

VOICE CONTROLLED LOCAL AREA NETWORKS

Dona mol Kurian¹, Athira Renjan², K.selva sheela³(AP/CSE)

¹PG Student/Department of CSE, Veerammal Engineering College, Dindigul.
donamolkurian.7@gmail.com

²PG Student/Department of CSE, Veerammal Engineering College, Dindigul.
athrarenjan8@gmail.com

³PG Student/Department of CSE, Veerammal Engineering College, Dindigul.
sheelaarume@yahoo.co.in

Abstract

The main aim of this paper to introduce voice controlled computer system using fuzzy logic. These allow the users to control the computer network or systems in a network using their voice. It specifies how to use fuzzy logic on an advanced 8-bit arduino microcontroller with HM2007 voice recognition system for voice command recognition. Over the next few years, voice will become a common interface in most of the non-pc devices, which will be using Linux. Voice recognition requires users to train the voice recognition kit to recognize his or her particular speech patterns. In a voice controlled system the users are provided with passwords by the administrator. The administrator has the privilege to add, edit and remove users. The users can change their passwords if needed. Using this system both the administrator and the users can perform system level operations and network level applications, it can control any system in a network. The system is efficient and effective in terms of speed and time. Voice recognition improves given faster processors, more money and more useful for physically challenged persons.

Keywords-*Fuzzy logic, HM2007 voice recognition kit, arduino micro controller*

1. Introduction

The human voice is the most natural user interface for communication and computing on a variety of devices. For command and control applications in PC and non-PC devices, voice recognition is an excellent interface. The hardware just needs the right programming and the sound of your voice. There is a wide research and development in this area trying to take maximum advantage of this technology, sand in coming years many new applications and research areas will continue to appear. We propose a computer system that uses HM2007

voice recognition kit and arduino microcontroller to demonstrate how to control the computer system using voice with fuzzy logic.

Speech recognition is also an integral part of this software, which is a technology that allows the computer to identify and understand words spoken by a person. Speaking is easier and more intuitive than selecting buttons and menu items. Human speech has evolved over many thousands of years to become an efficient method of sharing information and giving instructions. This computer software will free a person from having to sit in front of a computer and require them to do mouse clicks, hunt for tiny icons and navigate menu structures in order to obtain information, perform tasks or run programs. This software will make the system administrators jobs in ease mode. Using this system both the administrator and the users can perform system level operations and can use certain applications. The system is efficient and effective in terms of speed and time.

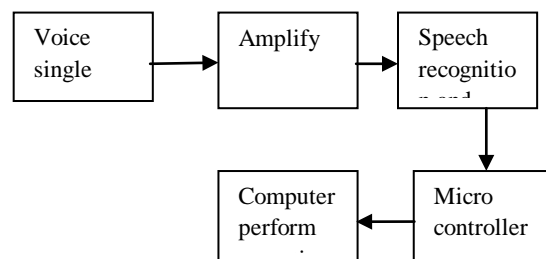


Fig:1 command extraction process

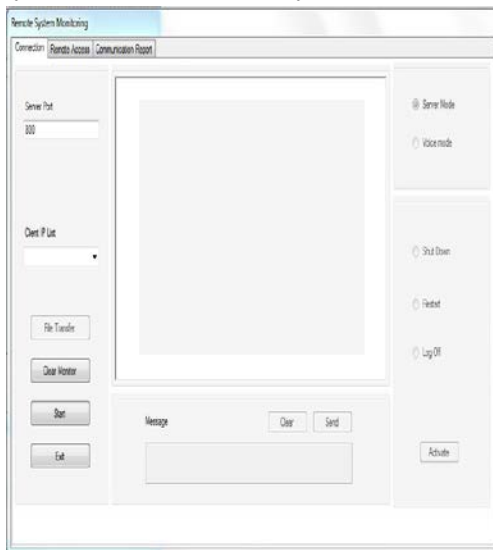
As shown in Fig. 1, the administrator provides voice commands and the voice commands amplified and stored by HM2007 voice recognition kit. microcontroller converts electrical pulse into digital signal. Finally computer perform the operation as per the voice command. In direct mode the admin can perform

the shutdown, restart and logoff the main advantage of the direct mode is file shearing all of these can be done by the following window

2. Modes of the operation

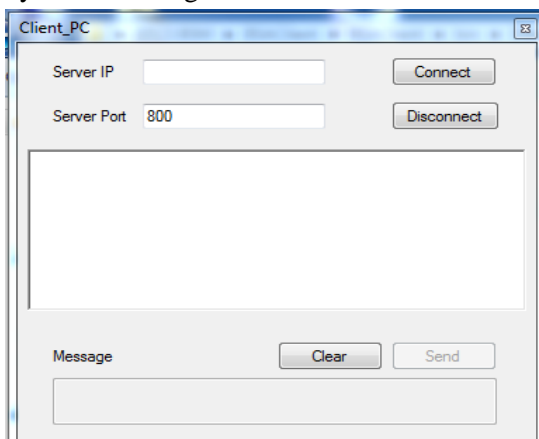
2.1 server mode

The server mode is designed to control all the system in the local area and the server mode ensure two modes of operation voice control and direct control in voice control admin sent voice commands and the client perform the requested operations and in client mode it control the system via mouse and keyboard



2.2 Client mode

The clients are should be controlled by the server via voice command and direct mode. The client can chat with the server and receive the file and send file to the server all of these can performed by the following window



3. Modules of operation

3.1 List Systems in Network

This module is accessible to all the users of the server application. All the systems, except the server system, connected to the network are listed in a combo box using this module. The user can select any of the listed systems and proceed to control it. The class named 'Directory Entry', which is in the 'System.DirectoryServices' namespace is used for this purpose. The Active Directory Services Interfaces (ADSI) helps the administrator to locate and manage resources on a network with relative ease, regardless of the size of the network. The ADSI helps the administrator to access the Active Directory Domain Services Hierarchy. The ADDS has different providers like Internet Information Services (IIS), Light weight Directory Access Protocol (LDAP), Novell Netware Directory Services (NDS) and WinNT. The WinNT provider provides the details of the Local Area Network. The WinNT provider is used along with the Directory Entry class to list all the systems in connected to the local network. The server system is excluded from this list by a simple comparison of the names of the system. The user can select any of the listed systems and click the 'Control' button to control it. When this button is clicked the form to select the controlling operation is displayed. The Microsoft Windows NT ADSI provider implements a set of ADSI objects to support various ADSI interfaces. The namespace name for the Windows NT provider is "WinNT" and this provider is commonly referred to as the WinNT provider. To access the WinNT provider, bind to any of the [ADSI object of winNT](#), using the [WinNT AdsPath](#). The WinNT provider is included in the ADSI system component for Windows Server 2003, Windows XP, and Windows 2000. If the client is a Windows 95/98/Me workstation or Windows NT 4.0 workstation/member server, the client must have either ADSI 2.5 or the Active Directory Client Extensions (DSClient) installed. The DSClient is the preferred ADSI add-on client

3.2 Direct mode

This module is accessible to all the users of the application. The user can select the controlling operation (Shutdown, Reboot and Logoff) to be

performed on the system selected in the previous module. When the controlling form is displayed the IP address of the local system (running the server application) and the client system (selected in the previous module) are displayed. This is done using the class 'IPAddress' which is in the 'System.Net' namespace. After selecting the controlling operation, all the user has to do is click the 'Ok' button. When the 'Ok' button is clicked the server application will send a message to the client system through the Network Stream specifying the operation to be performed. The class named 'Network Stream' which is in the 'System.Net.Sockets' namespace is used to get the stream of the client system using its IPAddress. The 'Stream Writer' Controlling a System (Direct). This module is accessible to all the users of the application. The user can select the controlling operation (Shutdown, Reboot and Logoff) to be performed on the system selected in the previous module. When the controlling form is displayed the IP address of the local system (running the server application) and the client system (selected in the previous module) are displayed. This is done using the class 'IPAddress' which is in the 'System.Net' namespace. After selecting the controlling operation, all the user has to do is click the 'Ok' button. When the 'Ok' button is clicked the server application will send a message to the client system through the Network Stream specifying the operation to be performed. The class named 'Network Stream' which is in the 'System.Net.Sockets' namespace is used to get the stream of the client system using its IPAddress. The 'Stream Writer' which is also in the 'System.IO' namespace. All these operations are done in a procedure which is started using the object of the class 'Thread' which is in the 'System. Threading' namespace.

3.3 Voice mode

This module is accessible only to the administrator. The administrator can control a client system even if he/she is not present in person at the system running the server application. A VCC unit (supporting AT

commands) is connected to the system running the Server application. The administrator can control a client voice commanding mode. The server application reads this command and decodes the information contained in it. The processes performed in the previous module are repeated based on the information in the VCC Command. Connect the HM2007 voice recognition kit to the 8 bit arduino microcontroller using wires. USB to serial port adapter from arduino microcontroller is connected to the serial port of the computer (com 3). Open the arduino microcontroller software and configure the terminal according to the requirements of the hardware module. Compile and upload the corresponding code to the I/O board and confirm that connection is working. Using a code, listen whether any new happening is occurring in the serial port. If some data is available in the serial port, the corresponding system level operations or some applications will be executed. Otherwise it will show error codes.

3.4 Client Application

Every system in the network, except the system running the server application, must run the client application. This application will constantly listen to the network for any messages from the server application. The communication between the server and the client will always take place through the port number 63000. The client application will listen to this port for any message or command arriving from the server. When a message is received the application will read it from the Network Stream and proceed to perform the operations specified in the message. The 'TcpListener' class which is in the 'System.Net.Sockets' namespace is used to listen for incoming connection requests on the specified IP address and port number. The 'TcpClient' class is used to accept a pending connection request. The 'NetworkStream' class is used to get the stream of the system and the 'StreamReader' class is used to read data from this Network Stream. The data read from the stream is checked to find out the operation to be performed on the system. A function, whose

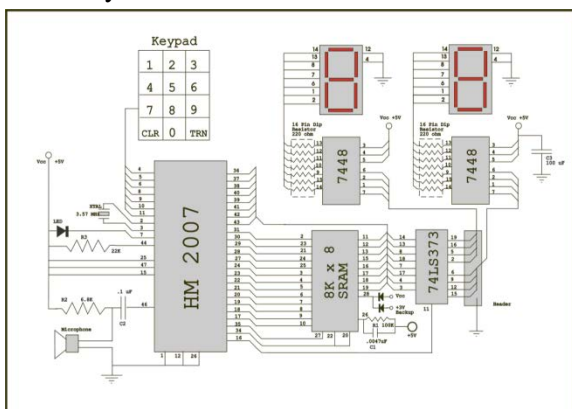
argument specifies the operation to be performed, is called with the relevant argument and the specified operation is performed. the class named 'ManagementBaseObject' and 'Management Class' belonging to the 'System.Management' namespace are used to perform the specified operations on the system. The parameters of the method to be called ("Win32Shutdown") are found using the 'ManagementBaseObject' and the method is invoked.

3. 5 Screen view

This module is accessible to the administrator. Administrator can monitor clients screen. Here current screen of the client is captured and send to administrator. We take screenshots as frequently as one, per second. Thus it appears as almost as a live video. The spying job is stealth, small and best of all nothing can get pass it. Screen View is based on Client-Server architecture.

4. HM2007 voice recognition kit

The speech recognition system is a completely assembled and easy to use programmable speech recognition circuit. Programmable, in the sense that you train the words (or vocal utterances) you want the circuit to recognize. This board allows you to experiment with many facets of speech recognition technology. It has 8 bit data out which can be interfaced with any microcontroller for further development. Some of interfacing applications which can be made are controlling home appliances, robotics movements, Speech Assisted technologies, Speech to text translation, and many more.



The speech recognition system is speaker dependant, meaning that the voice that trained the system has the highest recognition accuracy. But you can simulate independent speech recognition. To make the recognition system simulate speaker independence one uses more than one word space for each target word. Now we use four word spaces per target word. Therefore we obtain four different enunciations of each target word. (Speaker independent).The word spaces 01, 02, 03 and 04 are allocated to the first target word. We continue do this for the remaining word space. For instance, the second target word will use the word spaces 05, 06, 07 and 08. We continue in this manner until all the words are programmed. If you are experimenting with speaker independence use different people when training at argots word. This will enable the system to recognize different voices, inflections and enunciations of the target word. The more system resources that are allocated for independent recognition the more robust the circuit will become. If you are experimenting with designing the most robust and accurate system possible, train target words using one voice with different inflections and enunciation's of the target word.

5. Conclusion

In this paper Voice Controlled Computer system, we propose a computer system that uses HM2007 voice recognition kit and arduino microcontroller to demonstrate how to control the remote computers using voice with fuzzy logic. Speech recognition is also an integral part of this software, which is a technology that allows the computer to identify and understand words spoken by a person, The software shall also provide some future scope also. It can be modified so as to provide the facility to generate various operations and interfaces to generate applications for various LAN networks.

References

- [1] Speech Oriented Computer System Handling
Anil J Kadam¹, Pallavi Deshmukh², Amita
Kamat³, Neelam Joshi⁴, Ritika Doshi⁵,
International Conference on Intelligent
Computational Systems (ICICS'2012) Jan. 7-8,
2012 Dubai
- [2] Ben Mosbah, B., "Speech Recognition for
Disabilities People Information and
Communication Technologies, 2006. ICTTA
apros; 06.2nd, Issue , 24-28 April 2006
Page(s): 864 – 869
- [3] XiaoJie Yuan, Jing Fan, "Design and
Implementation of Voice Controlled
Tetris Game Based on Microsoft SDK" 978-
-61284-774-0/11 IEEE 2011
- [4] Pabmanabhan, Michel Pichney, Large
Vocabulary speech recognition algorithms",
IEEE Computer magazine, 0018-9162/02,
Pp.42-50-2000
- [5] Md. Abdul Kader, Biswajit Singha, and Md.
"Speech Enabled Operating System Control"
Proceedings of 11th International
Conference on Computer and Information
Technology (ICCIT 2008)25-27 December,
2008, Khulna, Bangladesh 1-4244-2136-
7/08 IEEE 2008
- [6] D. LeBlanc, Y. Ben Ahmed, S. Selouani, Y.
Bouslimani, H. Hamam, "ComputerInterface
by Gesture and Voice for Users with "
Special Needs 1-4244-0674-9/06 IEEE
2006
- [7] baseform adaptation for large
vocabulary hidden markov
model based speec recognition
systems" by Gediard Rigoll CH2847-
2/90/0000-0141 IEEE 1990
- [8] Fengyu Zhou, Guohui Tian, Yang
Yang, Hairong Xiao and Jingshuai Chen,
"Research and Implementation of Voice
Interaction System Based On PC in
Intelligent Space" Proceedings of the 2010
IEEE International Conference on
Automation and Logistics August 16-20
2010, Hong Kong and Macau 978-1-4244-
8376-1/10 IEEE 2010
- [9] Titus felix furtuna, "dynamic algorithms in
Speech recognition" revista informatica
. economica nr 2(46)/2008.
- [10] Nithin Pandey, Yesh Singal, Mridula
'Visual Studio .Net Programming' Wiley-
Dreamtech
Publication Pvt.Ltd,2002