Converting Computer Touch-pad into an Integrated Input Device

Ajinkya Jadhav
Student, Computer Department M.A.E., Alandi, Pune University, Pune

Animesh Rawat
Animesh Rawat
Student, Computer Department M.A.E., Alandi, Pune University, Pune

ABSTRACT
The application module named *TouchSwype* allows a user to type words into applications like Notepad or Microsoft Word without accessing the keyboard or mouse. Module *AbsTouch* bridges the gap between a touch-screen phone and the PC by allowing the user to use the touchpad as a touch-screen for a corresponding click on the screen. The touchpad acts as a scaled down version of the computer screen and makes the monitor a virtual touch-screen, free of cost. Opening any application demands a series of folder traversing or cluttering the desktop with "shortcuts". The third module in the application, aptly named *GestApp* is a gestural application launcher where you can gesture a predefined letter onto the touchpad to open any application of your choice. People with motor impairments, often cannot use a conventional mouse and keyboard. They may lack sufficient mobility to reach for these devices, or sufficient endurance to use them for more than a few minutes. Our application, Unistroke(the combination of TouchSwype, AbsTouch and GestApp), offers an innovative alternative for those people to control their desktop and enter text.

Keywords: TouchSwype, AbsTouch, Input Devices, Computer Access

1. INTRODUCTION

1.1 Motivation
This paper is motivated by two primary concerns, the need for better desktop computer access, particularly for people with motor impairments and the need for better data entry method, which helps disabled users and able-bodied users alike. Improving text entry and desktop access in these domains with a single versatile design is the primary goal of this research.

1.2 The Need for Better Computer Access
The value of access to computers and information technology has increased dramatically over the past decade. As the World Wide Web has grown, libraries, news sources, health and financial information, entertainment applications, and other resources have become increasingly available online. The Web's value has further increased with improved search. Other Internet applications like email and instant messaging have also grown in use and importance in keeping in touch with coworkers, friends, and relatives. In just over a decade, the Web has matured from a research curiosity to a factor contributing to one's very quality of life. For people who are home bound, computer-based entertainment may be an engaging alternative to watching television. Access to computers may also allow people to sustain relationships with their families and friends who are located elsewhere, providing communities for those who may otherwise be isolated.

But all of these activities depend crucially on good computer access. Although the importance of access has been recognized and numerous assisting technologies exist, providing access still remains a formidable challenge. Studies show that less than 60% of people who indicate a need for computer access devices actually use them. Furthermore, at least 35% of purchased solutions are never adopted.

Complex technologies in particular, like *voice recognition systems*, are subject to high abandonment rates.

Reasons cited for these failures include: High cost of devices, Device complexity, the need for extensive customization, the need for ongoing maintenance.

Additionally, the process of acquiring devices often proves prohibitively difficult and fails to incorporate enough input from the people for whom the devices are intended. Of prime importance, then, is simplicity in both the design of devices and in the process of adopting them. In being usable on readily available devices, **UNISTROKE** will provide simpler access solutions.

2. PURPOSE
Organizations have been coming up with solutions to text entry problems. But common desktop users have been deprived of these solutions due to various platform issues. Moreover, buying a touch-screen for the PC is an expensive affair for a normal user. Unistroke provides its users with a fully automated touch-pad which can help in One-touch desktop access, word-level typing and launching applications via gestures. Unistroke is the first such attempt at making the touch-pad an integrated input device. Unistroke opens a new platform in this area and further works can be carried out to enhance its features.

3. CURRENT WORK IN THIS AREA
Many devices exist for computer access. Alternative onscreen keyboards, head switches, sip-and-puff devices, voice recognition systems, and augmentative communication devices are few of the options available for computer access. But there can be obstacles to effective deployment. Many devices are prohibitively expensive. Others require extensive configuration or maintenance. Some might be unwieldy.

Touch-pad interaction techniques have been around for some time, but surprisingly few text entry techniques exist for touch-pads. None of these, however, is a generic touch-pad text entry technique like the one we are attempting to make. Most touch-pad techniques focus on control and selection tasks. Our aim in TouchSwype, by providing text entry techniques is to lower the barriers to computer access by using mechanisms already present. TouchSwype will depend on physical edges to provide high tactility and stability of motion. Similarly, one touch access has been there for years on smart phones, but it has not been used to control PCs. As mobile phones are small in size, touching the screen to control the User Interface on a cell phone had to happen one day. But with the over-reliance on mouse and arrow keys in PCs, the development of a better screen access method had become far-fetched. As for GestApp, something vaguely similar has been developed by Synaptics named Scrybe. It opens websites using touchpads which support multi-touch. This limits its functioning capacity to just one task.

4. DETAILS OF THE DESIGN

Unistroke module revolves around three modules namely Unistroke AbsTouch, Unistroke TouchSwype and Unistroke GestApp.

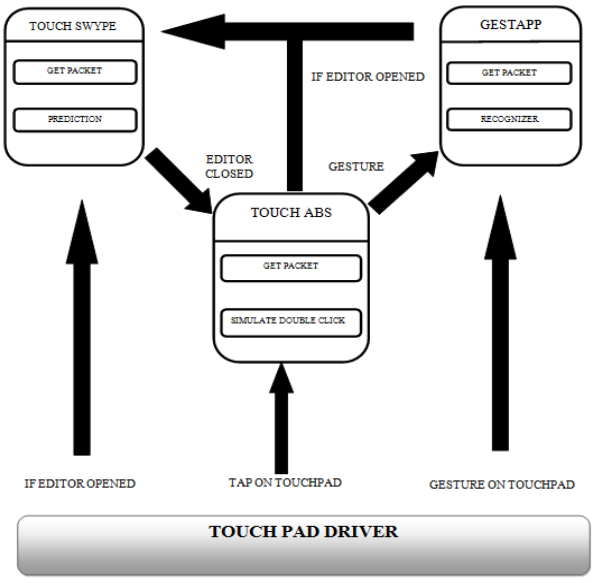

Fig. 1. Unistroke interaction with the Touchpad Driver.

5. WORKING

5.1 Working of AbsTouch

On application start-up, the control is with the AbsTouch module. Whenever the user taps on the Touch-pad, GetPacket function gets the correct tapped co-ordinates from the touch-pad driver and stores it in the database. These co-ordinates are up scaled using a predefined formula which depends on the screen resolution . Now that we have the co-ordinates of the tap on the screen, the cursor can be shifted to that position directly.

But when a user touches the touch-pad, he touches more than 1 pixel due to the finger width. This problem may create an erratic behavior with the cursor on the screen. So, the Normalizer component is used to normalize the received co-ordinates so that the behavior of the cursor is not erratic.

Now that the user is on the correct file/folder, when he lifts his finger from the touch-pad, a double-click is simulated so that the file is opened.

5.2 Working of GestApp

In case the user wants to open any application like Microsoft Word or Mozilla Firefox, he generally goes clicks on the start menu -> programs and finds the corresponding applications and clicks on it. Or else, he keeps it on the desktop for easy access. But after a while, the desktop looks cluttered.

GestApp provides the perfect alternative for users to open their applications with a letter gesture on the touch-pad.

When in AbsTouch mode and the user makes a gesture of "Z" on the touch-pad as shown in the adjacent figure (Red colored Z indicates the finger motion), the application will switch to GestApp mode and this module will get the Packets from the touch-pad driver and give it to Recognizer class which detects the letter.

Recognizer class involves a bit of Artificial Intelligence for error recovery as the user's gesture might not be exact all the time. If the application opened after the gesture is some editor, then the mode switches to TouchSwype or the mode switches to AbsTouch again.

Fig. 2. Finger motion for launching an application. Eg. Gesturing the letter 'Z' will launch an application corresponding to the letter Z.

5.3 Working of TouchSwype

When the app in focus is an editor app like Microsoft Word, Notepad etc., TouchSwype class is in control. In this module, the touch-pad acts like a keyboard with the Qwerty keyboard structure overlaid on it via software. the user can also place a transparent

sticker onto the touch-pad with all keyboard letters engrossed on it for ease of access.

Whenever the user wants to type a word, he will put his finger on the first letter and drag his finger to the next letter from there. This way he will go on dragging onto the next letters of the word. In places where the fingers traverses from the letters which is not a part of the word, the word formed is checked against a dictionary database and if matched, the word is output onto the editor.

TouchSwype gets the co-ordinates from the touch-pad corresponding to the letter typed on the touch-pad, and stores it and waits for next input. When all the letters are input, it checks whether the word is valid.

6. INNOVATION

The conception of Unistroke involves three major components which are, AbsTouch-The Desktop on your keyboard, TouchSwype - Keyboard overlay with word stroking and GestApp – A Gestural Application Launcher.

- **AbsTouch** - A touchscreen for our laptops might cost a fortune for a normal user. Imagine using an inbuilt technology to implement a similar innovative technology as the touch-screen which serves the same purpose. It enables the user to open any file/folder without the need for a mouse drag. This will save a lot of time and will make computer UI handling similar to smart phone touch.
- **TouchSwype** - It allows the user to input text in multiple ways. The user will be able to type the whole word in one swipe of the hand. It will greatly reduce typing time compared to typing words the traditional way (using keystrokes). It allows flexibility of selecting words from the dictionary in case of an ambiguity.
- **GestApp** - Generally to open any application (games,web browser,utilities) we either have to follow the file hierarchy or clutter the desktop with all kinds of shortcuts. Now imagine opening your most played game or web browser from the desktop itself with just a gesture on the touch-pad. GestApp allows the user to open any app with a gesture letter on the touch-pad.

Unistroke has the ability to sense which application is open and which application has focus right now and auto switches to the corresponding service. So, when you open a touch-pad from say Start -> Run, the service will switch from AbsTouch to TouchSwype and now you can swipe your text.

7. PRACTICAL APPLICATIONS

Unistroke offers many practical applications.

AbsTouch:

- *Eliminate need of mouse drag* : User has to drag his mouse to the file/folder before he opens it. AbsTouch eliminates the need for drag on the touch-pad.
- *Eliminate need of click/dblClick*: User has to double click on the file/folder after the pointer is on the file/folder.

A tap on the touchpad will do both these actions. When a user taps, the mouse pointer goes to the point corresponding to the tap and when the user lifts his finger, a double click is simulated. In future, building on this concept will **eliminate the need of a mouse**.

- *Virtual Touchscreen* : With AbsTouch, the touchpad acts as an inexpensive touchscreen alternative free of cost.

TouchSwype:

Keyboard overlay: The touch-pad functionality will switch to provide a keyboard overlay (i.e. the touch-pad will work like a keyboard) when any editor applications (eg. Notepad, Microsoft Word) are open.

Word Stroking : User has to type individual letters of a word on the Keyboard to form words. With TouchSwype , the user can type a word in a single stroke.

Similar applications on smart phones have reduced typing times to a great extent.

GestApp:

Gestural access: In order to open any application, the user has to go through a series of clicks on folder/start menu or has to place an application shortcut on the desktop to open any application. GestApp saves on time and provides an innovative way to access applications through letter gestures.

People with motor impairments:

People having Muscular Dystrophy, Cerebral Palsy, Parkinson's disease cannot use a conventional mouse and keyboard They may lack sufficient mobility to reach for these devices, sufficient motor control to switch accurately and efficiently between them, or sufficient endurance to use them for more than a few minutes.

8. CONCLUSION

We have described three means for easy desktop access and text entry via the preexisting touch-pad on the notebook (Laptop PC). The touch-pad (external or in-built) is small, light, inexpensive, and requires minimal configuration, giving it significant practical advantages as integrated control system over dedicated computer access technologies.

Moreover, buying a touchscreen of the size of a computer screen would be cost prohibitive. We described our designs and implementations of Unistroke. While these techniques can still be improved, this work has paved the way for their future refinement, and ultimately, better computer access.

9. FUTURE SCOPE

We are planning to extend beyond normal absolute touch recognition using error prediction and integrated text entry using probabilistic logic. In Future, some sort of a small, inexpensive touchscreen can be inserted directly below the touch-pad which shows a replica of the desktop while any explorer or non-editor window is open. This will eliminate the need to predict the coordinate of the touch and will result in accurate and faster touches leading to better computer access. If resistive touch screens are used instead of capacitive touch screens, a stylus can

used to point in the touch-pad which will reduce the minor errors introduced while using the considerably wide human finger.

When the user shifts to a editor like Notepad, Microsoft Word, etc, the small touch-pad will auto sense this action and switch its function to a keyboard eliminating the need to predict words and detect errors.

10. REFERENCES

[1] Herbert Schildt , "C Sharp, The Complete Reference"

[2] Jacob O. Wobbrock, Duen Horng Chau and Brad A. Myers, "An Alternative to Push, Press, and Tap-tap-tap:Gesturing on an Isometric Joystick for Mobile Phone Text Entry"

[3] Neil R. N. Enns (University of Toronto) and I. Scott MacKenzie(University of Guelph), "Touchpad-Based Remote Control Devices".

[4] Jacob O. Wobbrock, Brad A. Myers, Htet Htet Aung (Carnegie Mellon University) and Edmund F. LoPresti, "Text Entry from Power Wheelchairs".

[5] Jack Purdum," Wrox Press Beginning C# 3.0 An Introduction to Object Oriented Programming"

[6] Poika Isokoski, "Model for Unistroke Writing Time", Computer-Human Interaction Group, Dept. of Computer and Information Sciences, University of Tampere.