

绪论

- 1.虚拟现实发展简史
- 2.虚拟现实与增强现实
- 3.虚拟现实与人工智能

PART 1 入门篇- 新手的unity3d漫游指南

写作者：

第1章 初识神兵- 为什么选择Unity3d

主要内容：常用3D引擎对比分析，说明使用Unity3d进行VR/AR开发的优缺点

1.1 常用3D引擎对比分析

1.2 Unity的发展史

1.3 Unity软件安装、授权与服务

1.3.1 在Windows下的安装

1.3.2 在Mac 下的安装

1.3.3 Unity授权类型

1.3.4 Unity服务简介

第2章 蹒跚学步- 开始使用Unity

2.1 Unity 编辑器界面

2.1.1 界面布局

2.1.2 工具栏

2.1.3 菜单栏

2.1.4 工作视图

2.2 Unity Asset Store 开发资源介绍

2.2.1 Asset Store简介

2.2.2 Asset Store的基本使用

2.3 游戏对象、组件和Prefabs

2.3.1 游戏对象简介

2.3.2 组件简介

2.3.3 Prefabs简介

主要内容：Unity3d的基本使用介绍，包括界面，基本概念（物体、组件等等）

第3章 无往不利- 在Unity中使用C#进行编程

3.1 3D数学基础知识

3.1.1 3D坐标系

3.1.2 向量

3.1.3 矩阵

3.1.4 齐次坐标

3.1.5 四元数

3.2 C# 基本语法

3.2.1 基本数据类型

3.2.2 变量和函数

3.2.3 控制流程

3.2.4 类和方法

3.3 Unity脚本开发基础

- 3.3.1 创建脚本
- 3.3.2 MonoDevelop编辑器及Visual Studio
- 3.3.3 实战-如何访问游戏对象
- 3.3.4 实战-如何访问组件
- 3.3.4 UnityEvents
- 3.3.5 常用API

PART 2 进阶篇- VR/AR开发必知必会的Unity3d技能

第4章 创建一个新世界-Modeling

- 4.1 常见的3D建模软件
- 4.2 模型的创建、重拓扑和展UV
 - 4.2.1 3D模型的创建
 - 4.2.2 3D模型的重拓扑
 - 4.2.3 展UV
- 4.3 模型贴图
 - 4.3.1 PBS理论简介
 - 4.3.2 实战-使用Substance Designer和Painter制作材质
 - 4.3.2 实战-给3D模型添加贴图
- 4.4 为人物模型创建骨骼和蒙皮
- 4.5 将3D模型和动画导入到场景中
 - 4.5.1 实战- 将3D模型导入到场景中
 - 4.5.2 实战- 将3D人物角色动画导入到场景中

4.6 资源的导入和导出

4.6.1 Unity Asset Store资源的导入

4.6.2 资源的导出

主要内容：介绍如何创建基本的场景和人物

第5章 让画面栩栩如生

5.1 Shuriken粒子系统

5.1.1 Shuriken概述

5.1.2 实战-火焰粒子效果案例

5.1.3 实战-烟火粒子效果案例

5.2 Shader的使用

5.2.1 Shader概述

5.2.2 Unity内建Shader介绍

5.2.3 创建自己的Shader

第6章 有了光就有了一切- Enlighten

6.1 Unity光照系统 (Enlighten)介绍

6.2 Light组件简介

6.2.1 常见的光源类型

6.2.2 Light inspector参数设置

6.2.3 Shadows

6.3 全局光照

6.3.1 全局光照简介

6.3.2 烘焙

6.3.3 Lightmaps

6.3.4 Light Probe的使用

6.3.5 Refelection Probe的使用

6.4 实战-给游戏场景设置光照系统

第7章 让角色动起来- Mecanim动画系统

7.1 Mecanim概述

7.2 资源的导入

7.2.1 获取人形网格模型

7.2.2 导入动画

7.2.3 分解动画

7.2 配置人形角色动画

7.2.1 创建Avatar

7.2.2 配置Avatar

7.2.3 设置Muscle参数

7.2.4 设置Avatar Mask

7.2.5 角色动画的重定向

7.3 实战- 在游戏中使用人形角色动画

7.3.1 创建Animator Controller

7.3.2 设置基础状态机

7.3.3 添加更复杂的动画

7.3.4 使用Animator组件

7.3.5 编写控制角色动画的脚本

第8章 有了目标才有方向- Pathfinding

8.1 寻路系统简介

8.1.1 Unity中的寻路系统

8.1.2 寻路系统内部工作原理

8.1.3 创建NavMesh

8.1.4 NavMesh 烘焙设置

8.1.5 创建NavMesh Agent

8.1.6 创建NavMesh Obstacles

8.1.7 自动构建Off-Mesh Links

8.1.8 加载多个NavMeshes

8.1.9 配合其它组件使用NavMesh Agent

8.2 寻路系统相关参数详解

8.2.1 NavMesh Agent

8.2.2 NavMesh Obstacle

8.2.3 Off-Mesh Link

8.3 实战-在游戏中使用寻路系统

第9章 真实世界的物理法则- Physics

9.1 物理系统简介

9.2 物理系统的相关组件介绍

9.2.1 Rigidbody组件

9.2.2 Character Controller组件

9.2.3 碰撞体组件

9.2.4 布料组件

9.2.5 关节组件

9.2.6 其它组件

9.3 实战-给游戏添加物理系统

9.3.1 给物体添加Rigidbody组件

9.3.2 给人物添加Character Controller组件

第10章 且听风吟- Audio

10.1 Audio系统概述

10.2 Audio系统相关组件介绍

10.2.1 Audio Source

10.2.2 Audio Reverb Zone

10.2.3 Audio Mixer

10.3 实战-给游戏添加背景音乐和音效

第11章 一个人的世界很孤单- Network

11.1 UNET简介

11.1.1 授权服务器

11.1.2 非授权服务器

11.1.3 High Level API

11.1.4 Transport Layer API

11.1.5 WebGL Support

11.1.6 Internet Services

11.1.7 NetWorkView

11.2AssetBundle工作流程

11.2.1 如何下载AssetBundle

11.2.2 加载AssetBundle

11.2.3 从AssetBundle 中加载Assets

11.2.4 AssetBundle和Asset的卸载

11.3 实战-创建一个简单的多人在线系统

11.3.1 Photon+ 简介

11.3.2 游戏的目标

11.3.3 创建Unity项目并导入PUN

11.3.4 创建基本的Lobby场景

11.3.5 创建游戏场景

11.3.6 创建游戏角色和摄像机

11.3.7 添加网络功能

11.3.8 初始化角色和场景切换

11.3.9 玩家用户界面

第12章 没有菜单怎么点菜- UGUI界面系统

12.1 UGUI简介

12.2 常用UGUI控件简介

12.2.1 Canvas组件简介

12.2.2 视觉组件简介

12.2.3 交互组件简介

12.2.4 自动布局

12.2 实战-创建一个简单的3D菜单

12.2.1 创建一个Canvas

12.2.2 添加文本

12.2.3 添加图像

12.2.4 添加布局组

12.2.5 添加按钮

第13章 性能优化与脚本调试

13.1 性能优化的原则和常用方法

13.1.1 通用的优化方案

13.1.2 VR平台的优化

13.2 遮挡剔除

13.2.1 概述及其应用示例

13.2.2 使用Occlusion Area组件

13.2.3 使用Occlusion Portals组件

13.2.4 设置烘焙参数

13.2 Profiler

13.3 Frame Debugger

PART 3 实战篇- 虚拟现实开发实战

第14 章 虚拟现实开发入门

14.1 虚拟现实的技术根基

14.1.1 立体显示技术

14.1.2 多感知自然交互技术

14.1.3 3D场景和人物建模技术

14.1.4 3D全息投影技术

14.1.5 脑机接口技术

14.2 主流的AR/VR设备及开发SDK

14.2.1 Oculus Rift, Gear VR和Oculus SDK

14.2.2 HTC Vive和SteamVR

- 14.2.3 Sony PSVR
- 14.2.4 Google Daydream VR
- 14.2.5 微软HoloLens
- 14.2.6 高通Vuforia SDK

14.3 自然交互设备及SDK

- 14.3.1 Leap Motion
- 14.3.2 Intel Realsense
- 14.3.3 科大讯飞语音识别SDK
- 14.3.4 uSense SDK

14.4 AR/VR应用开发的基本流程

- 14.4.1 AR/VR应用开发基本流程
- 14.4.2 AR/VR应用开发的常用工具

14.5 AR/VR应用的设计

第15章 实战开发HTC Vive平台上的《切西瓜》游戏

15.1 HTC Vive平台开发概述

- 15.1.1 HTC Vive设备及平台简介
- 15.1.2 Lighthouse技术原理
- 15.1.3 HTC Vive手柄交互详解
- 15.1.4 SteamVR插件简介
- 15.1.5 VRTK插件简介

15.2 VR版《切西瓜》的产品策划

15.3 配置HTC Vive的开发测试环境

- 15.3.1 安装HTC Vive硬件
- 15.3.2 安装Vive和Steam软件

15.3.3 设置并打开SteamVR

15.3.4 运行SteamVR的测试场景

15.4 在Unity中导入所需的美术资源

15.5 导入SteamVR和VRTK插件

15.5.1 从AssetStore上下载并导入SteamVR插件

15.5.2 从Github上下载并导入VRTK插件

15.6 游戏的UI设计与开发

15.6.1 VR游戏中UI设计的注意事项

15.6.2 VR版《切西瓜》的UI设计思路

15.6.3 添加3D版的开始菜单

15.6.4 添加手柄和菜单的交互事件

15.7 添加游戏的核心机制

15.7.1 让西瓜从空中随机落下

15.7.2 让手柄化作手中的利刃

15.7.3 添加游戏输赢的机制

15.8 添加背景音乐和音效

15.8.1 添加背景音乐

15.8.2 添加交互时的音效

15.9 将游戏发布到Vive平台

15.9.1 Vive 平台介绍

15.9.2 注册为Vive的开发者

15.9.3 上传并提交游戏

15.9.4 等待审核发布

15.10 将游戏发布到Steam 平台

15.10.1 Vive 平台介绍

15.10.2 注册为Vive的开发者

15.10.3 上传并提交游戏

15.10.4 等待审核发布

第16章 实战开发Google Daydream VR平台上的3D虚拟样板间应用

16.1 Google Daydream VR平台开发概述

16.1.1 Google Daydream VR设备及平台简介

16.1.2 Google Daydream VR中的交互

16.2 3D虚拟样板间应用产品策划

16.3 配置Google Daydream VR的开发测试环境

16.3.1 准备好相关的开发硬件

16.3.2 系统和软件的安装

16.3.3 运行Daydream VR的测试场景

16.4 在Unity中导入所需的美术资源

16.5 游戏的UI设计与开发

16.5.1 3D虚拟样板间应用的UI设计思路

16.5.2 添加大厅中的菜单

16.5.3 添加手柄和菜单的交互事件

16.5.4 添加手柄上的提示信息

16.6 添加游戏互动机制

16.6.1 添加场景漫游功能

16.6.2 添加地板材质切换功能

16.6.3 添加和场景中物体的交互

16.7 添加背景音乐和音效

16.7.1 添加背景音乐

16.7.2 添加交互时的音效

16.8 将产品发布到Google Daydream VR平台

16.8.1 Daydream VR平台介绍

16.8.2 注册为Daydream VR的开发者

16.8.3 上传并提交游戏

16.8.4 等待审核发布

第17章 实战开发Oculus Rift平台上的《荒野求生》游戏

17.1 Oculus Rift平台开发概述

17.1.1 Oculus Rift设备及平台简介

17.1.2 Oculus Rift所使用的空间定位技术

17.1.3 Oculus Touch手柄交互详解

17.2 《荒野求生》产品策划

17.2.1 《荒野求生》产品的设计思路

17.2.2 《荒野求生》中的交互方式

17.3 配置Oculus Rift的开发测试环境

17.3.1 安装Oculus Rift 硬件

17.3.2 安装Oculus Home

17.3.3 在Unity中运行一个基本测试场景

17.4 在Unity中导入所需的美术资源

17.5 游戏的UI设计与开发

17.5.1 《荒野求生》的UI设计思路

17.5.2 添加3D版的开始菜单

17.5.3 添加手柄和菜单的交互事件

17.6 添加游戏交互机制

- 17.6.1 让野兽来得更猛烈些吧
- 17.6.2 给主角一把火力无穷的武器
- 17.6.3 添加游戏的输赢机制

17.7 添加背景音乐和音效

- 17.7.1 添加背景音乐
- 17.7.2 添加开火和野兽咆哮的音效

17.8 将产品发布到Oculus Home平台

- 17.8.1 Oculus Home平台介绍
- 17.8.2 注册成为Oculus的开发者
- 17.8.3 上传并提交游戏
- 17.8.4 等待审核发布

第18章 实战开发HoloLens平台的《精灵宝可梦》游戏

18.1 HoloLens平台开发概述

- 18.1.1 HoloLens设备及平台简介
- 18.1.2 HoloLens中的语音交互
- 18.1.3 HoloLens中的手势动作

18.2 HoloLens版的《精灵宝可梦》产品策划

18.3 配置HoloLens的开发测试环境

- 18.3.1 准备好HoloLens硬件
- 18.3.2 安装Visual Studio和其它软件
- 18.3.3 设置并运行HoloLens的测试场景

18.4 在Unity中导入所需的美术资源

18.5 游戏的UI设计与开发

18.5.1 HoloLens应用中UI设计的注意事项

18.5.2 AR版《精灵宝可梦》的UI设计思路

18.5.3 添加场景中的菜单

18.6 添加游戏的核心机制

18.6.1 让世界充满可爱的口袋妖怪

18.6.2 用手抓住你的最爱

18.7 添加背景音乐和音效

18.7.1 添加背景音乐

18.7.2 添加交互时的音效

18.8 将产品发布到UWP平台

18.8.1 微软UWP平台介绍

18.8.2 注册为微软的开发者

18.8.3 上传并提交游戏

18.8.4 等待审核发布

第19章 实战使用Unity和高通Vuforia SDK开发《口袋动物园》AR游戏

19.1 高通Vuforia SDK简介

19.2 《口袋动物园》的产品策划

19.3 配置开发测试环境

19.3.1 在Unity中导入Vuforia SDK

19.3.2 运行Vuforia的测试场景

19.4 在Unity中导入所需的美术资源

19.5 游戏的UI设计与开发

19.5.1 AR版《口袋动物园》的UI设计思路

19.5.2 设计《口袋动物园》的UI

19.6 添加游戏互动机制

19.7 添加背景音乐和音效

19.7.1 添加背景音乐

19.7.2 添加音效

19.8 将游戏发布到Android应用商城

第20章 综合实战-使用Photon+、科大讯飞SDK和Leap Motion SDK开发HTC Vive平台上的《VR虚拟聊天室》应用

20.1 综合实战概述

20.1.1 科大讯飞SDK简介

20.1.2 Leap Motion SDK简介

20.2 《VR虚拟聊天室》产品策划

20.3 配置HTC Vive的开发测试环境

20.4 在Unity中导入所需的美术资源

20.5 导入SteamVR和VRTK插件

20.6 添加科大讯飞的语音识别SDK

20.7 添加Leap Motion SDK

20.8 游戏的UI设计与开发

20.9 添加游戏互动逻辑机制

20.9.1 添加Photon+插件

20.9.2 添加对话功能

20.10 添加背景音乐和音效

20.10.1 添加背景音乐

20.10.2 添加游戏中的音效

20.11 发布游戏到Vive平台

20.11.1 Vive 平台介绍

20.11.2 注册为Vive的开发者

20.11.3 上传并提交游戏

20.11.4 等待审核发布

20.12 发布游戏到Steam平台

20.12.1 Vive 平台介绍

20.12.2 注册为Vive的开发者

20.12.3 上传并提交游戏

附录