



唐老狮系列教程

# 利用深度纹理实现 运动模糊效果 的基本原理

WELCOME  
TO THE  
UNITY  
SPECIALTY COURSE  
STUDY

版权所有：唐老狮 tpandme@163.com



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## | 主要讲解内容



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 主要讲解内容

1. 知识回顾 运动模糊效果
2. 关键注意点说明
2. 利用深度纹理实现运动模糊的基本原理



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 知识回顾 运动模糊效果





# 唐老师系列教程-深度纹理实现运动模糊基本原理

## 知识回顾 运动模糊效果

运动模糊效果，是一种 **用于模拟真实世界中快速移动物体产生的模糊现象** 的图像处理技术。

一般有两种常用方式：

**1.累积缓存：**物体快速移动时存储多帧图像信息，取它们之间的加权平均值作为最后的运动模糊图像

**优点：**质量高、效果好；**缺点：**计算量大，存储开销大

**2.速度缓存：**物体快速移动时存储多帧运动速度信息，利用速度来决定模糊的方向和大小

**优点：**性能较累积缓存好；**缺点：**效果较差，可能产生重影和伪影

我们之前学习的方式是基于累积缓存的，但是并没有存储多张场景信息

**而是采用一张渲染纹理RenderTexture保存之前的渲染结果，不断把当前渲染图像叠加到之前的渲染图像中，从而产生一种运动轨迹视觉效果**



# 唐老师系列教程-深度纹理实现运动模糊基本原理

## 知识回顾 运动模糊效果

而我们这节课将要学习利用深度纹理实现运动模糊

它其实就是基于我们之前提到的 **速度缓存** 的方式进行。

但是和我们之前学习的运动模糊效果一样，我们会基于速度缓存基本规则进行修改

**只需要用当前帧位置和上一帧位置进行计算，得到位置差，从而得到该像素的速度矢量。**

**想要得到位置差，我们可以利用深度纹理中的信息来进行计算。**

接下来，我们就来详细的学习它的基本实现原理



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## | 关键注意点说明





# 唐老师系列教程-深度纹理实现运动模糊基本原理

## 关键注意点说明

我们将要讲解的利用 深度纹理实现运动模糊效果 的这种方式有以下两点需要注意：

- 1. 这种实现方式 只适合 场景静止，即摄像机快速移动的情况，  
它不太适用于物体快速移动产生的运动模糊效果，只有摄像机移动时才能看到运动模糊效果**
- 2. 这种实现方式 并不是基于真实的物理运动规律来计算的，只是一种近似计算！  
它符合图形学基本规则：看起来对那么就是对的**

虽然它使用上有局限性，并且实现上也不符合物理计算规则，  
但是由于它实现出来的**效果和性能消耗都还不错**，因此它也是一种常用的运动模糊处理方案。  
通过对它的讲解，**主要让同学们感受下相同效果的不同实现思路！**





# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 利用深度纹理实现运动模糊基本原理



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 利用深度纹理实现运动模糊基本原理

首先，一句话描述它的基本原理：

**得到像素当前帧和上一帧中在裁剪空间下的位置，  
利用两个位置计算出物体的运动方向，从而模拟出运动模糊的效果。**

其中的关键点：

1. 如何得到像素当前帧和上一帧在裁剪空间下的位置
2. 如何得到运动方向
3. 如何模拟运动模糊效果



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

如何得到像素当前帧和上一帧在裁剪空间下的位置



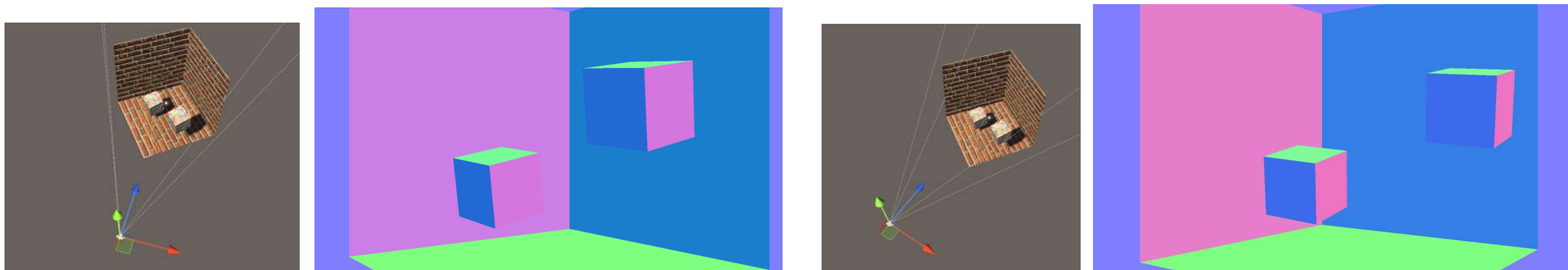


# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 如何得到像素当前帧和上一帧在裁剪空间下的位置

关键步骤：

1. 利用UV坐标和深度值组合成一个裁剪空间下的组合坐标  $\text{nowClipPos}$
2. 利用 这一帧 世界空间 $\rightarrow$ 裁剪空间 的变换矩阵  $\text{nowM}$  的逆矩阵  $\text{nowM}^{-1}$   
将刚才裁剪空间下的组合坐标 $\text{nowClipPos}$ 转换到世界空间中
3. 利用 上一帧的 世界空间 $\rightarrow$ 裁剪空间 的变换矩阵 $\text{oldM}$ 得到  
上一帧该世界空间下的组合坐标 $\text{oldClipPos}$ 在裁剪空间下的位置





# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 如何得到像素当前帧和上一帧在裁剪空间下的位置

关键步骤一：

利用UV坐标和深度值组合成一个裁剪空间下的组合坐标 `nowClipPos`

```
float4 clipPos = float4(uv.x, uv.y, depth, 1);
```

我们知道uv坐标空间下的值是0~1，通过宏取出来的深度值也是0~1，而裁剪空间下的坐标范围是-1~1

因此，我们需要将它利用简单的公式转换到裁剪空间坐标系下

```
float4 clipPos = float4(uv.x * 2 - 1, uv.y * 2 - 1, depth * 2 - 1, 1);
```

你可以认为这一步是把

带有深度值的uv坐标转换到了裁剪空间下

之所以加入深度值是为了让之后的裁剪空间转到世界空间或其他坐标空间更合理

相当于是把UV的2D坐标系转换到了3D坐标系中





# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 如何得到像素当前帧和上一帧在裁剪空间下的位置

关键步骤二：

**利用 世界空间—>裁剪空间 的变换矩阵的逆矩阵，将刚才裁剪空间下的nowClipPos转换到世界坐标中**

```
//相机投影矩阵到裁剪空间的变换矩阵
camera.projectionMatrix;
//世界空间到观察空间的变换矩阵
camera.worldToCameraMatrix;
//世界空间 到 裁剪空间的变换矩阵
Matrix4x4 worldToClipMatrix = camera.projectionMatrix * camera.worldToCameraMatrix;
//裁剪空间 到 世界空间的变换矩阵 (是worldToClipMatrix的逆矩阵)
Matrix4x4 clipToWorldMatrix = worldToClipMatrix.inverse;
```

我们只需要利用上面的这个裁剪空间到世界空间的变换矩阵，就可以将刚才得到的裁剪空间下的组合坐标变换到世界空间下了。

知识点回顾：逆矩阵的几何性质之一就是可以用它来进行逆向变换



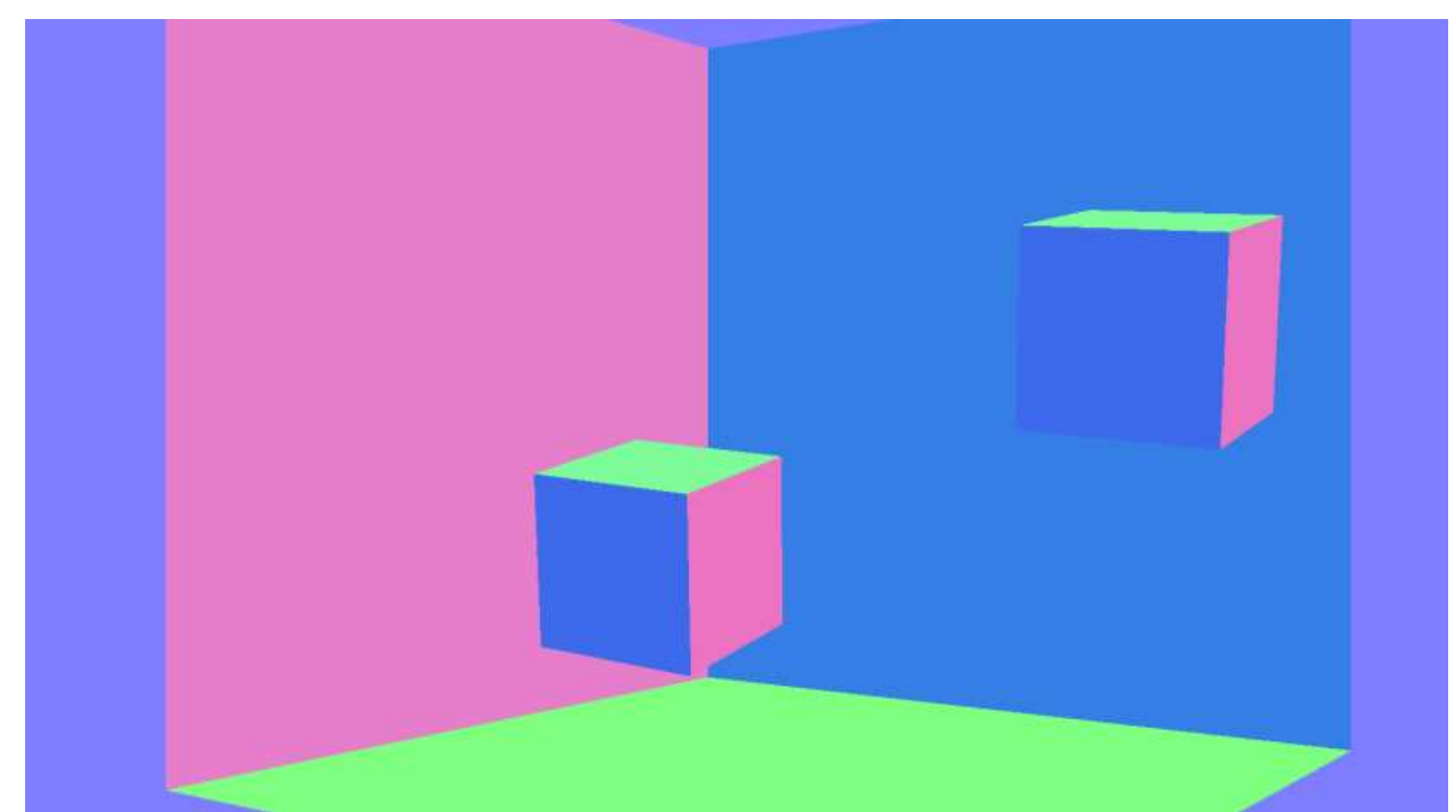
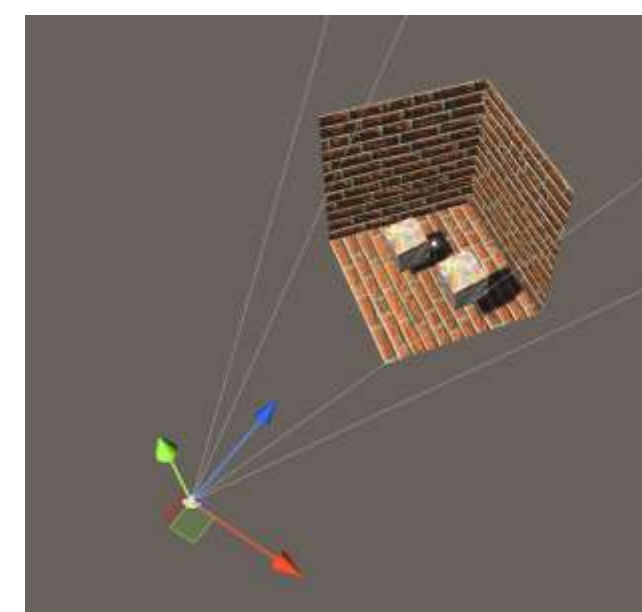
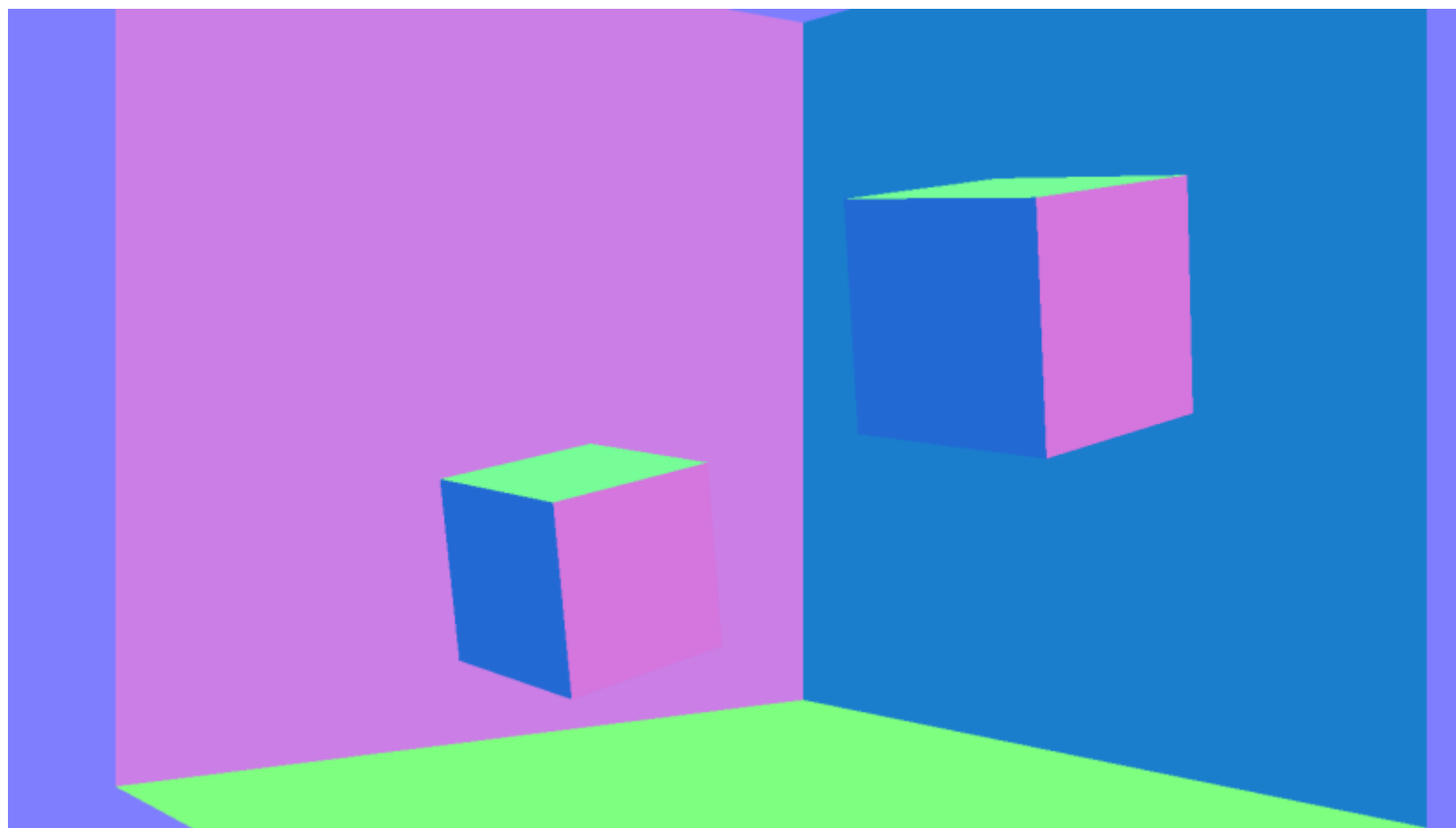
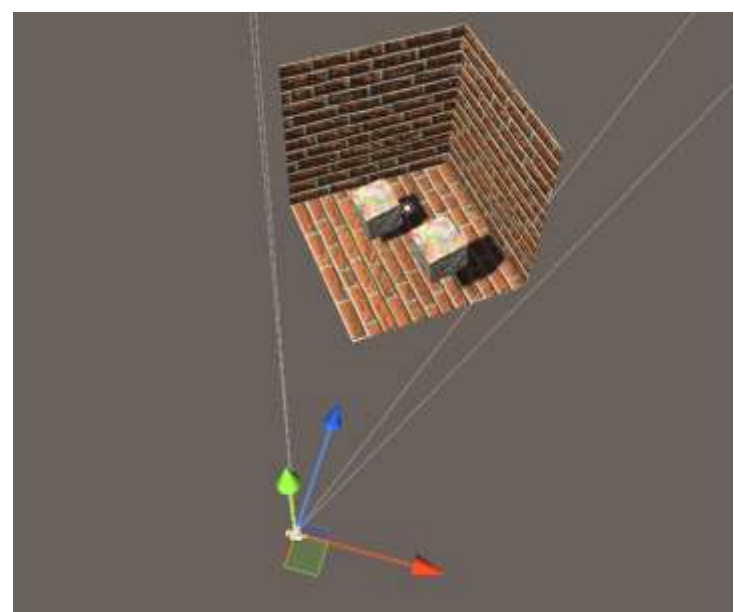


# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 如何得到像素当前帧和上一帧在裁剪空间下的位置

关键步骤三：

利用上一帧的世界坐标—>裁剪空间的变换矩阵得到上一帧该世界空间下的组合坐标在裁剪空间下位置





# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## | 如何得到运动方向

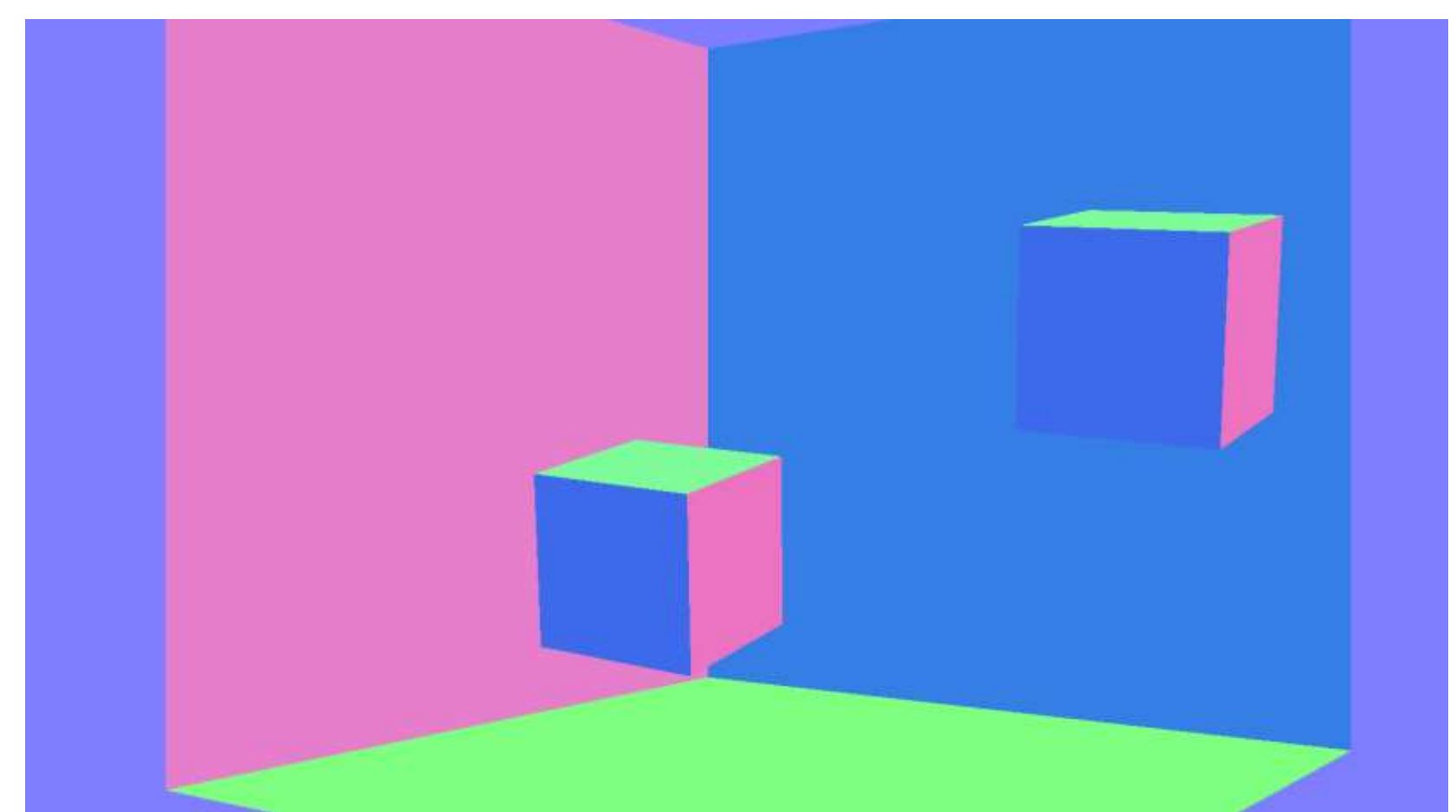
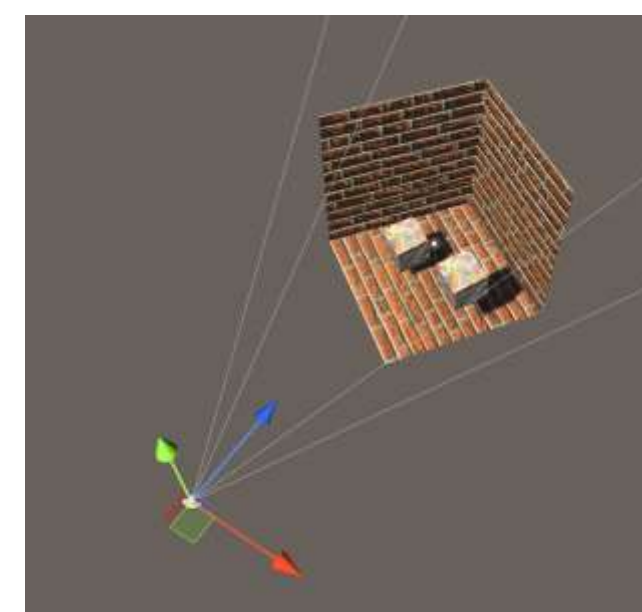
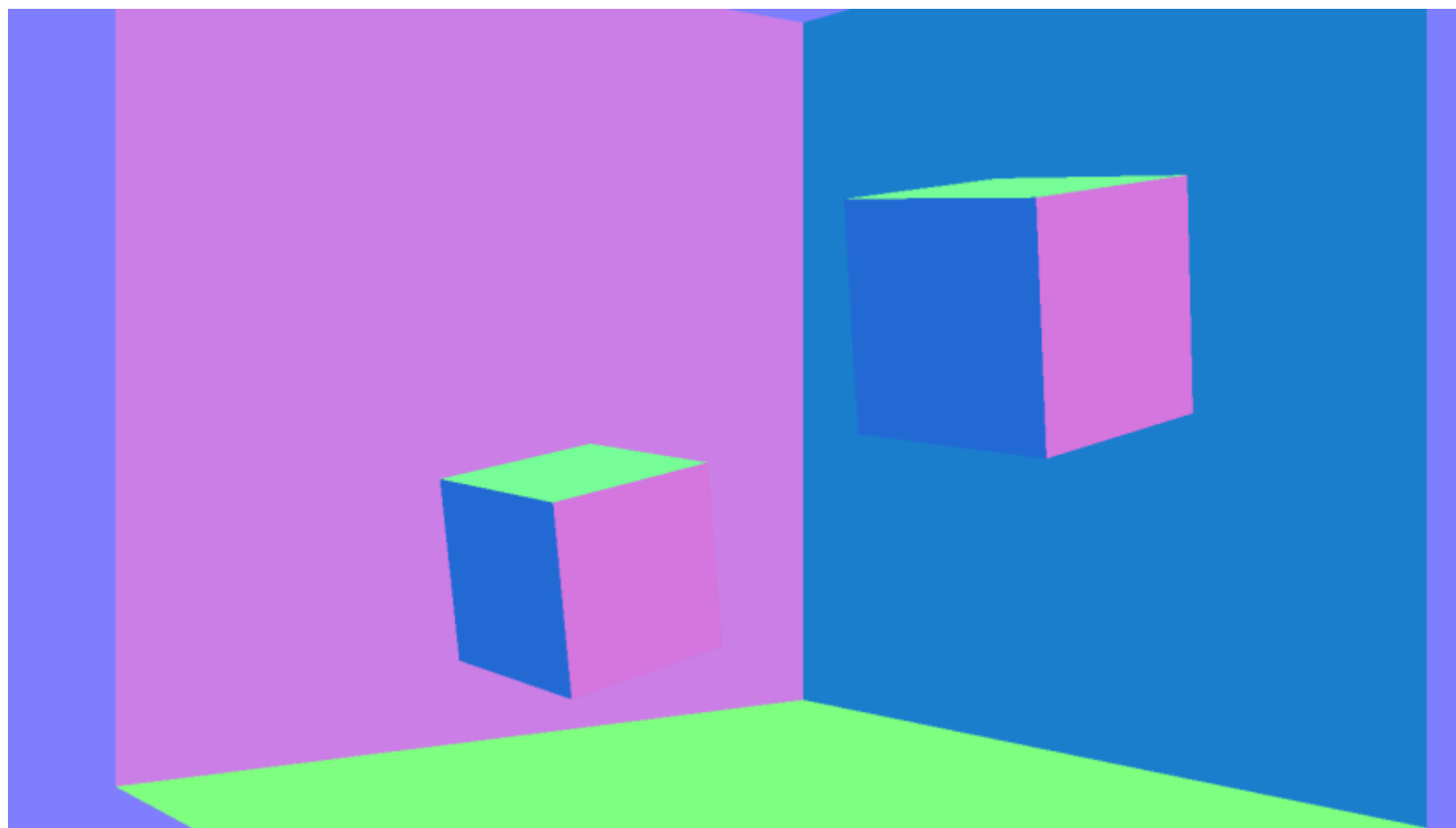
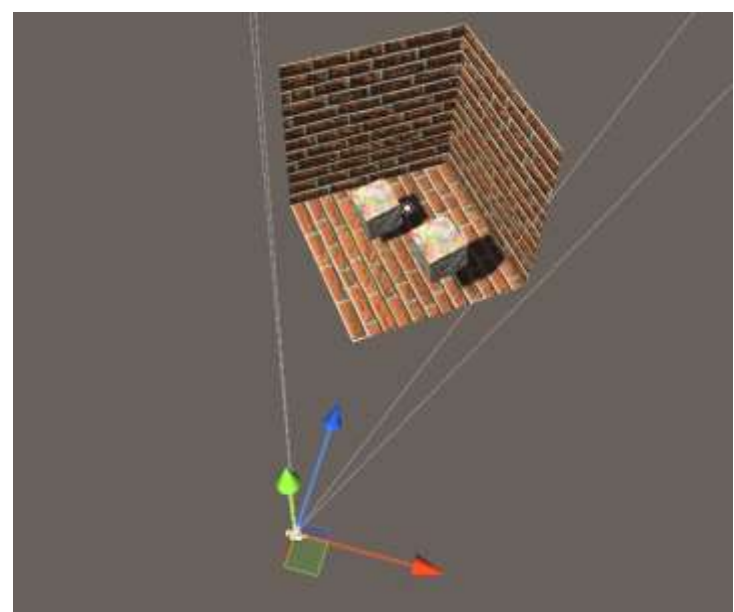


# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 如何得到运动方向

我们已经知道如何得到像素当前帧和上一帧在裁剪空间下的位置

直接用 当前位置.xy - 上一帧位置.xy 便可以得到移动方向







# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## | 如何模拟运动模糊效果



# 唐老师系列教程-深度纹理实现运动模糊基本原理

## 如何模拟运动模糊效果

有了像素在裁剪空间的移动方向，相当于知道了像素在“uv纹理空间的移动方向”

那么我们只需要利用这个方向在纹理中进行多次uv坐标偏移采样后，将得到的颜色累加起来，最后进行算数平均值计算即可

我们会加入一个 **模糊偏移量** 来控制模糊程度

只需要在每次采样时进行 **方向 \* 模糊偏移量** 的偏移采样即可

```
//uv初始值
float2 uv = i.uv;
//用于累加的颜色变量
float4 c = float4(0,0,0,0);
//for循环中进行3次采样，每次采样结束后进行一次偏移
//3次采样的uv坐标对应为：
//第一次： uv
//第二次： uv + 移动方向 * 模糊偏移量
//第三次： uv + 2 * 移动方向 * 模糊偏移量
for (int it = 0; it < 3; it++) {
    //颜色累加
    c += tex2D(_MainTex, uv);
    //uv偏移
    uv += 移动方向 * 模糊偏移量;
}
//算数平均值
c /= 3;
```





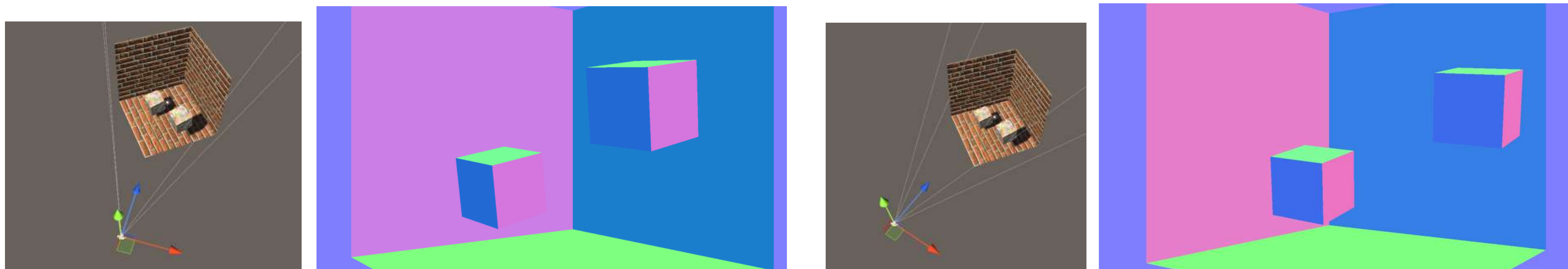
# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 利用深度纹理实现运动模糊基本原理

得到像素当前帧和上一帧中在裁剪空间下的位置，  
利用两个位置计算出物体的运动方向，从而模拟出运动模糊的效果。

关键点：

1. 如何得到像素当前帧和上一帧在裁剪空间下的位置
2. 如何得到运动方向
3. 如何模拟运动模糊效果







# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## | 总结



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 主要讲解内容

### 1. 知识回顾 运动模糊效果

**是一种 用于模拟真实世界中快速移动物体产生的模糊现象 的图像处理技术**  
**之前学习的方式是基于累积缓存的，这节课是基于速度概念来做的**

### 2. 关键注意点说明

**这种实现方式 只适合 场景静止，即摄像机快速移动的情况，**  
**它不太适用于物体快速移动产生的运动模糊效果。**

**这种实现方式 并不是基于真实的物理运动规律来计算的，只是一种近似计算！**

**它只是符合图形学基本规则：看起来对那么就是对的**



# 唐老狮系列教程-深度纹理实现运动模糊基本原理

## 主要讲解内容

### 3. 利用深度纹理实现运动模糊的基本原理

**得到像素当前帧和上一帧中在裁剪空间下的位置，  
利用两个位置计算出物体的运动方向，从而模拟出运动模糊的效果。**





# 唐老狮系列教程

Thank

谢谢您的聆听