

三维动漫游戏人物 角色仿真技术

姚俊峰/著



科学出版社

www.sciencep.com



三维动漫游戏人物 角色仿真技术

姚俊峰 著

科学出版社

北 京

内 容 简 介

本书针对目前动漫游戏开发过程中具有共性的关键开发技术,将项目组积累五年多的研究成果整理出来,分三个部分进行阐述。三维动漫游戏人物人头及表情仿真技术:三维人脸表情动画技术,基于混合模型的人脸表情模拟,三维发型设计研究。三维动漫游戏人物人体及运动仿真技术:基于特征的三维人体曲面模型重构,三维虚拟人运动人体建模实现,虚拟人运动骨骼蒙皮实现,虚拟人运动路径规划研究,虚拟人运动模拟实现。三维动漫游戏人物服装仿真技术:三维衣片网格划分研究,衣片放置及服装纹理的研究,布料动态模拟实现,衣片与人体碰撞检测及响应研究,三维虚拟服装试衣系统。

本书可作为高校动漫游戏及相关专业的本科生、研究生的教材,也适用于动漫游戏公司的中、高级开发人员阅读参考。

图书在版编目(CIP)数据

三维动漫游戏人物角色仿真技术/姚俊峰著. —北京:科学出版社,2009
ISBN 978-7-03-024803-9

I. 三… II. 姚… III. 三维-动画-计算机仿真 IV. TP391.9

中国版本图书馆 CIP 数据核字(2009)第 099911 号

责任编辑:王志欣 魏英杰 / 责任校对:宋玲玲

责任印制:赵 博 / 封面设计:耕 者

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码:100717

<http://www.sciencep.com>

新 蕾 印 刷 厂 印 刷

科学出版社发行 各地新华书店经销

*

2009 年 7 月第 一 版 开本: B5(720×1000)

2009 年 7 月第一次印刷 印张: 25 1/2

印数: 1—3 500 字数: 501 000

定价: 58.00 元

(如有印装质量问题,我社负责调换〈新蕾〉)

前 言

现代动漫游戏产业是借助于 IT(软件、网络)等高技术手段,设计、生产、销售和传播具有文化、艺术、娱乐价值的动画、漫画、游戏及其衍生产品的新兴产业。

在目前的国际动漫产业中,欧、美、日、韩呈争先态势。英国数字娱乐产业年产值占 GDP 的 7.9%,已成为该国第一大产业。美国的网络游戏业已经连续 4 年超过好莱坞电影业,成为全美最大的娱乐产业。日本动画业年产值在国民经济中占第六位,动画产品出口值远远高于钢铁。2007 年,全球数字内容产业产值达 2800 亿美元,与动漫游戏、动画业相关的周边衍生产品产值则在 6000 亿美元以上。

近年来,我国数字娱乐产业发展迅猛,2003 年全国动漫游戏产业总收益已超过电影业。北京、上海、广东、浙江、湖南等地的动漫游戏产业发展相对较快,同时,国家级的动画产业基地也相继建立。但对比国际动漫游戏产业的发展水平,我国动漫游戏产业只是刚刚起步。

本书针对目前动漫游戏开发过程中的共性关键技术,将项目组积累五年多的研究成果整理出来,希望能够起到抛砖引玉的作用。

书中论述重点共分为如下三个部分:

三维动漫游戏人物人头及表情仿真技术。人物人头技术包括计算机图形学,基于 OpenGL 的 MFC 编程技术,人体解剖学,立体视觉理论。表情模拟技术包括建立完整的人头模型,特定人头模型的生成,表情分析模块,表情生成模块,几何模型的光滑与光顺处理模块,光照效果处理与纹理映射模块,人脸表情动画模块。

三维动漫游戏人物人体及运动仿真技术。三维人体合成技术利用动态参数化人体建模,建立适用于人体运动模型系统和虚拟服装试衣系统的人体三维模型。运动仿真技术根据人体运动控制的实际操作,分为人体骨骼建模,人体运动控制,运动路径规划以及骨骼蒙皮。

三维动漫游戏人物服装仿真技术。具体包括人体模型导入,衣片模型,衣片缝合过程以及碰撞检测与响应原理。

有关计算机软件与项目应用问题,读者如果需要,可与作者联系。项目开发过程中得到厦门博鼎智文公司——厦门大学软件学院全息文化传媒创意产业联合研发中心王炳兴、陈琪楠和孙锦彬,福兴公司——厦门大学软件学院创意产业联合研发中心景冰锋、洪春节和吉绪发,福建省芙莱茵信息技术有限公司的大力

支持,在此深表感谢。

三维动漫游戏人物角色仿真技术内容丰富,发展迅速,是一个充满挑战的新领域。由于作者水平所限,书中不妥或错误之处在所难免,欢迎读者批评指正。

作 者

2009年6月

目 录

前 言

第一章 三维人脸表情动画技术	1
1.1 绪论	1
1.2 系统纵览及相关理论概述	1
1.3 数据模块	6
1.4 构造模块	11
1.5 控制模块	15
1.6 显示模块	19
1.7 结论	24
参考文献	25
第二章 基于混合模型的人脸表情模拟	26
2.1 绪论	26
2.2 人脸表情动画技术回顾	29
2.3 人脸的解剖学基础	37
2.4 三维人脸模型的建立	43
2.5 人脸纹理图的合成与模型的纹理映射	51
2.6 基于混合模型的人脸网格变形研究	59
2.7 人脸表情模拟系统的功能设计与实现	73
2.8 总结与展望	76
参考文献	78
第三章 三维发型设计研究	81
3.1 绪论	81
3.2 三维发型设计系统设计	82
3.3 三维发型设计系统功能	84
3.4 数据结构	89
参考文献	103
第四章 基于特征的三维人体曲面模型重构	105
4.1 绪论	105
4.2 基于人体特征线的躯干曲面成型理论研究	108
4.3 躯干曲面成型的系统设计及实现	127

4.4	系统展示	138
4.5	总结与展望	142
	参考文献	142
第五章	参数化三维人体建模研究	145
5.1	绪论	145
5.2	人体简介	147
5.3	系统设计	154
5.4	建模	165
5.5	结论	171
	参考文献	172
第六章	虚拟人运动骨骼蒙皮实现	173
6.1	绪论	173
6.2	系统的总体设计	175
6.3	系统的详细设计	181
6.4	系统的实现结果	194
6.5	结论	200
	参考文献	200
第七章	虚拟人运动路径规划研究	201
7.1	绪论	201
7.2	人体步行路径规划概述	202
7.3	启发式搜索技术	205
7.4	A* 路径搜索算法	208
7.5	A* 算法的模拟与实现	215
7.6	总结与展望	241
	参考文献	241
第八章	虚拟人步行模拟实现	243
8.1	基于人体步行的研究	243
8.2	Body 模型与 Biped 骨骼的绑定	249
8.3	模拟人体走路运动中各关节关键帧的设置	257
8.4	其他动画	275
8.5	Cal3D 浏览器的导入	281
8.6	结论	282
	参考文献	282
第九章	三维衣片网格划分研究	284
9.1	绪论	284

9.2 衣片网格剖分研究	288
9.3 三角形网格自动剖分的实现	292
9.4 真实感图形处理	311
9.5 结论	314
参考文献	315
第十章 衣片放置及服装纹理的研究	316
10.1 绪论	316
10.2 核心技术研究	317
10.3 衣片初始位置放置研究	324
10.4 服装分类与 CAD 服装制图	328
10.5 服装纹理研究	330
10.6 结论	334
参考文献	335
第十一章 布料动态模拟实现	336
11.1 绪论	336
11.2 布料模拟中的碰撞检测及其响应处理	341
11.3 光照和纹理	352
11.4 系统实现及实验结果	358
11.5 结论	360
参考文献	360
第十二章 衣片与人体碰撞检测及响应研究	361
12.1 绪论	361
12.2 衣片划分研究	361
12.3 建立层次包围盒	366
12.4 碰撞检测	370
12.5 碰撞响应	376
12.6 衣片缝合	378
12.7 总结与展望	380
参考文献	381
第十三章 三维虚拟服装试衣系统	382
13.1 绪论	382
13.2 系统的总体设计	384
13.3 系统的详细设计	387
13.4 系统的实现结果	391
参考文献	398

第一章 三维人脸表情动画技术

1.1 绪 论

三维人脸模型的局部控制与调整是实现三维人脸表情模拟及人脸动画的基础方法。其核心技术可以用于构造三维虚拟人脸;产品可应用于影视制作、网络通信、游戏娱乐、计算机动画、医疗诊断、人机界面和辅助教学等领域。

人脸模型的局部控制与调整是通过预先定义的局部规则集合(单一的局部规则控制特定的部位,规则集合控制整体脸部),使人脸模型从一种状态变换到另一种状态,从而实现脸部动作和表情变化的处理过程。常用的脸部模型控制与调整方法可分为三类^[1,2]:

① 插值变形法。插值技术为脸部模型调整提供了一种直观的方法。几何插值直接修改人脸网格点的位置。参数插值间接移动顶点的位置。

② 参数化方法。参数化技术通过人的先验知识选取一组独立的参数值来构造人脸和表情,能够对特殊的脸部模型实现精确控制。经少量计算,这些参数的组合能够提供较大范围的脸部表情。

③ 人脸生理模型方法。根据人脸的生理结构,定义脸部皮肤、肌肉和骨骼模型,通过肌肉收缩产生网格变形,从而模拟出人脸的物理形变^[3,4]。

从实践的角度分析系统的设计与实现,并采用自顶向下,逐步精确的方法来说明运用到的相关技术和方法。核心方法包括 Waters 的肌肉模型方法及特殊点标定的处理方法等。

本章内容主要包括系统纵览、数据模块、创建模块、显示模块和控制模块五部分,其中控制模块是关键的模块。

1.2 系统纵览及相关理论概述

1.2.1 人脸动作的纵向分类

1. 分类依据和分类结果

人脸表情变化复杂,但是从动作执行方式上,可以简单地把人脸的动作纵向

分为三类:皮肤的伸缩动作、特殊部位的翻转动作(下巴和眼皮)和特殊部位的旋转动作(眼球和头部)。如图 1.1 所示。



图 1.1 三维人脸平滑显示

2. 分类目的

人脸动作的纵向分类,使得对于具有不同特征的动作,可以采用不同的处理方法。这样更容易达到比较理想的模拟效果。

3. 处理方法

人脸皮肤用线性肌肉模型进行处理。眼皮和下巴用标定点方法实现翻转。眼球和头部用矩阵乘法实现旋转^[5]。

1.2.2 人脸动作的层次抽象

1. 抽象依据和抽象结果

人脸表情处于不断变化的过程中,是连续的动作,但是计算机模拟动态变化过程,往往是利用人类视觉反映其具有的特点,在只有微小变化的图像之间快速切换实现动画效果的。所以,可以把人脸的表情动作抽象为两个层次:动态层和静态层。由整体和局部的关系,又可以把静态层划分为两个层次:局部动作层和整体动作层。

2. 局部动作层

局部动作是对人脸某一特殊部位,在某一特定时刻动作的抽象。

人脸的局部动作可以简单分为五个脸部器官的动作:肌肉动作、眼睛动作、眼皮动作、下巴动作和头动作。每个局部动作又包括很多的局部动作。这些动作集

合共同构成局部动作层。

3. 整体动作层

整体动作是对人脸在某一特定时刻表情的抽象,是静态的表情,是人脸局部动作在此时间点动作的集合。

人脸的整体动作,往往传达了特定的意思,例如微笑的表情动作和愤怒的表情动作。所有的表情动作,共同构成整体动作层。

4. 动态层

整体动作(静态表情动作)是静态的,但是人脸是处于不断变化过程中的,是一个连续的过程。动态变化层就是对人脸动态变化的抽象。

5. 抽象的目的

对人脸动作的层次抽象,为人脸表情的层次化控制方法提供了依据,具体运用见后续章节。

1.2.3 系统总体设计思想

1. 面向过程设计

面向过程的设计思想是系统最基本的设计思想。系统用人脸数据结构实现了所有相关数据的隐藏。人脸数据结构是系统的中心,同时有三个大的过程对其进行操作,分别是初始化过程、事件处理过程和绘制过程。如图 1.2 所示。

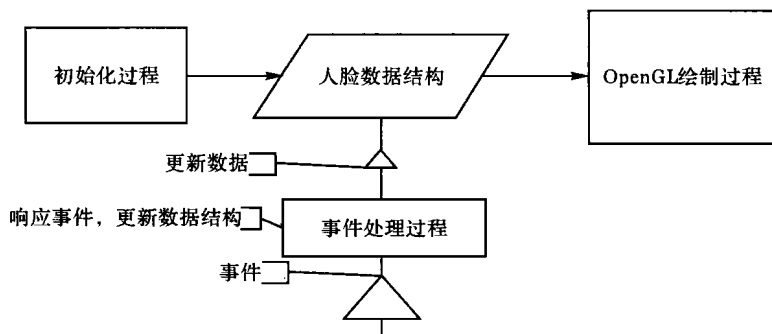


图 1.2 系统主要过程图

初始化过程是一个顺序过程。系统启动只执行一次,任务是初始化并构造人脸数据结构。

绘制过程是一个循环执行的过程。第一次绘制发生在初始化过程完成之后,进入一个大的循环,等待事件发生。当某些事件发生之后,绘制过程被重新调用,实现窗口的重绘^[6]。

事件处理过程是一个选择执行的过程。系统响应来自键盘、鼠标以及菜单等操作的不同事件,选择不同的处理过程完成相应的任务。其完成的主要任务是更改人脸数据结构。

2. 事件驱动设计

事件驱动设计主要利用图 1.2 事件处理过程中所用到的设计思想。它是以外来事件的响应为出发点,选择并处理不同的事件为目标。事件处理过程首先判断来自不同地方的不同事件,然后调用不同的处理过程对人脸数据结构的相应项进行更新。如图 1.3 所示。

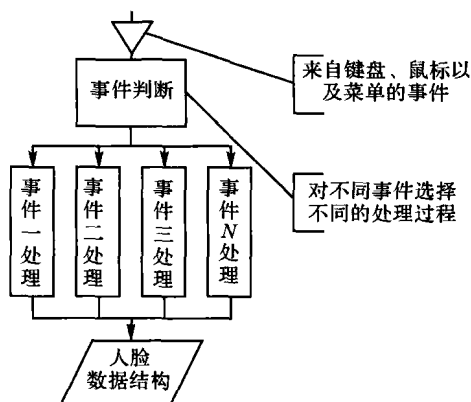


图 1.3 事件判断及处理示意图

3. 数据驱动

数据驱动主要指人脸动画的驱动方式。面向过程以及事件驱动的设计,只能实现人脸的静态表情。为了实现人脸的动画效果,系统利用数据驱动方法和 glut 时间函数的功能,构造了一个简单的动画播放平台。

1.2.4 系统模块划分

系统模块划分如图 1.4 所示。

1. 划分依据和划分结果

以面向过程设计思想为依据,将系统划分为数据模块和过程模块。

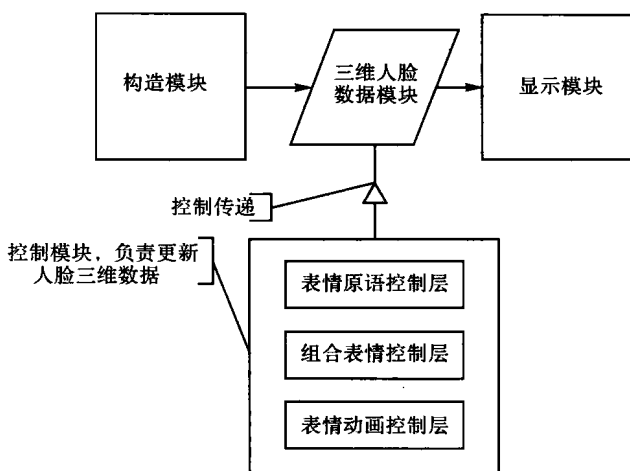


图 1.4 系统模块划分

以功能为依据,将过程模块划分为构造模块、控制模块和显示模块。

以人脸动作的层次抽象为依据,将控制模块又划分为表情原语层、组合表情层和表情动画层。它们分别对应人脸局部动作层、人脸整体动作层和人脸动态层。

2. 数据模块概述

数据模块是系统的核心,其他三个过程模块是以数据模块为中心实现的。

3. 构造模块概述

构造模块主要完成文件读写和人脸数据模型构造的任务。

4. 显示模块概述

显示模块是与 OpenGL 密切相关的一个模块,主要功能是根据不同的参数设置,绘制不同模式的人脸三维图形^[7]。

显示模块主要进行人脸的绘制和辅助结构的绘制。其中,人脸的绘制又包括线框绘制、顶点绘制、三角形绘制、平滑绘制和法向量绘制。辅助结构绘制主要是指肌肉向量的绘制和标定点的绘制。

5. 控制模块概述

控制模块负责响应外部事件,更新三维人脸数据模块。由图 1.4 可知,控制模块又分为表情原语控制层、组合表情控制层和表情动画控制层。

表情原语控制直接对人脸数据模型中的数据进行操作,为上层提供对人脸局部控制的接口。

组合表情控制间接对人脸数据模型中的数据进行整体更新,直接使用表情原语层提供的接口。

表情动画控制用来实现三维人脸动画,直接使用组合表情层提供的接口。

1.3 数据模块

1.3.1 数据模块描述

数据模块的功能是在三维空间中对人脸建模。

1.3.2 数据模块详细分析

1. 标定点

(1) 定义目的

下巴、眼皮是翻转处理部位,要对属于这些部位的三维顶点特殊处理。标定点定义的目的就是标定这些特殊部位的顶点^[8]。

(2) C语言定义结构

```
typedef struct TAG {  
    int  poly ;// poly 项用来标定特殊三维顶点所在的三角形.  
    int  vert ;// vert 项用来标定特殊三维顶点在三角形中的索引.  
} TAG ;
```

2. 三维空间顶点

(1) 定义目的

三维空间顶点是描述三维人脸的最小结构。三维空间顶点的定义为在三维空间中对人脸进行仿真建模提供了最基础的元素。

(2) C语言定义结构

```
typedef struct VERTEX {  
    float  v[3] ;// v 项用来定义三维空间点的三个坐标.  
    float  vv[3] ;// vv 项用来定义三维空间点的三个坐标,与第一
```

```
// 项区别在于此项一旦建立,不会再更改,方便人
// 脸数据的重置.
float norm[3] ;// norm 项用来存储顶点的向量,计算平均向量
// 之前,存储的是这个顶点所在三角形的法向
// 量,之后存储的是平均向量.
int count ;// count 项用来计数与顶点相连的三角形.
int poly[30] ;// poly 项用来存储与顶点相连的三角形索引.
} VERTEX ;
```

3. 组合表情

(1) 定义目的

组合表情定义的目的是描述人脸整体动作,即人脸表情。人脸表情一般由人脸局部动作共同组成,只有特定局部动作的组合,才能传递一定的信息。此结构一般由文件数据读入,不可随意更改。

(2) C 语言定义结构

```
typedef struct COMPOUND {
    char name[100] ;// name 存储表情的名字,如 Happiness,表示高
    // 兴的表情.
    float prim[27] ;// prim 存储 27 个组成表情的表情原语的控制
    // 参数.
} COMPOUND ;
```

4. 线性肌肉模型

(1) 定义目的

脸部皮肤变形所用,此定义来自 Waters 的线性肌肉模型^[9~11]。详细算法分析见控制模块。

(2) C 语言定义结构

```
typedef struct MUSCLE {
    char name[100] ;// name 存储肌肉向量的名称.
    float head[3] ;// head 肌肉向量头的坐标.
    float tail[3] ;// tail 肌肉向量尾的坐标.
```

```

float  zone ;// zone 肌肉向量的影响角度.
float  start ;// start 肌肉向量影响范围的开始位置.
float  end ;// end 肌肉向量的影响范围的结束位置.
float  flex ;// flex 肌肉的拉伸程度.
float  vflex ;// vflex 肌肉向量的拉伸程度,不变量,重置所用.
float  scale ;// scale 肌肉向量的调整度.
BOOL  bit_musc ;// bit_musc 控制肌肉向量的拉伸.
} MUSCLE ;

```

5. 空间三角形面片

(1) 定义目的

定义三维空间中建模人脸所需要的最基本结构。

(2) C 语言定义结构

```

typedef struct POLYGON {
    VERTEX * vert[3] ;// vert 为一个三维指针数组,分别指向构成
                        // 这个三角形的三个顶点的结构体.
} POLYGON ;

```

6. 人脸

(1) 定义目的

描述人脸的三维结构。

(2) C 语言定义结构

```

typedef struct HEAD {
    BOOL  bit_edge ;// 以 bit 开头的项,主要用来控制显示模式. 置
                    // 0 为关闭此显示模式,置 1 为打开此显示
                    // 模式,方便菜单控制人脸显示模式.

    BOOL  bit_poly ;
    BOOL  bit_vert ;
    BOOL  bit_norm ;
    BOOL  bit_patch ;
    BOOL  bit_comic ;

```

```
BOOL bit_musc ;
BOOL bit_tag ;
BOOL bit_face ;
BOOL bit_leye ;
BOOL bit_reye ;
int mode ;// mode 项用来选择系统的运行模式. 因为系统中存在不
// 同层次的控制,如动画层控制和原语层控制,不同层次
// 的控制往往会互相干涉,影响人脸结构,所以系统以三
// 个层次的控制为依据,划分为随机模式、调整模式和动
// 画模式.
int count_index ;// count 开头的项是计数器,主要是用来计数一
// 些结构,如 count_prim 就是用来计数人脸表
// 情原语的.
int count_vert ;
int count_poly ;
int count_leye_poly ;
int count_reye_poly ;
int count_leye_tag ;
int count_reye_tag ;
int count_jar_tag ;
int count_musc ;
int count_comp ;
int count_part ;
int count_prim ;
float x ;// x,y,lex,ley,rex,rex 分别控制头、左眼、右眼的旋转
// 角度.
float y ;
float lex ;
float leye ;
float rex ;
float reye ;
int * index ;// index 用来存储从文件中读入的三维顶点
// 索引信息.
float * vert ;// vert 用来存放从文件中读入的三维顶点
// 坐标信息.
```