



本章学习目标

- 了解 Vuforia 及其基础功能。
- 掌握使用 Vuforia 进行项目开发的步骤。

本章首先简要介绍 Vuforia 及其基础功能,然后重点讲解使用 Vuforia 进行项目开发的详细步骤和实现效果。

5.1 Vuforia 简介

Vuforia 是一款能将现实世界物体与虚拟物品进行互动体验的 AR 开发平台。该平台利用计算机视觉技术,可以实时地识别和跟踪现实世界中的平面图像及简单的 3D 物体,使开发者能够在现实世界和数字世界之间架起体验的桥梁。

Vuforia 通过 Unity 游戏引擎扩展提供了 C、Java、Objective-C 和 .NET 语言的应用程序编程接口,能够同时支持 iOS 和 Android 的原生开发,这也使开发者在 Unity 引擎中开发的 AR 应用很容易移植到 iOS 和 Android 平台上。

5.2 Vuforia 基础功能

1. Image Targets(图片识别)

通过 Vuforia 的图片识别功能,可以在图片上呈现模型的 AR 效果,如图 5.1 所示。

2. Cylinder Targets(圆柱体识别)

通过 Vuforia 的圆柱体图片识别功能,能在图片上呈现模型的 AR 效果,如图 5.2 所示。



图 5.1 Vuforia 呈现的图片识别效果



图 5.2 Vuforia 呈现的圆柱体图片的识别效果

3. Multi Targets(立方体识别)

通过 Vuforia 的立方体图片识别功能,能在图片上呈现模型的 AR 效果,如图 5.3 所示。

4. User Defined Targets(自定义识别)

通过截取当前屏幕内容作为识别图,呈现 AR 效果,如图 5.4 所示。



图 5.3 Vuforia 呈现的立方体图片的识别效果



图 5.4 Vuforia 的自定义图片识别效果

5. Text Recognition(文字识别)

Vuforia 还可以通过识别文字,呈现 AR 效果。目前文字基本支持英文,默认能够识别的英文单词必须是官方词库中的单词(十万个),用户也可以自定义添加英文单词,如图 5.5 所示。

6. Virtual Buttons(虚拟按钮)

在识别图上制作特殊的按钮区域,通过遮挡特殊区域来实现按钮的效果,如图 5.6 所示。



图 5.5 文字识别效果



图 5.6 挡住特殊区域来实现按钮的效果

7. Cloud Recognition(云识别)

将识别图和模型等资源放在网络端,通过识别图识别下载模型呈现 AR 效果,如图 5.7 所示。

8. Smart Terrain (Unity only)(智能地形)

以图片识别、圆柱体、立方体识别为基点,向四周扩散识别周围一些立方体、圆柱体等形状的物体,以此为基础建立模型场景,如图 5.8 所示。

9. Object Recognition(物体识别)

在手机上安装 Vuforia Object Scanner 并打印 A4-Object Scanning Target 360°扫描物

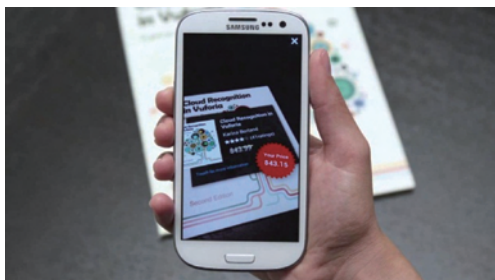


图 5.7 识别下载模型呈现 AR 效果

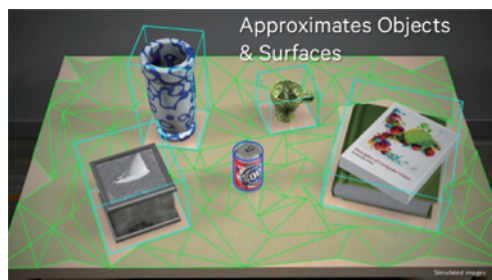


图 5.8 凭借立体图形建立模型场景

体,记录物体特征点,以此为基础进行识别,呈现 AR 效果,如图 5.9(a)~图 5.9(c)所示。



图 5.9 Object Recognition 识别过程

1) Penguin

Penguin 是 Vuforia 专门为 Smart Terrain(智能地形)而开发的一款 Demo 小游戏。以一瓶瓶装饮料为识别基点,向周围识别茶杯、书籍、花瓶等各种形状的物体,生成冰川场景,让小企鹅在冰雪中滑行,如图 5.10 所示。



图 5.10 Demo 小游戏

2) Find The Penguin

Find The Penguin 是 Vuforia 仿照 Pokémon Go 制作的一个 Demo。通过使用设备的陀螺仪和 Vuforia 的 User Defined Targets(自定义识别)功能,使模型能够固定在地面某一点,并且在周围随意活动,如图 5.11 所示。



图 5.11 Find The Penguin

5.3 Vuforia 项目开发

使用 Vuforia 进行项目开发,主要包括获取与导入 Vuforia SDK、搭建开发环境、创建 License Key、上传识别图、创建测试立方体、导入模型资源、创建虚拟按钮、动画制作、添加脚本和导出发布 10 个基本步骤。

5.3.1 获取与导入 Vuforia SDK

Vuforia SDK 已经内嵌到 Unity 引擎中,在安装 Unity 时使用默认的版本即可。也可以访问 Vuforia 的开发者官网 <http://developer.vuforia.com>,在主页的 Downloads 选项卡中下载最新版的 Vuforia SDK,如图 5.12 所示。从下载页面的说明可以看出,Vuforia 引擎中已经集成到最新的 Unity 版本。如果下载了最新的 Unity,在安装后就可以看到 Vuforia 模块。

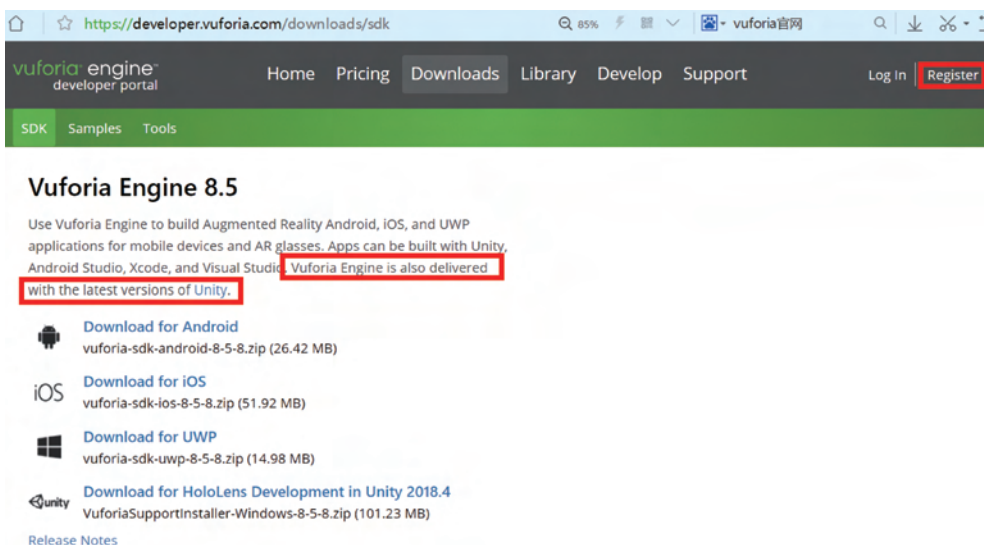
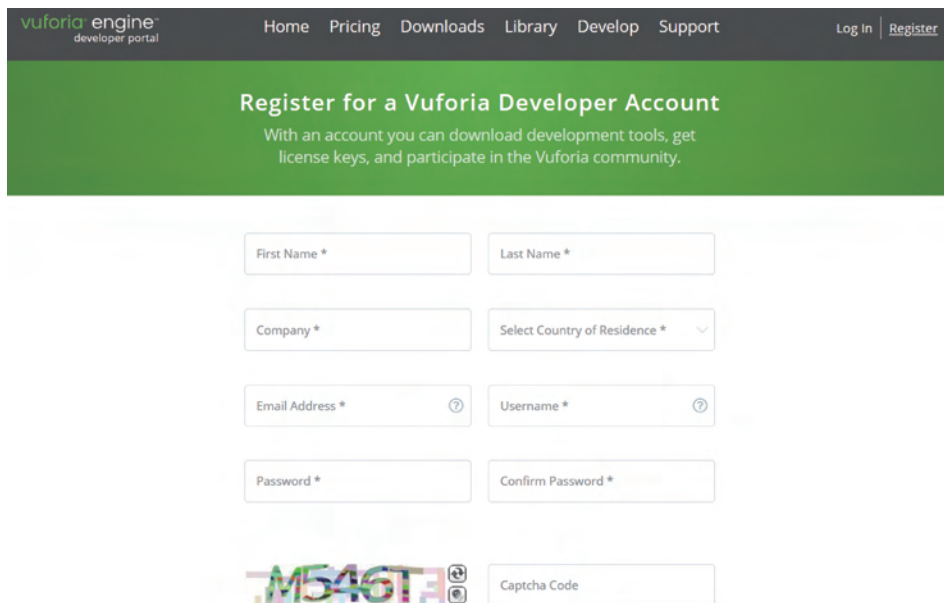


图 5.12 Vuforia SDK 下载页面

如果是新用户,需要单击右上角的 Register,完成账号注册才能登录下载 Vuforia SDK。如果新用户直接单击下载链接,也会自动跳转到账号注册页面,如图 5.13 所示。



The image shows the 'Register for a Vuforia Developer Account' page on the Vuforia engine developer portal. The page has a green header with the Vuforia logo and navigation links: Home, Pricing, Downloads, Library, Develop, Support, Log in, and Register. Below the header, a green banner contains the title 'Register for a Vuforia Developer Account' and a subtext: 'With an account you can download development tools, get license keys, and participate in the Vuforia community.' The registration form consists of several input fields: 'First Name *', 'Last Name *', 'Company *', 'Select Country of Residence *' (a dropdown menu), 'Email Address *' (with a help icon), 'Username *' (with a help icon), 'Password *', 'Confirm Password *', and a 'Captcha Code' field with a visual captcha image showing the letters 'M546T'.

图 5.13 新用户注册页面

安装 Unity 时如果使用的是默认版本,还需要在 Unity 中加载导入 Vuforia 模块。在 Unity 的 File 菜单选择 Build Settings,在弹出窗口中单击 Player Settings,在右侧的 Inspector 面板的 XR Settings 中如果看到有 Vuforia Augmented Reality 选项,就说明 Unity 中已经集成了 Vuforia,可以勾选该选项后的方框加载 Vuforia 模块,如图 5.14 所示。

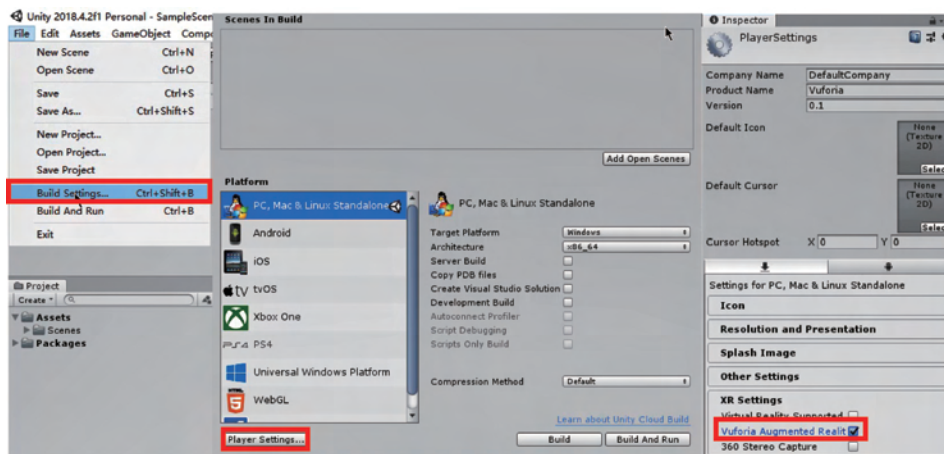


图 5.14 加载 Vuforia 模块

如果 Vuforia Augmented Reality 选项下是带一个下画线的链接,则说明安装的 Unity 中没有 Vuforia 模块,可以单击链接获取 Vuforia SDK 后再加载 Vuforia 模块。

5.3.2 搭建开发环境

加载 Vuforia 模块后,在 Unity 场景中看起来没有任何变化,只是在 Project 窗口中多

了 VuforiaConfiguration 项。要使用 Vuforia 开发项目还需要进行开发环境的搭建。首先导入 Vuforia 资源包,在 Unity 的 GameObject 菜单中选择 Vuforia Engine,添加一个 AR Camera,如图 5.15 所示,会弹出一个对话框,提示用户是否导入 Vuforia 的资源包。单击 Import 按钮,完成导入。

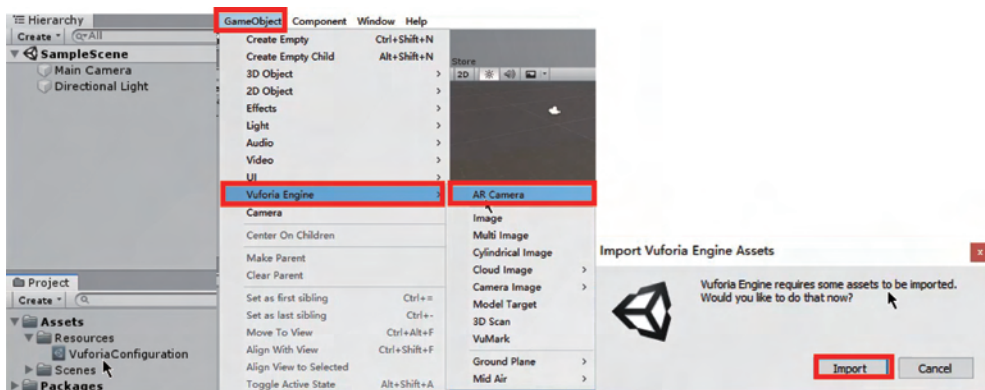


图 5.15 导入 Vuforia SDK 资源包

导入成功后,在 Hierarchy 窗口中会出现 AR Camera,如图 5.16 所示。这时需要把原来的 Main Camera 删除,将 AR Camera 移到 Hierarchy 窗口最上方。

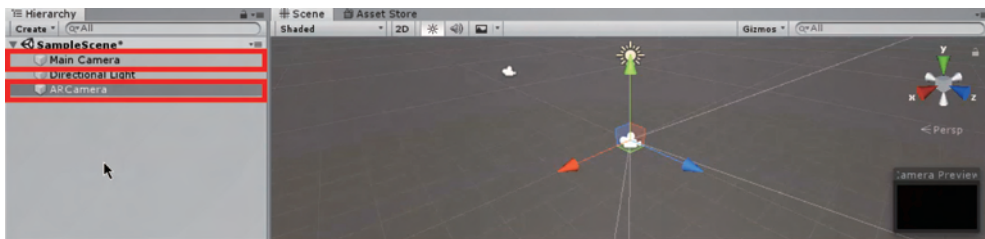


图 5.16 调整 AR Camera

然后添加识别图组件。在 GameObject 菜单中继续选择 Vuforia Engine,添加一个 Image,如图 5.17 所示,会弹出一个对话框,提示用户是否导入官方默认的认识图数据包。这里选择 Skip,使用自己提前准备好的识别图。

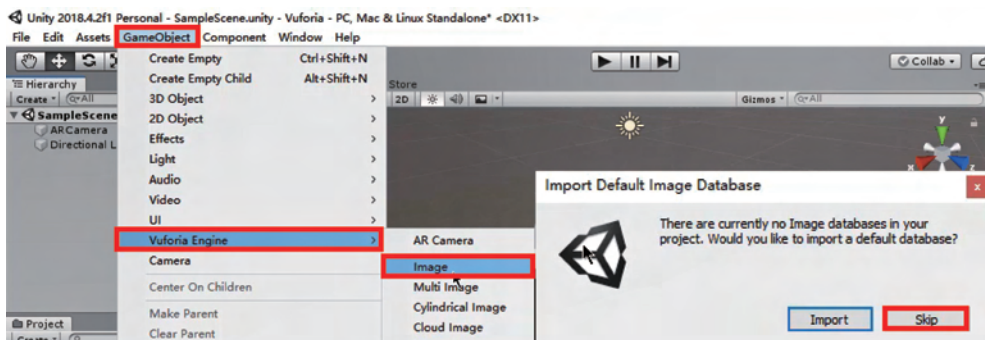


图 5.17 导入识别图数据包

在 Hierarchy 窗口中会出现 ImageTarget 组件,如图 5.18 所示。

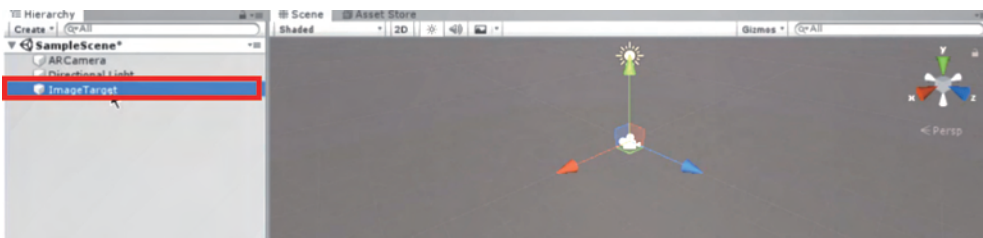


图 5.18 ImageTarget 组件

5.3.3 创建 License Key

登录到开发者平台,单击 License Manager(证书密钥管理),如图 5.19 所示。用户可以单击选择 Get Development Key 获取一个开发者版的密钥,也可以单击 Buy Deployment Key 按钮购买一个商业版的密钥。这里选取免费开发者版。

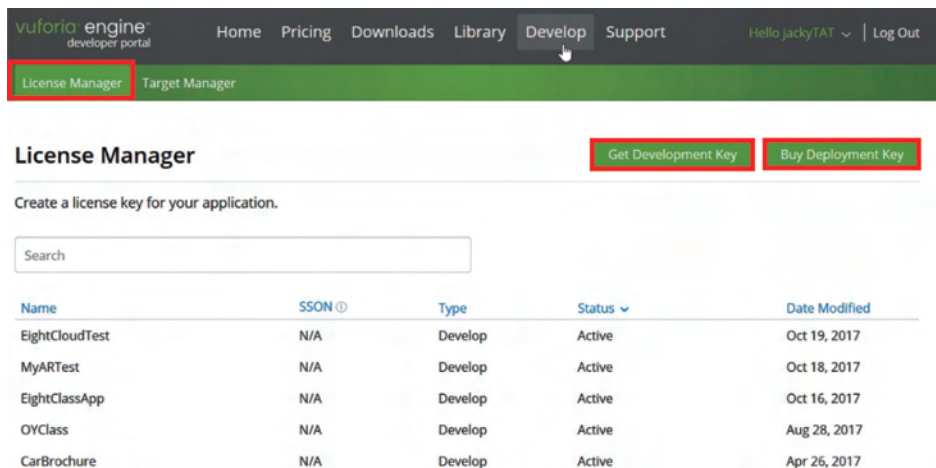


图 5.19 获取开发者版密钥

在 Add a free Development License Key 页面的 License Name 处填写证书名字(一般和要发布的 App 保持相同或相似即可),并在页面下方勾选同意获取 Vuforia Developer Agreement,最后单击 Confirm 按钮,如图 5.20 所示,就可以获取对应的密匙。也可以打开一个已有的密钥(以 MyARTest 为例),修改该密匙对应的证书名字,或单击 License Key,复制该 License Key 到剪贴板,如图 5.21 所示。

然后在 Unity 的 Hierarchy 窗口中单击 ARCamera,在右侧检视面板中单击 Open Vuforia Engine configuration,将刚才复制的 License Key 粘贴到 App License Key 右侧的矩形框内,如图 5.22 所示。

5.3.4 上传识别图

一般一个应用对应一个 License Key,一个 License Key 对应一个 Database。回到开发者后台,在 Target Manager 选项卡中,单击右侧的 Add Database 按钮,在弹出窗口中选择 Device,为数据库命名为 CarImage,单击 Create 按钮,如图 5.23 所示。

[Back To License Manager](#)

Add a free Development License Key

You can change this later

License Key

Develop

Price: No Charge

VuMark Templates: 1 Active

VuMarks: 100

☒ By checking this box, I acknowledge that this license key is subject to the terms and conditions of the [Vuforia Developer Agreement](#).

Cancel

Confirm

图 5.20 填写证书名字

License Manager > MyARTest

MyARTest

[Edit Name](#)[Delete License Key](#)

License Key

Usage

Please copy the license key below into your app

```
AbpzquX/////AAAAGcG+tkSto0Dyig2TmrKdoxVA86wvkjXY1/oBoiTO0aiD9yR7u3OZASxy2+D87WBqNBfZz2GUQ186FvmpAoY1zOz  
p9ultqx1lWwX6Ud4y6a1VedYoPG9nyAk6b7SHSGUEaME/Sy5+6bQVWCjn8Bu4hm7JaI5c+aDcb+YQC/05heS71FSVv9mJmuDGH3jR  
3vHKYyyAiTy1G5gLEZSYaXInZKSOfpdKxTishOpMfQcuvmurC+qa9Ujv1AstIH0XkDI2913u1UOmTvNFCcWbb31LMcVtdR8ohTiaX9S  
1X0T6rxBepEaAU6qXrznyOQ3QDng3TQuKjKIcJIngcdJBZ1CW08oTPL/5my4507r61XFP5L
```

Copied to clipboard

[View Vuforia 4 license key](#)

Plan Type: Develop

Status: Active

Created: Apr 18, 2016 15:47

License UUID: 69e7cff7080844fca6cf0a3134f2dd87

Device: Mobile

Permissions:

- Advanced Camera
- Watermark

图 5.21 复制 License Key

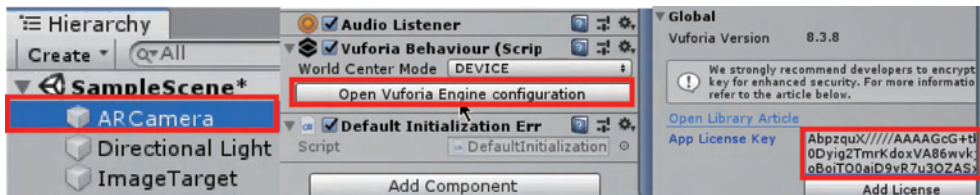


图 5.22 粘贴 License Key

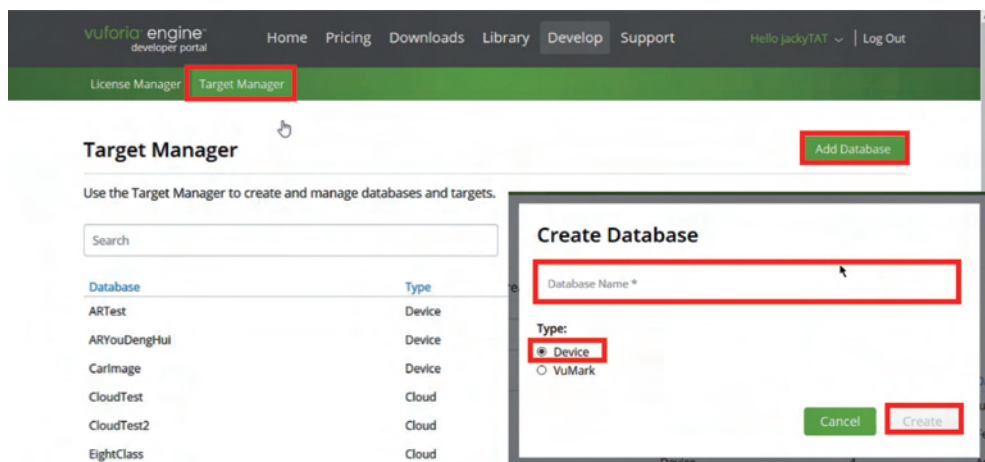


图 5.23 添加识别图数据库

单击 Add Target 按钮添加识别图,弹出窗口中会有 4 种识别图类型供选择,这里选择 Single Image,然后单击 File 后的 Browse 按钮,在出现的路径中选择要使用的识别图。其中识别图必须是 .jpg 格式或 .png 格式,并且不能超过 2MB。还要填写识别图的宽度(注意单位是米)和名字,单击 Add 按钮,如图 5.24 所示。识别图上传完成后,可以看到识别图的评定星级,星级越高,识别图的速度越快,追踪越稳定,可视角度越大。

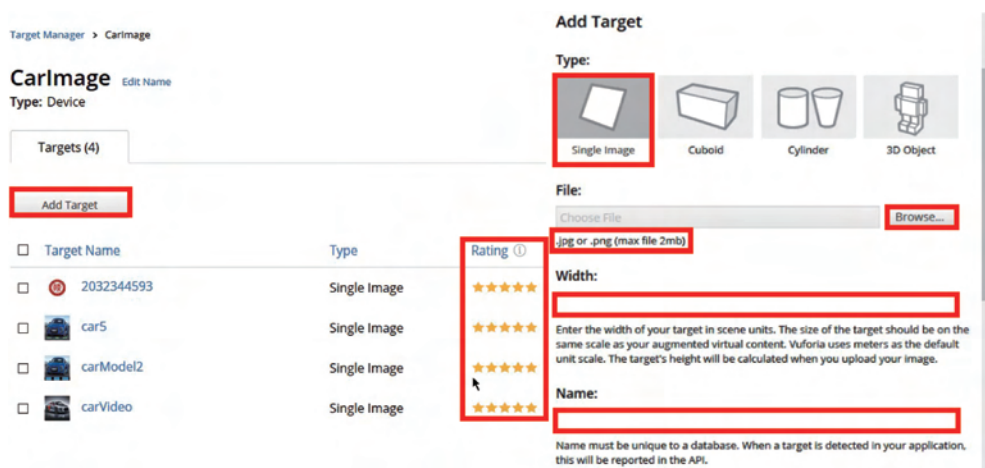


图 5.24 添加识别图

识别图还可以选择 Cuboid(立方体)和 Cylinder(圆柱体),不过需要在 Unity 中创建 Multi Image 或 Cylinder Image 与之对应,如图 5.25 所示。而识别图如果为 3D Object(三维物体),则需要后台工具进行辅助。

在添加的识别图中,勾选识别图 car5,单击 Download Database(1)按钮,如图 5.26 所示,弹出窗口中选择 Unity Editor,单击 Download 下载。

下载后会得到一个 CarImage unitypackage 的资源包,如图 5.27 所示。

将该资源包拖曳到 Project 窗口的 Assets 文件夹,在弹出窗口中单击 Import 按钮,如

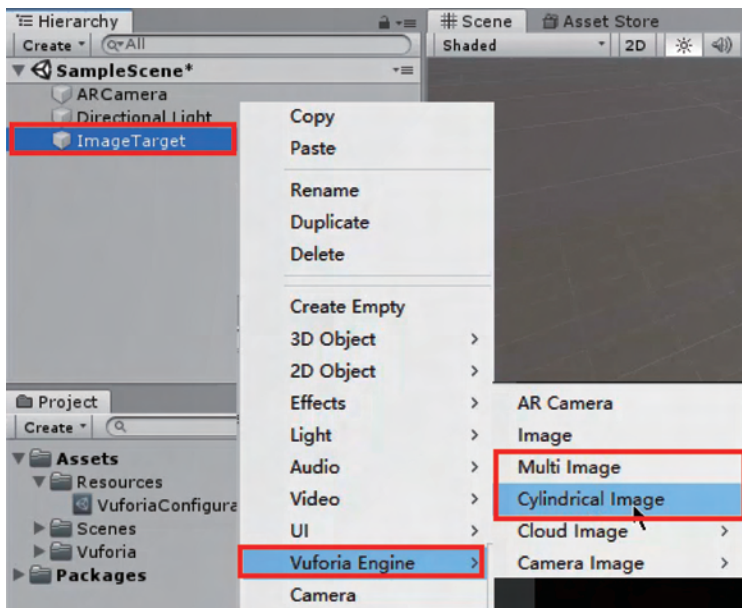


图 5.25 与 Cuboid 和 Cylinder 对应的图像组件

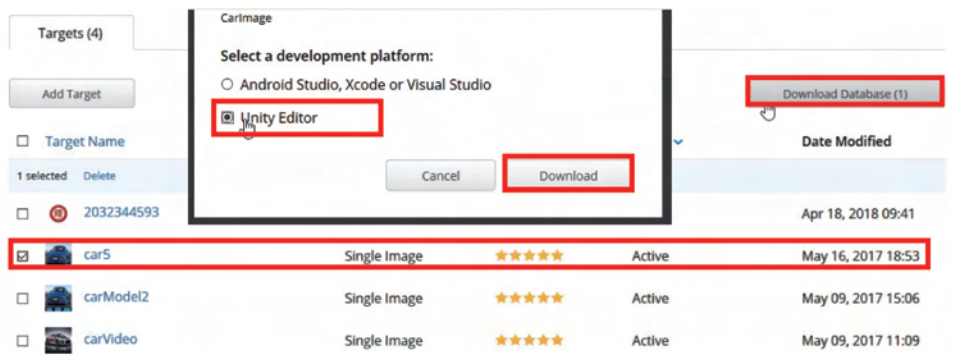


图 5.26 勾选识别图



图 5.27 下载的认可图资源包

图 5.28 所示。

单击 Hierarchy 窗口的 ImageTarget, 在右侧 Inspector 面板对应 Image Target Behaviour 组件下的 Database 参数后单击选择 CarImage, ImageTarget 参数后选择图片 car5, 如图 5.29 所示, 就可以在 Scenes 窗口中看到识别图。

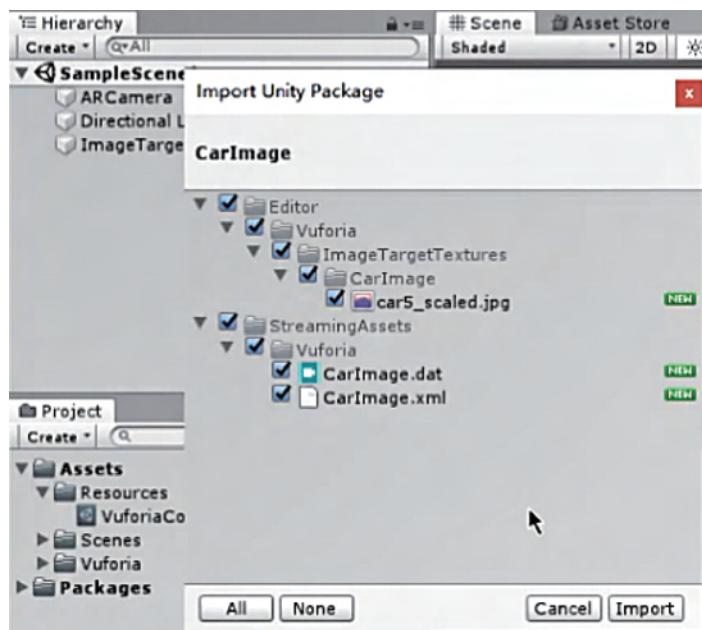


图 5.28 导入识别图资源包

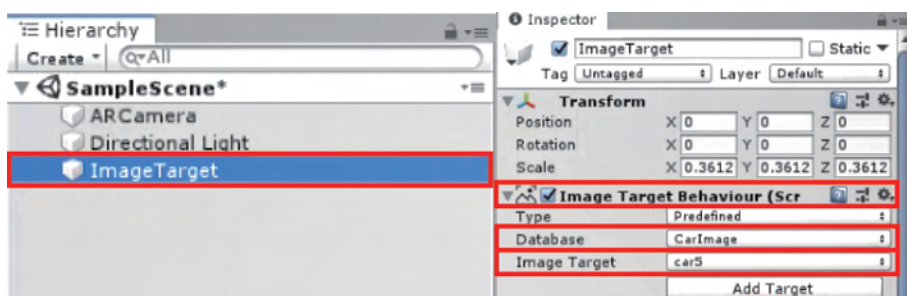


图 5.29 识别图参数设置

5.3.5 创建测试立方体

右击 Hierarchy 窗口的 ImageTarget, 级联菜单中依次选择 3D Object→Cube 命令, 如图 5.30 所示, 即在识别图上创建了一个立方体。



图 5.30 在识别图上创建立方体

单击 Hierarchy 窗口的 ARCamera,在右侧 Inspector 面板中单击 Open Vuforia Engine Configuration 命令,下拉滚动条到面板的下部,在 Webcam 中可以选择台式机或笔记本电脑对应的摄像头。HD Pro Webcam C920 是外接摄像头,Lenovo EasyCamera 是笔记本电脑自带的摄像头,这里选择前者,如图 5.31 所示。

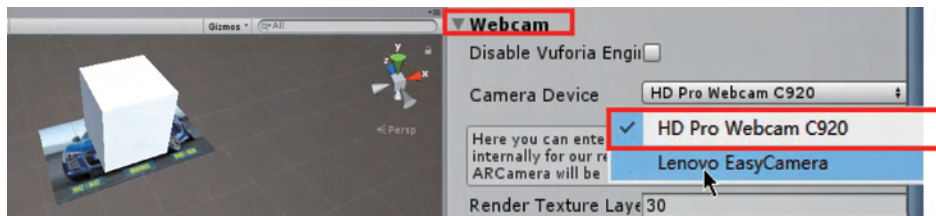


图 5.31 选择外接摄像头

单击 Scenes 窗口的运行按钮,可看到识别图上出现的立方体,如图 5.32 所示。

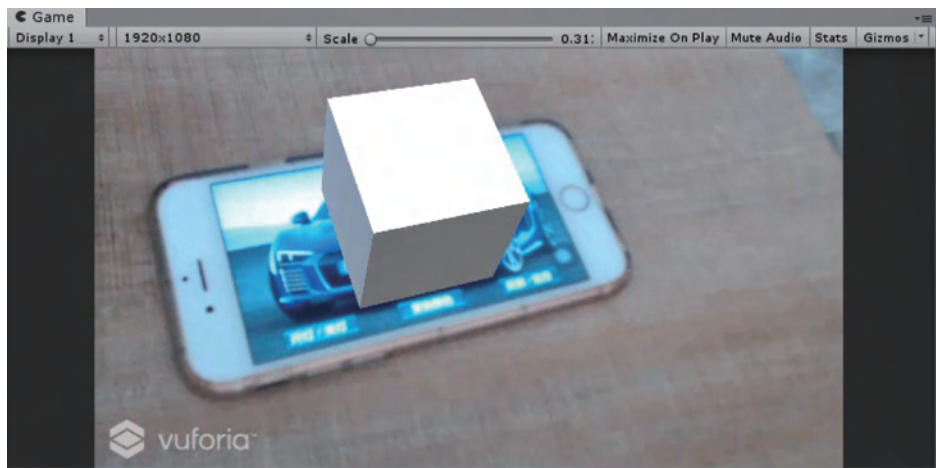


图 5.32 运行效果

5.3.6 导入模型资源

测试完成后,可以把立方体替换成小车。小车模型资源可在官方资源商店 Cartoon SportCar 中下载。首先在 Hierarchy 窗口中将 Cube 删除,然后单击 Scenes 窗口的 Assets Store,在资源商店中单击搜索按钮,在搜索框中键入关键字 Cartoon SportCar,如图 5.33 所示。

找到如图 5.34 所示的小车资源,单击选择下载到本地。

然后单击导入按钮,在弹出窗口中先单击 All 按钮,再单击 Import 按钮,即可将小车资源导入 Unity 中,如图 5.35 所示。

导入过程会出现弹出窗口,提示用户是否更新当前插件的 API,如图 5.36 所示,单击 Go Ahead 按钮。如果不更新,导入后的材质或某些插件的功能可能无法正常使用。

导入完成后,可在 Project 窗口看到导入的汽车资源,这里选择 Cartoon_SportCar_B01,将其拖曳到 Hierarchy 窗口的 ImageTarget 上。在 Scenes 窗口中调整汽车大小,并在 Inspector 面板中调整小车的 Rotation 参数(旋转 90°),使其适合识别图大小,如图 5.37 所示。

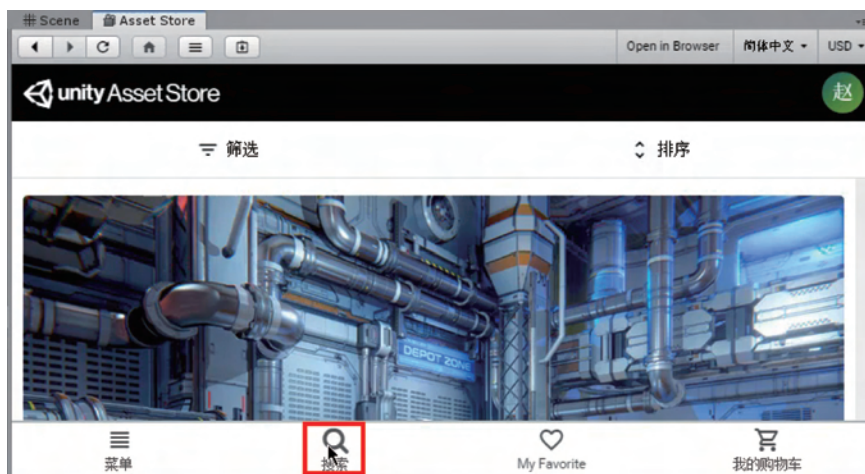


图 5.33 Unity 资源商店

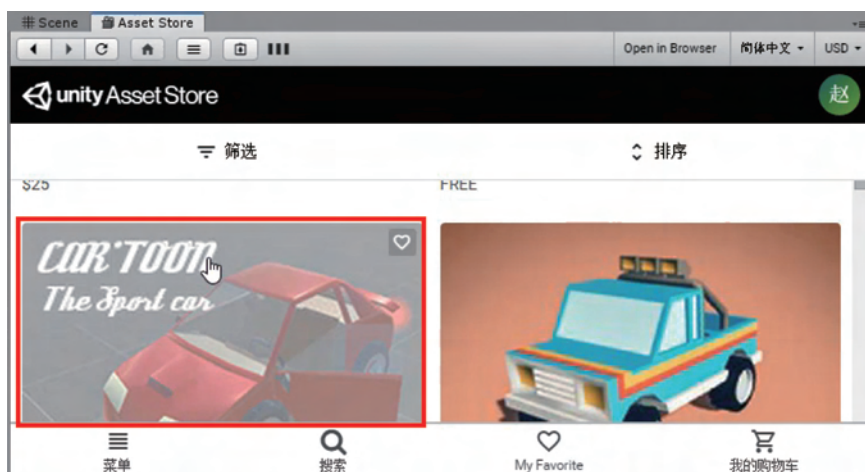


图 5.34 在资源商店中查找小车模型资源



图 5.35 导入小车模型资源

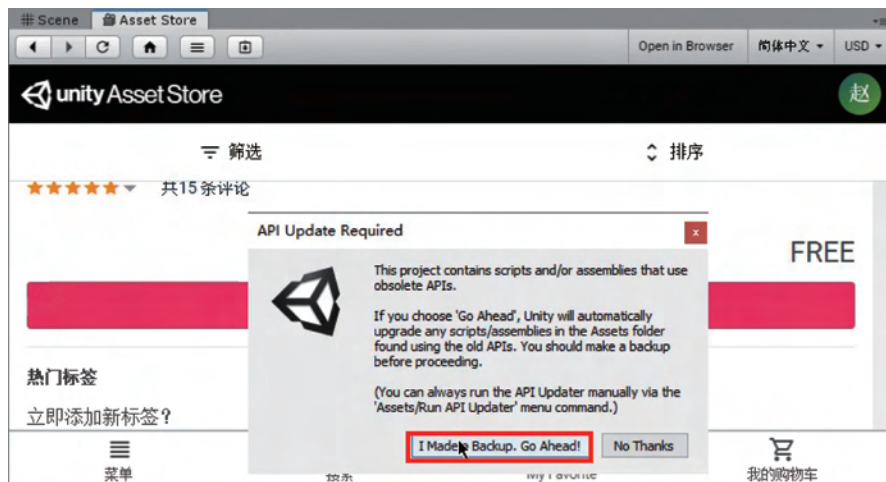


图 5.36 导入小车模型时的 API 更新需求



图 5.37 调整汽车模型大小与角度参数

使用键盘的 Ctrl+S 快捷键保存场景后,单击 Scenes 窗口的运行按钮,可在 Game 窗口中查看运行效果,如图 5.38 所示。



图 5.38 查看运行效果

5.3.7 创建虚拟按钮

接下来通过创建虚拟按钮来实现对小车模型的动画操作。在 Scenes 窗口可以看到识别图底部有 3 个提前使用 Photoshop 软件制作好的按钮,如图 5.39 所示。

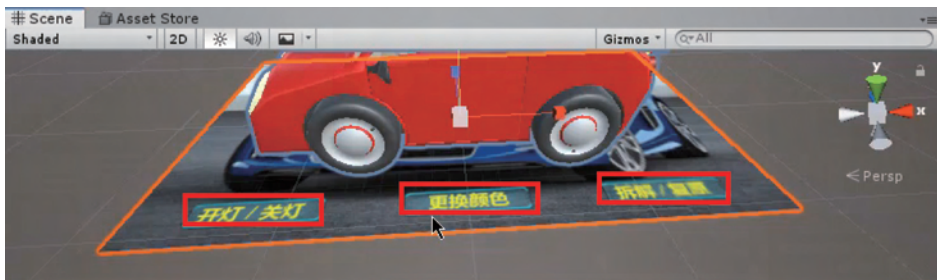


图 5.39 识别图上的按钮

Hierarchy 窗口中单击 ImageTarget, 在 Inspector 面板的 Advanced 组件下单击 Add Virtual Button 按钮, 如图 5.40 所示。

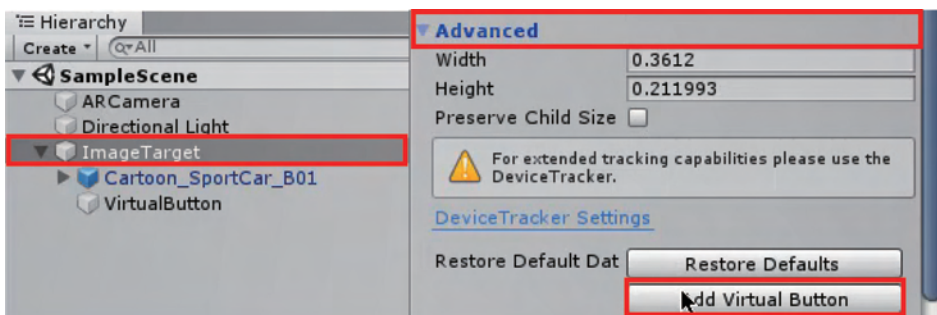


图 5.40 添加虚拟按钮组件

调整虚拟按钮大小,使其与识别图中按钮的大小重叠,如图 5.41 所示。



图 5.41 调整虚拟按钮大小

同样方法,再添加两个虚拟按钮,分别使其与识别图中对应按钮重叠,如图 5.42 所示。



图 5.42 添加并调整其他虚拟按钮

5.3.8 动画制作

虚拟按钮创建好后,还需要添加组件,使用户操作虚拟按钮时,小车可以实现开灯/关灯、更换颜色、拆解/还原等动画效果。首先在 Hierarchy 窗口中单击 Cartoon_SportCar_B01,在 Inspector 面板中将 Animator 组件移除,如图 5.43 所示。

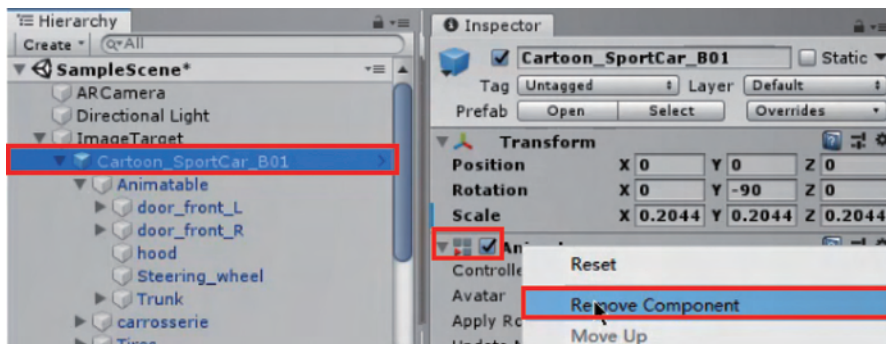


图 5.43 移除 Animator 组件

单击 Add Component 按钮,添加一个 Animation 组件,如图 5.44 所示。

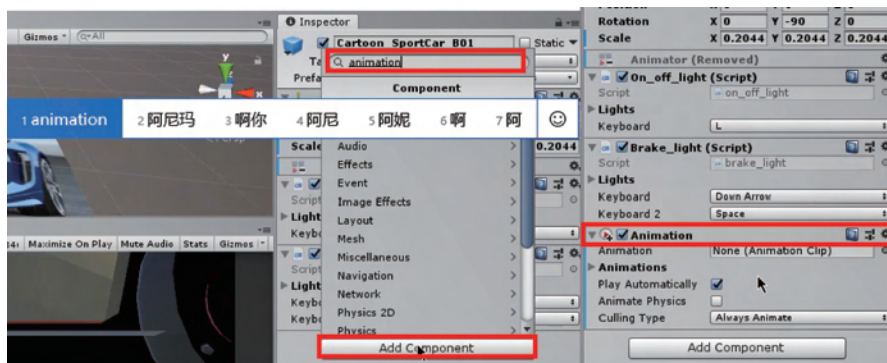


图 5.44 添加 animation 组件

然后在 Window 菜单下选择 Animation,或使用快捷键 Ctrl+6 打开动画编辑窗口,如图 5.45 所示。

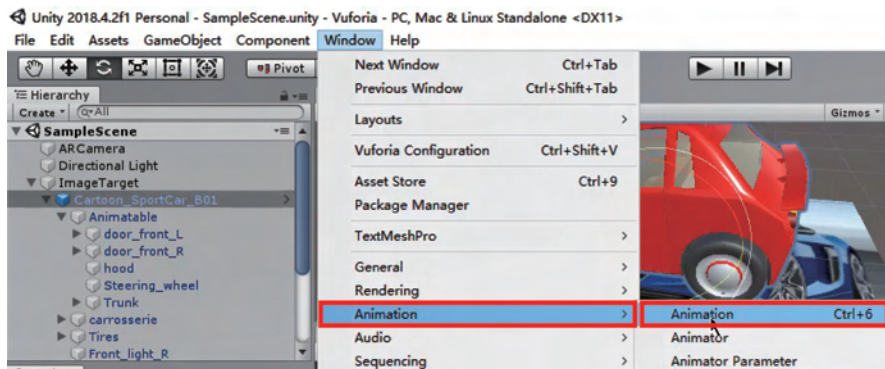


图 5.45 打开动画编辑窗口

在动画编辑窗口单击 Create 按钮,新建一个动画,如图 5.46 所示。

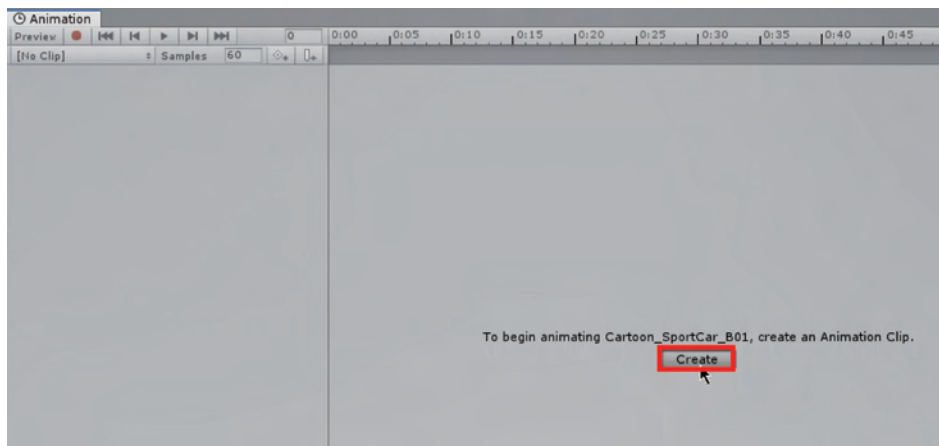


图 5.46 新建一个动画

将动画保存到 Vuforia 路径下的 Assets 文件夹中,为该动画新建一个文件 Animations,将新创建的动画命名为 CarAnimation.anim,保存到 Animations 文件夹中,如图 5.47 所示。

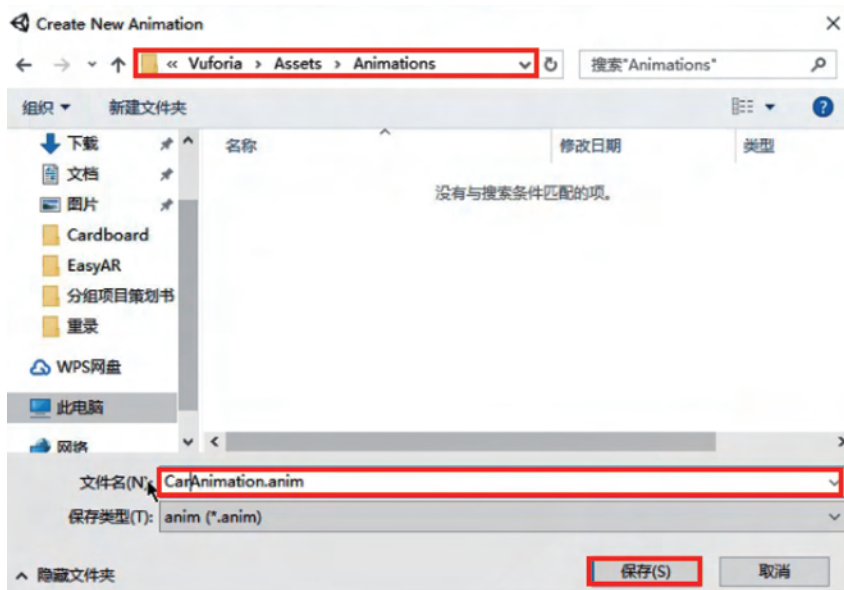


图 5.47 保存动画文件

这里以两侧车门的拆解动画制作为例(模型要支持拆解)。在 Animation 窗口单击 Add Property 按钮,然后在右侧的 Animatable 下找到 door_front_L,单击 door_front_L 的 Transform 组件下 Position 右侧的加号按钮,如图 5.48 所示。

调整 Position.z 的值为-1.5(即左侧车门最大移动位移为 z 轴的-1.5 的位置),并单击红色圆点按钮,如图 5.49 所示,可在 Scene 窗口中看到左侧车门已经拆解出来。

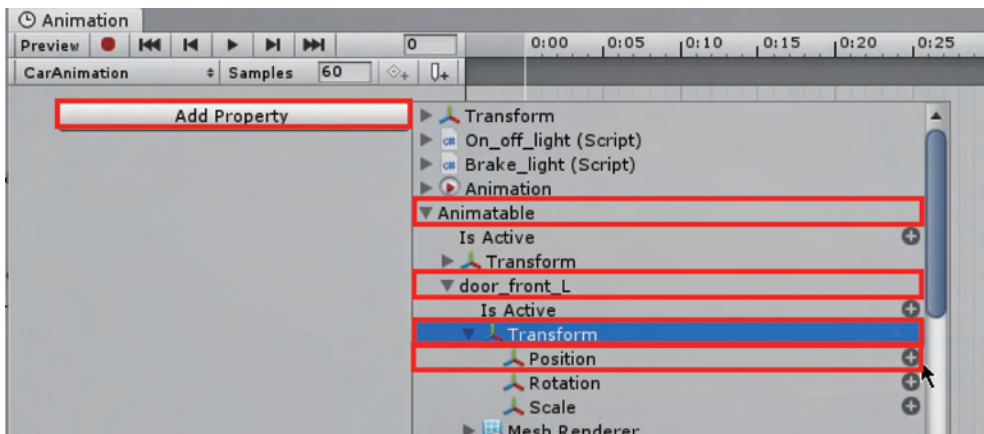


图 5.48 为左侧车门添加动画

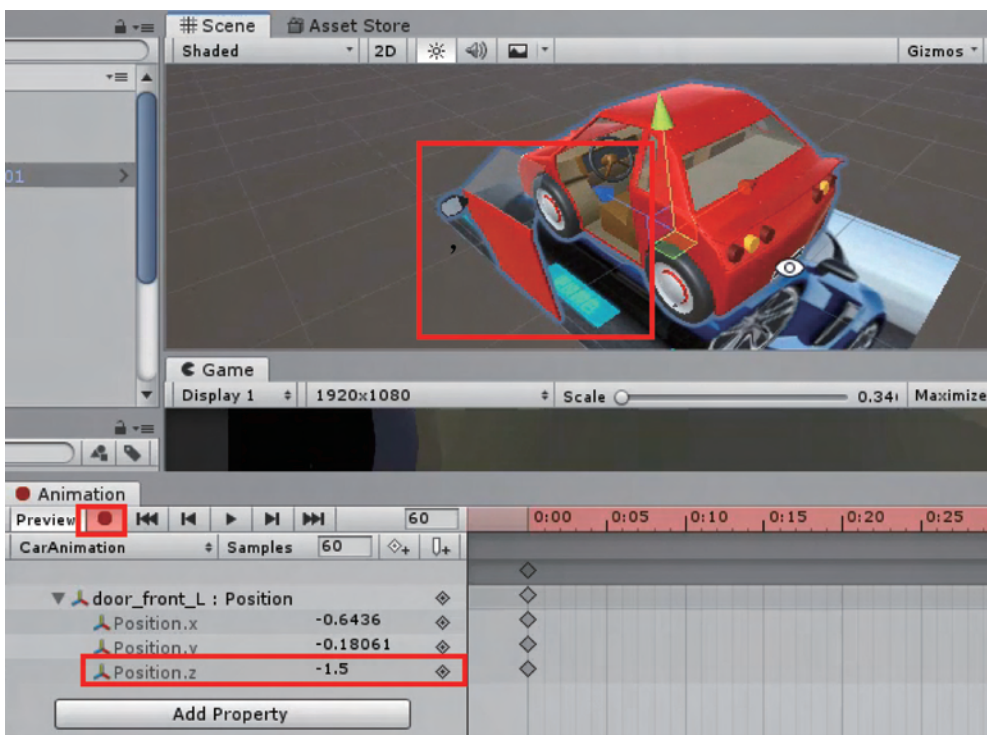


图 5.49 设置左车门最大移动位移

同样的方法做右侧车门的拆解。在 Animation 窗口单击 Add Property 按钮,然后在右侧的 Animatable 下找到 door_front_R,单击 door_front_R 的 Transform 组件下 Position 右侧的加号按钮,如图 5.50 所示。

调整右侧车门 Position.z 的值为-1.5(即右侧车门最大移动位移为 z 轴的 1.5 的位置),如图 5.51 所示。单击红色圆点按钮,并单击三角符号的运行按钮,在 Scene 窗口中查看车门拆解的动画。

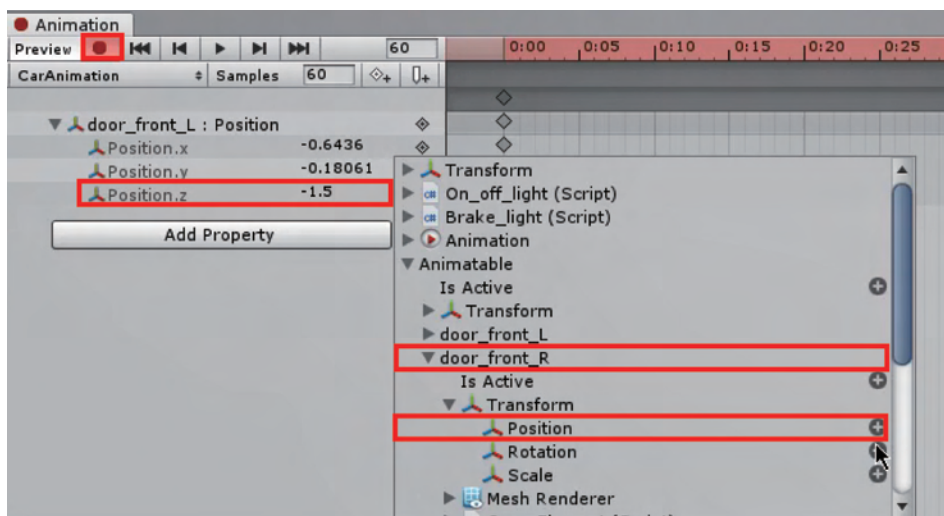


图 5.50 设置右车门最大移动位移

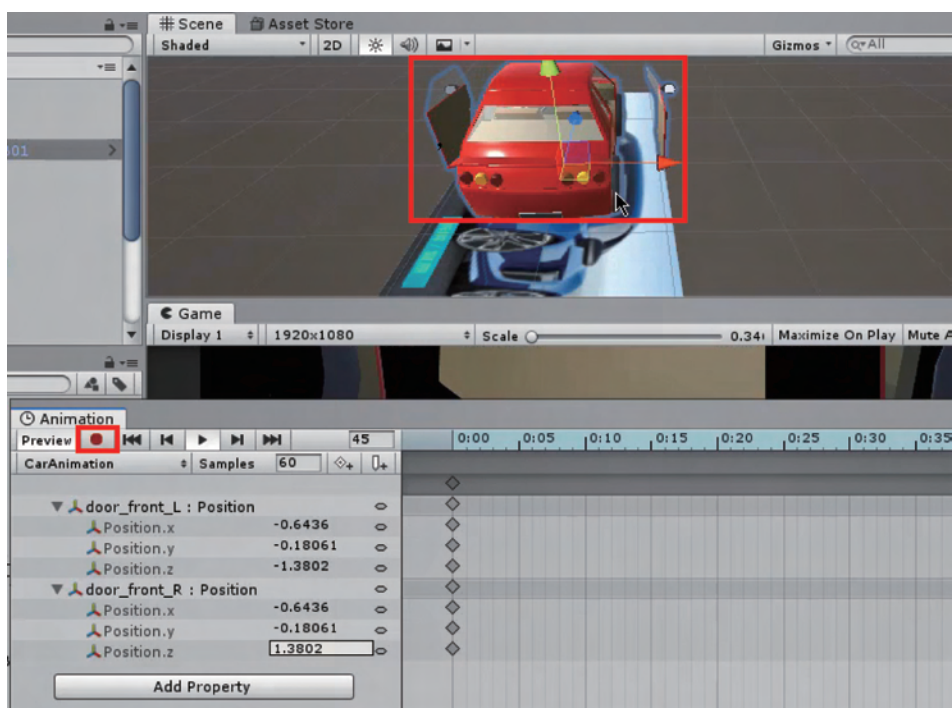


图 5.51 查看车门拆解动画

其他部位的拆解做法类似,读者可以自行进行车引擎盖、后备厢等的拆解动画制作。接下来实现单击虚拟按钮完成动画播放功能。在 Hierarchy 窗口中选择 Cartoon_SportCar_B01,将 Project 窗口中刚才制作的动画文件 CarAnimation 拖曳到右侧 Inspector 面板的 Animation 后的方框内,并且把 Play Automatically 后的勾选取消,即不设置自动播放,如图 5.52 所示。

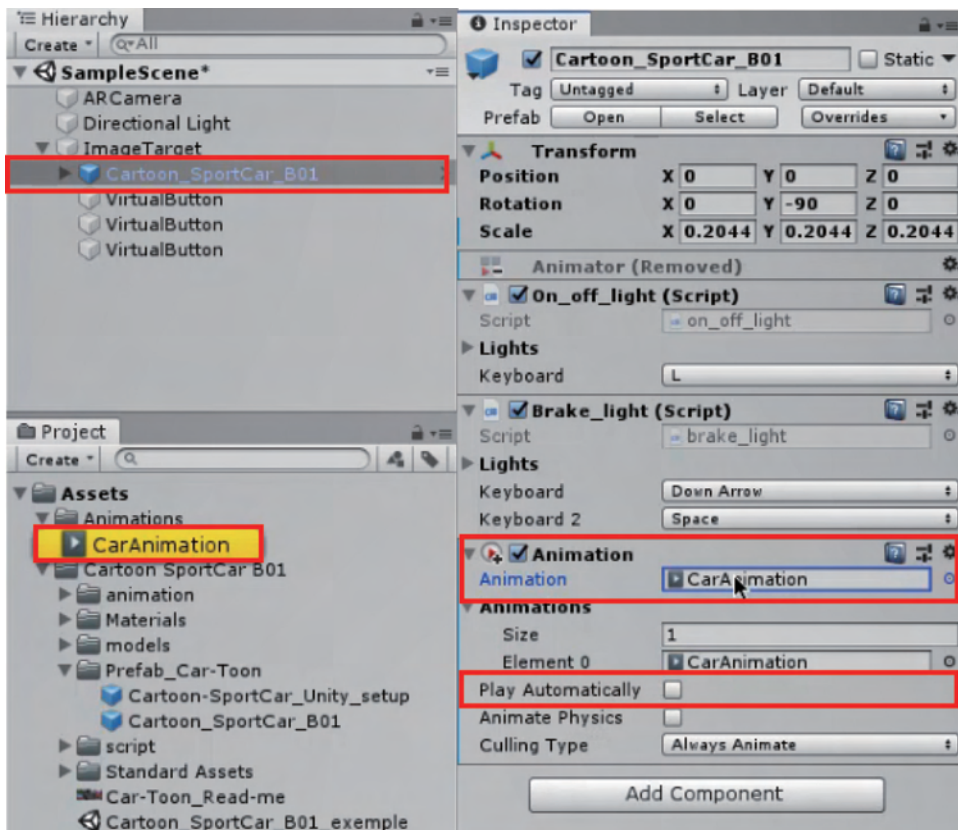


图 5.52 动画播放设置

单击 Project 窗口 Animations 下的 CarAnimation, 确认 Inspector 面板中的 Wrap Mode 为 Default, 即只播放 1 次, 如图 5.53 所示。

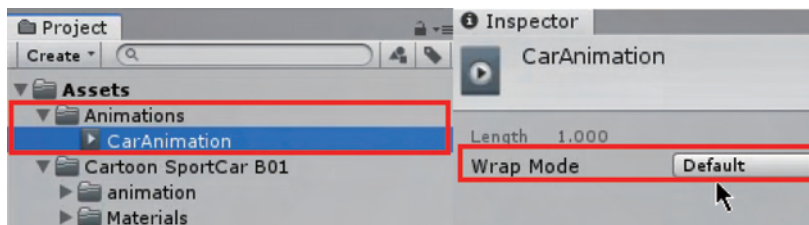
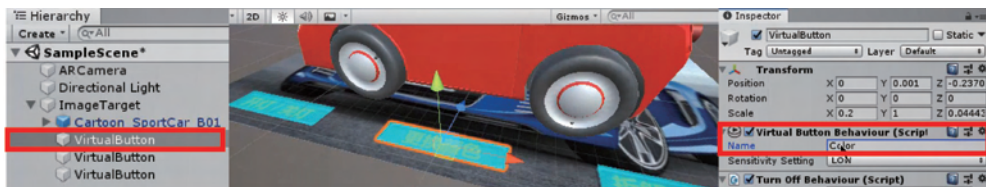


图 5.53 动画播放次数设置

单击 Hierarchy 窗口中的 VirtualButton 按钮, 该按钮对应“更换颜色”功能, 因此在 Inspector 面板中 Virtual Button Behaviour 组件下 Name 后修改名称为 Color。同理给拆解/还原按钮修改名称为 Boom, 给开灯/关灯按钮修改名称 Light, 如图 5.54 所示。

5.3.9 添加脚本

单击 Hierarchy 窗口的 ImageTarget, 在 Inspector 面板单击 Add Component 按钮添加脚本, 脚本名为 MyVirtualButtonBehaviour。脚本加载完成后, 在 Inspector 面板双击如图 5.55 所示的脚本文件名位置, 即可打开脚本。



(a) 修改更改颜色虚拟按钮名称



(b) 修改拆解/还原虚拟按钮名称



(c) 修改开灯/关灯按钮名称

图 5.54 修改设置

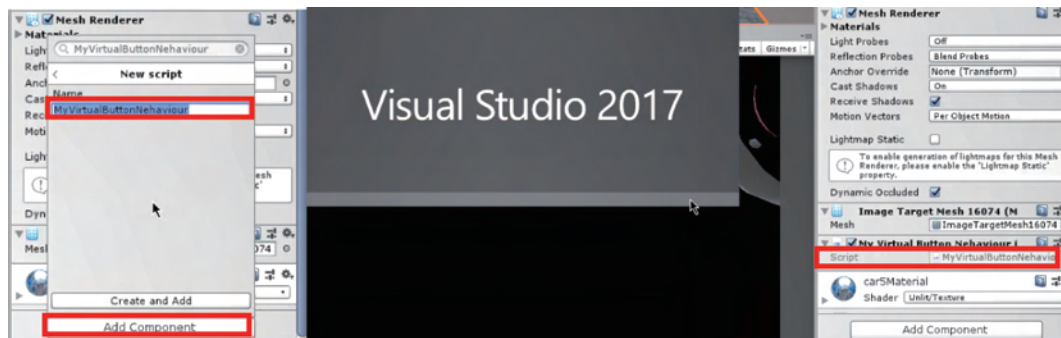


图 5.55 创建脚本

虚拟按钮是 Vuforia 的一个功能,在脚本中需要继承一个虚拟按钮接口。首先,添加一个引用 Vuforia 的命名空间,代码如下:

```
using Vuforia;
```

然后在 MonoBehaviour 后面加一个接口的继承:单击接口 IVirtualButtonEventHandler 名称,在下拉功能中选择实现接口,如图 5.56 所示。

添加实现接口后,即得到如图 5.57 所示的按钮按下和按钮释放两个方法,将其剪贴放到代码最后。

初始化虚拟按钮,代码如图 5.58 所示。

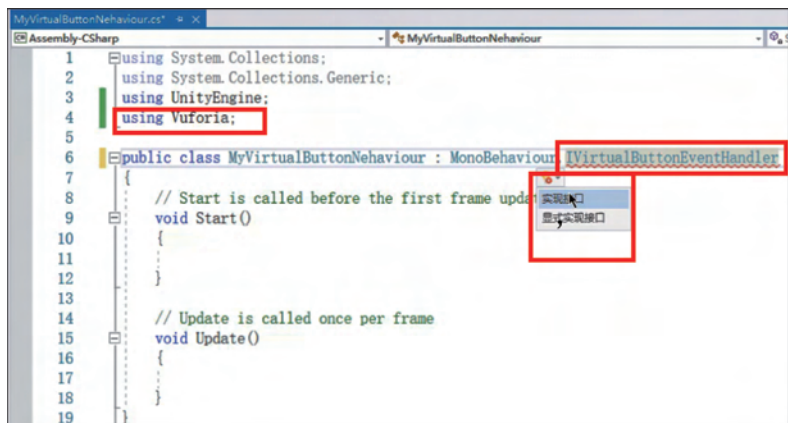


图 5.56 添加实现接口

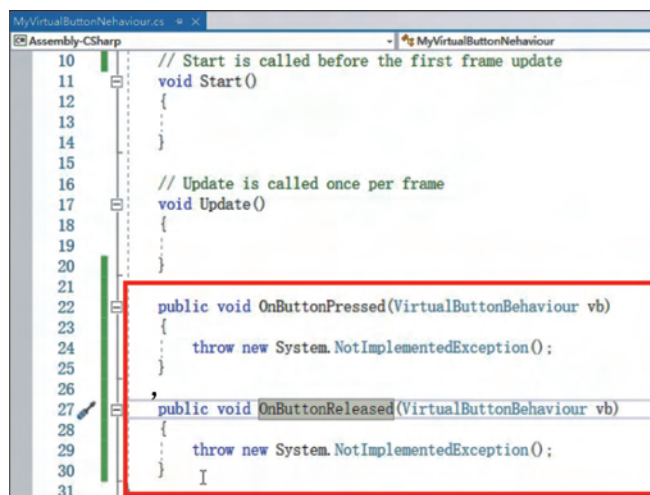


图 5.57 添加接口后得到的两种方法

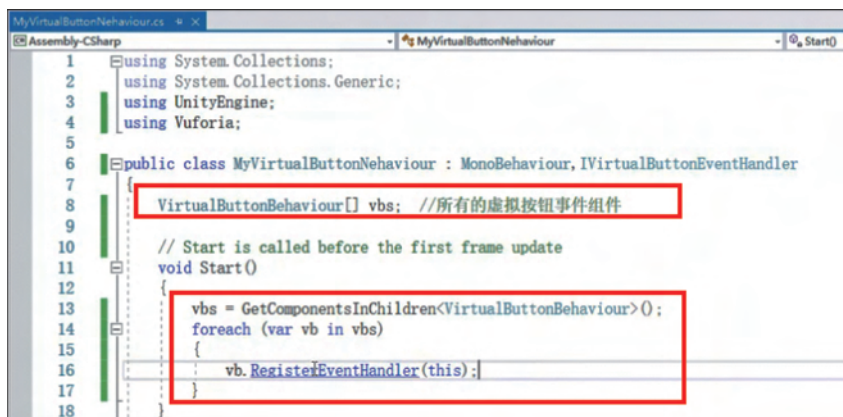


图 5.58 初始化虚拟按钮

添加按钮按下脚本代码如图 5.59 所示。

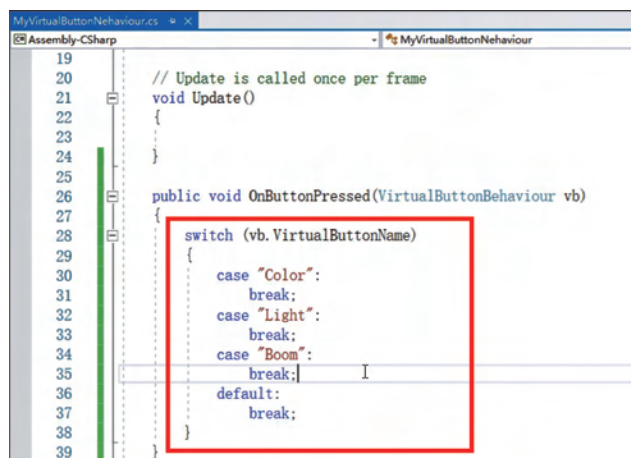


图 5.59 添加按钮按下脚本代码

添加按钮松开脚本代码如图 5.60 所示。

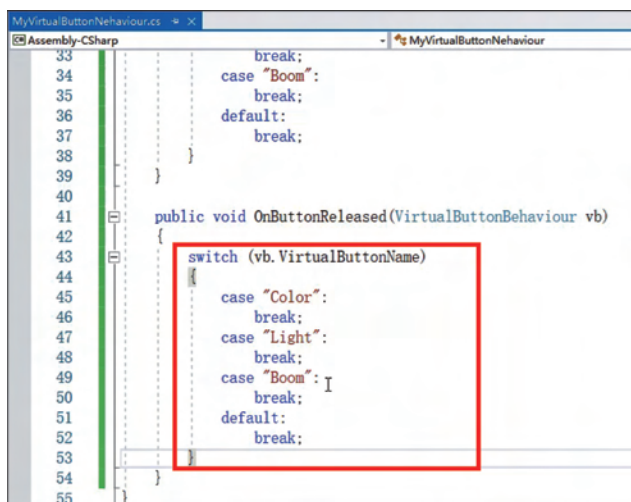


图 5.60 添加按钮松开脚本代码

声明一个拆解/还原动画组件,代码如图 5.61 所示。

然后在 Unity 中给虚拟按钮赋值。将 Hierarchy 窗口中的 Cartoon_SportCar_B01 拖曳到 Inspector 窗口 My Virtual Button Nehaviour 下的 Car Anim 组件后的方框内,如图 5.62 所示。

添加脚本使鼠标按下时播放 carAnim 组件,动画从第 0 帧开始播放,如图 5.63 所示。

还原相当于拆解动画的倒放,因此播放速度为-1,代码如图 5.64 所示。

运行查看效果,当手放在虚拟按钮上时,拆解动画开始播放,手移开后汽车还原,如图 5.65 所示。

其他像引擎盖、车轮等部件的拆解和还原类似,用户可以自行制作。

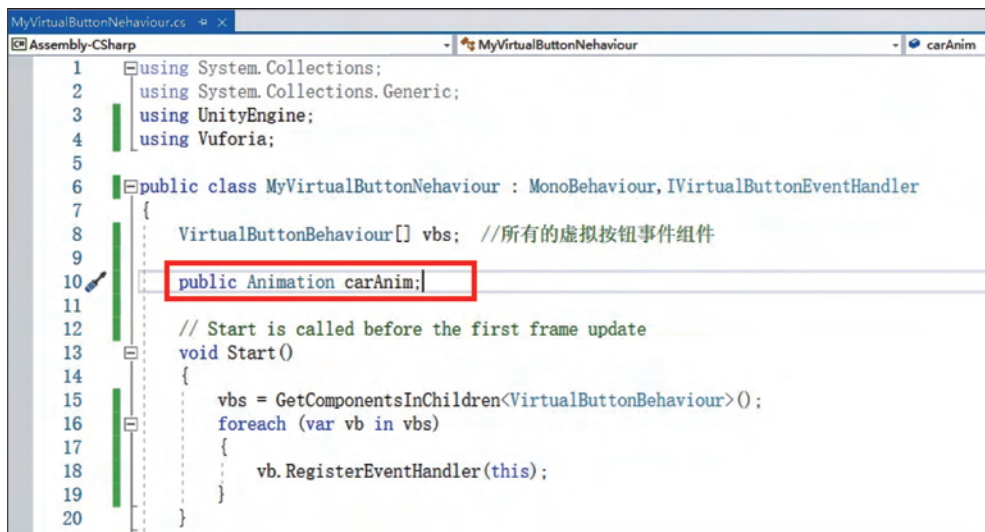


图 5.61 声明一个拆解/还原动画组件

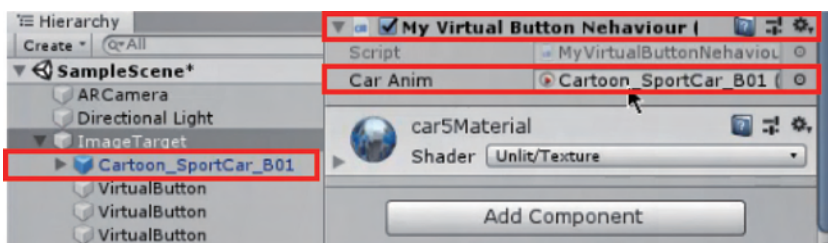


图 5.62 给虚拟按钮赋值

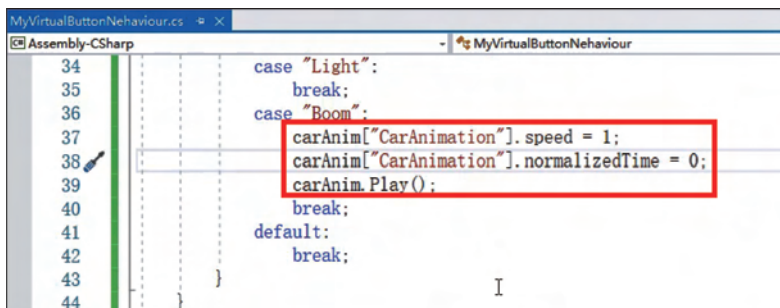


图 5.63 添加拆解按钮播放动画脚本

5.3.10 导出发布

将制作好的动画场景保存后可以导出发布到终端。首先,在 Unity 的 File 菜单中选择 Build Settings,弹出界面中切换到 Android 平台。其次,单击平台下方的 Play Settings 按钮,在右侧 Inspector 面板的 Other Settings 组件下填写对应的包名和 App 版本号,如图 5.66 所示。最后,单击弹出窗口的 Add Open Scenes 按钮,添加要导出的场景即可。

```

43
44     public void OnButtonReleased(VirtualButtonBehaviour vb)
45     {
46         switch (vb.VirtualButtonName)
47         {
48             case "Color":
49                 break;
50             case "Light":
51                 break;
52             case "Boom":
53                 carAnim["CarAnimation"].speed = -1;
54                 carAnim["CarAnimation"].normalizedTime = 1;
55                 carAnim.Play();
56                 break;
57             default:
58                 break;
59         }
60     }

```

图 5.64 添加释放按钮播放动画脚本



图 5.65 查看拆解/还原按钮动画播放效果

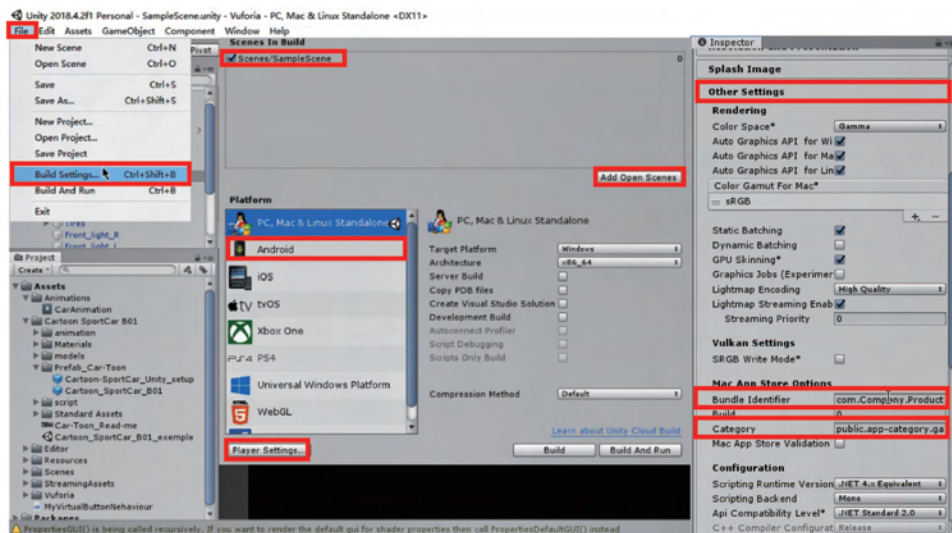


图 5.66 导出发布

习题

一、填空题

1. Vuforia 是一款能将现实世界物体与虚拟物品进行互动体验的_____开发平台。
2. Vuforia 通过 Unity 游戏引擎扩展提供了_____,_____, Objective-C 和 .NET 语言的应用程序编程接口,能够同时支持_____,_____的原生开发。
3. Vuforia 的基础功能主要有_____,圆柱体识别、立方体识别、_____,_____,_____,云识别、智能地形和物体识别。

二、选择题

1. Vuforia SDK 基础功能不包括()。
A. 图片识别
B. 自定义识别
C. 文字识别
D. 虚拟按钮
E. 云识别
F. 人像识别
G. 智能地形
2. 以下()不属于 Vuforia 项目开发的基本步骤。
A. 获取 SDK
B. 创建 License Key
C. 制作模型
D. 导入模型
E. 制作动画
F. 导出发布

三、简答题

1. 简单介绍 Vuforia。
2. 简述使用 Vuforia SDK 进行项目开发的主要步骤。