

多媒体数据网络实时传输系统的设计与实现*

Design and Implement of A Network Real-time Transmission System of Multimedia Data

李欣欣,李为民,梁济仁

Li Xinxin, Li Weimin, Liang Jiren

(广西民族学院电子与通讯工程学院,广西南宁 530006)

(Coll. of Elec. & Communication Engi., Guangxi Univ. for Nationalities, Nanning, Guangxi, 530006, China)

摘要:介绍一个利用Windows SOCKET 编程和广播数据报方式,通过局域网络实现的多媒体数据实时传输系统的功能、网络通讯编程、网络结构、网络的应用程序,以及程序调试应注意的问题。该系统可以实现语音、图像、视频、鼠标坐标以及控制信号等的多媒体数据实时传输,能有效地改善网络教学效果。

关键词:多媒体 数据传输 端口 线程 广播 远程控制

中图分类号:TP393 文献标识码:A 文章编号:1002-7378(2005)04-0263-06

Abstract: A real-time transmission system of multimedia data based on network is introduced. The windows socket and UDP broadcasting are used in the system. The system functions, network communication program, network structure and the implement of applications and the related questions on debugging are explained. The system can transmit audio, image, video, mouse position and remote control information instantly. The teaching on LAN can be improved effectively by using this system.

Key words: multimedia, data transmission, port, thread, broadcast, remote control

目前国内已有多种功能丰富的网络化“电子教室”软件。将这些电子教室软件安装到微机室内的联网计算机上,确实能改善教学效果,使计算机设备和网络资源得到充分利用。这些电子教室软件都是基于网络的多媒体数据实时传输技术,主要涉及到图像传输、数据文件传输、语音和视频信号传输等方面的技术。本文介绍一个能在网络中实现类似电子教室软件功能的多媒体数据实时传输系统,并给出其基本结构,设计方法和实现步骤。

1 系统功能

1.1 系统的功能

本系统拟利用现有的计算机实验室的PC机组成对等网。并将其中的一台PC机作教师机,编制一

套在网络上传输图像和语音的软件,附加一些控制功能,使得老师在教师机上播放的课件图像及语音能在网络上传输到各个学生机的屏幕和喇叭上。学生在各自的计算机上既能看到教师机屏幕的内容,还可以听到教师的讲述。

系统功能主要包括三个部分:声音、鼠标和图像的传输,视频的采集与播放以及远程控制。教师端和学生端的计算机所实现的功能如图1所示。



图1 系统功能

1.2 教师端计算机的功能

教师端计算机的操作系统为WIN98/WIN2000,配置有声卡、麦克风和USB摄像头。教师端的程序名为Teacher.exe,该程序运行后,显示的用户界面如图2所示。之所以将界面做成扁平状,是

收稿日期:2005-03-07

修回日期:2005-05-20

作者简介:李欣欣(1975-),男,广西梧州人,讲师,主要从事多媒体技术和机械制造及自动控制等方面的教学和应用研究。

* 广西民族学院科研项目(编号02WZLX0015)资助。

为了尽量少占用屏幕,不影响教师原来的屏幕教学操作。而此程序窗口的标题栏,则兼作操作状态显示。

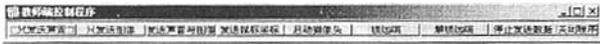


图 2 教师端用户界面

教师端的程序运行后,即工作在“同时发送声音与图像”的状态。此时,教师端计算机屏幕的图像和教师的讲话,均同时向网络广播。网上所有运行学生端接收程序的计算机,均可接收到同样的图像和声音。由于图像和声音在网上实时传输时,均会占用很大的带宽,因此系统还有“只发送声音”和“只发送图像”两个按键,供教师按实际情况选择。

当按下“发送鼠标坐标”的按键时,教师端只发送当前鼠标的坐标,而不发送图像信息。这适用于教师只是移动鼠标讲解课文,而不必刷新文字或图像时使用,这可改善接收端的声像效果。

“启动摄像头”按键是供教师用摄像头给学生显示实物或其它文字和图像材料时使用。

按下“锁定远端计算机”按键,会使学生端计算机的鼠标和键盘均不能操作。再按下“解锁远端计算机”键则会使学生端计算机的鼠标和键盘恢复操作。

1.3 学生端计算机的功能

学生端计算机操作系统为 WIN98/WIN2000,配置有声卡和音箱。学生端的程序名为 Student.exe。该程序运行后就启动接收功能,一旦收到教师端计算机送来的图像和声音信息,就会在自己的屏幕和音箱重放出来。程序只设有 2 项功能,分别为“停止接收”与“设置定时器”。其中的“设置定时器”功能用于设置网络故障发生时,学生端计算机自动解锁的时间间隔。

2 网络通讯编程模型和网络结构

本系统基于网络的多媒体数据实时传输,是利用网络编程接口 Windows SOCKET 完成通讯程序设计。利用 Windows SOCKET 来进行网络通讯,可以有面向连接和无连接通讯两种方式。面向连接通讯是通过 TCP 协议来完成的,它提供两台计算机间可靠无差错的数据传输功能。无连接通讯是通过 UDP(用户数据报协议)来完成的,它采用的数据报套接字由于不必建立连接,取消了重发校验机制,虽然不保证数据的可靠传递,但数据传输效率比连接协议高得多,能达到较高的通讯速率。它还提供向多个目标地址发送广播数据包的能力。本系统教师端与学生端计算机的网络通讯的程序,是用数据报套

接字的广播方式来实现的。系统的网络结构如图 3 所示,网络通讯程序的应用模型如图 4 所示。

系统工作时,教师机作为服务器端,只管发送数据,学生机作为客户端,只管接收数据。也就是说,数据是以单向广播方式传输的。

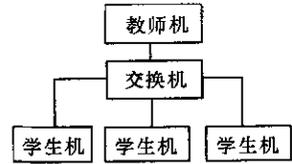


图 3 网络拓扑结构

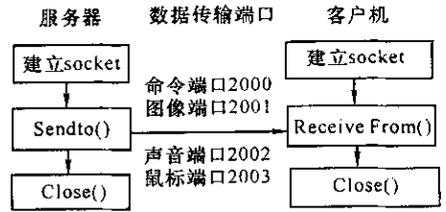


图 4 网络通讯模型

3 网络的应用程序

在教师和学生端程序中,都分别建立“声音、图像、鼠标、命令字”4 个 SOCKET,各自完成“声音、图像、鼠标、命令字”4 个发送和接收功能。其中鼠标 SOCKET 的端口为 2003,声音 SOCKET 的端口为 2002,图像 SOCKET 的端口为 2001,命令字 SOCKET 的端口为 2000。使用不同的 SOCKET 和端口来传输不同的信息,工作起来互不干扰,编程也方便。这是本程序的特点之一。下面分别介绍这两个应用程序的具体制作方法。

3.1 教师端应用程序

3.1.1 教师端应用程序的建立与初始化

在 VC++6.0 编程环境下建一个 Teacher 工程,步骤如下:

新建→工程→MFC 应用程序→project name = NetTeacher→单文档→step2、3 采用默认选项→step4 选择 Windows Sockets→step5 采用默认选项→step6 中类 CNetTeacherView 的基类选择 CFormView→完成。

工程建好后,调整界面大小,加入“只发送声音”,“只发送图像”,“同时发送声音与图像”,“发送鼠标坐标”,“启动摄像头”,“锁定远端计算机”,“解锁远端计算机”等几个功能按键。并加入对应的消息响应函数: OnSendAudOnly(), OnSendImgOnly(), OnSendAudImg(), OnSendMouseOnly(), OnButtonCamera(), OnLockRemote(),

OnUnLockRemote()。

以上各项功能的实现中,语音的传输为主线程,图像传输与鼠标坐标传输则使用不同的线程。图像传输与鼠标坐标传输两线程的函数分别为 UINTEngThreadProc (LPVOID pParam) 和 UINTEngMouseThreadProc (LPVOID pParam)。

教师端程序的初始化工作包括声音、图像、鼠标、命令字 4 个 SOCKET 的创建;另外,抓取屏幕图像的类 CDC 对象也要在这个步骤里进行初始化。初始化程序流程如图 5 所示。其中的 4 个 SOCKET 均采用广播数据报方式^[1],须使用参数 SOCK_DGRAM 和 SO_BROADCAST(它们的发送缓冲区的长度均用缺省值 8192 K)。



图5 教师端程序初始化

3.1.2 语音的录入与传输^[2]

语音录入与传输是教师端程序的主线程,它采用 Windows MDK 的低层音频服务。当应用程序不断向设备驱动程序提供音频数据时,设备驱动程序控制音频设备在后台完成录音和放音的具体操作。通过回调机制,我们可以检测到何时用完一个数据块,并及时传送下一个数据块,从而保证了声音的连续传输。程序运行时,单击图 2 中的“只发送声音”或“发送声音与图像”按钮,即可开始声音数据的发送。

实现语音传输的程序制作步骤如下:

步骤 1:用 waveInOpen(...)调用 CALLBACK_WINDOW 参数打开波形输入设备。但在使用声音之前,要把多媒体链接库 winmm.lib 包含在工程中。即在头文件 NetTeacherView.h 中加上如下语句: #pragma comment(lib, "winmm.lib")。

步骤 2:分别给音频数据块和音频数据缓冲区分配、锁定全局内存。

步骤 1 和步骤 2 在声音的初始化函数

AudDeviceInitial()中完成。声音的初始化函数使用 m_pwfx 为 WAV 文件定义各种参数:音频文件格式类型为 WAVE_FORMAT_PCM,采用单声道,每秒钟采样 8000 次,每个采样数据是 8 位等等。在给音频数据缓冲区分配全局内存时,为了保证音频数据的实时传输,要给它们分配 2 个数据缓冲区,以便当其中一个缓冲区满且正在传输时,另一个缓冲区可以继续接收话筒输入的音频数据。

步骤 3:初始化音频数据块结构各成员变量,主要是将每个缓冲区指针赋给对应数据块结构中的缓冲区指针变量 lpData,调用 waveInPrepareHeader(...)和 waveInAddBuffer(...)为输入设备打开音频数据缓冲区。

步骤 4:调用 waveInStart()函数准备录音。

步骤 3 和步骤 4 在声音的处理函数 AudProcess()中完成,它分别定义两个单声道音频数据缓冲区的数据头,以及每个音频数据缓冲区的大小;接着打开一个音频数据缓冲区,并使用定义好的参数开始录音。

步骤 5:为语音输入建立消息响应函数,在 CNetTeacherView 类中添加 MM_WIM_DATA 消息的响应函数 OnAudioInMsg()。其作用是当有语音输入时,产生 MM_WIM_DATA 消息,并使用响应函数 OnAudioInMsg()进行音频数据录音;在通过语音 SOCKET 端口 2002 发送该音频数据的同时,打开另一个音频数据缓冲区,继续进行录音。

3.1.3 图像的传输^[3]

教师端程序运行时,当用户按下“只发送图像”按钮后,即转至执行 OnSendImgOnly()函数,启动图像传输线程进行图像数据的发送。这时只发送图像数据,不发送声音和其它信息。由于图像传输数据量较大,占用较多系统资源,所以另外启动一个线程,这样可以比较好地兼顾系统性能问题。图像采集和传输流程如图 6 所示。

开始图像的采集和传输时,程序首先进行屏幕拷贝,也就是将屏幕上的内容以 BMP 的位图格式拷贝下来,然后转换为 JPEG 格式,最后通过网络将 JPEG 文件传输至学生端。该工作流程的主要功能为:(1)初始化图形设备及位图对象,调用 CDC 对象的成员函数 CDC::BitBlt(...)拷贝桌面屏幕内容。(2)将 BMP 文件转换为 JPEG 格式的文件并发送图像。

屏幕图像拷贝时,并没有连鼠标一起拷贝。为了同时传输鼠标信息,在发送一帧图像数据前,须将鼠

标位置所对应的像点变为蓝色小块。这里的鼠标以纯蓝色像点的形式发送,与“只发送鼠标坐标”情况不同。

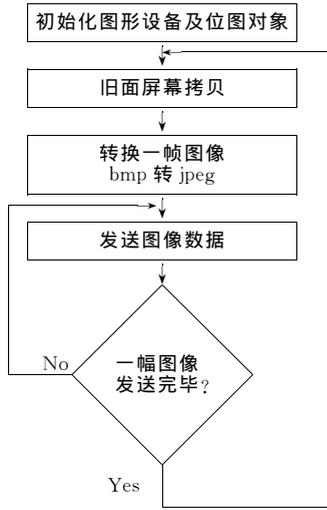


图6 图像的采集和传输流程

3.1.4 视频采集与播放^[4]

视频捕获的工作流程如图7所示。实现视频捕获功能的软件设计主要使用了VFW(Video for Windows)函数的7条命令:

```
capCreateCaptureWindow(...); capDriverConnect(...); capDriverGetCaps(...); capPreviewRate(...); capPreview(...); capCaptureAbort(...); capDriverDisconnect(...).
```

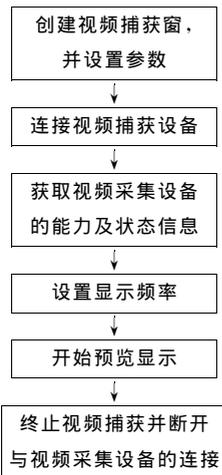


图7 视频捕获流程

具体实现此程序功能的步骤和源码如下:

(1)给工程文件增加VFW函数库:在Project→Setting→Links→Object/library modules:加入vfw32.lib。

(2)在应用程序窗口中增加一个“启动摄像头”按钮;在VC的资源编辑窗的主程序对话框适当位置增加一个Button控件,令其ID=IDD_

BUTTONCAMERA,caption=启动摄像头。

(3)增加一个视频采集对话框,用来显示视频图像和相关命令菜单:1)在VC的资源编辑窗增加一个Dialog,令其ID=IDD_CAMERA,并删除原来Dialog内的OK和CANCEL 2个按钮。2)在VC的资源编辑窗增加一个Menu,令其ID=IDR_CAMERA,并手工加入“视频格式”、“视频源”、“冻结图像”、“解冻图像”、“退出视频采集”5个菜单。3)在视频采集对话框属性Menu的ID项填入IDR_CAMERA。这样处理后,视频采集对话框就会有上述5个菜单出现。

(4)为视频采集对话框IDD_CAMERA增加一个新类:在Class Wizard中选中Add Class,在Class Information对话框中,令Name=Ccamera,File name=camera.cpp,Base class=CDialog,Dialog ID=IDD_BUTTONCAMERA。这时候,工程文件中就增加了camera.cpp和camera.h两个文件。这样视频采集的有关源码就可加入到camera.cpp中。

(5)编辑NetTeacherView.cpp,使得用户点击“摄像”按钮时,能弹出视频采集对话框。

(6)使用前面所说的7条AVICAP函数命令,给“视频格式”、“视频源”、“冻结图像”、“解冻图像”、“退出视频采集”5个菜单增加相应处理代码,使其实现视频图像的采集和显示功能。

至此,实现摄像功能的程序就做好了。运行程序,点击“启动摄像头”按钮,就会弹出视频采集框,并实时显示摄像头所采集到的视频图像。点击“视频格式”菜单,可以对视频数据流的分辨率和象素深度(位)等进行设置。点击“视频源”菜单,可以调整亮度、对比度和色彩饱和度等参数;点击“冻结图像”或“解冻图像”菜单,可以冻结图像或解冻图像。这些视频图像都能通过网络在学生端计算机显示出来。

3.1.5 鼠标坐标的传输^[4]

如果老师在某一段时间内只使用同一幅画面讲解课文内容,那么只发送鼠标坐标就足够了,而不必将屏幕图像全部发送出去。这时,从教师端计算机向学生端计算机的图像数据传输应停止发送,而只发送鼠标坐标。实际使用时,单击图2中的“发送鼠标坐标”按钮,即可开始鼠标数据的发送传输。

3.1.6 远程控制^[4]

为了不让学生端的计算机在上课时被用于做一些无关的事情,通常需要对学生端计算机进行限制以及远程控制。在教师端单击图2中的“锁远端”按钮,即可锁住所有学生端机器的键盘与鼠标。它通过

命令字 SOCKET 的 2000 端口发送命令字“6666” (“解锁”则发送命令字“7777”),当学生端的计算机从对应端口收到此命令字后,就执行锁(或解锁)键盘与鼠标的操作。

3.2 学生端应用程序

学生端应用程序主要用于对教师端数据的接收与响应。它也和教师端一样有声音、图像、鼠标、命令字四个 SOCKET,各个 SOCKET 的名称和收发的端口号与教师端一一对应。这样就在教师端和学生端之间建立起 4 条通讯链路,可以同时传输 4 路信息。这样设计程序的好处是:接收程序不必再用专门的语句来判断接收到的是命令字还是图像,或者是语音数据,而是由不同的 SOCKET 端口自动响应和接收处理不同的数据。例如,当监测到图像 SOCKET 的端口 2001 有数据送来时,就会转去执行图像数据的接收处理函数。从而使得编程和数据处理都变得十分方便。

3.2.1 学生端应用程序的建立与初始化

在 VC++6.0 编程环境下,学生端应用程序的建立步骤如下:新建→工程→MFC 应用程序(.exe)→NetStudent→单文档→step2、3 采用默认选项→step4 选择 Windows Sockets→step5、6 采用默认选项→完成。

和教师端一样,学生端程序的初始化工作包括声音、图像、鼠标、命令字 4 个 SOCKET 端口的创建,以及图像类 CDC 对象的初始化(其中图像与声音缓冲区的长度为 8192,命令与鼠标缓冲区的长度为 10)。为了使学生端程序一启动便处于接收状态,需要在函数 CNetStudentView::OnInitialUpdate() 中加入初始化各 SOCKET 端口接收功能的代码。学生端程序的初始化工作编程方法与教师端类似,不同之处是建立各个数据报 SOCKET 后,均加入一个 AsyncSelect(FD_READ)函数,以允许各个对象响应 FD_READ 事件,等待对应端口的数据到达。当数据到达后,则跳转至响应的 OnReceive() 函数,接收和处理各个数据。

3.2.2 语音的接收

语音接收的实现与语音的录入发送过程类似,仅是数据的流向相反。学生端计算机语音接收程序的工作流程如下:1)用 waveInOpen(...)调用 CALLBACK—WINDOW 参数打开波形输入设备。2)分别给音频数据块和音频数据缓冲区分配、锁定全局内存。这两步的代码与教师端相同,在此不再赘述。3)初始化音频数据块结构各成员变量,主要是将

每个缓冲区指针赋给对应数据块结构中的缓冲区指针变量 lpData,调用 waveOutPrepareHeader(...)使音频数据头的定义生效。4)从语音 SOCKET 的 2001 端口接收语音数据,并调用 waveOutWrite() 函数放音。

3.2.3 屏幕图像的接收

为了满足 VC 中默认的图像设备—BMP 格式图像设备的要求,学生端计算机在收到一帧 JPEG 图像数据后,必须转换为 BMP 格式,这是从 BMP 格式到 JPEG 格式转换的逆过程。只有这样,教师端的图像才能在不添加任何 VC 插件的情况下重现于学生端屏幕。图像数据的接收与转换过程如下:1)监听教师端的图像数据是否到达;2)从图像 SOCKET 端口 2001 接收图像数据;3)将收到的 JPEG 格式图像数据转换为 BMP 格式数据,使用 VC 默认的图形设备显示 BMP 图像。

3.2.4 鼠标坐标的接收

学生端计算机收到教师端发来的鼠标坐标数据后,通过调用函数 CWnd::SetCursorPos(),就可在屏幕的相应位置显示鼠标。

3.2.5 加锁—解锁键盘与隐藏学生端程序名

当学生端的接收程序从命令 SOCKET 的 2000 端口收到教师端发来的命令字后,就会由 OnReceiveCMD() 函数作出响应。在该函数中,通过判断所收到的命令字是“6”或“7”(尽管发送方发来的是多个“6”或“7”,这里只检测一个就够了),来实行加锁或解锁键盘与鼠标的操作。

当学生端计算机被远程控制锁住了键盘与鼠标后,学生端的 Ctrl+Alt+Del 组合键仍会起作用。如果此时在学生端按 Ctrl+Alt+Del 组合键进入任务管理器并停止接收程序的运行,那么远程控制就失去作用了。针对此问题,还必须让学生端接收程序名 student.exe 在系统的任务管理器中隐藏,使得学生不能随意在任务管理器中停止该接收程序的运行。

通常的做法是使用 API 函数 SystemParametersInfo (SPI_SETSCREENSAVER, 1, NULL, 0) 来实现这个功能。但是,这个函数功能在 WINDOWS 9x 中可以实现,在 WINDOWS 2000 中却得不到正确的结果。因此,我们采用另外一种方法,即使用函数 CWnd::SetWindowText(LPCTSTR) 来实现在任务管理器中隐藏学生端接收程序名这个功能。只要将其中的参数设置为空(即 SetWindowText(""));,则会使程序名从系统的任务管理器中消失,也就使学生端程序

不会被轻易中止或退出。

当学生端计算机的键盘与鼠标被锁住后,学生端的计算机不能响应键盘与鼠标消息,但仍可正常接收教师端计算机发来的图像和语音信息。而这时一旦发生网络故障,学生端的计算机就会既收不到图像和语音信息,也无法操作计算机。为了应对这个问题,学生端接收程序在锁键盘与鼠标的同时,还要不断监测网络是否通畅(通过定时中断,检测是否收到网络数据来判断网络通断)。如果在规定时间内(如300s)内收不到网络数据,就执行解锁操作,并弹出显示“网络不通”的消息框通知用户。从而避免计算机因收不到数据或解锁命令而处于“死锁”状态。

4 程序调试时应注意的问题

本系统在Windows 2000操作系统下工作正常,但尚不能兼容Windows XP环境。在调试和使用前,必须将所有计算机的显示属性作相同的配置(本文介绍的程序是针对学生端和教师端计算机的显示器属性均已设为1024×768像素,16位色彩的条件设计的),否则,接收端显示的图像颜色和形状会与发送方不同。其次,为了使教师和学生端的屏幕上的鼠标位置相对应,要在学生端调用函数ClientToScreen()把教师端屏幕坐标转换为学生端的客户区坐标。

调试时,图像与鼠标坐标数据的传输发送并不是越快越好,因为广播数据报通讯采用的是UDP协议,它没有流量控制功能。当接收端的处理速度跟不上发送速度时,就会产生丢包现象。因此,在发送端用适当的软件延时作为简单的流量控制,可以改善系统的工作效果。在发送每段图像数据时,加上“Sleep(ms)”作短暂延时,可减少学生端的屏幕闪烁。教师端的鼠标坐标数据如果连续发送过快时,学生端计算机反应速度跟不上,会出现明显的鼠标停滞现象。因此,也需要在两段数据发送之间添加一短暂延时。

参考文献:

- [1] 吴礼发. 网络程序设计教程[M]. 北京: 希望电子出版社, 2002.
- [2] 陈坚. 实例解释VISUAL C++.NET编程[M]. 北京: 希望电子出版社, 2002.
- [3] 梁晨宝. 用VC++制作实时教学工程[EB/OL]. http://www.cew.com.cn/htm/app/aprog/01_11_12_3.asp, 2001-11-12.
- [4] 张友生. 远程控制编程技术[M]. 北京: 电子工业出版社, 2002.

(责任编辑: 邓大玉 韦廷宗)

(上接第262页)

行分析和处理,以及复原网页记录等功能。该系统的实现采用了监听与识别数据包、分析数据包、减小丢包率等关键技术。该系统还需要进一步完成的工作是:(1)应用并行处理技术,使接收过程、处理过程和分析数据包的各功能之间能够并行处理;(2)增强还原效果,使能还原的网页内容尽可能多,尤其要实现图片的还原。

参考文献:

- [1] Warren G Kruse. 计算机取证: 应急响应精要[M]. 北

京: 人民邮电出版社, 2003.

- [2] 陈千, 马剑锋, 焦政, 等. Sniffer 技术在网络管理中的应用和研究[J]. 计算机工程与设计, 2004, 25(4): 536-539.
- [3] 李响, 李伟华. 面向入侵检测的模式匹配算法研究[J]. 计算机工程与应用, 2003, 39(6): 4-6.

(责任编辑: 邓大玉)