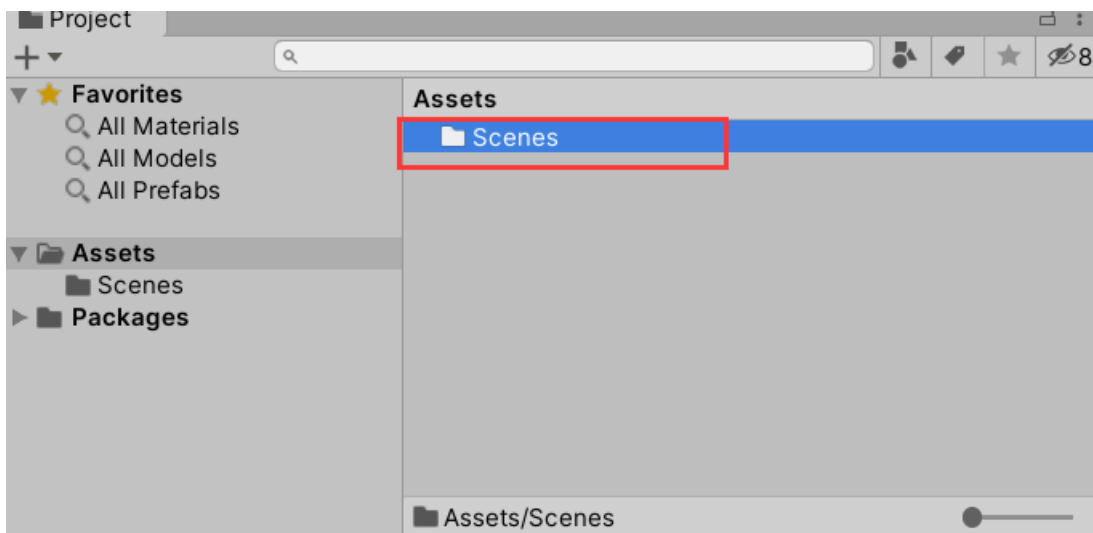


图 2 删除 Scene 文件夹

右键 Create->Scene 新建一个场景，命名为实验 1，然后双击打开（如图 3）。

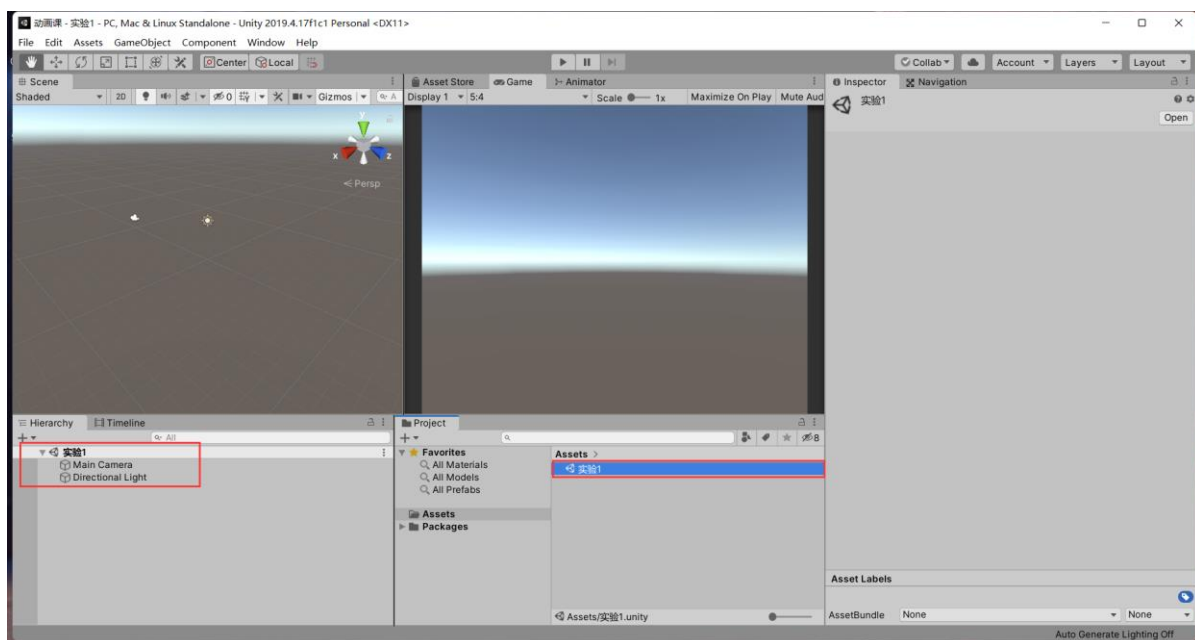


图 3 打开新的场景

将 Game 视图的比例改为 16: 9，将摄像头的 Clear Flags 改为 Solid Color，透视模式 (Projection) 改为正交透视 (Orthographic)

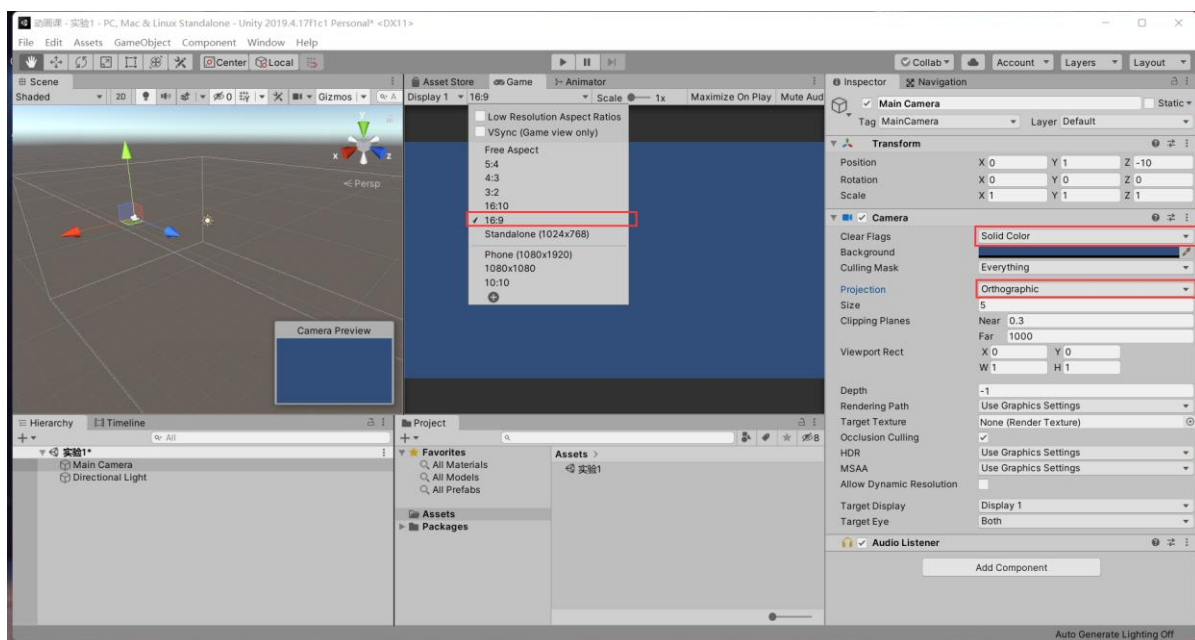


图 4 设置整体环境

2. 图片插值材质初始化

如图 5 所示，鼠标右键 Create->Shader->Unlit Shader 新建一个不发光着色器文件。

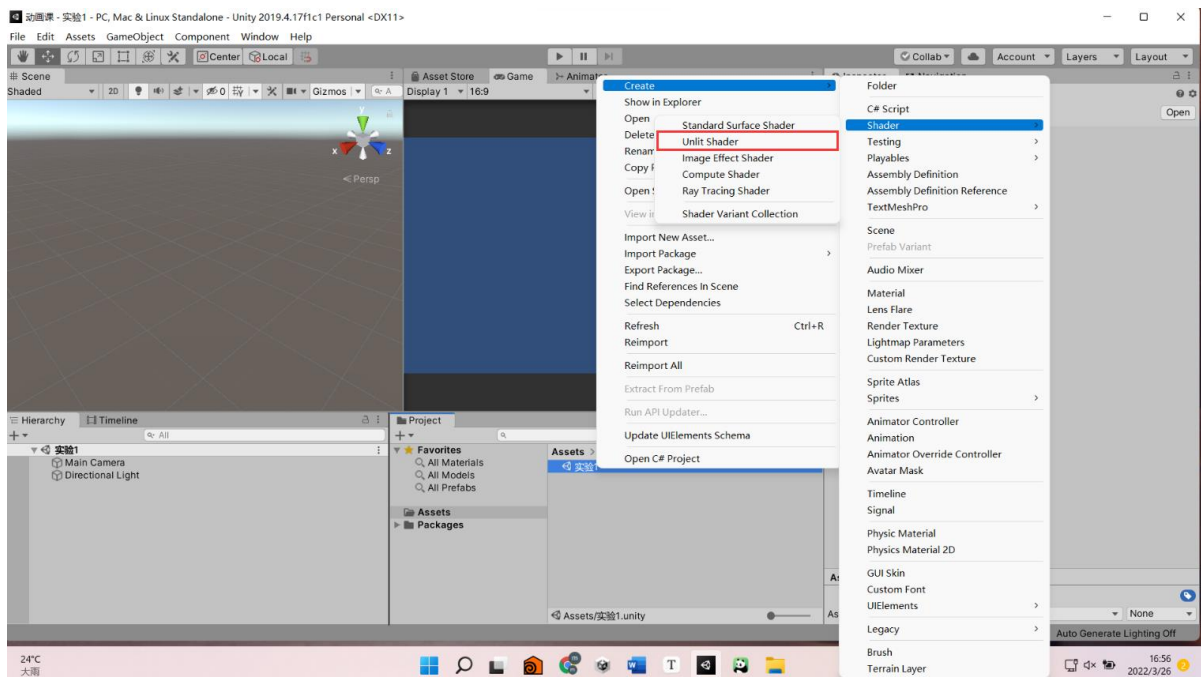


图 5 新建着色器

双击打开它 UnlitShader。

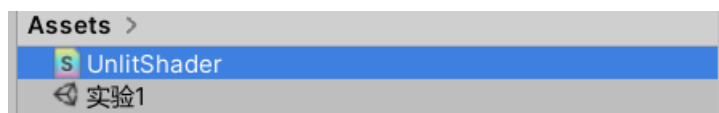


图 6 OpenGLPlat 项目属性页对话框

在“UnlitShader”中添加新的属性，首先将_MainTex 修改为_MainTex1, 然后添加一个新的_MainTex2, 最后添加一个_interpolation 参数供插值使用，如图 7 所示。

```
Properties
{
    _MainTex1 ("Texture1", 2D) = "white" {}
    _MainTex2 ("Texture2", 2D) = "white" {}
    _interpolation("interpolation",range(0,1)) = 0
}
```

图 7 Shader 中新的属性

在图 8 中所示位置，将注释代码改为下面的非注释代码。

```

struct v2f
{
    float2 uv : TEXCOORD0;
    float4 vertex : SV_POSITION;
};

//sampler2D _MainTex;
//float4 _MainTex_ST;
sampler2D _MainTex1;
float4 _MainTex1_ST;
sampler2D _MainTex2;
float4 _MainTex2_ST;

float _interpolation;

```

图 8 新增着色器中的全局变量

删除 v2f vert 和 fixed4 frag 中的一部分代码，如图 9 所示

```

v2f vert (appdata v)
{
    v2f o;
    o.uv = TRANSFORM_TEX(v.uv, _MainTex1);
    o.vertex = UnityObjectToClipPos(v.vertex);
    //UNITY_TRANSFER_FOG(o, o.vertex)
    return o;
}

fixed4 frag (v2f i) : SV_Target
{
    // sample the texture
    fixed4 col = tex2D(_MainTex, i.uv);
    // apply fog
    // UNITY_APPLY_FOG(i.fogCoord, col);
    return col;
}

```

图 9 删除不需要的着色器代码

在片元着色器（frag）中进行插值计算，由参数 interpolation 控制

```
fixed4 frag (v2f i) : SV_Target
{
    fixed4 col = _interpolation * tex2D(_MainTex1, i.uv) + (1 - _interpolation) * tex2D(_MainTex2, i.uv);

    return col;
}
```

图 10 进行图片的插值计算

在 Asset 处右键 Create->Material，创建一个新的材质球。

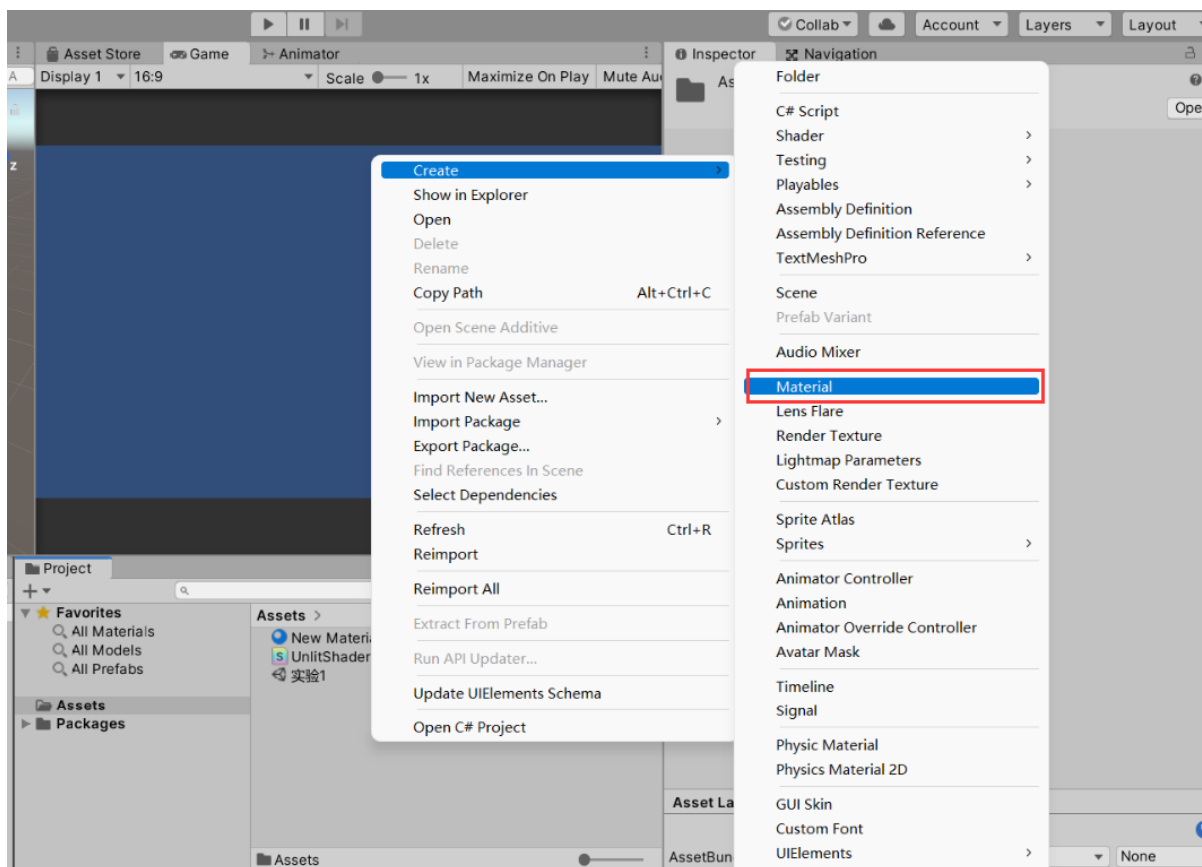


图 11 新建一个材质球

将“UnlitShader”拖到材质球上，该材质球就使用了这个 Shader

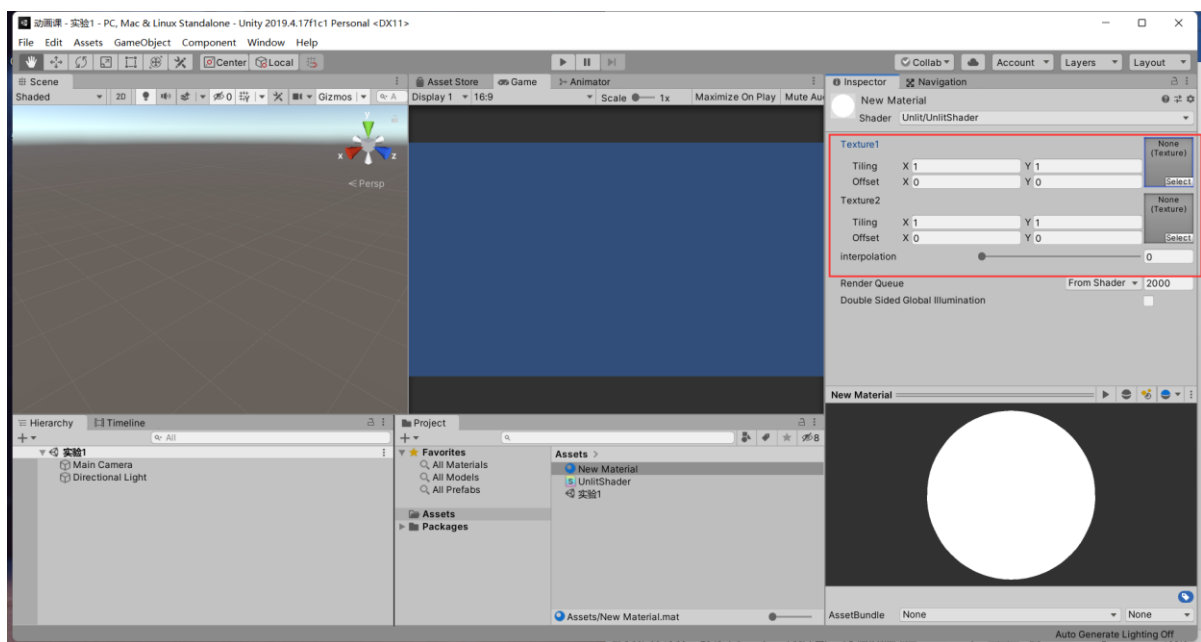


图 12 为材质球附上着色器

3. 初始化插值对象

选择你喜欢的两张图片导入Unity中,并将他们的纹理属性修改为精灵(Sprite)。

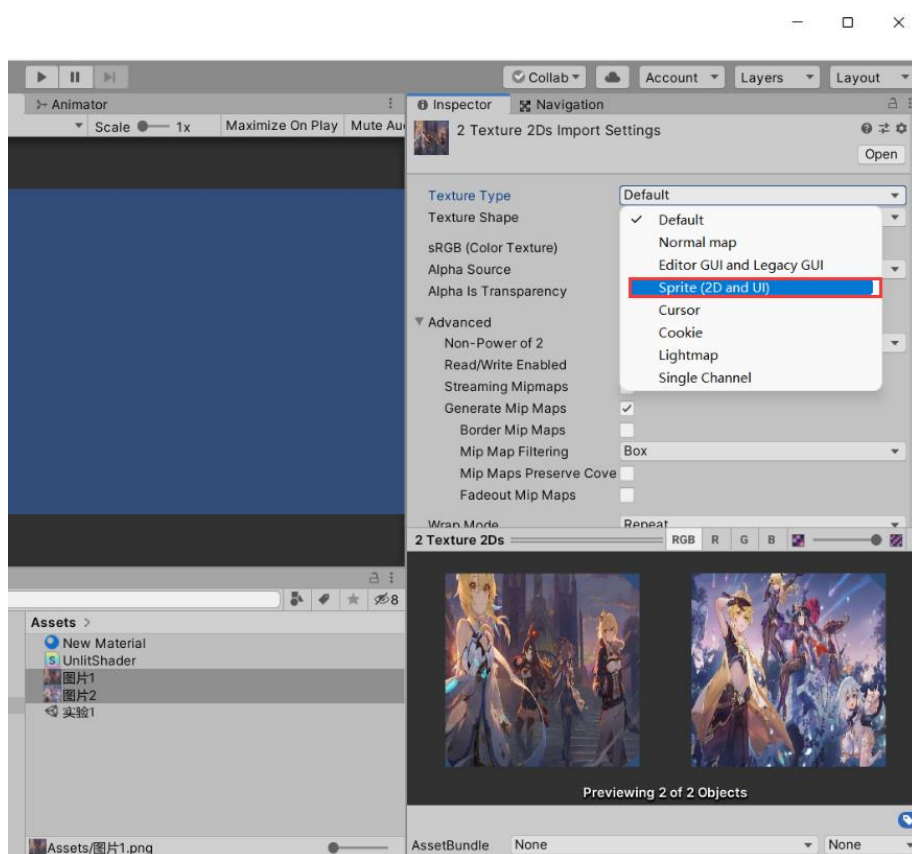


图 13 初始化将要插值的图片

如图 14 所示，将图片 1 导入场景中

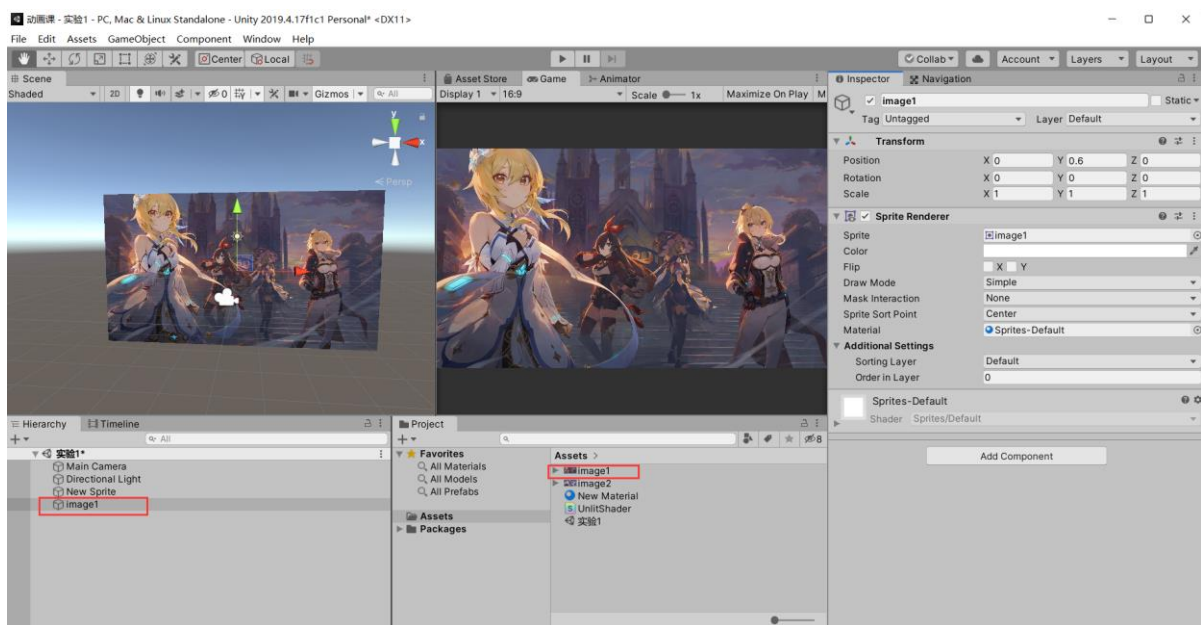


图 14 初始化将要插值的图片

将第二步创建的材质球赋予给这个对象的 Sprite Renderer 组件

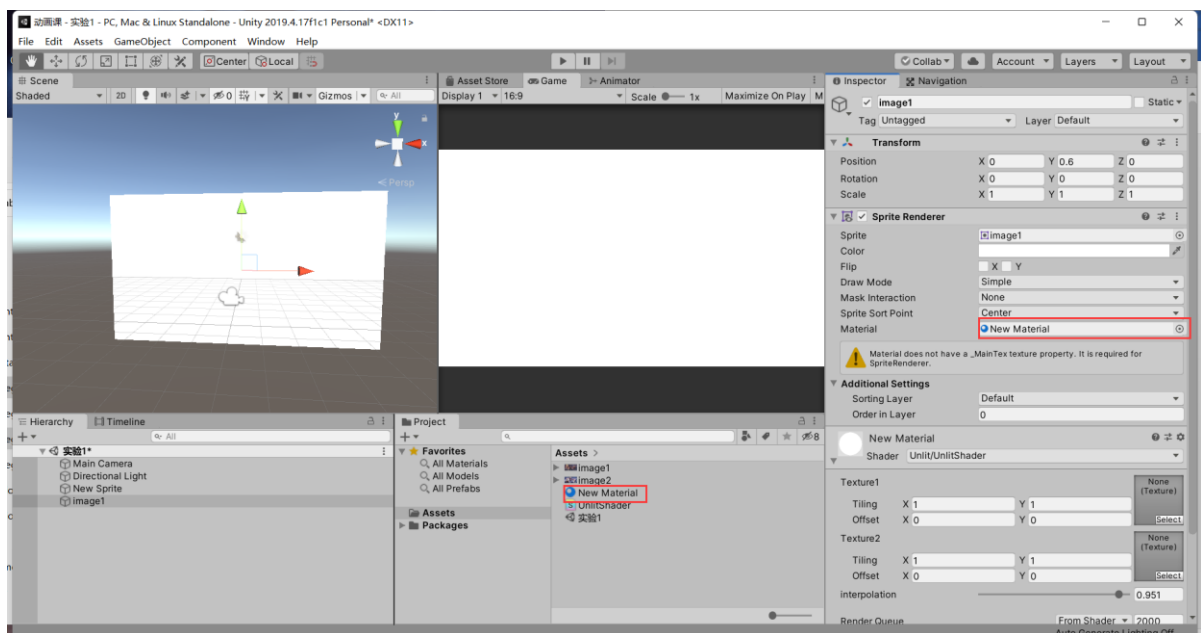


图 14 将材质球赋予这个对象

将两张图片分别放到材质球对应的位置

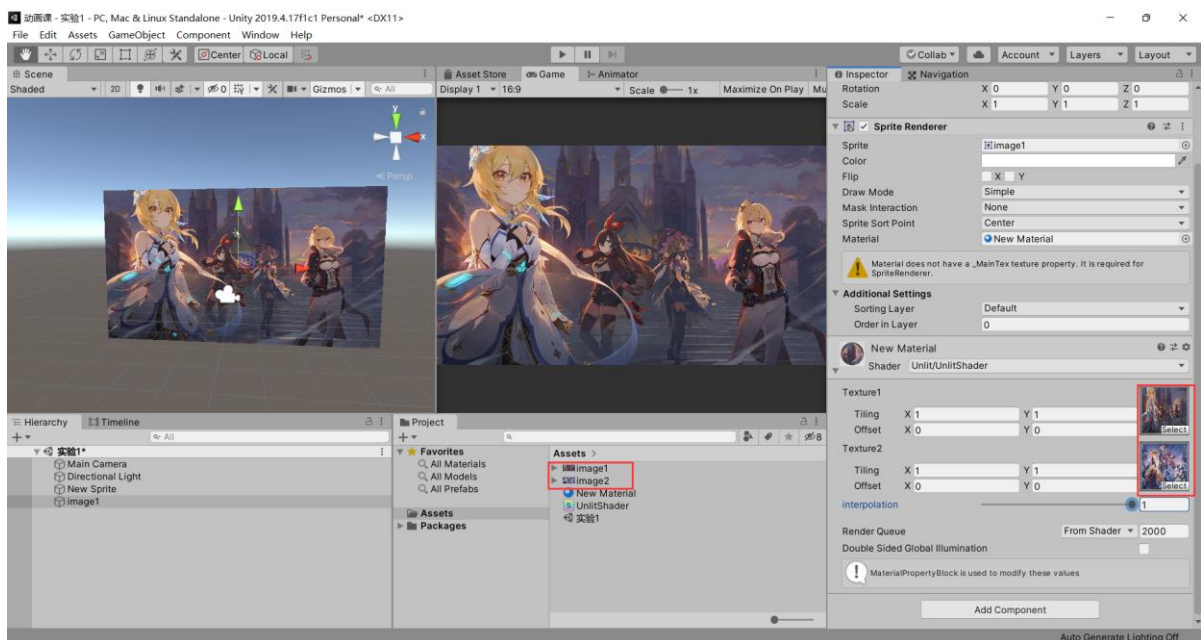


图 15 为材质球赋予图片

4. 实现图片插值动画

在 Assets 中，鼠标右键 Create->C# Script, 新建一个 C#脚本，命名为 Interpolation。

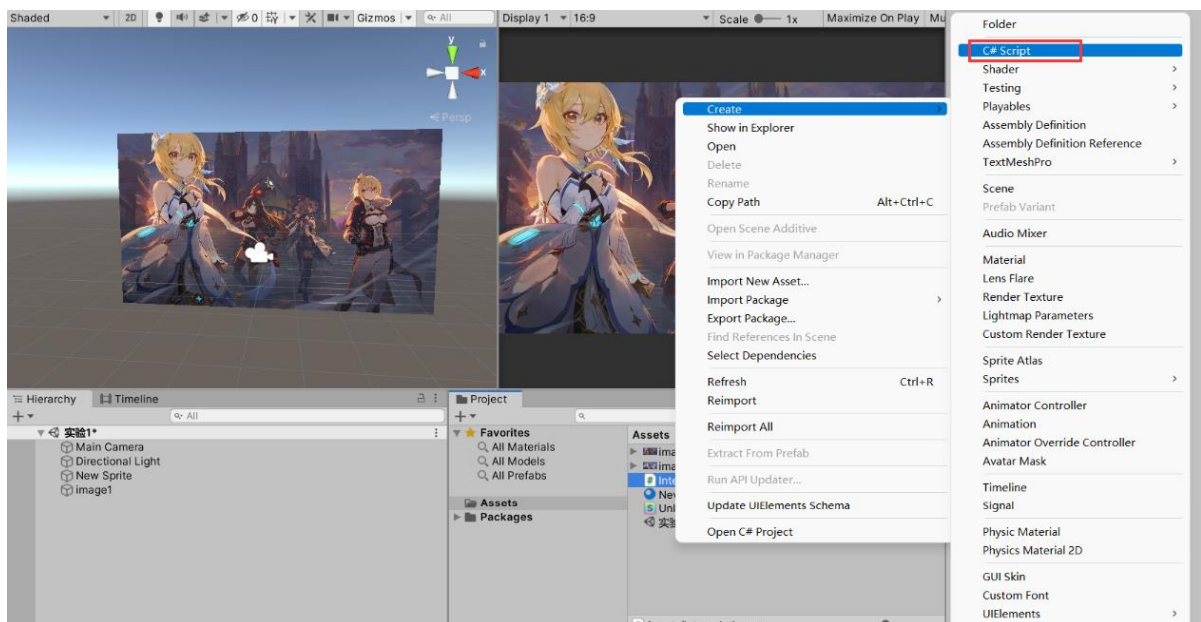


图 16 新建 C#脚本

在 “Interpolation” 脚本中，声明下图属性，供插值使用。

```
public class Interpolation : MonoBehaviour
{
    [SerializeField] private float step = 0.001f;
    private Material material;
    private bool flag = true;
    private float interpolation = 0;
}
```

图 17 声明属性

在 Void Start () 函数中进行材质的初始化，如图 18 所示：

```
void Start()
{
    material = this.GetComponent<SpriteRenderer>().material;
}
```

图 18 初始化材质

在 Void Update () 函数中进行插值的变化，核心思想就是让 interpolation 在 0-1 之间反复震荡，如图 19 所示

```
void Update()
{
    interpolation = flag ? interpolation + step : interpolation - step;
    if (interpolation > 1.0f)
        flag = false;
    if (interpolation < 0.0f)
        flag = true;
    material.SetFloat(name: "_interpolation", interpolation);
}
```

图 19 插值动画

最终的结果应该如图 20 所示：

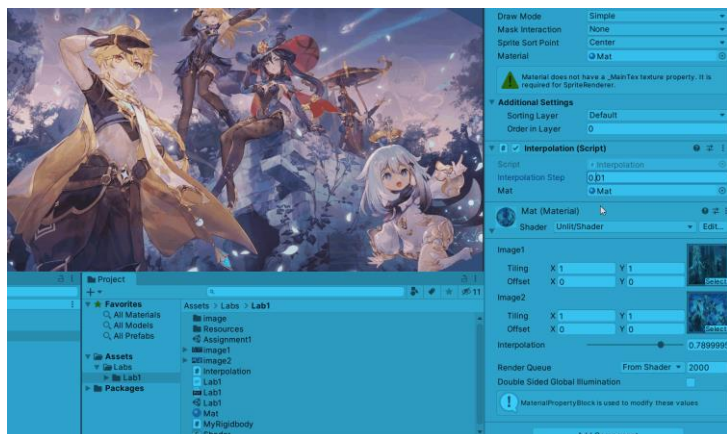


图 20 最终结果

结论分析：

列举实验中遇到的问题、解决的方法，总结实验的收获和体会以及尚存在的问题。

实验要求：

1. 确认机器已经安装 Unity2019（或更高版本）；参见“**实验环境要求**”。
2. 根据“**实验方法**”中步骤依次实践计算机动画编程方法，并达到**熟练掌握**程度。
3. 给 4 中添加/修改的代码逐句添加注释，可参考“参考书目”中教材或网络上资料。
4. 提交材料：**本次实验**（实验 1）需要填写实验报告并且提交*.Unitypackage文件。实验报告内容包括实验步骤、结果图、注释后的代码以及“**结论分析**”中所述内容等。实验报告必须是在 **Microsoft Office Word** 中进行格式排版后的报告。
5. 命名规则：学号_姓名，注意中间为“下划线”，**未按规则命名者按未交作业处理**。
6. **本课程所有实验的报告要求集成在一个文档中，并在课程结束后规定时间内提交即可，即课程结束前不必分开提交“实验 1”、“实验 2”等实验材料。**
7. 提交地址：ftp://211.71.149.149/yang_meng/homework/相应目录下。

参考书目：

- 计算机动画算法与编程基础，雍俊海 著，2008.7，清华大学出版社。

教研室主任意见：

签字：_____ 年 月 日

学院负责人意见：

签字：_____ 年 月 日

注：此表一式两份，一份于考前交到考试中心，一份随学生课程设计材料上交学院。