

第十章 计算机动画

10.1 计算机动画概述

计算机动画是将图形、图案和画面显示在屏幕上，并按一定的规则或预定的要求在屏幕上移动、变换，从而使计算机显示出动态变化的图形的技术。计算机动画所生成的是一个虚拟的世界，画面中的物体并不需要真正去建造，物体、虚拟摄像机的运动也不会受到什么限制，动画师可以随心

所欲地创造他的虚幻世界。目前，计算机动画已形成一个巨大的产业，随着计算机硬件性能价格比的快速提高，它综合利用计算机科学、艺术、数学、物理学和其它相关学科的知识在计算机上生成绚丽多彩的连续的虚拟现实画面，给人们提供了一个充分展示个人想象力和艺术才能的新天地。与传统媒体中的动态图形载体电影和电视相比，目前众多动画制作软件能使多媒体动画为更多的制作者所掌握，制作者能真正使自己的想法变成看得见的动画。

10.2 计算机动画的应用领域

计算机动画不仅可应用于电影特技、商业广告、电视片头、动画片制作、产品模拟试验、电子游戏，还可应用于计算机辅助教育、军事作战演习、训练模拟，甚至于法院的审理等领域。

10.3 计算机动画的分类和原理

计算机动画是计算机图形学和艺术相结合的产物，它是伴随着计算机硬件和图形算法高速发展起来的一门高新技术。动画是运动中的艺术，运动是动画的要素。计算机动画以其制作方法和表现特征通常可以分为二维动画和三维动画两种形式。

1. 二维动画

传统的卡通动画的实现是连续播放多帧画面，每幅画面表述的是运动物体的若干个瞬间，利用观看者的瞬间视觉残留而得到运动的视觉感受。二

维动画显示的主要是平面图形，制作时就象在纸上作画，通过对象的移动、变形、变色等手法表现其运动的效果；计算机动画原理也是一样，计算机动画的每一帧画面都是一幅数字化的图像。

二维动画不仅具有传统动画功能，还兼有计算机特有功能，例如，计算机生成的图像可拷贝、粘贴、翻转、放大、缩小，任意移位以及自动计算机背景移动等，具有检查方便，保证质量、简化管理、生产效率高、能够有效缩短制作周期。缺点为：在二维动画中，计算机只能起到辅助作用，

并不能取代画家用手工绘制的动画关键帧，并且画面在纸张、照片或计算机屏幕显示，无论画面的立体感多强，终究是二维空间上模拟真实三维空间效果。二维动画中计算机的作用为输入和编辑关键帧，计算和生成中间帧，定义和显示运动路径，交互给画面上色，产生特技效果，实现画面与声音同步，控制运动系列的记录等。

2. 三维动画

三维动画则显示立体图形，其制作就象是在摄影棚中拍电影：首先在三维视图中布置被摄对象的位置、规定其运动、安排好各种灯光，然后在特定位置架设好“摄影机”、也可设定摄影机的推拉摇移，最后计算机计算出在这一立体空间下“摄影机所见的”动态图像效果。尽管在常见的二维动画中也可以模拟三维的立体空间，但其图像的精确度等远不及三维动画。

计算机三维动画数据是在计算机内部生成的，而不是简单的外部输入。制作三维动画首先要创建物体模型，然后让这些物体在空间动起来，

如移动、旋转、变形、变色。再通过打灯光等生成栩栩如生的画面。

三维动画：画中的景物有正面、侧面和反面，调整三维空间的视点，能够看到不同的内容。

二维动画制作方法相对简单，表现的内容也较为简练，能很好地表现动态示意图之类的简单图形，二维动画的模拟三维立体效果在视觉上基本能满足立体的要求；三维动画能较为完美地表现三维立体效果，在对空间感要求较高的动画中，三维动画有其独到的魅力。

按计算机动画实现方法，计算机动画可分为：实时（**Real-Time**）动画、和帧（**Frame**）动画两种，电子游戏机的运动画面属于实时动画；根据运动控制方式将计算机动画分为关键帧动画和算法动画。

一、 关键帧动画

帧动画的基本原理类似幻灯片的制作与播放过程，即把整个动画过程划分为一个个片断，将每一片段作为一幅画像在屏幕上一定区域显示，然后把屏幕上的图像存储在文件中，在动态显示时再按顺序不断读取与播放这

些画面，产生动画效果。熟练的动画师设计卡通片中的关键画面，也即所谓的关键帧，然后由一般的动画师设计中间帧。在三维计算机动画中，中间帧的生成由计算机来完成，插值代替了设计中间帧的动画师。所有影响画面图象的参数都可成为关键帧的参数，如位置、旋转角、纹理的参数等。关键帧技术是计算机动画中最基本并且运用最广泛的方法。从原理上讲，关键帧插值问题可归结为参数插值问题，传统的插值方法都可应用到关键帧方法中。但关键帧插值又与纯数学的插值不同。一个好的关键帧插值方

法必须能够产生逼真的运动效果并能给用户提供方便有效的控制手段。一个特定的运动从空间轨迹来看可能是正确的，但从运动学或动画设计来看可能是错误的或者不合适的。用户必须能够控制运动的运动学特性，即通过调整插值函数来改变运动的速度和加速度。

二、实时动画

实时动画也称为算法动画，它是采用各种算法来实现运动物体的运动控制。在实时动画中，计算机对输入的数据进行快速处理，并在人眼察觉不

到的时间内将结果随时显示出来。实时动画的响应时间与许多因素有关，如计算机的运算速度是慢或快，图形的计算是使用软件或硬件，所描述的景物是复杂或简单，动画图像的尺寸是小或大等等。在实时动画中，一种最简单的运动形式是对象的移动，它是指屏幕上一个局部图像或对象在二维平面上沿着某一固定轨迹作步进运动。运动的对象或物体本身在运动时的大小、形状、色彩等效果是不变的。对象的移动因为相对简单且容易实现，又无需生成动画文件，所以在多媒体应用中经常采用。如果在文字、

图形图像、声音的基础上增加对象的移动，比如跳出文字等，以达到简单动画功能，则能大大丰富视觉效果。但是，对于中间没有停顿的复杂动画效果最好使用二维帧动画预先将数据处理和保存好，然后通过播放软件进行动画播放。这是因为微机，特别是低档微机的处理速度有限，实时处理和显示可能会使处理跟不上显示要求而有损于动画显示效果，甚至影响其它媒体数据如声音的播放。

算法动画是采用算法实现对物体的运动控制或模拟摄像机的运动控制。一般适用于三维情景。算法动画根据不同分为以下几种：

- （1）运动学算法：由运动学方程确定物体的运动轨迹；
- （2）动力学算法：从运动的动因出发，又力学方程确定物体的运动形式；
- （3）反向运动学算法：由已知链接物末端的位置和状态，反求运动方程以确定运动形式。

(4) 反向动力学算法: 由已知链接物末端的位置和状态, 反求动力学方程以确定运动形式。

(5) 随机运动算法: 在某些场合下增加运动控制的随机因素的算法。

用算法控制运动的过程包括: 给定环境描述、环境中的物体造型、运动规律、计算机通过算法生成动画帧。模拟摄影机实际是按照观察系统的变化来控制运动, 从运动学的相对性原理来看是等价的, 但也有其独特的控制方式。

为使计算机显示的动画连续、平稳、美观，就像看电影、电视一样，人们使用各种方法生成图形，这些方法基本上可归为以下几类：

（1）异或运算算法（XOR）

异或运算一个重要特性是具有“还原”作用，对屏幕进行一次异或操作显示一幅图像；再一次异或操作清除前一幅图像，使屏幕“还原”，产生动画效果。

本书第四章中介绍的橡皮条技术绘制直线使用了异或运算算法。

（2）块动画

在屏幕中，改变前景运动的物体再画面中的位置，而背景保持不动既可产生动画效果，这种产生动画方式称为块动画。这种方法是将动画物体的图像保存在储存区中，需要是快速从内存中拷贝到屏幕进行重复显示，并通过对该图像像素与背景像素进行异或算法，可使被前景所遮盖的背景图像部分还原。

（3）帧动画（同上面介绍的概念）

（4）图形变换动画

前面我们已经讲过对图形的二维变换、图形的三维变换，利用这些变换对要运动的图形对象进行上述变换形成动画，并且图形变换后图形的失真率很低。

10.4 目前计算机动画面临的问题

目前计算机动画面临以下问题：真实性和实时性、功能更强、速度更快、效果更好、使用更方便、真实运动生成、物体造型，人体动画、

绘制（渲染）。只考虑了动画算法和局部动画的控制，大型动画片的设计制作局部求精过程。

10.5 计算机动画程序设计案例

10.5.1 帧动画

一、程序设计功能说明

根据以上帧动画原理创建的计算机帧动画程序，演示一般植物生长的基本过程，如图 10-1 所示为程序运行时的主界面及其中的一帧。



图 10-1

二、程序设计步骤

1. 创建名称为“帧动画”单文档应用程序框架（创建单文档详细过程请参见第一章）
2. 编辑菜单资源，添加相应消息处理函数添（如表 10.1 所示）。

表 10.1 菜单资源表

菜单标题	标示符 ID	消息	消息处理函数
3. 播放帧动画	ID_PLAY_FRA ME	CONMM AN	OnIDTRANSLATI ON

用

MFC ClassWizar 对话框，【ClassName】和【Object IDs】中均选择 CMyClass，建立 WM_TIMER 消息处理函数 OnTimer(UINT

nIDEvent)

4. 添加帧动画

单击主菜单中的【插入】—>【资源...】，打开【插入资源】对话框，选择【Bitmap】单击【确定】按钮，利用绘图工具，在当前应用程序中，绘制 Bitmap1 如图 10-5-2 所示。以同样方法添加 Bitmap2、Bitmap3、Bitmap4、Bitmap5 帧图片资源。

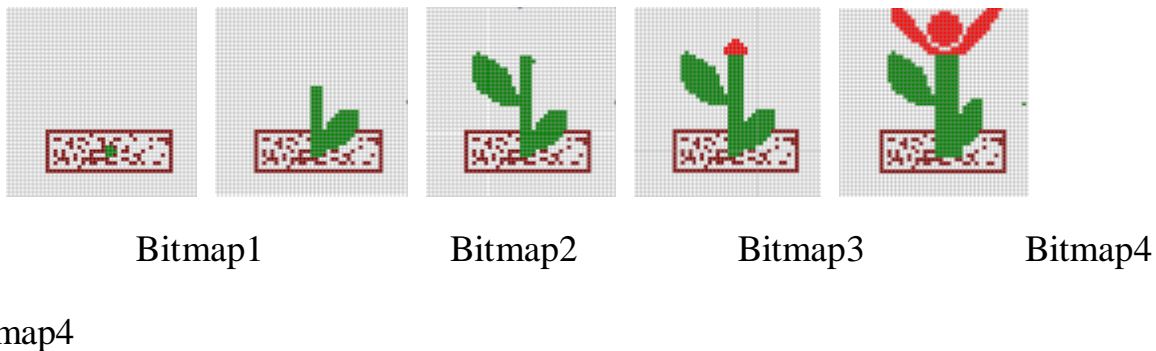


图 10-2

5. 在“帧动画 View.cpp”、“帧动画 View.cpp”、帧动画相应函数
添加如下代码

说明：下面仅列出帧动画 View.h、帧动画 View.cpp 的部分代码，黑体部分为手工输入其它文件代码全部为系统维自动建立，无需改正，为节省篇幅此处省略，详见光盘。

```
// 帧动画 View.h : interface of the CMyView class
```

```
.
```

```
.
```

.
程序代码见纸书

}

10.4.2 实时动画

三、程序设计功能说明

根据以上实时动画原理创建的计算机实时动画程序，演示一弹性小球在

正方体容器中的碰撞过程，如图 10-3 所示为程序运行时的主界面的一个瞬间。

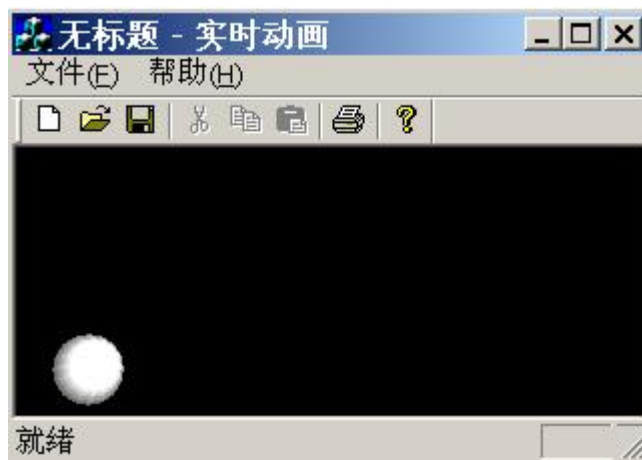


图 10-3

四、程序设计步骤

1. 创建名称为“实时动画”单文档应用程序框架（创建单文档详细过程请参见第一章）
2. 利用 MFC ClassWizar 对话框，【ClassName】和【Object IDs】中均选择 CMyClass，建立 WM_CREATE、WM_DESTROY、WM_TIMER 消息处理函数分别为 OnCreate()、OnDestroy()、

OnTimer()

3. 手工添加绘制球的基类

在工程中单击【文件】|【新建】，在弹出的新建对话框中，选择 C/C++ Header File，在【文件】名称输入栏中输入“Sphere”；同样，在工程中单击【文件】|【新建】，在弹出的新建对话框中，选择 C++ Source File，在【文件】名称输入栏中输入“Sphere”。在工作区中系统自动创建的相应的空文件中，分别添加以下此基类的头文件（.h 文件）和应用文件(.cpp 文件）。

// Sphere.h: interface for the CVirtualSphere class.

#ifndef _CVSPHERE_H

#define _CVSPHERE_H

class CSphere

{

[程序代码见纸书](#)

```
}  
  
}
```

4. 在“实时动画 View.h”、“实时动画 View.cpp”相应函数添加如下代码

说明：下面仅列出实时动画 View.h、实时动画 View.cpp 的全部代码，其它文件代码全部为系统自动建立，无需改正，为节省篇幅此处省略，

详见光盘。下面代码中黑体部分为手工输入，其它代码为系统生成。

```
// 实时动画 View.h : interface of the CMyView class
```

```
.
```

```
.
```

```
.
```

```
class CMyView : public CView
```

```
{
```


程序代码见纸书

}

10.6 练习题

1. 传统动画和计算机动画的区别？
2. 计算机动画研究的内容？目前主要应用领域？
3. 关键帧动画与算法动画？

4. 利用分别应用帧动画和实时动画原理，在 Visual C++中编程绘制在高速公路上奔跑的小汽车。