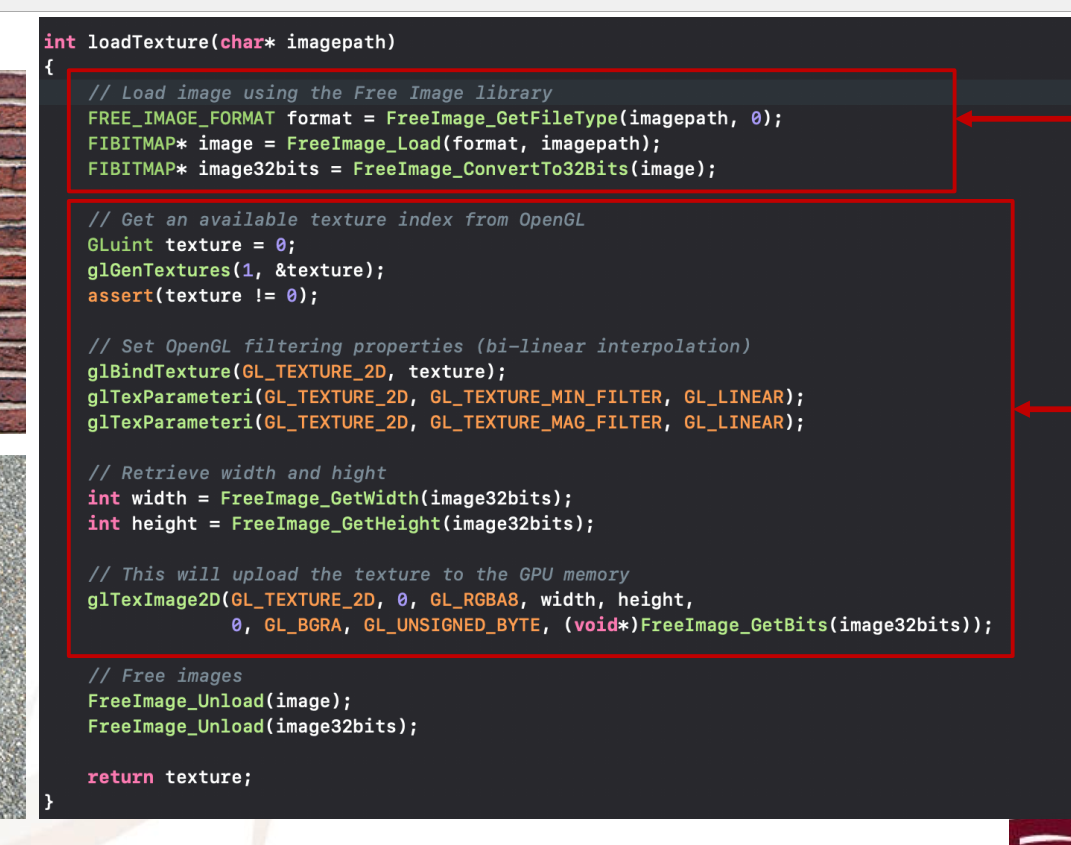
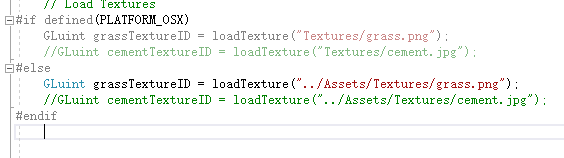
loadTexture:读取实际图片并转化成对应的texture,

opengl生成一个unique texture id

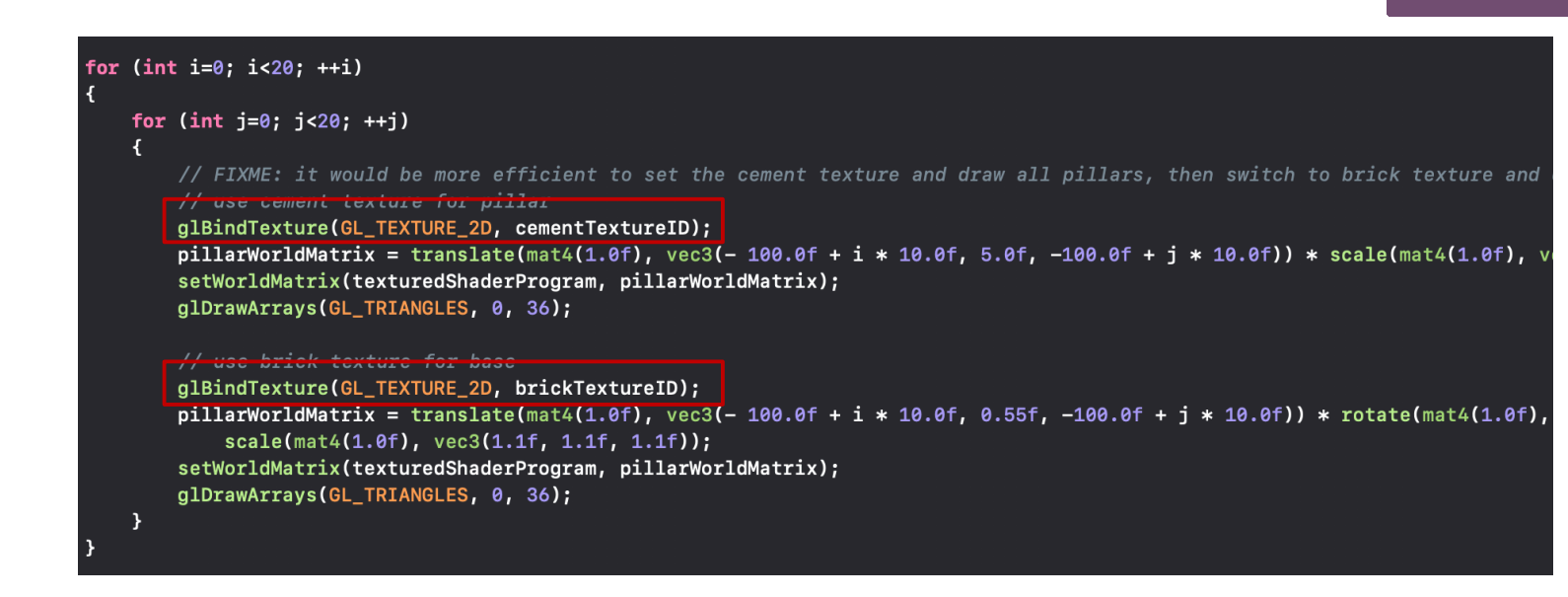
绑定这两个texture并存入虚拟内存中



最后return对这个材质的编号



这个cementTextureID就是Load image的id

利用texture作画  




怎么样不使用textureShader,而是民工法

s

顺序 草地贴图

2.圆柱体//阅读lab05先

3.轮胎贴图

4.阅读phong

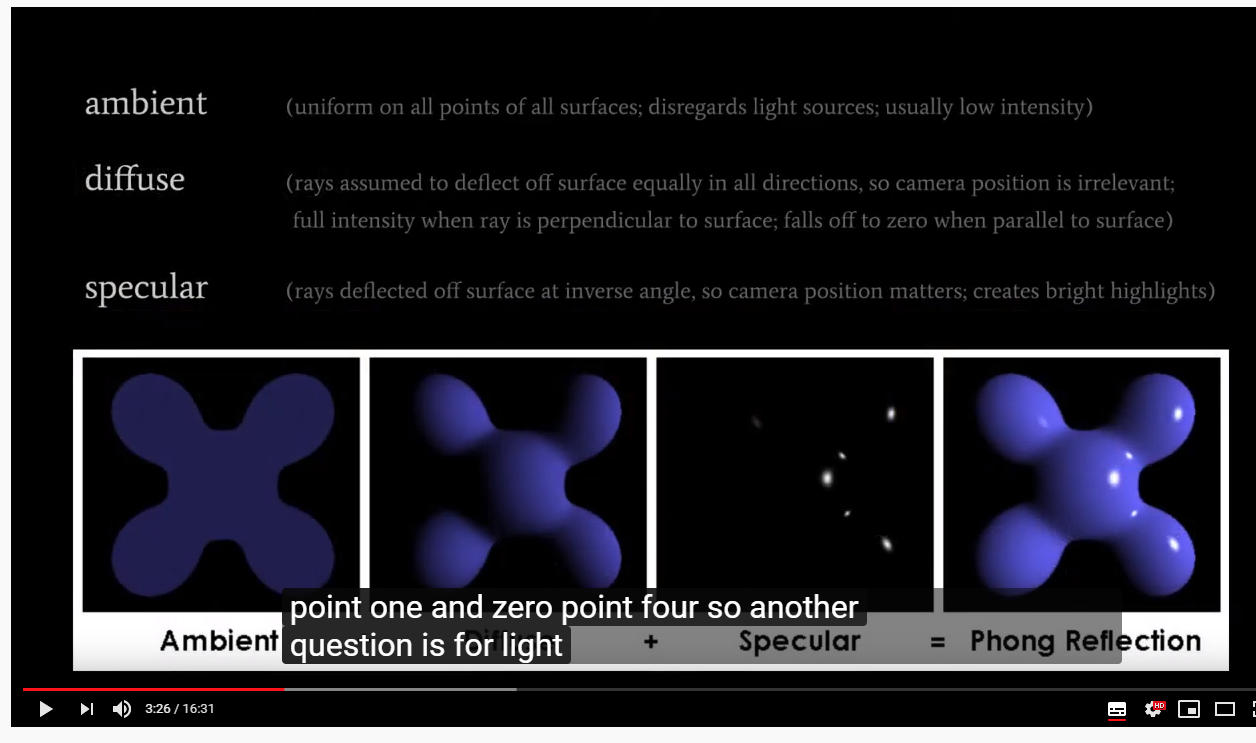
5.增加点光源 //不需要按键，所以应该应用到所有vertexShader

6.金属材质

7.使用双重处理法增加阴影

//基本都好了，问题在于光源没有随着变形

油管点光源材质



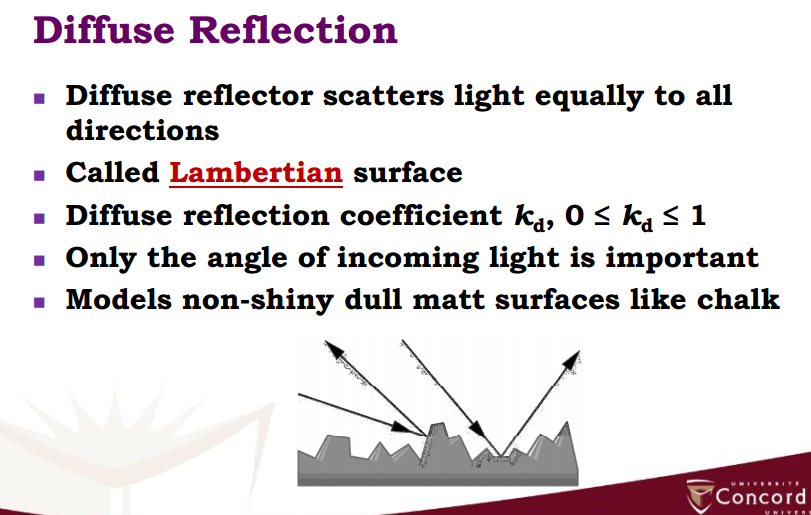
ambient，对每个点进行等强照射，(uniform)，决定了环境光的反射部分，因此没有一个地方是完全暗下去的//涂底色

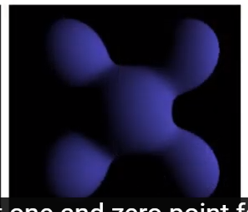
ambient light假设环境光就像水,空气一样，每个地方都碰得到//是一种环境

与光源位置距离啥的都无关，是一种environment



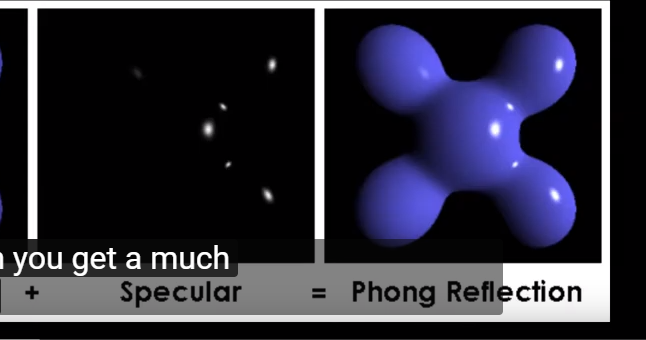
diffuse light



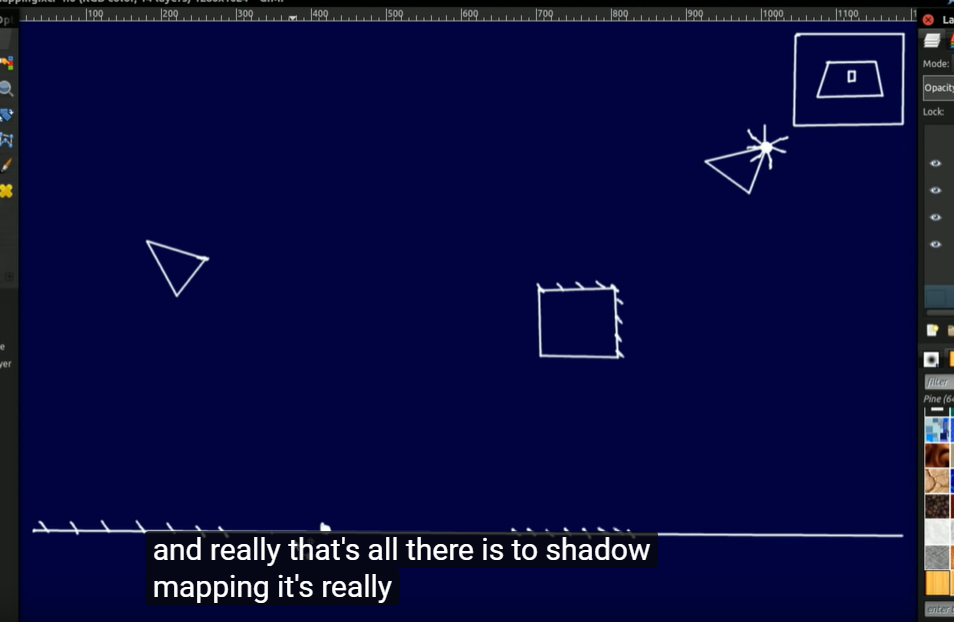


散射，入射光的角度很重要，假设完全平行，那就是0，暗，假设正好垂直，就是1 （light full intensity，pure white 111）,在垂直与平行之间的角度，0到1之间， diffuse light与camera位置无关

specular，与摄像机有关



为什么金属部分有的地方亮，因为光源照到然后正好反射进你的眼睛里



\*（所有可以看见的点）

从太阳的角度大概是这样一张图，然后我们记录下所有的depth，

然后如果我们所测试的值到太阳的距离，大于depth里记载的点，自然就在阴影中

shader

VERTEX//perfect

FRAGMENT//perfect

SIMPLEVERTEX//perfect

SIMPLEFRAGMENTSHADER//perfect

TEXTUREVERTEX //reverse Normal //不行就让texture部分超级大退化

TEXTUREFRAGMENT

TODO1 PERFECT

TODO2 near plane/radians //plane 要改 Radians都要改

给VBO加360位

然后用for loop填充新的点，然后draw GLquadstrip