跑酷Demo教程1

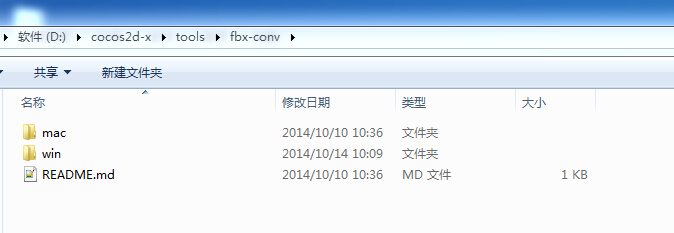
**[教程内所有素材来自网络，如有侵权，请联系删除]**

这个教程主要是通过分析一个跑酷的Demo，来向大家简单的介绍一下Cocos2d-x 3.3 里对3D游戏开发提供的一些功能，并介绍一下他们的使用方法。

1 载入角色模型

Cocos2d-x 3.3 提供了对Sprite3D的支持，即可以直接加载一个模型格式，到目前为止，支持三种格式的读取解析——obj（静态格式，仅用于内部测试），c3b（Cocos2d-x自带的一种模型格式），c3t（与c3b对应的一种模型格式，它是可以直接用文本编辑查看修改的，方便在游戏开发中Debug ）。

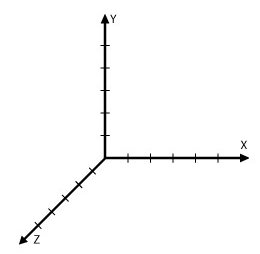
Cocos2d-x提供了一个工具，可以很方便的将FBX格式导出成c3b、c3t。该工具在cocos2d-x\tools\fbx-conv下，分别对mac和windos提供了版本。

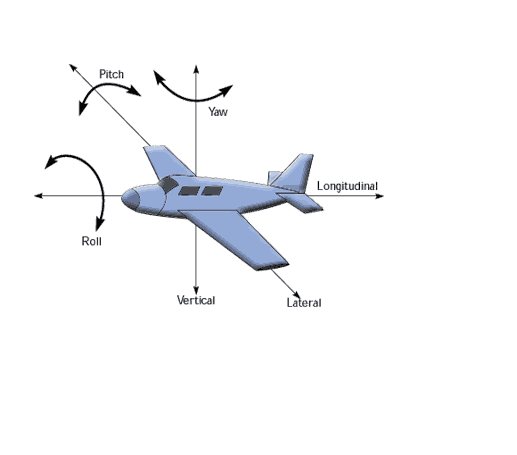


在将模型文件转换完成之后，我们就可以将其读取到场景之中去了。为了好合理的安排代码，我们把这段代码写在Player这个类中。

![C:\Users\ziwen.tang\AppData\Roaming\Tencent\Users\576419632\QQ\WinTemp\RichOle\D)AV36BQOTU~](YJ_GA0TOP.png](data:image/png;base64,)

第一行代码根据指定的模型创建了3D模型，第二行第三行主要是对模型大小进行校正，第二行矫正了一下模型的大小，第三行就和Cocos2d-x 中2D部分的功能不太一样，使用了Rotation3D。这里要稍微注意一下3D三个轴旋转的问题，这里接受的Vec3的X,Y,Z分别代表了物体在自己的局部坐标系中，绕着X轴，Y轴，Z轴进行的旋转。这里要注意一下X，Y，Z轴是怎么摆放的。

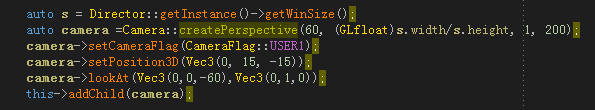
如左图所示，角色的模型坐标系的正面是朝向Z轴的负方向的，由此图可看出，绕着X,Y，Z轴的旋转分别相当于对俯仰角（pitch 、tilt），偏航角(yaw, pan),翻滚角（roll）。



设置好位置，大小，旋转方向了之后，接下来的就是设置它的动画，这里需要用Animation3D来生成相应的animation，最后在通过Animate3D来创建出action来播放动画。

如果一切顺利，那么你会在左下角的一个很小的位置看见角色………

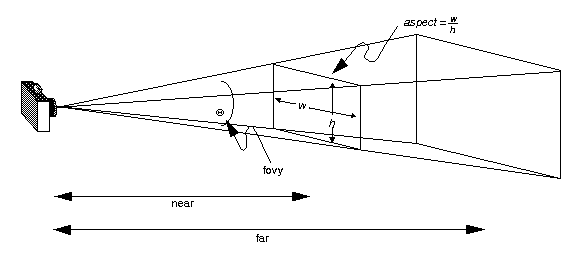
模型有了，在跑酷游戏里，视线，或者说相机是从人物的正上方稍稍往下望去，现在我们来设置相机。设置相机的代码也非常的简单，如下所示：



首先我们通过getWinSize()，获取了屏幕的宽和高，这是为之后创建摄像机时计算宽高比所准备的。

接着这一句，就是创建摄像机了，createPesepecitve的意思就是创建一个透视投影的摄像机，所谓的透视即在3d场景下，距离相机位置不同的物体，他们的大小，位置有所不同，可以想象一下自己站在铁路上，两根铁轨在无穷远处交汇在一点。

第一个参数是FOV，即视场角(field of view),它可以理解为你的视线左右能看多宽（以角度计）第二个就是上述所有的宽高比，最后两个是相机的近裁面和远裁面，这个也很好理解，距离相机比近裁面还要近的，比远裁面还要远的，都不会被渲染到。



接下来，我们设置了相机的标记，在cocos2d-x中，允许同时存在多个相机并让物体选择起作用的相机，设置标记的目的就在于表明相机的“身份”。让用户能通过这个标记的不同来设置物体受到影响的相机。

LookAt函数用于设置相机朝向何方，它接受两个参数，第一个是，目标的坐标，第二个向上方是何处，因为相机和人的眼睛看东西类似，在旋转角度不同看到的画面也不同，在这里我们直接使用和y轴正方向上的一个单位向量表示。

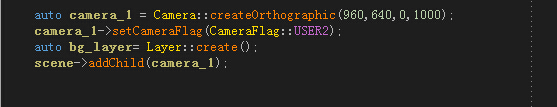
接下来，我们将相机插入到这个layer中。

相机设好了，我们就要将一系列的Sprite作用于相机，因此我们要用到Node的setCameraMask：

C:\Users\ziwen.tang\AppData\Roaming\Tencent\Users\576419632\QQ\WinTemp\RichOle\1SH37DSEN]Q3(PPRKP$DSC2.png，

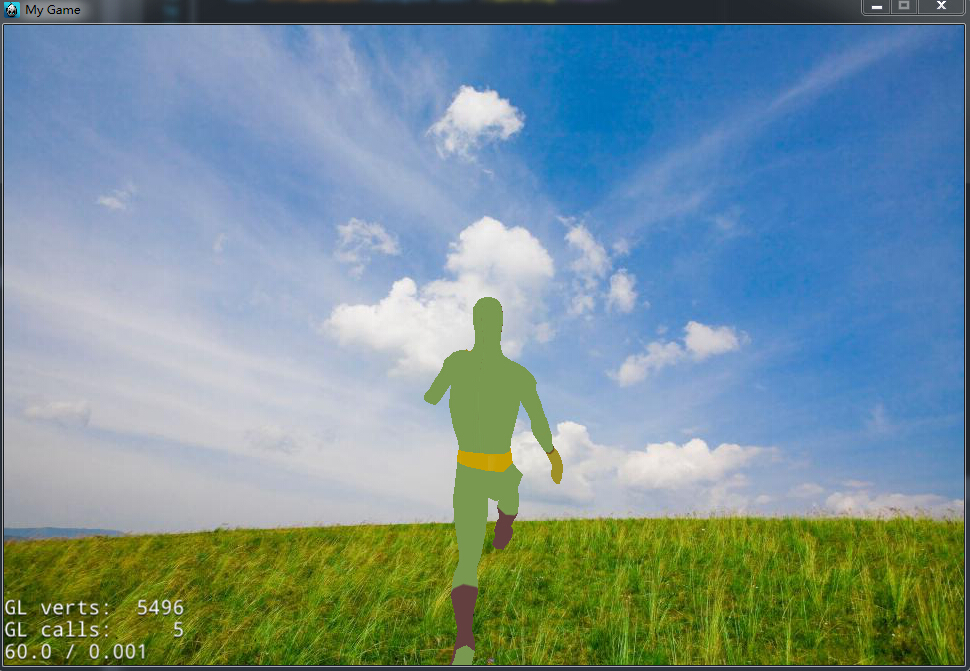
请注意这个函数只是将Node下当前的子节点设为使用这个相机，在SetCameraMask之后添加的子节点不属于此范畴——它们仍然会使用默认的相机，所以大家一定要十分小心，在确定使用相机的时候，一定要新添加节点之后重新SetCameraMask。

现在基本的东西我们都加好了，现在我们要加一个背景，背景是是用一张2D的图片去渲染的，所以它并不适用透视投影，我们直接创建一个正交投影就可以了：



接着创建一个一般的精灵，然后设置好相机就成了。

那么最后的效果如下：



本次教程就到这里，下一次我们会加入能行走的地面已经一些树，以及控制玩家移动的一些功能~