

DirectShow 技术的应用研究与开发^①

孟月华, 邓基园

(广州海格通信集团股份有限公司, 广州 510663)

摘要: 研究了采用 DirectShow 技术从音视频设备获取媒体数据并进行后期处理的问题. 随着多媒体技术应用的越来越广, 用户对多媒体产品的要求也越来越高. 早期的多媒体软件与音视频设备高度关联, 存在着通用性差、功能扩展难等诸多局限, 不能满足用户需求. 本文基于 DirectShow 系统对硬件的充分支持以及硬件设备无关性的特点, 提出了一种采用 DirectShow 技术的设计方法, 解决了多媒体软件受限于设备的问题. 经测试表明, 该方法达到了预期, 在多媒体领域将有良好的应用前景.

关键词: 多媒体; 音视频; 应用; DirectShow; Filter Graph Manager; Filter

Application and Development of DirectShow Technology

MENG Yue-Hua, DENG Ji-Yuan

(Guang Zhou Haige Communications Group Incorporated Company, Guangzhou 510663, China)

Abstract: The research has adopted the DirectShow technology to get media data from audio and video equipment and solve after-treatment problems. With the increasingly wide application of multimedia technology, users's requirements of multimedia products are becoming higher and higher. The early multimedia software, highly correlated with audio and video equipment, and with limitations such as poor commonality and the function expansion difficulty, can not meet users's needs. Based on the characteristics of DirectShow system's full support to hardware, and hardware's independence, this article has proposed a design method of applying DirectShow technology to solve problems that multimedia software is limited by equipment. Tests show that, the method has achieved the expectation and will have a good application prospect in the field of multimedia.

Key words: multimedia; audio and video; application; DirectShow; filter graph manager; filter

随着计算机多媒体技术的快速发展, 多媒体凭借着自身的优势越来越受到广泛关注和应用, 它的出现已经改变了人们传统意义上的工作与生活方式. 计算机多媒体技术是处理文字、数据、图像、声音、视频等信息的技术, 其中数据压缩和编码技术、媒体同步技术、多媒体网络技术等多媒体发展需要的关键技术^[1], 技术性强、难度高, 给广大多媒体应用开发人员提出了巨大的挑战.

DirectShow 为在 Windows 平台上处理各种媒体文件的回放、音视频采集等高性能要求的多媒体应用提供了完整的解决方案^[2]. 本文利用 Visual Studio 开发工具, 结合 DirectShow 特点, 阐述了一般音视频处理软

件的开发过程.

1 DirectShow 简介

DirectShow 是 Microsoft 媒体开发 DirectX 中的一部分, 是 Microsoft 公司提供的一套在 Windows 平台上进行流媒体处理的开发包. 采用 DirectShow 技术, 我们可以便捷地从音视频采集设备上捕获数据, 并且进行相应的后期处理. 它支持多种媒体格式, 包括 Avi、Asf、Mpeg、Dv、Mov、Wave 等等. 此外, DirectShow 还集成了 DirectX 其它部分 (如 DirectDraw、DirectSound) 的技术, 直接支持 DVD 的播放、视频的非线性编辑以及与数字摄像机的数据交换等^[1-3].

^① 收稿时间:2012-11-25;收到修改稿时间:2013-01-05

2 DirectShow工作原理

2.1 DirectShow 系统框架

DirectShow 系统框架为多媒体应用开发提供了一种标准, 该框架封装了大量的基本组件和各类应用案例, 简化了多媒体应用开发流程. 应用程序与 DirectShow 系统以及 DirectShow 所支持的软件、硬件之间的关系如图 1 所示.

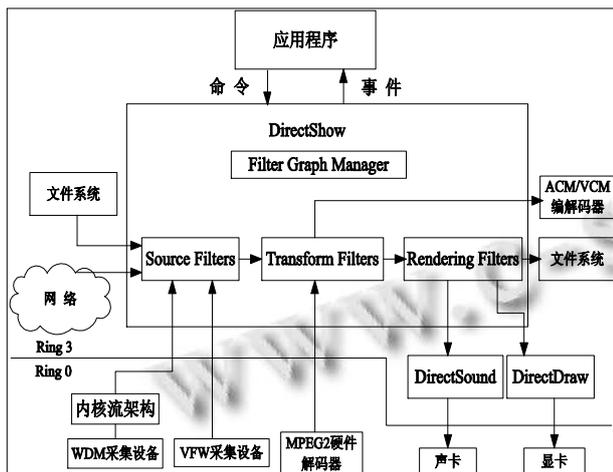


图 1 系统框图

DirectShow 主要由 Filter Graph 构成, Filter Graph 中包含了各种 Filter, 这些 Filter 在 Filter Graph Manager 的统一控制下, 按一定顺序连接在一起, 从而使数据在由 Filter 组成的链表中流动^[4], 图 1 中的箭头表示 Filter 链表中的数据流的方向.

在 DirectShow 系统框架外, 是与用户直接交互的应用程序. 应用程序要根据实际的功能需求引入相应的 Filter, 建立 Filter Graph, 然后通过 Filter Graph Manager 来控制系统的数据处理过程. DirectShow 通过事件通知机制, 实现应用程序与 Filter Graph Manager 之间的交互控制. Filter Graph 运行的时候接收到各种事件, 通过消息的方式发送到用户应用程序, 用户应用程序则根据事件类型做出相应的处理.

2.2 Filter 概述

Filter 是 DirectShow 中最基本的概念, 是 Filter Graph 中最小的功能模块. 对于每个 Filter, 都有其自己特定功能的 Pin, Filter 通过 Pin 来进行它们之间的连接. 如图 1 所示, 按照 Pin 输入输出的方向, 一般 Filter 分为三类: Source Filters、Transform Filter 和 Rendering Filters^[5].

Source Filters: 源过滤器, 该类 Filters 引入数据到

Filter Graph 中, 数据来源可以是文件、设备、网络等, 不同的源过滤器处理不同类型的数据源.

Transform Filters: 变换过滤器, 该类 Filters 获取输入流, 处理数据, 并生成输出流. 变换过滤器对数据的处理包括编解码、格式转换、压缩解压缩等.

Rendering Filters: 提交过滤器, 该类 Filters 在 Filter Graph 中处于最后一级, 它们接收数据并把数据提交给外设进行呈现.

DirectShow 提供了一系列标准的 Filter 用以支持基本的开发, 开发人员也可以根据实际需要开发自己的功能 Filter 来扩展 DirectShow 的应用.

2.3 DirectShow 对硬件的支持原理

DirectShow 通过包装 Filter 实现对音视频硬件设备的支持, 该类 Filter 工作在用户模式下, 在内部完成与硬件驱动的交互^[5-7], 简化开发人员对硬件的处理. DirectShow 集成了很多包装 Filter, 其中最重要的是为采用 WDM 驱动模型的硬件设计的 KsProxy Filter(KsProxy.ax). KsProxy.ax 能代表各种 WDM 模型的音视频设备, 对上层调用定义了统一的调用接口, 应用程序可控制采集格式、视频参数等.

2.4 DirectShow 的常用接口

DirectShow 是基于 COM 标准的应用系统, 大部分功能均通过 COM 接口来完成, 下面是一些常用接口的说明.

(1) IGraphBuilder 接口

用于构造 Filter Graph 的接口, 建立和管理 Filter;

(2) IMediaControl 接口

用于控制媒体流在 Filter Graph 的流动;

(3) IMediaEvent 接口

用于捕获播放过程中发生的事件, 并通知应用程序;

(4) IVideoWindow 接口

用于控制视频窗口的属性及显示控制;

(5) IMediaSeeking 接口

用于查找媒体的接口, 定位流媒体, 实现播放进度的精确控制;

(6) IPin 接口

用于管理两个 Filter 之间的 Pin;

3 视频监控软件开发

下面以一个视频监控软件开发为例说明采用 DirectShow 进行音视频软件的开发过程. 本例中视频

采集设备为一个 USB 接口的摄像头, 软件开发环境采用 Visual Studio 6.0 和 DirectShow SDK(9.0 版本).

3.1 搜索本机的视频设备

DirectShow 使用 WDM Video Capture Filter 支持 WDM 采集卡, 使用 VFW Capture Filter 支持 VFW 采集卡. 这两类采集卡的 Filter 均注册为 CLSID_VideoInputDeviceCategory 类型, 因此需要通过系统设备枚举来创建, 系统枚举流程如图 2 所示.

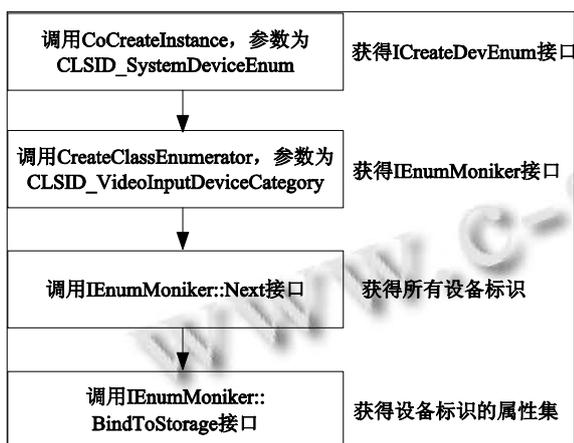


图 2 系统视频设备枚举过程

在枚举过程中存储每个设备的 Friendly Name 与设备标识指针 Device Moniker, 方便用户选择不同的设备进行操作. 本例中搜索到设备“USB 视频设备”(该设备的 Friendly Name).

3.2 创建 Filter Graph

采用 DirectShow 提供的辅助组件 Capture Graph Builder 来创建 Filter Graph, 流程如图 3 所示.

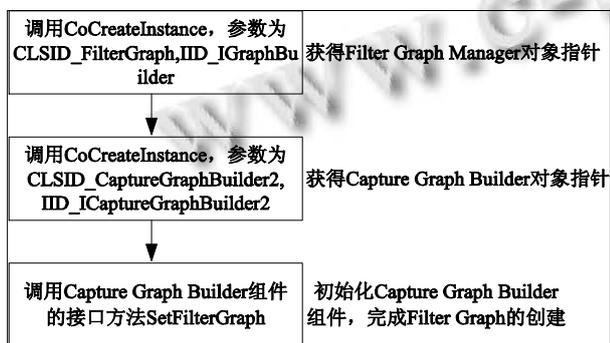


图 3 Filter Graph Manager 创建过程

3.3 获取功能接口对象

在 Filter Graph Manager 创建完成后, 调用该组件的

QueryInterfaces 接口, 获取并存储 IMediaControl、IMediaEventEx、IBasicVideo、IBasicAudio、IVideoWindow 等接口对象. 在多媒体软件中, 利用这些接口对象可以开发出如图像预览、图片快照、音视频播放控制、屏幕控制等具体应用.

3.4 Filter 链路建立

调用 Capture Graph Builder 组件的 RenderStream 接口对 PIN_CATEGORY_PREVIEW 类型的 Pin 完成后续连接, 代码如下:

```

    ICaptureGraphBuilder *pBuild; //第二步获取的对象 Capture Graph Builder
    IBaseFilter *pCap; //第三步获取的对象 Video capture filter
    hr=pBuild->RenderStream
    (&PIN_CATEGORY_PREVIEW,
    &MEDIATYPE_Video, pCap, NULL, NULL);
  
```

3.5 视频显示

调用 IVideoWindow 对象进行指定窗口的显示, 调用 IMediaControl 对象的 Run 方法控制 Filter Graph 开始运行, 代码如下:

```

    RECT winRect;//应用程序传入的窗口对象
    IVideoWindow *pIVideoWin;// 第三步获取的 IVideoWindow 对象
    IMediaControl *pIMediaCtrl;// 第三步获取的 IMediaControl 对象
    pIVideoWin->put_Left(0);
    pIVideoWin->put_Top(0);
    pIVideoWin->put_Width(winRect.right-winRect.left);
    pIVideoWin->put_Height(winRect.bottom-winRect.top);
    pIVideoWin->put_WindowStyle(WS_CHILD|WS_CLIPCHILDREN|WS_CLIPSIBLINGS);
    pIMediaCtrl->Run();
  
```

本软件将上述开发过程的 DirectShow 部分封装为一个独立的类, 包括 Filter Graph Manager 的创建、系统设备的枚举以及对设备各种控制接口的操作等, 上层应用程序只需实例化该类的对象, 即可调用该类提供的公共接口, 完成各种控制功能.

运行该软件, 选择本机枚举到的视频设备(本例中为“USB 视频设备”), 点击“开始”、“抓图”、“停止”、“屏

幕切换”等功能按钮,可观察到视频预览、即时抓图、停止预览以及屏幕切换等功能. 如下图 4 所示.

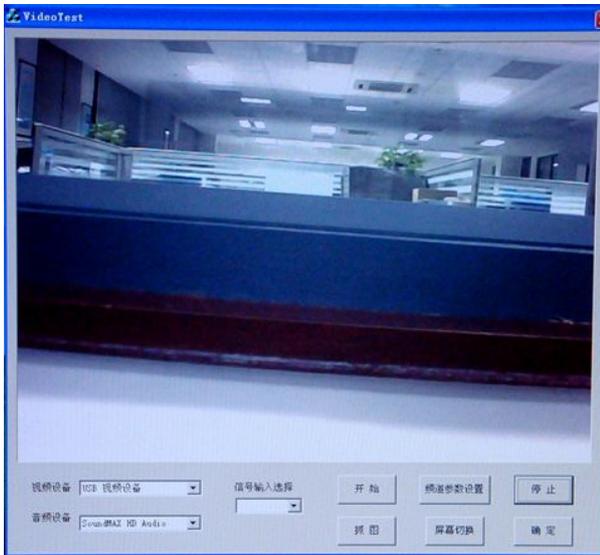


图 4 视频监控软件功能测试图

本文档详细说明了 DirectShow 的基本操作及视频处理部分的设计. 基于 DirectShow 对流媒体的强大解决方案, 应用软件还可引入不同的 Filter 来处理不同的媒体数据流, 实现不同的多媒体应用. 本软件通过引入 TV Tuner Filter, 同时完成了模拟电视卡的简单控制. 在软件主界面上选择电视卡的音频设备和视频设备(安装模拟电视卡驱动后, 系统可枚举到模拟电视卡的音频设备和视频设备), 点击“开始”按钮即可播放电视节目. 采用上述设计方法开发的视频监控模块和有线电视模块已成功应用在多媒体调度终端上, 效果良好.

4 结语

DirectShow 开发包实现了多媒体应用的总体框架和底层的操作, 包括对硬件设备的充分支持. 在该框架基础上, 应用程序开发人员能够专注重音视频采集、多媒体播放、视频会议、视频监控、智能终端等多媒体应用, 不需关注多媒体处理的细节, 较容易开发出具有特色的多媒体应用软件, 不仅可以降低开发成本、缩短开发周期, 还提高了多媒体应用的通用性和扩展性.

本文讨论了 DirectShow 的基本原理, 通过一个视频监控软件的实例描述了采用 DirectShow 来实现视频预览、抓图、电视播放等功能的开发过程. 随着多媒体技术的发展以及用户需求的多样化, DirectShow 技术在多媒体应用领域将有广阔的发展前景.

参考文献

- 1 陆其明. DirectShow 开发指南. 北京: 清华大学出版社, 2003. 1-15.
- 2 文坤, 高胜法. DirectShow 程序设计原理及应用. 计算机系统应用, 2006, 15(3): 25-28.
- 3 周永刚, 张有忱. 基于 DirectShow 与数字减影技术的视频监控. 计算机系统应用, 2009, 18(2): 92-94.
- 4 谢屈波, 王六森, 夏定纯. 基于 DirectShow 的图像采集的实现. 软件导刊, 2010, 9(6): 183-184.
- 5 关一, 张新荣, 董玉友. 基于 DirectShow 的多镜头图像采集系统的设计与实现. 电子测量技术, 2007, 30(6): 73-75.
- 6 郑晓曦, 刘维. 基于 DirectShow 的无线音视频采集与传输系统的研究. 数字技术与应用, 2001, 12: 44-45.
- 7 金钧, 刘锦高. 基于 DShow 的多路视频实时采集处理系统. 信息技术, 2011, 8: 138-140.