数字媒体学院

<心生>

游戏设计文档

指导教师：　张军
成员：1030515409 金卓群　1030515420 郝思正
2018/6/6

“献给所有帮助过我和爱过我的人。“　　——<心生>游戏团队

目录

一、游戏概述

1.1 游戏类型

冒险类、剧情类。

1.2 内容概述

玩家扮演主角——小姑娘艾卡（Eika），通过简单的指令克服路途上的困难，找到她的父亲，从而解开对父亲不辞而别的疑问

1.3 基本概念

游戏具有两个关卡，主角艾卡（Eika）先是在一个开放地图上自由巡游并且完成指定的任务——躲过怪物的视野，悄悄拿走精魂，以此祈得神明的指引。
之后艾卡（Eika）必须被迫不断前进，并且在路途中收集神秘的气息。

1.4 目标人群

青少年玩家（12-28 岁），游戏本身的难度并不高，不需要长时间的练习和高超的操作水平，《心生》游戏更注重故事性和代入感，在游戏给人带来放松和愉悦感的同时，更希望玩家能通过游戏所讲述的故事获得些许感动和感悟。

1.5 游戏特点

操作难度低；画面精美；代入感；

1.6 游戏的操作和基本玩法

移动：w a s d
切换攻击/移动状态：q
镜头旋转：鼠标右键
跳跃：space
攻击：在攻击状态下，站定，鼠标左键单击目标位置

1.7 游戏背景:

主角艾卡（Eika）是一个八岁的小姑娘，天性活泼开朗，一举一动都惹人喜爱。平日里最喜欢跟他的爸爸一起去公园玩，他们在公园里观察蚂蚁搬运食物，秋天的人字形大雁和春天的“剪刀手”小燕子。同邻居家的小狗一起你追我赶，还有一棵棵树：柏树、梧桐、香樟……

“爸爸好像什么都知道，爸爸可太厉害啦！”艾卡（Eika）心里想。

但先天性心脏病却像是一块阴霾，遮住了艾卡阳光灿烂的笑容。

二、游戏世界和主要玩法

2.1 游戏世界

游戏剧情:

在一次手术麻醉之后，艾卡似乎并不是沉沉睡去，而是进入到了另一个世界。这个世界同梦一般梦幻和美好的同时，似乎也有一些诡谲的气息……

眼看着爸爸也离自己越来越远，不知道跑到哪里去了。

“不管怎么样，先找到爸爸在一起离开这里吧！”艾卡暗自对自己说。

在附近寻觅的艾卡发现了爸爸的帽子——“大飞碟“也在这里。在昔日伙伴“大飞碟”先生的指引下，艾卡决定前往巨石祭祀台祈求神明的帮助，在躲避开巨大、恐怖的怪物“猴姆沃克”的重重包围、苦苦追赶终于集齐了 6 颗精魂。通过献上精魂的祭祀之后，神明显示出的福祉指引她前往新的地点：瘟疫长廊——或许父亲就站在长廊的尽头等待艾卡。

在瘟疫长廊，她必须尽快向前，犹疑会让她吸入太多的有毒气体，最终被瘟疫所感染。在路上搜集“神秘的气息”，并且躲避致命的、已被感染的僵尸人的突然袭击。

紧张的跋涉之后，艾卡终于到达了“瘟疫长廊”的尽头。

“嗯？这是什么东西？”艾卡惊讶道。

一路上所搜集的一个个“神秘的气息”，竟然从口袋跑出来开始自行汇聚和结合……

慢慢的，气息结合成了一个隐约的人形，越来越清晰……，艾卡屏住呼吸，紧张又充满期待的看着这一切。

“爸…爸爸！”，艾卡难以抑制地叫出来。

经过艾卡和父亲的再次会面，艾卡终于明白了父亲的心意。她决定要永远正直、善良，成为父亲一样的人，这也是父亲对她的期待。

世界设定:

在游戏世界中，现实世界和精神世界的进入界限是肉体死亡。在肉体还生存着时，人的精神虽然存在，但是精神此时被肉体所束缚，无法进入精神世界，更不能同其他人的精神在精神世界共存。

游戏中，艾卡在手术中出于濒死的状态，而艾卡的父亲则是肉体刚刚死亡，所以才有了在精神世界相见的可能性。

2.2 主要玩法

关卡 1:

玩家通过简单的移动和攻击指令完成给定任务，在怪物的紧密包围之下，尝试和寻找最佳路径，捕捉任务物品。

怪物视觉缺陷的机制，让玩家可以通过走位来混淆怪物的视听，成功隐藏自己以便进行下一次前进（在任何情况下，只要用其他物体挡住自己的身体，怪物就会不知所措）。而随地可见的石头，也让弱小的玩家有了同怪物一叫高下的资本（对怪物进行投掷可以略微降低怪物的血量，更重要的是，角度合适的情况下可以一定程度击退怪物）。

任务物品来回的游动，要抓住他们可需要不少时间，你该如何让不懂人话的怪物乖乖的等在外面呢？

关卡 2:

玩家要通过控制平移和转向，在瘟疫长廊中收集“神秘的气息”，这个关卡会强制不断前行，使你不得不去面对未知且危险的道路。

这一路上你会遇到很多可怕的僵尸怪，会朝向你发起知名的攻击，但再厉害的怪物也有弱点，他们无法自由移动，你要做的就是运用走位技巧，尽力躲避他们。

成败在此一举，你能否在有限的时间内搜集到足够的“”神秘的气息“，并且成功躲避僵尸群的攻击呢？

2.3 游戏灵感和故事设计

（1）游戏灵感

游戏的直接灵感来源于一个视频，题为：国外一父亲把心脏移植给儿子，录下的对他儿子说的话。（是一个演员演的，但真情实感，令人动容，同时也是电影《迫在眉梢》的片段。链接：https://www.bilibili.com/video/av17956969?from=search&seid=15777922266905374255）。

下面是视频的台词：

“我的孩子，我给你找到了一颗新的心脏。但我还有一点儿事要跟你说：

爸爸希望你永远要听你妈妈的话，因为她是你最好的朋友，家人太重要了。至于女孩儿，你现在还太小，但等时机到来的时候，记住把她们当作公主，好好对待她们，因为她们本来就是公主。如果你说你想做……（哽咽），如果你说你想做什么事，放手去做吧。因为你应当言行一致。如果有能赚钱的机会，把握住它，即使你得孤注一掷，别像你爸爸一样犯傻。有钱会让很多事都容易一些。别抽烟，更不要吸毒，努力让自己保持善良、诚实。如果有人试图要伤害你，做个男子汉，为自己挺身而出。不要去让坏事缠身，因为世界上美好的事情有太多，你该去追求。

我会一直与你同在，无论发生什么，我就在这（指向孩子的心脏位置）。

回见了，儿子。

我爱你。“

（2）故事设计

我们想展现的世界和故事，就是在心脏移植手术进行的前后数小时之内，父亲与孩子的离别与重聚：

离别：一开始，父亲在现实世界不辞而别（由于取出心脏，肉体已经死亡），主角艾卡并不理解父亲的离开，由此踏上了前往追寻父亲的路程。

重聚：经过一路的追寻和经历，在精神世界重见父亲之后，艾卡终于了解了父亲离别的缘由和父亲的心意——父亲情愿以生命为代价来延续自己的生命，对于父亲来说，看到自己活着，便已是看到了未来。比起悲伤，更要做的应该是不去辜负父亲无私和伟大的爱，努力去做一个正直、善良的人，父亲所期待的人。

游戏最终的名字：《心生》，一方面游戏是围绕“心脏”和“生命”的话题展开和讲述的；另一方面，“心生”也同音“新生”，显然艾卡获得了新生（新的生命），更不仅仅是肉体上的生命得到了延续，通过对父亲疑问的追寻，相信艾卡从里到外都获得了一次新生。而父亲把艾卡视作挚爱，视作自己的未来，同样的是一次死亡，也是一次新生。我们在此祝福艾卡，也祝福艾卡的父亲。

在游戏情境中，父亲为了自己的孩子献出了自己的心脏，付出了生命。在我们每个人的生活中，可能不会有人为我们付出整个生命。但是，不论是朋友、家人甚至是陌生人，他们心甘情愿的付出自己的时间、金钱、精力，只为了让你能好一些，而不是为了自己什么，这都是爱。为受到别人的恩惠和帮助而欣喜亦或局促不安，更该怀着感恩的心，为了这大大小小份份爱意，活出更好的自己，这才算是对他们的报答。

我们的游戏想法比较冲动，选择了从游戏背景和游戏故事全部需要从头构想的一款注重故事性的游戏，在原画、模型和动画的缺乏和受限的条件下，这样类型的故事并不是一个太好的选择。但我们仍然决定尽量将它完整地呈现出来。

“献给所有帮助过我和爱过我的人。“
——《心生》游戏制作团队

三、人物、物品和地形设计

3.1 人物设计（图+简介）

（1）艾卡：
基本信息：
年龄：8
性别：女
身高：1.2m
发色：棕色
装束：可爱的
喜欢的颜色：粉色和蓝色
喜欢的食物：胡萝卜
喜爱的活动：同爸爸去公园玩

设计参考：

（2）艾卡的父亲——西蒙：

年龄：32

性别：男

职业：工程师

（3）帽子

西蒙年轻时打猎出游所穿戴的帽子，同时也是艾卡童年时期的玩伴之一，因其形状被艾卡称呼为“大飞碟”。

（4）怪物-侯姆沃克：

听觉灵敏，视觉很差，智力偏低。进攻性强，善于啃咬猎物。生命力顽强，推荐进行躲避，避免与之正面交战。

3.2 资源设计

（1）以太精魂

简介：传说中组成宇宙万物的基本物质，具有神奇的力量，出现在祭司台周围。

（2）神秘的气息

简介：在瘟疫长廊发现的气息，艾卡总觉得很熟悉但又说不上来是什么味道。

（3）故事图片

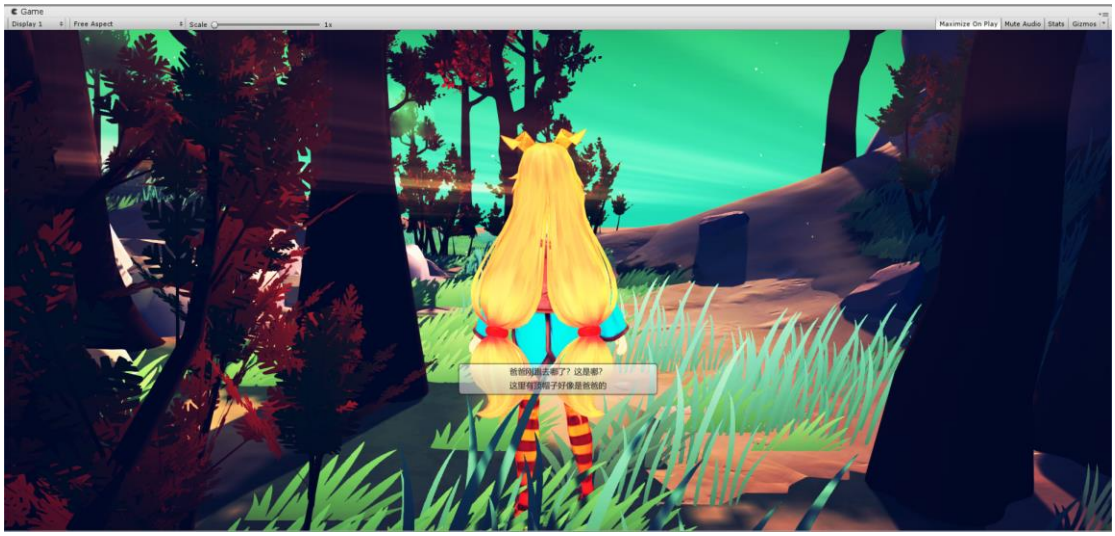
在开头的背景介绍中，艾卡发病昏厥——救护车赶来——同医生进行谈话——艾卡同父亲分别的图片均为创作、二次创作：

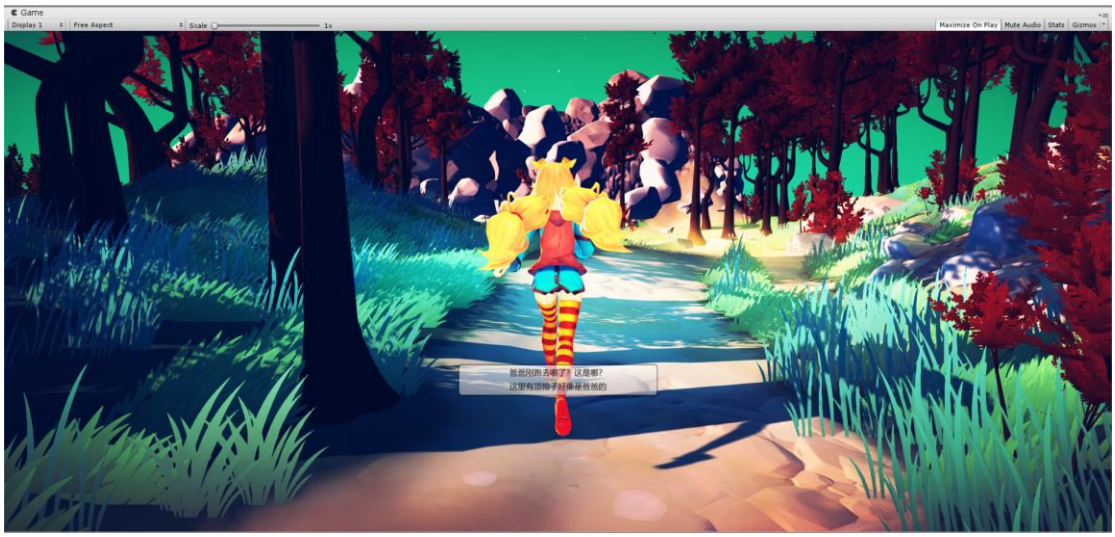

AMBULANCE

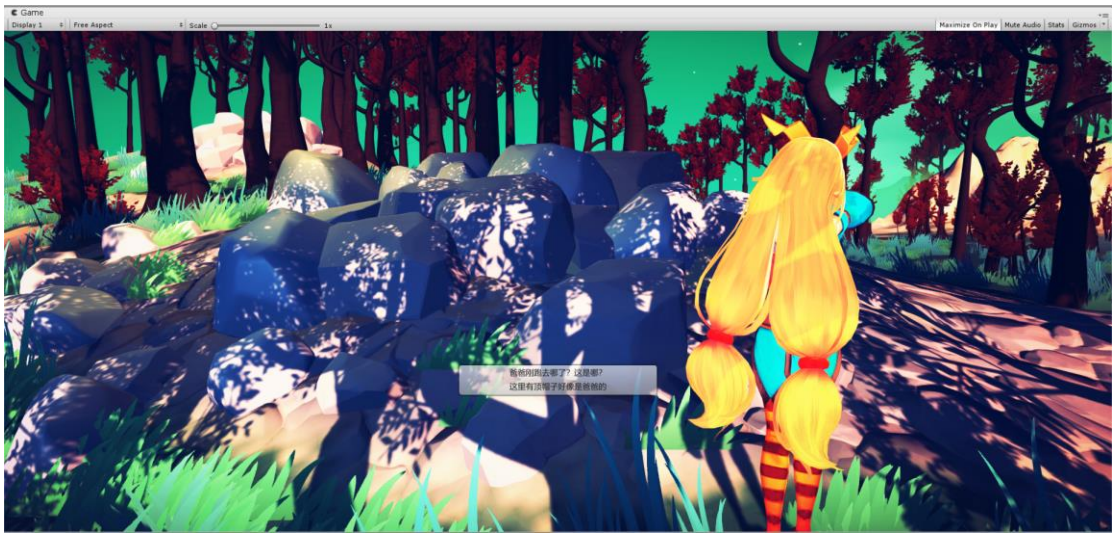
四、资源制作和引用

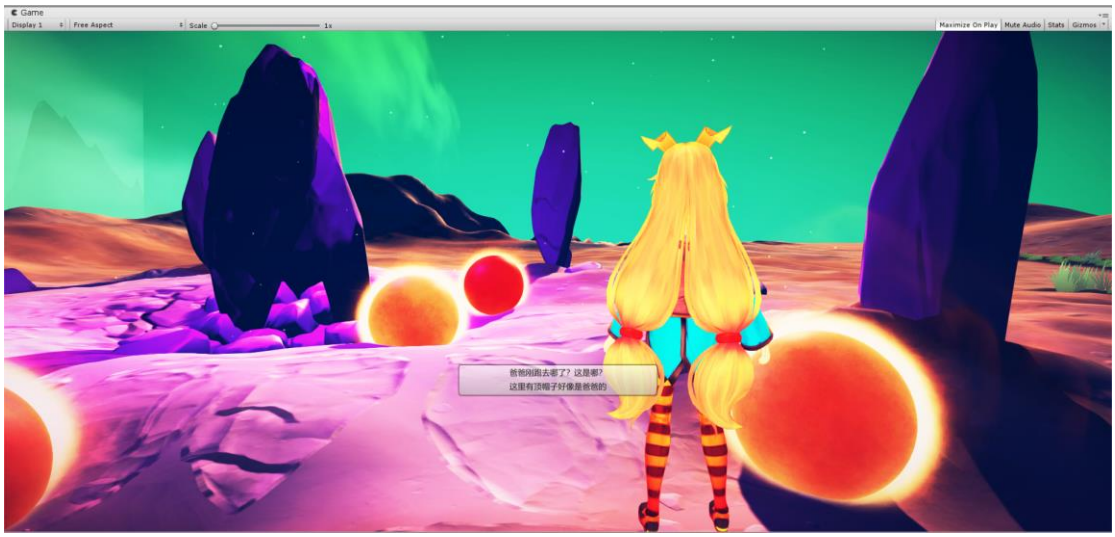
4.1 地形制作

地形是根据需要自行建立的。利用 5 种地形贴图，多个岩石贴图和法线贴图，三种草资源和三种树木模型，尽力构造出一个自然和神奇的游戏世界。







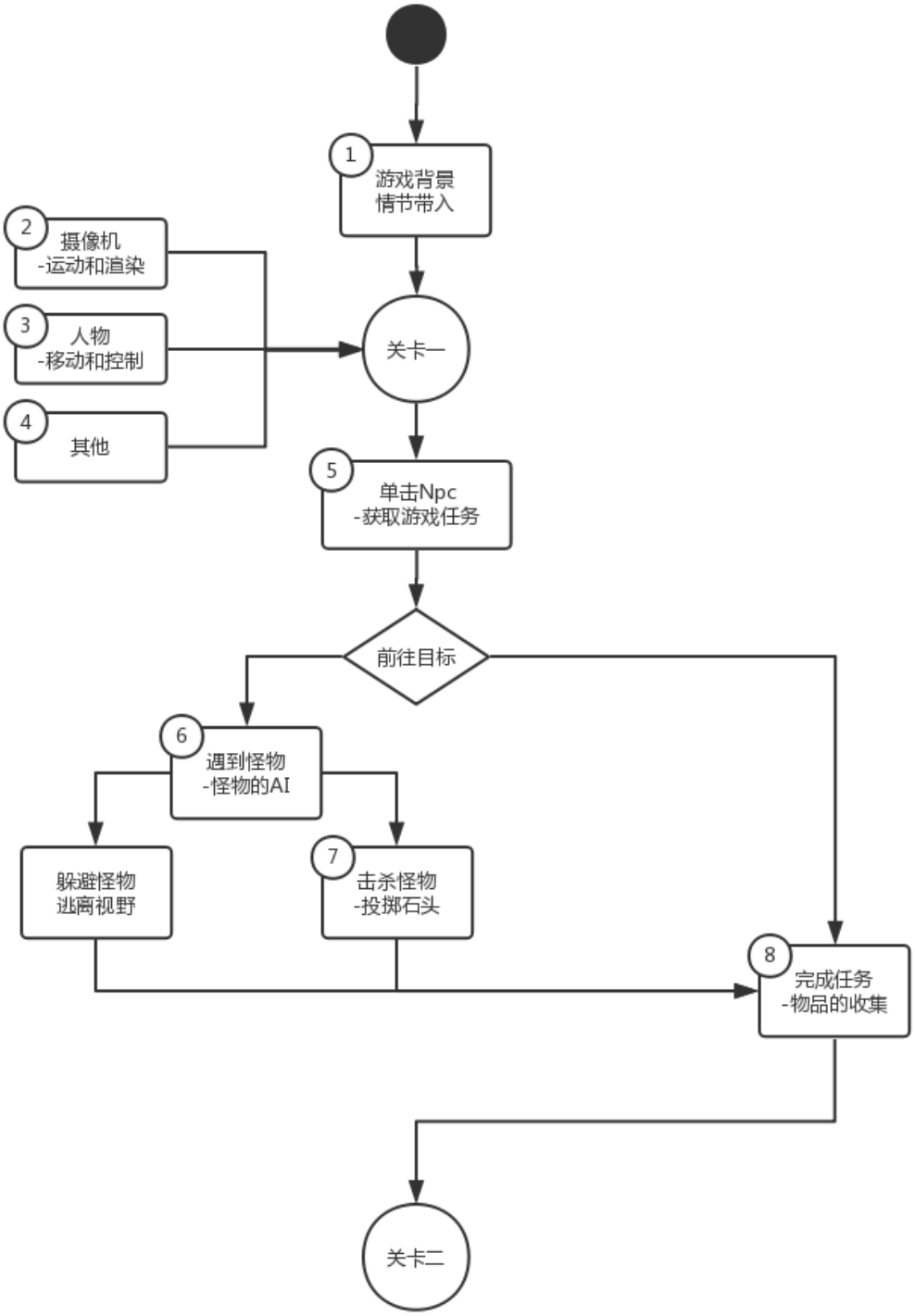


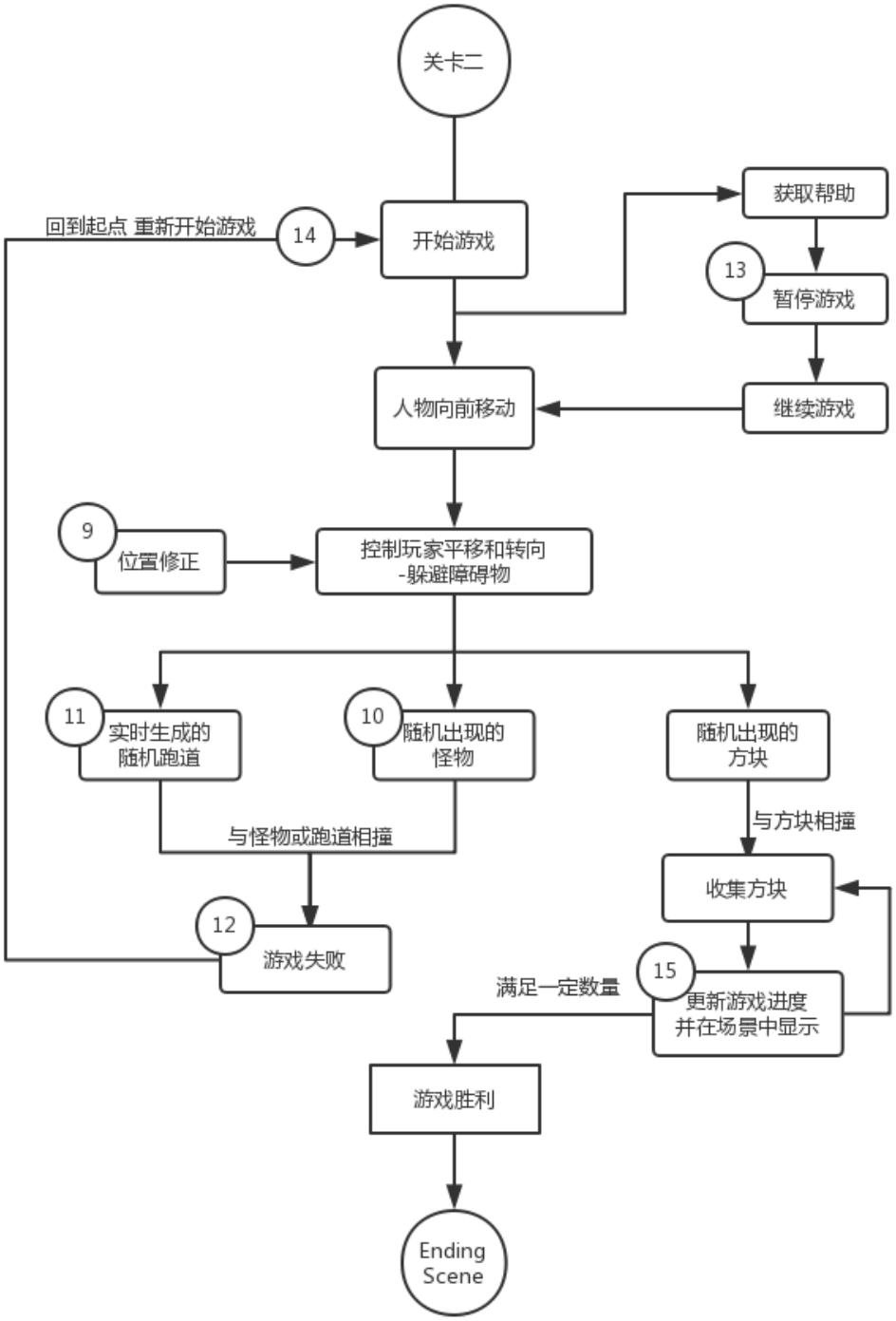
4.2 人物、npc、怪物模型：

淘宝美工，网络美工。

五、游戏流程和程序执行

5.1 游戏流程总览:


1
游戏背景
情节带入
2
摄像机
-运动和渲染
3
人物
-移动和控制
4
其他
关卡一
5
单击Npc
-获取游戏任务
前往目标
6
遇到怪物
-怪物的AI
躲避怪物
逃离视野
7
击杀怪物
-投掷石头
8
完成任务
-物品的收集
关卡二



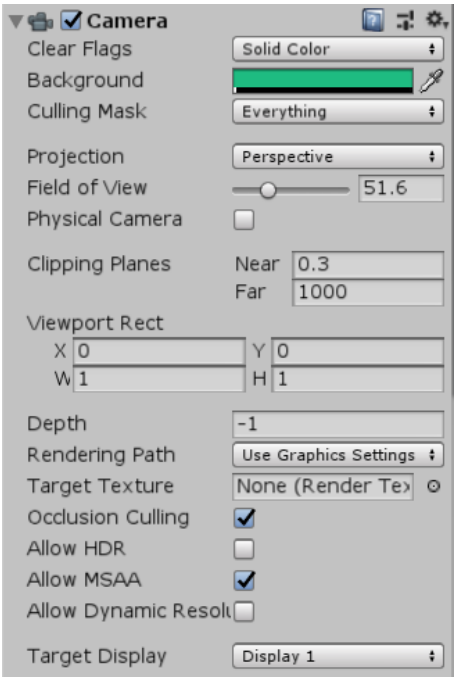
5.2 程序具体说明:

（1）游戏背景和代入

通过自己绘制的图片和对白切换完成背景交代。

（2）摄像机的运动和渲染

A. 基本参数的调节：



B. 视角调节脚本

ThirdPersonCamera.cs 控制了第一关卡中摄像机视角的调节，摄像机位置主要是跟随在主角的后方，获得包括主角在内的正前方视野。

首先，我们在主角的层级下添加了一个空物体 standardPos，置于主角的后方适当位置，作为摄像机的

然后，为了让玩家可以获得更宽阔的视野，我们允许玩家鼠标右键拖动来获得左右方向的视野：

transform.RotateAround(playerPos, new Vector3(0f, 1f, 0f), Input.GetAxis("Mouse X") * m_sensitivityX);

同时，利用 ctrl 键可以切换到主角正前方位置，向后看的视角，这里也是在主角层级下增加了一个在正前方的空物体，且它的正方向是面对主角的：

当玩家松开移动视角的任何指令之后，视角会迅速但不是立即切换到标准视角，我们规定了一个 float 型变量 smooth，并运用 Vector3.Lerp();来让这一过程变得更加自然和容易接受：

Vector3.Lerp(transform.position,standardPos.position,Time.fixedDeltaTime * smooth);

C. 渲染画面脚本

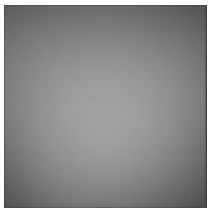
这里的脚本是 Unity 官方 Assets——StandardAssets-ImageEffects-中的三个脚本，分别是：

Antialiasing.cs：图形保真

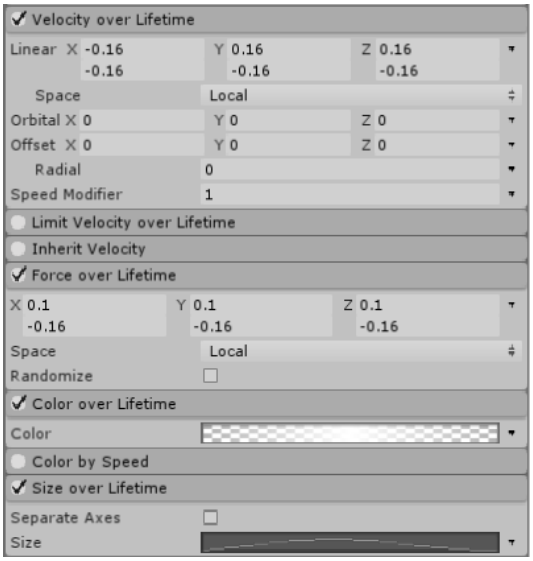
Screen Overlay :屏幕覆盖

Color Correction Lookup :色矫正

VignetteAndChromaticAberration：渐晕与色差

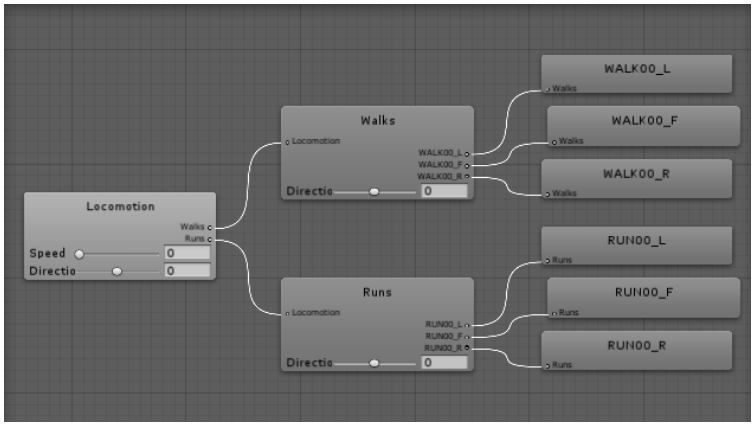

（screenoverlay 的贴图）

D. 灰尘-粒子系统
为了使画面效果更符合我们的预期，在摄像机层级下增加了一个子物体——粒子系统 Dust：它是一种暖浅黄色、发出微光的粒子，在一个 box 范围内不断产生，速度和受力在一个范围内随机、透明度和大小根据生命长度变化。



（3）角色的移动和控制
角色可以 走路，跑步，前进，向左，向右，后退，转向，跳跃，休息伸展。
这一切通过两个脚本来完成：PlayerController.cs 和 PlayerSkill.cs

A. 角色的前后左右移动：
首先给角色添加了 Animator 组件，将 向前跑步、转向跑步、跳跃、后退等动画添加到状态机中，并设置 float 和 bool 类型的控制参数。
并在移动指令中运用 BlendTree 通过速度和方向判定出不同的移动状态，触发不同的动画：



通过 get 热键的水平和竖直位移量获得移动的位移数据信息：
float h = Input.GetAxis("Horizontal");
float v = Input.GetAxis("Vertical");

根据得到的位移数据触发不同的动画动作，并令角色在世界移动。

B. 角色的跳跃和休息
角色必须在前进状态下才可以跳跃；跳跃的方式是通过对 rigidbody 施加力来实现的；跳跃时碰撞体的位置也会改变；为了更加真实，跳跃状态中人物距地面的距离过小时，动画已经基本触地，碰撞体位置就会重置到站定位置：

通过 raycast 进行距地面高度的判定：
Ray ray = new Ray(transform.position + Vector3.up, -Vector3.up);
RaycastHit hitInfo;
if (Physics.Raycast(ray, out hitInfo)){}

如果角色并不在跑动状态中，jump 对应的热键 Space 被触发，播放休息动画：

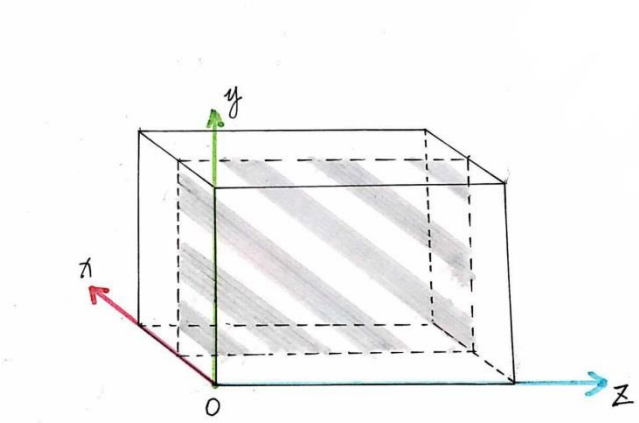
（4）其他
UI 的激活和冻结。

（5）Npc 事件（maozi.cs）
发布任务 Npc：始终面向角色，点击弹出对话框，通过交谈发布任务。

（6）怪物的行为（BossArtificialIdiot.cs）
怪物作为一个生命体，也有自己的特征属性和行为准则：随机游走、追击目标、攻击、死亡四个状态，怪物一开始处在随机游走状态。
并且怪物有听力较好而视觉和智力较差的设定，这一点体现在当你走进怪物的一定范围，不论怪物是否面向角色，他都将对角色进行狩猎（靠听力）；但如果在被狩猎的过程中角色利用其他物体遮挡住怪物到你身上的视线（Raycast）怪物会不知所措，进入随机游走状态。同时角色的极限跑速是大于怪物的，通过逃跑拉开一定距离也可以消减怪物对你的仇恨。

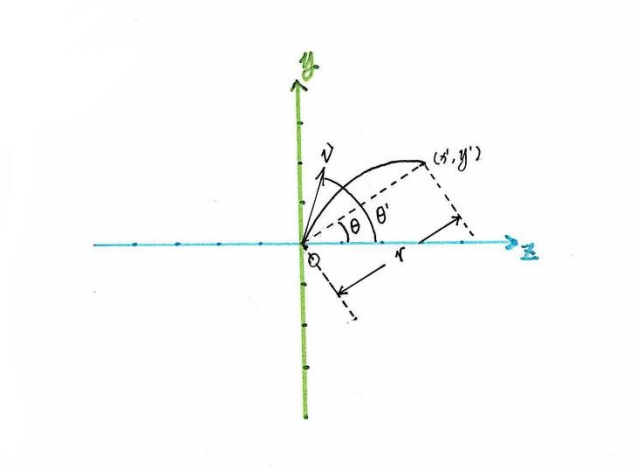
（7）人物技能——石头投掷

游戏中，角色可以投掷石头击打怪物，虽然可以直接让石头飞向怪物，或者去给定一个上抛角度，让投掷运动的曲线看起来比较自然，但我们最终决定去亲自计算出最省力的投掷角度和力度，让角色变成一个聪明人。

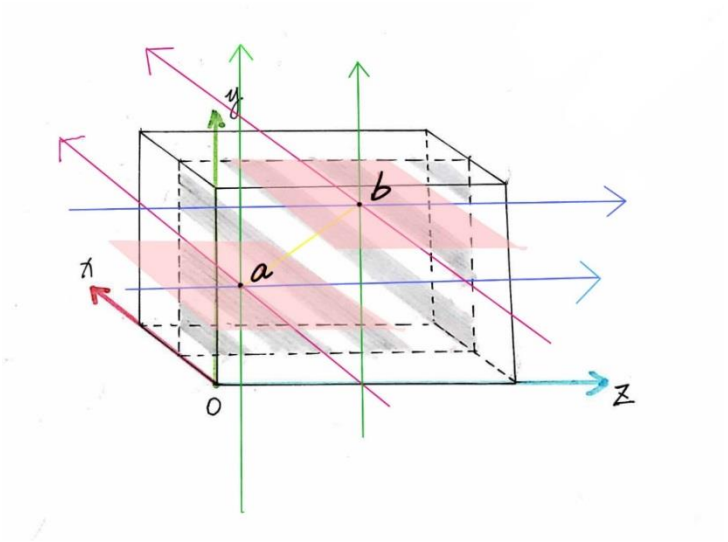


得到最佳投掷角度的同时，还应得到此角度对应的速度大小，大小和方向构成一个类似力的向量。

我们知道在二维平面上的投掷体运动是这样的：



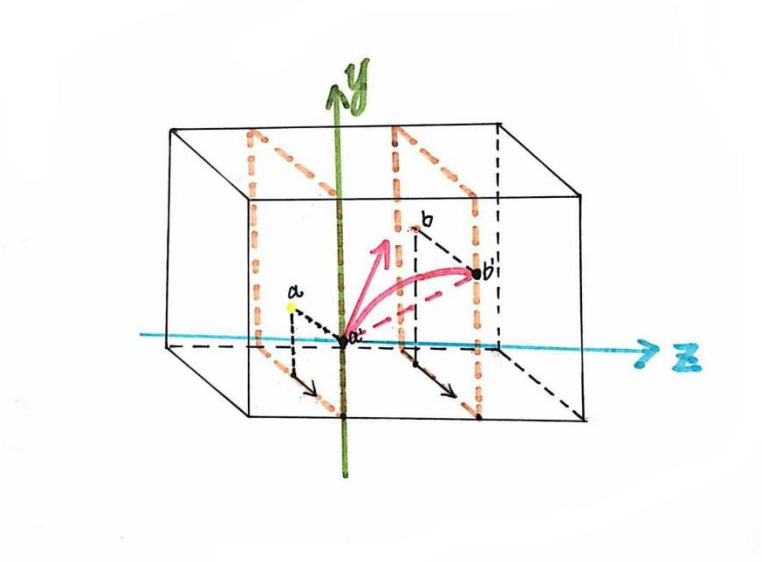
而在三维空间中，投掷体从 a 运动到 b 的计算看起来要更复杂：



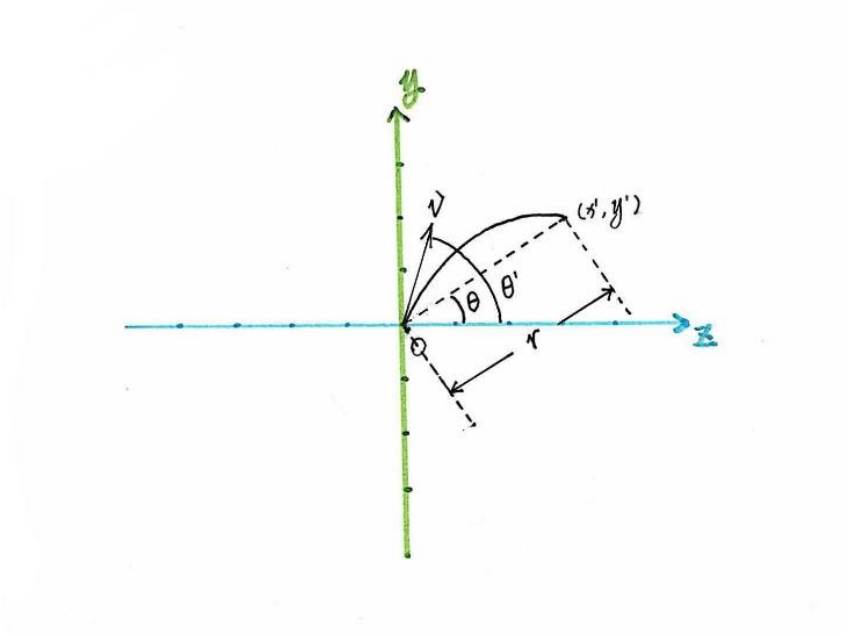
为了得出投掷体从 a 运动到 b 的最佳投掷角度（最省力）。我们明确的是：

- 人物始终正立，垂直于 xz 轴面。
- 重力 g 始终等于引力常量，y 轴负方向。
- z 轴方向的位移只受到 z 轴初始速度的影响。

我们先不去考虑 x 轴向的速度，把 a、b 点投射到 yz 轴构成的平面，得到 a'和 b'，先计算平面上 a'到达 b'点的最佳角度。最后再补上 x 轴向的速度（不会影响 z，y 轴向的速度）。



这样，我们就将问题转化到二维平面上来了：



在二维平面上，如图所示，落点坐标为（x', y'），设达到落点的时间为 t，由 x 和 y 方向的运动公式可以得出：

x' = V*cos(theta')*t;

y' = V*cos(theta')*t;

分三种情况讨论 x'<0; x'>0 和 x'=0 ，得：

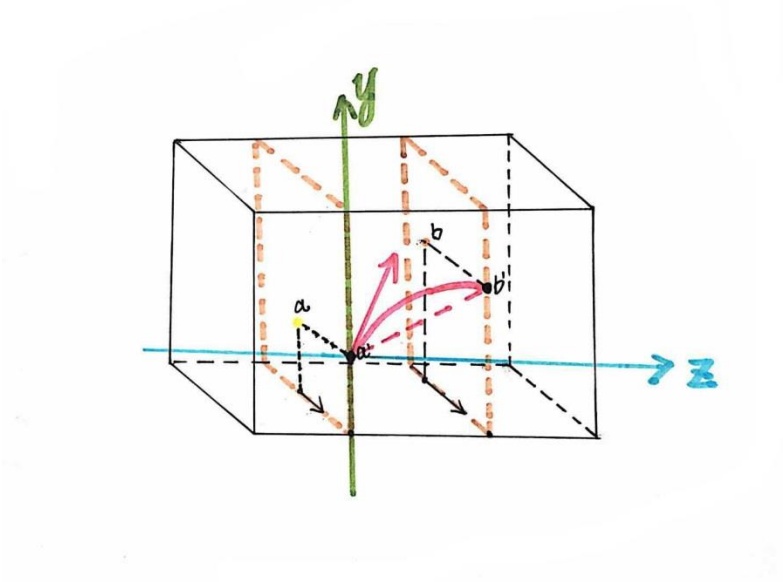
$$v^2 = \frac{g*x'^2}{x'*sin2\theta' - y'*cos2\theta' + y'} = \frac{g*r^2*(\cos\theta)^2}{\sin(2\theta' - \theta) + sin\theta}$$

若使 v 最小，需 sin(2θ'-θ) =1

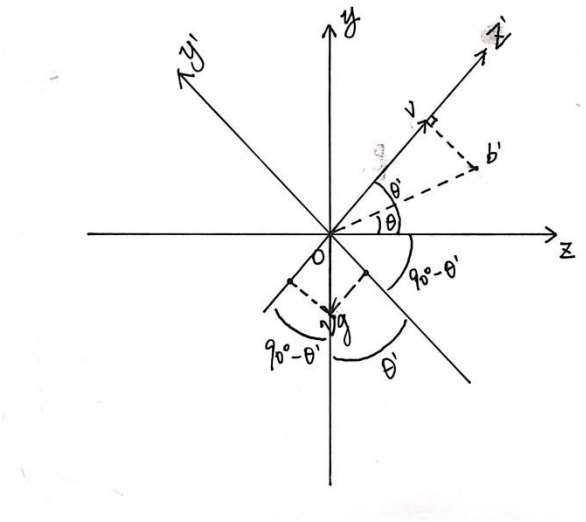
从而有 最佳角度为：

$$\theta' = \frac{PI}{4} + \frac{\theta}{2}$$

知道了最佳投掷角度，还需要知道投掷体的初始速度大小：



为了清晰，我们把上图的 yz 轴构成的平面截取下来:



将坐标轴旋转θ度，得到新的 y 轴 y'，z 轴 z'。将重力在新的 yz 轴上分解，则在到达 b'的时间里，y'轴向的位移全部都是重力作用的，得:

（*d=位移在 y'轴的分量）

$$t = \sqrt{\frac{2d}{g\cos\theta}}$$

从而 z'轴向，由 s =vt 易得速度 v:

$$vz' = \frac{Sz'}{t}$$

从而 y'轴向速度用于抵消重力分量的影响，易得:

$$vy' = \frac{1}{2}g\cos(90-\theta)t\text{^}2$$

最后再根据三维空间中 x 轴的位移量和 t，得到 x 轴向速度，求和即为所求速度。

（8）任务物品的搜集

任务物品发生器间隔一定时间发生任务物品，任务物品的存在有上限。任务物品不断移动，需要玩家去通过碰撞抓捕任务物品。

（9）玩家位置修正

在物理世界中，玩家每次进行平移旋转的操作都会产生一定的偏差，有时是因为受到的轻微的外力，有时是物理引擎的问题（transform.translate/rotate），最好的办法是直接对对象的 position、rotatation 属性进行操作。

我们将道路划分为三条跑道，玩家平移时只能在这三条跑道上切换，怪物和要收集的物品也只会在这条跑道上出现，平移时不是直接沿着某个坐标轴平移，而是直接计算出玩家应该在的位置，这样在每次位移之后都可以修正，但在长时间不操作玩家的情况下大概可以感受到视野可见的误差。

而旋转时，我们会记录每次旋转，旋转次数为偶次，当前道路就是与初始状态一致（竖向）；旋转次数为奇次，就是横向道路，以此来预测下一次旋转时的方向，从而在 rotation 上直接对游戏对象进行旋转，直接对角度值进行定量；而位置上会将玩家的位置与当前它所在的地图块的位置相减，算出玩家选择转向时，最靠近哪一个跑道，就自动修正位置到那个跑道上。

此外，对玩家的平移做了限制，只能在跑道的三条道上进行左右平移，设置了在中间跑道上时状态为 0，左移为-1，右移为 1，使其不会因为平移而撞墙，让玩家更专注于前行的道路。

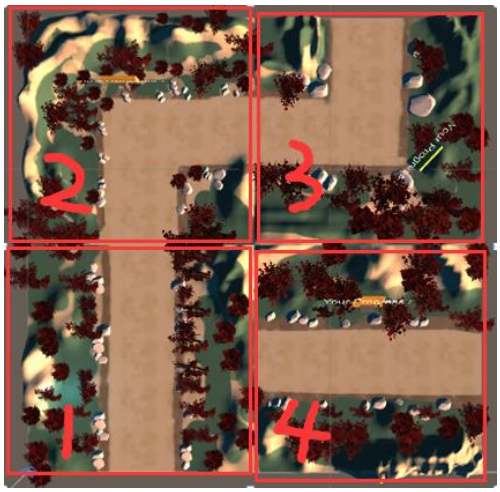
（10）怪物出现的位置和行为

由于我们把跑道划分成三道，那么怪物也理应只出现在这三条跑到的中央，所以自定义了一个函数 OutputPos()，它的输出只有三个值，并且是随机选择输出。由于这些位置是和自身地图块上跑道的分布有关，跑道竖向时，要对 x 方向做限制（只能取三条跑道中央的值），跑道横向时，要对 z 方向做限制， 而所以在生成地图块的同时，怪物的位置也会一同随机确定。

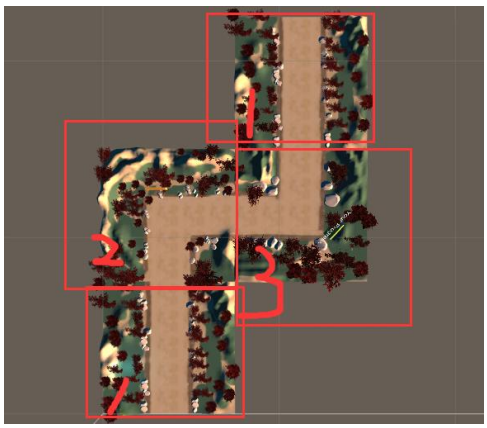
将怪物的位置向量与玩家位置向量相减，可以得到一个旋转的角度值，使怪物会一直面向玩家，并且当两者距离小于一定范围，怪物会执行 Attack 动画，从而表现出当玩家靠近怪物时，怪物会进行的攻击的…视觉效果。

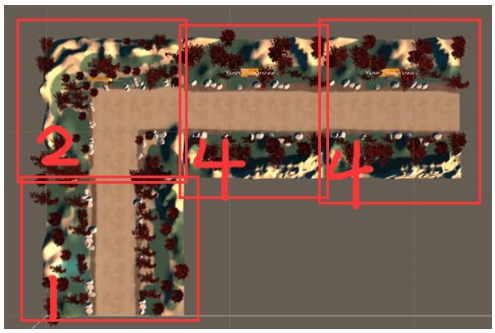
（11）无限地图随机生成

我们绘制了四块地图块，规定他们的规则，但不规定排列方式和数量，比如一号地图后面可以铺任意数量的一号地图本身和二号地图，二号后面可以铺任意数量的四号地图和三号地图，它们的选择是随机的，所以每次重新开始游戏后，你所穿过的地形都是不一样的。生成是随着游戏进行不断进行的，在道路上设置碰撞体（选择 is Trigger），可以感知到玩家已经到达某一块地图，然后对这些地图块进行增加或删减的操作。

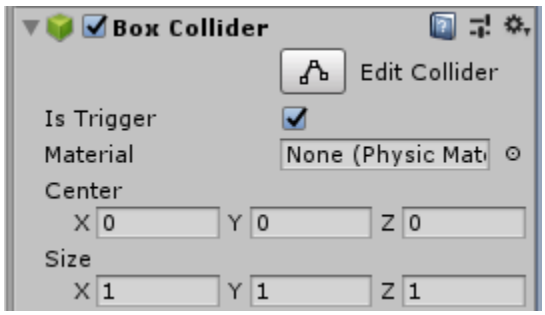


【四种地图块】

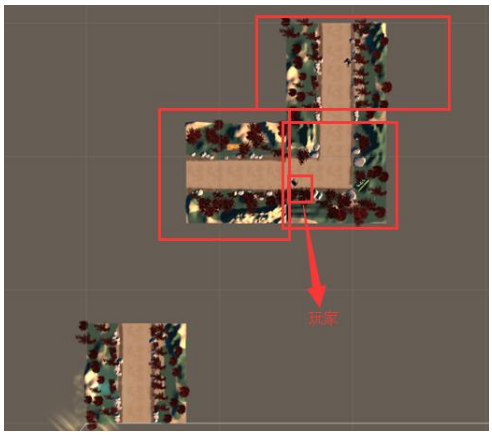


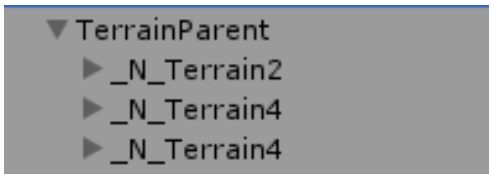


由于玩家视角有限，如果将（无限）地图在游戏开始就绘制好，会占用内存，所以我们选择在游戏进行中绘制一块块地图：我们给每一块地图的跑道的入口设置一个 Box Colider，选中 isTrigger，使其失去物理效果，但任然能被我们检测到碰撞（OnTriggerEnter）。当玩家进入一块新地图的时候，就会触发下一地图块预制体的向前生成。



我们将所有克隆的地图块都并入一个父对象（TerrainParents）中，使其长度不大于 3，一旦有新地图块生成，需要存入这个父对象，就将最先进入的地图块删除（先进先出），保持整个场景只有四块地图块在绘制，包括开始游戏时你所见到的第一块地图（用于重生），以及玩家在跑动过程中，它所在位置附近的三块地图块。





（12）判定游戏失败

在所有地图块跑道的两侧、怪物本身 都设置了 Box Colider，
而游戏主体本身带 Rididbody 和 Box Colider，当游戏主体与跑道或是怪物发生碰撞时，就会被判定游戏失败，需回到起点重新开始游戏。

（13）判定游戏暂停

选择游戏帮助时，会弹出一个窗口显示了一些基本操作。由于整个关卡二都是基于玩家的移动不断进行不断向前生成，所以只要将速度设为 0，即可实现游戏暂停的效果。

（14）游戏重新开始

玩家游戏失败后，会回到出发点重新开始游戏，这时我们复原游戏对象的位置到原点，并且将旋转角度重置，将原本存放地图块的父对象中的内容清空，游戏进度清空，即可回到原始状态。

（15）更新游戏进度并在场景中显示

在我们控制整个游戏运行的脚本（UI）中，有一个公共变量_ progress，这可以帮助任何脚本读取到当前游戏进度。当玩家与方块接触时，_progress 的值就会+10。我们在场景中设置了一些木板，上面放了 3Dtext 组件，用来显示并且会实时更新_progress 的值，当玩家路过他们的时候，可以帮助确认目前游戏进度。

六、项目后期

6.1 游戏测试

游戏开发也是一个不断迭代的过程，最初的设计也会因为自身技术原因，或者是玩家体验不好而被否决，游戏测试就成为了不可缺少的环节。我们通过自己测试关卡、小组讨论关卡的合理性、邀请同学友情测试来不断的完善游戏环节，修正游戏 BUG。

6.2 尚未解决的问题

（1）关卡二中，在第一块地图上旋转 180°会有跳跃到未知空间的 BUG
（2）关卡二中，在转角旋转时，碰撞体的旋转会扫到旁边的道路，如果这是恰好出现怪物，则判定死亡了。

6.3 项目总结

在本次游戏开发的过程中，我们对 Unity 的各种功能都有了深刻了解和应用，包括利用 UI 组件做特效，摄像机镜头渲染，对游戏对象的基本操作和信息获取，动画及动画机的使

用，地形制作，音效播放，脚本间各变量传递，场景转换等。也学习到了游戏开发的一般流程以及注意点，比如任何场景都会有一个 manager 脚本控制整个游戏进程，对三维数学在游戏中的应用有了更深的学习，比如点和矢量和坐标系之间的转换。