CASE开发进阶文档

在准备进行case制作的时候，需要如下的一些资料。下面逐一介绍一下这些资料及用途：

1. BaiduAR——工程项目，项目中已经将渲染引擎集成好，运行时只需要替换相应的资源包即可（要在真机下运行）。资源包中包括所呈现的3D模型、贴图、纹理图片、音频以及json文件（json文件是我们需要修改的，也是我们在做case的过程中主要学习去写的）。
2. AR-新协议-iOS版本详细说明——协议格式框架，即json文件的格式框架。所有的json文件都是按照这个文件所呈现的框架来写的。这份说明中对各个模块都有示例说明，相关参数、字段等也有一些注释，大部分都可以看懂的，可以参照着自己来尝试写一下。（里面稍微注意一下的就是“指令相关”的部分，因为指令在执行的过程中可能存在冲突，如：一个点击事件播放了一段音频，这段音频还未结束，又点击了另一个按钮，这时候前一个指令还未完成，后面的指令又来了，所以存在一个执行的逻辑问题。所以这个指令相关部分解释了这个问题。）
3. Demo触发图——即程序要扫描的图片。
4. Case制作工具——即完整的case制作过程需要的相关工具，如下：（因为一开始我们只写json文件，3D模型是处理过的，我们直接用即可。这个过程我们要将设计所给的3D模型自己处理一下才能应用到工程中。）
5. 模型空节点处理——设计所给我们的3D模型中可能存在着未命名的节点（用XCode打开DAE文件，在Scene graph中找到节点，然后展开到最深层，可以看到有的节点被命名为<untitled>，所以这些节点我们需要给他重命名，统一命名一下即可。由于存在的节点很多，资料中提供了一个示例python脚本及使用说明，然后运行即可。）
6. 模型压缩工具——设计所给的3D模型是比较大的，不能直接添加到工程中运用，需要我们进行压缩后才能使用。（文件夹中给了三个文件：copySceneKitAssets与scntool及使用说明，按照使用说明运行即可。）
7. 练习——里面提供了一些需要实现的实例，可以参考着已经写好的示例来完成这些实例。这些练习主要从易到难，会有1个模型到多个模型、静态到动态等，逐步复杂，一步一步来实现即可。实现顺序：静态模型—动画模型—序列帧—音乐，按照这个顺序来做即可。

（注：一开始的时候，先根据示例json文件了解其原理，之后再深入去实现整个过程。）

练习要求：

1. 静态模型：扫描触发图，实现模型展示即可。
2. 动画模型：扫描触发图，实现点击模型旋转与模型自动缩放。
3. 序列帧：扫描触发图，实现序列帧顺序播放。
4. 添加音效：扫描触发图，添加下一步按钮，每点击一个按钮播放对应音效（当没播完时，点击下一步了，会产生指令的逻辑问题需注意）

上面大概的介绍了一下所需要的文件资料及一些操作方法，下面是实际制作case的一个过程，供参考。

步骤如下：

1. 查看3D模型。看一下设计所给的3D模型中是否存在未命名的空节点，如果不存在，跳过第一步即可；如果存在，运行脚本（资料中提供了script.py）将其重新统一命名。将xxx.DAE文件放到相应目录下，打开terminal到对应目录，运行python script.py xxx即可。
2. 进行模型压缩。新建一个文件夹，暂时起名为model，更改后缀为.scnassets，将我们生成好的模型文件拷贝进去。SceneKit对于动态添加文件夹写了两个脚本，一个是copySceneKitAssets、另一个是scntool。将copySceneKitAssets、scntool文件拷贝到与model.scnassets同级目录下，进入终端并cd到该目录下，运行下面这条命令即可：./copySceneKitAssets model.scnassets -o model-o.scnassets
3. 编写json文件。参考着之前写好的json文件，先将整体的框架搭出，然后再向里面添加所需要的一些特效。
4. 组装目录。将处理好的3D模型、贴图、音频及json文件放入到指定的资源目录下面即resource.bundle中。

下面介绍一下case目录（附录中有示例目录截图）：最外层是自己定义的文件夹（注意：最外层自己定义的文件夹名字要在ShellViewController.m中按钮的点击事件中（path）更改为对应的名字），里面即为res文件夹与json文件。在res文件中主要包含主要包含models、sounds、target文件夹。models中主要存放3D模型及一些贴图；sounds中主要存放所需的背景音乐及其他一些音频；target中主要存放了一张image\_target.png图片（这个用默认的即可）及一个model文件，model文件夹中存放了已经设计好的触发图的模型（制作方法：登录[http://ardev.baidu.com/dev/list](javascript:openUrl('http://ardev.baidu.com/dev/list'))网址，第一次登录需要注册一下，之后点击创建应用，选择开发者模式，填写信息，上传识别图即为我们的触发图，格式要求jpg，图片不要太大，宽高不要超过1024px，点击确定，等待半小时到一小时左右，当所创建的应用可以下载了即可。下载的文件包括我们需要的触发图模型以及一个文本文件roi.txt，该文本文件中的数据对应我们缩写的json文件中图片大小与位置信息。前两个对应originImage的宽高属性，最后两个对应featureImage的宽高属性，至于中间两个则需要进行一个简单运算生成的数据对应featureImage的position属性。运算为：第三个数据+第五个数据/2；第四个数据+第六个数据/2，生成的这两个数据对应position的前两个值，最后一个值为0即可）。（注：目录仅供参考，自己可以按照自己所需的进行修改）

1. 运行，扫描对应的触发图即可。

附录（仅供参考）：

