

Visual C++

音频/视频技术

张玲玲 编著

实战与实战



- 全面、深入的案例讲解，几乎囊括了音频/视频技术的所有知识点
- 以经验为后盾、实践为导向，深入浅出地讲解案例开发全过程
- 全程案例视频教学，书中所有案例均赠送多媒体语音视频讲解

清华大学出版社

Visual C++

音频/视频技术 开发 与 实战

张玲玲 编著

清华大学出版社
北 京

内 容 简 介

本书由浅入深地讲解了使用 Visual C++ 2010 开发音频和视频项目的基本知识,并通过示例来讲解具体的实现流程。

全书内容分为 16 章,详细讲解了使用各种软件 and 平台进行音频/视频多媒体编程的技术,以案例为对象展示实现过程,分析技术难点。主要内容包括 DirectSound 音频开发、DirectShow/VFW 视频开发、MMX/SSE 多媒体汇编编程、DM642 DSP 音频和视频算法优化、MPEG-4/H.264 编码、CCS 开发、XviD CODEC 编码和解码。

本书系统地介绍了使用 Visual C++ 2010 进行流媒体编程的基本思路和方法,采用案例为主的叙述方式,将大量的理论和技术融入具体的案例剖析中。书中采用的案例均来源于作者的实际开发工作,具有很好的实用价值,方便广大开发者在开发中进行参考或直接应用。随书所附光盘包含书中示例的源文件,并包括示例讲解视频,便于读者加深对项目示例的理解。

本书内容丰富、结构安排合理、工程实用性强,能够为广大数字媒体或音频/视频开发人员、高等院校相关专业的课程设计、毕业设计提供参考,同时也可以作为科研单位、企业进行流媒体开发的技术指导用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Visual C++音频/视频技术开发与实战/张玲玲编著. --北京:清华大学出版社,2012.3

ISBN 978-7-302-27986-0

I. ①V… II. ①张… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 018662 号

责任编辑:魏莹 宋延清

封面设计:杨玉兰

责任校对:周剑云

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 刷 者:清华大学印刷厂

装 订 者:三河市新茂装订有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:39.25

字 数:950 千字

(附 DVD1 张)

版 次:2012 年 3 月第 1 版

印 次:2012 年 3 月第 1 次印刷

印 数:1~4000

定 价:75.00 元

前 言

2006—2008 年是土豆、优酷、酷 6 和腾讯等各大视频网站蓬勃发展的时期，这些网站吸引了众多的用户群体。

此期间也是 PPLive、PPStream 和 UUSee 等各大网络电视迅速发展的时期，还是迅雷、电驴等下载工具迅速普及的年代。

2009 年末，中国开启了移动 3G 时代，它标志着用移动手机设备高速上网变成了现实，让手机看网络视频的梦想成真了。

.....

为了满足广大编程爱好者的学习要求，也为了满足一些技术人员的钻研需求，作者花费数年精力精心策划并编写了这本书，期待能够引导读者逐步学会音频开发、视频开发、编码技术、解码技术和网络监控技术。

1. 本书内容

本书由浅入深地讲解了使用 Visual C++ 2010 开发音频和视频技术的知识，并通过具体的示例来讲解实现流程。

全书分为 16 章，具体内容如下。

第1章	介绍音频/视频文件格式，编解码知识及开发工具。
第2章	Visual C++ 开发的技术基础和核心——MFC。
第3章	介绍使用DirectSound实现音频开发的基本方法。
第4章	介绍使用DirectSound实现混音和3D音频的具体方法。
第5章	介绍使用Visual C++实现音频录制的方法。
第6章	详细介绍使用DirectMusic实现音频处理的具体方法。
第7章	介绍DirectShow开发环境的搭建方法。
第8章	详细剖析DirectShow的核心技术。
第9章	详细介绍DirectShow技术的常见应用。
第10章	详细介绍使用VFW技术处理AVI视频的知识。
第11章	详细讲解MMX编程技术的基本知识。
第12章	详细讲解SSE编程技术的基本知识。
第13章	详细讲解DM642算法和芯片的核心技术。
第14章	详细讲解MPEG-4编/解码技术的基本知识。
第15章	和读者一起讨论H.264/AVC的具体编码算法的基本知识。
第16章	用Visual C++ 2010开发一个视频监控系统。

2. 本书的特点和优势

(1) 科学的知识划分

在具体内容编排上,根据 VC++视频开发中不同领域知识点的难易程度,为读者规划出最佳的学习模块。读者只要按照章节去学习,就能轻松地掌握这门技术,并获得最优的学习效率和最佳的学习效果。

(2) 知识点的通俗性和全面性

书中用最通俗的语言讲解了 VC++视频开发的各个知识点,遵照循序渐进、由浅入深的原则,便于读者对内容的理解。本书不但涉及了此项技术的常用领域,而且对高难度的应用进行了详细的介绍,并对应地提出了问题的解决方案。

(3) 典型的示例,深入性的示例讲解

本书在讲解基础知识的过程中穿插讲解了对应的示例,并且针对每个重要的知识点,始终以对应示例的讲解来加深对知识的理解。针对重要知识点或示例,给读者提出了注意事项、建议忠告和使用技巧,使读者的知识能够得到升华。

(4) 实用性

本书注重实用性,以经验为后盾,以实践为导向,以实用为目标,深入浅出地讲解了在开发过程中的种种问题。特别是,在讲解时非常注重实践与理论的形象结合,为了使读者能够通过读懂源代码来掌握这项技术,在示例中还绘出了流程图。本书的所有源代码都已调试通过,并且放在本书所附带的光盘中,读者拿来即可使用。

(5) 全面性

本书内容全面,从基本的语法入手,以恰当的示例来引导,由浅入深地讲解各门技术的基本理论知识,所讲解的内容几乎囊括了此技术的所有知识点,其中许多内容是市场上其他同类书籍较少涉及的。

3. 本书读者对象

本书在内容安排上由浅入深,写作风格上采用层层剥洋葱式的分解,充分以示例举证,非常适合入门级的 VC++视频开发技术的初学者阅读,同时也适合具有一定 C++语言基础,想对 C++开发技术进一步了解和掌握的中级人员阅读。如果你是下列类型的读者,本书会带领你迅速进入 C++开发殿堂:

- 有一定 VC++开发经验的读者。
- 从事单片机开发和嵌入式开发的研究人员和工作人员。
- 高等院校相关专业的学生,或需要编写论文的学生。
- 从事视频开发、音频开发和解码编码等相关工作的技术人员。

4. 致谢

本书由张玲玲编写,参与编写的还有王振丽、熊斌、王教明、万春潮、郭慧玲、侯恩静、吴善才、程娟、王文忠、陈强、何子夜。

在编写过程中得到了清华大学出版社编辑很大的帮助,在此对他们表示衷心的感谢。由于作者水平有限,书中难免存在一些不足和错误之处,读者使用本书时若遇到问题,可以发送邮件到 729017304@qq.com,我们会及时回复。

目 录

第一篇 基础篇

第 1 章 音频/视频技术的现状	3	2.1.1 了解 MFC 编程的特点	38
1.1 了解一些音频、视频文件格式	3	2.1.2 关于 MFC 类的分层结构	39
1.1.1 音频格式	3	2.1.3 MFC 中的 WinMain()函数	39
1.1.2 视频格式	5	2.2 消息交流的意义	40
1.2 常用编解码技术介绍	6	2.2.1 消息概述	40
1.2.1 音频的编码和解码	6	2.2.2 Windows 句柄	42
1.2.2 视频的编码和解码	9	2.2.3 传送消息	42
1.2.3 音频视频编码方案	10	2.2.4 消息与事件	43
1.3 编/解码技术的发展	10	2.3 使用对话框	43
1.4 不得不说的音视频处理平台	12	2.3.1 构建对话框应用程序	44
1.5 必备的常用音视频开发工具	12	2.3.2 设置对话框属性	47
1.5.1 Visual C++集成开发工具	12	2.3.3 常见对话框的种类	52
1.5.2 多媒体 API——DirectX	17	2.4 控件	58
1.5.3 CCS 开发系统	33	2.4.1 Windows 公共控件	58
第 2 章 与 MFC 深情相拥	38	2.4.2 ActiveX 控件	59
2.1 MFC 印象	38	2.5 MSDN 中的巨大资源	62

第二篇 DirectX 篇

第 3 章 DirectSound, 天籁之音靠你实现	67	3.3.2 使用 DirectSound 设备 对象	74
3.1 什么是缓冲区	67	3.3.3 使用 DirectSound 缓冲区 对象	78
3.1.1 缓冲区的种类	67	3.4 神奇的 WAV 数据	87
3.1.2 缓冲区通知	67	3.4.1 WAVE 文件结构	87
3.2 DirectSound 开发基础	69	3.4.2 使用 WAV 数据	88
3.2.1 DirectSound 和 Direct Music	69	3.4.3 读取 WAV 数据	89
3.2.2 DirectSound SDK 的结构	70	3.4.4 计算 WAV 声音的持续 时间	89
3.2.3 DirectSound 的播放功能	71	3.4.5 三种播放 WAV 文件的 方法	90
3.2.4 开发步骤	72		
3.3 DirectSound 音频开发对象	73		
3.3.1 DirectSound 对象介绍	73		

3.5	DirectSound 环境也需要搭建	95
3.5.1	DirectSound 开发环境配置	95
3.5.2	配置开发环境	97
3.6	梳理播放音频文件的过程	100
3.6.1	第一步: 创建 DirectSound 对象	100
3.6.2	第二步: 设置设备协作级别	100
3.6.3	第三步: 创建播放缓冲区	101
3.6.4	第四步: 创建 DirectSound Buffer 对象	104
3.7	小试牛刀——打造一个多彩的音频播放器	106
3.7.1	创建项目程序框架	106
3.7.2	添加控件和变量	107
3.7.3	封装 CDirectSound 类实现 DirectSound	108
3.7.4	对话框界面操作	118
3.7.5	编译和调试	121
3.8	总结 DirectSound 音频编程的步骤	122

第 4 章 混音和 3D 音效编程 124

4.1	混音开发技术	124
4.1.1	混音技术原理	124
4.1.2	小试牛刀——实现三路音频混音	125
4.2	DirectSound 3D 技术	130
4.2.1	DirectSound 3D 介绍	130
4.2.2	听众空间参数	131
4.2.3	获取 3D 听者	133
4.2.4	三种处理模式	134
4.2.5	声音的锥效应	135
4.2.6	DirectSound 3D 缓冲区对象	135
4.2.7	试听声音特技	136
4.3	小试牛刀——实现 3D 音效	138
4.3.1	创建项目程序框架	138
4.3.2	添加控件和变量	139

4.3.3	实现流程分析	141
4.3.4	编译和调试	155

第 5 章 录制音频 158

5.1	如何实现声音的录制	158
5.1.1	录音的基本原理	158
5.1.2	剖析实现流程	159
5.2	小试牛刀——用 DirectSound 录制音频	167
5.2.1	封装类	167
5.2.2	管理声音的函数	170
5.2.3	编译和调试	180
5.3	Windows CE 下的录音程序	181
5.3.1	Windows CE 环境	181
5.3.2	使用缓冲实现音频回环录制	182
5.3.3	小试牛刀——一个功能强大的录音程序	187

第 6 章 DirectMusic 音频开发 195

6.1	什么是 DirectMusic	195
6.1.1	组成	195
6.1.2	功能	195
6.2	DirectMusic 的体系结构	196
6.2.1	核心层和接口	196
6.2.2	DirectMusic 的对象和接口	196
6.2.3	DirectMusic 的数据流	199
6.2.4	DirectMusic 的消息	200
6.2.5	可下载银色技术(DLS)	203
6.2.6	Microsoft 软件合成器	203
6.2.7	小试牛刀——打造一个音乐演奏器	203
6.3	小试牛刀——打造回音和测量工具	206
6.3.1	实例分析	206
6.3.2	具体实现	207

第 7 章 DirectShow 来了 220

7.1	DirectShow 基础	220
-----	---------------------	-----

7.1.1 DirectShow 介绍	220	8.5 时间的意义	262
7.1.2 剖析 DirectShow 的构成	221	8.5.1 参考时钟	262
7.1.3 常用的 DirectShow 接口	224	8.5.2 时钟时间	263
7.2 搭建 DirectShow SDK 开发环境	225	8.5.3 时间戳	263
7.2.1 获取 DirectShow SDK	225	8.5.4 实时源(Live Source)	264
7.2.2 安装 DirectShow SDK	225	8.6 动态重建可靠吗	266
7.3 配置 DirectShow SDK 开发环境	228	8.6.1 动态重连	266
7.3.1 生成 DirectShow SDK 库	228	8.6.2 Filter 链	268
7.3.2 配置 Visual C++ 2010	233	8.7 剖析 SDK 基类源码	269
7.3.3 几个调试问题的解决方法	235	8.7.1 COM 实现部分和部分辅助 源码	270
7.4 小试牛刀——使用 DirectShow 技术	239	8.7.2 与 Pin 密切相关的类	274
7.4.1 滤镜和图像滤镜介绍	239	8.7.3 CBaseFilter 类	277
7.4.2 编写一个 DirectShow 程序	240	8.7.4 CSource 类	280
7.4.3 使用 graphedit.exe 播放 AVI 文件	241	8.7.5 CTransformFilter 类	281
7.4.4 编程实现播放 AVI 文件	242	8.7.6 CBaseRenderer 类	282
第 8 章 DirectShow 核心技术分析.....	246	8.7.7 CBaseAllocator 类	284
8.1 Filter Graph 及其组成	246	8.8 小试牛刀——开发一个视频 播放器	286
8.1.1 DirectShow 中的 Filter	246	8.8.1 系统分析和设计	287
8.1.2 Media Type(媒体类型)	247	8.8.2 实现媒体控制类	294
8.1.3 媒体样本 Samples 和分配器 Allocators	250	8.8.3 创建播放器主题	304
8.2 构建 Filter Graph	250	8.8.4 添加背景图片	313
8.2.1 链表构建组件	251	第 9 章 DirectShow 基本应用	314
8.2.2 Graph 构建概述	251	9.1 DirectShow 的基本应用	314
8.2.3 智能连接详解	252	9.1.1 视频提交(Video Rendering).....	314
8.3 数据流在 Filter Graph 中的流动	253	9.1.2 枚举系统的设备和过滤器	317
8.3.1 传输(Transports)	255	9.1.3 连接两个 Filter	321
8.3.2 样本 Samples 和分配器 Allocators	255	9.1.4 获得 Filter 或者 Pin 的接口 指针	322
8.3.3 Filter 状态	257	9.1.5 查找与某个 Filter 的上下相连 的 Filter	324
8.3.4 拉模式	258	9.2 DirectShow SDK 典型开发案例	327
8.4 事件通知机制(Event Notification)	259	9.2.1 用 DirectShow 实现视频 马赛克处理	327
8.4.1 从队列中取事件	259	9.2.2 DirectShow 驱动摄像头	334
8.4.2 当事件发生时	260	9.2.3 DirectShow 视频采集	338
		9.2.4 录制视频	349

9.2.5 音频采集	354	10.3.2 创建视频窗口	374
9.2.6 DirectShow 在网络方面的 应用	357	10.3.3 创建回调函数	375
9.3 编写自己的 Filter	358	10.3.4 设置视频图像	377
9.3.1 配置开发环境	358	10.3.5 捕获预览视频	378
9.3.2 创建工程	359	10.3.6 设置视频格式和图像 参数	380
9.3.3 为 Filter 创建入口函数	360	10.3.7 编码、解码捕获的视频	381
9.3.4 实现 Filter 的类厂对象	361	10.3.8 打开文件操作	382
9.3.5 实现自己的 Filter	363	10.3.9 窗口诊断处理	383
9.3.6 最后的总结	365	10.3.10 实现 CvfwappDoc 类的 接口	383
第 10 章 以 VFW 实现视频处理	367	10.4 视频压缩管理器	385
10.1 揭开 VFW 的神秘面纱	367	10.4.1 视频压缩管理器初步	385
10.1.1 VFW 概述	367	10.4.2 视频压缩管理器初步	386
10.1.2 VFW 开发流程	367	10.4.3 三个重要的函数	390
10.1.3 VFW 视频捕获流程	368	10.4.4 小试牛刀——使用 VCM 编码、解码	392
10.1.4 视频编辑和播放	369	10.5 使用 XviD CODEC 算法	397
10.2 VFW 的视频预览	370	10.5.1 XviD 基础	397
10.2.1 基本步骤	370	10.5.2 编译 XviD	400
10.2.2 编码演示	372	10.5.3 小试牛刀——演示 XviD 编码、解码	401
10.3 小试牛刀——用 VFW 实现视频 捕获和预览	373		
10.3.1 建立单文档应用程序	373		

第三篇 编码、解码篇

第 11 章 MMX 带来新的体验	413	第 12 章 SSE 编程	440
11.1 MMX 指令基础	413	12.1 SSE 基础	440
11.1.1 MMX 多媒体扩展	413	12.1.1 SSE 指令	440
11.1.2 MMX 指令剖析	414	12.1.2 其他的几种技术	441
11.1.3 CPU 寄存器	419	12.2 SSE/SSE2 剖析	442
11.2 MMX 内联汇编的一般原则	420	12.2.1 SSE/SSE2 原理	443
11.3 MMX 在 C++ 中的使用	421	12.2.2 SSE/SSE 编程设计	444
11.4 MMX 代码开发规则	423	12.2.3 SSE 指令简述	446
11.4.1 开发规则	423	12.3 小试牛刀——SSE 编程实例	458
11.4.2 开发建议	423	12.3.1 SSETest 项目	458
11.4.3 MMX 代码开发策略	424	12.3.2 SSESample 项目	461
11.5 小试牛刀——MMX 编程实例	425		

第 13 章 最流行的 DM642..... 465

- 13.1 认识 TMS320DM642 465
- 13.2 CCS 集成开发环境简介 466
 - 13.2.1 CCS 集成开发环境简介..... 466
 - 13.2.2 DSP/BIOS 插件 467
 - 13.2.3 硬件仿真和实时数据交换 ... 470
 - 13.2.4 CCS 文件和变量 470
- 13.3 常见应用介绍..... 473
 - 13.3.1 DSP DM642 上移植
Yaffs2..... 473
 - 13.3.2 基于 TMS320C64x/DM64x
的视频编码优化..... 475
 - 13.3.3 DM642 视频采集与回放..... 478
 - 13.3.4 视频会议系统的主流解决
方案与技术趋势..... 482
- 13.4 DSP 视频算法优化..... 487
 - 13.4.1 基于 DSP 视频算法系统优化
策略..... 488
 - 13.4.2 基于 DSP 的信道译码算法
优化..... 491
- 13.5 基于 DSP 的 JPEG 图像解码算法的
实现..... 495
 - 13.5.1 JPEG 图像算法基础..... 496
 - 13.5.2 小试牛刀——JPEG 编码算法
实现 DSP 优化..... 498

第 14 章 MPEG-4 编/解码分析..... 508

- 14.1 MPEG-4 编/解码基础 508
 - 14.1.1 基于对象的 MPEG-4 视频
编码..... 508
 - 14.1.2 XviD 介绍 508

- 14.2 MPEG-4 编/解码剖析 509
 - 14.2.1 MPEG-4 视频编码 509
 - 14.2.2 MPEG-4 视频解码 523
 - 14.2.3 销毁解码器 535
- 14.3 DM642 下实现 XviD CODEC
优化 537
 - 14.3.1 水平方向插值补偿..... 537
 - 14.3.2 垂直方向插值补偿..... 538
 - 14.3.3 斜角插值补偿 540
- 14.4 运行 XviD CODEC 系统 542
 - 14.4.1 YUV 原始视频数据及其
显示 542
 - 14.4.2 VC 平台编译\运行
XviD CODEC 543
 - 14.4.3 系统运行效果 545

**第 15 章 H.264/AVC 带来不一样的
感觉 547**

- 15.1 H.264/AVC 基础 547
 - 15.1.1 技术特点 547
 - 15.1.2 编码结构与格式 549
- 15.2 X264 视频编码从此开始..... 553
 - 15.2.1 X264 应用工程..... 554
 - 15.2.2 libx264 编码库 561
 - 15.2.3 视频帧的存放和获取..... 562
 - 15.2.4 视频帧压缩编码 VCL..... 565
 - 15.2.5 码流 NAL 打包输出 572
- 15.3 X264 核心模块的 MMX/DSP
汇编优化 573
 - 15.3.1 MMX/DSP 汇编优化 574
 - 15.3.2 DM642 线性汇编优化 574

第四篇 综合实战篇**第 16 章 一切皆有可能——打造一个
视频监控系统..... 581**

- 16.1 系统分析..... 581
 - 16.1.1 功能分析..... 581

- 16.1.2 系统目标 582
- 16.1.3 功能结构 582
- 16.1.4 系统预览 582

16.2	设计数据库.....	584	16.4.2	设计主窗体	592
16.3	开发前的准备	584	16.5	其他辅助功能模块	605
16.3.1	选购监控卡	584	16.5.1	设计管理员登录模块.....	605
16.3.2	准备软件.....	585	16.5.2	监控管理	608
16.3.3	安装监控卡	585	16.5.3	广角自动监控模块.....	611
16.4	实现核心模块.....	586	16.5.4	视频回放模块	613
16.4.1	设计公共模块.....	586	参考文献.....		615



第一篇

基础篇

无论是多媒体视频服务站点优酷和土豆，还是靠美轮美奂的 3D 视觉和音效冲击力吸引观众眼球的《阿凡达》，都有一个共同点，那就是都与视频、音频相关。

如今无论是在家中欣赏大片，还是在上下班的路上听音乐，忽然间发现，我们现在的生活已经离不开多媒体音频和视频服务了。

本书是奉献给音频、视频开发工程师们的参考书，也是奉献给即将进入这一行业的初学者们的教材。本书的序幕由此拉开。

在第一篇的内容中，将呈献给大家最基本的入门知识。本篇主要涉及如下两章的内容。

- 第 1 章 音频/视频技术的现状
- 第 2 章 与 MFC 深情相拥

第 1 章 音频/视频技术的现状

过去的 10 年是互联网技术迅猛发展的时代，网络仿佛在一夜之间进入了寻常百姓的家中。网络的发展促进了社会的进步，信息时代给我们的人生带来了崭新的体会。在网络迅速普及之时，视频技术受到了普通人的青睐，例如视频点播、互联网电视、移动视频等多姿多彩。然而作为开发人员，则亟需了解——这些神奇的功能是怎样实现的呢？在本书中，将引导读者循序渐进地掌握这些神奇功能的实现技术。

1.1 了解一些音频、视频文件格式

音频和视频两者之间有着密切的联系，例如好莱坞科幻大片《阿凡达》无论其视频界面多么美轮美奂，也离不开震撼的音效做点缀。

对于音频和视频的格式，多数读者都能随便说出几种，如 WAV、MP3、AVI 等。其实除了这几种格式外，还有很多其他音频和视频格式，接下来将分类全面介绍。

1.1.1 音频格式

(1) MP3

MP3 是 Fraunhofer-IIS 研究所的杰出成果。在 MP3 出现前，以有损方式进行音频编码最好的效果能够压缩到 4:1。而强大的 MP3 可以实现 12:1 的高压缩比，正是因为这种性能的巨大提升，并且在实现高压缩比的同时还能够保持较好的音质，所以使得 MP3 迅速地流行起来。

MP3 是世界上第一个实用化的有损压缩编码方案，几乎所有的播放软件都支持它，也曾经产生了许多第三方的编码工具。但是自从 Fraunhofer-IIS 宣布对编码器征收版权税之后，对应的编码工具逐渐淡出了人们的视线。编码工具的淡出注定会造成 MP3 的疲态，就像历史更迭一样，你如果静止不前，就将迟早被后来者替代。

在 MP3 之后推出了很多新的编码技术，这些新技术能在相同的比特率下实现比 MP3 更优秀的音质。但是因为 MP3 的影响力实在太大了，当前市面上随处可见支持 MP3 的软件，而且主流手机也几乎都支持 MP3。总之，虽然有很多新格式拥有比 MP3 更好的音质，但是 MP3 依然是世界上最流行的音频压缩技术，并且将在今后的一段时间内继续流行下去。

(2) MP3PRO

为了使 MP3 在将来仍然能保持竞争力，Fraunhofer-IIS 研究所、Coding Technologies 公司和 Thomson Multimedia 公司通过共同研发，推出了新格式——MP3PRO。MP3PRO 与先前的 MP3 相比，最大优势是在 64k 的比特率下仍然能提供近似 CD 的音质，而 MP3 需要 128k 的比特率。MP3PRO 源于 MP3，它专门对 MP3 技术中损失的音频细节进行了独立编码处理，并捆绑在原来的 MP3 数据上，在播放的时候通过二次合成以获得更好的音质。

效果。

MP3PRO 与 MP3 是兼容的，所以它的文件类型也是 MP3。虽然 MP3PRO 是一种优秀的技术，但是由于技术专利费用的问题以及其他技术提供商的竞争，MP3PRO 并没有得到普及。感兴趣的读者可以登录 Coding Technologies 的网站，免费下载 Demo 播放/压缩工具和 Winamp 播放插件。

(3) WAV

WAV 是微软 Windows 系统本身自带的音频格式，因为 Windows 系统的普及性，此格式已经成了当前最主流的音频格式之一。WAV 文件里面存放的每一块数据都有自己独立的标识，通过这些标识可以告诉用户究竟是什么数据。WAV 格式的设计非常灵活，该格式本身与任何媒体数据都不冲突，只要有软件支持，我们甚至可以在 WAV 格式里面存放图像。在 Windows 平台上，通过 ACM 结构及相应的驱动程序，例如 CODEC 编码/解码器，可以在 WAV 文件中存放超过 20 种的压缩格式，比如 ADPCM、GSM、CCITT G.711、G.723 和 MP3 等。

(4) Windows Media

Windows Media 也是一种网络流媒体技术，本质上跟 Real Media 是相同的。Microsoft 推出 Windows Media 的目的是抗衡 Real Media，微软在推出 Windows Media 后，声明不再支持其他音频压缩技术。所以从 Windows XP 系统开始，把原来提供的 MP3 压缩功能都拿掉了。

(5) MIDI

MIDI 技术最初应用在电子乐器上，用来记录乐手的弹奏，以便以后重新播放。当在电脑中引入支持 MIDI 合成的声音卡之后，MIDI 逐渐成为了一种音频格式。在 MIDI 中不但有乐谱，而且还记录了每个音符的弹奏方法。MIDI 有两个版本，分别是 General MIDI 和 General MIDI 2。在 MIDI 上还衍生了许多第三方的非标准技术，例如比较著名的 X-MIDI。

X-MIDI 是由日本 YAMAHA(雅马哈)公司发明的，在原有的 MIDI 具有 128 种乐器的基础上扩充到了 512 种，并增加了更多的演奏控制，配合 YAMAHA 自己的波表播放软件或支持 X-MIDI 的硬件，可以还原出非常动听和接近真实乐器效果的音乐。

(6) MP4

MP4 是美国网络技术公司 GMO 和 RIAA 联合公布的一种新的音乐格式。MP4 在文件中采用了保护版权的编码技术，只有特定的用户才可以播放，这样就有效地保证了音乐版权的合法性。MP4 的压缩比达到了 1:15，虽然比 MP3 的体积更小，但是并没有使音质降低。但因为只有特定的用户才能播放的原因，所以没有被普及。

(7) QuickTime

QuickTime 是苹果公司于 1991 年推出的一种数字流媒体产品，QuickTime 能够支持几乎所有主流的个人计算平台，可以通过互联网提供实时的数字化信息流、工作流与文件回放功能。

(8) MD

MD(Mini Disc)是 Sony 公司的产品，因为它使用了 ATRAC 算法来压缩音源，所以能够在一张小小的盘中存储 60~80 分钟以 44.1kHz 采样的立体声音乐。ATRAC 算法是一套

基于心理声学原理的音响译码系统, 可以把 CD 唱片的音频压缩到原来数据量的 1/5, 而声音质量没有明显的损失。ATRAC 可以利用人耳听觉的心理声学特性, 以及人耳对信号幅度、频率、时间的有限分辨能力, 在编码时将人耳感觉不到的成分不编码, 不传送, 这样就减少了某些数据量的存储, 从而达到了既保证音质又缩小体积的目的。

(9) AAC

AAC 是由 Fraunhofer IIS-A、杜比和 AT&T 共同开发的一种音频格式, 是 MPEG-2 规范的一部分。AAC 音频算法在压缩能力上优势巨大, 远远超过了以前的一些压缩算法, 例如 MP3 等。AAC 可以同时支持多达 48 个音轨、15 个低频音轨、更多种采样率和比特率、多种语言的兼容能力、更高的解码效率。AAC 可以在比 MP3 文件缩小 30% 的前提下提供更好的音质效果。

1.1.2 视频格式

(1) AVI

AVI 最早是由 Microsoft 开发的, 含义是 Audio Video Interactive, 表示把视频和音频编码混合在一起进行存储。AVI 格式有比较多的限制, 只能有一个视频轨道和一个音频轨道, 但还可以有一些附加轨道, 例如文字等。

注意: 在一些非标准插件中, 可以加入最多两个音频轨道。

(2) WMV

WMV 即 Windows Media Video, 是微软公司开发的一组数字视频编解码格式的统称。WMV 采用了 ASF(Advanced Systems Format)封装格式, 通过 ASF 封装的 WMV 具有“数字版权保护”功能。

(3) MPEG

MPEG 是 Moving Picture Experts Group 的缩写, 是一个国际标准组织(ISO)认可的媒体封装形式, 受到大部分机器的支持。MPEG 有多种存储方式, 可以适应不同的应用环境。MPEG 的控制功能丰富, 可以有多个视频、音轨、字幕等。为了适应时代潮流, MPEG 的简化版本 3GP 专门用于准 3G 手机。

(4) Real Video 或 Real Media(RM)

RM 是 Real Networks 开发的一种文件容器, 通常只能容纳 Real Video 和 Real Audio 编码的媒体, 带有一定的交互功能, 允许编写脚本来控制对视频的播放。

(5) QuickTime Movie

QuickTime Movie 是苹果公司开发的容器, 因为苹果电脑在专业图形领域的统治地位, 所以 QuickTime(QT)格式基本上成为电影制作行业的通用格式。1998 年 2 月 11 日, 国际标准化组织(ISO)认可 QuickTime 文件格式作为 MPEG-4 标准的基础。QT 可存储的内容相当丰富, 除了视频、音频以外, 还可支持图片、文本字幕等。

(6) 3GP

3GP 是一种 3G 流媒体的视频编码格式, 是为了配合 3G 网络的高传输速度而开发的, 是当前手机中最为常见的一种视频格式。

(7) ASF

ASF 是 Advanced Streaming Format 的缩写, 是微软为了与 Real Player 竞争而推出的一种产品。ASF 可以直接在网上观看视频节目, 并且由于它使用了 MPEG4 的压缩算法, 所以拥有很好的压缩率和图像质量。

(8) FLV

FLV 是 Flash Video 格式的简称, 是 Macromedia 公司开发的属于自己的流媒体视频格式, 是基于 Sorenson 公司的压缩算法而开发出来的。FLV 是一种新的视频格式, 由于它形成的文件极小, 加载速度也极快, 这就使得从网络观看视频文件成为可能。FLV 的意义重大, 有效地解决了视频文件导入 Flash 后使导出的 SWF 格式文件体积庞大, 无法在网络上很好地使用等缺点。

(9) RMVB

RMVB 是由 RM 视频格式升级而延伸出的新型视频格式, RMVB 打破了 RM 格式使用的平均压缩采样的方式, 在保证平均压缩比的基础上能更加合理地利用比特率资源。在静止和动作场面少的画面场景采用较低编码速率, 从而节约更多的带宽空间, 这些带宽会在出现快速运动的画面场景时被利用掉。这就在保证了静止画面质量的前提下, 大幅地提高了运动图像的画面质量, 从而在图像质量和文件大小之间达到了平衡。

1.2 常用编解码技术介绍

通过前面的介绍, 我们了解了常见的视频和音频格式。因为这些格式都是通过解码和编码技术来实现音频和视频播放功能的, 所以编码和解码技术才是整个音频开发和视频开发的核心, 也是本书的重点。在接下来的内容中, 将简要介绍音频和视频编解码技术的基本知识。

1.2.1 音频的编码和解码

处理音频数据的过程分为编码和解码两个过程, 这两个过程的具体说明如下。

- 编码: 把存放在波形文件里的数字音频数据转换为压缩的形式, 也就是比特流的形式。
- 解码: 把比特流重建为波形文件。

同一种音频文件可以支持多种编码格式, 但是大多数音频文件仅仅支持一种音频编码。在现实中, 常用的音频文件格式有如下两类。

- (1) 无损格式: 例如 WAV、PCM、APE、TTA、FLAC 和 AU。
- (2) 有损格式: 例如 MP3、OGG Vorbis、WMA 和 AAC。

1. 无损压缩格式

无损压缩格式是指毫无损失地将声音信号进行压缩的音频格式。现实中常见的 MP3、WMA 等都是有损压缩格式, 它们与无损格式 WAV 相比, 都丢失了相当多的信号, 当然