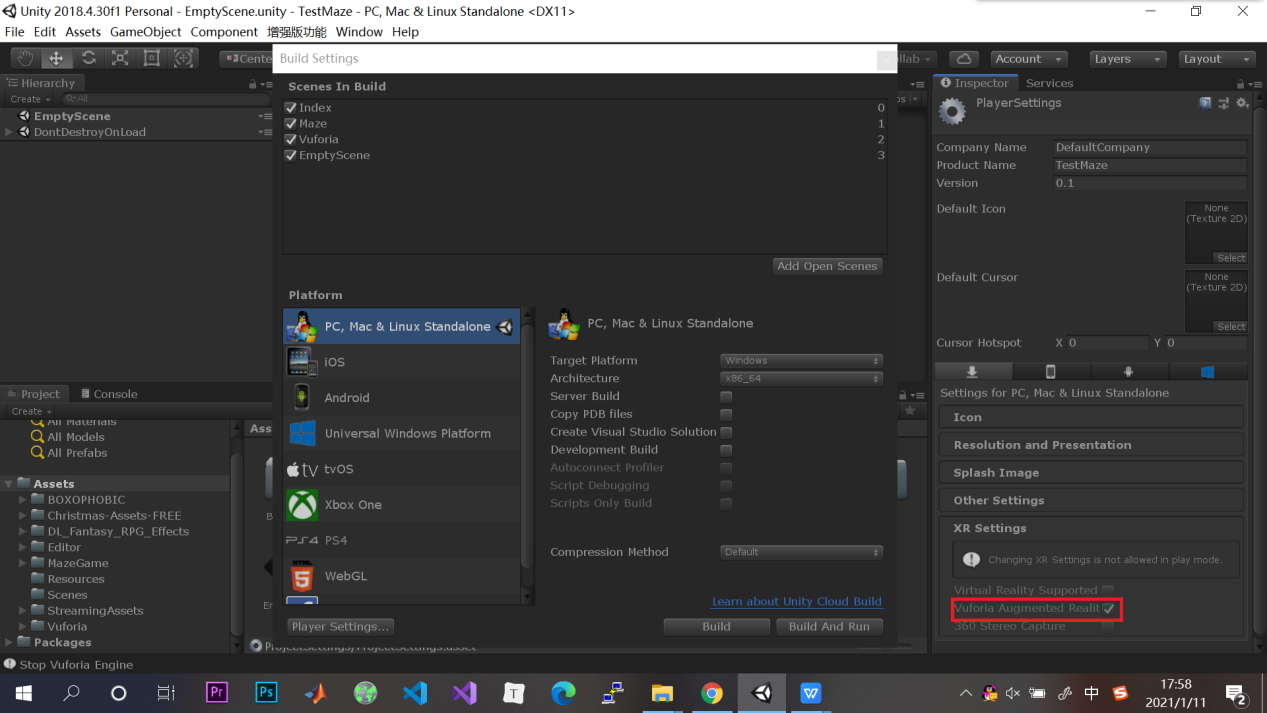
# 设计开发说明

\* 注意，在本地打开资源包时，需要Unity先下载Vuforia包，并在Player Settings中勾选Vuforia Augmented Reality选项。

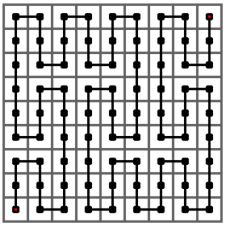
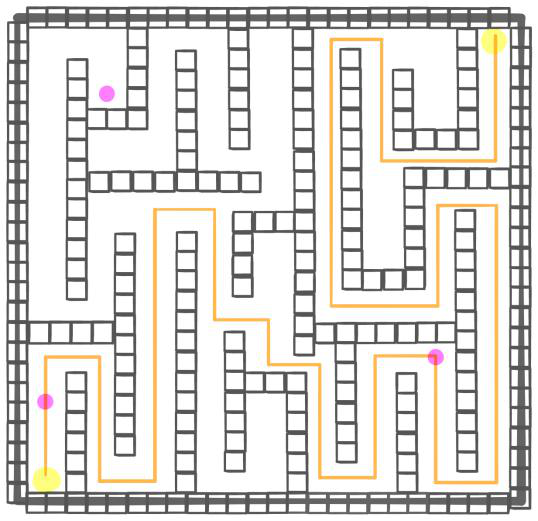


## 游戏简介

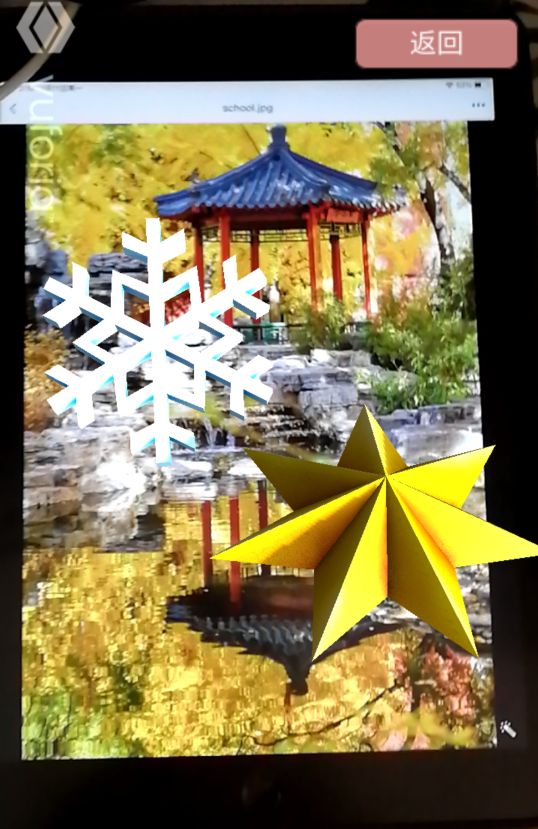
本游戏是一款基于Vuforia的益智类游戏，玩家可以以第一人称视角在迷宫中移动并收集物品。当玩家碰撞物品时，游戏切换到AR场景，将游戏包提供的图片放在摄像头前，物品会以立体形态显示，即可完成收集。当玩家找到终点，游戏结束，并统计用时和收集到的物品个数。

## 设计灵感

1. 迷宫的地图灵感来源于三阶皮亚诺曲线，如左下图。修改道路并添加收集物品后的迷宫如右下图，橙色为正确路线，红点为物品位置，黄点为起点和终点。

1. 收集物品的玩法灵感来源于最近学习的Vuforia AR，如下图。下载school.jpg在其他设备中打开，对准摄像头后即可收集到刚刚碰撞的物体。

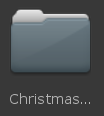


## 用到的游戏资源包

1. 天空盒子BOXOPHOBIC



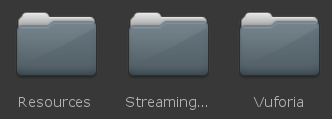
1. 圣诞物品Christmas-Assets-FREE



1. 粒子特效DL\_Fantasy\_RPG\_Effects

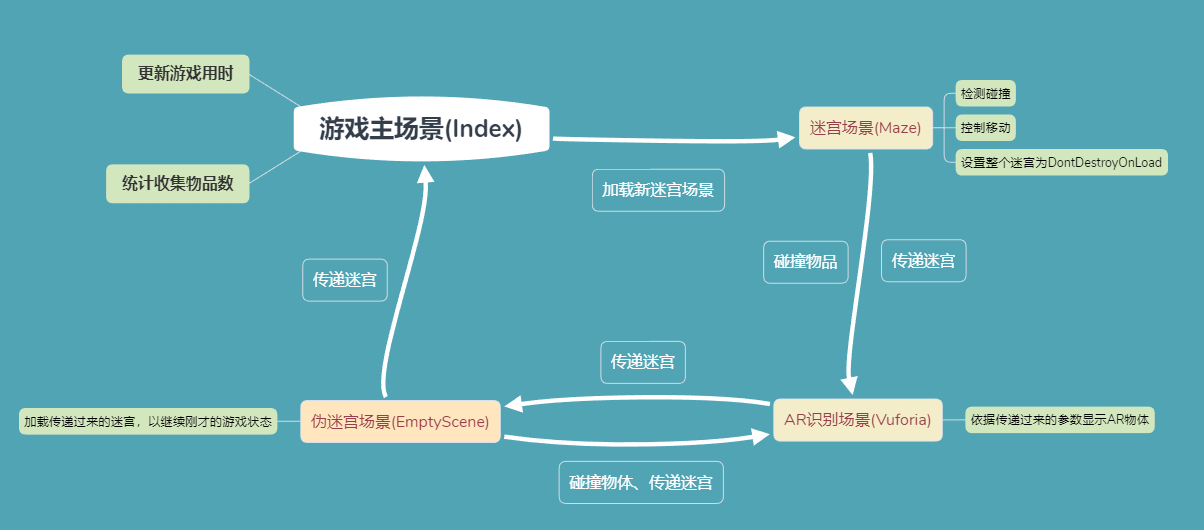


1. Vuforia相关资源包以及图片数据库



## 游戏架构

#### 游戏场景

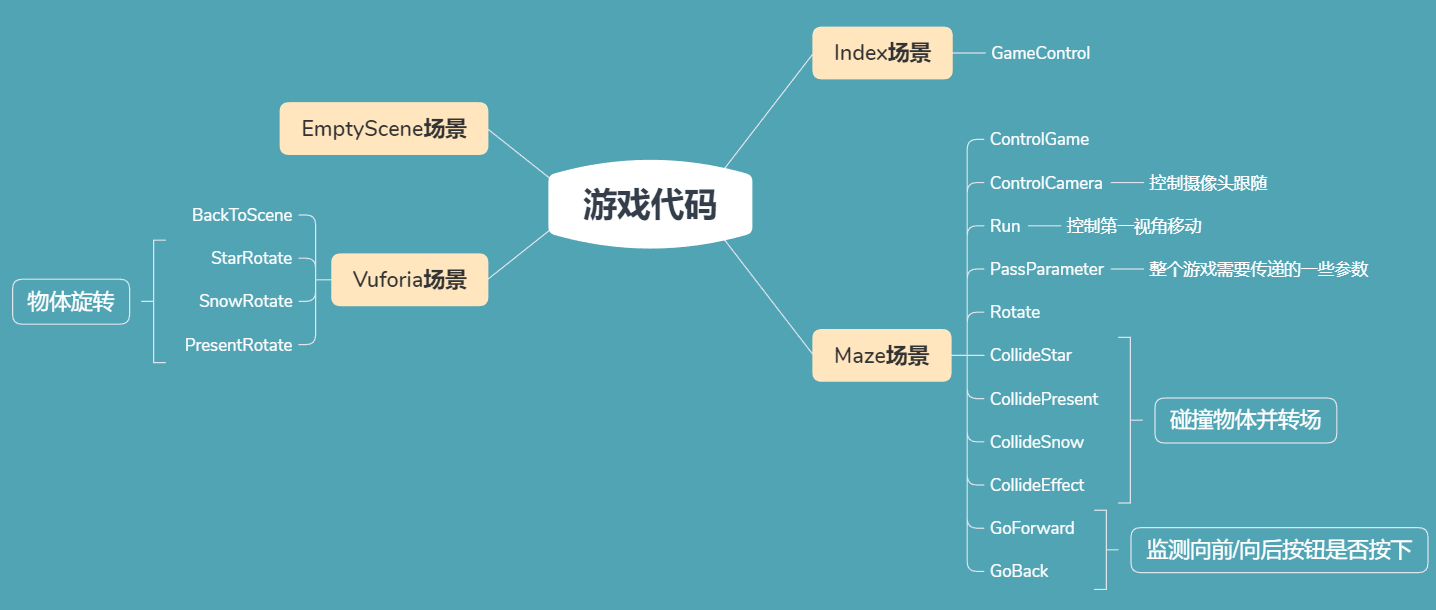


#### 项目目录

#### 2]D9OK%8}2N(@VZMB]B%]8W

如图，MazeGame中是游戏的代码和资源，其他文件夹是Unity官网下载并导入的免费资源。

#### 代码文件



1. Index场景
   1. GameControl
      1. 点击“开始游戏”按钮转场进入Maze场景
      2. 点击“退出游戏”按钮退出游戏
      3. 更新游戏结束传递过来的参数（游戏时长、收集物品数）
2. Maze场景
   1. ControlGame
      1. 点击“重新开始”按钮返回Index场景
      2. 点击“退出游戏”按钮直接退出游戏
      3. 由于整个迷宫场景保存在一个名为Parameter的GameObject之下，因此将Parameter设置为不销毁状态，即可保存游戏状态
   2. ControlCamera
      1. 固定主相机的位置为第一人称视角
      2. 判断当前游戏平台，若当前平台为Android端（即触摸屏），控制手指在屏幕1/3以外处可以旋转摄像机视角；若平台为PC端，用鼠标控制视角
   3. Run
      1. 通过两个float类型的参数h和v控制移动方向和速度，当平台为PC端时，h和v由键盘赋值，android端通过GoForward和GoBack监测按钮传递过来的参数进行赋值。
   4. PassParameter
      1. 定义整个游戏需要传递的一些参数，如：游戏时间、获得物品个数、每个物品是否已被获得的状态、是否成功通关
      2. 计时
   5. Rotate
      1. 控制需要被收集的几个物体自己旋转
   6. CollideStar、CollideSnow、CollidePresent
      1. 物体被碰撞时，收集物品数+1、标记当前物品已获取
      2. 销毁场景中的该物体
      3. 设置向前/向后的按钮的状态为并未按下
   7. CollideEffect
      1. 碰撞到终点的粒子特效时，标记游戏通关
      2. 返回Index场景
   8. GoForward、GoBack
      1. 定义一个静态的bool变量isDown以标记按钮状态，长按按钮为true，抬起按钮为false
      2. 长按向前按钮时，h=0、v=1；长按向后按钮时，h=0、v=-1，以便Run文件中控制玩家移动
3. Vuforia场景
   1. BackToScene
      1. 由于迷宫场景在Parameter中被传递过来，因此点击“返回”按钮前，先将Maze场景的UI按钮先禁用以免干扰
      2. 点击“返回”按钮时，将Parameter的UI组件启用，将其传递到EmptyScene场景中并转场，实现继续游戏的效果
   2. StarRotate、SnowRotate、PresentRotate
      1. 控制物体不同的旋转方式
      2. 根据PassParameter中的参数，将未收集到的物体影藏，已收集到的物体显示

## **遇到的问题及解决方法**

#### 保存游戏状态

由于LoadScene方法会将原场景删除，打开AR相机后再次返回迷宫场景又会从头开始。因此将整个迷宫场景放在一个名为Parameter的空游戏对象下，作为它的子对象。通过DontDestroyOnLoad(Parameter)，转场时会把它传递到其他场景并且状态保持不变。当需要返回迷宫场景继续游戏时，将它传递到一个空场景中打开，看起来就像返回到了原来的游戏状态。

返回场景时打开的是空场景，而不是Maze场景的原因是：Maze场景本身打开就会有迷宫，传递过来的迷宫再加载进来就会有两个迷宫场景造成冲突。

#### **不同平台使用不同游戏方式**

电脑使用键盘而手机使用触屏，如果都用Input.GetMouseButton(0)来判断是否点击会产生问题：电脑鼠标会一直在屏幕上，但手机触屏手指会离开屏幕，当手指离开屏幕时GetMouseButton就认为鼠标的位置一直在点击刚刚最后离开的位置，导致视角失控。

因此要判断不同平台，PC端用Input.GetMouseButton(0)判断，Android端用Input.touchCount > 0 &&Input.GetTouch(0).phase == TouchPhase.Moved来判断。

1. **触屏控制导致的问题**

游戏在手机端左手点击按键控制前后移动，右手长按进行视角旋转，但是代码中只要有一个手指长按就可以转动视角，因此当点击按钮时也会被识别成在转视角。

解决方法是，转动视角时设置在屏幕宽度的1/3以外长按才有效，以免左右手发生冲突。

同时将单指和多指分开判断，单指时手指的位置是Input.GetTouch(0)，多指时手指的位置是Input.GetTouch(1)。

1. **碰撞物体导致按钮一直处于按下状态**

当一直点击着前进按钮碰撞到物体后，会直接切换场景，而按钮一直处于被按下状态，即使返回场景后并未点击屏幕，物体也会一直移动。因此，在切换场景前要手动设置按钮状态为false。

1. **横竖屏转换**

PC拍照适合横屏，手机拍照时适合竖屏，因此在导出apk前，我直接将vuforia场景中的按钮旋转成竖向的，并没有进行游戏中动态的屏幕旋转。

1. **UI控件屏幕自适应**

参考于博客：<https://www.cnblogs.com/czw52460183/p/10956665.html>

## **尚未解决的问题**

1. **Vuforia AR相机不支持PC端**

这个我也是整个游戏做完了才知道，很无奈但也没办法，PC端直接点击返回无视这个功能吧。其他能够支持AR相机的环境我也试了，没有Mac电脑没法打包IOS、虚拟机上打包实在太繁琐了，而且中国地区手机号也不支持注册成为IOS开发者；Universal Windows Platform平台，如果不发布在Windows应用商店的话，打包证书有效期只有一天，而且只能在本地用powershell打包并下载，如果发布到应用商店还要交费。

1. **Android端卡顿**

当左手先点击按钮移动时，右手再滑动视角时有事会识别不到。

## **可优化的地方**

1. 通关时记录最短用时，以及播放通关特效、提示弹窗等功能。
2. 迷宫随机生成，无限关卡。可以将一阶皮亚诺曲线作为新手引导，然后基于三阶皮亚诺曲线设置障碍物和通路。

## **参考文献**

1. Unity 开发AR之 Vuforia：

<https://www.cnblogs.com/chinarbolg/p/9601441.html>

1. 高通Vuforia识别3D物体，发布PC程序的那些事：

<https://blog.csdn.net/gheartsea/article/details/85115305>

1. unity DontDestroyOnLoad使用详解：

<https://blog.csdn.net/qq_40666620/article/details/107951560>

1. Unity屏幕自适应原理：

<https://www.cnblogs.com/czw52460183/p/10956665.html>

1. Unity 触摸屏旋转和缩放：

<https://blog.csdn.net/qq_39097425/article/details/83271993?utm_medium=distribute.pc_relevant.none-task-blog-OPENSEARCH-1.control&depth_1-utm_source=distribute.pc_relevant.none-task-blog-OPENSEARCH-1.control>