

# 基于 Directshow 的 H. 264 网络视频监控客户端实现

彭 锋, 林和志, 黄联芬

(厦门大学, 福建 厦门 361005)

**摘要:** 为了实现视频监控客户端对 H. 264 实时播放, 采用 Directshow 与 MFC 相结合的方法, 用 Directshow 作为视频播放的驱动, 从网络接收数据, 完成缓存, 解码显示及封装成 MFC 可调用的 API, MFC 用来设计户界面, 对播放过程进行相应的控制, 达到了播放从网络接收到的 H. 264 视频的目的. 经过 PC 机上验证, 可以流畅地从网络上接收播放 H. 264 视频流, 并且丢包率很小. 在此用 Directshow 来实时流畅的播放从网络上接收到的 H. 264 视频流。

**关键词:** H. 264; Directshow; MFC; 网络视频监控

中图分类号: T N919-34

文献标识码: A

文章编号: 1004-373X(2011)08-0118-03

## Implementation of Directshow based H. 264 Network Video Monitoring at Client

PENG Feng, LIN He zhi, HUANG Lian fen

(Xiamen University, Xiamen 361005, China)

**Abstract:** In order to realize the real time broadcasting of H. 264 at video monitoring client, a method that combines Directshow with MFC is adopted. Directshow taken as a drive of the video player receives data from the network, performs cache, decoding and display, and provides the API for MFC, which is used to design UI to control the process of playing. After verification on PC, the player can smoothly play the H. 264 video stream received from network, and the packet loss rate is low. The innovation of this paper is to use Directshow to play smoothly real time H. 264 stream received from network.

**Keywords:** H. 264; Directshow; MFC; network video monitoring

## 0 引言

视频监控以其直观方便信息内容丰富而广泛应用于安保、监控等场合, 成为商业、交通、住宅等领域防范的重要手段。随着数字视频压缩编码技术的成熟、多媒体计算机性能的提高和网络带宽的增大, 远程视频监控成为当今监控技术的主流<sup>[1]</sup>。目前视频监控系统中采用较多的图像压缩标准为: MPEG-4, H. 263, H. 264 等几个标准, 其中 H. 264 比现有的 H. 263, MPEG-2 具有明显的优势<sup>[2]</sup>。H. 264 作为视频压缩标准使用在安防行业, 以其高的数据压缩比, 对网络带宽的要求更低, 可节省大量带宽资源, 编码图像质量高, 可提供更高的传输质量, 在视频监控中的应用越来越广泛<sup>[3]</sup>。2009 年 7 月的 H. 264-10 标准支持多视频流解码, 即用户可以通过一个视频流采集到多个摄像机的监控情况, 因此开发 H. 264 网络视频监控客户端是安防监控中的重要部分。

## 1 系统组成与功能

网络视频监控系统由前端监控设备、视频服务器和

监控终端三大部分组成。<sup>[4]</sup> 基于 Directshow 的 H. 264 网络视频监控客户端系统组成如图 1 所示。网卡从网络接收数据后根据供 Directshow 的 SourceFilter 读取作为数据源, Directshow 完成对接收数据的处理, 包括存储、解码、显示等部分, 并且封装成相应的 API, 供 MFC 调用。MFC 完成了界面的设计, 以及把 Directshow 封装的 API 作为相应的消息响应函数。而本系统的重点及难点就在于利用 Directshow 完成网络数据的接收与播放。

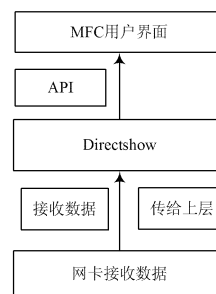


图 1 客户端系统框图

## 2 Directshow 开发

### 2.1 Directshow 开发简介

Directshow 是一个开放性的应用框架, 属于 Di-

rectX 家族一员,是一套基于 COM (Component Object Model) 的编程接口。Directshow 为 Windows 平台上处理各种格式媒体文件的回放、音视频采集等高性能要求的多媒体应用提供完整的解决方案,目前可支持多种数据格式有 ASF、MPG、AVI、MP3 和 WAV<sup>[5]</sup>,典型的 Directshow 系统如图 2 所示。

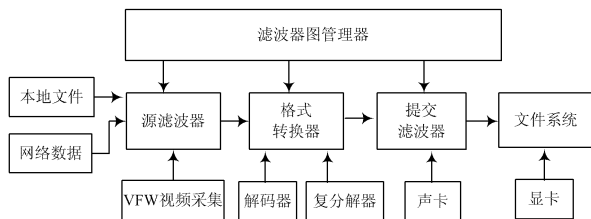


图 2 Directshow 系统框图

Directshow 基本原理是流水线,即将单元滤波器 Filter 串联起来建立起相应的 Filter Graph,然后通过 Filter Graph Manager 来控制整个数据处理过程并与应用程序交互。<sup>[6]</sup>按照功能来分,Filter 大致可分为 3 种类型:

- (1) Source Filters: 主要负责获取数据。数据源可以是本地文件,网络接收数据,采集卡及数字摄像机。
- (2) Transform Filters: 主要负责数据的格式转换。它是整个滤波器图(Filter Graph)的处理核心,完成数据分流/合成、解码/编码等。
- (3) Rendering Filters: 主要负责将处理后的数据流提交给外部设备。外部设备包括文件系统、显卡、声卡及网卡<sup>[7]</sup>。

Filter 的工作模式可分为推模式和拉模式。在推模式中,source filter 产生数据,并将其递交给下游的 filter,下游的 filter 被动地接收数据并处理它们,再将数据传递给它的下游 filter。其中数据传输的单位为 Sample。在拉模式下,source filter 与一个 pull filter 连接,pull filter 向 source filter 请求数据,source filter 回应请求并传递数据。推模式使用 IMemInputPin 接口,而拉模式使用 IAsyncReader 接口<sup>[8]</sup>。

两个滤波器相连的接口是 Pin,可以分为输入 Pin 和输出 Pin 且两个相连的 Pin 必须不同类型。每个 Pin 都是滤波器私有对象,滤波器可以动态地创建 Pin,销毁 Pin,自由地控制 Pin 的生存时间。两个滤波器的 Pin 相连时,需协商统一数据流的类型、缓存的大小、数据传送的机制等。

### 2.2 H. 264 网络视频客户端的 Directshow 实现

本系统采用推模式来实现,系统的客户端所用到的 Filter 以及数据的传送流程如图 3 所示,根据各个功能模块分为多个 Filter 进行开发,最后构建成一个 Filter-Graph 链路。

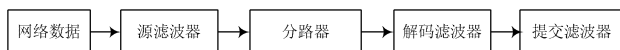


图 3 推模式下的 Filter 及数据传送流程图

Source Filters 从指定的端口号接收网络视频数据,每接收到一帧就把数据对 Samples 进行填充,并推给下一级的 Filter。Source Filter 的实现类为 CnetworkReciverFilter,由 CBaseFilter 继承而来。

CNetworkReciverFilter 主要是实现了 Filter 的构架。它由 GetPinCount, GetPin 等函数构成<sup>[9]</sup>。

Source Filter 的输出 Pin 的实现类为 CnetOutputPin,由 CBaseOutputPin 继承而来,主要是完成对 Pin 上媒体类型的设置以及 Pin 连接时媒体类型的检查,其实现函数是 GetMediaType 和 CheckMediaType,主要修改其中媒体的类型,以及 Samples 的大小和个数。

```

HRESULT CNetOutputPin:: GetMediaType(int iPosition,
CMediaType * pmt)
{
    /* ***** */
    pmt > InitMediaType ();
    pmt > SetType(& MEDIATYPE_Stream);
    pmt > SetSubtype(& MEDIASUBTYPE_H264);
}

```

设置了主媒体类型 Stream,次媒体类型为 H. 264。

```

HRESULT CNetOutputPin:: CheckMediaType(const CMedia
Type * pMediaType)
{
    if(pMediaType > majortype! = MEDIATYPE_Stream)
    {return E_INVALIDARG;
    }
    if ( pMediaType > subtype! = & MEDIASUBTYPE _
H264)
    {return E_INVALIDARG;
    }
}

```

用于 Pin 上媒体类型的检查,如果主媒体为 Stream,次媒体为 H. 264,则 Pin 可以连接成功。

CNetworkReciverFilter 为没有注册的 Filter,因此其创建过程如下:

```

CNetworkReciverFilter* REciverFilter= new CNetworkRe
civerFilter( NULL, & hr );
CNetOutputPin * NtoutputPin = new CNetOutputPin
(& hr, REciverFilter, L"264Stream");

```

复分解 Filter 和解码 Filter 的创建: 下载 Elecard 公司的 MPEG Push Demultiplexer 以及 ffdshow 的解码器。下载完成后,找到其中的 .ax 文件,用 regsvr32.exe+ 路径+ 文件名注册即可,可以在 Directshow Filter 库里面找到,如图 4 所示。Demultiplexer 主要是完成对音视频的分离以及对帧率的控制,使其适应解码器的要求。

Filter 的创建用 CoCreateInstance 函数。代码

如下:

```

HRESULT hr = CoCreateInstance(CLSID_FilterID,
NULL,
CLSCTX_INPROC,
IID_IGraphBuilder,
(void* *) & m_pGB);

```

而显示 Filter 采用的是微软提供的 VM9 VideoRenderer。

FilterGraph 的创建, BuildGraph 函数首先创建了一个 FilterGraph 组件, 然后获得一系列的控制接口, 接下来是把创建 SourceFilter 以及 MPEG Push Demultiplexer, 并将其添加进 FilterGraph 此时并不能直接创建 ffdshow 解码 Filter, 并且把 MPEG Push Demultiplexer 的输出 Pin 与其输入 Pin 相连, 因为此时 MPEG Push Demultiplexer 是没有输出 Pin 的。运行 FilterGraph, 使 SourceFilter 与 MPEG Push Demultiplexer 都运行在 RUN 状态, SourceFilter 接收到一定数据并且传递给 MPEG Push Demultiplexer, 让其分析所接受到的媒体格式, 停止运行 FilterGraph, 用 GetUnconnectedPin() 函数得到 MPEG Push Demultiplexer 的输出 Pin。然后就可以创建 ffdshow 解码 Filter 以及 VideoRenderer Filter, 并且把创建好的 Filter 添加到 FilterGraph 中去, 连接余下的 Filter, 然后运行 FilterGraph 即可。

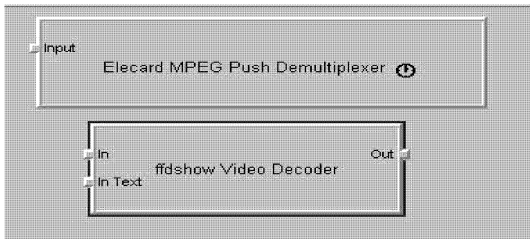


图4 下载的软件界面

### 3 界面设计

图5为用户界面菜单示意图, 利用MFC完成界面的设计, 其主要有准备, 播放及退出按钮。

准备: 是完成对IP及端口的设置, 以及完成对FilterGraph的构建。

播放: 对FilterGraph的控制, 使Graph工作在播

放模式, 每个Filter运行在RUN模式下, 完成对视频的显示。

退出: 断开FilterGraph, 释放各个接口资源。

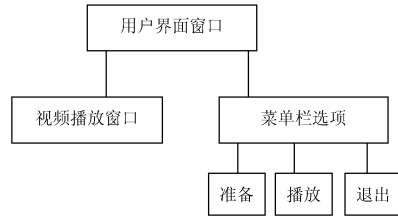


图5 用户界面菜单示意图

### 4 结语

该系统采用Directshow技术和MFC, 完成了视频监控客户端的设计。在Directshow设计时使用的复分解以及解码器都使用了第三方的Filter, 大大减少了Filter开发的工作量<sup>[10]</sup>。经过在PC机上的验证, 能够播放从网络接收到的图像。

#### 参 考 文 献

- [1] 邹纯宏. 基于Directshow和RTP的网络视频监视系统设计与实现[D]. 哈尔滨: 哈尔滨理工大学, 2007.
- [2] 李宾, 高平. H. 264编码系统的特点及其应用前景[J]. 电视技术, 2003(6): 19-21.
- [3] 刘玉娜. 基于H. 264/AVC视频编解码关键技术的研究[D]. 成都: 电子科技大学, 2009.
- [4] 童勇木, 张兴会. 网络视频监控系统的研究[J]. 天津工程师范学院学报, 2006, 16(2): 19-23.
- [5] 郭昊. 基于Directshow技术实现视频采集[J]. 火控雷达技术, 2008, 37(2): 97-100.
- [6] 邹纯宏, 苑惠娟, 赵婷婷. 基于RTP和Directshow的网络视频监控系统[J]. 研究与开发, 2007, 26(3): 43-45.
- [7] 马姗姗, 钱建生, 程德强. 基于H. 264的远程视频监控系统的设计与实现[J]. 电视技术, 2004(6): 77-79.
- [8] 陆其明. Directshow开发指南[M]. 北京: 清华大学出版社, 2003.
- [9] 陆其明. Directshow务实精选[M]. 北京: 清华大学出版社, 2003.
- [10] 肖振中, 梁晋. 基于Directshow的网络视频监控系统的设计与实现[J]. 计算机工程与应用, 2006, 12(3): 211-233.

作者简介: 彭 锋 男, 1984年出生, 湖北人, 硕士研究生。主要研究方向为无线通信系统。