

基于 unity3D 的跑酷游戏的设计与实现

崔雪杨 键

四川大学锦城学院 四川 成都 611731

【摘要】目前虚拟现实技术与智能手机越来越紧密结合,本文主要基于Unity3D跑酷游戏的设计以及实现,对游戏里面主角、地形、障碍物等关键技术进行了介绍。虚拟现实跑酷游戏可以将玩家带入更逼真的世界,戴上Gear VR还可以进行体育锻炼,益智益体。

【关键词】Unity3D; 跑酷游戏; VR

1 引言

游戏开发在各国都有很大的技术和游戏交流圈,每当有游戏发布或预热宣传,都会引起一次不小的讨论,这会涉及到游戏开发技术的运用和硬件技术突破 [1]。智能手机非常普及的同时,年轻人的最爱之一——游戏,使得我们身边的低头族越来越多。本文将虚拟现实与跑酷结合起来,开发了第三人称视觉的VR跑酷游戏,跑酷游戏本身具有实时性和交互性,现在给玩家又多了沉浸性。近年来移动通信网络技术飞速发展,越来越多的人通过智能手机利用碎片化的时间来玩休闲类游戏,寻找乐趣放松心情,于是手游行业得到了快速的发展,其中跑酷类游戏因为其诸多优点而深受大众喜爱 [2]。

现在市面上有疯狂动物园、神庙逃亡、汤姆猫炫跑等等的第三人称视角的跑酷游戏,本游戏在此基础上添加了虚拟现实交互技术。

2 Unity 3D 引擎和 Gear VR 概述

Unity Technologies 开发了 Unity3D, Unity3D 是一个全面整合的专业游戏类引擎。Unity 是世界领先的实时 3D 开发平台,编辑和发布具有强大的跨平台性。据 Unity 官方最新数据统计,全球所有平台 (steam/pc/ 控制台 / 手机) 上 50% 的游戏、60% 的 AR/VR 内容均由 Unity 提供支持 [3]。

Gear VR 头盔有三个部分“信息库”、“商店”和主界面,该产品沉浸感强、APP 功能强大,支持三星手机,比其他的 VR 设备便宜很多。Gear VR 不会很笨重、可以戴在头上,佩戴也很舒适,适合跑酷游戏。

3 游戏开发前的准备工作

3.1 游戏类型

本第三人称跑酷类游戏开发工具是 Unity3D, 其中涉及了 C#, 通过三星智能手机与 Gear VR 实现。

3.2 运行的平台

本文是以 Unity2018.3.9f1 开发的跑酷游戏, 提前安装 Unity 软件和 Visual Studio 2019 软件, 并且进行更新。

3.3 导入美术资源

第一种方法, 软件 3dsMax 或 Maya 可以制作出 Unity 中所需要的三维模型, 可以在合适位置添加光源,

调整参数。但是注意将游戏模型存储为 FBX 格式, 然后将模型的材质、贴图文件整理在一个文件里面。在 Unity 中新建新工程, 导入 FBX 模型, 注意调整建筑缩放比例, 之后在上面模型添加材质贴图。第二种方法, 也可以从 Asset Store 找到自己所需要的游戏组件, 直接导入 Unity。

3.4 unity 中使用跑酷游戏插件

在 Asset Store 找到游戏插件 3d runner 导入插件到 Unity 中, 开始创建场景、主角、障碍物、功能组件、音效设置等。

3.5 操作方式

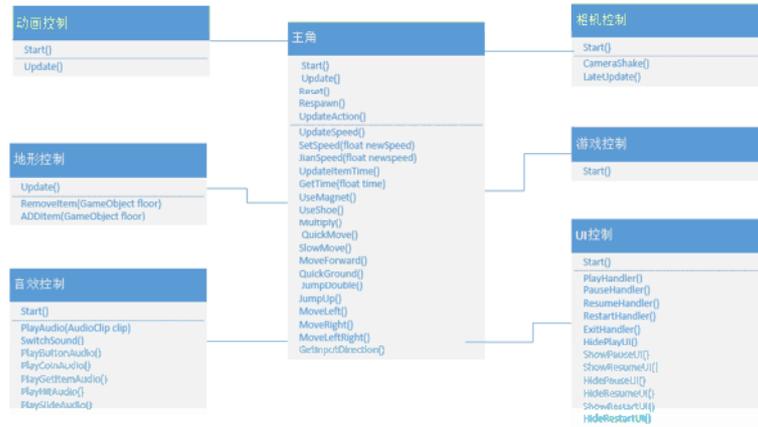
设定 VR 跑酷游戏在一个街道里进行, 玩家只有一次生命, 靠手势识别进行各种运动, 躲避障碍物并且获得高分, 中间游戏难度增加, 场景速度会随着游戏时间增加而增加, 碰到加成道具可以有各自技能, 碰到障碍物就死亡。障碍物有公交车、小汽车、栅栏、路障、隧道墙壁, 加成道具具有强力磁铁, 超级跑鞋、双倍积分、暴走、慢走。界面会显示音效设置, 左边金币、磁铁、暴走、双倍积分、跑鞋的数量, 右边显示有磁铁、跑鞋、双倍积分、慢走、暴走持续时间, 中间有重新开始、开始、重启、暂停。当玩家碰到障碍物, 玩家死亡, 跑酷结束。

Gear VR 的操作步骤: 取下镜头保护罩, 将手机与 Gear VR 对接。听到一段提示音, Oculus 应用程序会自动启动, 将手机锁定, 你便可以戴上 Gear VR 准备进入虚拟世界了。

4 游戏设计

4.1 地形的设计

VR 跑酷游戏需要有连贯的背景。但是背景过长, 开发的数据成本势必会增加, 游戏大小会受到影响影响, 游戏体验流畅感很不好。因为跑酷游戏中街道模型有限制, 所以需要街道重复与销毁。街道地形上面随机放置了障碍物、楼房、路灯、加成道具, 建立一个 pattermanager 脚本, 在其中定义两个工具类, 一个用于存储道具信息 (道具品种和位置), 一个用于存储道具的集合。存放已经生成好的场景集合, 然后随机访问场景。在每次道路交替时, 清除道路上的所有物体, 从场景类的 list 中随机取出一种, 根据其中的物体类列表一次取出并放入道路, 从而实现随机场景的功能。



4.2 主角的设计

跑酷游戏的目的是玩家尽可能跑得更远，去躲避较多障碍物得到高分。在期间玩家的移动速度会增加，场景增加障碍数量，玩家躲避障碍的难度会加大，要想拿高分，玩家就需要更谨慎。作为动作游戏中的重要一类，跑酷游戏需要玩家在奔跑过程中躲避随机产生的障碍物，由此满足玩家享受跑酷游戏的动感节奏，带给玩家丰富的满足感，生成的障碍物是VR跑酷游戏丰富性的重要体现。

将玩家放置在街道场景的开头，先给玩家添加相机跟随和手势识别。获取滑动方向函数 `GetInputDirection`，判断鼠标左键是否按下，记录鼠标按下位置、鼠标松开的位置，后者减前者的坐标得到滑动方向的向量。通过点积，我们其实可以计算两个向量的夹角判断鼠标在对 X 和 Y 的正方向的夹角，y 夹角小于 45 度，为向上滑动；y 夹角大于 135 度，为向下滑动；x 夹角小于 45 度，为向右移动；x 夹角大于 135 度，为向左移动。玩家可以双连跳，但是禁止三连跳，而且中途玩家速度会递增，设置初始速度，新加三个变量——路程变量，速度递增变量，最大速度变量。根据距离加速，在最大速度之前每隔一段距离加速，最大速度之后停止增加。并且在玩家运动过程中根据手势识别播放相应动画以及音频。

```
void GetInputDirection() {
    inputDirection = InputDirection.NULL;
    if (Input.GetMouseButtonDown(0)) {
        activeInput = true; mousePos = Input.mousePosition;
    }
    if (Input.GetMouseButton(0) && activeInput) {
        Vector3 vec = Input.mousePosition - mousePos;
        if (vec.magnitude > 25) {
            var angleY = Mathf.Acos(Vector3.Dot(vec.normalized, Vector2.up)) * Mathf.Rad2Deg;
            Debug.Log(angleY);
            vec.normalized Vector2.up;
            var angleX = Mathf.Acos(Vector3.Dot(vec.normalized, Vector2.right)) * Mathf.Rad2Deg;
            Debug.Log(angleX);
        }
    }
}
```

```
inputDirection = InputDirection.
Up; AudioManager.instance.PlaySlideAudio();
} else if (angleY > 135) {
    inputDirection = InputDirection.Down;
    AudioManager.instance.PlaySlideAudio();
} else if (angleX < 45) {
    inputDirection = InputDirection.Right;
    AudioManager.instance.PlaySlideAudio();
} else if (angleX > 135) {
    inputDirection = InputDirection.Left;
    AudioManager.instance.PlaySlideAudio();
} Debug.Log(inputDirection); } }
```

4.3 加成道具和障碍设计

在VR跑酷游戏中，为了增加游戏的难度及趣味性，用加成道具提供乐趣，用不同动作躲避不同障碍物。VR跑酷中设置了磁铁、金币、跑鞋、双倍积分、慢走加成道具，以及公交车、小汽车、栅栏、路障、隧道墙壁等障碍物。Obstacle脚本里面除了地面和磁铁，其余障碍物碰到玩家死亡，播放死亡动画。

PlayerController脚本里面 `UpdateItemTime` 函数获取各个道具的时间，每个加成道具添加启用、时间、停用功能代码。其中给金币添加碰撞体，然后勾选为触发器，为其添加一个脚本，内容是碰到“Player”标签后则销毁自身，并产生一个粒子特效，同时金币数量加1。其中暴走状态 `QuickMove` 函数，倒计时很快，是因为有两个协程共用一个时间，两个协程同时在跑，需要看是否在暴走，然后保存暴走前的速度，以及暴走的剩余和持续时间和暴走协程。在 `QuickMove` 函数中如果协程已经创建则停止，然后初始化协程再启动协程。在主角速度激增加成中需要设置主角速度，导入一个新的速度，新速度不能大于最大速度。然后更新速度，累加距离，距离达到了需要更新的值就设置新的速度，然后重置距离计数器。还有强力磁铁需要在碰撞体代码中检测是否碰到金币，加入吸收金币功能，磁铁代码调用磁铁函数，启用后，初始化时间，激活碰撞体，吸收金币，时间结束后，停止功能，关闭碰撞器。

要显示加成道具图标、剩余时间显示、道具数量显示、金币显示。

4.4 相机的设计

玩家与障碍物发生碰撞时，相机会发生抖动，原理是短时间内更新相机的坐标。首先，判断相机是否处于抖动状态，不在抖动状态，就需要进入协程，用倒计时来实现。然后，相机在 x 轴稍微移动 x 的值为随机值，在 -0.1 到 0.1 之间，y、z 的值不变。最后相机不抖动的时候，相机跟随。它需要定义一个 pos 记录相机的移动位置，设置摄像机跟随的目标，摄像机距离主角的距离和高度，引用摄像机的位置，初始化摄像机的位置，利用摄像机跟随目标函数，来实现相机跟随原理。

4.5 动画的设计

在玩家运动过程中，要根据手势识别，游戏播放相应动画。在 AnimationManager 脚本里面引入向左、向右、向上跳、向下滑行、死亡动画。

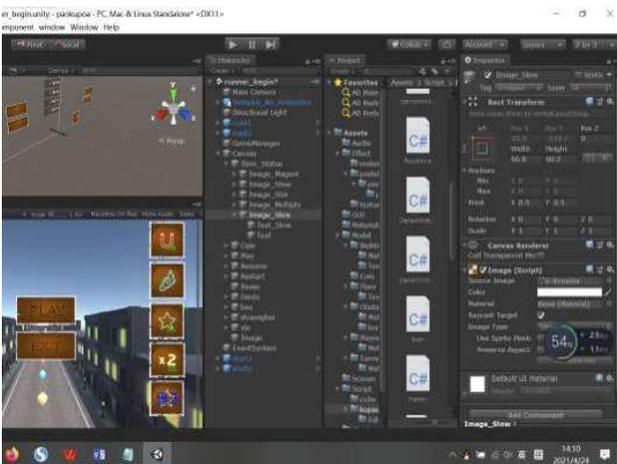
4.6 音效的设计

在 Assets 的 Audio 里面找到各种音效，进行添加。在 GameManager 上面添加控制音效的脚本，引入 button、金币、技能、撞击、滑动音频，添加播放音频函数，设置音频的开关函数。在 PlayerController 脚本里面的 GetInputDirection() 函数将相应音效实例化。在 AudioManager 脚本里面加播放音效函数，到达哪个位置，播放哪种音效。切换音效开关取反，设置 Image 的 Sprite。设置各种音效，跨脚本访问，全局实例播放音效。把每种对应音效放置在主角控制音效的面板里面。

4.7 UI 控制

在 Canvas 中新建加成道具、金币、开始、暂停、重启、重新开始，在 image 上面添加事件触发器，在 UI 控制脚本中实现逻辑关系。

4.8 游戏效果



4.9 游戏的发布与运行

制作完成游戏，需要将文件发到三星手机上面。而且因为 Oculus 的所有应用都是通过设备 ID 认证，所以需要注册 Oculus 账号并登陆，并完成激活。手机插入 Gear VR 会看到提示，再把手机取下，并安装相应的软件；手机插好后会有一个安全提示，然后进入主界面，下载后就会出现在 Library 中，分为视频、体验、游戏、应用四个分类，下载几个应用后，再把手机插到底 Gear VR 眼镜上，就可以体验 VR 虚拟现实的效果。

结束语

本文介绍了基于 Unity3D 的跑酷游戏的设计与实现，包括了地形、人物、障碍物等等的处理，对玩家进行了详细介绍。VR 会让玩家更有沉浸感，可以让玩家仿佛处于另一个世界。本设计虚拟现实与跑酷游戏相结合，让低头族抬起头来，是一款益智益体的游戏。

【参考文献】

- [1] 王洋, 王元鸿, 胡志慧. 基于 Unity 网络游戏开发技术的研究与实现 [J]. 科学技术创新, 2021(10):95-96.
- [2] 窦杨辉, 王巍, 何鑫, 毛文泰. 基于 Unity 的校园 3D 跑酷游戏的研究与开发 [J]. 计算机时代, 2017(03):52-54+57.
- [3] 李敏, 张世邀, 廖成林. 基于 Unity3D 的 VR 跑酷游戏设计与实现 [J]. 信息与电脑 (理论版), 2020, 32(12):119-122.