图形学大作业报告

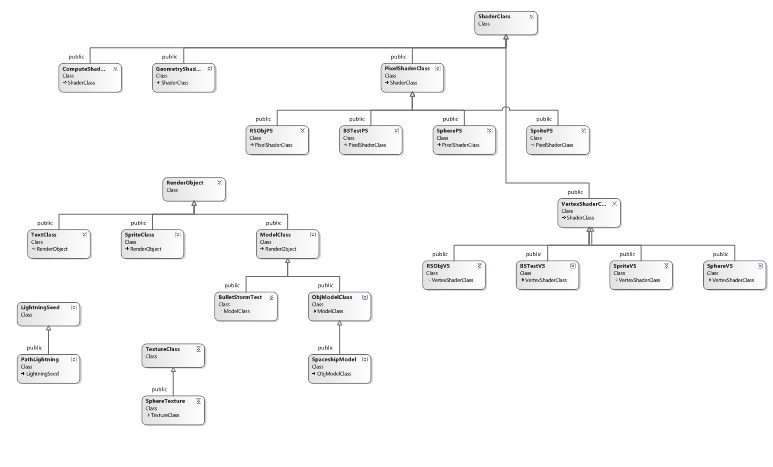
1. 同组人员

陈星丞（组长）、想谢华梁、邓宏琛、黄骁。

1. 程序引擎

在本次大作业中，我们实现了一个初具规模的游戏引擎，可以很方便的创造游戏场景、提供交互、设计游戏逻辑等，为基本的游戏规则与创造提供了合理的逻辑接口。

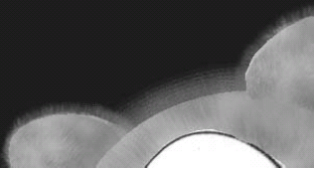
游戏引擎类图：



1. Shader运用
   1. Fur Shells And Fins

实现分为Shell和Fin的两部分。Shell是利用AlphaBlend的方法，逐层将原有模型向外扩展，绘制一层层带有透明度的外壳并叠加而成。缺点是不能显示出毛发的线状效果。

如下图：

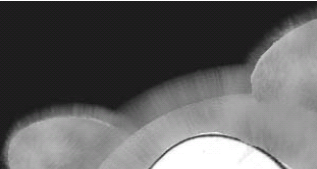


为了弥补此缺陷，NViDIA的例子中又创造了Fin去模拟线状效果。Fin是通过Geometry Shader在垂直于视线的方向建立新的三角形并进行绘制得到的。

如下图：

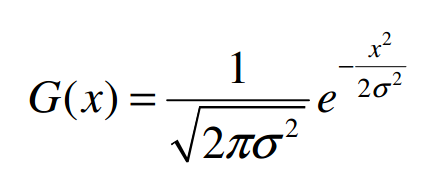


合并效果：

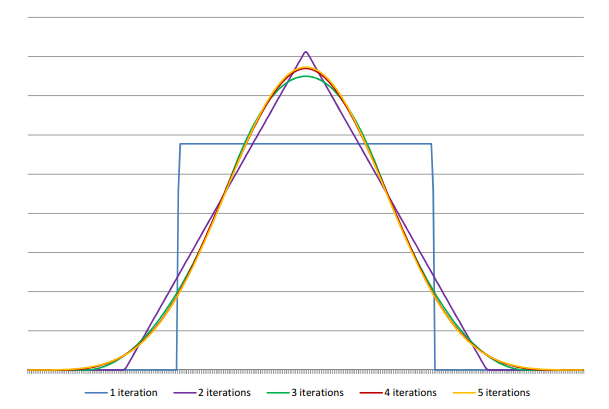


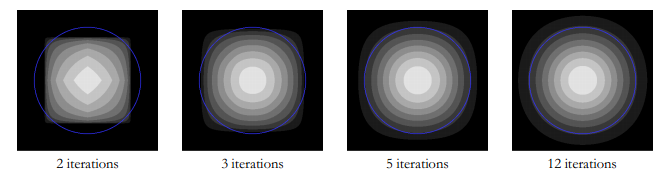
* 1. Constant Time Gaussian Blur

高斯模糊是指用高斯核对图片进行模糊处理。



高斯滤波可以通过方盒滤波进行近似模拟。





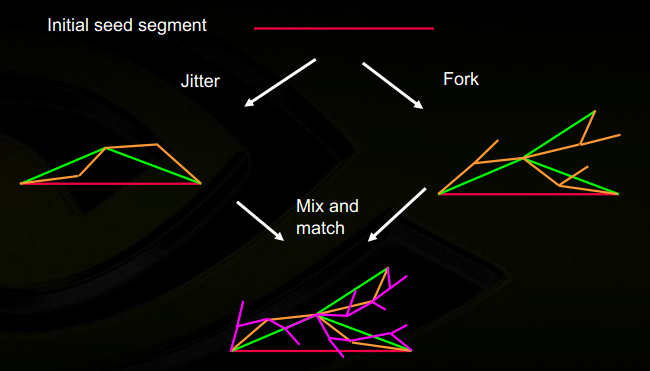
同时我们知道，使用前缀和的方法可以在常数时间里获得任意大小的方盒的和，此即Constant Time的由来。

在实际实现中，先把全部屏幕绘制到一个texture buffer上，再对行、列进行高斯滤波，得到最终的结果。

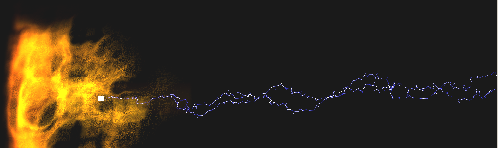


* 1. Lightning

首先通过GeometryShader产生对应折线，之后产生和线段以及屏幕平行的四边形用于绘制。

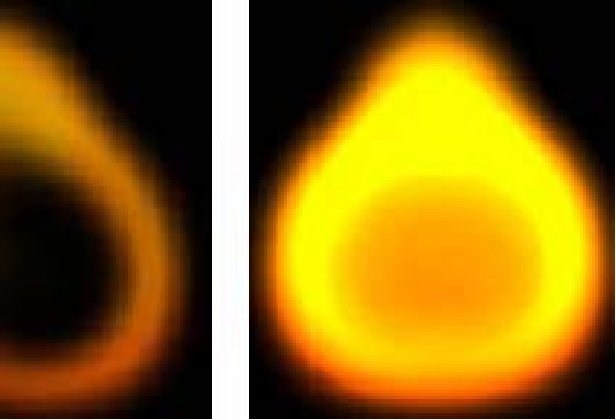


游戏中：



* 1. Perlin Fire

在一个立方体中使用simplex noise对一旋转的火焰贴图进行扰动：

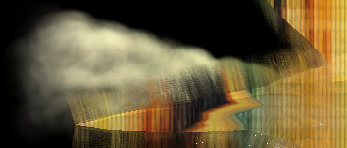


具体实现是每一个像素分别计算多层，再求和，即是多层扰动的火焰效果的叠加。

* 1. Soft Particles

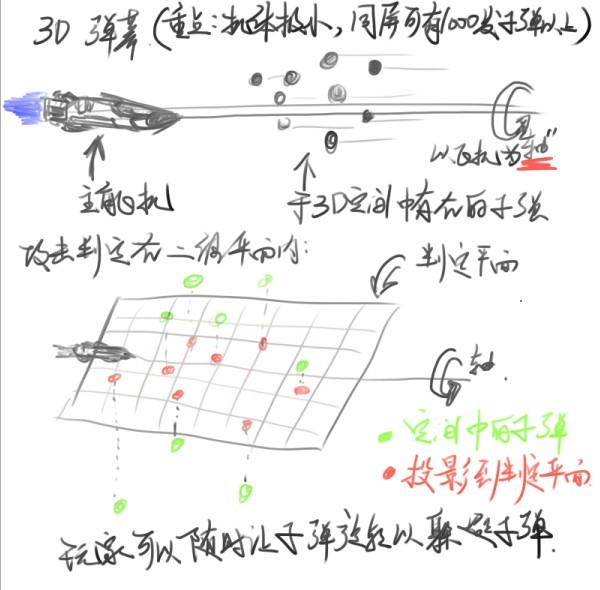
创建于屏幕平行的四边形绘制烟雾,并使用深度缓存控制粒子。

游戏中：



1. 游戏规则介绍

如下图：



1. 最终效果