Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 7, July 2014, pg.291 – 299

RESEARCH ARTICLE

Vocal Mouse: Various Phases

Nibha, Vikram Nandal

M.Tech student, Department of CSE, R.N. College of Engineering & Management

Assistant Professor, Department of CSE, R.N. College of Engineering & Management

Nibhadua91@gmail.com, vikramcse@live.com

Abstract- Non-speech vocal sounds can be used along with command words as input to enable people (especially those with motor disabilities) to control computer interfaces effectively.
The Vocal Mouse will track both speech and non-speech vocal features including pitch, volume, and vowel quality in real time using audio signal processing. This application will enable the user to control the mouse pointer smoothly and continuously by vocalizing various vowel sounds corresponding to the desired direction of movement. Under conventional speech-driven pointer control, only spoken words as commands are used. The rate at which such parameters can be changed is also limited by the speed at which each command phrase can be uttered and recognized by the speech recognizer. Movement direction is specified by vowel sound vocalizations, in which different vowel sounds are mapped to each of the four cardinal directions. By varying the vowel sound and the volume continuously, the pointer's movement direction and speed can be smoothly controlled using the Vocal Mouse pointer control. This dissertation has combined the functionality offered by the Vocal Mouse with traditional speech recognition engines to explore ways to extend the capability of voice-based interaction with new user interface technologies.

Keywords: DP, HCI, IEEE, IVR, LPC, MPCC, VM

I. INTRODUCTION

Non-speech vocal sounds can be used along with command words as input to enable people (especially those with motor disabilities) to control computer interfaces effectively.

After a great amount of research on speech recognition technology from the past 10 years, significant improvements are achieved in its accuracy and capability. Some of the applications like interactive voice response (IVR) systems for automated call centers and telephone services have become popular. Various commercial products like Dragon Naturally Speaking from Nuance Communications, Inc., and Windows Speech Recognizer, are added to Windows

OS since Vista. The low cost of the hardware required for speech interaction (a microphone) also makes it an attractive solution for hands-free computer access.

It is important to enhance the expressivity of voice-based interaction due to number of reasons. People who are suffering from physical disabilities can't use standard input devices like keyboard and mouse, so they have only one option for gaining access to the computer i.e. hands-free input methods. In the United States alone, there are over 700,000 people with disabilities of the spinal cord, 70% of them are unemployed. These people have limited mobility and motor control so they have few options available to access computers, to obtain or retain employment, to stay connected with people and gather information around them. In the nut shell, such individuals are not able to show their creativity. These issues extend to people with other motor impairments as well, including the 46 million adults in the United States diagnosed with arthritis, the one million with Parkinson's disease, and the 50,000 children and adults with muscular dystrophy.

Enhanced voice-based interaction will also give benefit to people without motor impairments who find themselves in impairing situations. In situations like driving or interacting with a wall-sized display, hands-free interaction can be more suitable than traditional manual input devices. In desktop applications such as computer aided design tools that demand multiple dimensions and simultaneous channels of input, voice input can serve as an additional input modality to augment the standard keyboard and mouse interaction for greater control. One of the major limitations of the current voice-based input recognition method is that they can give input only in the form of words as commands. User's vocal utterances are processed at the word level, which results in discrete interaction. This discrete motion could not be used for performing tasks like scrolling, zooming etc which requires input in the continuous form for continuous motion. This limitation is being removed in Vocal Mouse as it works for both speech and non-speech sound inputs. So, it can produce both discrete and continuous motion depending upon the user task requirement.

II. METHODOLOGY

Methodology of constructing the proposed system will consists of various modules. Each module uses different techniques and algorithms to perform its specific tasks. After a particular module completes its task, its output will become input for the next module. In the end the combined effort of each module will be displayed. Flowchart for various modules of the proposed system is shown in figure 3.2.

Module 1:- Acoustic signal processing
Module 2:- Pattern recognition
Module 3:- Motion control

1. Signal Processing

The goal of the signal processing module is to extract low-level acoustic features using LPC technique that can be used in estimating the vocal characteristics. Process of feature extraction is shown in figure 3.3.The features to be extracted are:-
Energy
Normalized cross-correlation coefficients (NCCC)
Gain

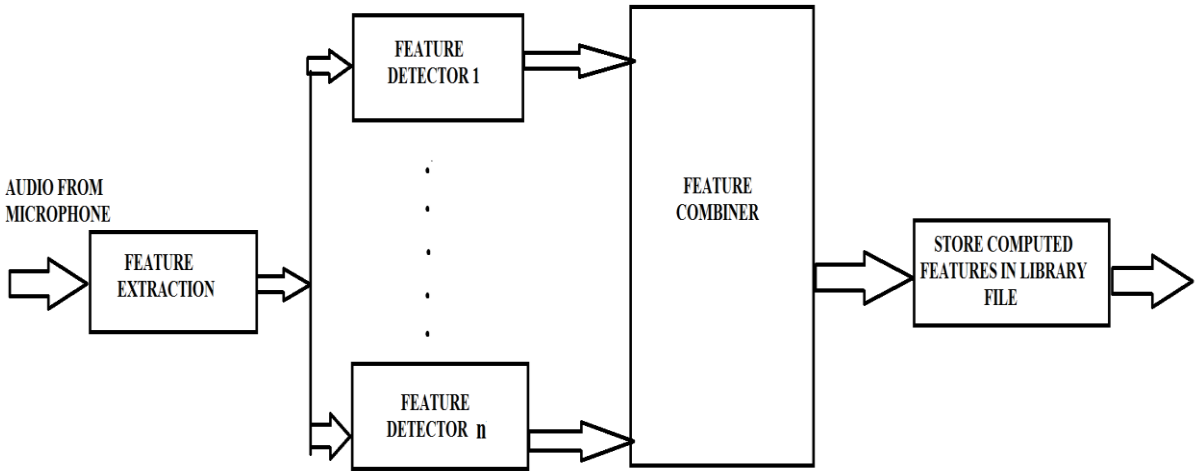

Fig 1: Process of feature extraction

For feature extraction, the speech signal is PCM sampled at a rate of Fs =10,000Hz. Energy is measured on a frame-by-frame basis with a frame size of 25ms and a frame step of 10ms. Pitch is extracted with a frame size of 40ms and a frame step of 10ms. Multiple pattern recognition tasks may share the same acoustic features. Therefore, it is more efficient to separate feature extraction from pattern recognition.
For the implementation of this module, get four words (up, down, left, right) from microphone and compute their features. Firstly, save the calculated features in a feature matrix and then store them in some other file. User is given 1 second to say each word. User will press enter and say the specified word in 1 second. Features of all the spoken words are stored in a feature matrix fw. Matrix fw is a 2-D matrix with 4 rows (one for each word) and 17 columns (17 features are extracted).

2. Pattern Recognition

This module uses the acoustic features to extract desired parameters. The estimation and classification system must simultaneously perform energy computation (available from the input), pitch tracking, vowel classification, and discrete sound recognition.

2.1 Approaches to be used:-

- **Minimum feature distance technique**

This proposed technique is based on calculating distances between the spoken word and each word in the library shown in figure 3.5.

- D=features of spoken word –features of word stored in library at training time.
- Sum up all the corresponding differences.
- Take the square root of the total calculated difference.
- Perform the above step 1, 2 and 3 calculations for each command in library.
- Above steps will result in four feature distance values (if number of commands stored in library are four)
- Print the word with minimum feature distance.
- The result will correspond to word spoken by user.

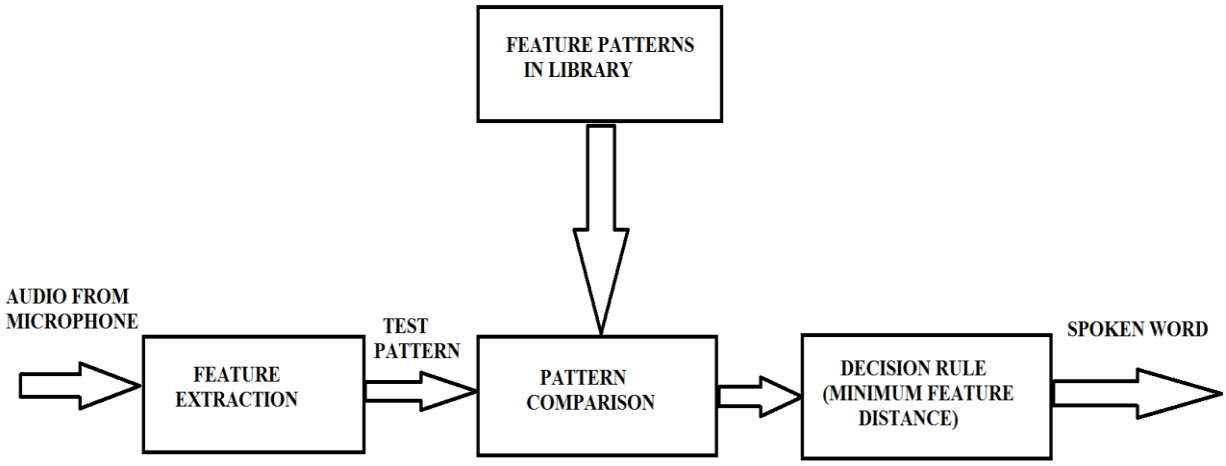

Fig 2: Process of pattern recognition

Example showing the working of proposed algorithm:-

- **Features of various commands stored in library in the form of a matrix:-**

Up◊	$a_1, a_2, a_3, a_4 \ldots\ldots\ldots\ldots a_n$
Down◊	$b_1, b_2, b_3, b_4 \ldots\ldots\ldots\ldots b_n$
Left◊	$c_1, c_2, c_3, c_4 \ldots\ldots\ldots\ldots c_n$
Right◊	$d_1, d_2, d_3, d_4 \ldots\ldots\ldots\ldots d_n$

- Features of spoken word:-
- $s_1, s_2, s_3, s_4 \ldots\ldots\ldots\ldots\ldots\ldots s_n$

- Calculate corresponding differences
- $s_1-a_1, s_2-a_2, s_3-a_3, \ldots\ldots\ldots\ldots\ldots, s_n-a_n$

- Sum up all the corresponding differences.
- $\text{Sum}(s_1-a_1, s_2-a_2, s_3-a_3, \ldots\ldots\ldots\ldots, s_n-a_n)$

- Take the square root of the total calculated difference.
- $D1=\text{Sqrt}(\text{Sum}(s_1-a_1, s_2-a_2, s_3-a_3, \ldots\ldots\ldots\ldots, s_n-a_n))$

- Perform the above calculations for down, left and right also.

- Result will be a set of four feature distance values
- D_1, D_2, D_3, D_4 e g:- (17.6334,15.5128,7.5142,22.9444)

Print the word with minimum feature distance.

3. **Motion** Word spoken by user is "left" because of its minimum feature distance as per taken sample values.

This module enhances the features of the system. System can also be trained to work for Punjabi (native language of Punjab) commands. Punjabi words such as ਉੱਪਰ,ਥੱਲੇ,ਖੱਬੇ, ਸੱਜੇ can be used by Punjabi system users instead of up, down, left, right commands respectively.

III. LINEAR PRIDICTIVE CODING

Reasons for choosing LPC for feature extraction

- LPC is a good speech signal model especially for the quasi steady state voiced regions of speech. In these regions LPC provides good approximation to the vocal tract spectral envelope.
- The way of applying LPC for analysis of speech results in the reasonable separation of source-vocal tract. Because of this representation of tract characteristics can be easily done.
- LPC is based on the idea that current speech sample can be closely approximated as a linear combination of past samples.
- LPC is an analytically tractable model. The method of LPC is simple and can be easily applied either to hardware or software. It is based on mathematical equations.
- LPC shows good results for recognition applications. Performance of LPC-based speech recognizers is better than speech recognizers based on filter-bank front ends.

Working of LPC algorithm is shown in figure 3.4.

Steps of LPC

- The basic idea of LPC is that speech signal at time n, s(n), is the linear combination of past k signals .

$S(n)=A_1.s(n-1) + A_2.s(n-2)+……..A_k. s(n-k)$ (1)

Where A_1, A_2, $A_3$are constant coefficients

$$S(n)=\sum_{i=1}^{k} A_i. s(n-i) + G\, u(n) \quad (2)$$

Where
G is the gain of the excitation
u(n) is the normalized excitation

Expressing the above in z-domain, we get

$$S(z)= \sum_{i=1}^{k} A_i. z^{(-i)}. S(z) + G\, u(z) \quad (3)$$

Transfer function is

$$H(z) = S(z) / G\, u(z)$$
$$= 1/(1-\sum A_i. z^{(-i)})$$
$$= 1/A(z)$$

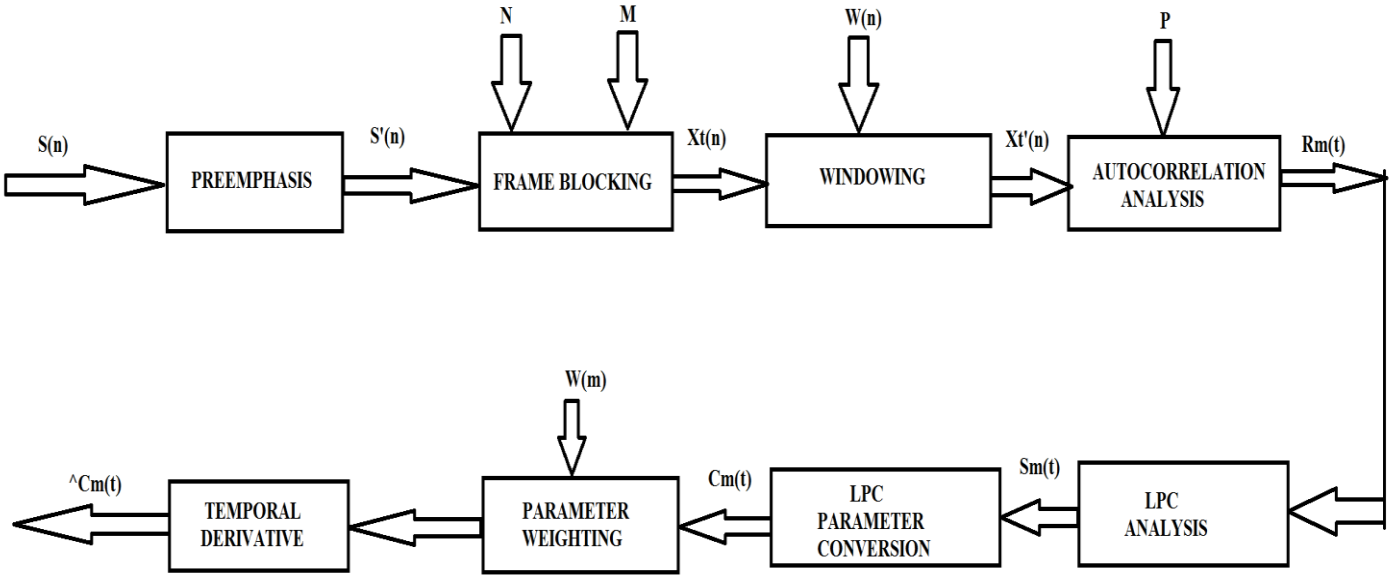

Fig. 3: Block diagram of LPC (Linear Predictive Coding)

Proposed Algorithm:-After describing the full detail of working of VM the outlined algorithm for the system looks like:-

1) Call the microphone working verification module and repeat this step until user does not close the application
2) Call the noise reduction module. Perform this step until the entire background noise is not getting removed.
3) Call the signal processing module and extract acoustic features like energy, MFCC etc. Perform this step only once.
4) Call pattern recognition module in which energy smoothing, pitch and format tracking and discrete sound recognition are performed. This module involves minimum feature distance technique.
6) Call motion control module to transform energy, pitch, vowel quality and discrete sound become acoustic parameters into direction, speed and other motion related parameters.
7) Call application driver which will take motion control parameters to launch corresponding actions.
End of algorithm.

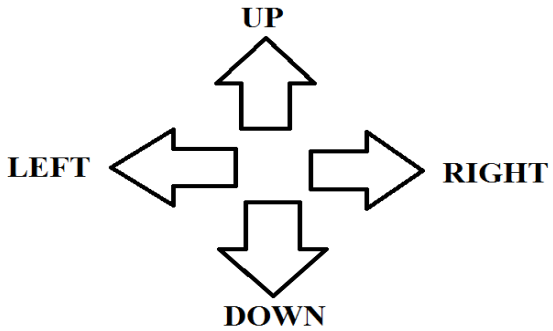

Fig 4:- Proposed algorithm will move the mouse pointer in four directions.

IV. RESULT(MOUSE MOVEMENT)

Module 1

Training Phase:-

Module 2 of Vocal Mouse project starts with training phase. Get four words (top-left, top-right, bottom-right, bottom-left) from microphone and compute their features. Firstly, save the calculated features in a feature matrix and then store them in some other file. User is given 1 second to say each word. User will press enter and say the specified word in 1 second. Features of all the spoken words are stored in a feature matrix fw shown in . Matrix fw is a 2-D matrix with 4 rows (one for each word) and 17 columns (17 features are extracted).

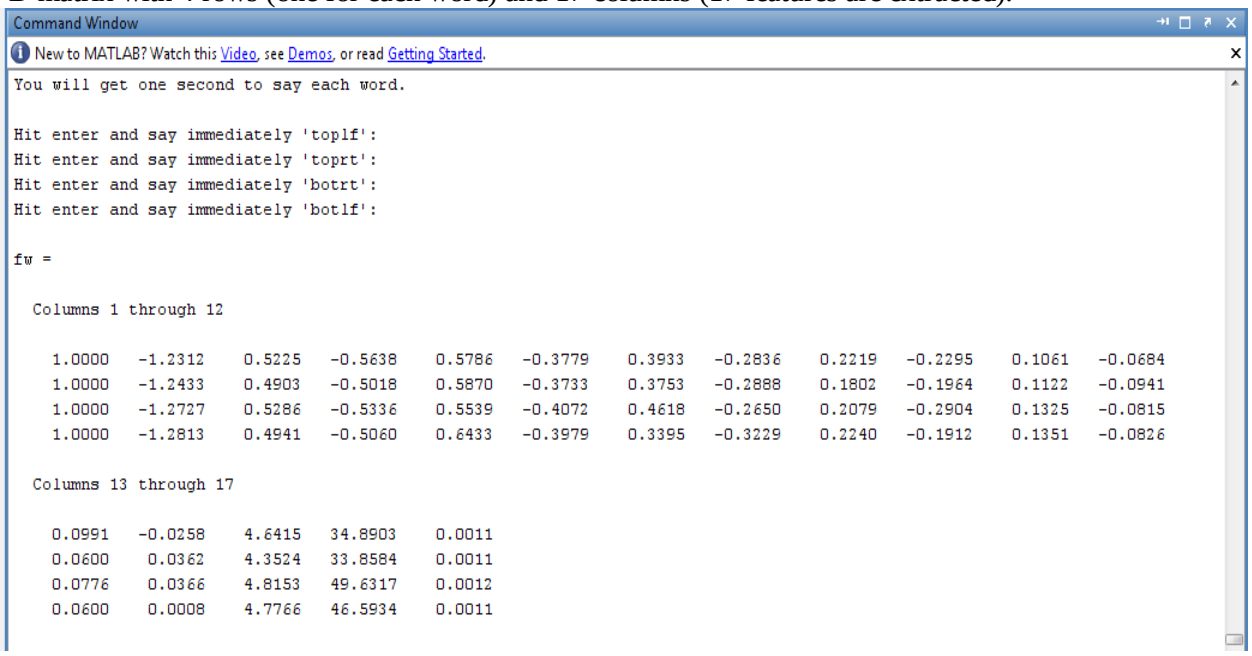

Fig 5: Feature matrix (fw) containing acoustic features for top-left, top-right, bottom-right, bottom-left

Contents of this matrix can be loaded to library data file for further reference shown in figure. This is the completion of training phase.

Testing Phase:-

During the testing phase, User is given 1 second to say any word. User will press enter and say the needed command word in 1 second. Features of the spoken words are computed and these features are compared with feature patterns already stored in library during training time. For this comparison, minimum feature distance approach is used. Vocal Mouse system will compute resulting spoken word on the basis of minimum distance. Output will be displayed on the console window of MATLAB giving "spoken word is:…". After the command recognition, system will perform task as per the requirement of the user.

Command:-"TOP-RIGHT"

User is given 1 second to say any word.
User will press enter and say the "TOP-RIGHT" command word in 1 second.

Features of "TOP-RIGHT" spoken word are computed and these features are compared with feature patterns of TOP-LEFT, TOP-RIGHT, BOTTOM-RIGHT, BOTTOM-LEFT that are already stored in library during training time.
For this comparison, minimum feature distance approach is used.
Vocal Mouse system will compute distances on the basis of minimum distance technique.
Calculated distances as per taken input sample are:-
TOP-LEFT◊D1=6.0274
TOP-RIGHT◊D2=4.9582
BOTTOM-RIGHT◊D3=20.6667
BOTTOM-LEFT◊D4=17.6346
Recognized word is "TOP-RIGHT" as top-right command is having minimum distance i.e. 4.9582
Output will be displayed on the console window of MATLAB giving "YOU SAID: TOPRT" along with FFT (Fast Fourier Transform) of "TOP-RIGHT" command as shown in figure 4.12.
After the command recognition, mouse pointer will start moving diagonally in the top-right direction

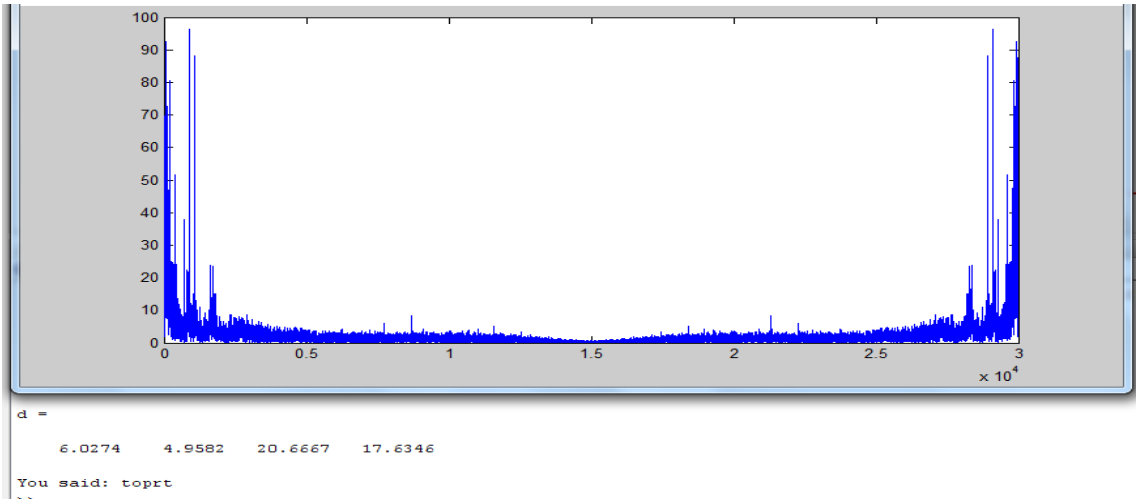

Fig 6: Fast Fourier Transform of spoken word (top-right)

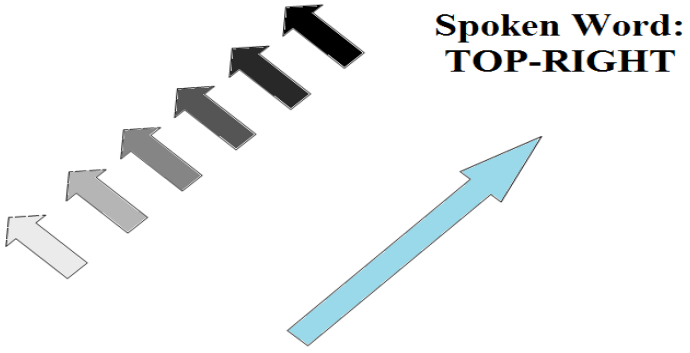

Fig 7: Resultant mouse movement with command "top-right"

V. FUTURE SCOPE

Implementation of the Vocal Mouse can be continued further and various kinds of analysis can be performed by recruiting participants, in particular those with various motor impairments, to try out our system. In future, vocal mouse can be used to develop applications such as drawing and games. These applications play an important role in enriching the lives of people especially those whose range of activities may be limited due to some disability. Vocal Mouse is well suited for such applications, and will be exploring ways in which the system can be best used to support them. Future planned improvements in the algorithms underlying the Vocal Mouse (to improve accuracy, user-independence, adaptation, and speed) will further increase the VM system's viability, and combined with practice could improve VM enough so that it becomes a reasonable alternative compared to a standard mouse's performance.

Enhancing Vocal Mouse Application

As more users express interest in the Voice Mouse, a number of enhancements need to be made to ensure that it offers as intuitive and effective a solution as possible.

One area of enhancement is in the visual feedback and the user interface for supporting self-diagnosis of recognition issues. A testing mode can be provided for the user to test whether or not the system is responding properly. More information should be conveyed to the user that would allow them to troubleshoot instances when the system fails to recognize certain inputs.

A method for automatically coaching the user to improve their vocalization, especially of the vowel sounds, will be beneficial. New module can be added to Voice Mouse which will provide video samples of each sound, real-time feedback during the testing mode as well as explicit suggestions to the user regarding how to change their mouth shape or other features to approach the desired sound.

The main obstacle and sources of frustration that a user faces is "false positives". False positive means system generates some recognition event when the user did not intend to vocalize. This could happen either when the user forgets that the system is processing vocal input and begins speaking or making some sounds, typically when the user's attention is away from the user interface, or when the system picks up some background noise and incorrectly recognizes it as some valid vocalization. In both cases, the user may not realize that the system has processed the false positive events until sometime later, e.g., when the user turns his attention back to the interface, at which point the user may become confused about what had happened, and possibly quite frustrated about not knowing what to undo if the exact series of actions that were inadvertently executed is not immediately apparent. In such a situation, quick method is required to disable current processing. A possibility for future work for addressing this issue is the use of various external contexts such as the user's "head posture" and "gaze" to disable voice input when the user is likely disengaged from the interface.

REFERENCES

1. Bohan, M., Thompson, S., Scarlett, D., Chaparro, (2003) "Gain and target size effects on cursor positioning time with a mouse", Human factors and ergonomics society, 47th annual meeting (pp. 737-740).
2. Dai, L., Goldman, R., Sears, A., Lozier, J. (2004)"Speech-based cursor control: a study of grid based solutions", 6th international ACM SIGACCESS conference, ASSETS 2004, Atlanta, Georgia, USA.
3. Feng, J., Sears, A., Karat, C. (2006) "A longitudinal evaluation of hands-free speech-based navigation during dictation", International Journal of Human-Computer Studies, 64(6), 553-569.
4. Igarashi, T., Hughes, J. F. (2001)"Voice as sound: using non-verbal voice input for interactive control", 14th annual ACM symposium, Orlando, Florida, USA.

