

# iOS开发-OpenGL ES入门教程1



(/u/815d10a4bdce) (/u/815d10a4bdce) + 关注

2016.03.25 16:29\* 字数 770 阅读 23185 评论 112 喜欢 108 赞赏 4 阅读 23185 评论 112 喜欢 108 赞赏 4

## 前言

这里是一篇新手教程，环境是**Xcode7+OpenGL ES 2.0**，目标写一个OpenGL ES的hello world。

OpenGL ES系列教程在这里 (<http://www.jianshu.com/notebooks/2135411/latest>)。

OpenGL ES系列教程的代码地址 (<https://github.com/loyinglin/LearnOpenGLES>)

你的star和fork是我的源动力，你的意见能让我走得更远。

## 核心思路

通过GLKit，尽量简单地实现把一张图片绘制到屏幕。

## 效果展示



## 具体细节

### 1、新建OpenGL ES上下文

```
- (void)setupConfig {
    //新建OpenGL ES 上下文
    self.mContext = [[EAGLContext alloc] initWithAPI:kEAGLRenderingAPIOpenGL ES2];
    //2.0, 还有1.0和3.0
    GLKView* view = (GLKView *)self.view; //storyboard记得添加
    view.context = self.mContext;
    view.drawableColorFormat = GLKViewDrawableColorFormatRGBA8888; //颜色缓冲区格式
    [EAGLContext setCurrentContext:self.mContext];
}
```

GLKView\* view = (GLKView \*)self.view; 这里需要在storyboard里面把view的类设置成GLKView，其他代码是OpenGL ES上下文的创建。

## 2、顶点数组和索引数组

```
//顶点数据，前三个是顶点坐标，后面两个是纹理坐标
GLfloat squareVertexData[] =
{
    0.5, -0.5, 0.0f,    1.0f, 0.0f, //右下
    0.5, 0.5, -0.0f,    1.0f, 1.0f, //右上
    -0.5, 0.5, 0.0f,    0.0f, 1.0f, //左上

    0.5, -0.5, 0.0f,    1.0f, 0.0f, //右下
    -0.5, 0.5, 0.0f,    0.0f, 1.0f, //左上
    -0.5, -0.5, 0.0f,    0.0f, 0.0f, //左下
};
```

顶点数组里包括顶点坐标，OpenGL ES的世界坐标系是[-1, 1]，故而点(0, 0)是在屏幕的正中间。

纹理坐标系的取值范围是[0, 1]，原点是在左下角。故而点(0, 0)在左下角，点(1, 1)在右上角。

索引数组是顶点数组的索引，把squareVertexData数组看成4个顶点，每个顶点会有5个GLfloat数据，索引从0开始。

## 3、顶点数据缓存

```
//顶点数据缓存
GLuint buffer;
glGenBuffers(1, &buffer);
glBindBuffer(GL_ARRAY_BUFFER, buffer);
glBufferData(GL_ARRAY_BUFFER, sizeof(squareVertexData), squareVertexData, GL_STATIC_DRAW);

glEnableVertexAttribArray(GLKVertexAttribPosition); //顶点数据缓存
glVertexAttribPointer(GLKVertexAttribPosition, 3, GL_FLOAT, GL_FALSE, sizeof(GLfloat) * 5, (GLfloat *)NULL + 0);
glEnableVertexAttribArray(GLKVertexAttribTexCoord0); //纹理
glVertexAttribPointer(GLKVertexAttribTexCoord0, 2, GL_FLOAT, GL_FALSE, sizeof(GLfloat) * 5, (GLfloat *)NULL + 3);
```

这是本章节的核心内容。

- glGenBuffers申请一个标识符
- glBindBuffer把标识符绑定到 GL\_ARRAY\_BUFFER 上
- glBufferData把顶点数据从cpu内存复制到gpu内存
- glEnableVertexAttribArray 是开启对应的顶点属性
- glVertexAttribPointer设置合适的格式从buffer里面读取数据

## 4、纹理贴图

```
- (void)uploadTexture {
    //纹理贴图
    NSString* filePath = [[NSBundle mainBundle] pathForResource:@"for_test" ofType:@"jpg"];
    NSDictionary* options = [NSDictionary dictionaryWithObjectsAndKeys:@(1), GLKTextureLoaderOriginBottomLeft, nil]; //GLKTextureLoaderOriginBottomLeft 纹理坐标系是相反的
    GLKTextureInfo* textureInfo = [GLKTextureLoader textureWithContentsOfFile:filePath options:options error:nil];
    //着色器
    self.mEffect = [[GLKBaseEffect alloc] init];
    self.mEffect.texture2d0.enabled = GL_TRUE;
    self.mEffect.texture2d0.name = textureInfo.name;
}
```

- GLKTextureLoader读取图片，创建纹理GLKTextureInfo
- 创建着色器GLKBaseEffect，把纹理赋值给着色器

## 基础

代码带了很多注释，百度下相应的概念，会有更多解释。  
如果对OpenGL ES感兴趣，但是却毫无图形学基础的，可以看看LearnOpenGL教程 (<https://learnopengl-cn.github.io/>)。

## 思考题

- 1、代码中有6个顶点坐标，能否使用更少的顶点显示一个图像？
- 2、顶点缓存数组可以不用glBufferData，要如何实现？
- 3、如果把这个图变成左右两只对称的熊猫，该如何改？

这里 (<https://github.com/loyinglin/LearnOpenGLES/tree/master/Tutorial01-GLKit>)可以下载demo代码。

## 思考题答案

- 思考题1：**  
可以使用四个顶点，绘制2个三角形 的6个顶点中有2个是重复的，使用索引可以减少重复。
- 思考题2：**  
顶点缓存数组可以不用glBufferData，要如何实现？ 顶点数组可以通过glBufferData放入缓存，也可以直接通过glVertexAttribPointer最后一个参数，直接把顶点数组从CPU传送到GPU。区别： glBufferData里面的顶点缓存可以复用， glVertexAttribPointer是每次都会把顶点数组从CPU发送到GPU，影响性能。
- 思考题3：**  
如果把这个图变成左右两只对称的熊猫，该如何改？ 把屏幕切分成4个三角形，左边两个三角形同上，右边两个三角形的纹理坐标的x值调整即可。



落影loyinglin (/u/815d10a4bdce)

写了 170235 字，被 4747 人关注，获得了 2741 个喜欢

/u/815d10a4bdce 写了 170235 字，被 4747 人关注，获得了 2741 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...

赞赏是支持别人，也是加深自己的记忆。

赞赏支持



(/u/fc2c357b4235b4058e50)

喜欢 (/sign\_in?utm\_source=desktop&utm\_medium=not-signed-in-like-button)

108

更多分享

(http://cwb.assets.jianshu.io/notes/images/3359206

被以下专题收入，发现更多相似内容

- iOS Dev... (/c/3233d1a249ca?utm\_source=desktop&utm\_medium=notes-included-collection)
- iOS开发 (/c/072d578bf782?utm\_source=desktop&utm\_medium=notes-included-collection)
- OpenGL ... (/c/044a5240577d?utm\_source=desktop&utm\_medium=notes-included-collection)
- iOS开发专题 (/c/c258bc0ea6bd?utm\_source=desktop&utm\_medium=notes-included-collection)
- OpenGL... (/c/71a20298524f?utm\_source=desktop&utm\_medium=notes-included-collection)
- iOS备忘录 (/c/536499c136f6?utm\_source=desktop&utm\_medium=notes-included-collection)
- OpenGL ... (/c/408442c9c764?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://www.jianshu.io/app/download?utm\_source=desktop&utm\_medium=click-note-bottom-bind)