

Unity3D 中室内探索的设计

吕姣霖 王春洁

四川大学锦城学院计算机与软件学院 四川 成都 610065

【摘要】随着移动互联网的兴起,游戏让人们可以不时地放松心情,调节生活节奏。探索游戏的目的是让用户更加真实地体验到剧情的跌宕起伏,满足用户好奇心,在虚拟世界中体会到不同于现实世界的故事情节。本款游戏基于Unity平台,创造了一个让人感到紧张刺激的虚拟房屋,玩家需要跟随主角在房屋中进行探索,最后还原房屋中所发生的事件,以及成功走出房间。

【关键词】探索游戏;设计;Unity;解谜

引言

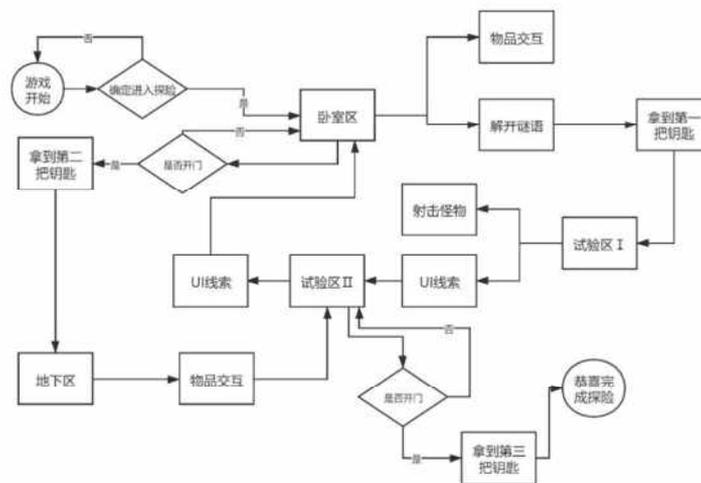
本款游戏是在Unity平台上构建的3D沉浸式第一人称视角的游戏,以C#作为游戏的代码脚本,用3DS MAX平台来搭建模型。Unity3D为一款虚拟现实引擎,平台可以提供一套软件解决方案,多用于创作交互式的2D或者3D内容^[1],配合着由C语言和C++衍生出来的面向对象的编程语言C#,由3DS MAX制作导入模型,三款软件结合起来使游戏开发过程更加严密。

1 游戏设计

1.1 游戏流程

玩家打开游戏首先看见的是游戏的主菜单界面,在此界面中玩家可以选择是否开始探险,如果选择是则跳转到游戏主场景,计时器开始计时;如果选择否依然

停留在主菜单。在主场景界面中首先到达的是主角卧室,玩家会与卧室里的物品进行交互,比如发现门被锁住、衣柜中的信息、墙上画的信息等等,当玩家解开房间中的谜语,拿到道具第一把钥匙之后,触发门可以打开。玩家可以来到试验区I,主角此时进入怪物的检测范围内,玩家需要捡起武器道具射击怪物,然后根据场景中的UI线索进入试验区II,玩家在试验区II中会得到更多关于探索房间秘密的线索,根据线索回到卧室找到第二把钥匙,如果谜语找到则无法进入下一场景。拿着第二把钥匙可以进入地下区,地下区主要交代剧情,解开房间的谜题,玩家得到线索去往试验区II拿到第三把钥匙,打开地下区的门,走出房间;跳转到游戏结束界面,计时器结束计时。如图一所示:



图一: 游戏流程设计

1.2 剧情背景设计

本探索游戏的剧情背景为在上个世纪七十年代,当地发生过一场核泄漏事件,在此后的四十年间方圆三十公里,都无人居住,其中有一个破旧的房子,曾经是一家三口的居住地,房子的男女主人都死于核泄漏,只有他们的小儿子存活下来。而最近有路过的人时不时听到里面传出诡异的声音。此刻政府派出我们的主角警察探员寻访这个房子,结果发生了出人意料的事情。主

角需要在房间中不断与物品交互得到线索,最后找出了该房间的主人以及房间中发生的事情。

1.3 场景设计

游戏场景主要分成三个部分,第一部分为游戏准备场景,第二部分为游戏运行场景,第三部分为游戏结束场景。第一场景与第三场景以UI界面为主,帮助玩家了解游戏目的,第二场景主要以主角与物品互动为主。第二部分为游戏的主要场景,由四个房间构成,分别为

卧室、试验区 I、试验区 II、地下区；卧室、试验区 I、试验区 II 为光线昏暗的暗黑风格，主要物品有生活用品、实验用具以及实验器材等，地下区为温馨的风格，主要物品有实验成果、检测机等。

1.4 游戏主角设计

游戏主角为一个中年男子，职业为警员，目的为探索未知房间中的秘密。主角的功能主要分为主角运动、主角拾取、主角射击。

(1) 主角运动：玩家通过按键输入控制主角运动，先需要进行判定主角是否在以 isGround 地面上层；如果在，则可以使用 WSAD 键进行上下左右移动，空格键可以跳跃，此时摄像机机会跟随主角的视角运动。

控制水平、垂直方向的方法为 `horizontalMove = Input.GetAxis("Horizontal") * (moveSpeed - 2f); verticalMove = Input.GetAxis("Vertical") * moveSpeed;` 控制跳跃的方法为 `velocity.y = jumpSpeed[2];`

(2) 主角拾取：主角拾取分为道具拾取和武器拾取。玩家通过鼠标点击道具实现道具拾取，道具上有圆形检测器，当标签为 Player 的主角进入检测范围后，点击鼠标左键进行 fire1 事件，销毁道具之后同时去往下一个房间的条件 key 变为真。具体实现代码如下：

```
if (cols[i].name == "Player")
{if (Input.GetButtonDown("Fire1"))
{Destroy(gameObject, 0.2f);
key = true;}}
```

武器拾取为按键拾取，主要逻辑为在游戏开始前主角身上已经绑定了一个隐藏起来的武器，所以刚开始在游戏场景中是看不见的，在到达试验区 I 后，跟随提示找到武器道具后，需要通过按下 z 键拾取武器副本，这个时候使用 Destroy 函数销毁场景中的武器副本并且使主角身上的武器显示出来，具体实现如下：

```
Destroy(gameObject);
palyer.SetActive(true);
```

(3) 主角射击：主角射击为按键输入响应事件，运用了射线检测函数去发射激光束，在主角拾取到了武器之后，按下 Q 键可以发射激光束，激光束会是射线加

特效，在没有碰到怪物时，是向无穷远发射；在碰到怪物后，就在怪物身上停止，具体实现如下：

```
gunParticles.Stop();
gunParticles.Play();
gunLine.enabled = true;
gunLine.SetPosition(0, transform.
position);
shootRay.origin = transform.
position;
shootRay.direction = transform.
forward;
```

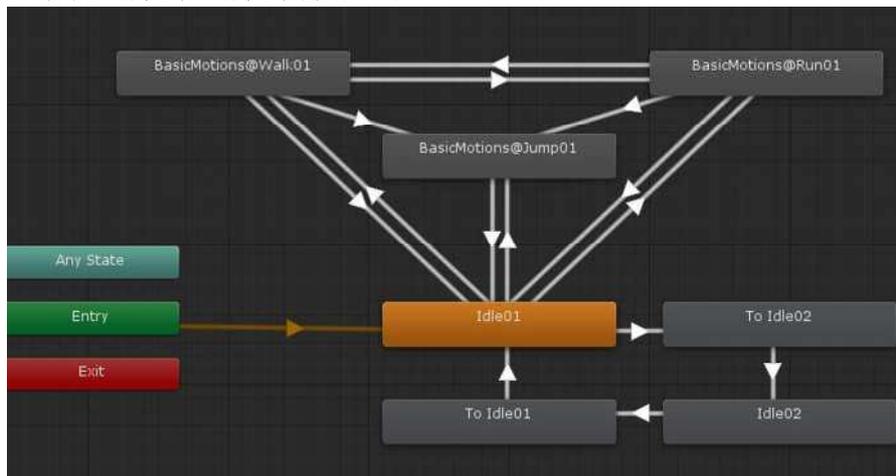
1.5 怪物设计

怪物为变异动物，来源是当地原有动物经过了核辐射变异后形成。怪物只在试验区 I 的方形检测器范围内会出现，当标签为 Player 的主角进入检测范围内

void OnTriggerEnter(Collider other) 时会隔一段时间出现怪物，怪物会实例化特效，然后在两秒之后销毁特效，当怪物碰到标签为 Wall 的墙体或者标签为 Player 的激光束时会自动销毁，怪物的生成主要使用了 IEnumerator 携程函数，携程启动后每隔 1s 生成一个敌人，在第 2 秒后调用关闭携程的函数，使携程关闭。怪物的运动方向为全体向墙体运动，并不会自动追踪主角，也不会伤害主角，目的是推动剧情的发展，营造恐怖气氛。

1.6 动画设计

Player 主角的动画主要有 Idle、Jump 和 Run 三种状态。Idle 空闲状态是游戏刚开始的状态，是指没有对主角进行任何指示，主角不需要进行任何响应的一种状态；Jump 跳跃状态是主角往上运动一段，然后快速恢复的一个状态；Run 状态是主角在水平方向和垂直方向运动的状态，可以由任意状态转换到 Idle 状态，也可以从 Idle 状态转换到任意状态。但是 Jump 与 Run 不能直接状态，Run 可以直接到 Jump 状态，Jump 会先到 Idle 状态再到 Run 状态，多少种状态之间转换是依靠按键输入响应事件。如图二所示：



图二：动画设计

1.7 UI 设计

本探险游戏的 UI 主要分为三个部分，一是游戏的开始和结束场景的 UI；二是道具背包系统的 UI；三是

与物品交互时出现的线索 UI。

(1) 开始与结束场景：新一个场景，如下面代码所示：

SceneManager.LoadSceneAsync(“SampleScene”);
当获取到按钮响应事件后跳转到主游戏场景中。

(2) 道具背包系统: 当进入道具的检测范围时, 可以点击道具进行拾取后触发摧毁道具, 同时改变背包 UI 中的 image 的 bool 值, 使其先设定的图片在背包位置中显示出来。

(3) 文本线索: 当主角进入主要物品的触发检测范围时, 该物品的文本就会显示出来, 每一个文本框的大小可以根据文本的内容在代码中进行调整, 目的是为了推进剧情的前进, 确保玩家能按照设定的游戏流程来完成任务, 主要代码如下:

```
GUIStyle style1 = new GUIStyle();  
style1.fontSize = 45;  
style1.normal.textColor = Color.green;  
GUI.Window(0, new Rect(960, 540,  
windowkuan, windowchang), MyWindow, name);
```

1.8 音乐音效设计

(1) 音乐的制作: 第一个背景音乐是整个游戏场景都在循环的音乐, 根据故事剧情以及背景所以整个场景循环播放了一个烘托恐怖气氛风声的音乐, 是用以下代码实现:

```
AudioClip clip = Resources.  
Load<AudioClip>(name);  
MediaPlayer.clip = clip;  
MediaPlayer.Play();  
MediaPlayer.Stop();
```

第二个背景音乐是随着主角的移动到不同的场景会切换的音乐, 在卧室区跟实验区 I、II 是按照剧情播放的比较恐怖的音乐, 在地下区是按照剧情播放的比较温馨的音乐, 用了一个存放音乐的数组, 先是默认播放第二个音频, 所有的背景音乐的播放都会一直循环, 主要以下代码实现:

```
this.GetComponent<AudioSource>().clip  
= audios[1];  
this.GetComponent<AudioSource>().  
Play();  
this.GetComponent<AudioSource>().clip  
= audios[0];  
this.GetComponent<AudioSource>().  
Play();
```

(2) 音效的制作: 音效是根据剧情在场景的不同地方做了不同的触发, 当主角进入触发范围之后, 就会播放音效, 并且只播放一次, 所以有下楼的音效 电流的音效 动物的音效等等, 例如用下楼的音效是用以下代码实现的:

```
private void OnTriggerEnter(Collider  
collision)  
AudioManager.Instance.PlaySound(“开门后”);
```

1.9 计时器设计

本探险游戏有计时器功能, 用来表示此次探险共花费的时间, 将计时器放在窗口左上角, 计时器不断变化, 当主角走出屋子时, 走到房门时就会触发触发器,

此时计时结束。实现计时器功能的主要代码为: $i += \text{Time.deltaTime}; \text{text.text} = i.\text{ToString}(\text{“f2”});$

2 游戏测试

本款游戏是室内探险类的一个场景游戏, 主体是以第一人称视角来进行操作的, 摄像机跟随主角 Player 的视角移动, 第一人称的操作会给玩家带来强烈的沉浸式感觉; 玩家通过房间里的文本提示破解机关最终走出实验室, 游戏中的线索设置适中, 不至于让玩家过于轻松或者过于困难地完成游戏, 整个过程要根据所触发的文本框提示才能最终完成探险。为了让玩家有更好的游戏体验感, 游戏中的鼠标会被隐藏, 所以在点击道具时需要玩家具有一定的空间感。游戏测试结果如图三:



图三: 游戏结果

结语

本探险游戏亮点在于剧情前期的紧张刺激与后期的温馨感人的反转, 使得用户更加感兴趣和印象深刻, 同时也希望通过这样强烈的故事情节对比, 营造一个为人类做出伟大贡献的博士, 借此对像在此次新冠疫情中同样为生物科学研究作出贡献的科研人员致敬。在前期模型的搭建中遇到了很多问题, 会出现导入模型时在软件中找不到的情况, 这就使得需要重新制作, 所以在制作的过程中要随时存档文件, 这样就会避免因为软件问题而导致工作量的增加。游戏所有的特效都是在 Unity 中利用粒子系统实现的, 主要用到的场景有生物碰到墙壁或者碰到主角销毁释放死亡特效以及主角射击生物, 生物死亡, 释放特效; 还有水管里不断流出的水以及水池里的血水都是利用粒子系统制作的。目的就是为静态的建筑添加上动态的效果, 使之更加贴近真实世界。游戏制作的难点还在于对于鼠标的隐藏, 为了让玩家有更好的体验感, 需要不停的测试物品的位置大小, 最终这些难点一一攻克, 形成完整的小游戏。

【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] 白新国, 赵思锁. 基于 Unity3D 的“Pyramid”VR 游戏设计与实现 [J]. 电子设计工程, 2021, 29(03):136-140.