Volume 4, Issue 11, November 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Dragon Voice: Implementing Chirognomy Using Text and Speech Recognition Techniques

[1]Sukhwant Kaur, [2]Trishna Thrinath, [3]Sachin Jangir, [4]Dhruv Verma
[1]Assistant Professor, [2, 3, 4] Student
School of Engineering and IT, Manipal University, Dubai

Abstract*— *In today's world, where everything works on our fingertips, laptops, mobile phones, and similars are not a rare view. There has been a drastic increase in the statistics that approximately quantifies the usage of electronic gadgets worldwide and since 2000; the graph has never stopped growing [6]. These electronic miracles have grown to play a vital role in the fields of education and communication. Yet, we realize that there exist many beings that are deprived of these blessings. According to the statistics published by UNICEF, in the year1992, nearly 48 million children were born with physical and mental impairments [9]. Despite advances in education, in developing countries only less than five percent of disabled children are enrolled in schools. This is why disabled children are often at risk of abuse, exploitation and neglect. The intervention of community workers, government policies, philanthropist campaigns, etc. has also failed to bring in much of a change to the scenario in the past two decades. But at the same time, the growth of the Internet over the years was drastic and over whelming. Very often we come across cases where students enrol for online education due to lack of money or time. When this option exists, why can't it be extended to the disabled beings too? Our hunt for a convincing answer to this question has led the way to the creation of our application which will help the people who suffer from speech and hearing impairment to get access to the world of communication.

Key Words: Speech, Text, Words, Sentences, Video, Sign Language, Speech Recognition Engine

I. INTRODUCTION

In many parts of the world, we face a scarcity of teachers or institutes dedicated to enlighten the challenged souls. Moreover, the sympathetic approach of the judgmental society that has concreted its existence everywhere often tend to discourage the disabled people to try out something new in life. The best solution to this situation would be a user friendly application which itself can act as a tutor and provide help to the people in need to find their way to education.

In this paper, we are presenting an application called Dragon Voice which can recognize speech and text inputs and in turn display the sign languages along with the definition, synonyms and antonyms of the input. Through the development of our application, we believe, that it will be a boon for the less fortunate since it helps them to fill their little world with the joy of education and easy communication.

There are many websites and few applications in the industry that provide similar facilities but a drawback in all of them is that the input can only be a single word. Combinations of multiple words or sentences are not taken as an input in any of these ventures. Also many of these sign language converters are accessible only through their host websites. Our application - Dragon Voice rise to an appealing rescue in this scenario. Dragon Voice is a user-friendly application that can be used to train the mute students at the comforts of ones place of stay itself. This application can be downloaded easily to any Windows system and also it can take inputs which are larger than one word. Our application has the added feature that it shows all details related to the input unlike its similars in the industry. With the simple, yet attractive presentation and the user manual that is provided, the application stands out due to the fact that the user needs minimal prior technical knowledge to get accustomed to the working and usage.

Hence, we are proposing our application which will definitely help the people in need to achieve their goals of education and communication.

Our paper is organized as follows: Section II of the paper explains the working of the various modules of the application and how they are connected to each other. Section III guides us through the implementation and working of the application. Lastly,Section IV narrates the issues faced on completion of our application and hence the future scope of our application which ensures to further enhance the user experience.

II. DESIGN OF THE APPLICATION

The software is divided into three modules:-

- Presentation Module
- Speech to Sign Module
- Text to Sign Module

Fig. 1 shows the system overview of our application.

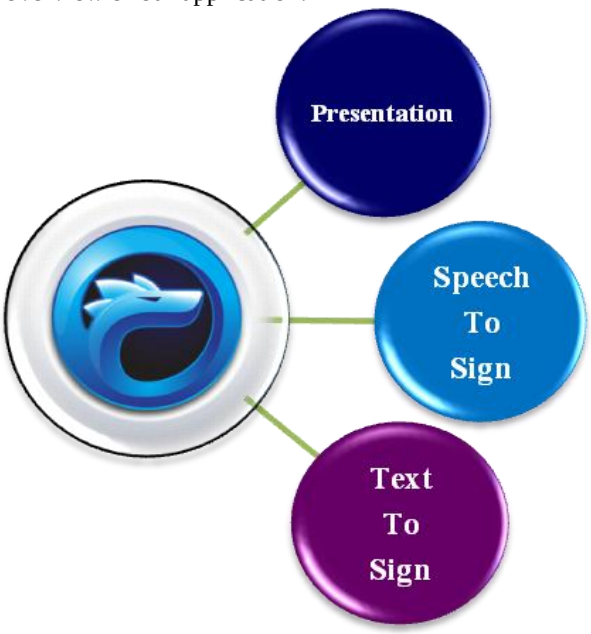

Fig.1: System Overview

A. Presentation Module

The presentation module played a vital role to help us design the user interface. Since our aim was to create a visually soothing interface, we decided that the application should have a colour combination of various shades of blue. Therefore, from Dragon Voice logo [Fig. 2] to backgrounds we have presented a simple yet attractive template deigned in blue.

Fig.2: Dragon Voice Logo

B. Speech to Sign Module

This module is the heart of our application. One of the biggest challenges we faced while developing this application was in deciding what method to adopt to recognize the input given as speech. Since this application was initially aimed to work on Windows platform, using namespace that contains Windows Desktop Speech technology techniques for implementing speech recognition was considered to be the best choice.

Windows Desktop Speech Technology software offers a basic speech recognition infrastructure that digitizes acoustical signals, and recovers words and speech elements from audio input. Our application use the System.Speech.Recognition namespace to access and extend this basic speech recognition technology by defining algorithms for identifying and acting on specific phrases or word patterns, and by managing the runtime behaviour of this speech infrastructure.

The instances of Speech Recognizer and Speech Recognition Engine supplied with Grammar objects provide the primary access to the speech recognition engines of the Windows Desktop Speech Technology [10]. Here, Grammar class provides run time objects that allow an application to specify a specific combination of words, choices of words, and other speech elements that the Speech platform uses to identify meaningful phrases.

We used the Speech Recognizer class to create our application that needed the speech recognition technology provided by Windows, which we could configure through the control panel as recommended by John Sharp [2]. This helped us to accept input through a device's default audio input mechanism.

We required more control over the configuration and type of recognition engine; thus, we built an application using SpeechRecognitionEngine, which runs in-process. Using the SpeechRecognitionEngine class, we could also dynamically select audio input from devices, files, or streams [11].

E.g.: SpeechRecognitionEngine srecognize = new SpeechRecognitionEngine ();

Once the application recognizes the input speech, it splits the received input into tokens (words) which are then searched on our online host website for corresponding sign language video [1]. The appropriate result video then gets displayed on screen dedicated for this purpose on the application.

C. Text to Sign Module

This module plays a very important role in the context that our application can be used for an all-rounder language development. Here, the words entered in the given textbox are taken as input. Later these inputs are converted to tokens followed by the same procedures carried out in section B.

The user is provided with an outstanding option where he can hear whatever input that has been entered. This helps to improve knowledge on the pronunciations of the inputs. This requirement is fulfilled with the SpeechSynthesizer class object [3].

The inputted words are stored in the buffer memory provided to the application by using a method call .text ().

E.g.: abcTextBox.text ();

The SpeechSynthesizer class' object with the help of the SpeakAsync method [4] that takes the object as an argument converts text to speech and through the speakers of the device allow the user to listen as to what input has been given.

E.g.: Object.SpeakAsync (textbox.text).

This module is also used when the user wants to give commands verbally to the device. Therefore, an interactive user experience is confirmed through this module.

The whole process of communication among the above said modules is shown in Fig. 3.

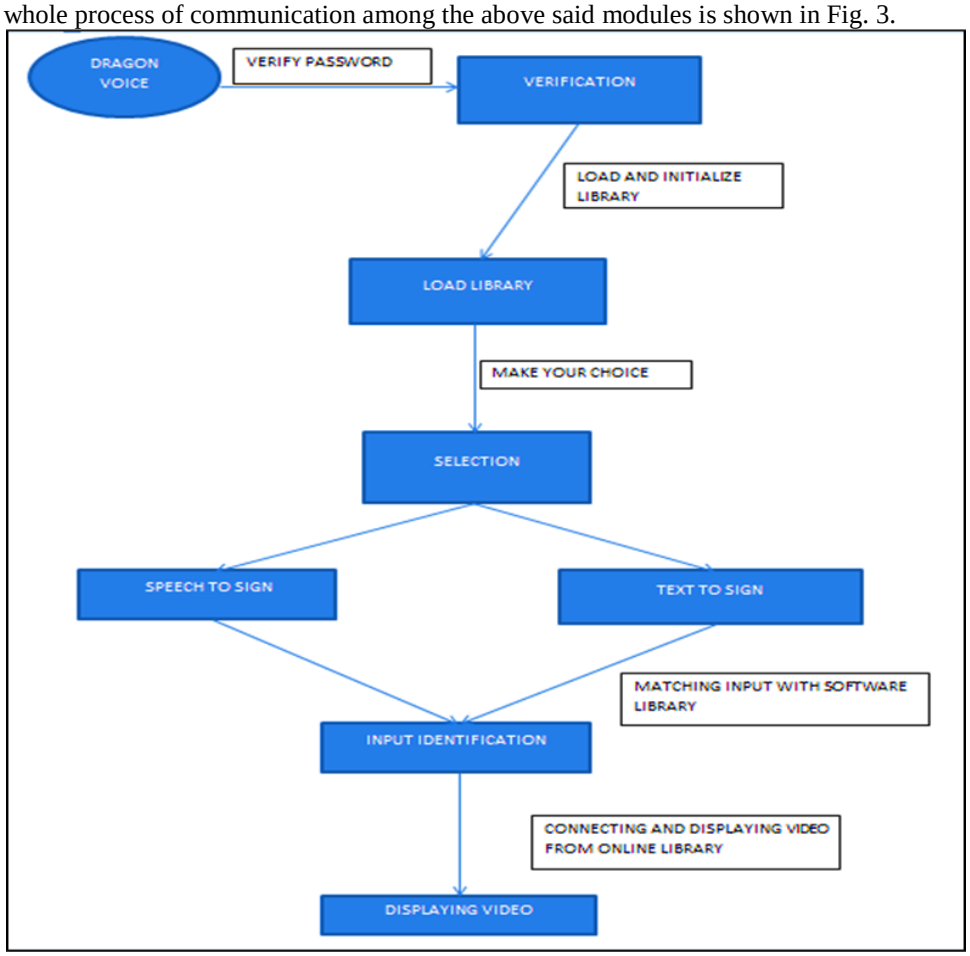

Fig.3: Software Design

III. IMPLEMENTATION AND RESULTS

The application is created on a C# platform [5]. It takes speech and any combinations of words as input. The output expected is the video depicting the given inputs in sign language. Dragon Voice uses an online website [7] which provides a reliable video library, well updated in the field of sign languages. The application works efficiently on any Windows platform be it laptop, mobile etc. Access to the Internet is also an important prerequisite for our application.

The front-end of the application is the user-interface where the inputs are accepted. This interface is designed to achieve a simple and attractive look that provides the user with a soothing viewing experience. The website from which the video is sourced, serves as the backend. The input is searched for in the website

A portion of the application is solely dedicated to store frequently used words so that the user can access them without the need for any buffering or access to internet connection. An option to clear this "history" is also provided [12].

Minimum requirement to run this application on any device is that it should have a RAM of 512MB and the operating system should be a Windows 7 or any of its newer versions.

Further in this section, we describe the working of our application – Dragon Voice.

As soon as the application is started, a loading screen appears. At this stage, all libraries used in the application is activated. Fig. 4 shows the loading status of the application.

Fig.4: Loading Screen

The user interface of the application is shown in Fig. 5. A text box that can be used to enter the input in the form of text is provided along with a button that can be pressed so that input in form of speech can also be entered. The result video is shown right below the text box. This video can be adjusted to be watched in any desired quality. It can also be viewed/replayed multiple number of times without buffering or re-entering the inputs.

This interface also provides a 'Clear' button that helps the user to refresh the application at any desired or unexpected situation. This action also clears the recently given input.

Fig.5: Main User Interface

As shown in Fig. 5, the interface also includes a library that stores all the frequently used sign videos. This is to provide an easy and quick access to the users. The user is granted the privilege to delete either all or particular saved videos from the list.

Our application was tested for various inputs. The application was found to be running correctly when an input size of one word was tested [Fig. 5].

The next test case was: "This is a demo." where a sentence of four words was given as input. In this case, the videos with the sign languages of all the four words are merged and displayed as a single video.

Fig.6: Result Screen displaying the first word

From the output we understood that the sign language for the first word is shown correctly without any errors [Fig. 6]. As soon as the video for the first word is shown, the sign language video for the second word [Fig.7] trails along.

Fig.7: Result Screen displaying the second word

Following the sign language display of the second word, the third word was recognized and displayed [Fig. 8].

Fig.8: Result Screen displaying the third word

Finally, when the fourth word was also displayed in sign language [Fig. 9], we were able to conclude that our application was working correctly for various test cases.

Fig.9: Result Screen displaying the fourth word

IV. FUTURE SCOPE

Among the intended modifications of the application, the most prominent ones are as follows:

A. Platform independence

On launching this application we realized that we could satisfy a larger number of users if our application could be run on various OS. Converting our application written in C# to a Java code would be the first step to the modification of the application. This is planned in order to achieve platform independence and in order to reach out to the aid of a larger number of users.

B. Signature Library

We encountered issues like low-resolution videos and incorrect sign language displays for a few inputs. To avoid this issue we reached a decision that rather than leasing an online video library, we intend to create a website completely dedicated to dragon voice where the user will be able to access the sign language video created by the team of dragon voice itself, while enjoying the benefits of the application on the website too. This way we could be sure of the quality and content of all the videos displayed.

C. Providing aid

During the testing phase of the application, we found that multiple words could have the same sign language actions. As expected, these were happening for synonyms. Hence, we plan to modify our application where every time the user gives an input, along with the video, a column will be displayed that will contain the definition, synonyms and antonyms of the words in the input. This, we believe will aid in widening the user's knowledge of the language.

D. Multi Linguistic

At present, our application provides excellent output when the language of input is English. Our next aim is to expand the scopes of the application so that it provides the user the freedom to use language of his choice. This surely would increase the number of customer beneficiaries.

E. Speech Recognition Options

This corresponds to our plan of modifying the application to make it platform independent which will require us to use a more general or overall speech recognition engine. We are at present, debating on whether Google's Siri will be able to fulfil our requirement or if there is a need to develop a speech recognition engine from the team's side. A convincing conclusion to this would decide the progress of further development of our application.

V. CONCLUSIONS

There are numerous educational applications which provide similar facilities when compared to our application [8], but Dragon Voice stand out due to its uniqueness in presentation and simplicity of operation.

The follow-up modifications intended to be executed on the application, gives a reason for a more exciting user experience that we hope to present at the earliest.

In this paper, our application - Dragon Voice is attempted with the purpose of understanding the basic issues involved in developing application for educational purposes. An easy to use and understand interface and high levels of abstraction is aimed in this application taking into account the wide variety of people that would be accessing it and their limitation to get used to a complex system. The application is successfully implemented on mobile phones, tablets, computers and it works efficiently on all devices that have Windows as its operating system.

ACKNOWLEDGMENT

The completion of our application and this paper would have been impossible without the help and support of few. We would like to thank our Prof. Sukhwant Kaur for her unflinching encouragement and motivation.

A big thank you goes out to our dear family and friends for believing in us and our ventures. Our achievements would be incomplete without their inspiring suggestions and advices.

Last but not the least; we bow in gratitude to Him, who makes it all possible.

REFERENCES

[1] Andrew Trolsen, *Pro C# and the .NET 4.5 Framework*, Apress, 6th Edition.
[2] John Sharp, *Microsoft Visual C# 2013 Step by Step*, Microsoft Press, Edition 2013.
[3] Jon Skeet, *C# in Depth*, Manning, 3rd Edition.
[4] Tom Archer and Andrew Whitechapel, *Inside C#*, Microsoft Press, 2nd Edition.
[5] F. Reena Sharma and Geethanjali Wasson, "Speech Recognition and Synthesis tool: Assistive Technology for Physically Disabled Person," *International Journal of Computer Science and Telecommunication* [Volume 3, Issue 4, April 2012]
[6] Use of electronic gadgets, [Online]. Available: http://www.forbes.com/forbes/2008/1110/108b.html
[7] Sign language video, [Online]. Available: http://www.handspeak.com/world

[8] Speech to sign language, [Online]. Available: https://www.google.ae/webhp?sourceid=chrome-instant&ion=1&espv=2&ie=UTF-8#q=speech%20to%20sign%20language
[9] Disabled children education statistics, [Online]. Available: http://www.unicef.org/education/files/vol2disabileng.pdf
[10] Speech Recognizer, [Online]. Available: http://www.msdn.Microsoft.com
[11] Tokens, [Online]. Available: http://www.codeproject.com
[12] Recent Videos, [Online]. Available: http://www.sympletech.com

