

# 基于 WinSock 通信的屏幕监控系统<sup>①</sup>

高照恒, 陈家琪

(上海理工大学 光电信息与计算机工程学院, 上海 200090)

**摘要:** 详细介绍了屏幕监控软件的设计思路 and 开发技术。使用 WinSock API 接口和 MFC 网络编程技术, 系统采用 C/S 模式, 基于 UDP 协议进行通信, 客户端(Client)通过 Winsock 发送抓取监控屏幕的指令, 服务器端(Server)接收服务器端传来的指令抓取屏幕, 并将其转化为 JPEG 数据流, 向服务器端传输图像数据流。实验结果表明, 该方法能高质量实时的对计算机屏幕进行监控。

**关键词:** WinSock; 数据流; UDP; 冗余; 霍夫曼编码

## Screen Monitoring System Based on WinSock Communication

GAO Zhao-Heng, CHEN Jia-Qi

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200090, China)

**Abstract:** This paper describes the design of the screen monitoring software ideas and development techniques. With the WinSock API interface and MFC Internet programming technology, the system uses C/S model, based on UDP protocol for communication. The client side (Client) sends commands to crawl the screen Control through the Winsock. The server side (Server) receive capture screen instructions from server-side, and translate them into JPEG data streams, the server-side image data stream. Experimental results show that the method can easily realize the computer screen monitoring.

**Keywords:** WinSock; data flow; UDP; redundancy; huffman encoding

屏幕监控是指通过在本机上安装的监控软件能够在线浏览和操作在远方的安装有监控服务器软件的计算机屏幕, 以实现监控的目的。目前, 屏幕监控程序一般分两个部分: 一部分是客户端程序 Client, 另一部分是服务器端程序 Server。在使用前需要将客户端程序安装到被控端计算机上, 将服务器端程序安装到主控端计算机上。客户端主要是监听、执行服务器端发来的请求或指示, 如键盘按键、鼠标的操作等, 同时还要定期发送屏幕位图信息到服务器端。而服务器端则负责把接收到的屏幕位图信息显示在服务器端的窗体上并相应的把客户端的键盘、鼠标动作发送给客户端应用程序。为了获得流畅的远程屏幕图像, 这样就提出了 2 个问题: 如何实现通过网络与远端机的数据

进行交换; 如何有效地压缩和传输远端桌面屏幕。

### 1 WinSock 通信技术

套接字是支持 TCP/IP 协议的网络通信的基本操作单元, 是网络通信的基石; 它是不同主机间进行双向通信的端点, 构成了单个主机及整个网络间的编程界面。要通过网络进行通信, 你至少需要一对套接字<sup>[1]</sup>。其中一个运行于客户端, 另一个运行于服务器端, 根据连接启动的方式以及本地套接字要连接的目标, 套接字之间的连接过程可以分为三个步骤: 服务器监听, 客户端请求, 连接确认。

(1) 服务器监听, 是服务器端套接字并不定位具体的客户端套接字, 而是处于等待连接的状态, 实时监

<sup>①</sup> 收稿时间:2010-07-10;收到修改稿时间:2010-08-18

控网络状态。

(2) 客户端请求,是指由客户端的套接字提出连接请求,要连接的目标是服务器端的套接字。为此,客户端的套接字必须首先描述它要连接的服务器的套接字,指出服务器端套接字的地址和端口号,然后就向服务器端套接字提出连接请求。

(3) 连接确认,是指当服务器端套接字监听到或者说接收到客户端套接字的连接请求,它就响应客户端套接字的请求,建立一个新的线程,把服务器端套接字的描述发给客户端,一旦客户端确认了此描述,连接就建立好了。而服务器端套接字继续处于监听状态,继续接收其他客户端套接字的连接请求<sup>[2]</sup>。

套接字可以根据通信的类型分类,不同的套接字之间也可以互相通信;一般应用于程序的套接字可分为两类:

(1) 流套接字:建立在 TCP/IP 传输层上,提供双向的无记录边界的数据流服务,其传输数据有序、无重复,适用于处理大量数据;面向连接,通信双方进行数据交换之前,必须建立一条路径来确定双方通信的路由,保证双方都是活动的、可彼此响应的。

(2) 数据报套接字:数据报套接字是无连接的,直接建立在 IP 层上,可支持双向数据流动,但不能保证数据传输的可靠性、有序性及无重复性。

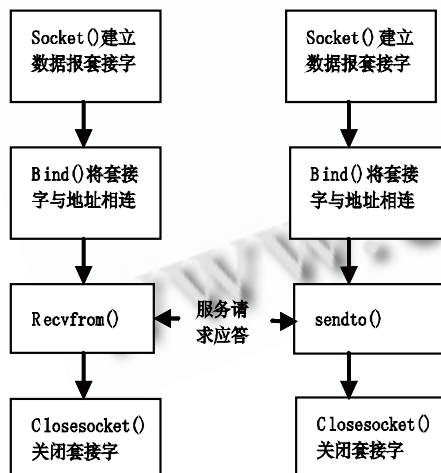


图 1 无连接套接字应用程序时序图

## 2 监控系统结构

应用程序体系结构是指应用程序内部各组件间的组织方式,目前比较流行的大体上分为两种:客户机/服务器(Client/Server,简称 C/S)结构和浏览器/服务器

(Browser/Server,简称 B/S)结构<sup>[3]</sup>。C/S 结构是将数据库内容放在远程的服务器上,在客户机上安装相应软件,是一种两层或三层结构的系统。与 B/S 结构相比,C/S 结构具有以下优势:

(1) 能够实现许多功能强大的监控功能,直接对监控端操作,监控效率高,占用带宽小。

(2) 所有数据都在服务器端进行管理,保证了数据的完整性、可靠性、安全性和易维护性。

(3) 对于相同的任务,C/S 模式速度大于 B/S 模式,大大减轻了网络的负担,数据通信量小。

屏幕监控系统是一个客户机/服务器系统,程序分服务器端和客户端两部分。其总体结构如图 2 所示。

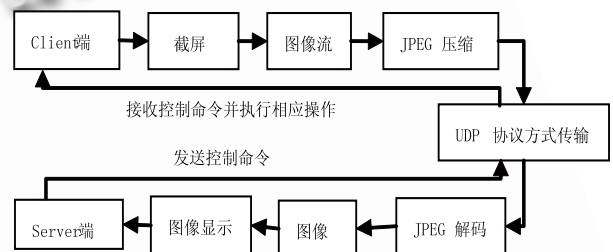


图 2 屏幕监控系统图

客户机端的工作流程:

(1) 将屏幕信息转换为位图信息,并将其压缩成 JPEG 数据流并发送该数据流。

(2) 通过 UDP 协议将位图信息发送到服务器

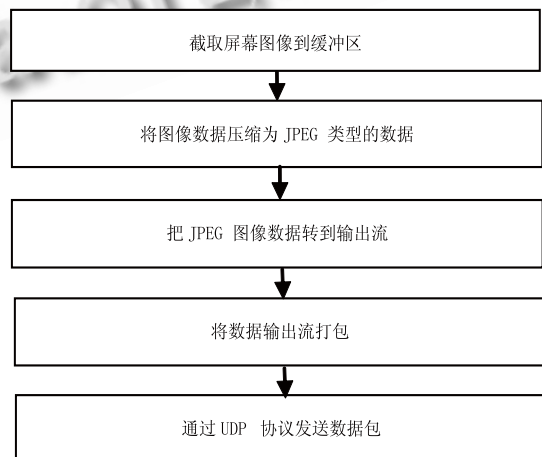


图 3 客户机工作流程图

服务器端工作流程:

(1) 向被控端发送鼠标和键盘操作指令。

(2) 接收被控端传来数据包，组合成数据流图像。

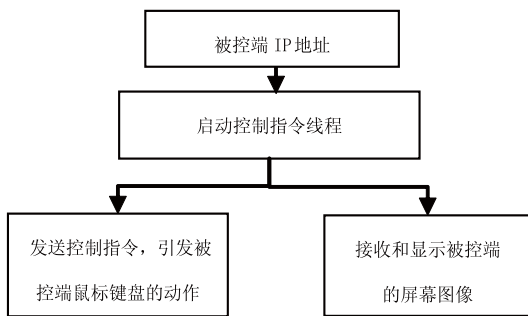


图 4 服务器工作流程图

### 3 客户端屏幕截图

远程屏幕截图就是在被控制端截取当前屏幕，形成一个位图文件，然后把该文件发送到控制端计算机并显示出来。这里主要需解决好屏幕抓取的大小定位、数据存储和传输等问题。

通常获取屏幕图像的方法有以下两种:创建屏幕设备描述表 Device Context (DC)方法和 DirectX (DX)方法。这两种屏幕捕捉方法各有优劣，屏幕 DC 方法适用范围广，能够捕捉大部分的屏幕图像，但捕捉速度较慢。DX 方法捕捉速度快，但要求机器必须安装 DirectX，适用范围较窄。在 CPU 为 Athlon XP1700+，使用显卡 Geforce2 GTS 并安装 DirectX8.1 的测试环境中，对 DC 和 DirectX 两种方法分别进行了 3 次捕捉试验，每次连续捕捉 100 帧，每种方法在不同分辨率下的每帧平均捕捉时间如表 1 所示<sup>[4]</sup>。

表 1 屏幕图像采集速度比较

屏幕尺寸	DC/ms	DX/ms
1024*768	450	122
800*600	270	74

为克服 DC 和 DX 方法较慢及适用范围较窄的不足，采用直接屏幕拷贝(Direct Copy Screen, DCS)方法。图像采集分为屏幕拷贝和色彩过滤两部分。在屏幕拷贝后，根据控制端的需要对屏幕位图进行色彩过滤，完成色位转换，过滤大量的颜色数据，减少数据量。

### 4 屏幕图像的压缩

对于屏幕分辨率为 1024\*1280 像素的系统来说，屏幕成真彩色位图时其位图大小为 3.75MB。如果不对

位图数据进行压缩，则需要分为许多数据包来发送，这样会造成服务器端的延迟，需要将位图数据压缩成 JPEG 数据流，对屏幕图像进行实时的压缩和传输。数据压缩的可能性在于数据本身存在的冗余<sup>[5]</sup>。对于图像而言，图像数字化之后，形成了大量的二进位数据。然而，由于图像的冗余性，表示一幅图像并不需要那么多二进位。一般来讲，数字图像存在以下类型的冗余：

(1) 空间冗余：一幅图像中相邻像素的灰度值(彩色值)基本上是一致的(除去图像边缘像素点)，图像相邻像素的数值是相关的。

(2) 时间冗余：在视频图像中，不同的图像帧是相关的。

(3) 频域冗余：在 RGB 彩色图像中不同色平面或频段是相关的。

(4) 信息熵冗余：也称为编码冗余。对于实际图像数据的每一个像素，很难得到它的信息熵，因此在数字化一幅图像时表示像素的比特通常存在冗余。

(5) 视觉冗余：人类的视觉系统对于图像的注意是非均匀和非线性的，特别是人类的视觉系统并不是对于图像中的任何变化都能感知。由于人是各类图像处理系统的最终接受者和评价者，而人眼的视觉特点产生了相应的视觉心理冗余。

数字图像压缩的目的在于去除图像的冗余和不必要的信息，在不降低图像视觉质量的情况下，减少图像的数据量和存储空间，提高图像的传输速率。通过对图像信源进行编码，使编码数据流小于原始数据流，达到压缩的目的。

图像数据压缩技术总的来说就是利用图像数据固有的冗余性和相干性，将一个大的数据文件转换成较小的同性质的文件。屏幕信息保存为位图数据里流，给传输带来了很大不便，图像压缩得到了广泛的运用。压缩的目的就是满足存储容量和传输带宽的要求，而付出的代价是大量的计算。

大多数图像常含有单色大面积图块，而且某些颜色比其它颜色出现更频繁。为了节省空间，在对数据进行编码时，就有可能对那些经常出现的数据指定较少的位数表示，而那些不常出现的数据指定较多的位数表示。这样从总的效果看还是节省了存储空间。用这种方法得到的代码，其码的位数，也即码长就是不固定的，故称为变长码<sup>[6]</sup>。

霍夫曼编码(Huffman Encoding)是一种概率匹配方法进行信源编码的熵编码方法,通过利用已变信号的统计特性,给其分配高效代码来实现数据压缩,用于去除图像数据的统计冗余。霍夫曼编码通过用不固定长度的编码代替原始数据来实现的,平均码率可以接近信息源熵值的一种编码方法。

霍夫曼编码最初是为了对文本文件进行压缩而建立的,它的基本思路是图像数据中出现频率越高的值,其对应的编码长度越短,反之出现频率越低的值,其对应的编码长度越长。霍夫曼编码的基本方法是先对图像数据扫描一遍,计算出各种像素出现的概率,按概率的大小指定不同长度的唯一码字,由此得到一张该图像的霍夫曼码表。编码后的图像数据记录的是每个像素的码字,而码字与实际像素值的对应关系记录在码表中,当然码表是附在图像文件中的。

在计算 Huffman 表时需要将原始图像数据扫描两遍:第一遍要精确地统计原始图中每个灰度出现的概率;第二遍是建立哈夫曼并进行编码<sup>[7]</sup>。霍夫曼编码将各数据出现的概率按从大到小的顺序排列,然后再按照下列规律列成~个二叉树结构,便可直接写成霍夫曼代码,具体步骤如下:

(1) 将两个最小概率的自由结点组成二叉树枝,树枝的权为其概率值;

(2) 建立二叉树枝的父结点,父结点的权为两树枝权的代数和;

(3) 将父结点和余下的最小权自由结点组成新的二叉树枝和结点。

由于要建立二叉树并遍历二叉树生成编码,因此数据压缩和还原速度较慢。但是该编码方法简单有效,而且编码效率相当高,因而得到了广泛应用。Huffman 编码码字平均长度接近信息符号的熵值,其最高压缩效率可达到 8:1

## 5 图像数据流的传输

传统的屏幕图像传输方法是定时截取整幅图像后压缩传输,这会消耗大量网络带宽,尤其在窄带环境下,该方法无法实现流畅的屏幕图像传输。实际上在电脑使用的大多数时间中,任一微小时间内屏幕图像发生的改变并不大,如果只对发生改变的图像进行传输,将能大大减少屏幕数据传输的流量<sup>[8]</sup>。因此,对屏幕图像采用分块传输的方法,即将屏幕图像划分为

4×4 共 16 块,定时同前一刻比较判断每块图像是否发生改变,仅仅对发生改变的屏幕图像块进行压缩和传输。由于只需要传输发生改变的图像数据,在进行远程控制或者监控时,控制端所显示的远端屏幕就能够得到较好的刷新速率,保持图像传输的稳定和流畅。

为了将本地数据传输到服务器,首先构造 CSocket 套接字对象,使用 Creat()函数创建数据格式的套接字,再通过 Bind()函数将该套接字与准备发送数据的接口绑定在一起,套接字创建成功后,可以使用 Sendto()函数方法发送数据到服务器。当客户机端传来数据时,服务器通过 recvfrom()接收图像数据,并且在接受完完整的图像数据后需要重新拿组合成一个完整的 JPEG 数据流,将其显示在窗口中<sup>[9]</sup>。

## 6 结束语

本系统是采用 VC++设计的单线程屏幕监控系统,可以实现很多远程控制功能,如获取目标电脑屏幕图像、窗口的名称,如果该系统能实现多线程,功能将进一步完善,图像传输效率和质量将进一步提高。经测试系统运行良好,有很好的推广和应用价值。

## 参考文献

- 1 Richter J.王建华,张焕生,后丽坤译.Windows 核心编程.北京:机械工业出版社,2004.286—302.
- 2 张千里.网络安全实用丛书——网络安全基础与应用.北京:人民邮电出版社,2007.15—77.
- 3 周兵,李伟姜,等.基于 B/S 模式的组件化网络多媒体监控系统结构研究.计算机科学,2006,33(1):100—104.
- 4 Stevens WR.范嬗华,胥光浑,张涛译.TCP/IP 协议详解(卷 1).北京:机械工业出版社,2003.
- 5 全子一.图像信源压缩编码及信道传输理论与技术.北京:北京工业出版社,2006.
- 6 吴功宜,董大凡,王珺,刘乾.计算机网络高级软件编程技术.北京:清华大学出版社,2008.
- 7 Richter J.王建华,张焕生,后丽坤译.Windows 核心编程.北京:机械工业出版社,2004.286—302.
- 8 Johnson AM. Windows System Programming.天津:机械工业出版社,2004.
- 9 Stroustrup B. The C++ Programming Language. Third Edition. New Jersey:Murray Hill, June 1997.
- 10 Schneier B.Managed Security Monitoring: Network Security for the 21st Century. Computer&Security, 2001,20(6):491—503.