**Bloom**

**什么是Bloom效果？**

让画面中较亮的区域扩散到周围区域中，造成一种朦胧的效果。

**Bloom的实现原理是什么？**

根据一个设定阈值提取图像中较亮区域，把它们存储到一张RT中，然后利用高斯模糊对这张纹理进行模糊处理，模糊图像会将明亮的像素渗入到较暗的像素，使其看起来会发光，以此模拟光线扩散的效果，然后将它和原图像进行混合得到最终效果。

**叠加模糊**

使用Bloom金字塔的顶部作为最终图像产生统一的混合，但它看起来不像什么发光的东西。我们可以通过逐步向上采样回到金字塔底部，在一张图像中累积所有的级别来得到想要的结果。

**双三次滤波上采样**

虽然高斯滤波会产生平滑的结果，但我们在上采样时仍执行双线性滤波，这可能会使辉光显得像块状。这在原始图像中收缩较高的地方（尤其是在运动时）中最为明显。

我们可以通过切换到双三次滤波来平滑结果。对此虽然没有硬件支持，但我们可以使用源码库中Filtering.hlsl文件中定义的SampleTexture2DBicubic方法来解决。我们定义一个GetSourceBicubic方法，在里面调用SampleTexture2DBicubic方法，需要传递源纹理和采样器、屏幕UV坐标以及纹素尺寸\_PostFXSource\_TexelSize通过ZWXY排序的矢量。除此之外还有一个最大纹理坐标的参数，该参数为1，以及另一个未使用的参数，该参数可以为零。然后在BloomCombinePassFragment方法中采样第一个源纹理时使用双三次滤波进行上采样。

**一半的分辨率**

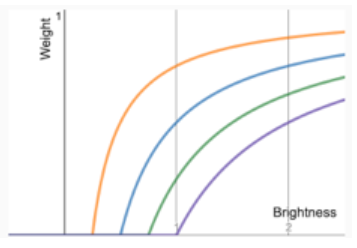
由于所有的纹理采样和绘制，Bloom可能需要大量时间才能生成。一种简单的降低成本的方式是以一半的分辨率生成它，这将更改效果的外观，因为实际上相当于跳过了第一次迭代。

**阈值**

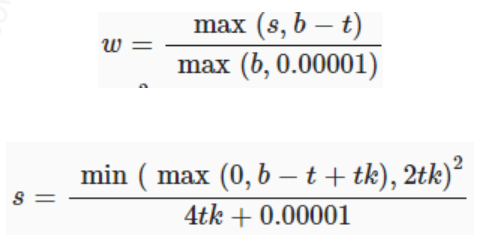
现在我们的Bloom效果适用于所有对象，让它们发光。通常我们需要设定一个阈值来提取图像中较亮的区域，对其进行模糊处理后模拟光线扩散的效果。但我们不能突然消除效果中的颜色，因为这会在逐渐过渡的地方引入尖锐的边界，相反应该将颜色乘以一个权重：



​ 其中b为亮度，t是配置的阈值，我们使用最大的颜色RGB通道为b，阈值为0时结果总是为1，这将保持颜色不变，随着阈值增加，权重曲线会向下弯曲，在b<=t处为0。由于曲线的形状很像膝盖，它也被称为膝盖曲线。下面是阈值分别为0.25、0.5、0.75和1时的曲线。



​该曲线在某个角度达到0，意味着虽然过渡比Clamp更平滑，但仍然有一个突然的终止点。我们可以通过改变权重来改变膝盖的形状。



其中k是膝盖，为0-1的滑动区间。下图是阈值为1，膝盖分别为0、0.25、0.5、0.15和1时的曲线。

