Navigo – Accessibility solutions for Cerebral Palsy Affected

Himanshu Pokhariya
DA-IICT, Gandhinagar
himanshu2004@gmail.com

Parag Kulkarni
DA-IICT, Gandhinagar
parag85@gmail.com

Vasudhara Kantroo
DA-IICT, Gandhinagar
vasudhara@gmail.com

Tanushree Jindal
DA-IICT, Gandhinagar
tanushree.jindal@gmail.com

Abstract

This paper presents the architecture of an integrated teachable interface, designed and implemented to provide accessibility solutions to the Cerebral Palsy patient in the virtual and urban space. It enables multimodal interaction between the Palsy user and computers, handhelds and remote controlled appliances for enhanced usability.

The idea is to redesign the system by changing the way we interpret the input, to suit the user's needs. The USP of the solution is that it is extremely dynamic, flexible and customizable to the entire range and levels of a complex condition as Cerebral Palsy. It works on the basis of allowing an individual to define simple variable inputs for himself and the system enabling him using the same by various specially designed tools, interfaces and feedback mechanisms.

Keywords

Human Computer Interaction, Cerebral Palsy, Accessibility Solutions

1. Introduction

This is an overview of a system which works as a comprehensive accessibility solution for Cerebral Palsy patient. The product name of this initiative is 'Navigo' and would be referred to as the same subsequently.

1.1 Target Audience – Cerebral Palsy patients

Cerebral Palsy is the most common childhood physical disability. It is a permanent physical condition, not a disease that affects movement. According to a new international consensus definition, "Cerebral Palsy (CP) describes a group of disorders of the development of movement and posture, causing activity limitations that are attributed to non-progressive disturbances that occurred in the developing fetal or infant brain. The motor disorders of cerebral Palsy are often accompanied by disturbances of sensation, cognition, communication, perception, and/or behavior, and/or by a seizure disorder" (Rosenbaum et al, 2005)".

Cerebral palsy is an umbrella term in that it refers to a group of different conditions. It is suggested that no two people with CP are alike even if they have the same diagnosis. It is a non-progressive disorder, so while on one hand, it usually doesn't worsen; on the other, the condition can never be significantly improved either [1]

A noteworthy point here would be that we need to know most Palsy patients are as intellectually sound as any of us. They are merely held back due to the limitations in the current interfaces and environments that have not been designed keeping in mind their specific needs and abilities. The patients are unable to use the conventional input devices such as the keyboard, mouse and voice recognition due to their problems of spasticity and dysarthria [4].

1.2 Present solutions – need for same

Various kinds of assistive technologies are available in the market. These include alternative input devices such as electronic pointing devices, force-feedback joysticks, trackballs, alternative keyboards and touch-screens that [7] have been designed to make use of the user's capabilities.

The major shortcomings with these devices are that they are either very need-specific, or have a very limited scope, or are too expensive. None of them offers a generic and complete solution for the various problems that a Cerebral Palsy affected person has. Navigo has attempted to achieve a universal inexpensive solution for Cerebral Palsy, keeping in the mind the audience in emerging economies.

1.3 Overview of the idea

Navigo has explored the power of software to alter the way we interact with existing hardware, thus keeping the cost of the system very low. The primary emphasis has been on being able to deduce meaning out of any kind of input that a user is able to provide the system with, and designing a universal but customized solution. This input could be a tap on the keyboard, a dysarthric utterance, or any kind of gesture representation convenient to the user. The system

International Conference on Computational Intelligence for Modelling
Control and Automation,and International Conference on
Intelligent Agents,Web Technologies and Internet Commerce (CIMCA-IAWTIC'06)
0-7695-2731-0/06 $20.00 © 2006 IEEE

dynamically adapts to the individual user's need, strengths and weaknesses.

For the virtual space, an integrated set of tools has been designed and developed to interface the Palsy user with this system enabled with personalized Multi Modal Interaction. For the urban space, we have designed a dysarthric-voice driven Universal remote-control for appliances. In the very near future, when every appliance would be remote control driven, if appliances' controls be made available to the CP user remotely through his own, so called 'unintelligible' voice, it would empower him immensely [2].

2. System Architecture

2.0 Diagram

The schematic of the system architecture is as represented in Diagram 1. The various aspects of the system have been explained in detail in the following sections.

2.1 Inputs to the system [Multi Modal Interaction]

Navigo is basically a software interface that allows for the users' interaction with a system or an environment in a way that any relevant data could be extracted from the user and meaningful information could be deduced from it. Therefore, it should work for as varied types of inputs as possible. In the first version of this software, these inputs have been limited to the following types:

2.1.1 Speaker dependent Phoneme Recognition
2.1.2 Movement Tracking
2.1.3 Image/Gesture Recognition

2.1.1 Speaker-dependent Phoneme Recognition. One of the primary inputs to the system would be the user's utterances. Since, a major portion of our target audience suffers from varying levels of Dysarthria; they cannot talk articulately, so normal voice recognition systems don't work for Palsy patients [6]. Hence the idea is to allow the user to provide phonemes or any kinds of simple sounds/ words that are convenient for the user to utter and train the system accordingly.

The system employs a speaker-dependent limited-vocabulary speech recognizer.

i. Keeping in mind the variation in a CP user, his voice is compared to his voice only
ii. Limited vocabulary to incorporate the user's disabilities and high degree of accuracy.

In the beginning, the user trains the system for his voice and his articulation of various phonemes. The system gauges the variation in the various utterances and therefore, the comfort level of the user in uttering them. Accordingly, it assigns meanings and contexts to each of them. For example, the utterance with minimum variation could be used most frequently by the user. Another significant aspect of the system is that it lets the user train itself for his utterance. So while a User A might pronounce the alphabet 'A' as 'aaa', a User B might pronounce it as 'ae' and still have the system work for him. Therefore the system is adaptive and flexible.

2.1.2 Movement Tracker. The CP affected have problems in fine-coordination of limbs, which hinders usage of the keyboard and the mouse. Some have problems of grip which renders the mouse unusable for them. Different patients have different degrees of fine and gross co-ordination. To cater to their varying abilities and disabilities, we have reconfigured the existing keyboard and mouse to the 'Tryboard' and the 'Glove Mouse'.

The Tryboard is the same old keyboard, reconfigured for our user. When the software is running, the keyboard instead of assigning actions to key-presses, would assign action to taps/hits on 'key-areas'. The Tryboard would be the keyboard divided into gross divisions (3-6), depending on the convenience of the user. Now, tapping a certain key area would have various contextual meanings attached. Say the keyboard is divided into three key-areas, on the desktop, these key-areas could signify 'Enter', 'Right Click' and 'Next' functionalities. Meanings of key taps would change contextually.

The other device is the 'Glove-Mouse'. The circuitry of an optical mouse is fitted on to a glove or a wrist band. Since Palsy patients cannot hold the normal mouse, the 'Glove Mouse' could simplify things for them.

2.1.3 Image/Gesture Recognition. Another kind of input that could be obtained from the user is hand-gestures or facial-gestures. Keeping in mind that phoneme-articulations may not suffice or at least might limit the user's capabilities, this engine has been incorporated. It works in the same fashion as the Sound Engine. The user trains the system for various personalized gestures and assigns meanings to them. The primary reason why this kind of an input mechanism would be suitable for this system is the fact that a lot of Palsy patients cannot utter a significant number of phonemes also. In addition, there are cases of quadriplegia (all four limbs paralyzed), paraplegia and diplegia (two limbs paralyzed) who cannot use the movement tracker efficiently. Since the kind of gesture for which a response is to be generated is chosen by the user, a convenient gesture – eyebrow movement, lip movement

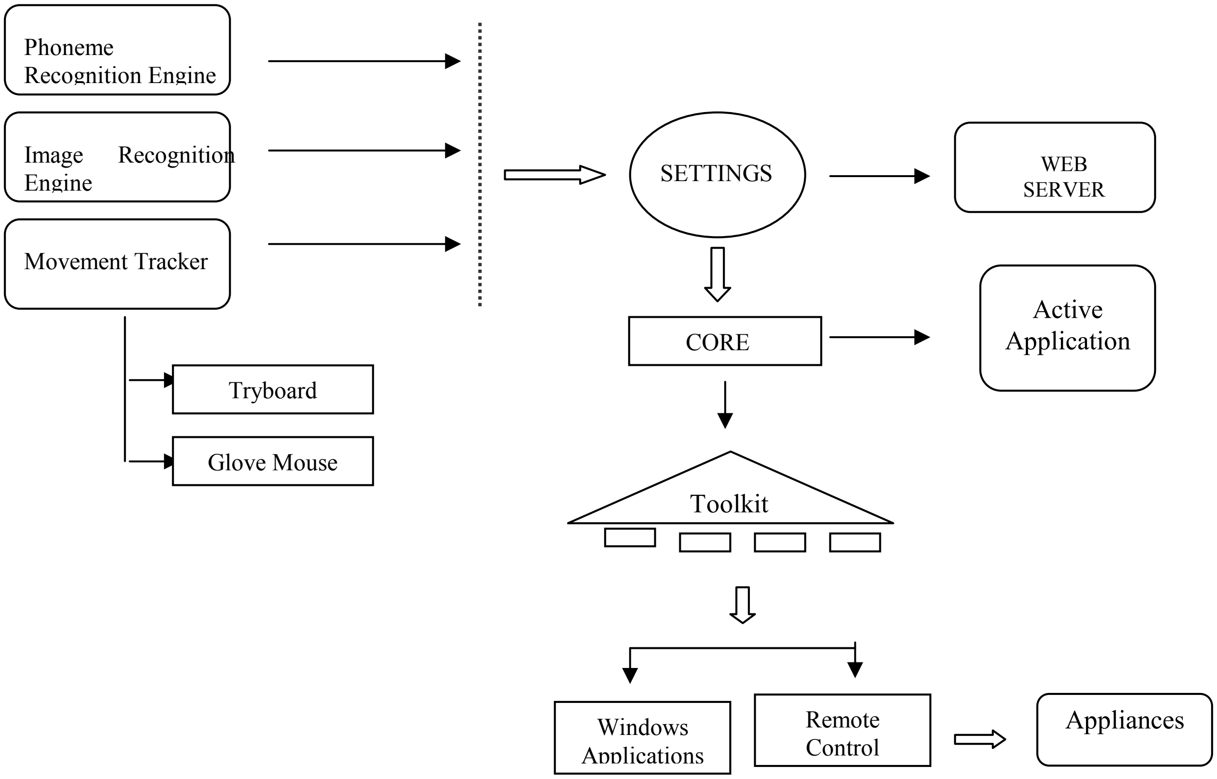

Diagram 1: System Architecture

could be assigned responses and the use of the computer made possible.

2.2 Settings for Dynamicity & Mobility [User Centered Model + Intelligent Environment]

The most unique part of the whole application 'Navigo' is that it adapts to the users' capabilities, and learns; hence allowing a great deal of flexibility. Every type of input received is assigned meanings dynamically and depending on the context.

Before using the system, it has to be trained and certain settings done as described in this section. The inputs from the Speech and Image Recognition Engine have been grouped into Alphabets, Numbers and Control words.

2.2.1 Alphabets. These can be further grouped to reduce the number of inputs required to be provided. The groups can be formed by the user or it can also be automated. The basis of clustering is frequency of use, similarity in the phonetic sound (e.g. a, e and i could be grouped together) and ease of pronunciation. Each group is assigned an input by the user (phoneme, gesture or both).

2.2.2 Phonetic Classification. There were various considerations for deciding the criteria for the phonetic classification. The various factors that were taken into account for the same included the following.

- The most frequently used alphabets like 'e', 't', 's', 'a' etc should be suitable grouped and placed so that there is minimum effort in typing or using them.
- The sounds or phonemes which less variation should correspond to the alphabets more frequently used.
- At the same time, the alphabets should be spread out such that all the alphabets with similar frequency in a language should not be grouped together.
- It is preferable if the default sound assignment to alphabet groups is such that it is intuitive for the user to recollect or use them

Keeping all these factors in mind, a system of phonetic classification would function for 'Navigo'. This would be utilized when the automatic grouping of alphabets is employed by the user.

2.2.3 Numbers. CP children are particularly good in relating with numbers. Various tools of the system utilize this capability of the user. During the settings, the user gives inputs for as many numbers as he can. The system functions for a minimum of 2 or 3, but is faster and more convenient for say, inputs for 10 numbers.

2.2.4 Control Words. These words are essential for

- Mode switching [The system has different modes defined, which would be introduced later]. Modes can be assigned individual inputs or all invoked through one Master Mode.

- Enabling shortcuts [Favorite actions of a user can be assigned shortcuts. These are to be used if the user has excess control words.]
- Frequently used functions [Enter, Back, Next]

A minimum of 5 is defined here. [A master mode, no shortcut and the rest would be frequently used functions] These are a part of the detailed design and to be empirically verified.

Thus every user can define his individual settings and on the whole, the system has the capability of functioning in infinite ways. This is done to cater to the entire spectrum of users, every severity level and disability. So, a user with just five distinct utterances (severely dysarthric) and/or gestures (severely spastic) could use the system as well as a user with fifteen utterances. The only difference would be in terms of the ease of use.

2.2.5 Tryboard Settings. In case of the keyboard usage, the primary consideration would be his fine-coordination and requirements. Again, while training the system, the user could choose whether he/she wants the keyboard to be divided into 3, 4, 5 or 6 areas in different contexts. This would permit the user to work on the system with his preference of granularity. A person with greater problem in fine coordination could choose to have just three key areas, while a person with better fine motor coordination could use six areas. The functionality of each area can be assigned and modified at run-time as per convenience.

2.3 Toolkit [Designing Teachable Interfaces]

Given the inputs, we now present the Navigo Toolkit. It comprises the following:

- Virtual Mouse
- Onscreen Keyboard
- Word Predictor
- Custom Tray

Each feature corresponds to a different invokable mode.

2.3.1 Virtual Mouse. A Navigo-input driven virtual mouse has been implemented whose design is as described below.

Firstly, the screen is divided into an n*n grid. Here n is the number of 'numbers' for which the user has defined his settings. Each grid block can be accessed by its row and column number. By giving the input (utterance/gesture) for a particular row and column in a specified time, the user can directly reach any block on the screen conveniently.

To navigate in various directions, the user's alphabet settings are used. In order to increase the granularity of this mouse, the user invokes the split function. Therefore, every bit of space on the screen is easily accessible.

2.3.2 Onscreen Keyboard. To incorporate for various keyboard functions and special characters that cannot be used, the onscreen keyboard is designed. Each key can be accessed using its row and column number combination or its group input (if it is an alphabet). The user can now use all the functions of a normal keyboard using his customized inputs [6].

2.3.3 Phoneme/ Word Predictor. This module helps the user type text, by suggesting common and previously used words as possible completions. This can work in conjunction with the various inputs (tryboard, speech, gesture). The words suggested depend on the current context, e.g. in the address-bar of a browser, the names of common/favorite websites will be suggested.

Following are the features of the predictor:

- Predicts phonemes, syllables and words (depending on the advancement of speech recognition system used.) based on context.
- Limited vocabulary : For simplicity and ease of use
- Learning capability : Words used often given priority and included in list
- Options numbered : Easier to communicate the choice
- Number of options listed limited and can be set
- For specific cases, commonly used characters and change mode options listed. [Usage simplified : Fewer rules to memorize]

The phoneme predictor would assist the user to form words by prompting the closest and most plausible options.These options could be availed of as each word would be numbered or by using the 'Next' option of the Tryboard in that mode.

2.3.4 Custom Tray. It is an invokable tray, quick to use with multiple functionalities. It contains a graphic list of all the running programs, an option for calling a new program, accessing the desktop. Each icon is also supported with its phoneme

An extension of the Custom-Tray serves as a display of the current utterance, current mode and the keyboard status. This is active feedback mechanism of the system.

2.3.5 Universal Remote [Ubiquitous System]. Enabling Cerebral Palsy Patients to easily navigate home environments as they navigate the computer environments is plausible by employing a similar principle. The idea being that if a Universal remote control; one remote control for all remote control enabled devices could be made phoneme or dysarthric voice driven, it would make all appliances operable for CP patients. This not only increases their ease and convenience, but also results in an increased sense of confidence and self-belief as well.

The voice driven remote would comprise of a Bluetooth enabled transceiver, a voice chip (e.g. HM2007 –a 40 word voice chip), and a multiplexer or microprocessor. The settings used by the software in computer navigation could be transferred to the remote control. Additional settings for the various appliances could be made in the software itself. These settings would translate into the users' preferences for navigation of the real-world environments. Therefore, using 'Navigo', the user could be empowered, especially considering the fact that most of the devices and appliances we use would soon be remote controlled.

2. Implementation Details

The application has been primarily built in .NET environment specifically for the Windows platform. The language used was C#. Inputs such as that of speech where extensive processing was required were obtained from Matlab and suitably interfaced with .NET.

3.1 Speech Engine

A speaker dependent isolated word Speech Recognizer has been implemented in Matlab. MFCC features are extracted using MelCepstrum Anlaysis [with 12 coefficients] for the training set. A Gaussian Mixture Model (GMM) is then designed and trained, and parameters estimated using Expectation Maximization algorithm of Dempster et al. Live input is taken through Matlab the probability of the particular input belonging to each cluster defined in GMM is measured and suitably analysed. This output is redirected to .NET through the interface MATLAB COM Builder.

3.2 Active Application Tracker

This module keeps track of the application which is currently active. This is helpful in determining the current context so that different modes can be implemented. The active application is determined using Win32 API calls.

3.3 TryBoard

Each zone of the TryBoard implements a particular functionality depending on the current context. Each key-press is captured using system-wide keyboard hooks and an event is raised (using Win32 API calls) depending on the zone to which to the key belongs. The system hooks prevent the normal applications from receiving original (undesired) inputs from the keyboard. The active application tracker (described above) is used to get the current context.

3.4 Windows Navigation

Using inputs acquired from various means (phoneme recognizer, tryboard, gesture recognizer), the appropriate Win32 API calls are made to allow the user to navigate the Windows Operating System. All the features of the windows GUI shell (explorer.exe) can be used

3.5 Image Processing Engine

Live video input is captured using the standardized Microsoft Win32 API, DirectShow. The gesture recognition module is in its initial implementation stage.

3.6 Toolkit

The toolkit has been implemented using Win32 API and C# programming. The mouse movements and key presses have been simulated using Win32 calls.

4. EMPIRICAL ANALYSIS

Profiles

User A – Severe [Severe dysarthria and quadriplegia]

Settings – Minimal

Sounds: Alphabets: 5, Numbers: 0, Controls: 1

Tryboard Areas: 3

Gestures: 3 [To be used for keyboard taps]

The one control word is the Master Mode and the rest of the modes can be invoked thereafter using alphabets. Alphabets replace numbers in every scenario. The mouse grids are not numbered and can only be used using the direction guide with maximum 5 directions. These directions can be set by the user during settings or dynamically. Typing is slow as the size of the alphabet groups is big and hence choosing the correct alphabet is a tedious task. Since the user is quadriplegic and can effectively use neither of his hands, he uses 3 distinct facial gestures to simulate keyboard taps.

User B – Moderate

Settings – Average CP User

Sounds: Alphabets: 12, Numbers: 6, Controls: 6

Tryboard Areas: 5-6 [Changed dynamically]

Gestures: Defined if required.

The Control Words can be assigned preferences by the user at the time of the settings. The system will ensure that the critical functions missed out would be included as keyboard functions. If involving gestures is a convenient option for the user, these can assign extra control words [e.g. shortcuts to favorite programs]

User C – Mild Dysarthria, Moderate motor co-ordination

Settings – Average CP User

Sounds: Alphabets: 26, Numbers: 10, Controls: 10

Tryboard Areas: 3

Gestures: Defined if required.

Since the user is mildly dysarthric, he can customize the system to his needs very well and maximize its performance. No grouping is required and typing is very efficient if used in conjunction with the word predictor. Effective usage of the virtual mouse with high degree of granularity is possible.

5. Uniqueness and Innovation

The uniqueness of Navigo stems from its ability to decipher mere sounds, hand-hits and gestures and assign meaning to them. It simplifies the basic input system to be able to extract as much information from the disabled user by means convenient to him. The USP of the application is that no new hardware is required to render the system usable. The current system has been redesigned and reconfigured (the keyboard) to suit the needs of the audience. This is especially suitable for the user groups in emerging economies [3].

The focus of design has been on keeping the model simple and consistent, ensuring that it is easy to learn and use. The movement tracker, the phoneme and image recognition systems work in tandem with one another to create a user-friendly experience.

5.1 Future Scope

The following is the scope of extending this initiative and making it a more effective solution.

- For further versions, this software could be made platform independent, so as to reach a wider audience.
- The various engines could be enhanced such that the sound engine would work for syllables as well, the image recognition system could work in tandem with sound engine to interpret lip-movement along with the sound utterance to lead to a more accurate and appropriate response.
- The remote control device could be upgraded to a camera enabled device in order to make it gesture-input driven as well.

5.2 Conclusion

This paper is an attempt to show that innovative solutions towards reconfiguration of a current system to suit needs of specific marginalized user groups could be effective in terms of economic viability, ease of usage, psychological uplift and simplicity of design.

6. References:

[1] Cerebral Palsy: Hope Through Research, NINDS. Publication date July 2006, NIH Publication No. 06-159

[2] Study protocol: SPARCLE--a multi-centre European study of the relationship of environment to participation and quality of life in children with cerebral palsy, BMC Public Health. April 2006

[3] Economic costs associated with mental retardation, cerebral palsy, hearing loss, and vision impairment --- United States, 2003. MMWR Morbidity and Mortality Weekly Report 2004;53(3):57-59.

[4] K.C.K. Kuban, and Alan Leviton , Cerebral Palsy, The New England Journal of Medicine,Volume 330:188-195, January 20, 1994 , Number 3

[5] Rupal Patel and Deb Roy, Teachable Interfaces for Individuals with Dysarthric Speech and Severe Physical Disabilities, MIT Media Lab, 2001

[6] Nitin Sawhney and Sean Wheeler, Using Phonological Context for Improved Recognition of Dysarthric Speech, Speech Interface Group, MIT Media Lab, 1999

[7] Assistive Technology Decision Tree by UnumProvident, Microsoft Accessibility Resources, Copyright 1999

[8] Fieldwork: Visited Polio Institute, BM Institute of Mental Health, VS Hospital at Ahmedabad, and Spastic School of Haryana.