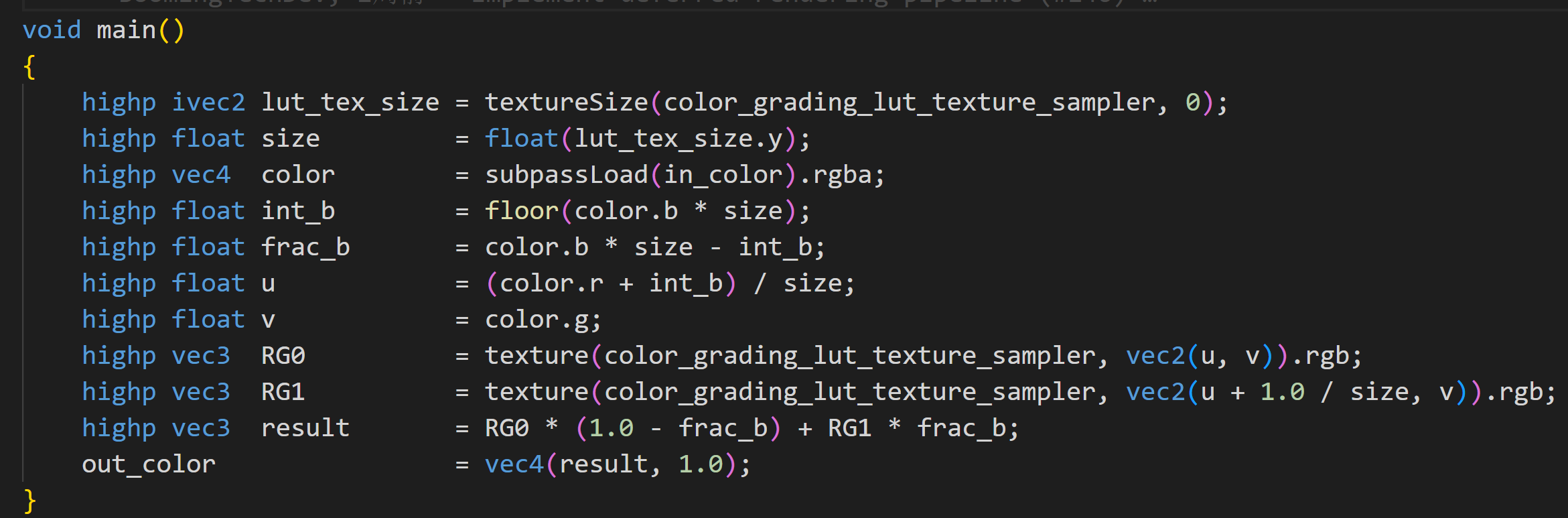
### Homework2

王宇鑫 [railgunhamster@qq.com](mailto:railgunhamster@qq.com) / 微信GAMES104课程群10：我，莫得感情 / 微信号：RailgunHamster

1. 基础：



抄的ue4的。由于我们用的是2D的一张图256\*16，而不是16\*16\*16的3D纹理。

所以没法直接rgb三个维度的uv直接求值。

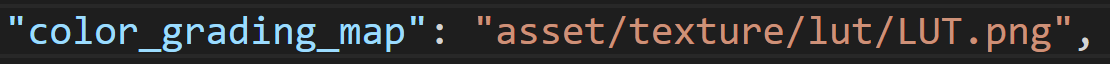
16个块，每块里是rg的2d纹理，可以直接取值（texture），所以关键是这16块蓝色的。它们不是连续的，所以得插值。

于是首先将blue拆成了整数部分和小数部分，整数部分用于索引第几块，小数部分用于前后两块插值。

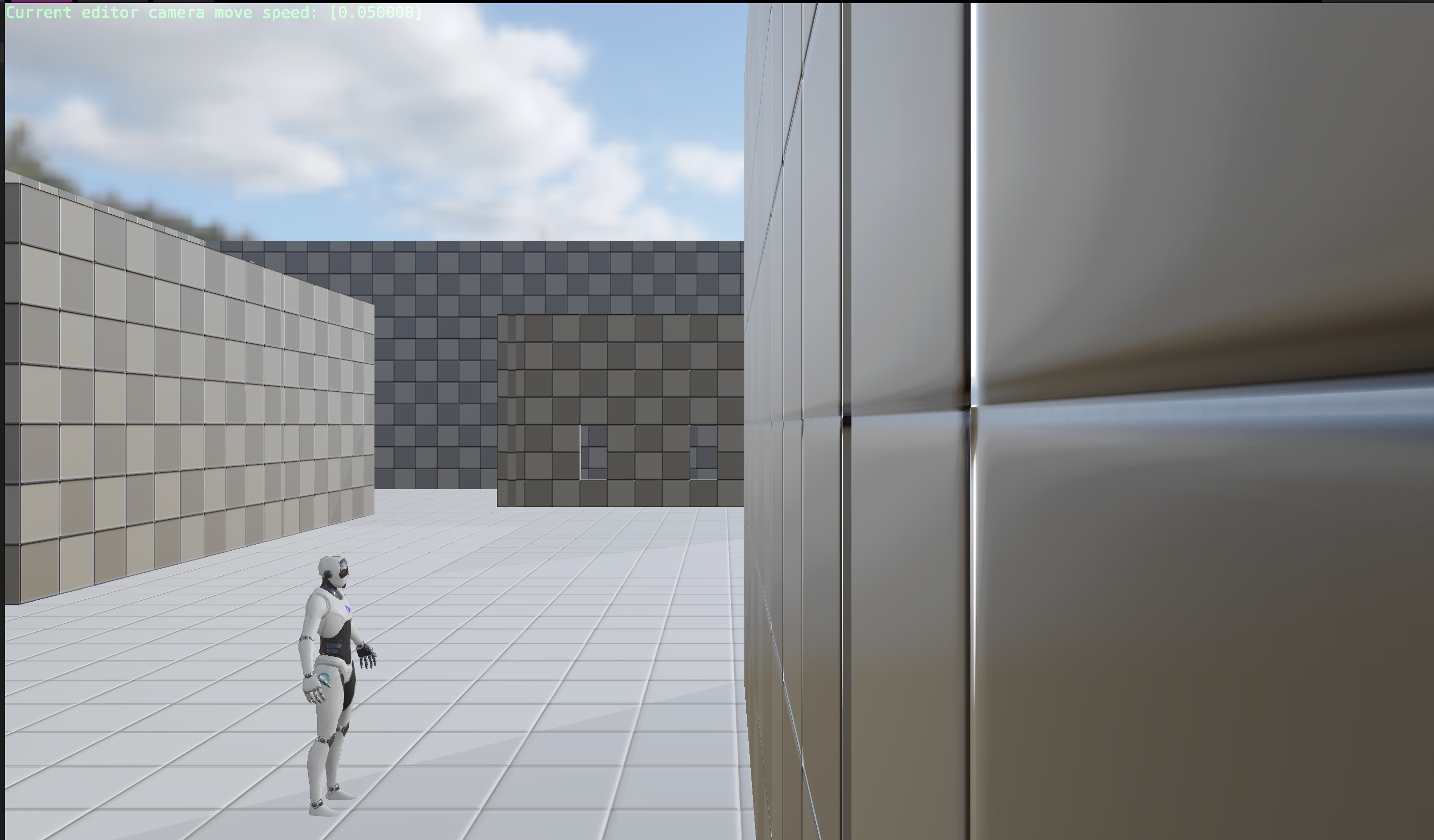
最终的uv，v就是green，u是（blue\*16之后的整数部分（0-16）+red）/16与u+（1/16）的lerp结果。

1. 基础：

这题就是替换一个lut？ps调了一个lut（根据neutral的lut来，很多教程）。



效果：



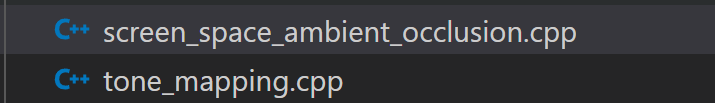
不一定好看（）

1. 提高：

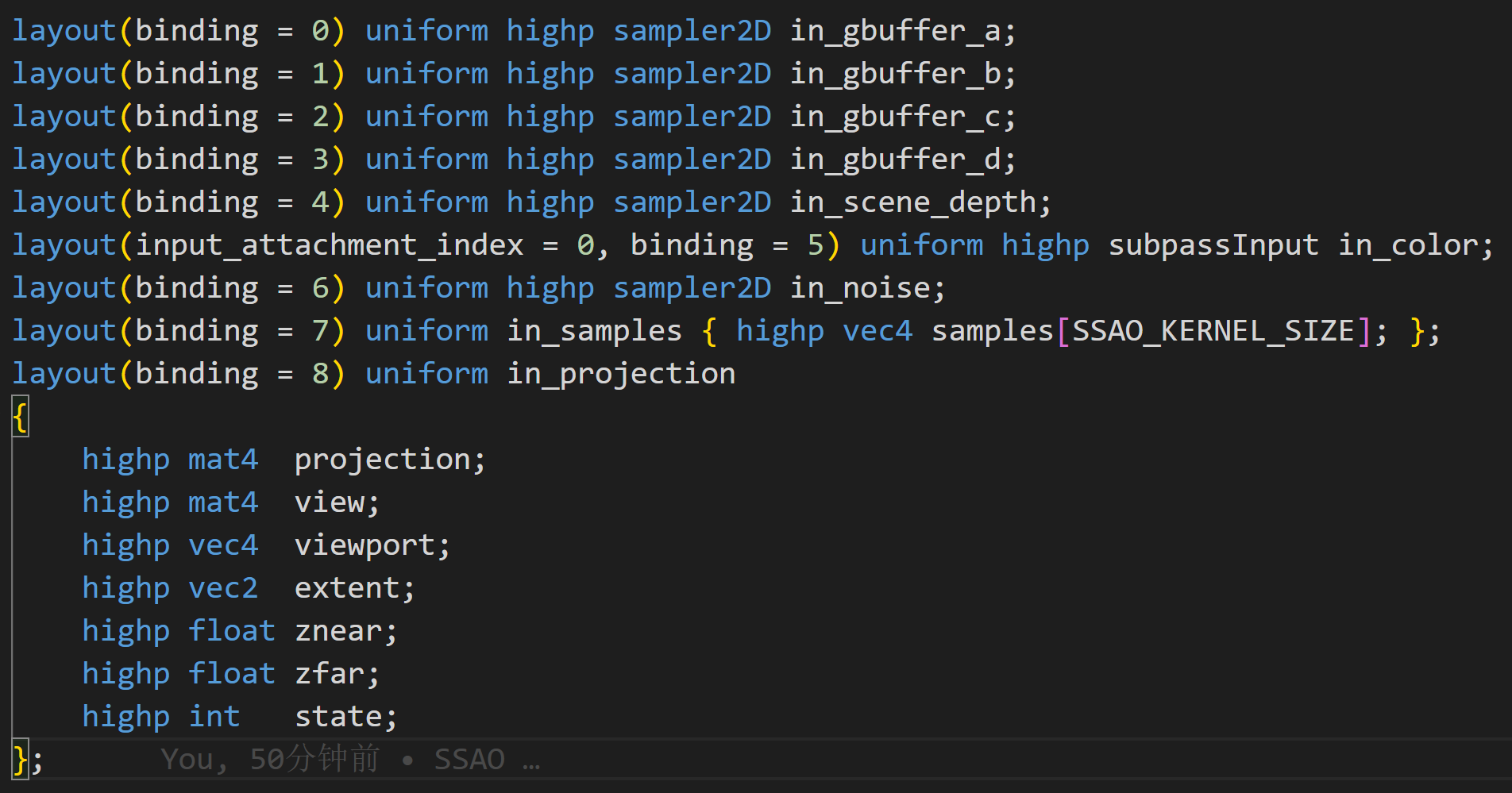
做了一个丐中丐的SSAO。但写起来还是很痛苦，无论是vulkan还是架构还是shader（）

实现思路是回溯性+模仿性的，比较偏学习性质。

1. 首先要加一个pass，专属ssao用。排在tone mapping前。又是个屏幕空间的算法，所以先模仿color\_grading的代码，加一个ssao的cpp文件，实现一个ssaopass继承自renderpassbase：

里面的代码先直接复制color\_grading的代码，然后根据需要改。

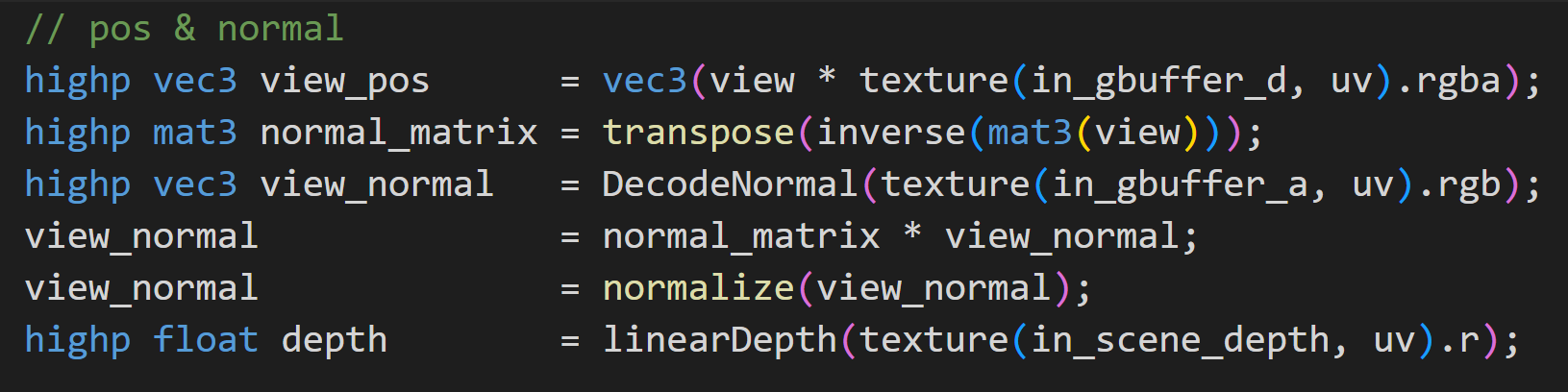
1. 在所有需要的地方都加一下，比如initialize，draw，updateAfterRecreate之类的。
2. 创建对应的fragment shader，vertex shader就用full screen就行了。然后把需要的资源列出来：



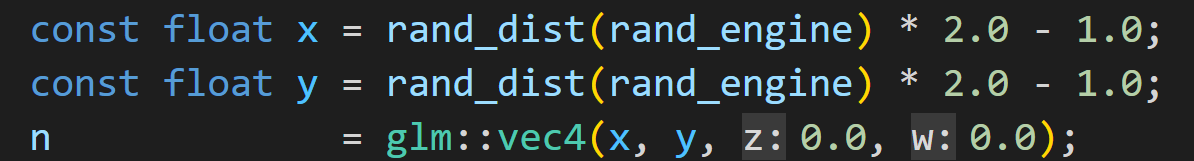
复用一下abc和depth，加了一个d存储世界坐标，加了几个噪声图、uniform的采样、以及一些信息（因为ssao要用到相机坐标的pos、normal、lineardepth，我又懒得重新搞个pass算，索性把东西全传进来都在这里计算了）。

1. 很痛苦的过程，让cpp把这里需要的资源生成并传进来，这里就不说了，因为太痛苦了，各种数据结构各种参数。前后pass的衔接、资源复用等等等等。
2. 实现shader功能，看的是OpenGL的教程。

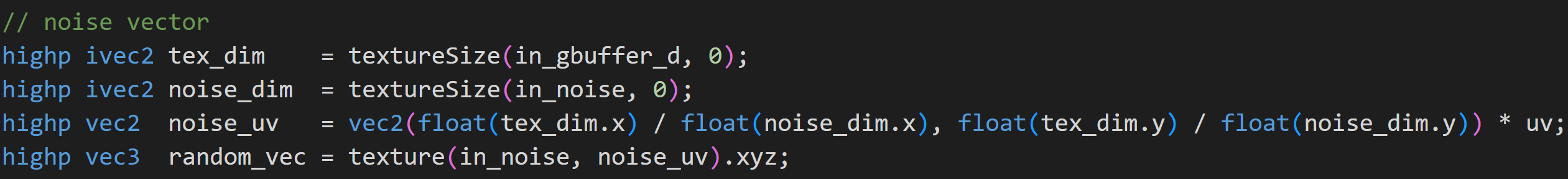
首先是计算相机空间的position和normal，以及线性的linear depth



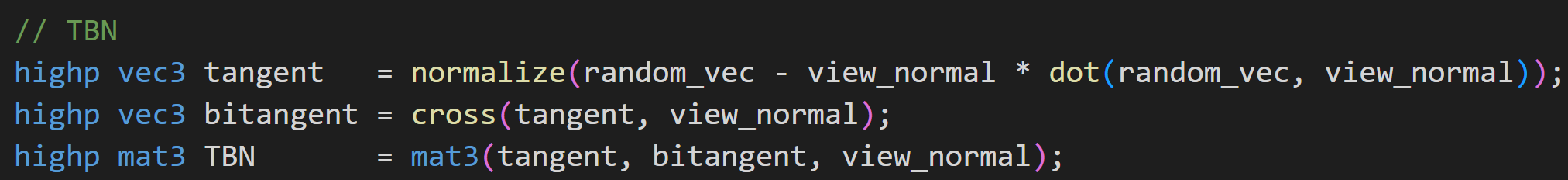
从Noise图中取提前生成的vector：



就是一个（-1----1， -1------1， 0，0）的随机vec4



根据这个vec和相机空间normal计算TBN矩阵



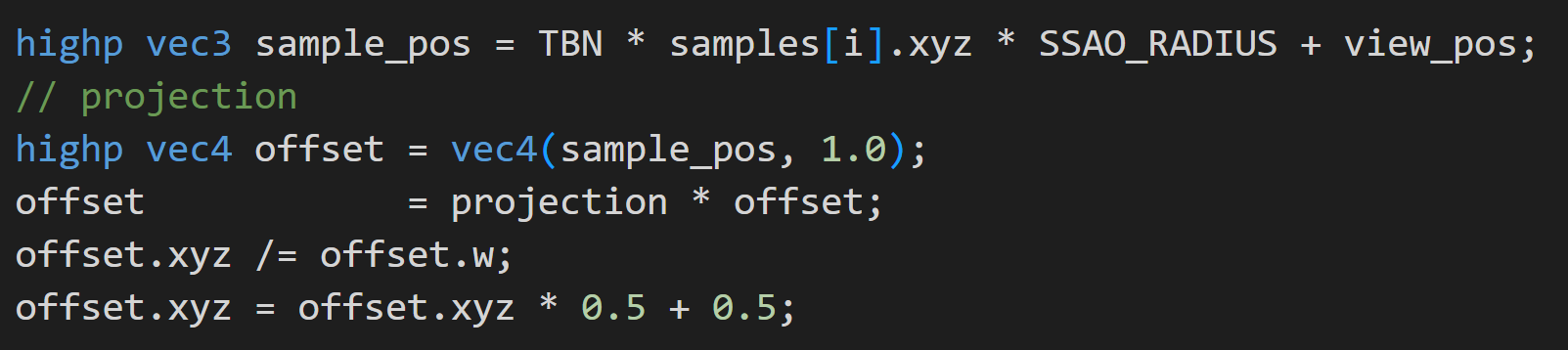
采样

得到一个衰减值，阻挡越多越黑。

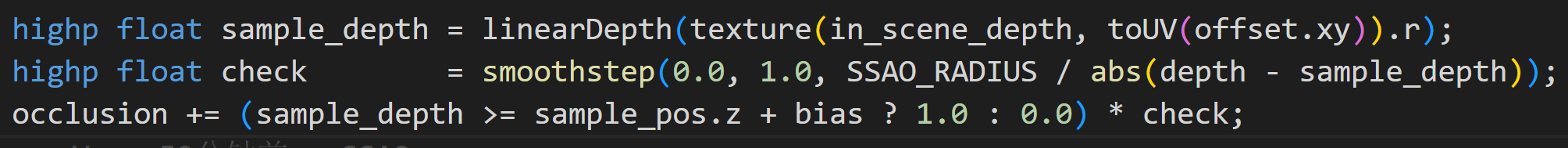


从当前相机坐标向采样点方向前进，采样点类似noise，（-1----1，-1------1，0-----1）的半球

投影到NDC坐标系

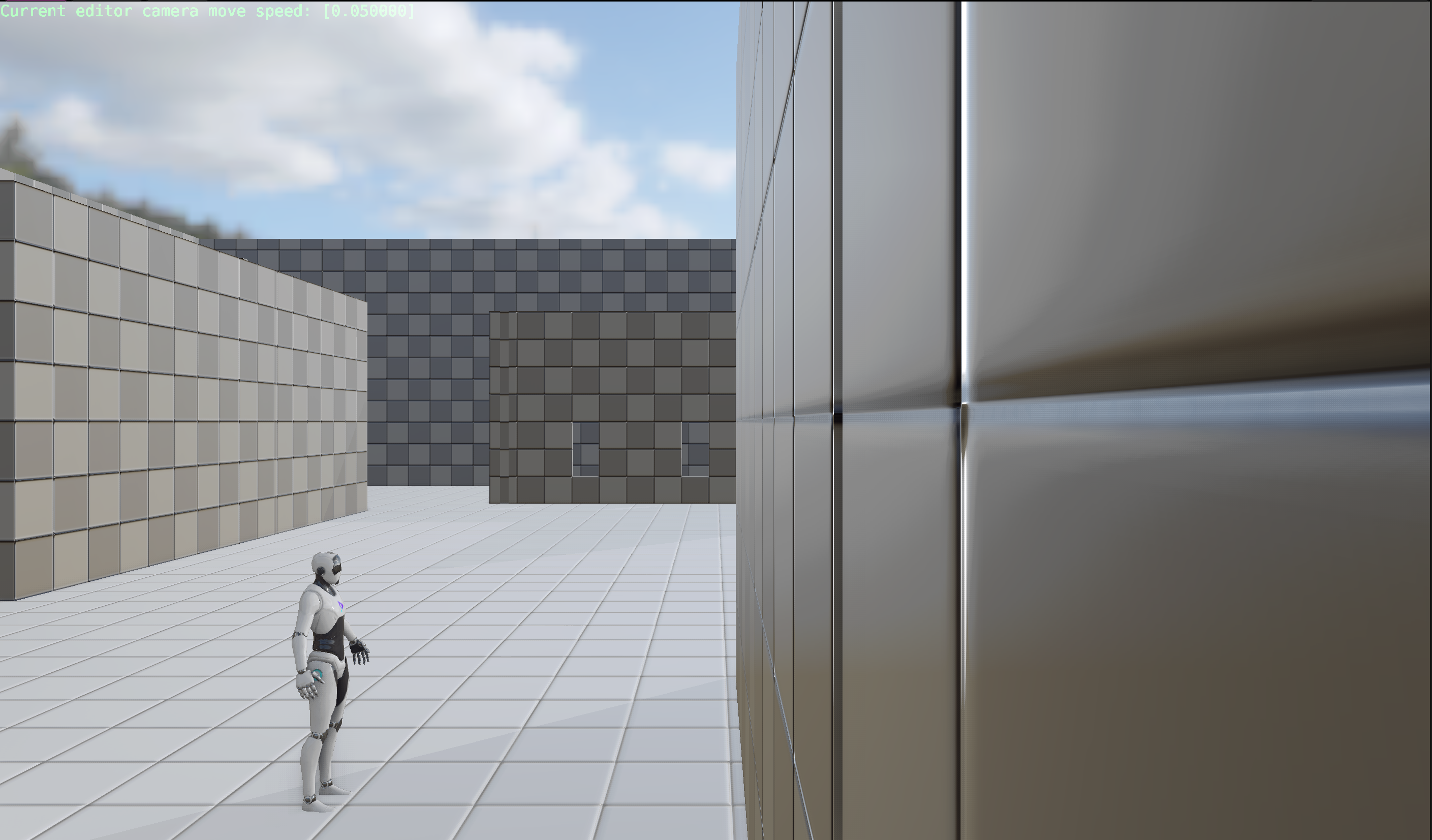
拿到采样点位置的物体的深度，与采样点的深度比较，看是否被阻挡。

Smoothstep是柔和一下过渡，并且让深度差距过大的点不产生阻挡



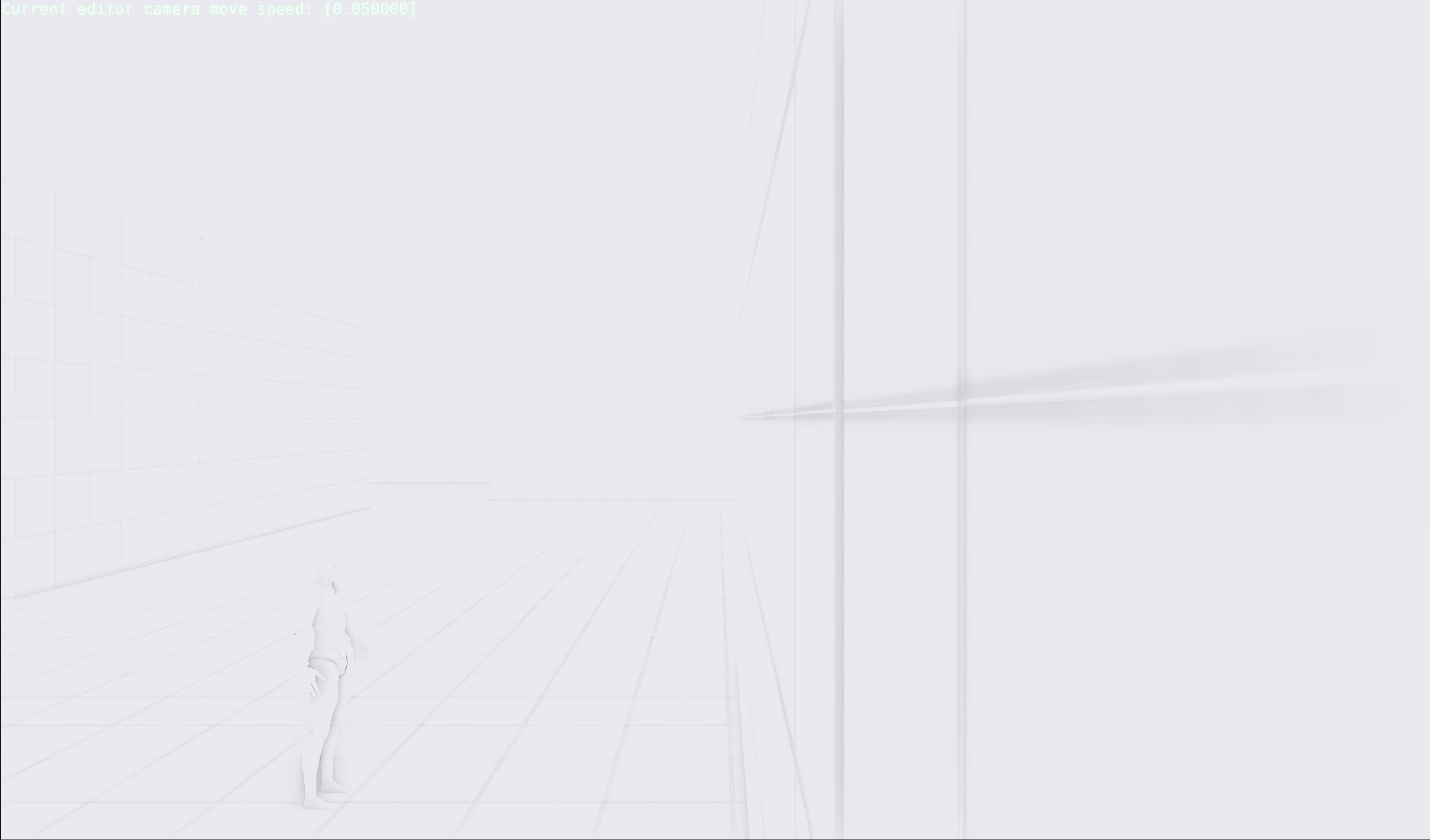
### 结果：

开启后：



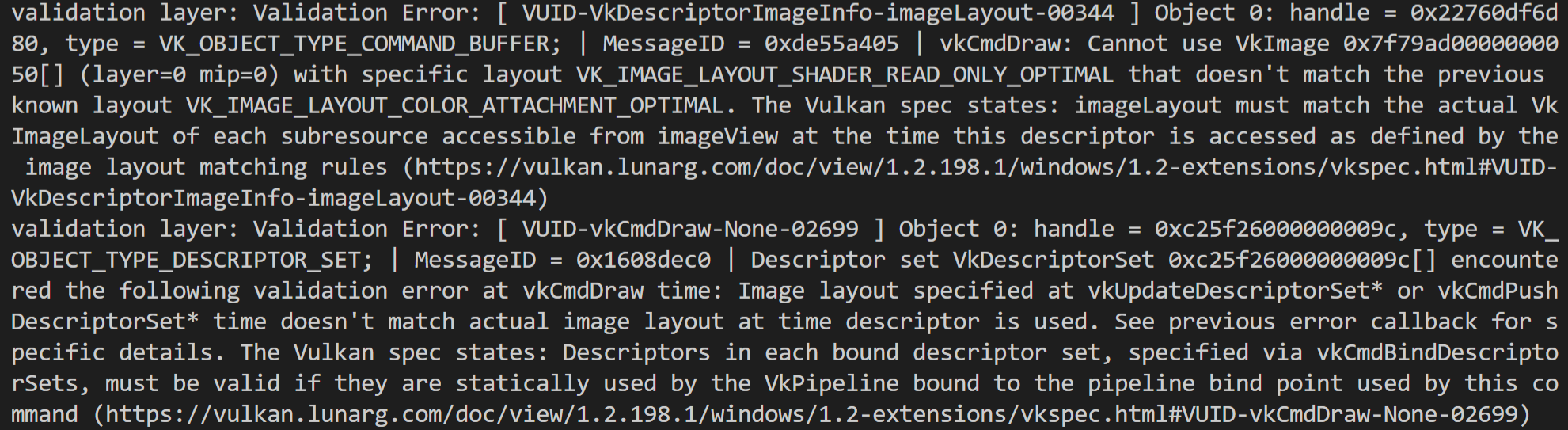
由于这么看实在是看不太出来，左上角菜单栏可以开关对比（这样就看得出来了），并且也有只显示SSAO的按钮：

only：



BUG：

1. 移动相机的时候有概率出现黑影闪烁，懒得调了，累了。
2. 更痛苦的是vulkan的imagelayout的transition，我找了很多资料以及调试了半天还是没解决，只要是debug模式，validation layer会一直提示：



由于我加进来的gbuffer\_d没自动的从VK\_IMAEG\_LAYOUT\_COLOR\_ATTACHMENT\_OPTIMAL转为VK\_IMAGE\_LAYOUT\_SHADER\_READ\_ONLY\_OPTIMAL，就一直提示这个，所以image的模式是有问题的，renderdoc里也能看出来。倒是也没影响结果就是了，所以我懒得改了。

如果老师们知道这个问题的原因和改进的方式，希望能告诉我，谢谢。这个文档开头有邮箱或者是微信的相关信息。

当然，我还试图看了vulkan configuration的synchronization层，那报错信息就更多了，什么write after write error，看着头晕，肯定和我subpass dependencies或者资源管理等代码的不严谨有关。不过我确实也不知道严谨的方式是啥。Vulkan也太折磨了。

### Last：

最后，希望项目能把precompile了的meta\_parser源码放出来，或者至少提供高版本的exe，不然整个项目的代码都只能在c++17下编译了，我尝试编译20或者23版本，结果好像meta\_parser不行，所以就只能放弃了。

以及调试的过程还是很痛苦，如果能像ue一样有一个快捷键直接触发截屏并renderdoc启动，会好受很多（）

感谢本项目相关人员的付出！