

高等学校数字媒体技术系列教材

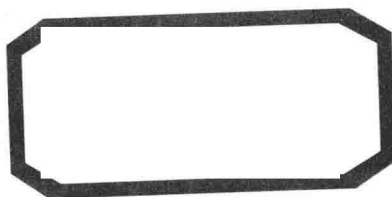
# 游戏设计 与实现

*Game Design  
and  
Implementation*

王鑫 潘翔 编著

高等教育出版社

高等学校



才

# 游戏设计与实现

Youxi Sheji yu Shixian

王 鑫 潘 翔 编著

高等教育出版社·北京

## 内容提要

本书介绍了游戏设计与实现所需掌握的各方面基础知识,侧重于给读者介绍游戏开发所需的各种基础理论和编程工具,进而引导读者掌握基于游戏引擎的游戏开发思想,通过较完整的游戏案例来解释游戏设计与实现的整个流程。本书所选用游戏引擎或开源代码库均为 OGRE、Unity3D 等主流引擎。

本书在每章后均附有习题,适合作为计算机科学与技术、数字媒体技术、软件工程和游戏动漫等专业高年级本科生和研究生的教材;同时,有一定编程基础知识的工程人员,也可以通过本书来学习游戏开发所需的各方面概要知识。

## 图书在版编目(CIP)数据

游戏设计与实现 / 王鑫, 潘翔编著. -- 北京: 高等教育出版社, 2016. 5

ISBN 978-7-04-045063-7

I. ①游… II. ①王… ②潘… III. ①游戏-软件设计-高等学校-教材 IV. ①TP311.5

中国版本图书馆 CIP 数据核字(2016)第 050902 号

策划编辑 韩 飞      责任编辑 韩 飞      封面设计 王 琰      版式设计 杜微言  
插图绘制 杜晓丹      责任校对 刘 莉      责任印制 韩 刚

出版发行 高等教育出版社

社 址 北京市西城区德外大街 4 号

邮政编码 100120

印 刷 北京东君印刷有限公司

开 本 787mm × 1092mm 1/16

印 张 17

字 数 380 千字

购书热线 010-58581118

咨询电话 400-810-0598

网 址 <http://www.hep.edu.cn>

<http://www.hep.com.cn>

网上订购 <http://www.hepmall.com.cn>

<http://www.hepmall.com>

<http://www.hepmall.cn>

版 次 2016 年 5 月第 1 版

印 次 2016 年 5 月第 1 次印刷

定 价 32.00 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换

版权所有 侵权必究

物料号 45063-00

# 数字课程资源使用说明

与本书配套的数字课程资源发布在高等教育出版社易课程网站，请登录网站后开始课程学习。

## 一、注册/登录

访问 <http://abook.hep.com.cn/1875809>，点击“注册”，在注册页面输入用户名、密码及常用的邮箱进行注册。已注册的用户直接输入用户名和密码登录即可进入“我的课程”页面。

## 二、课程绑定

点击“我的课程”页面右上方“绑定课程”，正确输入教材封底防伪标签上的 20 位密码，点击“确定”完成课程绑定。

## 三、访问课程

在“正在学习”列表中选择已绑定的课程，点击“进入课程”即可浏览或下载与本书配套的课程资源。刚绑定的课程请在“申请学习”列表中选择相应课程并点击“进入课程”。

## 四、资源说明

与本书配套的易课程数字课程资源包括电子教案、室内场景编辑器介绍、学生作品等，以便读者学习使用。



## 数字媒体技术系列教材编委会

主任：卢 苇（北京交通大学）

副主任：吴中海（北京大学）

委员：（按姓氏拼音排序）

董槐林（厦门大学）

贾金原（同济大学）

李 祥（东华理工大学）

王崇文（北京理工大学）

王方石（北京交通大学）

王万良（浙江工业大学）

叶德健（复旦大学）

曾智勇（福建师范大学）

朱 青（北京工业大学）

# 序

数字媒体包括了文字、图形、图像、音频、视频和动画等各种媒体形式，互联网、移动、IPTV、数字出版等各种传播形式，以及媒体的采集、存取、加工和分发的数字化过程。近20年来，随着互联网应用的普及，数字媒体技术发展迅速，已成为推动我国文化创意产业和电子信息产业发展的重要力量。为此，迫切需要培养一支适应时代要求和产业需求、具有创新能力、复合交叉型的数字媒体技术专业队伍。

数字媒体是信息技术和媒体艺术相结合的一门交叉学科，美国麻省理工学院的媒体艺术与科学实验室是该学科的探索者和领跑者。我国早期的数字媒体技术专业人才培养起源于计算机辅助设计与图形学专业方向和数字图像处理专业方向。2004年教育部批准设置数字媒体技术专业后，许多高校纷纷开设数字媒体技术专业，面向动画、游戏、新媒体、虚拟现实等领域培养本科生或研究生，受到了社会和企业的欢迎。

为了进一步推动数字媒体技术专业人才培养，教育部示范性软件学院建设工作办公室于2010年3月成立了数字媒体技术专业规范研制专家组，旨在进一步加强高等学校数字媒体技术专业建设和人才培养，组织高水平教材的编写工作。2011年10月，专家组编制完成《高等学校数字媒体技术专业规范》；同时依据专业规范，委托有丰富教学实践经验的教师编写一批专业基础课程和专业核心课程教材。这套教材正是过去两年来的重要工作成果之一。

数字媒体技术的发展日新月异，新概念、新方法、新技术层出不穷，这一切需要我们与时俱进地反映到大学课程的教学计划和教学内容中。相信这套教材的出版能够起到积极推动各高校数字媒体技术专业建设、提高教学水平和人才培养质量的作用。

希望广大教师在教学过程中对教材提出宝贵的意见和建议，使其在广泛使用的过程中不断完善。

示范性软件学院联盟

2015年3月

# 前 言

本书介绍了游戏设计与实现所需掌握的各方面基础知识，侧重于给读者介绍游戏开发所需要的各种基础理论和编程工具，通过较完整的游戏案例来解释游戏设计与实现的整个流程，进而引导读者掌握基于游戏引擎的游戏开发思想。在笔者给数字媒体技术专业学生讲授“游戏开发课程”过程中，发现市面上既能兼顾游戏开发理论知识又能结合主流游戏引擎讲解游戏案例的图书非常少。而大学游戏课程需要给学生传授的首先是游戏开发基础理论知识，如本书第2章的事件消息驱动的应用程序开发，第4章的各类三维图形坐标变换；其次，只掌握基础理论知识还不够，还需要能理论联系实际，使用目前市面上主流的、开源的、适用于教学用途的游戏开发引擎来制作出完整的游戏案例，如本书第5章的一个基于 Unity 的3D 射击游戏。只有这样才算真正掌握游戏设计与实现技术，达到“游戏开发课程”的教学目标。基于这样的想法，笔者在教学过程中积累素材，逐渐形成了本书的各个章节内容。

游戏设计与实现，涉及诸多方面的内容：计算机图形学、图像处理、矩阵计算、物理仿真、音频特效、人工智能、计算机网络和移动开发等，每项内容都可以独立写成一本书。而本书定位于通用教材类书籍，所以在内容取舍上去掉了物理仿真、移动开发和人工智能。计算机网络等方面的内容，对这些内容有学习需要的读者可以在市面上找到更专注在这些方面的书籍。

## 本书内容概要

**第1章 游戏开发概述** 对游戏开发历史、过程、类别做了一个系统的概论，并以一个游戏循环代码框架来给读者直观认识游戏开发的本质。

**第2章 Window 游戏开发基础** Windows 事件消息驱动的编程机制比较复杂，但游戏开发并不需要掌握细节，通过讲解 WinMain() 函数的过程来紧抓核心知识点和总体 Win32 应用程序的骨干结构，让读者理解 Win32 编程的精髓。

**第3章 2D 游戏引擎编程接口** 精灵动画、碰撞检测、地图技术是2D 游戏引擎编程的核心技术模块，每种模块的细分知识均有详细介绍。

**第4章 3D 游戏编程的数学基础** 3D 游戏编程的难点在于如何理解坐标变换。本章重点将3D 游戏编程所需的理论知识结合代码实现，给读者进一步强化。其中3D 空间中方位与角位移小节中做了较多的介绍，四元数这类知识在高等数学不做介绍的，在本章做了重点介绍。本章还剖析了 OGRE 引擎基础数据代码库的实现，以代码案例形式给读者降低理解难度，加深理解深度。

**第5章 一个基于 Unity3D 的射击游戏** Unity3D 是目前主流的游戏开发引擎，其有免费教

育版本，可以让学生零成本学习。本章以一个完整的小型射击游戏如何一步步在 Unity3D 引擎中被制作出来为纲要，串联起了掌握 Unity3D 所需要的各种核心组件。本章的目标是教会读者如何使用一个游戏引擎。

**第 6 章 3D 游戏引擎编程接口** 本章目标是带着读者深入到游戏引擎架构本身去学习商业级别游戏引擎主要由哪些核心模块、子系统所构成。讲解了引擎子系统之间的关系、层别。游戏引擎中的场景管理和资源管理是游戏引擎中穿针引线的模块，因此本章对这两大模块也特别做了深入讲解，场景管理部分还结合了开源引擎 Unreal 的 UDK 场景编辑器来做介绍。

**第 7 章 游戏场景管理与加速计算** 本章对游戏场景管理与图形加速计算方面的核心算法做了详细的描述，这些算法是游戏引擎底层技术的思想指导。通过本章的学习，可以帮助读者掌握这些底层算法细节，自己去撰写底层游戏引擎代码。

**第 8 章 游戏用户界面** 游戏的本质就是玩家与界面不断交互的图像展示过程。本章对各种游戏界面与设备做了系统介绍，特别介绍了一些新兴的交互界面技术比如 Kinect、Leap Motion 等。

本书由浙江工业大学计算机学院王鑫、潘翔两位老师共同执笔。浙江工业大学计算机学院王万良院长、中国传媒大学游戏设计系陈京炜主任对书稿提出了很多宝贵的意见，在此表示一并感谢。

本书适合作为计算机科学与技术、数字媒体技术、软件工程和游戏动漫等专业高年级本科生和研究生的教材；同时，有一定编程基础知识的工程人员，也可以通过本书来学习游戏开发所需的各方面概要知识。限于编者水平有限，书中难免有错误之处，敬请批评指正。

编 者  
2016 年 2 月



# 目 录

<b>第 1 章 游戏开发概述</b> .....	1	2.3.1 总是从 WinMain() 开始	24
1.1 计算机游戏历史 .....	1	2.3.2 程序剖析 .....	25
1.2 什么是游戏 .....	2	2.3.3 设置 MessageBox 样式 .....	26
1.3 游戏的分类 .....	2	2.4 Windows 应用程序基本结构 .....	28
1.3.1 角色扮演游戏 .....	2	2.5 创建 Windows 窗口类 .....	28
1.3.2 网页游戏 .....	2	2.6 注册 Windows 类 .....	34
1.3.3 动作游戏 .....	3	2.7 创建窗口 .....	34
1.3.4 冒险游戏 .....	4	2.8 事件句柄 .....	37
1.3.5 策略模拟类游戏 .....	4	2.9 主事件循环 .....	42
1.4 游戏设计的相关知识 .....	5	2.10 产生一个实时事件循环 .....	47
1.4.1 计算机图形学 .....	5	本章小结 .....	48
1.4.2 艺术设计 .....	6	习题 2 .....	48
1.4.3 人工智能 .....	6	参考文献 .....	49
1.4.4 人机交互 .....	7	<b>第 3 章 2D 游戏引擎编程接口</b> .....	50
1.5 案例分析：一个典型的游戏循环		3.1 精灵动画 .....	50
代码框架 .....	8	3.1.1 基本概念 .....	50
1.5.1 典型游戏循环结构 .....	8	3.1.2 图像合成 .....	51
1.5.2 打字游戏实例 .....	12	3.1.3 精灵动画的闪烁问题 .....	52
本章小结 .....	19	3.2 2D 地图技术 .....	53
习题 1 .....	19	3.2.1 固定地图 .....	53
参考文献 .....	19	3.2.2 滚屏地图 .....	54
<b>第 2 章 Windows 游戏开发基础</b> .....	20	3.3 2D 碰撞检测 .....	54
2.1 事件模型 .....	20	3.3.1 地图格子划分检测 .....	54
2.2 匈牙利符号表示法 .....	21	3.3.2 外包围盒检测 .....	55
2.2.1 变量的命名 .....	22	3.3.3 像素检测 .....	55
2.2.2 函数的命名 .....	22	3.4 二维游戏案例 .....	56
2.2.3 类型和常量的命名 .....	22	3.4.1 界面设计 .....	57
2.2.4 类的命名 .....	22	3.4.2 程序流程 .....	57
2.2.5 参数的命名 .....	23	3.4.3 关键数据结构 .....	61
2.3 最简单的 Windows 程序 .....	23	3.4.4 核心函数 .....	61

本章小结 .....	63	4.7.5 旋转 .....	104
习题3 .....	63	4.7.6 投影 .....	107
<b>第4章 3D 游戏编程的数学基础</b> .....	64	本章小结 .....	109
4.1 坐标系 .....	64	习题4 .....	109
4.1.1 笛卡儿坐标系 .....	64	参考文献 .....	110
4.1.2 常用坐标系 .....	65	<b>第5章 一个基于 Unity 的3D 射击</b>	
4.1.3 坐标系嵌套 .....	67	<b>游戏</b> .....	111
4.1.4 坐标系转换 .....	67	5.1 Unity 的工程组织结构 .....	111
4.2 向量 .....	68	5.2 新建工程, 导入资源 .....	112
4.2.1 向量的数学定义 .....	68	5.2.1 新建工程 .....	112
4.2.2 向量的几何定义 .....	68	5.2.2 Unity 工作界面介绍 .....	113
4.3 矩阵 .....	69	5.2.3 Unity 资源导入 .....	117
4.3.1 矩阵的数学定义 .....	69	5.3 创建开始场景 .....	118
4.3.2 矩阵的几何解释 .....	70	5.3.1 创建开始场景 .....	118
4.3.3 矩阵和旋转 .....	71	5.3.2 脚本讲解 .....	120
4.4 3D 空间中方位与角位移 .....	73	5.4 创建设置场景 .....	121
4.4.1 什么是方位 .....	73	5.4.1 创建设置场景 .....	121
4.4.2 矩阵形式 .....	74	5.4.2 脚本讲解 .....	122
4.4.3 欧拉角 .....	75	5.5 创建游戏场景 .....	125
4.4.4 四元数 .....	75	5.5.1 场景创建 .....	125
4.4.5 各种方法比较 .....	79	5.5.2 地形 .....	125
4.5 OGRE 中的数学应用 .....	80	5.5.3 城墙底部 .....	126
4.5.1 平移 .....	80	5.5.4 城墙上部 .....	127
4.5.2 缩放 .....	83	5.5.5 平行光光源和点光源 .....	128
4.5.3 旋转 .....	83	5.5.6 袭击物爆炸时的几种	
4.5.4 其他 .....	88	音效 .....	130
4.5.5 总结 .....	89	5.5.7 各种袭击物 .....	132
4.6 OGRE 简介 .....	89	5.5.8 子弹 .....	142
4.6.1 OGRE 的设计理念 .....	89	5.5.9 手枪 .....	143
4.6.2 OGRE 的设计亮点 .....	89	5.5.10 沙尘暴效果 .....	146
4.6.3 OGRE 子系统概述 .....	93	5.5.11 技能: 流星火雨 .....	146
4.7 OGRE 简单数学应用实例实现 .....	96	5.5.12 技能: 极度冰寒 .....	149
4.7.1 OGRE 配置 .....	97	5.5.13 主摄像机 .....	151
4.7.2 OGRE 中的重要知识点 .....	100	5.5.14 为场景添加索引 .....	153
4.7.3 平移 .....	100	5.6 生成游戏 exe .....	153
4.7.4 缩放 .....	103	5.7 游戏运行效果 .....	154

本章小结	155	7.2.1 潜在可见集的构造	201
习题5	155	7.2.2 使用潜在可见集	203
<b>第6章 3D 游戏引擎编程接口</b>	156	7.2.3 潜在可见集的其他编码方法	204
6.1 3D 游戏引擎框架分析	157	7.2.4 流行的 PVS 计算方法	205
6.1.1 硬件层、设备驱动层、操作系统层、第三方 SDK 与中间件	157	7.3 遮掩剔除技术	207
6.1.2 平台无关层、核心系统层、资源管理器层、用户界面设备层	160	7.3.1 遮掩体	209
6.1.3 绘制器层、调优调试工具层、碰撞处理与物理计算层	161	7.3.2 选择遮掩物	210
6.1.4 动画层、声音层、网络层、Gameplay 基础系统层	164	7.3.3 混合型遮掩物选择方法	210
6.1.5 游戏特定相关系统层	166	本章小结	210
6.2 场景管理子系统	166	习题7	211
6.3 基于 UDK 的游戏场景设计	169	参考文献	212
6.3.1 场景搭建简介	169	<b>第8章 游戏用户界面</b>	213
6.3.2 UDK 场景编辑器介绍	170	8.1 游戏界面设备种类	214
6.3.3 地形模式基础操作	171	8.1.1 命令行界面	214
6.3.4 摄像机模式基础操作	174	8.1.2 图形用户界面	215
6.3.5 场景光源光照设置	176	8.1.3 基于触屏技术的图形用户界面	215
6.3.6 游戏场景构建	178	8.1.4 语音识别界面	216
6.4 资源管理子系统	181	8.1.5 基于摄像机的界面	217
6.4.1 脱机资源管理	182	8.1.6 基于传感器的界面	219
6.4.2 运行时的资源管理	186	8.2 人机交互编程技术	220
本章小结	190	8.2.1 CEGUI	220
习题6	190	8.2.2 Kinect	222
参考文献	191	8.2.3 Leap Motion	233
<b>第7章 游戏场景管理与加速计算</b>	192	8.3 三维游戏界面设计	238
7.1 场景可视化处理	192	8.3.1 用户界面设计原则	238
7.1.1 八叉树	192	8.3.2 用户界面设计过程	239
7.1.2 BSP 二叉空间分割	195	8.3.3 用 HID 进行交互	239
7.1.3 BVH 层次包围盒技术	199	本章小结	240
7.2 PVS 技术	201	习题8	241
		参考文献	241
		<b>参考答案</b>	242

# 第 1 章 游戏开发概述

## 1.1 计算机游戏历史

游戏这个名词，实际上是先于计算机名词的出现而存在的，如丢手绢、走迷宫等。而随着数字娱乐的兴起和大众化，游戏也有了新的概念。本书讨论的是计算机游戏（Computer Games）。和普通游戏相比较，这种游戏明显的一个特点是以数字化的形式存在于手机、计算机、iPad 等数字设备上，是现实中游戏主题的软件实现。因此，游戏产业与计算机软硬件、互联网等的发展速度极为相关。计算机游戏为游戏参与者提供了一个虚拟的空间，从一定程度上让人可以摆脱现实世界，在另一个世界中扮演真实世界中扮演不了的角色。同时，多媒体技术的发展，使游戏给人们带来更多的体验和享受。在后续讨论中，本书还是采用游戏一词，而不是计算机游戏。

由于游戏本质上还是一种计算机软件，因此，游戏的发展几乎是和电子计算机的发展并行的。而且这种软件的设计通常具有个性化、极具想象力等特点。因此，得到了一些大学编程高手的青睐。在 20 世纪 60 年代，电子计算机已经进入美国大学校园，大学生对于这一新生事物表现出了极大的兴趣，出现了一大批程序设计高手。这些编程高手开始考虑如何设计游戏软件。在 1961 年，麻省理工学院学生斯蒂夫·拉塞尔在美国 DEC 公司生产的 PDP-1 型电子计算机上成功开发了《宇宙战争》（Space War）。这是真正运行在计算机上的第一款交互式游戏，他也被尊称为计算机游戏的发明人。而在游戏产业发展史上另外一个具有里程碑意义的事件是《乒乓》游戏。该游戏被放置在一家酒吧里，结果大获成功，玩的人络绎不绝，结果因为投币过多而使主机停止工作。《乒乓》证明了电子游戏可以用来获取经济效益，也证明了电子游戏产业诞生的可能，于是，1972 年 1 月 27 日，布什内尔成立世界上第一家专注于电子游戏生意的公司——雅达利（Atari）。雅达利随后开创了辉煌的街机产业，创造了一个不朽的传奇，其发明者拉尔夫·贝尔（Ralph Baer）也被称为真正的“电子游戏之父”。从 1980 年代开始，普通微机也开始普及，多媒体技术日趋成熟，更是推动了游戏产业的飞速发展。而在硬件方面，图形显示卡技术使 3D 游戏成为可能。在 20 世纪 90 年代以后，网络技术飞速发展，而且网络速度越来越快，玩家只需要通过互联网就可以实现多人游戏，突破了多人游戏的空间限制。特别是进入 21 世纪，网络游戏更是成为游戏的一个新的发展方向。

计算机游戏自 1972 年由威尔·克劳舍（Will Crowther）编写的一段简单的 FORTRAN 程序

开始，已经历了数十年风风雨雨。从最开始避开陷阱的简单地图，到今天即时战略、角色扮演、动作冒险、经营策略、休闲养成等各种类型的游戏；从2D到3D，游戏无论在技术上还是画面上都以惊人的速度不断突破。它的成长速度是如此之快，出乎任何人的意料。现在，北美的电子游戏产业（包括TV Game和PC Game）的收入已经接近了整个电影产业的收入，这就足以证明游戏行业潜力的巨大。

## 1.2 什么是游戏

尽管游戏这个名词对任何人来说都是耳熟能详，但是，如何给出一个精确的定义一直以来是相关领域专家要解决的问题。这些领域包括人类学、哲学、历史学以及游戏设计等。定义任何牵涉人类多种行为的术语都是一件非常棘手的事情，任何人只要能举出一个反例，这个定义就不准确。对于游戏的定义分析，比较权威的可参考*Rules of Play*，在该书中，Salen和Zimmerman列举了这些定义，并对其进行了分析，试图提取出游戏定义中的共性特征。可以发现，大多数定义都涉及游戏规则、目标任务、玩和假想。同时包括了制定决策或者游戏之类。因此，游戏是在一种假设的虚拟环境下，参与者按照规则行动，实现至少一个既定的重要目标任务的娱乐性活动。实际上这种定义有效地区分开了玩具和游戏之间的区别。虽然同是作为玩的一种对象，可是玩具并没有严格的规则定义。

## 1.3 游戏的分类

对于游戏的分类，有不同的分类标准。这些分类标准包括游戏主题、游戏发展的时间段、游戏所采用的主要技术等。其中游戏主题分类最为典型。通过游戏主题分类，可以对当前游戏有一个比较清晰的认识。

### 1.3.1 角色扮演游戏

角色扮演游戏（Role Play Game，RPG）是由玩家扮演游戏中的一个或数个角色，有完整故事情节的游戏。玩家可能会与冒险类游戏混淆，其实区分很简单，RPG游戏更强调的是剧情发展和个人体验，一般来说，RPG可分为日式和美式两种，主要区别在于文化背景和战斗方式。日式RPG多采用回合制或半即时制战斗，如《最终幻想》系列，大多国产中文RPG也可归为日式RPG之列，如《仙剑》、《剑侠》等；美式RPG如《暗黑破坏神》系列。

### 1.3.2 网页游戏

网页游戏又称为无端网游，中文也简称页游，是基于Web浏览器的网络在线多人互动游戏，无须下载客户端，打开网页仅用浏览器即能玩。尤其适合上班一族，只要几分钟的设定，游戏内

的自动成长功能使得即使关闭网页，游戏仍然可以继续，如网页版《捕鱼达人》（图 1-1），以及经典的《石器部落》。

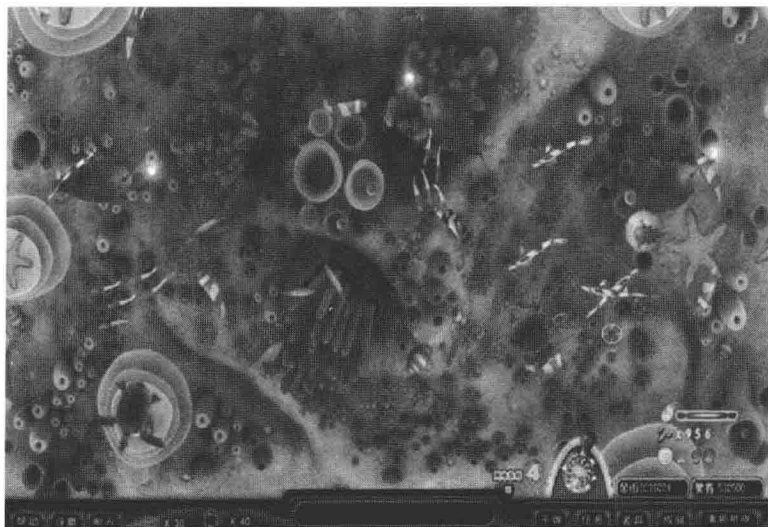


图 1-1 捕鱼达人之深海狩猎

### 1.3.3 动作游戏

动作游戏是玩家控制游戏人物用各种武器消灭敌人以过关的游戏，不追求故事情节，如熟悉的《超级玛丽》、可爱的《星之卡比》、华丽的《波斯王子》（图 1-2）等。计算机上的动



图 1-2 波斯王子

作游戏大多脱胎于早期的街机游戏和动作游戏，如《魂斗罗》、《三国志》等，设计主旨是面向普通玩家，以纯粹的娱乐休闲为目的，一般有少部分简单的解谜成分，操作简单，易于上手，紧张刺激，属于“大众化”游戏。

### 1.3.4 冒险游戏

冒险游戏（Adventure Game, AVG）是由玩家控制游戏人物进行虚拟冒险的游戏。与RPG不同的是，AVG的特色是故事情节往往以完成一个任务或解开某些谜题的形式出现的，而且在游戏过程中刻意强调谜题的重要性。AVG也可再细分为动作类和解谜类两种，动作类AVG可以包含一些格斗或射击成分，如《生化危机》系列（图1-3）、《古墓丽影》系列、《恐龙危机》等；而解谜类AVG则纯粹依靠解谜拉动剧情的发展，难度系数较大，代表是经典的《神秘岛》系列。



图1-3 生化危机

### 1.3.5 策略模拟类游戏

策略模拟类游戏是玩家运用策略与计算机或其他玩家较量，以取得各种形式胜利的游戏，或统一全国，或开拓外星殖民地。策略游戏可分为回合制和即时制两种，回合制策略游戏如《三国志》系列、《樱花大战》系列；即时制策略游戏如《命令与征服》系列、《帝国》系列、《沙丘》等。后来有些媒体将模拟经营游戏，即SIM（Simulation）类游戏，如《模拟人生》、《模拟城市》、《过山车大亨》、《主题公园》（图1-4）等也归作策略模拟类游戏。



图 1-4 主题公园

## 1.4 游戏设计的相关知识

游戏设计是一个技术和艺术相结合的产物，具有明显的跨学科性。其中最为重要的无疑是计算机图形学、艺术设计、人工智能和人机交互。本节将对这些游戏相关的知识进行介绍。

### 1.4.1 计算机图形学

由于游戏是一种计算机中的虚拟运行场景，而计算机图形学正是研究如何将这种虚拟场景在计算机中有效地展现，从而模拟真实世界场景。计算机图形学主要解决游戏中的两个关键问题：第一是场景渲染；第二是动画生成。对于场景的渲染，包括对物体形状、光学性质、表面纹理和粗糙程度，以及物体间的相对位置、遮挡关系等的模拟。其中，光照和表面属性的模拟依旧是当前研究的难点。近年来大规模场景游戏越来越流行，这些三维场景数据量特别大，又要求实时绘制，因此，绘制的实时性也成了另外一个研究重点，需要在最短时间内绘制出最真实的场景，提高游戏的流畅度。对于实时绘制，多分辨率方法和基于图像的绘制（Image Based Rendering, IBR）是最为典型的方法。多分辨率方法是在一定误差范围内，删除部分点、边、面，从而简化所绘制场景的复杂程度，加快图形绘制速度。而 IBR 则是首先确定光源的绘制方法，然后从一系列已知的图像中生成未知视角的图像。这种方法由于不需要建立场景的几何模型和光照模型，也不需要进行如光线跟踪等极费时的计算，尤其适用于野外极其复杂场景的生成和漫游。另外，计算机游戏中的场景模拟已经从最初绘制简单的 2D 室内场景发展到现在 3D 模拟野外自然景物，如绘制山、水、云、树、火等。人们提出了多种方法来绘制自然



景物，如绘制火和草的粒子系统（Particle System），基于机理模型的绘制植物的方法，绘制云的细胞自动机方法等。

计算机游戏中图形显示的另一个重点是计算机动画的实现。事实上，计算机动画也只是生成一幅静态的图像，但是每一幅图像都是对前一幅图像做一小部分的修改（如何修改便是计算机动画的研究内容），这样，当这些画面连续播放时，整个场景就动起来了。早期的计算机动画灵感来源于传统的卡通片，在生成几幅被称作“关键帧”的画面后，由计算机对两幅关键帧进行插值生成若干“中间帧”，连续播放时两个关键帧就被有机地结合起来了。计算机动画内容丰富多样，生成动画的方法也多种多样，如基于特征的图像变形、二维形状混合、轴变形方法、三维自由形体变形（Free-Form Deformation, FFD）等。近年来，计算机游戏普遍将注意力转向基于物理模型的计算机动画生成方法，骨骼动画和基于物理系统的刚体动画都是向前发展的重点。这是一种崭新的方法，该方法大量运用刚体力学和流体力学的方程进行计算，力求使动画过程体现出最适合真实世界的运动规律。然而要真正达到真实运动是很难的，如人的行走或跑步是全身的各个关节协调的结果，要实现很自然的人走路动画计算方程非常复杂，计算量极大，基于物理模型的计算机动画的许多内容仍在不断发展中。

### 1.4.2 艺术设计

对于游戏，主题是吸引玩家的一个主要因素。但是，对于一个好的游戏，需要有一个非常艺术化的表示，从而能够吸引主要玩家。因此，艺术设计对于一个好的游戏起着非常重要的作用。对于游戏中的艺术设计，概念设计、草图设计和视觉风格设计将贯穿整个游戏开发过程。在游戏开发的前期设计阶段，需要绘制各种草图并创作情节串联图板，以便展示和充实设计师的想法。然后建模人员以该草图作为参考，通过 Maya、Photoshop 等软件实现数字化造型和设计，并根据设计文档图，由美工人员绘制游戏角色概念设计草图。开发人员在构造模型和编程的时候将以这份草图作为参考，然后依照设计文档的要求制作各种模型和纹理，包括角色、建筑物、交通工具和各种图标。视觉风格设计则是确立整个游戏的统一视觉风格（Visual Style），也就是游戏外观，包括色彩配置、文化背景、特殊图案、角色设计风格、建筑设计风格、界面设计风格等。通常，最后要编写一部风格指导手册（Style Guide）。风格指导手册的作用是使所有美工组成员进行工作时有所依据。当他们进行具体工作时可以对照风格指导手册，根据其要求调整自己的作品，从而保证整个游戏画面风格一致。

### 1.4.3 人工智能

近年来，人工智能在游戏中的作用日益增长。原因主要有5个：一是为玩家提供合适的挑战；二是使玩家处于亢奋状态；三是提供不可预知性结果；四是帮助完成游戏的故事情节；五是创造一个生动世界。这个生动世界可以是类似现实生活中的世界，也可以是与现实世界完全不同的世界。但不管何种世界都要求有一整套能够自圆其说的游戏规则。

在游戏制作过程中，实现人工智能的关键有虚拟现实与拟人化、动画效果与机器角色场景