Design an advanced video/voice conferencing system by using API and TAPI telephony to transfer video/voice throw networks and voice throw analog telephone line.

Omar A. Mohammad*

Abstract

The Main idea of Voice over Internet Protocol (IP) is to transfer the sound to a group of Bits pool in the matrix of Byte Array and then to transport the encapsulation in Datagram packets or in Stream Data over the network in order to receive the audio by the other party through a collection of Packets again in the matrix of Byte Array. The reading process is done in accordance with the principle of FIFO (First in First out).

In this paper, a conference system has been designed to send and receive video and utilize the services API and TAPI telephony in order to transfer voice and video from one computer to another or to transfer voice from computer to landline Analog Phone and take advantage of TAPI telephony through the program contained in the operating system TAPI v3.0 in the operating system Windows XP. The proposed system has been designed by using Microsoft Visual Basic, Net programming. The designed system has been tested between two networks for sending and receiving video and sending voice between computer and phone. The results have displayed high quality of video using API and TAPI Technology.

Keywords: (IP) Internet Protocol,(FIFO) First in First out,(API) Application Programming Interface ,(TAPI) Telephony Application Programming Interface

تصميم نظام المؤتمرات المتقدم للفيديو والصوت وباستخدام API و TAPI telephony لإرسال الفيديو والصوت عبر الشبكات أو الصوت من خلال خط الهاتف.

الخلاصة

تتلخص الفكرة الأساسية من نقل الصوت عبر بروتوكول الانترنيت (IP) بتحويل الصوت الى مجموعة من الـ Bits تجمع في مصفوفة Byte Array ثم كبسلته ليتم نقله بشكل Datagram packets أو بشكل Stream Data عبر الشبكة ولاستلام الصوت بالطرف الأخر يتم تجميع الـ Packets مرة أخرى في مصفوفة Byte Array وتتم عملية القراءة وفق مبدأ FIFO (أي القادم أولا يعرض أولا).

في هذا البحث تم تصميم نظام مؤتمرات لإرسال واستقبال الفيديو بالاستفادة من خدمات API و TAPI telephony وذلك لنقل الصوت والصورة من جهاز كمبيوتر الى جهاز كمبيوتر أخر أو لنقل الصوت من جهاز الكمبيوتر الى الهاتف الأرضي Analog Phone والاستفادة من TAPI telephony من خلال البرنامج الموجود مع نظام التشغيل TAPI v3.0 والمتاح مع نظام التشغيل Windows XP. إن النظام المقترح بني بأستعمال لغة Visual Basic.Net. وإن النظام المصمم اختبر بين شبكتين بإرسال واستقبال الفيديو و إرسال الصوت من الحاسبة الى الهاتف، وأظهرت النتائج جودة عالية للفيديو باستخدام تقنية API مع تقنية TAPI.

* College of Education, The Islamic University, Baghdad-IRAQ.

Introduction

Video conferencing uses telecommunications of audio and video to bring people at different sites together for a meeting. This can be as simple as a conversation between two people in private offices (point-to-point) or involve several sites (multi-point) with more than one person in large rooms at different sites. Besides the audio and visual transmission of meeting activities, videoconferencing can be used to share documents, computer-displayed information, and whiteboards. Simple analog videoconferences could be established as early as the invention of the television. Such videoconferencing systems usually consist of two closed-circuit television systems connected via coax cable or radio. During the first manned space flights, NASA used two radiofrequency (UHF or VHF) links, one in each direction. TV channels routinely use this kind of videoconferencing when reporting from distant locations, for instance. Then mobile links to satellites using specially equipped trucks became rather common. [1]

This technique was very expensive, though, and could not be used for applications such as telemedicine, distance education, and business meetings. Attempts at using normal telephony networks to transmit slow-scan video, such as the first systems developed by AT&T, failed mostly due to the poor picture quality and lack of efficient video compression techniques. The greater 1 MHz bandwidth and 6 Mbit/s bit rate of Picture phone in the 1970s also did not cause the service to prosper.

It was only in the 1980s that digital telephony transmission networks became possible, such as ISDN, assuring a minimum bit rate (usually 128 kilobits/s) for compressed video and audio transmission. During that time, there was also research into other forms of digital video and audio communication. Many of these technologies, such as the Media space, are not as widely used today as videoconferencing but are still an important area of research.[3][4] The first dedicated systems started to appear in the market as ISDN networks were expanding throughout the world. One of the first commercial Videoconferencing systems sold to companies came from PictureTel Corp. which had an Initial Public Offering in November, Videoconferencing systems throughout the 1990s rapidly evolved from very expensive proprietary equipment, software and network requirements to standards based technology that was readily available to the general public at a reasonable cost. Finally, in the 1990s, IP (Internet Protocol) based videoconferencing became possible and more efficient video compression technologies were developed, permitting desktop, or personal computer (PC)-based videoconferencing. In the 2000s, video telephony was popularized via free Internet services such as Skype and iChat, web plugins and on-line telecommunication programs which promoted low cost, albeit low-quality, videoconferencing to virtually every location with an Internet connection. In May

2005, the first high definition video conferencing systems, produced by LifeSize Communications, were displayed at the Interop trade show in Las Vegas, Nevada, able to provide 30 frames per second at a 1280 by 720 display resolution. Polycom introduced its first high definition video conferencing system to the market in 2006. Currently, high definition resolution has now become a standard feature, with most major suppliers in the videoconferencing market offering it.[1,2]

1. Conference System Technology

The core technology used in videoconferencing system is digital compression of audio and video streams in real time. The hardware or software that performs compression is called a codec (coder/decoder). Compression rates of up to 1:500 can be achieved. The resulting digital stream of 1s and 0s is subdivided into labeled packets, which are then transmitted through a digital network of some kind (usually ISDN or IP). The use of audio modems in the transmission line allow for the use of POTS, or the Plain Old Telephone System, in some low-speed applications, such as videotelephony, because they convert the digital pulses to/from analog waves in the audio spectrum range. The other components required for a videoconferencing system include:[3]

- **Video input** : video camera or webcam
- **Video output**: computer monitor , television or projector
- **Audio input**: microphones, CD/DVD player, cassette player, or any other source of PreAmp audio outlet.
- **Audio output**: usually loudspeakers associated with the display device or telephone
- **Data transfer**: analog or digital telephone network, LAN or Internet

There are basically two kinds of videoconferencing systems:

1. **Dedicated systems** have all required components packaged into a single piece of equipment, usually a console with a high quality remote controlled video camera. These cameras can be controlled at a distance to pan left and right, tilt up and down, and zoom. They became known as PTZ cameras. The console contains all electrical interfaces, the control computer, and the software or hardware-based codec. Omnidirectional microphones are connected to the console, as well as a TV monitor with loudspeakers and/or a video projector. There are several types of dedicated videoconferencing devices:

- Large group video conferencing are non-portable, large, more expensive devices used for large rooms and auditoriums.

- Small group video conferencing are non-portable or portable, smaller, less expensive devices used for small meeting rooms.
- Individual video conferencing are usually portable devices, meant for single users, have fixed cameras, microphones and loudspeakers integrated into the console.

2. **Desktop systems** are add-ons (hardware boards, usually) to normal PCs, transforming them into video conferencing devices. A range of different cameras and microphones can be used with the board, which contains the necessary codec and transmission interfaces. Most of the desktops systems work with the H.323 standard. Videoconferences carried out via dispersed PCs are also known as e-meetings.

1.1 Conferencing layers

The components within a Conferencing System can be divided up into several different layers: User Interface, Conference Control, Control or Signal Plane and Media Plane.

- Video Conferencing User Interfaces could either be graphical or voice responsive. Many of us have encountered both types of interfaces, normally we encounter graphical interfaces on the computer or television, and Voice Responsive we normally get on the phone, where we are told to select a number of choices by either saying it or pressing a number. User interfaces for conferencing have a number of different uses; it could be used for scheduling, setup, and making the call. Through the User Interface the administrator is able to control the other three layers of the system.
- Conference Control performs resource allocation, management and routing. This layer along with the User Interface creates meetings (scheduled or unscheduled) or adds and removes participants from a conference.
- Control (Signaling) Plane contains the stacks that signal different endpoints to create a call and/or a conference. Signals can be, but aren’t limited to, H.323 and Session Initiation Protocol (SIP) Protocols. These signals control incoming and outgoing connections as well as session parameters.
- The Media Plane controls the audio and video mixing and streaming. This layer manages Real-Time Transport Protocols, User Datagram Packets (UDP) and Real-Time Transport Control Protocols (RTCP). The RTP and UDP normally carry information such the payload type which is the type of codec, frame rate, video size and many others. RTCP on the other hand acts as a quality control Protocol for detecting errors during streaming.[4,5]

1.2 Multipoint video conferencing

Simultaneous videoconferencing among three or more remote points is possible by means of a Multipoint Control Unit (MCU). This is a bridge that interconnects calls from several sources (in a similar way to the audio conference call). All parties call the MCU unit, or the MCU unit can also call the parties which are going to participate, in sequence. There are MCU bridges for IP and ISDN-based videoconferencing. There are MCUs which are pure software, and others which are a combination of hardware and software. An MCU is characterized according to the number of simultaneous calls it can handle, its ability to conduct transposing of data rates and protocols, and features such as Continuous Presence, in which multiple parties can be seen on-screen at once. MCUs can be stand-alone hardware devices, or they can be embedded into dedicated videoconferencing units.[6]

The MCU consists of two logical components:

1. A single multipoint controller (MC), and
2. Multipoint Processors (MP), sometimes referred to as the mixer.

The MC controls the conferencing while it is active on the signaling plane, which is simply where the system manages conferencing creation, endpoint signaling and in-conferencing controls. This component negotiates parameters with every endpoint in the network and controls conferencing resources While the MC controls resources and signaling negotiations, the MP operates on the media plane and receives media from each endpoint. The MP generates output streams from each endpoint and redirects the information to other endpoints in the conference. Some systems are capable of multipoint conferencing with no MCU, stand-alone, embedded or otherwise. These use a standards-based H.323 technique known as "decentralized multipoint", where each station in a multipoint call exchanges video and audio directly with the other stations with no central "manager" or other bottleneck. The advantages of this technique are that the video and audio will generally be of higher quality because they don't have to be relayed through a central point. Also, users can make ad-hoc multipoint calls without any concern for the availability or control of an MCU. This added convenience and quality comes at the expense of some increased network bandwidth, because every station must transmit to every other station directly.

1.3 Video conferencing modes

Videoconferencing systems have several common operating modes that are used:

1. Voice-Activated Switch (VAS);
2. Continuous Presence.

In VAS mode, the MCU switches which endpoint can be seen by the other endpoints by the levels of one’s voice. If there are four people in a conference, the only one that will be seen in the conference is the site which is talking; the location with the loudest voice will be seen by the other participants. Continuous Presence mode display multiple participants at the same time. The MP in this mode puts together the streams from the different endpoints and puts them all together into a single video image. In this mode, the MCU normally sends the same type of images to all participants. Typically these types of images are called “layouts” and can vary depending on the number of participants in a conference.[7]

2. <u>Application Programming Interface (API)</u>

An application programming interface (API) is a particular set of rules and specifications that a software program can follow to access and make use of the services and resources provided by another particular software program that implements that API. It serves as an interface between different software programs and facilitates their interaction, similar to the way the user interface facilitates interaction between humans and computers. An API can be created for applications, libraries, operating systems, etc, as a way to define their "vocabularies" and resources request conventions (e.g. function-calling conventions). It may include specifications for routines, data structures, object classes, and protocols used to communicate between the consumer program and the implementer program of the API. An API can be:

- General, the full set of an API that is bundled in the libraries of a programming language, e.g. Standard Template Library in C++ or Java API.
- Specific, meant to address a specific problem, e.g. Google Maps API or Java API for XML Web Services.
- Language-dependent, meaning it is only available by using the syntax and elements of a particular language, which makes the API more convenient to use.
- Language-independent, written so that it can be called from several programming languages. This is a desirable feature for a service-oriented API that is not bound to a specific process or system and may be provided as remote procedure calls or web services. For example, a website that allows users to review local restaurants is able to layer their reviews over maps taken from Google Maps, because Google Maps has an API that facilitates this functionality. Google Maps' API controls what information a third-party site can use and how they can use it.[9,10]

3. <u>Telephony Application Programming Interface (TAPI</u>)

The Telephony Application Programming Interface (TAPI) is a Microsoft Windows API, which provides computer telephony integration and enables PCs running Microsoft Windows to use telephone services. Different versions of TAPI are available on different versions of Windows. TAPI allows applications to control telephony functions between a computer and telephone network for data, fax, and voice calls. It includes basic functions, such as dialing, answering, and hanging up a call. It also supports supplementary functions, such as hold, transfer, conference, and call park found in PBX, ISDN, and other telephone systems. TAPI is used primarily to control either modems or, more recently, to control business telephone system (PBX) handsets. When controlling a PBX handset, the driver is provided by the manufacturer of the telephone system.[11,12]

Some manufacturers provide drivers that allow the control of multiple handsets. This is traditionally called "third-party control". Other manufacturers provide drivers that allow the control of a single handset. This is called "first-party control". Third-party drivers are designed to allow applications to see and/or control multiple extensions at the same time. Some telephone systems only permit one third-party connection at a time. First-party drivers are designed to allow applications to monitor and/or control one extension at a time. Telephone systems naturally permit many of these connections simultaneously. Modem connections are by nature first-party. TAPI can also be used to control voice-enabled telephony devices, including voice modems and dedicated hardware such as Dialogic cards

4. Proposed System Design

The proposed system contains the database server, that store information about the user such as phone number, addresses , computer name ,computer internet addresses. Each user can be connect with other user using video communication, in addition that can be using TAPI Telephony to connect computer with Analog phone and vice versa.

The database designed according to the proposed system in figure (1), that contains information about users in table (1).

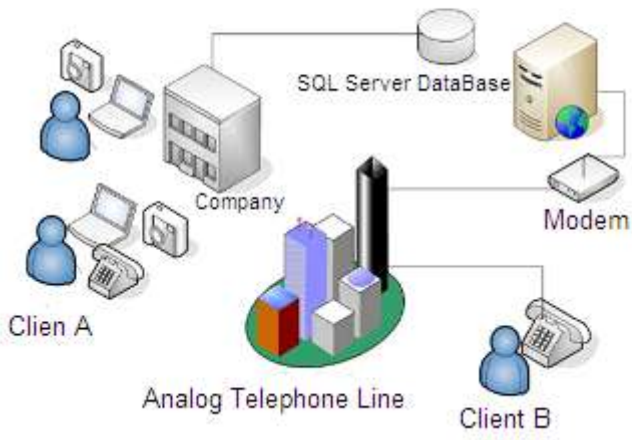


Figure (1) The proposed system Design.

Name@Department	String (Primary Kay)
Computer Name (hidden for Clients)	String
Telephone Number	String
Group ID	String (foreign Kay)
Other Info (Email Address)	String

User - information

Column Name	Data Type	Length	Allow Nulls
Name_Department	varchar	50	
Telephone_Number	varchar	50	
Computer_Name	varchar	50	
Group_ID	varchar	50	
Other_Info	varchar	150	✓

Table (1) The information about users.

After designing the database, the system has been client monitors that use by user in room or company or any other place to connect with other client or server. When user determines the other side want to connect the Combo box is display and when check the system makes SELECT Query to the private Telephone Number for the user. The system takes the private Computer – Name and then can make call using the TAPI Telephony or VOIP for voice or video. So there are three cases:

In the first case that using TAPI Telephony the system using Modem and analog line for connect.

In the second case that using VOIP connect, the system start by send voice to the Computer – Name that want to connect with to tell client about the existence of a voice call waiting. In the case of accepting converts audio to client computer. In the case three that using VOIP connects for video the system using the same stapes as in the second case but sending the video to the client. The proposed system used the TAPI Telephony in Visual Basic Dot Net by using the methods that provided by custom file to deal with TAPI 32.dll that contains a large numbers of methods likes:

1. TapiRequestMakeCall (string stNumber, string stDummy1, string stDummy2, string stDummy3) this function used for communication by using modem that used the phone number for connect.

2.PhoneShutdown(hPhoneApp app)

[DllImport (“tapi32.dll”]

Static extern long phoneShutdown(long hPhoneAPP); This function used for closed the TAPI Telephony program.

3.LongWinAPIphonySetVolume (

HPHONE hPhone,

DWORD dwHoookSwitchDev,

DWORD dwVolume);

This function used for determine the degree of sound and using Hexadecimal for representation the degree such as 0x00000000 this value indicated no sound, 0x0000FFFF this indicated the maximum sound.

In this work used the **tapiRequestMakeCall** Method for input the phone number to connect anther computer as shown in the algorithm (1) :

Algorithm (1) : The request for call.

```
< system.Runtime.InteropServices.DllImport("tapi32.dll" ) >_
Shard Function tapiRequestMakeCall ( ByVal stNumber As String, ByVal stDummy1 As
String,ByVal stDummy2 As String, ByVal stDummy3 As String ) As Long
End Function

Dim RetVal As Long = tapiRequestMakeCall (Telephone_number, " " , " " ," " ) ;
```

For connect two or more computers used one computer as server and anther computer as client by using the algorithms for sending and receiving sound as shown in the algorithm (2):

Algorithm (2) : Sending and Receiving Sound.

```
Private Sub Voive_In ( )
    Dim br As Byte ( )
    r.Bind (New IPEndPiont(IPAddress.Any,Integer.Parse(text_ReceiveingPORT.Text)))
    Do While True
    br = New Byte (16383)
    r.Receive(br)
   m_Fifo.Write(br,0,br.Length)
Loop
End Sub

Private Sub Voive_Out ( )
     ( By Val data As IntPtr,By Val size As Integer)
    If m_RecBuffer Is Nothing OrElse m_RecBuffer.Length< size Then m_RecBuffer = New
Byte(size -1)
End If
System.Runtime.InteropServices.Marshal.Copy(data,m_RecBuffer,0,size)
r.SendTo(m_RecBuffer,NewIPEndPoint(IPAddress.Parse(Server_name),
Integer.Parse(text_Sendingport.Text)))
End Sub
```

For sending and receiving video between many computers using the algorithm (3):

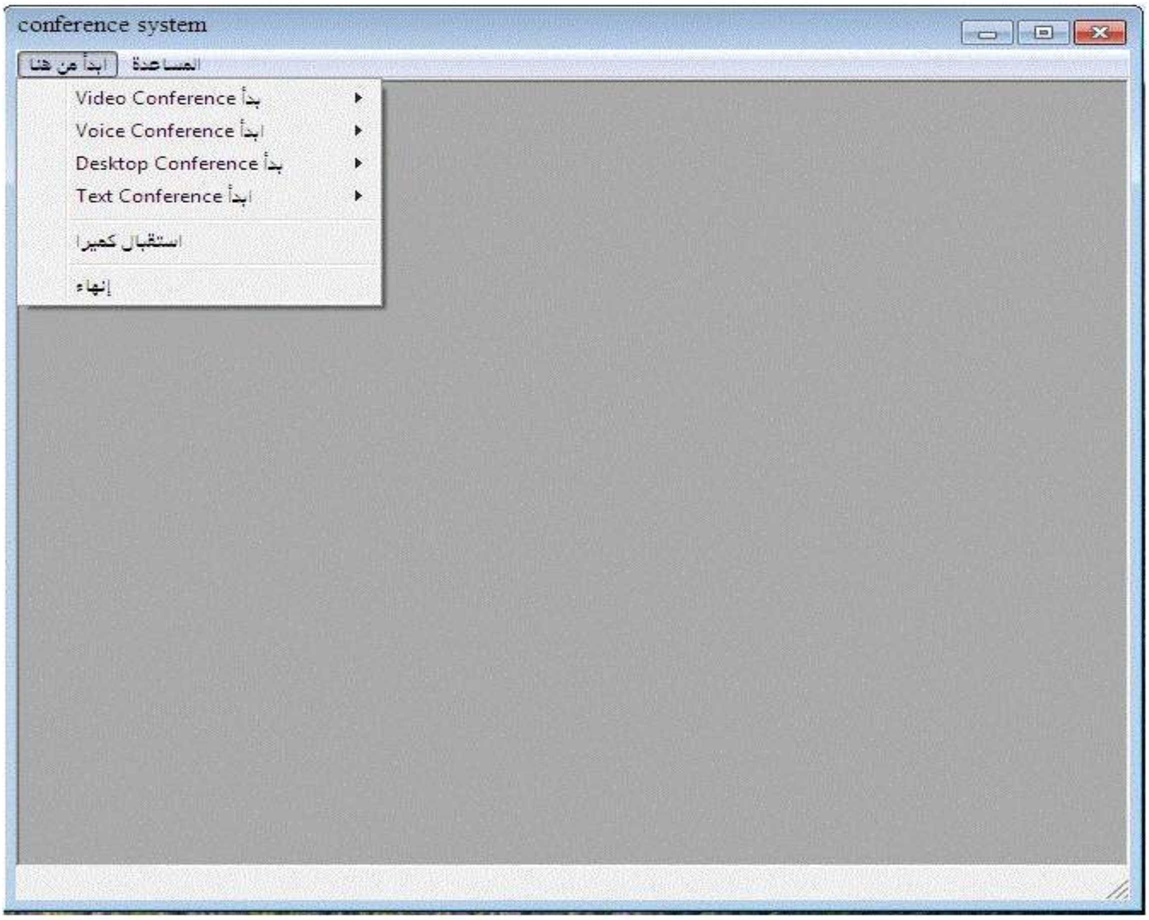
Algorithm (3) : The video conference.

```
Private Sub Start_Receiving_Video_Conference ( )
   Try
   Mytcp1 = New
TcpListener ( Integer.Parse(text_Camera_rec_port.Text))
Mytcp1.Start( )
Mysocket = mytcp1.AcceptSocket( )
Ns = New NetworkStream (mysocket)
pictureBox2.Image = Image.FormString(ns)
mytcp1.Stop( )
If mysocket.Connected = True Then
Do While True
    Start_Receiving_Video_Conference ( )
Loop
End If
Myns.Flush( )
Catch e1 As Exception
End Try
End Sub
```

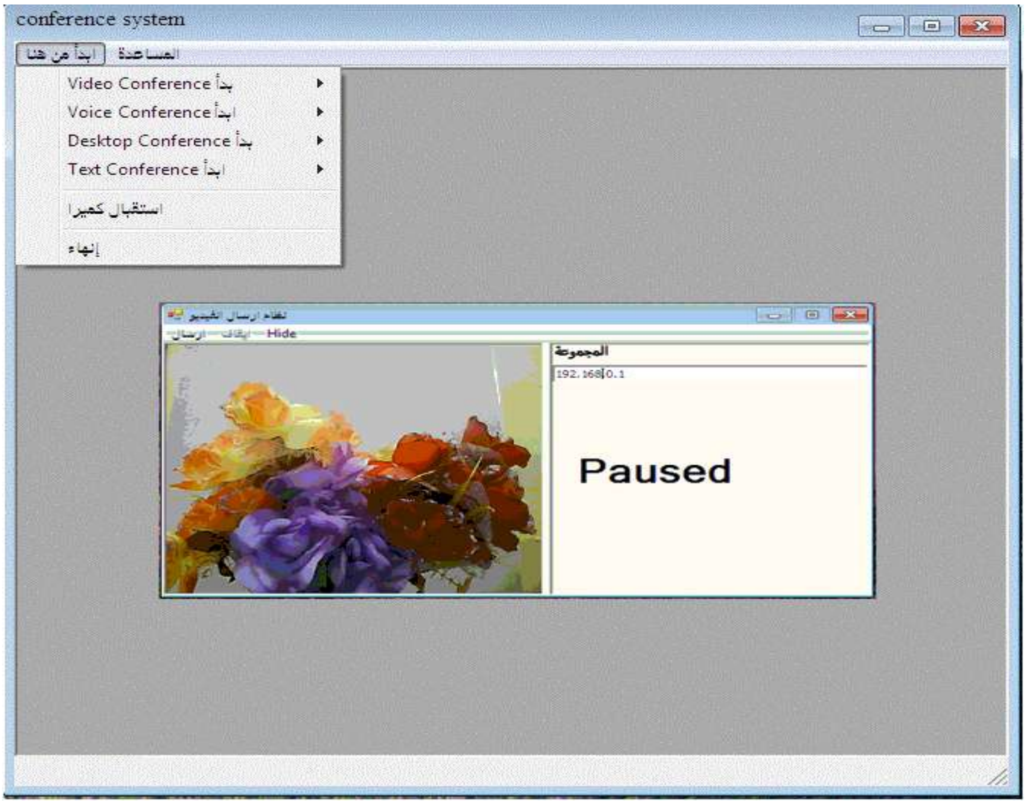
```
Private Sub Start_Sending_Video_Conferece (By Val remote_IP As String,By Val port_number
As Integer)
     Try
     Dim ms As memoryStream =  New MemoryStream
     pictureBox1.Image.Save(ms,System.Drawing.Imaging.ImageFormat.Jpeg)
     Dim arrImage As Byte ( )  =  ms.GetBuffer( )
     Myclient = New TCPClinet(server_IP,port_number)
     Myns = myclient.GetStream( )
     Mysw = New BinaryWrite(myns)
     Mysw.Write(arrImage)
     Ms.Flush( )
     Mysw.Close ( )
     Myclient( )
End Try
```

5. Results and Discussion

When run the program must chose the types of conference such as video conference or voice conference to communicated with other computer as shown in figure (2)



the video conference must write the port number to send and received video between computers as shown in figure Figure (2) the types of conference system (video and voice).



.

Then program used for video conference between computers as shown in figures (4) and figure (5).

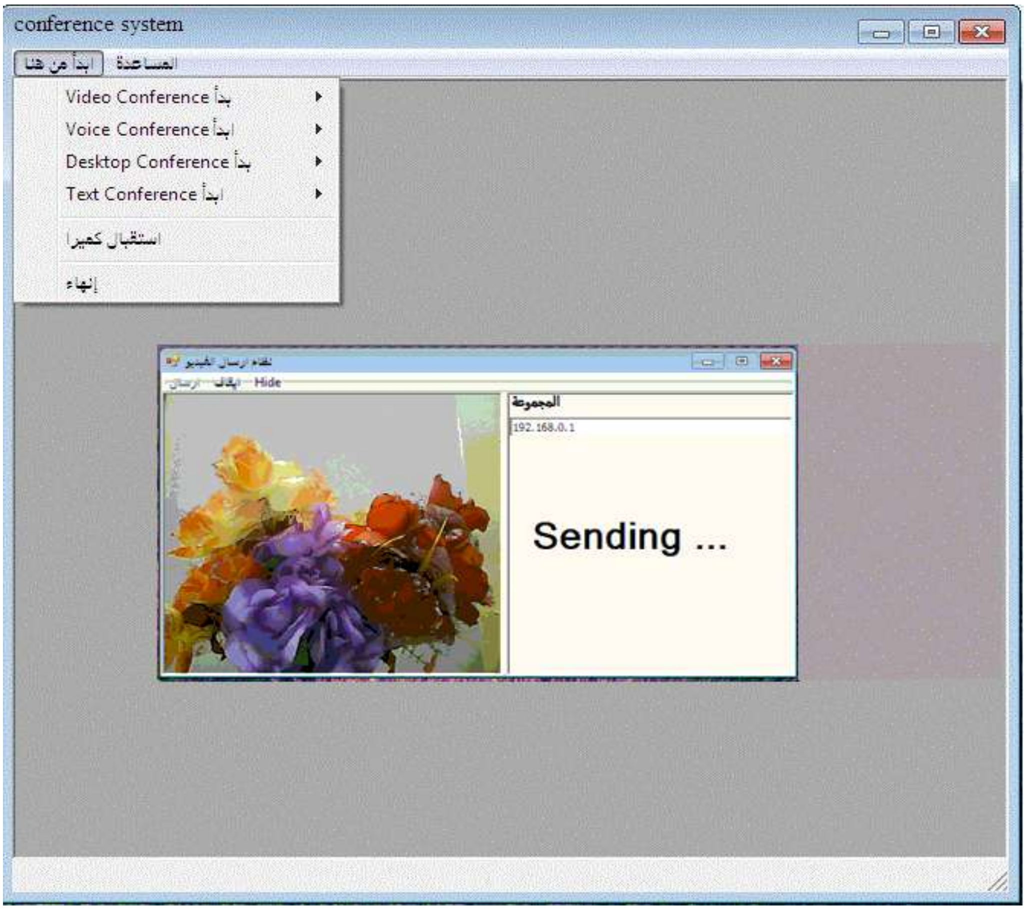


Figure (5) the video conference system after sending video

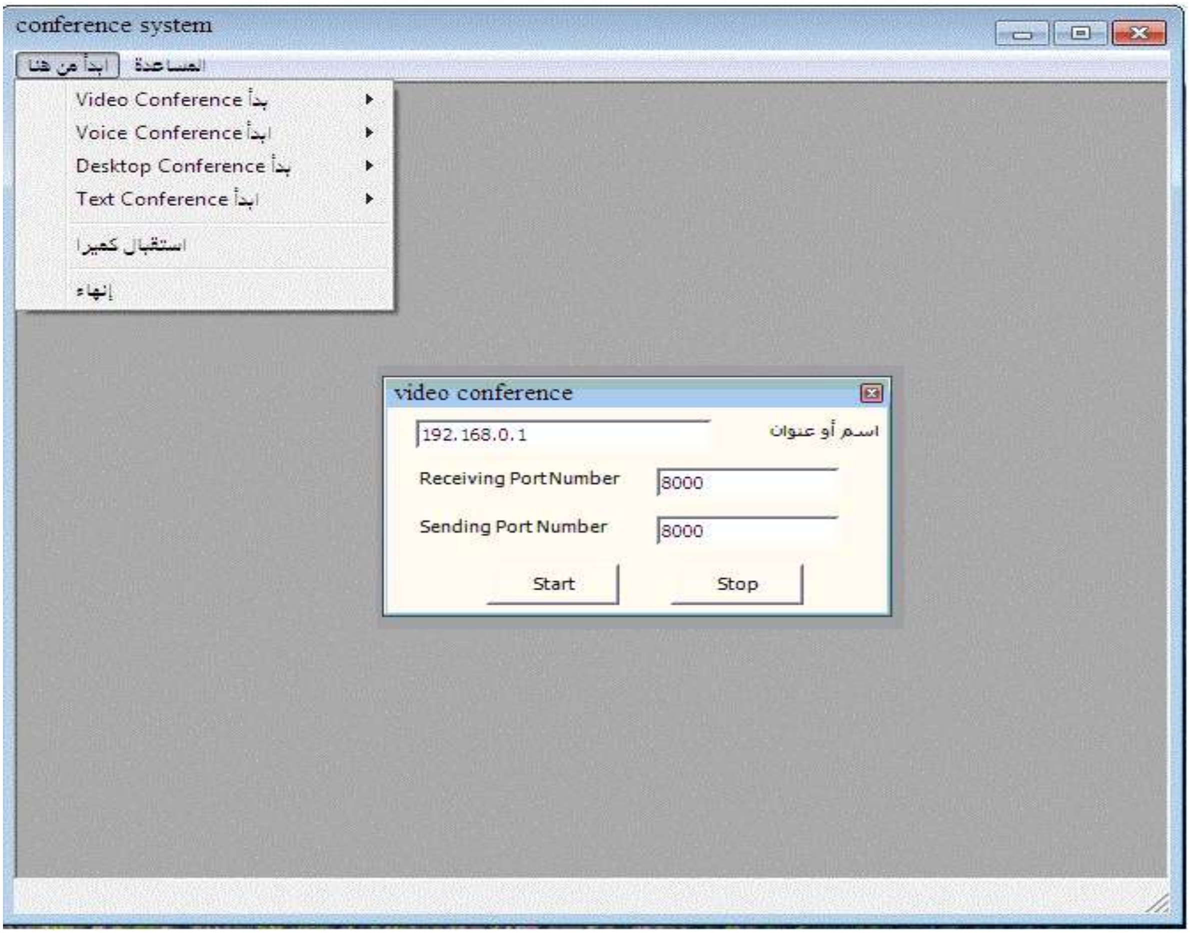


Figure (5) the video conference system after sending video

When choosing video conference can used the stream format such as frame rate , frame interval, output size, quality, compression methods and color space as shown in figure (6).

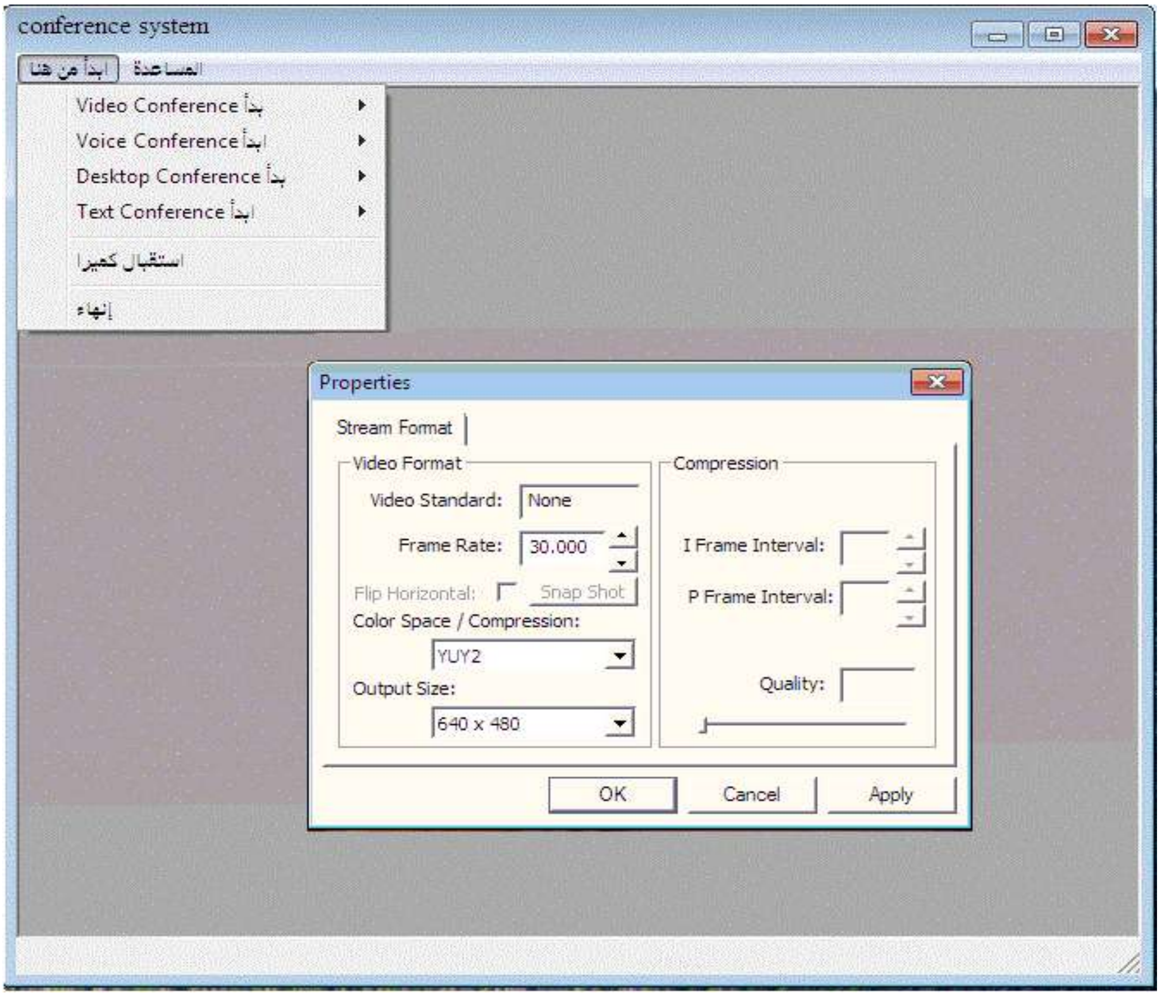


Figure (6) the video conference stream format

There are two major factors affecting the quality of video or voice transmission over networks. These are:

1. Delay.

2. Bandwidth

1. Delay

End-to-end delay does not affect voice quality directly, but instead affects the flow of conversation and may result in talker overlap [13]. ITU-T Recommendation G.114 provides limits for one-way transmission time (delay) [14]. These times are shown in Table 2.1. If the delay through the network is less than 150 ms, users will not notice. When the delay is between 150 ms and 400 ms, users will notice hesitation in the conversation. This hesitation can affect how each listener perceives the mood of the conversation and interruptions are more frequent. As the delay approaches 400 ms, the conversation gets out of beat. Beyond 400 ms, the delay is very obvious to users and conversation becomes virtually impossible.When tested proposed program that using API and TAPI found the delay time is 135 ms so that is very good result and the user not notice the delay time.

One-Way Transmission Time (Delay)	User Acceptance
0 to 150 ms	Acceptable for most users.
150 to 400 ms	Acceptable, but has impact on conversation.
400 ms and above	Unacceptable.

Table (2) G.114 Limits for One-Way Transmission Time.

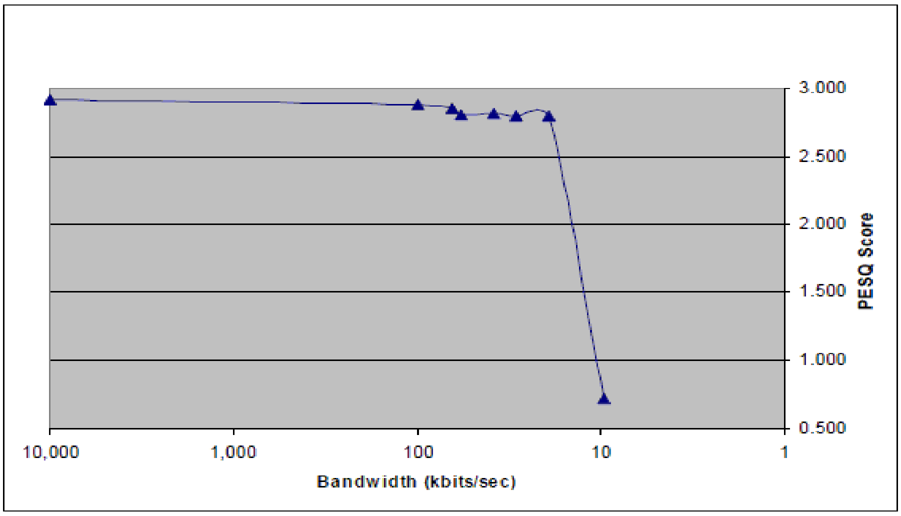
2. Bandwidth Limitations

Bandwidth limitations can have a severe effect on video / voice quality in a system. The minimum bandwidth required for the transmission of voice over an IP network is dependent on the audio codec being used. In addition to the voice information, various header information increases the minimum bandwidth requirements. Bandwidth requirements for various audio codecs are shown in Table 3.

Audio Codec	Encoding	Bandwidth Required (information alone)	Bandwidth Required (with IP header)
G.711	PCM	64 kbps	80 kbps
G.723	ACELP	5.6 / 6.4 kbps	16.2 / 17.1 kbps
G.729A	CS – ACELP	8 kbps	24 kbps

Table (3) Bandwidth Requirements for Various Audio Codecs.

The bandwidths used in this test were: 10,000, 100, 64, 57.6, 38.4, 28.8, 19.2, and 9.6 kbit/sec. The audio codec used for this and all remaining tests is G.723, which has a nominal bit rate of 6.4 kbit/sec. The notation of S1, S2, and S3 used in this project is used to indicate speakers 1 through 3 respectively. The PESQ scores resulting from each of the five testing iterations for S1, S2, and S3 were averaged together. A PESQ score of 3.0 is the best possible score and indicates excellent voice quality. A score between 2.0 and 3.0 indicates good voice quality with some noticeable degradation. A score between 2.0 and 1.0 indicates the voice quality is significantly degraded and a listener may encounter difficulty in recognizing some words. A score below 1.0 indicates voice quality which is degraded to a point where a listener will have difficulty recognizing all words. The results for S1, S2, and S3 are shown in Figure 7, Figure 8, and Figure 9 respectively. The PESQ scores from S1, S2, and S3 were averaged together to show the general trend in voice quality as bandwidth is decreased. The average of the PESQ scores is shown in Figure 10.



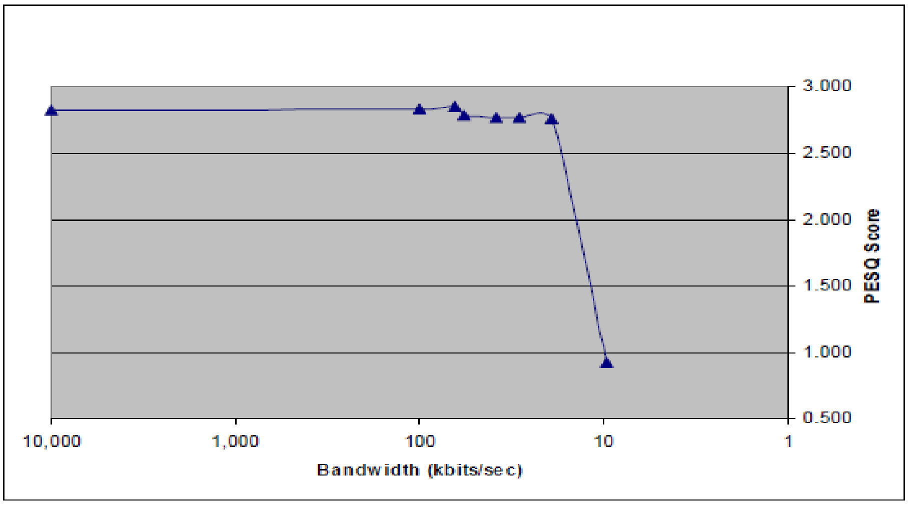


Figure (8). Average PESQ Scores for Varied Bandwidth – S2

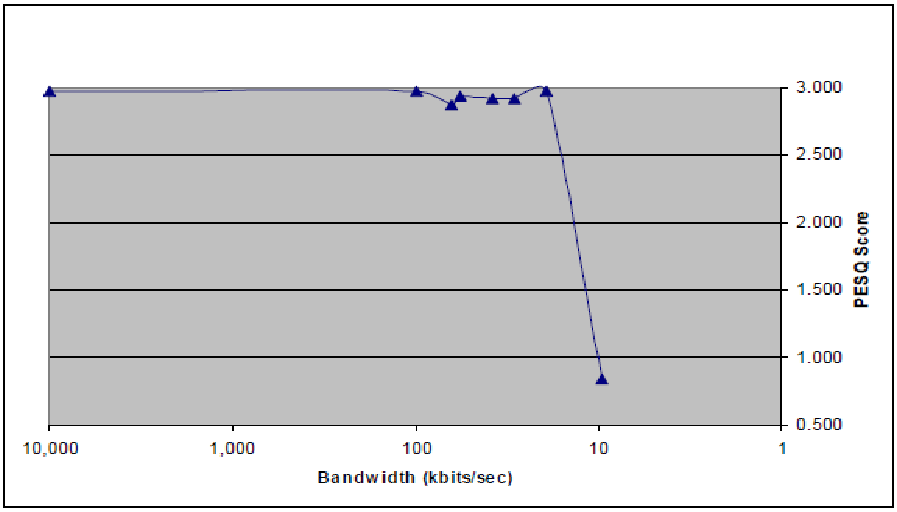
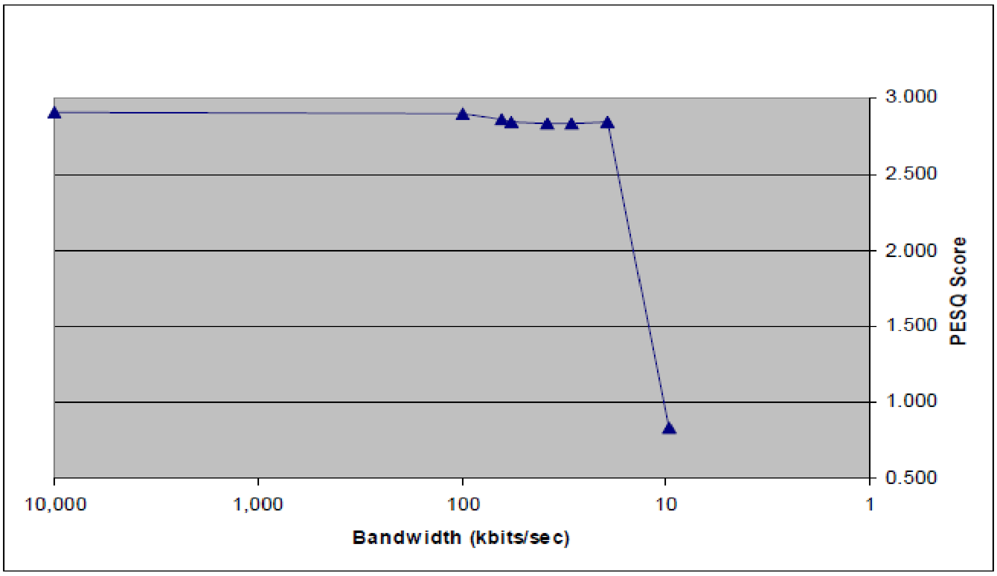


Figure (9). Average PESQ Scores for Varied Bandwidth – S3



6. Conclusion

A system design that using VoIP network communication was very useful to conference system. The main goal of this project was to design, implement, and test a conference system to video / voice sending and receiving. A system that connected computers using the API and TAPI telephony in order to transfer voice and video from one computer to another computer. This Functions more useful in conference system so when used its the delay time are reduce to less than 150 ms and the bandwidth required are reduced to the range of acceptable numbers. The packet loss is smaller and remained in the acceptable levels so that not effected on the quality of the video or on the voice.

References

1. Weiner, Richard.”Conferencing techniques: Audio, video and Internet”. Public Relations Tactics, 7, 12-13. , June 2000.

2. SangWoo Han and JongWon Kim, "High-quality Video Service for Access Grid", in Proc. of International Conference on Artificial Reality and Telexistence, Seoul, Korea, November 2004.

3. Hong Liu; Mouchtari,” Voice over IP signaling”, IEEE Communications Magazine, Vol.: 38 Issue: 10 , Oct. 2000.

4. Thomsen, G.; Jani, Y.” Internet telephony” IEEE Spectrum , Vol.: 37 Issue: 5, May 2000

5. Jonathan Davidson, “Voice over IP,” Cisco Systems, June 1998, http://www.dtr.com.br/cdrom/cc/sol/mkt/ent/ndsgn/voice_dg.htm April 2003.

6. Mehta, P.; Udani, S.,” Voice over IP” IEEE Potentials, Vol.: 20 Issue: 4, Oct.-Nov. 2001.

7. M. Baldi, G.P. Picco, and F. Risso, Designing a Videoconference System for Active Networks, in Proceedings of the 2nd International Workshop on Mobile Agents, Stuttgart, September 1998, http://www.polito.it/~risso/pubs/index.htm.

8. R.J.B. Reynolds, A.W. Rix, “Quality VoIP – an engineering challenge,” BT Technology Journal, April 2001, pp. 24

9. S. Yamarthy, “Subjective and Objective Evaluation of Voice Quality Over Internet Telephony,” U. of New Hampshire, Durham, 2000

10. Alberta Education, Video-conferencing Research Community of Practice:Research Report. Conference Alberta, 2006. http://www.vcalberta.ca/community/Research_Summary_Report_word_version_fi nal.pdf

11. “IP Telephony with TAPI 3.0,” MSDN Library, April 2000.

12. Akimichi Ogawa, Katsushi Kobayashi, Kazunori Sugiura, Osamu Nakamura, Jun Murai, "Design and Implementation of DV based video over RTP", Packet Video Workshop 2000, May 2000.

13. S. Pracht, D. Hardman, “Voice Quality in Converging Telephony and IP Networks,” Agilent Technologies Whitepaper, August 2002, pp. 11

14. “One-way Transmission Time,” ITU-T Recommendation G.114, May 2000.