

OpenGL ES进阶教程7-天空盒效果



作者 落影loyinglin (/u/815d10a4bdce) [+ 关注](#)

2016.05.08 10:56 字数 1662 阅读 2210 评论 13 喜欢 15 赞赏 1 阅读 2210 评论 13 喜欢 15 赞赏 1 (/u/815d10a4bdce)

教程

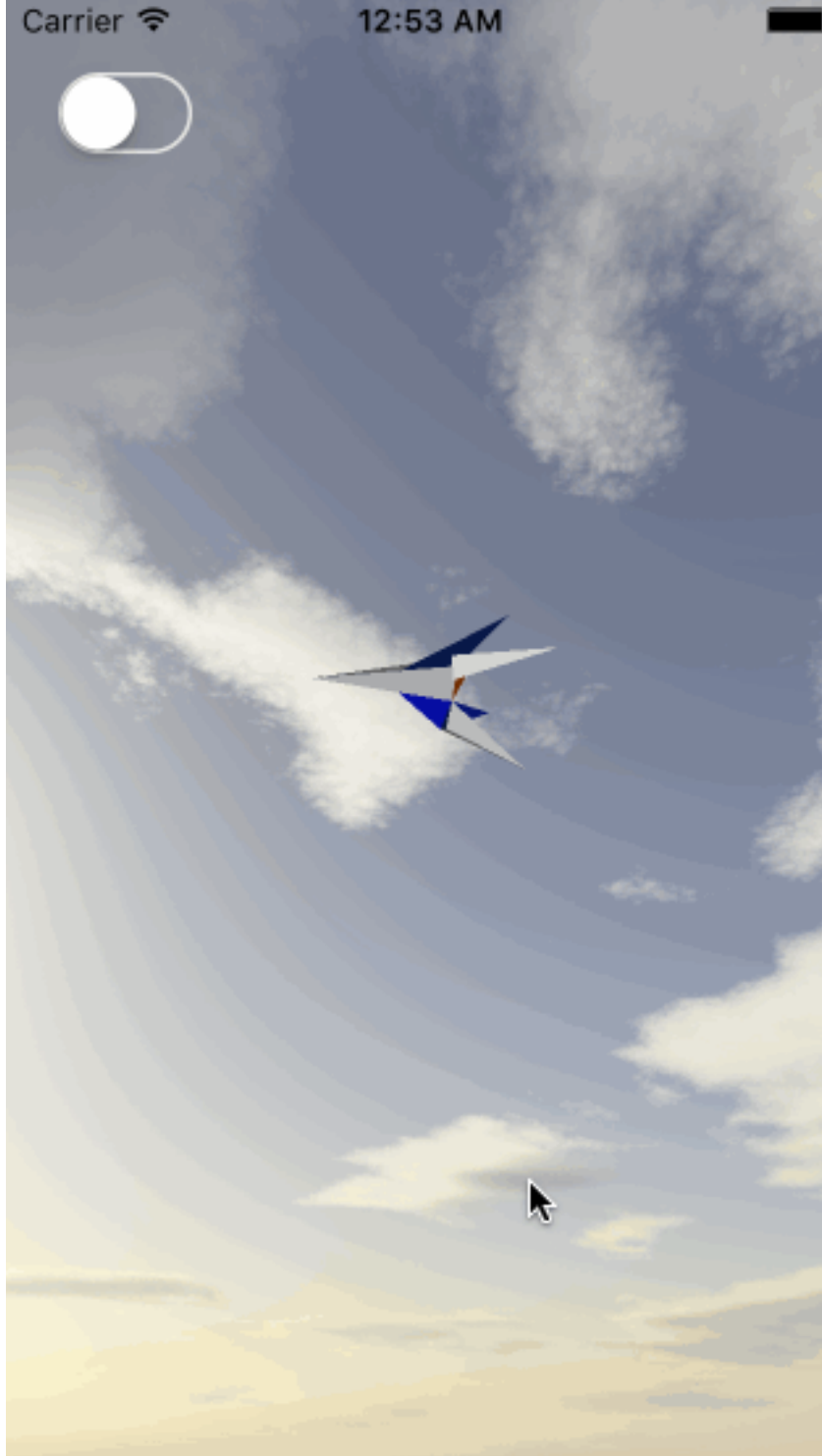
OpenGL ES入门教程1-Tutorial01-GLKit (<http://www.jianshu.com/p/750fde1d8b6a>)
OpenGL ES入门教程2-Tutorial02-shader入门 (<http://www.jianshu.com/p/ee597b2bd399>)
OpenGL ES入门教程3-Tutorial03-三维变换 (<http://www.jianshu.com/p/87c5413c1fc7>)
OpenGL ES入门教程4-Tutorial04-GLKit进阶 (<http://www.jianshu.com/p/ed7fb9555839>)
OpenGL ES进阶教程1-Tutorial05-地球月亮 (<http://www.jianshu.com/p/a82f3f66ddddd>)
OpenGL ES进阶教程2-Tutorial06-光线 (<http://www.jianshu.com/p/4e1a28f23e75>)
OpenGL ES进阶教程3-Tutorial07-粒子效果 (<http://www.jianshu.com/p/b6d2441209f8>)
OpenGL ES进阶教程4-Tutorial08-帧缓存 (<http://www.jianshu.com/p/1193b98634a2>)
OpenGL ES进阶教程5-Tutorial09-碰碰车 (<http://www.jianshu.com/p/3b532f6fcedf>)
OpenGL ES进阶教程6-Tutorial10-平截体优化 (<http://www.jianshu.com/p/bc151ff65cef>)
这一次的内容是精心准备的天空盒特效，为了节约大家时间，这次在教程里面不贴代码，demo部分的内容都是干货。
写这个demo的过程中遇到了一些坎，最后会提到。
特别留意天空盒纹理坐标推导和顶点数据对象切换。

概念准备

天空盒特效：OpenGL ES提供了一个立方体贴图（cube mapping）的专门用于产生天空盒效果的纹理贴图模式。
举例：一个人，站在立方体的中间，上下左右前后看到的都是立方体的图片。

效果展示

为节省流量，gif比较模糊，清晰效果可以看demo。



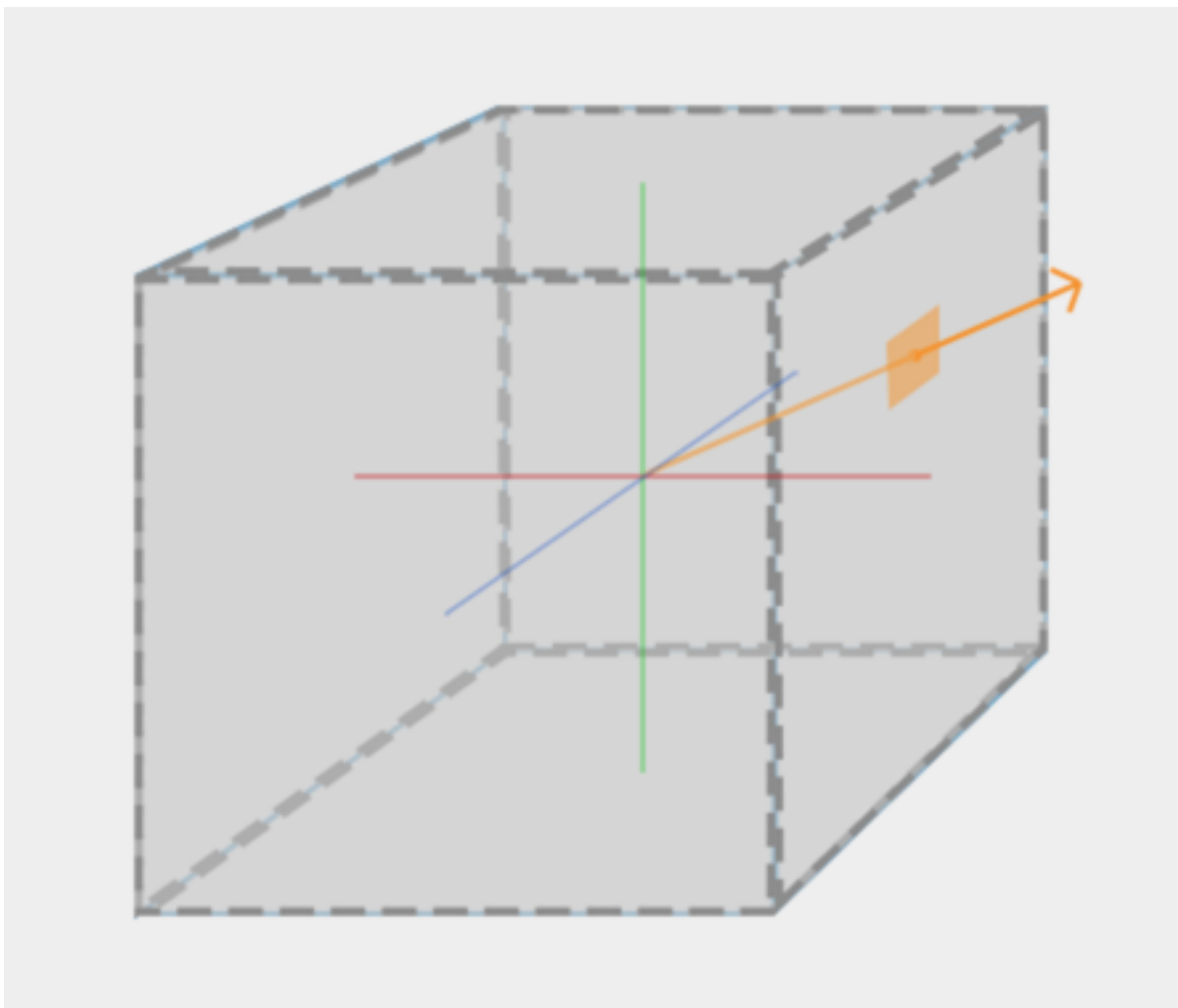
核心思路

天空盒的核心就是通过**方向**来取样纹理，纹理坐标被当作方向向量，建立适合的正方体后，位置坐标就是纹理坐标。

具体细节

1、尺寸大小

天空盒的尺寸可以随意，但是需要足够大以容纳渲染的场景。
同时天空盒的**中心**要尽可能**贴近视点**的眼睛位置，避免太近产生纹理拉伸。



2、纹理坐标到纹素推导（核心）

纹理坐标(s, t, r)被当作方向向量看待，每个纹理单元都表示从原点所看到的纹理立方体上的图像。

如果是texture2D的情况，纹理坐标(s, t)会直接返回相应位置的纹素；

textureCube的情况，首先读取cube纹理，然后以正方体中心为原点，(s,t,r)为方向，求出正方体和方向向量的交点位置(a, b)，按照位置(a, b)在纹理上面选择相对应的纹素。

举个例子：

对于(s, t, r) , 假设 $S = \text{fabs}(s)$, 同理有T R。

如果 $S > T$ 并且 $S > R$ 。那么可以确定交点在面+x 和 -x

根据s的±可以确定±x面。
直线过原点和点(s, t, r) ， 那么也会过点(1, t/s, r/s)。
(t/s) 和(r/s)就是对应的纹素位置。
考虑到立方体的面为[-1, 1]，那么可以把 (t/s + 1) / 2，这样就得到真正的纹素坐标((t/s + 1) / 2, (r/s + 1) / 2)

3、顶点数据对象切换（核心）

- **glBindVertexArrayOES**
很多应用会在同一个渲染帧调用多次glBindBuffer()、glEnableVertexAttribArray()和glVertexAttribPointer()函数（用不同的顶点属性来渲染多个对象）
新的顶点数据对象(VAO) 扩展会几率当前上下文中的与顶点属性相关的状态，并存储这些信息到一个小的缓存中。之后可以通过单次调用glBindVertexArrayOES() 函数来恢复，不需要在调用glBindBuffer()、glEnableVertexAttribArray()和glVertexAttribPointer()。
- **VAO和VBO**
VBO： 顶点缓冲区对象(buffer-object)，用于存储顶点坐标、纹理坐标、顶点法线、顶点颜色等。
VAO： 顶点数据对象，记录顶点数据存储状态信息的状态对象（status-object）。

This extension introduces vertex array objects which encapsulate vertex array states on the server side (vertex buffer objects).

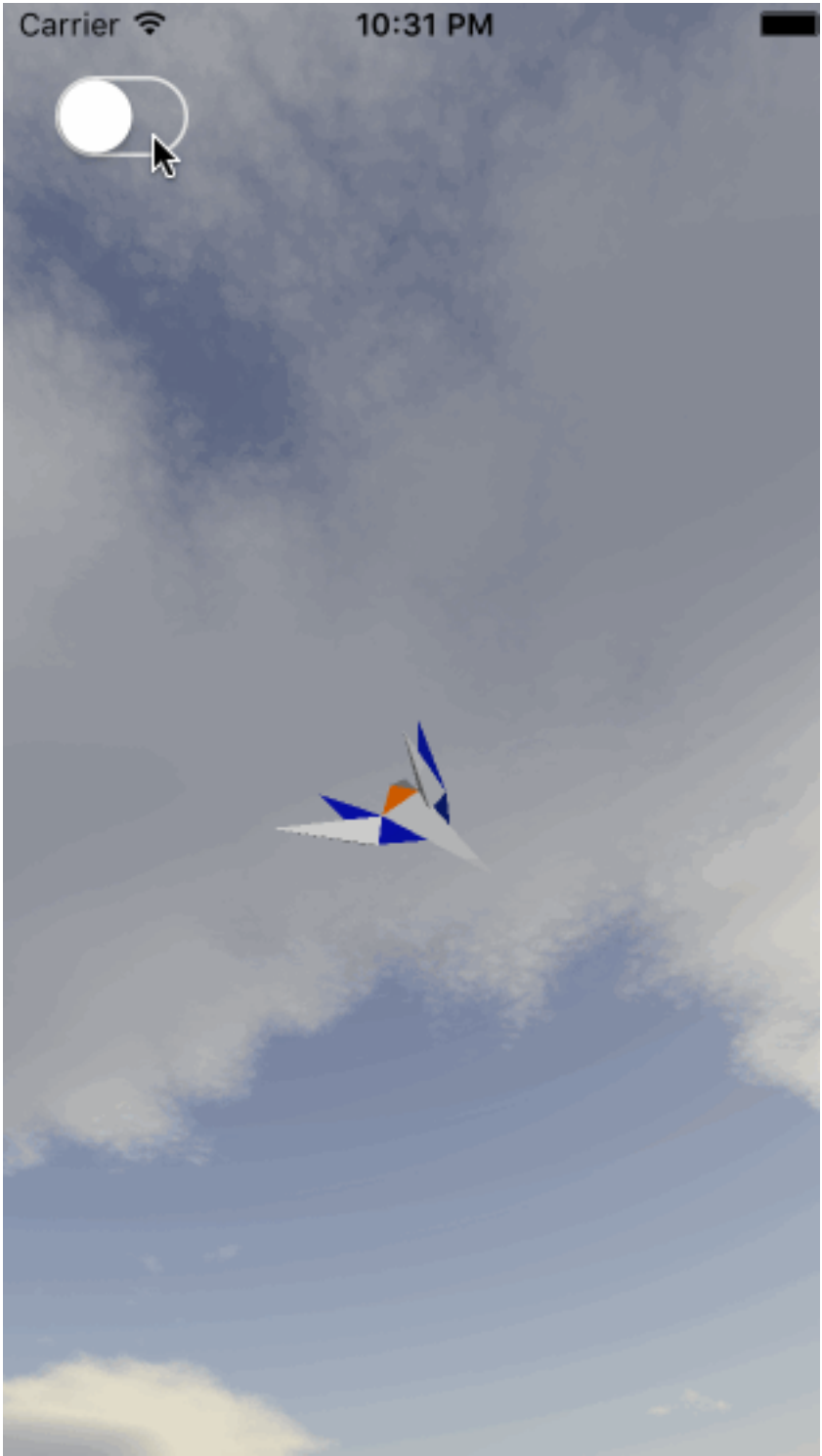
These objects aim to keep pointers to vertex data and to provide names for different sets of vertex data. Therefore applications are allowed to rapidly switch between different sets of vertex array state, and to easily return to the default vertex array state.

Q:Should vertex array objects be sharable across multiple OpenGL ES contexts?

A: No.

总结

demo实现过程遇到最大的一个坑，如下：



暂停的适合，天空盒的效果会消失！

然后开始寻找问题所在，最后发现问题代码出现在这里

```
// 增加角度
if (!self.mPauseSwitch.on) {
    self.angle += 0.01;
}
```

实在无法理解为什么角度的改变会影响天空盒的显示。

回顾了一下OpenGL ES的绘制过程，从顶点缓存到变换、着色到帧缓存，发现天空盒的绘制都没有问题。

接着开始思考，会不会是飞机的绘制影响了天空盒的绘制？因为这是两个着色器，存在不同的顶点数据和纹理。

于是尝试在绘制完天空盒后调用下面，防止天空盒绑定的数据缓存被飞机的影响。

```
glBindBuffer(GL_ELEMENT_ARRAY_BUFFER, 0);
glBindTexture(GL_TEXTURE_CUBE_MAP, 0);
```

在绘制完飞机后调用下面，防止飞机的顶点数据去影响到天空盒。

```
glDisableVertexAttribArray(GLKVertexAttribPosition);
glDisableVertexAttribArray(GLKVertexAttribNormal);
glBindTexture(GL_TEXTURE_2D, 0);
```

然而并没有什么用 ==！

问题搁在心头好几天，每天都会尝试，也在google查GLKSkyboxEffect相关的问题，可惜GLKSkyboxEffect是苹果官方自己实现。

本来天空盒是上周就已经实现好，为了写这篇文章实现了一个暂停的功能，就出现这个bug。虽然去掉暂停功能很正常，但是demo里面存在问题。

经过很多天尝试后，已经可以确定的是，是飞机的绘制影响了天空盒的位置，角度的旋转只是隐藏了bug。

开始寻找非OpenGL ES的文章，看看OpenGL的天空盒实现，同时查看苹果官方的文档。

最后偶然在苹果的文档中看到一个关键词 OES ，我似乎明白了什么。

OES是OpenGL ES的一个非标准扩展，天空盒里面有用到，而我并没有处理。

尝试用OES来管理飞机的顶点模型。

```
//    glGenVertexArraysOES(1, &_mPositionBuffer);
//    glBindVertexArrayOES(_mPositionBuffer);
```

问题果然迎刃而解：暂停的时候天空盒是正常的。

接下来开始尝试不用OES，寻找问题的根源。

最后的结论：天空盒的绘制调用了 `glBindVertexArrayOES()`，可是在绘制结束后没有 `glBindVertexArrayOES(0)`；导致飞机的顶点数据影响了天空盒的顶点数据。

解决方案：在绘制完天空盒后调用 `glBindVertexArrayOES(0)`；，问题完美解决。


Tips

天空盒还有两部分内容：一个是切图，这个比较简单，用CoreGraphics即可；另一个是用Shader来实现天空盒，而非GLKSkyboxEffect，这部分加进来篇幅就过长了。这两部分都在github上面放了源码，都已经编译调试没问题，下载完可以直接运行。

附上代码 (<https://github.com/loyinglin/LearnOpenGLES/tree/master/Tutorial11-%E5%A4%A9%E7%A9%BA%E7%9B%92/SkyBox>)

思考题

- 为什么视点距离天空盒太近会产生纹理拉伸？



落影loyinglin (/u/815d10a4bdce)

写了 171405 字，被 4763 人关注，获得了 2745 个喜欢

(/u/815d10a4bdce)写了 171405 字，被 4763 人关注，获得了 2745 个喜欢

+ 关注

工程师一枚，喜欢思考，喜欢游戏，喜欢运动。做过什么已经不重要，未来的方向以及当下的准备是生活的...

♡ 喜欢 (/sign_in?utm_source=desktop&utm_medium=not-signed-in-like-button) | 15














更多分享

被以下专题收入，发现更多相似内容 (http://cwb.assets.jianshu.io/notes/images/3857392)

-  iOS 开发 (/c/2ffaa203eb6a?utm_source=desktop&utm_medium=notes-included-collection)
-  iOS Dev... (/c/3233d1a249ca?utm_source=desktop&utm_medium=notes-included-collection)
-  iOS学习 (/c/1332c736fe39?utm_source=desktop&utm_medium=notes-included-collection)
-  iOS入的那些坑 (/c/2ac262be8a53?utm_source=desktop&utm_medium=notes-included-collection)
-  OpenGL ... (/c/044a5240577d?utm_source=desktop&utm_medium=notes-included-collection)
-  ios零碎记录 (/c/9af112b55b3a?utm_source=desktop&utm_medium=notes-included-collection)
-  iOS开发专题 (/c/c258bc0ea6bd?utm_source=desktop&utm_medium=notes-included-collection)

展开更多 ▼

 登录/注册

为你个性化推荐内容

(/sign_in?utm_source=desktop&utm_medium=not-signed-in-bind)

 下载简书App

随时随地发现和创作内容

(http://app.jianshu.io/download?utm_source=desktop&utm_medium=click-note-bottom-bind)