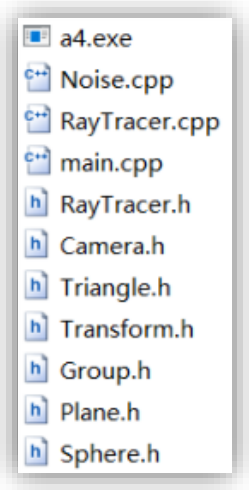
计算机图形学实习 5

一、 文件目录



这里只给出了做出修改的文件。

1. a4.exe 为 release 编译可执行文件；
2. Noise.cpp、RayTracer.cpp、main.cpp、RayTracer.h 为实习五修改的文件；
3. 其他为实习四修改的文件；
4. 未修改的文件为实习四和实习五提供的所有初始代码文件。

二、 编译环境

Windows 10 + visual studio 2010

三、 实现功能

1. 光线追踪功能。追踪阴影、反射、折射光线，根据指定的递归深度进行递归追踪。
2. PerlinNoise 噪声实现类大理石纹理。
3. 反走样：抖动采样、高斯模糊及下采样。
4. 实现处理命令参数：-shadows、-bounces、-jitter、-filter。

四、 可执行文件

编译生成的可执行文件为 a4.exe，可以直接使用题目给出的命令。

1. -shadows 参数使追踪包含阴影。
2. -bounces 参数指定递归深度。
3. -jitter 参数指定使用抖动采样。
4. -filter 参数指定使用高斯模糊和下采样。

所有执行命令：

```
a4 -input scene06_bunny_1k.txt -size 300 300 -output 6.bmp -shadows -bounces 4 -jitter -filter
a4 -input scene10_sphere.txt -size 300 300 -output 10.bmp -shadows -bounces 4 -jitter -filter
a4 -input scene11_cube.txt -size 300 300 -output 11.bmp -shadows -bounces 4 -
```

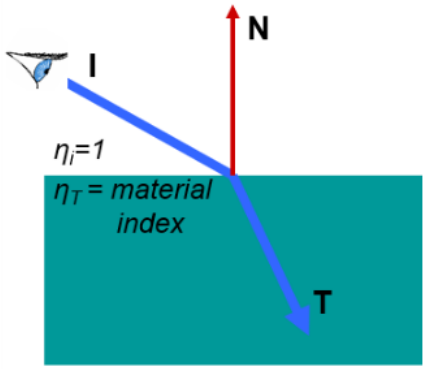
jitter -filter
a4 -input scene12_vase.txt -size 300 300 -output 12.bmp -shadows -bounces 4 -jitter -filter

五、 实现过程

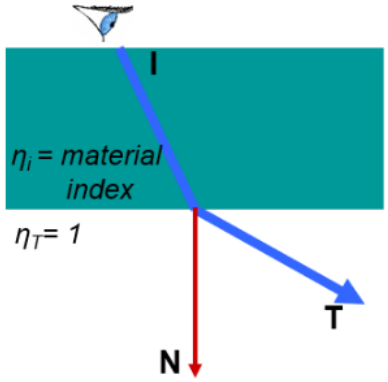
1. 处理保存命令参数。递归深度可以用 int 变量来保存，其他可以用 bool 变量来保存作为指示变量。
2. 根据场景等初始化对应的 RayTracer，Main 函数中颜色值的获取通过调用 RayTracer.traceRay()函数来获取。
3. 实现求解反射光线方向函数 mirrorDirection。公式为:

outcoming=(incoming - Vector3f::dot(incoming, normal) * 2 * normal).normalized()

4. 实现求解折射光线方向函数 transmittedDirection。该函数的功能是解决 normal*incoming 点乘小于 0 的情况，即入射光线方向和法向量夹角大于 90° 的情况，如下图 1 所示：



【图 1】入射光线与法线夹角>90°

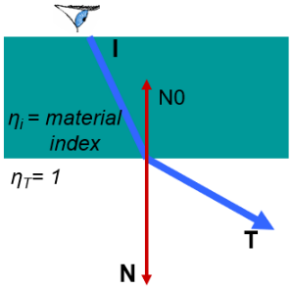


【图 2】入射光线与法线夹角<90°

当出现了图 2 所示的情况时，需要处理为图 1 的情况来解决，该函数只用来解决图 1 的情况。下述第 7 点解决了此问题。公式为：

$$t = \frac{n(d - N(d \bullet N))}{n_t} - N\sqrt{1 - \frac{n^2(1-(d \bullet N)^2)}{n_t^2}}$$

5. RayTracer::traceRay 函数添加阴影。求解所有 hit 处阴影，从 hit 处向所有光源产生阴影光，与所有物体求交，如果有 shadow_hit 则该光源对该 hit 颜色无贡献。因为命令行控制是否有阴影，构造 RayTracer 时需要保存是否加入阴影成分，在实现该功能时使用 if(shadow){//code}进行条件判断。
6. RayTracer::traceRay 函数添加反射光。调用 mirrorDirection 函数获取反射光线的方向，并构造反射 Ray 递归调用 traceRay 函数获取反射颜色值。
7. RayTracer::traceRay 函数添加折射光。如果 hit 处材料的折射系数小于 0，则没有折射情况，仅累加上述第 6 点求得的颜色值即可。如果 Hit 处材料的折射系数大于 0，则需要进行折射光的计算。首先需要解决上述第 4 点的两种问题：
   1) N*I<0，即入射光线和法线的夹角大于 90° ，该情况调用 transmittedDirection 函数可以求得折射光线位置，折射光线要进入的物体的折射率为 hit 处的折射率。
   2) N*I>0，即入射光线和法线的夹角小于 90° ，此时折射光线要进入的物体是空气，折射率取为 1.0，此时调用 transmittedDirection 函数求解折射光线时，参数 Normal 需取反为-Normal,如下图所示，N0 代替 N 传入 transmittedDirection 才可求得。



【图 3】转化求解

8. 根据公式计算折射光，递归计算折射光颜色。最后需要混合反射和折射光的颜色，使用 Schlick 对 Fresnel 方程的近似.我们会计算出对反射颜色的权重 R，而对折射颜色使用 1−R. R 由下式给定：

$$R = R_0 + (1-R_0)(1-c)^5, R_0 = (\frac{n_t - n}{n_t + n})^2, c = \begin{cases} abs(d \bullet N), n \le n_t \\ abs(t \bullet N), n > n_t \end{cases}$$

如果有折射光，则 color += R * reflectionColor + (1 - R)*refractionColor。
如果计算没有折射光（即全反射），则 color += reflectionColor。
递归时递归深度+1，递归的折射系数为折射进入的物体的折射系数。注意达到最大递归深度后递归该函数。

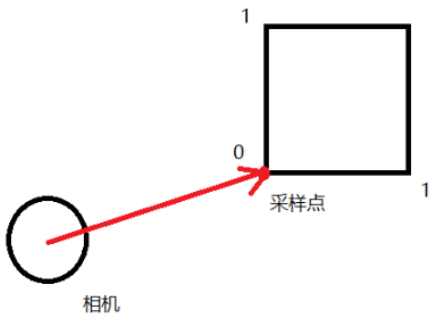
实现perlin噪声。Perlin噪声实现过程纹理在Noise::getColor函数中，先计算

M(x, y, z)= sin(ωx + aN(x, y, z))

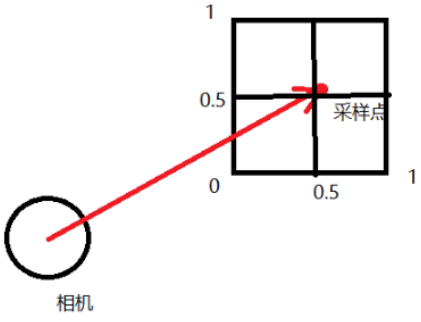
然后再针对两个颜色进行线性插值。M值范围为[-1,1],那么 N=(M+1)/2 值范围为[0,1]，使用N进行插值，噪声产生的新颜色为：

N * color[0] + (1 - N)*color[1]

9. 实现抖动采样。首先是分辨率扩大三倍，weight 和 height 对应乘以 3 即可。其次，由于我使用的是每个像素的左下角点产生相机投射的光线，因此和实习要求中的抖动略有差别。实习要求中以像素中心点采样代替像素，抖动范围为[-0.5,0.5]；我是以像素左下角采样代替像素，抖动范围为[0,1]。如下图所示：



1) 左下角抖动范围为[0,1]



2) 中心采样点抖动范围为[-0.5,0.5]

10. 实现高斯模糊和下采样。这一块需要将之前计算的进行保存，然后根据高斯模糊的公式进行计算：

$$I'(i,j) = I(i,j-2)K(0) + I(i,j-1)K(1) + I(i,j)K(2) + I(i,j+1)K(3) + I'(i,j+2)K(4).$$

针对数组计算防止越界，针对 j 的减法操作如果越界则以 0 代替，针对 j 的加法操作则与 height 取余，代码如下：

```
gaussian_pixels[i][j] =
    0.1201f*sub_pixels[i][(j - 2) > 0 ? j - 2 : 0] +
    0.2339f*sub_pixels[i][(j - 1) > 0 ? j - 1 : 0] +
    0.2931f*sub_pixels[i][j]+
    0.2339f*sub_pixels[i][(j + 1) % height] +
     0.1201*sub_pixels[i][(j + 2) % height];
```

下采样则计算单个像素的 9 个网格方格的均值即可。

六、 问题与解决

1. 计算折射光线的问题需要注意的两种情况，如上面说明的，这个纠结了很久，因为代码不好调试，分析过程较长。
2. 抖动采样的问题。我之前是采样像素的左下角的点，和实现要求的像素的中心点有别。这个我没有改动，针对性地修改了抖动区间。
3. 浮点数计算精度问题。相机产生光线需要针对浮点数计算，可能产生误差，之前我将 image 视为单位 1 weight 和 height，产生的图片有一些“异样线条”，这是因为有些光线未产生，造成与物体没有求交。我将 1.0 变为 1.00000001 即解决问题。浮点数误差没有完美解决，这是针对遇到的问题进行了解决。

七、 实现效果

实现效果在下几页进行展示。

作者：AnDJ

【0 次反射】

【1 次反射】

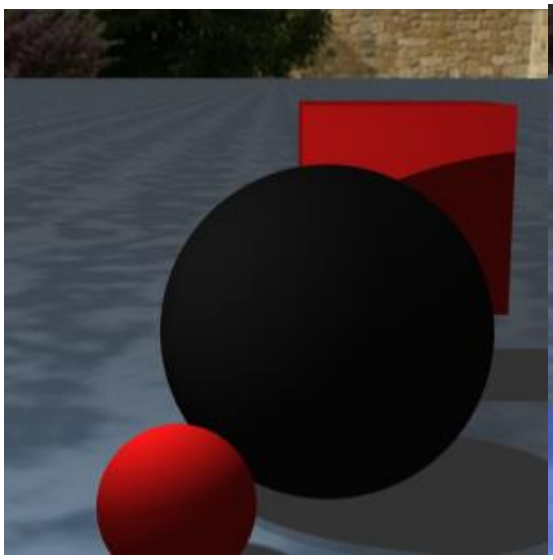
【2 次反射】

【3 次反射】

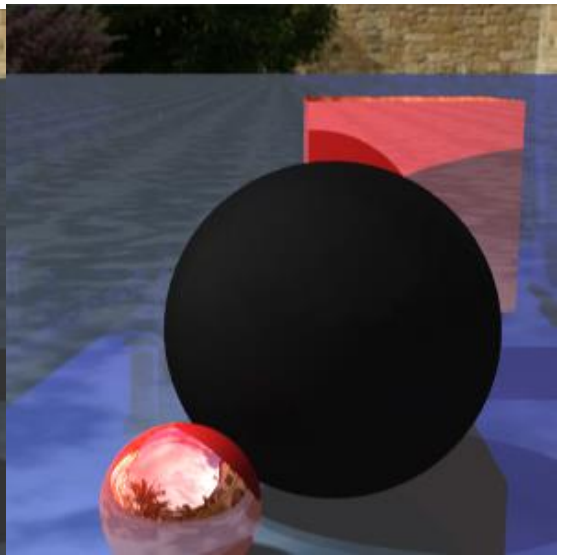
【4 次反射】

【无反走样】

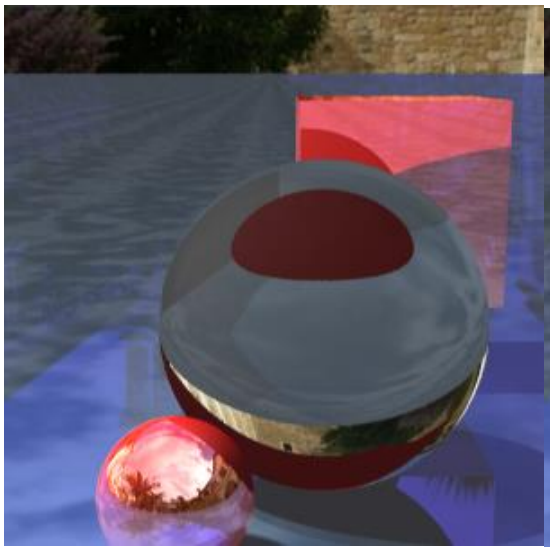
【无阴影】



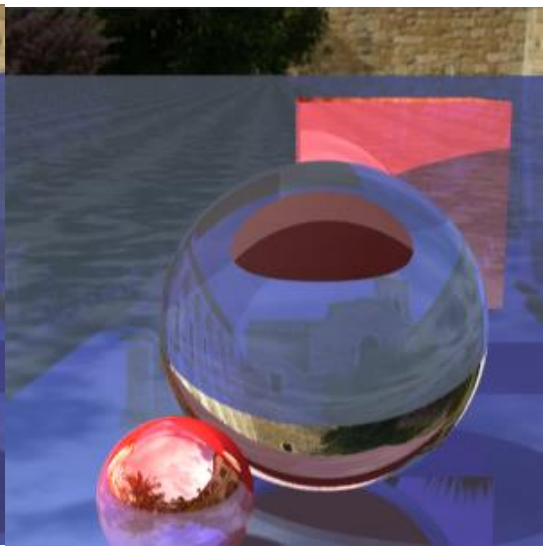
【0 次反射】



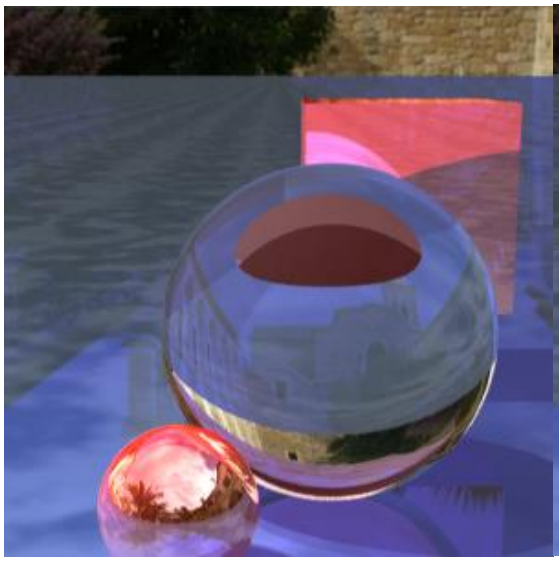
【1 次反射】



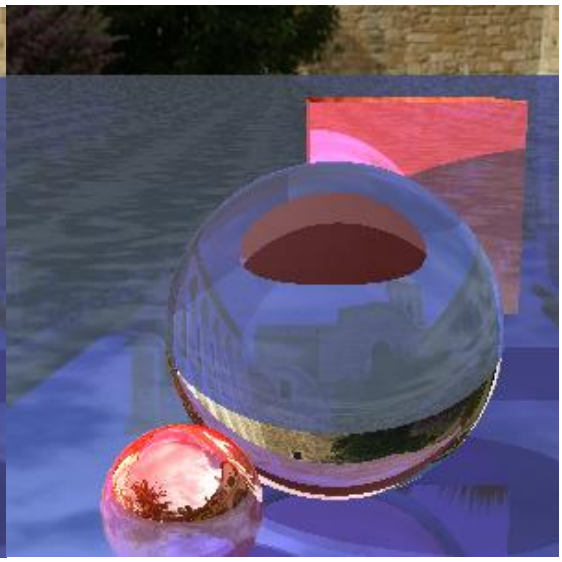
【2 次反射】



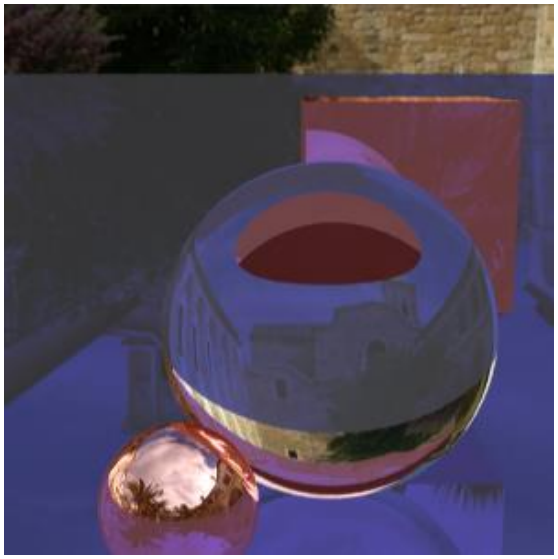
【3 次反射】



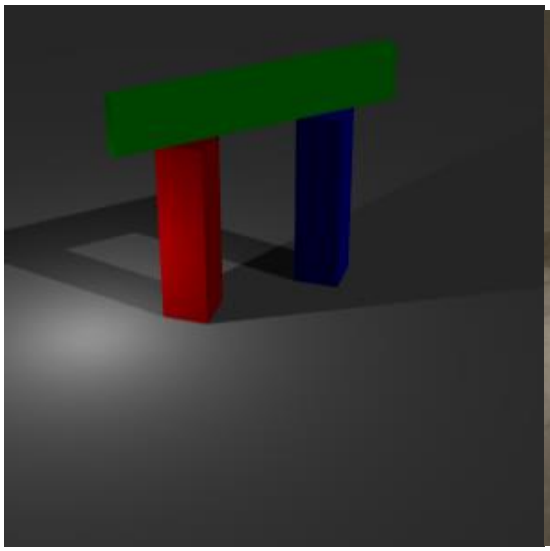
【4 次反射】



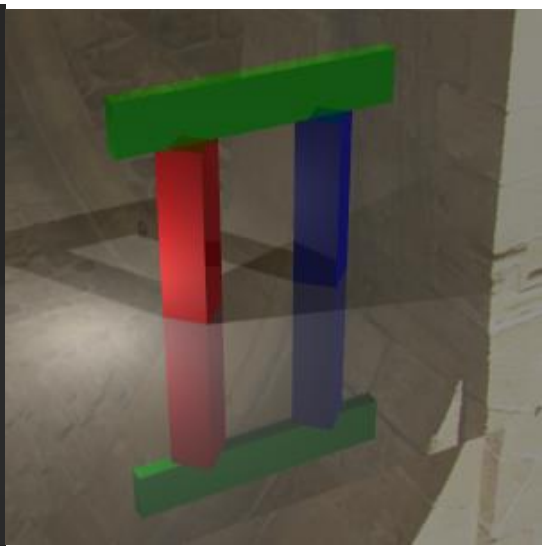
【无反走样】



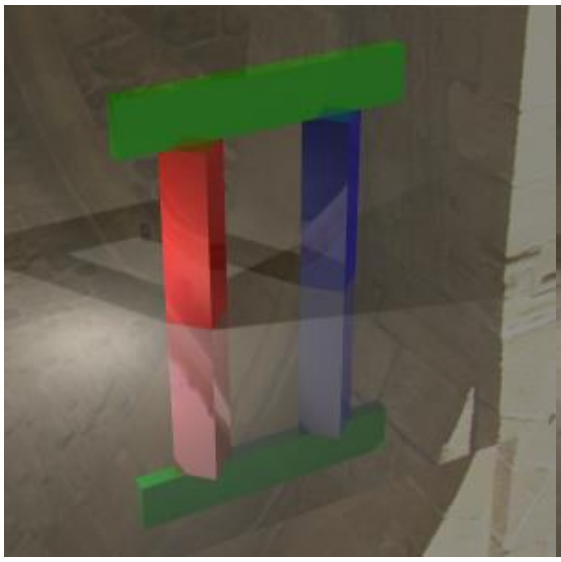
【无阴影】



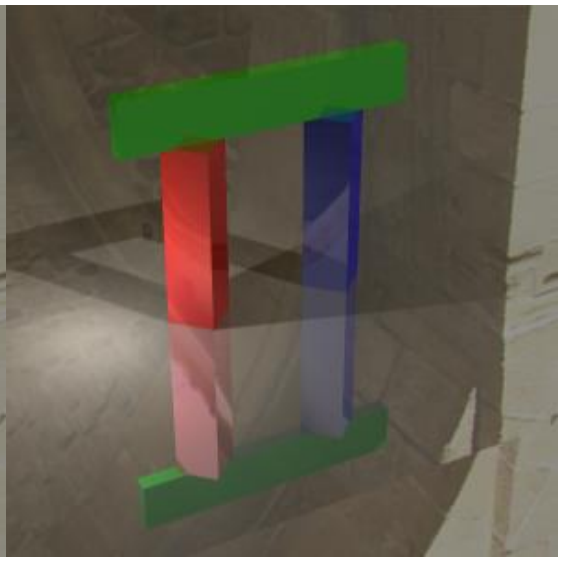
【0 次反射】



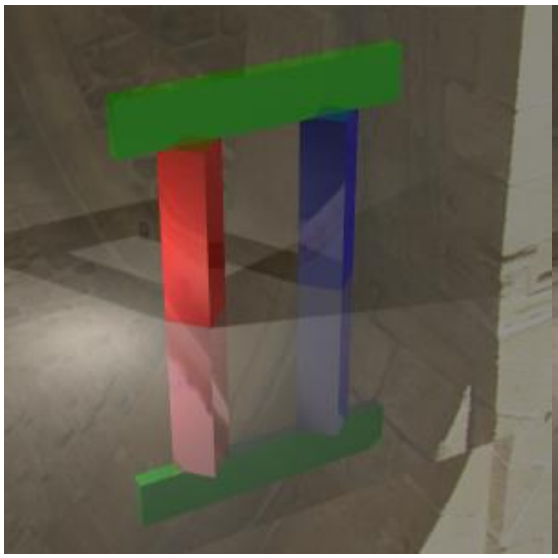
【1 次反射



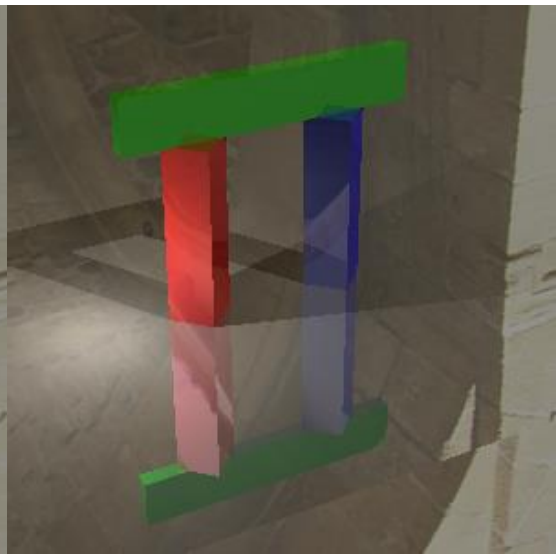
【2 次反射】



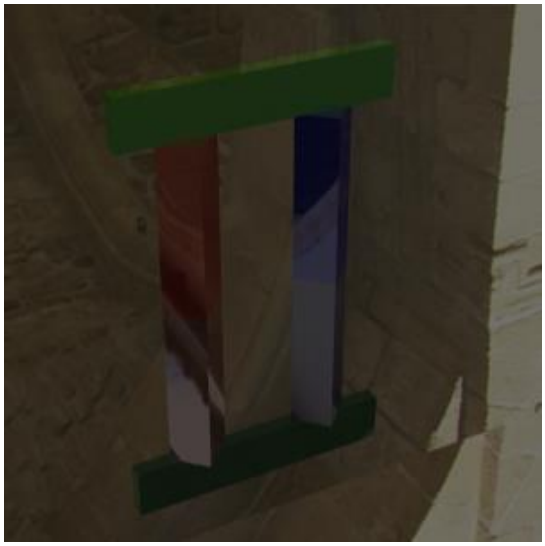
【3 次反射】



【4 次反射】



【无反走样】



【无阴影】

【0 次反射】

【1 次反射】

【2 次反射】

【3 次反射】

【4 次反射】