Sign Language Recognition System

Apoorva Ranjini S S
Dept. of Computer Science & Engineering
Vidya Vikas Institute of Engineering & Technology
Mysore, India
e-mail: apoorvaranjini69@gmail.com

Chaitra M
Dept. of Computer Science & Engineering
Vidya Vikas Institute of Engineering & Technology
Mysore, India
e-mail: chaitramayappa35@gmail.com

Deepika V
Dept. of Computer Science & Engineering
Vidya Vikas Institute of Engineering & Technology
Mysore, India
e-mail: deepikav414@gmail.com

Mrs Jyothi M Patil
Assistant Professor
Dept. of Computer Science & Engineering
Vidya Vikas Institute of Engineering & Technology
Mysore, India
e-mail: jyothi.nagraj@gmail.com

Abstract— For many deaf and dumb people, sign language is the principle means of communication. Normal people in such cases end up facing problems while communicating with speech impaired people. In our proposed system, we can automatically recognize sign language to help normal people to communicate more effectively with speech impaired people. This system recognizes the hand signs with the help of specially designed gloves. These recognized gestures are translated into a text and voice in real time. Thus this system reduces the communication gap between normal and the speech impaired people.

Keywords- Hand Gestures, ASL, Sensing, Signal Conditioning Processing and Communication Stage, Converting Text to Speech, Calibration.

_____*****_____

I. INTRODUCTION

Communication involves the exchange of information, and this can only occur effectively if all participants use a common language. As known to many, Sign languages are the only medium through which most of the educated deaf-mutes communicate today [1]. Sign language has proven effective in communicating across a broad spectrum of requirements from everyday needs to sophisticated concepts. There are different sign languages according to countries. Most popular and widely used amongst them is the American Sign Language.

ASL is a visual language based on hand gestures. Figure 1 shows hand gestures of ASL for English alphabets. It has been well-developed by the deaf community over the past centuries and is the 3rd most used language in the United States today. The sign language recognition system which we have developed uses a glove fitted with sensors that can interpret the 26 English letters in American Sign Language (ASL).

This paper aims to design a sign language recognition system for the vocally disabled people. We have turned to glove-based technique as it is more practical in gesture recognition which involves the use of specially designed sensor glove which produces a signal corresponding to the hand sign. As the performance of the glove is not affected by light, electric or magnetic fields or any other disturbance, the data that is generated is accurate. The data from the sensor glove is given to the system through the serial port and it is displayed in the text editor with voice output through speakers. The user can also browse the required data using sign compatible web browser.

Figure. 1 Hand gestures of American Sign Language for alphabets.

II. RELATED WORKS

Before trying to find a better way for implementing Sign Language Recognition System, we have to study the existing systems. We found few papers explaining different approaches to tackle this problem. First of them, titled 'Hand Gesture Recognition for Human-Machine Interaction' [2], achieves a 90% recognition average rate implementing recognition of 26 different hand postures. It uses a complex method to recognize the hand postures. It uses the color

947

image and performs several transformations in its color space to detect the skin color pixels. Then, it detects the BLOBs from them and selects the appropriate BLOB containing the hand posture. Then there is a series of steps being executed to process the hand posture BLOB and classify it further. This paper uses Hausdorff distance algorithm for comparing the processed input image with the database and calculating the distance between them.

The second approach that we studied was from the paper titled 'An Image Processing Technique for the Translation of ASL Finger-Spelling to Digital Audio or Text' [3]. This paper used a different approach with Adaptive Statistical Database which evolves and adapts the changes as the input comes along. It extracts the image from the video stream, eliminates its background using RGB filtering and thresholding. Then, it performs edge detection on the binary image and crops the hand posture part from it. Then, it resizes the image to a standard size and then uses error matrix to calculate error among all the images in the database and the processed captured image.

The third paper which we studied used a special range camera to detect the hand gestures. Though we do not intend to use any such special camera device with our approach, we studied this paper to understand various other ways of implementing hand gesture detection for sign language translation. This paper, titled 'Real time Hand Gesture Recognition using a Range Camera' [4], used highly complex mechanism with many constraints and limitations that we do not intend to observe in our implementation method. So, we have not considered any part of this paper suitable for our study.

The fourth approach that we have studied was from the paper titled 'Translation of Sign Language Finger-Spelling to Text using Image Processing' [5]. This paper uses hand-posture detection method to detect sign language finger-spellings and translate them successfully with best results. First it captures video using an internal or external webcam. The video is captured and the current image frame from that video after every 4 seconds is extracted. The 4 second delay is the time frame for the user to be able to change the finger-spelling gesture. This extracted image is processed to extract the feature content from it. This processed image is then compared with the statistical database of all the finger-spellings. The processed image is then compared to every image in the database and the error between them is calculated. The image with minimum error or the image that matches the input image the most is considered. The corresponding alphabet to the matched image is considered and is displayed to the user for the finger-spelling performed.

III. CONSTRAINTS

The approaches previously discussed have many problems.

The background of every image that is captured should be of a distinctively different color (preferably black) which can be easily differentiated.

The finger-spelling should not contain any kind of movement. Hence, it does not support finger spellings of alphabets J and Z as they contain movement of hand in their finger-spellings.

The image should consist of only the hand without any accessories or any other body part being visible.

It is less accurate and contains many errors and lighting also creates many problems during image processing.

IV. PROPOSED SYSTEM

Our proposed system overcomes all the constraints of existing approaches. The system architecture of our proposed system consists of hardware module and software module as shown in the figure 2.
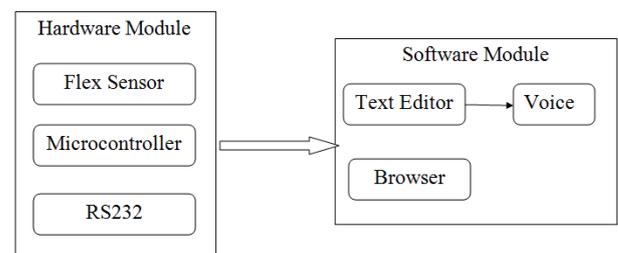

Figure. 2 System Architecture.

The user performs some actions in the form of hand gestures, it is given as input to the flex sensors which gives low voltage analog signal as output. This low voltage analog signal is processed by the LM358 comparator and the low voltage analog signal is amplified. The amplified analog signal is given to A/D converter where the analog signal is converted into digital signal. This digital signal voltage is verified if the value of the voltage matches to any of the voltage present in the microcontroller program then the signal is processed by the microcontroller program to generate the alphabet corresponding to TTL voltage format. If the signal verification fails the input has to be provided once again. Since the system does not understand the TTL logic it has to be converted into serial voltage. This can be done by max232 IC. After conversion of TTL to serial voltage, the data bytes are transmitted bit by bit to the computer using serial to USB converter. After this, data bits from the USB is extracted by the text editor and displayed on the screen along with voice output.

The software interface also provides a change option, if it is selected it changes the mode from text editor to browser where we can give input to browser using sensor glove. If this change option is not taken then there is no change in the mode. After displaying the text along with voice as output, text editor terminates. Similarly after returning the desired web page, the browser also terminates.

A. Hardware Module

The hardware module in our proposed system focuses on designing the sensor glove which comprises of many different components. Based on the usage of these components hardware module is divided into four different stages.

948

1) Sensing stage: Sensing stage consists of flex sens--or. Flex sensor is as shown in figure 3, it changes the resistance depending on the amount of bend on the sensor. They convert the change in bend to electrical resistance – the more the bend, the more the resistance value. They are usually in the form of thin strip from 1"-5" long that vary in resistance. They can be made as unidirectional or bi-directional. The sizes of flex sensor vary from 1kΩ to 20kΩ.
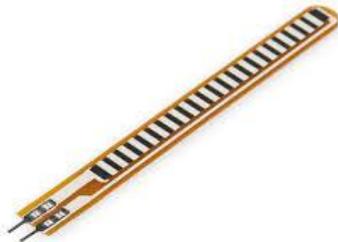

Figure. 3 Flex Sensor.

Flex Sensors are analog resistors. These resistors work as variable analog voltage divider. Inside the flex sensor are carbon resistive elements within flexible substrate [6]. More carbon means less resistance. When the substrate is bent the sensor produces resistance output relative to the bend radius. The flex sensor achieves great form factor on a thin flexible substrate. When the substrate is bent, the sensor produces a resistance output correlated to the bend radius as shown in the figure 4. Smaller the radius higher will be the resistance value. By using this resistance value, the hand gestures made by the user are recognized.
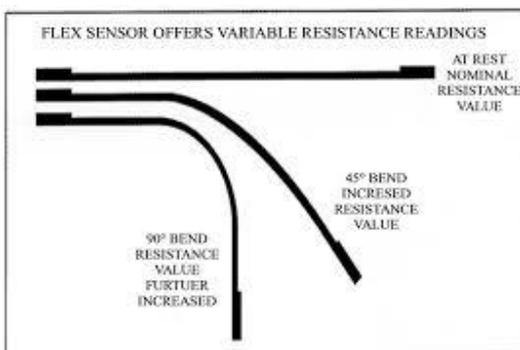

Figure. 4 Flex sensor bends proportional to varying degree of resistance.

The output from the sensing stage is given to the signal conditioning stage for further processing.

2) Signal Conditioning Stage: Signal conditioning is the process of manipulating an analog signal from the flex sensor in such a way that it meets the requirements of the next stage for further processing. It includes amplification to increase power of a signal, filtering to filter noise from the signal, isolation to isolate possible sources of signal perturbations and any other processes required to make sensor output suitable for processing after conditioning. In order to perform signal conditioning, LM358 comparator is used.

LM358 comparator is a voltage operational amplifier which is as shown in figure 5. It consists of two independent, high-gain, frequency-compensated operational amplifiers that were designed specifically to operate from a single supply over a wide range of voltages [7].

Figure. 5 LM358 Comparator.

The output from signal conditioning stage is given to processing stage.

3) Processing Stage: Processing Stage involves three steps. Firstly, Analog to Digital converter (ADC) which is an electronic integrated circuit. It converts analog signal to digital signal as shown in figure 6. It is needed because microcontrollers can only perform complex processing on digitized signals and not on analog signals. When signals are in digital form, they are less susceptible to the deleterious effects of additive noise [8].
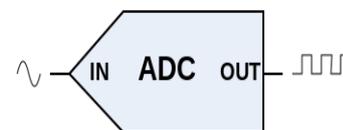

Figure. 6 Electrical symbol of ADC.

Secondly, PIC16F877A microcontroller is programmed to convert the digital signal voltage to TTL voltage for the corresponding alphabet.

The Figure 7 shows the control flow of processing the digital signals. The input to the microcontroller is provided by five flex sensors which indicate the movement of the finger. This input in the form of low voltage analog signal is amplified by the comparator. The amplified analog signals are given to the five adc channels of the microcontroller. These signals are then converted into digital signals. The digital signal values that are obtained are compared with the range of calibration values. If the value lies between the calibration ranges, the corresponding letter in character string format is sent to MAX232. If the digital signal value does not lie in any of the calibration ranges then this value is discarded and the input is taken again. The mikroC pro compiler is used to write and compile this program into hexadecimal file. This hexadecimal file is imported onto the PICkit2 programmer application/debugger which is a development tool with an easy to use interface for programming and debugging micro chips flash families of microcontrollers [9]. By using this programmer application

949

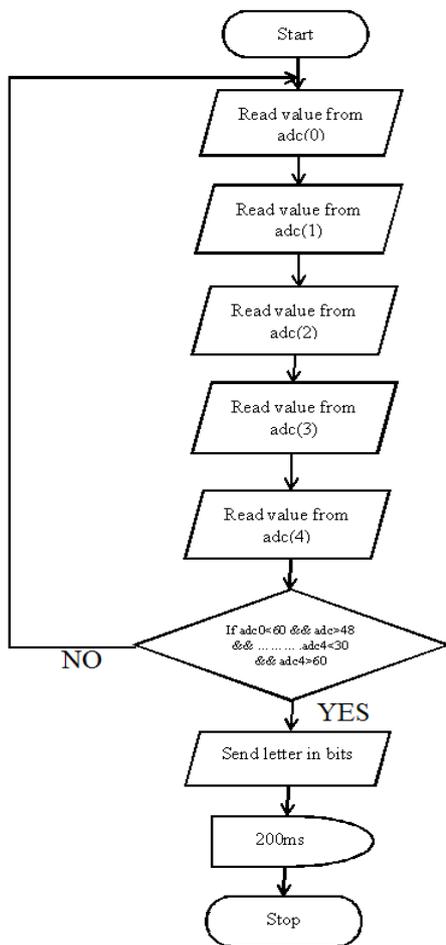we flash the program which is compiled by the mikroC pro compiler on to the microcontroller as shown in figure 8.

Figure. 7 Flow chart for conversion of digital signals to letters.

Figure. 8 Flashing the program into Microcontroller.

The incoming TTL voltage from the microcontroller has to be converted into serial voltage. This can be done by using max232 IC which is as shown in figure 9.

Figure. 9 MAX232 IC.

Max 232 acts as a buffer driver for the processor. It accepts the standard digital logic values of 0 and 5 volts and converts them to the RS232 standard of +10 & -10 volts [7]. The microcontrollers give a 0 to 5V output and require a intermediate buffer circuit to convert the 0 to 5 volts to +10 and -10V required by the RS232 port.

4) *Communication Stage*: Device Communication is a piece of equipment or hardware designed to move information or data from one place to another. We use RS232 and serial to USB converter for serial communication. The interfacing of MAX232 with RS232 is as shown in figure 10.
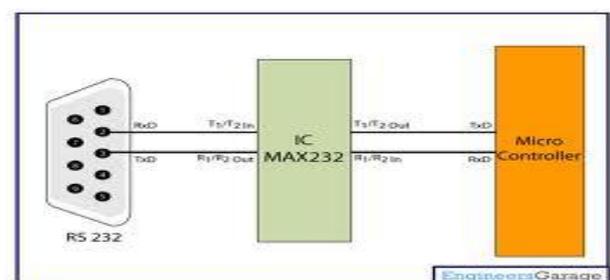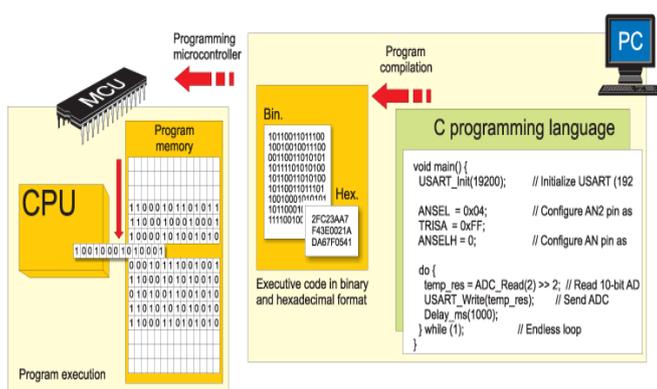

Figure. 10 Interfacing of MAX232 with RS232.

The sensor glove that we have designed is as shown in figure 11.

Figure. 11 Sensor Glove.

The input sent from the sensor glove is given to the serial port present on the system. The data from the glove is stored in the input buffer of the serial port.

950

B. Software Module

The Software module of Sign Language Recognition System consists of two modes: text editor mode and web browser mode. The user can switch from one mode to another mode using predefined hand gesture stored in the microcontroller.

The application that we have designed displays the current mode in which it is present, port data that is the current data or an alphabet that is sent from the sensor glove to the system, instruction mode that is it concatenates the previous data with the current data and checks whether there is a command defined for it, if this is true the desired action will be carried out. It consists of two buttons to control the entire application that is start button to start the application and stop button to close the application. It consists of an additional talk button which is enabled when the user is in editor mode. It also consists of a URL bar and a go button which is enabled when the user is in browser mode.

Before starting the application first we should verify that if the hardware device is connected to the system or not. If the device is connected, calibration has to be done as shown in table 1.

Gesture	Description
	Hold all the fingers straight and upward to set the maximum value for each finger.
	Fold all the fingers except thumb to set the minimum value.
	Fold only the thumb finger to set the thumb Calibration value

Table. 1 Calibration Gestures.

Once the calibration is done, to start the application gesture 7 has to be made. Once this is done, the start button is pressed and the color changes to green and the default editor mode is activated. Then a thread is started which extracts the data bytes from the serial port and displays it on the screen. The figure 12 shows how the data is read from the serial port and displayed on the text editor screen.

Initially the input is sent from the sensor glove to the serial port on the system. If the data terminal ready pin on the serial port is set to 1, then the data is stored in the input buffer. Otherwise, if the data terminal ready pin is set to 0, then the incoming data is discarded. Once the data is stored in the input buffer, the buffer size will be initialized

to SZ. And SZ is checked if it is greater than 0 and if the data is read within the time out period. If this condition is satisfied, the data bytes are read and it is converted in to ASCII value. After converting into ASCII value, the corresponding text is displayed on the screen. If this condition is not satisfied, then the data is discarded.
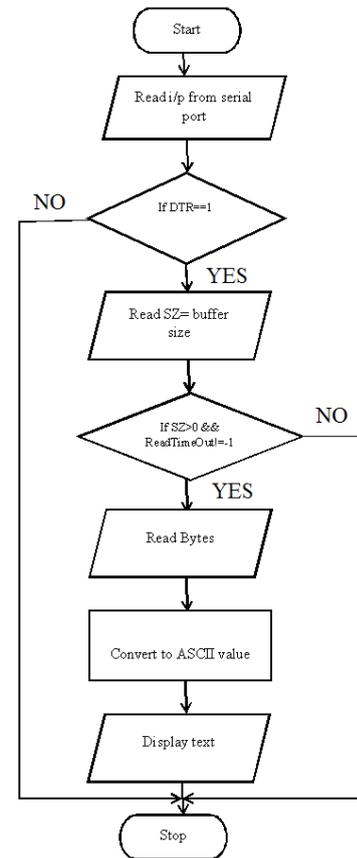

Figure. 12 Flow chart of text editor.

The commands that are enabled in the editor mode are as shown in the table 2.

Gesture	Action
7T	Text Editor Mode
7A	Voice
78	Dot
8B	Space
8D	Backspace

Table. 2 Text Editor Commands.

When the gesture 7E is made the talk button is pressed and the text displayed on the text editor is extracted and converted into corresponding voice.

Text-to-speech technology is used in this proposed system which makes the computer speak. The TTS gets the text as an input and then a computer algorithm called TTS

engine analyzes the text, pre-processes the text and synthesizes the speech with some mathematical models. The TTS engine usually gives kind of sound data like wave, mp3 etc as an output. There are two major modules in a TTS system as shown in Figure 13.
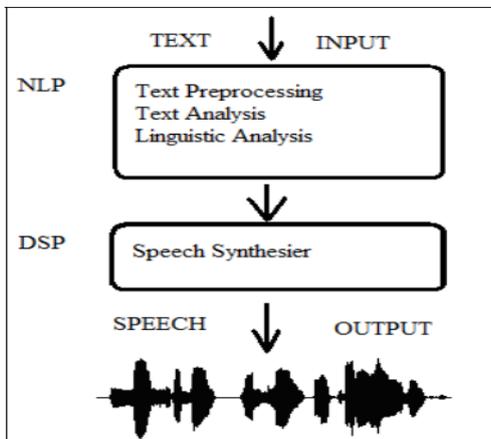

Figure. 13 Two modules of TTS system.

One of them is Natural Language Processing (NLP) Module that converts the written text input in phonetic transcription [10]. NLP module is also called as text processing module. The task of the text processing module is to process the input plain text and to convert it to the pronunciation form, usually represented by a sequence of phones and prosodic marks. A normal text contains many abbreviations, acronyms, special characters. So they must be processed to convert them to segments called words or standard text format. Combination of phonemes gives rise to next higher unit called syllables which is one of the most important units of a language. In most languages, the sequence of phones carries information about what speech shall be synthesized. On the other hand, the prosodic mark specifies how the speech shall be produced.

The other one is Digital Signal Processing (DSP) Module that transforms the symbolic information into speech. It is also called as speech production module. This synthesizer is the actual device which generates the output sound from information provided by NLP module in some format, for example in phonetic representation

The speech can be synthesized by concatenating different pieces of recorded speech from the database. Various methods can be employed for synthesized speech. Speech can be synthesized using rule based, concatenation methods. During speech synthesizes various parameters must be considered for the speech like frequency, pitch, stress, intonation, noise levels. These parameters vary over time to create a wave of speech.

The principle of concatenative synthesis is based on the assumption that continuous human speech could be divided into smaller segments. These segments are stored in a speech segment database (SSD). During synthesis, the segments are picked up from the database and smoothly concatenated.

The information about the pronunciation of the utterance to-be-synthesized (namely the sequence of phones plus prosodic marks) appears at the input of the speech

production module [11]. The module processes it and as a result, the output speech that matches the pronunciation information is produced as shown in figure 14.
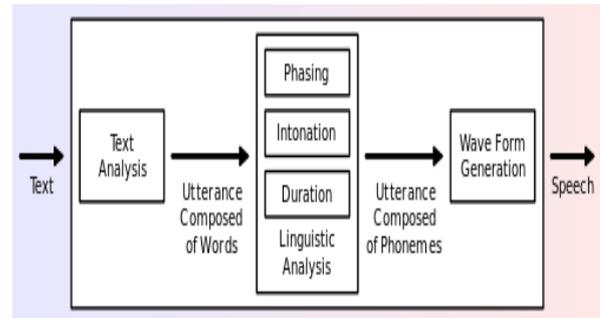

Figure.14 Block diagram of speech synthesis.

Web Browser is a software application for retrieving, presenting and traversing information resource on the World Wide Web. This Browser is designed for physically challenged people to browse and view desired Web Pages.

The Web Browser in the proposed system is completely enabled through the sensor glove and all the operations in the web browser is gesture enabled. The web browser mode is activated by the hand gesture 7B and the web browser tab is displayed. The commands that are enabled in the browser mode are as shown in table 3.

Gesture	Action
7B	Browser Mode
78	Dot
8B	Space
8D	Backspace
7F	Clear
7I	Forward
7J	Backward
7L	Reload
7M	Go
7Y	History open
7N	History close
8I	Scroll up
8J	Scroll Down
8A	Scroll Right
8F	Scroll Left
8Y	Click
7Z	Close
87	Tab
8M	Cursor Right
8N	Cursor Left
8L	Cursor Up
8T	Cursor Down

Table. 3 Web Browser Commands.

In order to type the content or text into the web pages another mode called browser editor mode is provided which is enabled when the gesture 8Z is performed inside the browser mode. The commands that are enabled in the browser editor mode are as shown in the table 4.

Gesture	Action
8Z	Browser Editor Mode
7B	Browser Editor Exit
8	Enter

Table. 4 Browser Editor Commands.

The figure 15 shows how the web page is displayed. The data from the input buffer is extracted and converted into ASCII value and displayed on the URL address bar of the browser. Once the URL address is entered into the bar, the http request is processed and the URI information is extracted like domain name, port number, address etc. Then this information is used to check the validity of the URL. If the given URL is valid, then connection is established between web browser and server. And the desired web page is displayed on the web browser window. If the URL is invalid the http request is discarded.
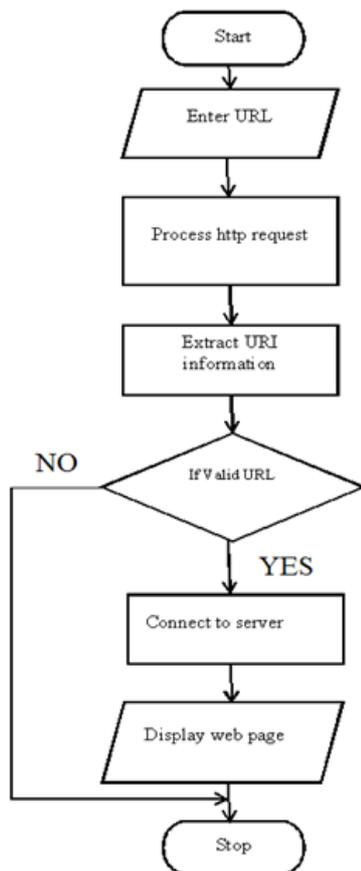

Figure. 15 Flow chart of displaying web page.

V. CONCLUSION

With the proposed system, we found that the sensor glove that we have designed overcomes the disadvantages of using webcam for gesture recognition. This sensor glove can be used by deaf and dumb people in their daily life more effectively to communicate and convey their thoughts and views to normal people in an efficient manner. It provides the result more accurately.

REFERENCES

[1]   Research paper on sign language recognition for deaf and dumb (Sept 2013 International journal of advanced research on computer science and software engineering).
[2]   Elena Sánchez-Nielsen, Luis Antón-Canalís and Mario Hernández -Tejera, Journal of WSCG, Vol.12, No.1-3, ISSN 1213-6972. Hand Gesture Recognition for Human-Machine Interaction.
[3]   Chance M. Glenn, Divya Mandloi, Kanthi Sarella, and Muhammed Lonon, An Image Processing Technique for the Translation of ASL Finger-Spelling to Digital Audio or Text. The Laboratory for Advanced Communications Technology / CASCI ECTET Department / CAST Rochester Institute of Technology Rochester, New York 14623.
[4]   Zhi Li, Ray Jarvis. Real time Hand Gesture Recognition using a Range Camera. Monash University, Wellington Road Clayton, Victoria AUSTRALIA.
[5]   Krishna Modi , Amrita More, International Journal of Computer Applications (0975 – 8887) Volume 77 – No.11, September 2013. Translation of Sign Language Finger-Spelling to Text using Image Processing.
[6]   Flex Sensor, mech207.engr.scu.edu.
[7]   MAX232, RS232, LM358 Comparator, www.howstuffworks.com.
[8]   PIC16F877A microcontroller, www.mikroe.com.
[9]   PICKIT2 user guide, www.microchip.com.
[10]   A Step towards Making an Effective Text to speech Conversion System,Vol. 2, Issue 2, Mar-Apr 2013
[11]   Implementation of a text to speech system with machine learning algorithms in Turkish, 2012.