

目录

- 顶点着色器 --Vertex Shaders
- 片段着色器 --Fragment shaders
- 从顶点到像素点 --Vertexes to pixels
- 顶点着色器 --Vertex Shaders
- 使用Ps或C4D时OpenGL做什么？
- 总结

顶点着色器 --Vertex Shader

顶点着色器Vertex Shader 允许你一次修改一个顶点（一个顶点就是一个三角形的一个角，记住在计算机中所有的图形都是由三角形组成的）  
在C4D中建模和贴图过程就是在调整顶点着色器的过程。

片段着色器 --Fragment shaders

片段着色器Fragment shaders 允许你一次修改一个像素点的颜色（以及纹理贴图的坐标位置）。

这些听起来都很复杂，不过别被吓跑。

着色器Shader 听起来非常神秘和困难，但是实际上它们并非那样的。

片段着色器其实就是控制图片中每一个像素点，就像你在画一幅像素画。

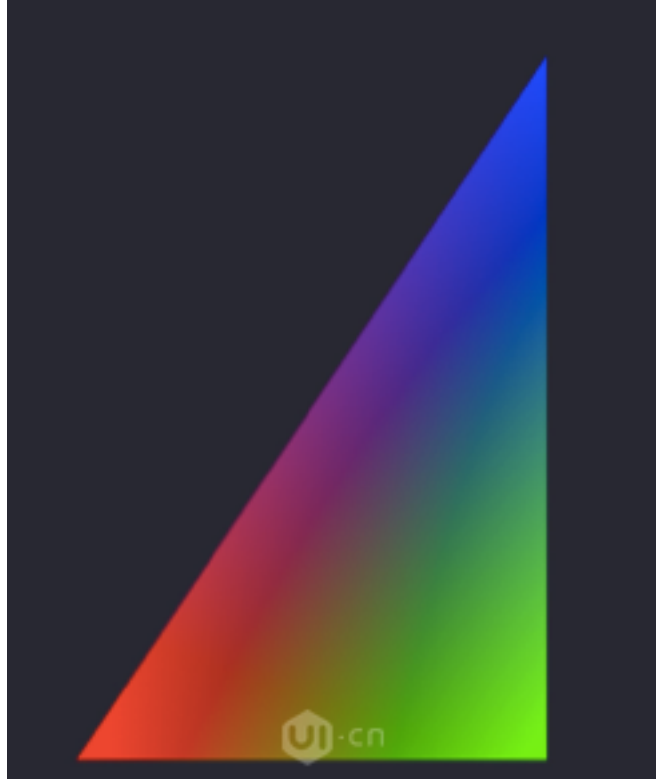


## 从顶点到像素点 --Vertexes to pixels

在制作动画片时，熟练的艺术家不会画出每一个单独帧。他们绘制关键帧，然后让其他人（或者现在是计算机）去填充位于关键帧之间的中间帧，这个处理被称为 tweening。

类似地，在 2D 和 3D 图形处理中，我们必须指出我们所画的位置和颜色，不过我们只需要对一些点的样本集合（类似于关键帧）进行操作，这些点被称为 vertex（顶点）。实际上，我们创造了一个线框模型。

OpenGL 接着会添加插入这些样本点之间的所有点。具体的方法是：把这些点组成三角形-因为这是最简单的形状，因此它用三个角的顶点值来计算三角形里所有点的值。



就像上图一样。看看红色角、绿色角和蓝色角的颜色是如何在三角形内部混合起来的。它确实很简单。

OpenGL 需要知道每个顶点的  $x, y, z$  位置坐标，以及它的颜色 - 或者，假如你把一个纹理图形 粘贴在线框模型上时，图形的哪一部分会被绘制在这个顶点上。

所以，每个顶点都有三条关键信息：

- 顶点的  $x, y, z$  位置坐标
- 颜色（如果你设置过）
- 纹理映射（例如纹理贴图上的哪一个  $x, y$  点被用于这个顶点）

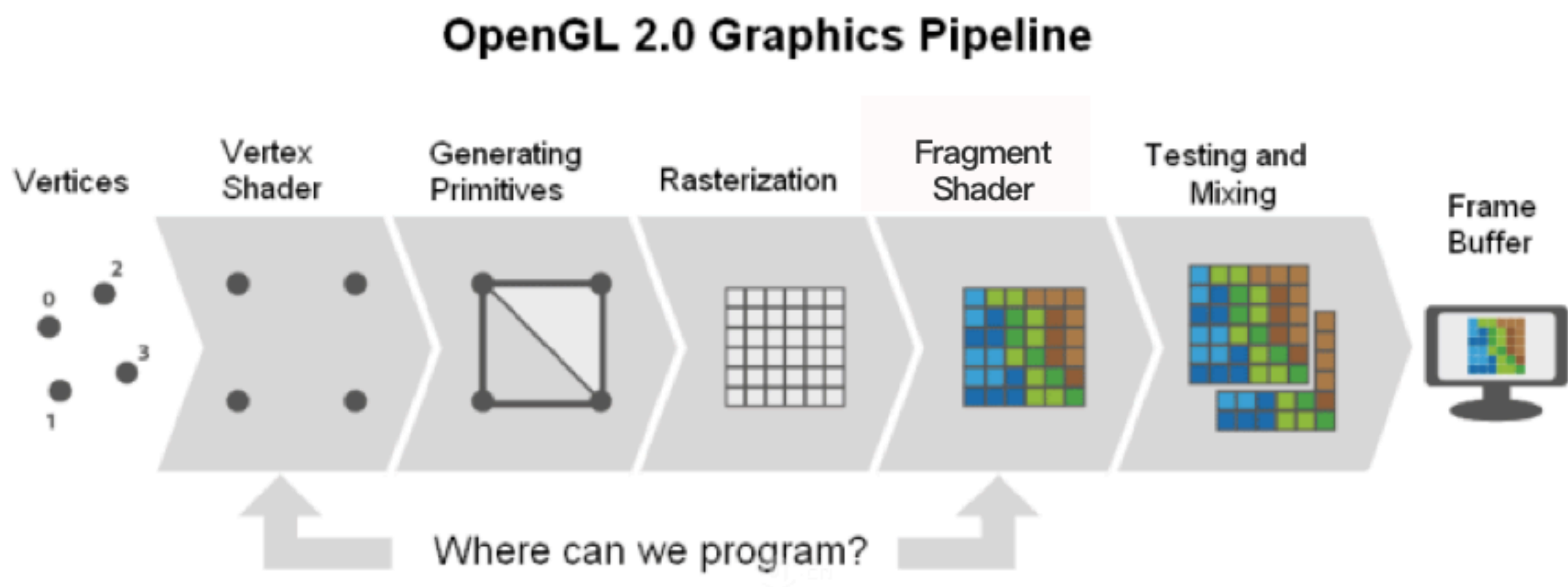
OpenGL 做了其他一大堆非常复杂、名字很长的事情，当然，我们所关注的仅仅是我们所提供的顶点集合的信息，以及 OpenGL 在屏幕上向这些顶点中插入的像素点和像素点的颜色。

## 使用Ps或C4D时OpenGL做什么？

OpenGL 就像一个长长的管道。你从这一头把你的绘图指令放进去，像素点会从另一头出来，这些像素点会在你的屏幕上形成一幅 3D 图像。在管道内部是一系列复杂处理，我们不会假装理解。

因此让我们聚焦到管道外。

在管道上有两个位置被挖了两个洞（Vertex Shader 和 Fragment Shader），因此你可以通过这两个洞看到管道里流过的信息，并且你还可以进到里面去修改这些信息，这些信息处于整个体系的最底层，下图是细节：





我们可以理解：

- Photoshop用非常友好的UI界面，在Fragment Shader阶段实施大量像素算法集合的工具
- C4D用非常友好的UI界面，在Vertex Shader阶段实施建模，在Fragment Shader界面把3D空间在某个摄像机角度转换成2D像素集合的工具

假设设计师踩油门（ Photoshop ）开车，在此过程中会有发动机（ Fragment Shader ）、轮胎（ Vertex Shader ）等等汽车部件在为你在高速路（ 计算机 ）中以200km/h的速度猛飙（ 创意 ）。

## 总结

如果你看到这里那么恭喜你，你已经了解到计算机图形学的冰山一角。而这冰山一角对一名优秀的设计师已经足够了解之前没有涉足过的领域。

当然设计师最重要的还是创意本身！

如果让工程师做类似下面Gif图像扭曲的效果时，99.99%的工程师会拒绝。这时候鄙视的目光加上不屑的语气甩他一句“这效果用GLSL Fragment Shader做啊，Android和iOS都支持的，你难道不会吗？” :)



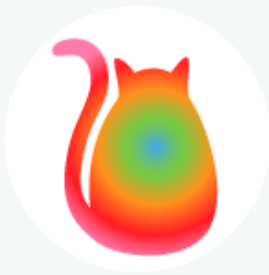
像素 ps C4D photoshop CINEMA4D OpenGL



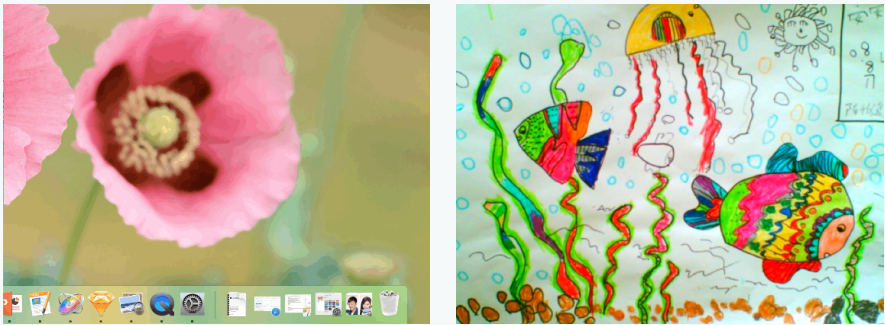
快给朋友分享吧！



0人已收藏



夏尔猫叔   
User Centered Design, I'm Player.  
26粉丝/6关注



还可以输入500个字

表情

Ctrl+Enter

评论

