









词汇表

* **OpenGL**： 一个定义了函数布局和输出的图形API的正式规范。
* **GLAD**： 一个拓展加载库，用来为我们加载并设定所有OpenGL函数指针，从而让我们能够使用所有（现代）OpenGL函数。
* **视口(Viewport)**： 我们需要渲染的窗口。
* **图形管线(Graphics Pipeline)**： 一个顶点在呈现为像素之前经过的全部过程。
* **着色器(Shader)**： 一个运行在显卡上的小型程序。很多阶段的图形管道都可以使用自定义的着色器来代替原有的功能。
* **标准化设备坐标(Normalized Device Coordinates, NDC)**： 顶点在通过在剪裁坐标系中剪裁与透视除法后最终呈现在的坐标系。所有位置在NDC下-1.0到1.0的顶点将不会被丢弃并且可见。
* **顶点缓冲对象(Vertex Buffer Object)**： 一个调用显存并存储所有顶点数据供显卡使用的缓冲对象。
* **顶点数组对象(Vertex Array Object)**： 存储缓冲区和顶点属性状态。
* **索引缓冲对象(Element Buffer Object)**： 一个存储索引供索引化绘制使用的缓冲对象。
* **Uniform**： 一个特殊类型的GLSL变量。它是全局的（在一个着色器程序中每一个着色器都能够访问uniform变量），并且只需要被设定一次。
* **纹理(Texture)**： 一种包裹着物体的特殊类型图像，给物体精细的视觉效果。
* **纹理缠绕(Texture Wrapping)**： 定义了一种当纹理顶点超出范围(0, 1)时指定OpenGL如何采样纹理的模式。
* **纹理过滤(Texture Filtering)**： 定义了一种当有多种纹素选择时指定OpenGL如何采样纹理的模式。这通常在纹理被放大情况下发生。
* **多级渐远纹理(Mipmaps)**： 被存储的材质的一些缩小版本，根据距观察者的距离会使用材质的合适大小。
* **stb\_image.h**： 图像加载库。
* **纹理单元(Texture Units)**： 通过绑定纹理到不同纹理单元从而允许多个纹理在同一对象上渲染。
* **向量(Vector)**： 一个定义了在空间中方向和/或位置的数学实体。
* **矩阵(Matrix)**： 一个矩形阵列的数学表达式。
* **GLM**： 一个为OpenGL打造的数学库。
* **局部空间(Local Space)**： 一个物体的初始空间。所有的坐标都是相对于物体的原点的。
* **世界空间(World Space)**： 所有的坐标都相对于全局原点。
* **观察空间(View Space)**： 所有的坐标都是从摄像机的视角观察的。
* **裁剪空间(Clip Space)**： 所有的坐标都是从摄像机视角观察的，但是该空间应用了投影。这个空间应该是一个顶点坐标最终的空间，作为顶点着色器的输出。OpenGL负责处理剩下的事情（裁剪/透视除法）。
* **屏幕空间(Screen Space)**： 所有的坐标都由屏幕视角来观察。坐标的范围是从0到屏幕的宽/高。
* **LookAt矩阵**： 一种特殊类型的观察矩阵，它创建了一个坐标系，其中所有坐标都根据从一个位置正在观察目标的用户旋转或者平移。
* **欧拉角(Euler Angles)**： 被定义为偏航角(Yaw)，俯仰角(Pitch)，和滚转角(Roll)从而允许我们通过这三个值构造任何3D方向。