

<<Android 4游戏编程入门经典>>

图书基本信息

书名：<<Android 4游戏编程入门经典>>

13位ISBN编号：9787302301059

10位ISBN编号：7302301050

出版时间：2012-11

出版时间：清华大学出版社

作者：(美) 策希纳(Zechner, M.),无(美) 格林(Green, R.) 著

页数：534

字数：876000

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Android 4游戏编程入门经典>>

内容概要

《android

4游戏编程入门经典》囊括了成为一名成功的android游戏开发者所需的全部知识。本书是一本可以信赖的指南，能够帮助你在移动游戏开发中迅速“升级”。学完本书并加以融会贯通，就能够编写出成功的android游戏。

《android

4游戏编程入门经典》将引导你在android平台上完整地开发出3个示例游戏，包括2d游戏和3d游戏。本书首先介绍游戏开发的理论基础，而且通过只使用剪刀、笔和纸从头设计一个完整的游戏，教会你游戏设计的基础知识。接下来，你将学习编程基础知识，并开始创建基本的游戏引擎和可玩的游戏。你将以一个有趣的2d游戏作为起点，最终创建一个完善的3d太空入侵者游戏的克隆版本。这些项目将一步步地丰富你的知识和技能。

主要内容

设置和使用android开发工具：准备好编写自己的游戏
经典的2d游戏编程：创建让人爱不释手的动作游戏和平台游戏
android图形和音频：让游戏看起来和听起来都如此美妙
游戏机制：充分利用碰撞检测、物理知识和精灵动画
3d游戏编程：以简单的方式在游戏中添加复杂的3d
最后一步：发布游戏，收到崩溃报告，帮助用户解决问题

《android

4游戏编程入门经典》详尽地介绍了移动游戏所需的一切基础知识，书中讲解的知识和技能将帮助你成为一名出色的移动游戏开发者。

<<Android 4游戏编程入门经典>>

作者简介

Mario Zechner白天是一名软件研发工程师，到了晚上就变身为热情高涨的游戏开发者，以Badlogic Games的名义发布游戏。

他开发了基于Android平台的游戏Newton，和基于Windows、Linux和Mac OSX的游戏Quantum以及其他众多游戏原型和小型游戏。

目前，他正致力于一个名为libgdx的开源跨平台游戏开发解决方案。

除了编写代码以外，他还积极撰写关于游戏开发的教程和文章，这些可以在网上特别是他的博客(<http://badlogicgames.com>)上免费查看。

<<Android 4游戏编程入门经典>>

书籍目录

第1章 android，后起之秀

- 1.1 android简介
- 1.2 版本分裂
- 1.3 谷歌的角色
 - 1.3.1 android开源项目
 - 1.3.2 android market
 - 1.3.3 挑战赛、设备播种计划和谷歌i/o
- 1.4 android的功能和体系结构
 - 1.4.1 内核
 - 1.4.2 运行库和dalvik虚拟机
 - 1.4.3 系统库
 - 1.4.4 应用程序框架
- 1.5 软件开发工具包
- 1.6 开发人员社区
- 1.7 设备，设备，设备
 - 1.7.1 硬件
 - 1.7.2 设备的范围
- 1.8 所有设备之间的兼容性
- 1.9 不同的手机游戏
 - 1.9.1 人手一台游戏机
 - 1.9.2 随时上网
 - 1.9.3 普通用户与游戏迷
 - 1.9.4 市场很大，开发人员很少
- 1.10 小结

第2章 从android sdk开始

- 2.1 搭建开发环境
 - 2.1.1 安装jdk
 - 2.1.2 安装android sdk
 - 2.1.3 安装eclipse
 - 2.1.4 安装adt eclipse插件
 - 2.1.5 eclipse快速浏览
 - 2.1.6 一些实用的eclipse快捷键
- 2.2 android环境下的hello world
 - 2.2.1 创建项目
 - 2.2.2 进一步分析项目
 - 2.2.3 编写应用程序代码
- 2.3 运行和调试android应用程序
 - 2.3.1 连接设备
 - 2.3.2 创建一个android虚拟设备
 - 2.3.3 运行应用程序
 - 2.3.4 调试应用程序
 - 2.3.5 logcat和ddms
 - 2.3.6 使用adb
- 2.4 小结

第3章 游戏开发基础

<<Android 4游戏编程入门经典>>

3.1 游戏类型

3.1.1 休闲游戏

3.1.2 益智游戏

3.1.3 动作和街机游戏

3.1.4 塔防游戏

3.1.5 创新

3.2 游戏设计：笔比代码更强大

3.2.1 游戏的核心机制

3.2.2 一个故事和一种艺术风格

3.2.3 画面和切换

3.3 代码：具体细节

3.3.1 应用程序和窗口管理

3.3.2 输入

3.3.3 文件i/o

3.3.4 音频

3.3.5 图形

3.3.6 游戏框架

3.4 小结

第4章 面向游戏开发人员的android

4.1 定义一个android应用程序：清单文件

4.1.1 [manifest]元素

4.1.2 [application]元素

4.1.3 [activity]元素

4.1.4 [uses-permission]元素

4.1.5 [uses-feature]元素

4.1.6 [uses-sdk]元素

4.1.7 10个简单步骤建立android游戏项目

4.1.8 市场过滤器

4.1.9 定义游戏图标

4.2 android api基础

4.2.1 创建测试项目

4.2.2 活动的生命周期

4.2.3 处理输入设备

4.2.4 文件处理

4.2.5 音频编程

4.2.6 播放音效

4.2.7 音乐流

4.2.8 基本图形编程

4.3 最佳实践

4.4 小结

第5章 android游戏开发框架

5.1 制定计划

5.2 androidfileio类

5.3 androidaudio、androidsound和androidmusic

5.4 androidinput和accelerometer-handler

5.4.1 accelerometerhandler：手机哪一面朝上

5.4.2 compasshandler

<<Android 4游戏编程入门经典>>

- 5.4.3 pool类：重用相当有用
- 5.4.4 keyboardhandler
- 5.4.5 触摸处理程序
- 5.4.6 androidinput：优秀的协调者
- 5.5 androidgraphics和androidpixmap
- 5.5.1 处理不同屏幕大小和分辨率的问题
- 5.5.2 androidpixmap：人物的像素
- 5.5.3 androidgraphics：满足绘图需求
- 5.5.4 androidfastrenderview
- 5.6 androidgame：合并所有内容
- 5.7 小结

第6章 mr. nom入侵android

- 6.1 创建资源
- 6.2 建立项目
- 6.3 mrnomgame：主要活动
 - 6.3.1 资源：便捷的资源存储
 - 6.3.2 设置：跟踪用户的选项设置和高分榜
 - 6.3.3 loadingscreen：从磁盘获取资源
- 6.4 主菜单画面
- 6.5 helpscreen类
- 6.6 高分榜画面显示
 - 6.6.1 渲染数字
 - 6.6.2 画面的实现
- 6.7 抽象
 - 6.7.1 抽象mr. nom的世界：模型、视图、控制器
 - 6.7.2 gamescreen类
- 6.8 小结

第7章 opengl es介绍

- 7.1 opengl es概述以及关注它的原因
 - 7.1.1 编程模型：一个比喻
 - 7.1.2 投影
 - 7.1.3 规范化设备空间和视口
 - 7.1.4 矩阵
 - 7.1.5 渲染管道
- 7.2 开始之前
- 7.3 glsurfaceview：从2008年开始，事情变得简单了
- 7.4 glgame：实现游戏接口
- 7.5 绘制一个红色的三角形
 - 7.5.1 定义视口
 - 7.5.2 定义投影矩阵
 - 7.5.3 指定三角形
 - 7.5.4 综合示例
- 7.6 指定每个顶点的颜色
- 7.7 纹理映射：轻松地创建壁纸
 - 7.7.1 纹理坐标
 - 7.7.2 上传位图
 - 7.7.3 纹理过滤

<<Android 4游戏编程入门经典>>

- 7.7.4 释放纹理
- 7.7.5 有用的代码片段
- 7.7.6 启用纹理
- 7.7.7 综合示例
- 7.7.8 texture类
- 7.8 索引顶点：重用是有好处的
- 7.8.1 代码整合
- 7.8.2 vertices类
- 7.9 半透明混合处理
- 7.10 更多图元：点、线、条和扇
- 7.11 2d变换：操作模型视图矩阵
- 7.11.1 世界空间和模型空间
- 7.11.2 再次讨论矩阵
- 7.11.3 第一个使用平移的示例
- 7.11.4 更多的变换
- 7.12 性能优化
- 7.12.1 测量帧率
- 7.12.2 android 1.5平台下hero的奇特案例
- 7.12.3 使opengl es渲染如此慢的原因
- 7.12.4 移除不必要的状态改变
- 7.12.5 减小纹理大小意味着需要获取更少的像素
- 7.12.6 减少opengl es/jni方法的调用
- 7.12.7 绑定顶点的概念
- 7.12.8 写在结束之前
- 7.13 小结

第8章 2d游戏编程技巧

- 8.1 写在开始
- 8.2 向量
- 8.2.1 使用向量
- 8.2.2 一点三角学的知识
- 8.2.3 实现一个向量类
- 8.2.4 一个简单的用法示例
- 8.3 2d物理定律浅析
- 8.3.1 牛顿和欧拉，永远的好朋友
- 8.3.2 力和质量
- 8.3.3 理论上的运动
- 8.3.4 运动的实现
- 8.4 2d碰撞检测和对象表示
- 8.4.1 边界形状
- 8.4.2 构造边界形状
- 8.4.3 游戏对象的属性
- 8.4.4 宽阶段和窄阶段碰撞检测
- 8.4.5 一个详细的示例
- 8.5 2d照相机
- 8.5.1 camera2d类
- 8.5.2 示例
- 8.6 纹理图集

<<Android 4游戏编程入门经典>>

8.7 纹理区域、精灵和批处理：隐藏opengl es

8.7.1 textureregion类

8.7.2 spritebatcher类

8.8 精灵动画

8.8.1 animation类

8.8.2 示例

8.9 小结

第9章 super jumper：一个2dopengl es游戏

9.1 核心游戏机制

9.2 背景故事和艺术风格

9.3 画面和切换

9.4 定义游戏世界

9.5 创建资源

9.5.1 ui元素

9.5.2 使用点阵字体处理文本

9.5.3 游戏元素

9.5.4 用于救援的纹理图集

9.5.5 音乐与音效

9.6 实现super jumper

9.6.1 assets类

9.6.2 settings类

9.6.3 主活动

9.6.4 font类

9.6.5 glscreen

9.6.6 主菜单画面

9.6.7 帮助画面

9.6.8 高分画面

9.6.9 模拟类

9.6.10 游戏画面

9.6.11 worldrenderer类

9.7 是否需要优化

9.8 小结

第10章 opengl es：进入3d世界

10.1 准备工作

10.2 3d中的顶点

10.2.1 vertices3：存储3d空间位置

10.2.2 示例

10.3 透视投影：越近则越大

10.4 z-buffer：化混乱为有序

10.4.1 完善上一个例子

10.4.2 混合：身后空无一物

10.4.3 z-buffer精度与z-fighting

10.5 定义3d网格

10.5.1 立方体：3d中的“helloworld”

10.5.2 一个示例

10.6 矩阵和变换

10.6.1 矩阵堆栈

<<Android 4游戏编程入门经典>>

10.6.2 用矩阵堆栈实现分层系统

10.6.3 木箱太阳系的简单实例

10.7 小结

第11章 3d编程技巧

11.1 准备工作

11.2 3d中的向量

11.3 opengl es中的光照

11.3.1 光照的工作机制

11.3.2 光源

11.3.3 材质

11.3.4 opengl es中如何对光照过程进行运算：顶点法线

11.3.5 实践

11.3.6 关于opengl es中光照应用的一些建议

11.4 材质变换(mipmapping)

11.5 简单的照相机

11.5.1 第一人称照相机或欧拉照相机

11.5.2 一个欧拉照相机的示例

11.5.3 跟随照相机

11.6 加载模块

11.6.1 wavefront obj格式

11.6.2 obj加载器的实现

11.6.3 使用obj加载器

11.6.4 关于加载模型的一些建议

11.7 3d中的一些物理知识

11.8 碰撞检测与3d中的对象表达法

11.8.1 3d中的边界形状

11.8.2 边界球重叠测试

11.8.3 gameobject3d与dynamic-gameobject3d

11.9 小结

第12章 droid invaders游戏

12.1 游戏的核心机制

12.2 游戏的故事背景与艺术风格

12.3 屏幕与场景切换

12.4 定义游戏世界

12.5 创建资源

12.5.1 用户界面的资源

12.5.2 游戏资源

12.5.3 音效与音乐

12.6 开始编写代码

12.7 assets类

12.8 settings类

12.9 主活动

12.10 主菜单

12.11 游戏设置画面

12.12 模拟类

12.12.1 shield类

12.12.2 shot类

<<Android 4游戏编程入门经典>>

- 12.12.3 ship类
- 12.12.4 invader类
- 12.12.5 world类
- 12.13 gamescreen类
- 12.14 worldrender类
- 12.15 游戏优化
- 12.16 小结

第13章 发布游戏

- 13.1 关于测试
- 13.2 成为注册开发人员
- 13.3 给游戏的apk包签名
- 13.4 将游戏发布至market
 - 13.4.1 上传资源
 - 13.4.2 产品详情
 - 13.4.3 发布选项
 - 13.4.4 发布
 - 13.4.5 市场推广
- 13.5 开发人员控制台
- 13.6 小结

第14章 进阶内容

- 14.1 社交网络
- 14.2 位置识别
- 14.3 多玩家功能
- 14.4 opengl es 2.0以及更多内容
- 14.5 框架及引擎
- 14.6 网络资源
- 14.7 结束语

章节摘录

版权页：插图：内容提供程序（content provider）：该组件允许将应用程序的部分数据用于其他应用程序。

意图（intent）：这是由系统或应用程序本身所创建的消息，它可传递到任何对其感兴趣的一方。

意图可以告诉我们一些系统事件，例如，SD卡正在被移除或USB电缆正在进行连接等。

意图也可被系统用于启动一些应用程序的组件，如活动。

我们也可激活自己的意图来要求其他应用程序执行一个操作，例如打开照片库来显示一幅图像或启动Camera应用程序来拍一张照片。

广播接收者（broadcast receiver）：用于响应特定的意图，并可执行一个动作，例如启动一个特定的活动或向系统发送另一个意图。

Android应用程序没有单一的程序入口点，这与我们熟悉的桌面操作系统不同（例如，Java中的main（）方法就是程序的入口点）。

相反，启动Android应用程序组件或命令其执行一个指定的动作都是通过特定的意图实现的。

应用程序中所包含的组件和这些组件所响应的意图全都定义在应用程序的清单文件中。

Android系统通过使用该文件来获知应用程序的构成，如应用程序启动时显示的默认活动。

注意：本书中只关注活动，所以只讨论清单文件中该类型组件的相关清单文件部分。

如果你想了解其他组件，那么可以到Android开发人员网站中学习有关清单文件的更多知识。

清单文件除了定义一个应用程序的组件外，还有很多的用途。

下面的列表总结了在游戏开发背景下清单文件的相关部分的用途。

应用程序的版本号，用于在Android Market上显示和使用。

应用程序所运行的Android系统版本。

应用程序要求的硬件配置文件（如多点触摸、特定屏幕分辨率和对OpenGL ES 2.0的支持等）。

使用特定组件的权限，如SD卡的写权限或联网栈的访问权限。

我们将在下面的小节中创建一个模板清单，并通过稍微的修改而将其重用于本书开发的所有项目中。

为此，我们可以看看用于定义应用程序所需的所有相关XML标记。

<<Android 4游戏编程入门经典>>

编辑推荐

《Android 4游戏编程入门经典》囊括了成为一名成功的Android游戏开发者所需的全部知识。
《Android 4游戏编程入门经典》是一本可以信赖的指南，能够帮助你在移动游戏开发中迅速“升级”

。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>