

Readme

小组

第三小组

成员：彭晨晗、刘嘉熙、古丽斐热、张祺

项目简介

小组选择的项目为实现烟花粒子特效系统，模拟了烟花在空中爆炸的过程。在这个系统中，你可以看到烟花发射，烟花爆炸并可以在过程中听到烟花发射、升空以及烟花爆炸的声音。

系统还实现了烟花光照效果，在每朵烟花的爆炸中心激活了一个点光源，从而使得烟火闪烁的效果更加凸显，项目以一个黑夜的天空盒作为背景，黑夜中烟花绽放照亮夜空，颇有极光氛围。

使用了高斯模糊等后处理技术，使得烟花周围有一个光晕效果，实现了辉光特效。

总体上实现了烟花爆炸瞬间的光辉璀璨效果。



实现方法

粒子系统

实现了多种类型的烟花，例如发射烟花，喷泉，渲染烟花尾迹，生成爆炸后的火花等Particle 粒子结构

1. Particle 粒子结构

结构体包含了粒子的位置、速度、颜色、大小、寿命以及类型等信息

其类型有如下几种enum Type { LAUNCHING, SPARKLE, TRAIL, FOUNTAIN, DEAD, METEOR, SNOW };发射，闪烁，尾迹，喷泉，死亡，流星和雪花

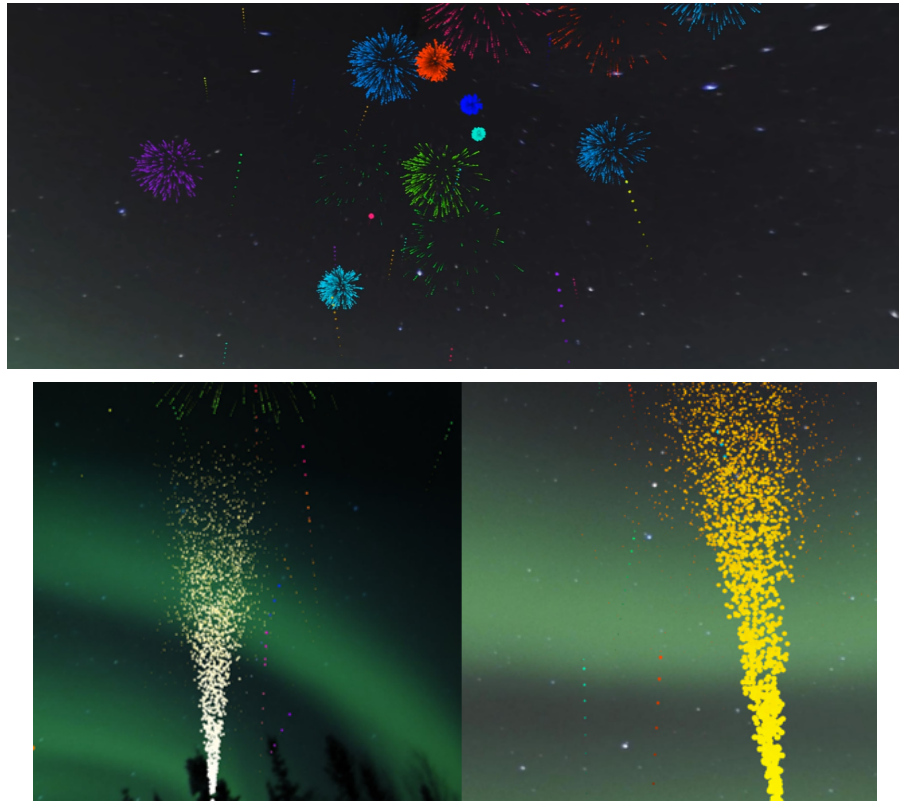
当一个烟花的生命值大于零时，它就是活着的。对于活着的烟花，它的速度会因重力而改变，位置因速度而改变。当一个烟花的生命值小于等于零时，转为Dead。如果这个烟花的类型是发射，它就会爆炸。

2. Launcher类

该类负责管理所有烟花，它包含了一个粒子数组，每个粒子都有自己的位置、速度、颜色、大小、生命周期和类型。并负责模拟所有烟花的运动，包括处理重力和加速度的影响

每一帧渲染时被调用，模拟函数遍历所有粒子，对每个粒子更新其位置和速度，并且计算出每个粒子与摄像机的距离

3. launchFirework 函数实现**发射烟花**，launchFountain 函数实现**烟花喷泉**，renderTrails 函数实现烟花尾迹



4. 实现了重力加速度，使烟花呈现出从高处坠落的效果

5. sortParticles函数防止遮挡，保证距离近的粒子会被渲染在距离远的粒子之前

天空盒

使用六张图片构建立方体(3D)纹理贴图

如果一个粒子的位置低于地面的高度或者高于天空盒的高度，那么这个粒子就会被标记为死亡状态。在下一帧时，这个粒子不会再被渲染；使用深度测试函数，使天空盒看起来是远方的背景，而不是一个可以穿过的盒子或者屏幕，并限制相机位置。

烟花光照

Blinn-phong光照模型：环境光照、漫反射光照和镜面光照

在每朵烟花的爆炸中心，激活一个点光源，将该点光源的颜色选定为烟花的颜色，爆炸结束后再删除这个点光源。

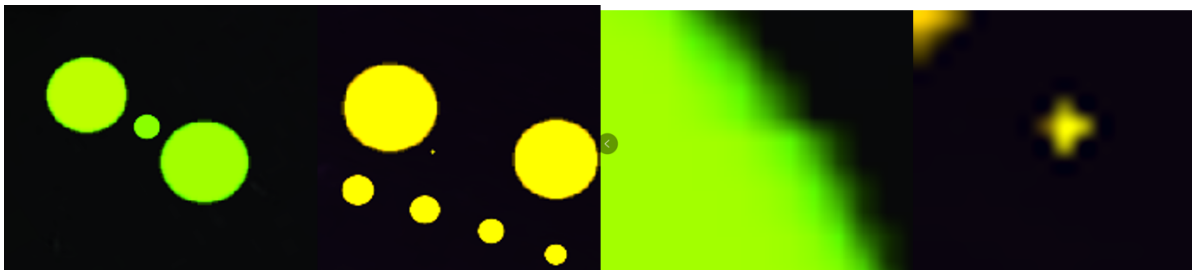
定义Light类型，其中包含位置、颜色和生命周期的信息



辉光特效

首先，在帧缓冲区中渲染烟花，然后将帧缓冲区内容复制到纹理。其次，使用一个片段着色器来对这个纹理进行高斯模糊，最后将结果渲染到屏幕上。

在着色器中对烟花的爆炸帧进行高斯模糊处理，利用一个权重数组，包含了高斯权重函数的值，定义模糊程度，对图像中每个像素使用多个权重进行加权平均，然后将得到的值作为新的像素值来更新图像，达到辉光特效效果



音效

使用了 irrklang 库实现音效，Launcher构造函数中，使用 `irrklang::createIrrKlangDevice()` 函数创建一个 irrklang 设备。这个设备是用于播放音效的主要接口

烟花发射、爆炸时，使用 `play2D()` 函数播放音效

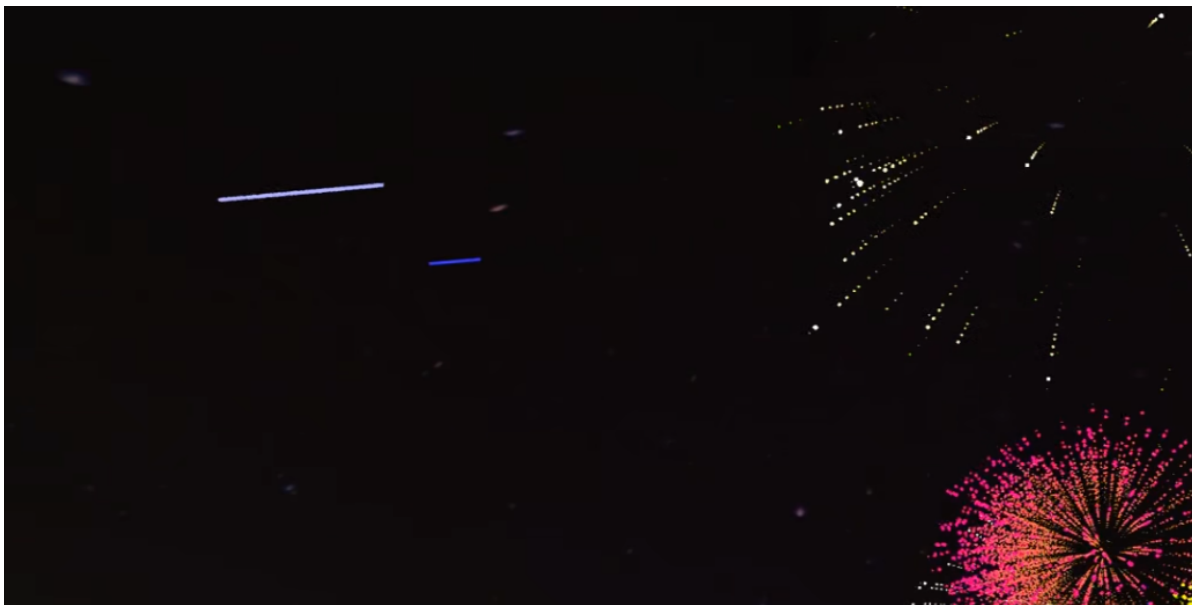
落雪

下雪功能是通过生成一些粒子来模拟下雪的效果，每个粒子都有自己的位置、速度和生命周期。粒子在生命周期内会从顶部（天空）向下移动，并根据函数产生的随机偏移来模拟雪花飘动的效果



流星

流星功能也是通过生成粒子来实现的，每个粒子都有自己的位置、速度和生命周期。流星会从天空中飞过。



其他

演示

整体效果请见demo目录中的演示图片与视频

代码与运行

代码位于/code文件夹，可执行文件位于/release文件夹

