Using Handhelds to Help People with Motor Impairments

Brad A. Myers, Jacob O. Wobbrock, Sunny Yang, Brian Yeung, Jeffrey Nichols, and Robert Miller
Human Computer Interaction Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213
bam@cs.cmu.edu
http://www.cs.cmu.edu/~pebbles/

ABSTRACT

People with Muscular Dystrophy (MD) and certain other muscular and nervous system disorders lose their gross motor control while retaining fine motor control. The result is that they lose the ability to move their wrists and arms, and therefore their ability to operate a mouse and keyboard. However, they can often still use their fingers to control a pencil or stylus, and thus can use a handheld computer such as a Palm. We have developed software that allows the handheld to substitute for the mouse and keyboard of a PC, and tested it with four people (ages 10, 12, 27 and 53) with MD. The 12-year old had lost the ability to use a mouse and keyboard, but with our software, he was able to use the Palm to access email, the web and computer games. The 27-year-old reported that he found the Palm so much better that he was using it full-time instead of a keyboard and mouse. The other two subjects said that our software was much less tiring than using the conventional input devices, and enabled them to use computers for longer periods. We report the results of these case studies, and the adaptations made to our software for people with disabilities.

Keywords: Assistive Technologies, Personal Digital Assistants (PDAs), Handicapped, Disabilities, Hand-held computers, Palm pilot, Muscular Dystrophy, Pebbles.

INTRODUCTION

About 250,000 people in the United States have Muscular Dystrophy (MD), which is the name given to a group of noncontagious genetic disorders where the voluntary muscles that control movement progressively degenerate. One form, called Duchenne Muscular Dystrophy (DMD), affects about one in every 4,000 newborn boys. With Duchenne, boys start to be affected between the ages of 2 and 6, and all voluntary muscles are eventually affected [10]. First affected are the muscles close to the trunk, and nearly all children with DMD lose the ability to walk sometime between ages 7 and 12. In the teen years, activities involving the arms, legs or trunk require assistance. Becker Muscular Dystrophy is a much milder version, and the onset can be in late adulthood. A related disorder is Spinal Muscular Atrophy, which is an inherited neuromuscular disease that causes weakness in the body, arms, and legs. It affects both boys and girls starting at 6 months to 3 years, and progresses rapidly [10].

Figure 1. 12-year old Kevin has Duchenne Muscular Dystrophy. He is operating his PC by using two hands to control the stylus on a Palm running Remote Commander.

People with these and related disorders, like the rest of the population, are increasingly using computers. Unfortunately, these disorders often make it difficult for them to move their arms, wrists and fingers, and therefore conventional keyboards and mice become difficult or impossible for them to use. We are investigating how a handheld computer, such as the Palm, can be used to enable people with disabilities to access their PCs (see Figure 1).

This paper reports on how we have adapted our Pebbles software for the Palm to facilitate its use by people with motor impairments. In particular, we adapted the Pebbles Remote Commander program, which allows the Palm to be used as if it was the PC's keyboard and mouse. In general, the Pebbles project is studying many ways that handheld computers can interoperate with PCs, with each other, and with smart appliances [11]. These applications are aimed at business meetings, offices, classrooms, military command posts, and homes. The Pebbles software is available for downloading from: http://www.pebbles.hcii.cmu.edu/. Pebbles stands for **P**DAs for the **E**ntry of **B**oth **B**ytes and **L**ocations from **E**xternal **S**ources.

In a preliminary case study of four people with MD, we found that our modified Pebbles applications successfully allowed the use of the PC for extended periods of time for

To appear in:
Fifth International ACM SIGCAPH Conference on Assistive Technologies; ASSETS 2002. July 15-17, 2002. Edinburgh, Scotland.

people who found it difficult or impossible to use a conventional keyboard and mouse. The observations we made include:

- A conventional keyboard and mouse is usually faster than Remote Commander, for people who can still use those devices. However, using Remote Commander as an alternative input method is still valuable because they can switch to using the Palm when their hands become tired using conventional input devices.
- The physical buttons on the Palm device (and on commercial touch pads) require too much force for our subjects to push. Providing on-screen buttons that could be lightly tapped with a stylus was more successful. Since the Palm power button was too difficult to press, we needed to add an easy way to prevent the Palm from turning itself off.
- It is useful to have a small on-screen keyboard since the users have difficulty moving. The ability to adapt the keyboard to individual requirements is also desirable.
- The provided metal Palm stylus is too short and heavy, and an inexpensive long plastic stylus works better.
- Seeing the Palm screen often requires additional lighting at the computer, or a backlit Palm screen. A long cord from the PC to the Palm is also necessary.
- We needed to change the Remote Commander software to allow the key-repeat to be turned off, to provide easier ways to pop up the on-screen keyboard, and to make it easier to tap to generate left mouse click events.

MOTIVATION

Children and adults with MD and other disorders use computers for the same purposes as everyone else: to send electronic mail, surf the web, write up homework and reports, play games, etc. People with disabilities may find that using a computer is even more valuable because they may be homebound more often and have less access to other forms of entertainment and employment. In addition to the physical limitations imposed by the disease and by being in a wheelchair, often there is an increased susceptibility to infections. A simple cold can quickly progress to pneumonia and can be life-threatening due to a weakened respiratory system and heart. For example, Jennifer, one of our test subjects, is being educated at home because of her parents' concerns about respiratory infections [18]. Using a computer allows Jennifer to interact with her fourth-grade classmates and write up her homework in what her father calls "virtual mainstreaming" [18].

Two of our adult test subjects use computers in their work. They help create and maintain web pages for others and interface with their collaborators through email. Web authoring is an attractive task since it can be performed at home at whatever pace is comfortable.

Of course, all of our subjects report using their computers for entertainment. Computers supply another option besides watching television that people with disabilities can operate independently. Using a computer can also be much more intellectually stimulating.

Unfortunately, although a large number of assistive technology products are available today, many people with physical disabilities do not have adequate input devices. A significant barrier is that many of today's assistive devices are very expensive. Two different studies of students found that less than 60% of those who indicated that they needed adaptations to use a computer actually used adaptations [4]. When asked why they did not use adaptations, students overwhelmingly cited the high cost. Other reasons included that the devices are unavailable, the students do not know where to get devices, they do not know how to use the equipment, and equipment is too expensive to maintain [4].

By using commercially available consumer hardware and free software, we can deliver an adaptive technology for people with physical disabilities at a very inexpensive price. For example, the least expensive Palm devices are around $100, compared to $400 or so for a specialized keyboard for the handicapped. The Palm is also programmable and adaptable, so the layout and functions of the assistive software can be adjusted (as described below). Another advantage is that our software combines the keyboard and mouse functions in the same very small space, so only one device is needed.

RELATED WORK

There is much work on assistive devices and making computers more accessible to people with disabilities. However, there is less work focused on assistive technologies for neuro-muscular disorders. Exceptions include Trewin and Pain's report on a modeling technique to help evaluate problems with keyboard configurations [19], and several comparative evaluations of commercial devices (e.g., [5]). Also, the Archimedes project at Stanford investigated using a handheld, called the Communicator, to help the disabled [16].

Operating systems, including newer versions of Windows and the Macintosh, have built-in accessibility modes that enable people with disabilities to avoid pressing two keys at once, or holding down the mouse button while dragging. Microsoft has a large website devoted to accessibility (http://www.microsoft.com/enable/) which lists over 330 different keyboard enhancement utility products which might be of use to people with various disorders. For people with no ability to use their hands, there are products such as head and eye trackers, chin and mouth controlled devices, "sip and puff" devices, speech recognition, etc. Of particular relevance to the work discussed here are the Enkidu products (see http://www.enkidu.net/), which provide portable and handheld stand-alone interfaces, including a palm-size device with a soft keyboard that allows users to write sentences that the device will speak aloud using text-to-speech. These communication devices sell for $2,700 to $4000 each.

Other research relevant to our system is the work on new input technologies for handhelds, such as Quikwriting [15], and new kinds of keyboards (e.g., [8] [20]).

METHODOLOGY

For this work, we created adaptations of our Remote Commander software. We then gave this software to four subjects who have MD. One, Jennifer, lives in New York, so we were not able to perform direct observations, and all information has been collected through electronic mail from her father. Local subjects volunteered after reading an article in the Pittsburgh-area MD newsletter. We visited the three local subjects in their homes. In exchange for allowing us to observe them, we loaned them a Palm IIIx (or an IBM Workpad which is the same except the case is black). Palm and IBM donated the devices. After interviewing the subjects about their background, daily routine, and computer use, we asked them to show us how they use computers today. (We also interviewed one child's mother.) We then installed our software on the Palm and on their PCs, and performed an initial evaluation of how well the Palm software seemed to be working for them. Many of our observations and software changes described below were a result of the difficulties we observed in this initial use. We left the Palm and the software, and then came back later for a follow-up visit to observe how the subjects were doing.

We used a form of the *contextual inquiry* method for the interviews [7], since it provides information about tasks and actions in the appropriate context. As often observed in conventional user studies, our subjects were generally unable to articulate problems they were having or propose ideas for improvements. It was only through directly observing and interacting with the subjects, as recommended by the contextual inquiry method, that we were able to gain insights for design improvements.

Technology

The Remote Commander program was the first application created as part of the Pebbles project, and was originally put on the web for free downloading in February, 1998. Since then, it has been downloaded over 30,000 times. Remote Commander allows strokes on the main display area of the Palm to control the PC's mouse cursor (Figure 2-a), and for Graffiti input to substitute for the PC's keyboard input. Remote Commander has a custom pop-up on-screen "soft" keyboard that contains all of the keys on a standard PC keyboard (Figure 2-b). As with the regular Palm keyboard, Remote Commander's keyboard pops up when the user taps in the "abc" or "123" areas at the bottom of the Graffiti area. Since Remote Commander inserts the input into the PC's low-level input stream, it can be used to supply input to any PC application. The original goal for Remote Commander was to enable multiple people at a meeting to take turns controlling a PC using their handhelds. In addition to the Palm version shown here, we implemented versions of Remote Commander for Microsoft PocketPC and WindowsCE devices.

(a) (b)

Figure 2. (a) The main screen in Remote Commander is used as a touchpad. The labels (Shft, Ctrl, etc.) describe what the physical buttons do, and also serve as on-screen buttons. (b) The pop-up on-screen "soft" keyboard has all the PC's keys. The top blank area above the keyboard can still be used as a touchpad to move the mouse.

Another Pebbles application, called Shortcutter, allows users to create custom panels of buttons, sliders, knobs, and pads to control any PC application. Although we did not adapt Shortcutter or measure its performance in our case studies, we demonstrated its use and some subjects found it useful and created their own panels of controls.

All of the Pebbles applications, including Remote Commander, must communicate with the PC. Pebbles supports a variety of transport protocols, including serial cables (such as the cradle supplied with Palm devices or a regular cable), infrared (IR), and wireless local-area network technologies such as 802.11b and BlueTooth. For the current study, we used fifteen-foot serial cables donated by Synergy Solutions. IR seems attractive since it is built into Palms already, but it is directional and short range, so maintaining a long communication between the Palm and the PC is difficult and drains a lot of battery power. However, it would certainly be desirable to have an effective low-power wireless communication method to replace a cable. Fortunately, wireless is coming, driven by many other handheld and telephony applications, so we are optimistic about using a wireless technology for Remote Commander in the future.

Keyboard and Mouse Tests

To help measure the subjects' computer input rate, we implemented a keyboard test and a mouse test from the literature. For the mouse test, we used the classic pointing experiment from Card, Moran and Newell [1] (Experiment 7B, pp. 250). A picture of the test application is shown in Figure 3. The test consists of clicking between a central stationary target and two outlying targets. The current target is shown in a different color and alternates from side to center to other side and back. The outer targets can have one of three widths (16, 64, or 128 pixels) and be two distances from the center (36 or 144 pixels). The sequence of condi-

tions is randomly generated in advance. The number of times the mouse is clicked in the correct place and in incorrect places (errors) is measured.

The typing test is shown in Figure 4, and is based on a test by Soukoreff and MacKenzie [17]. A phrase is displayed and the user is asked to type it exactly. Subjects are allowed to fix mistakes, but they are not required to do so. When they are satisfied with the phrase, they hit the ENTER key, at which point a new phrase appears or the test ends. Phrases may contain capital and lower-case letters, quotation marks and other punctuation. All subjects see the same phrases in the same order.

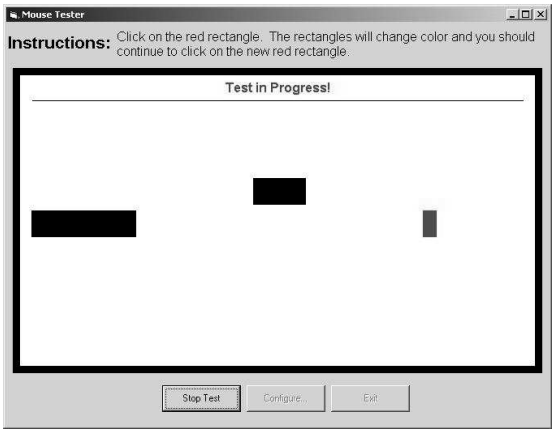


Figure 3. Screen for the mouse test.

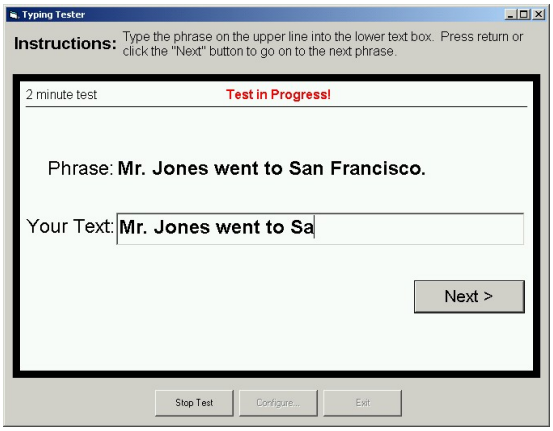


Figure 4. Screen for the typing test.

CASE STUDIES

The next sections discuss our initial four users with disabilities. After that, we discuss the adaptations we made to the Pebbles software to take into account our observations of their use of the programs.

Jennifer

Jennifer is a 10-year old girl with a type of MD known as "Spinal Muscular Atrophy, Type 2." Her father found the original versions of Remote Commander and Shortcutter on the web, and tried them out as a way for Jennifer to more easily use her computer. It was so successful that he contacted the MD national magazine, *Quest*, which wrote an article about Remote Commander [18]. Jennifer and her father appeared on the annual MD television telethon in September, 2001 and mentioned their experience with Remote Commander.

Jennifer lives in New York, and we have not had any direct observation of her. Her results on the mouse and typing tests showed her to be about 24% slower with Remote Commander than a mouse, and 35% slower with the Remote Commander compared to a regular keyboard. However, Jennifer's father reports that:

> "Jen uses the Palm about 50% of the time. She has greater speed in using the regular keyboard and mouse, but she fatigues much more rapidly using them. The Pebbles software permits her to use the PC over a much greater amount of time, though slower. That is very important." *— by email on 14 Apr 2001*

Kevin

Kevin is a 12-year old boy with Duchenne Muscular Dystrophy (see Figure 1). He has used computers since he was in first grade (age 6), and was able to use a regular keyboard and mouse at that point, but with difficulty. He never learned to touch-type, and does not know the QWERTY keyboard layout. He has very little strength in his fingers, and cannot move his arms at all. He has now completely lost the ability to use a keyboard and mouse. He tried using a small commercial touchpad, but found the buttons to be too difficult to press. Also, the computer is shared with his parents, and the touchpad *replaced* the mouse, making it difficult for others to use the computer. Kevin had tried speech-recognition software, but could never get it to work well enough. With Remote Commander, he was able to use a stylus to move the cursor and select letters from the pop-up soft keyboard. He held the stylus in his right hand, but found it most effective to use his left thumb to help move the stylus, especially to tap on letters on the left side of the screen (see Figure 1). He was able to use the mouse feature without looking down at the Palm, but had to look down to use the on-screen keyboard.

Dan

Dan is 27 years old and also has Duchenne Muscular Dystrophy. He started using an Atari computer with a joystick when he was 8 or 10 years old, and used Macintoshes in high school and an IBM at home. He went to college, where he was able to use a mouse while resting his arm on a wrist-pad. Starting a few years ago, he lost the ability to type, but discovered that he could still press the keys using the eraser ends of two pencils (see Figure 5). He reports, "it works, but sometimes my fingers get sore, because of the pencils."

About two years ago, he lost the ability to use a mouse, and found a trackball he liked (also shown in Figure 5). Dan needs someone to move his hands into position before he can begin using the devices. He has dictation software, but never uses it. Dan said he "spends a decent amount of time on the computer" and sends and receives about 10 emails a day. He also runs a web page about a golf game.

Figure 5. Dan was typing using two pencils before he started using the Palm. His trackball is on the wristpad.

After we delivered the Palm, he started using it exclusively. Originally, he tried the on-screen soft keyboard, at one point trying out two styluses at once. However, he stopped using the soft keyboard because his neck got sore from continually looking up and down. He learned the Palm's Graffiti language and then used it for all text entry. He said he put the trackball aside and only used the real keyboard to turn on the computer. He created a Shortcutter panel to launch his favorite applications. In our tests, Dan was 40% slower using Remote Commander instead of the trackball, but 36% *faster* using Remote Commander rather than the regular keyboard.

Subject4

Subject4 is a 53-year old man with Becker Muscular Dystrophy. (He prefers to remain anonymous.) He was employed as a teacher and an engineer but left in 1995 on disability. He currently works at home, authoring and maintaining sophisticated database-driven web pages. Subject4 is still able to use a conventional mouse and keyboard, but he has great difficulty moving his hand to and from the mouse. He uses his fingers to help "walk" his hand across the desk. The Palm software was appealing to him because it has the keyboard and mouse control together in the same small place. Subject4 was 40% slower with Remote Commander compared to a mouse, and 58% slower than a regular keyboard.

After a week with the Palm, Subject4 reported:

> "I have been using the Palm primarily for the mouse functions when I am near the keyboard. I have found the Palm most useful when I am sitting at my desk (away from the keyboard). I can use it to bring up email and other functions [so] I don't have to [move to] be right in front of the screen." *— by email on 10 May 2001*

Since movement is difficult for people with MD, the ability to stay where they are and still perform some computer commands can save considerable time and effort.

OBSERVATIONS AND MODIFICATIONS

As part of our analysis of the needs of people with disabilities, we made a number of observations, which led to changes in the software and environment. These are described in the following sections.

Requirements

Although our subjects are able to move their fingers, they are very weak, and cannot exert much pressure. This means, for example, that they are unable to press the physical buttons on the Palm. They also tap the Palm screen lightly.

Since they have limited range of movement, it is best if all of the important functions are clustered together in a small area. Even moving to the far side of the small 3¼-inch LCD display screen of the Palm is difficult for Kevin.

Physical Interface

One of the first challenges was getting the Palm into a convenient position for the subjects. The Palm comes with a cradle, but pressing on the Palm screen while it is in the cradle causes it to rock, and sometimes even fall over. It is also too high and at the wrong angle for our subjects. Therefore, we tried the Hotsync serial cable sold by Palm. Unfortunately, this cable is very short and the subjects could not get the Palm in a good position with the cable plugged into the back of their computer. Synergy Solutions gives away a 15-foot serial cable free to people who purchase their SlideShow Commander product (which is based on our Pebbles software and licensed from CMU, see http://www.slideshowcommander.com/), and they donated ten cables for use by this research. These worked very well for our subjects, and can be seen in Figure 1.

The next physical issue was the stylus. The Palm III comes with a short metal stylus that fits into the back of the case (see Figure 6-a). Unfortunately, our subjects found this stylus to be too heavy to hold comfortably for long period of time. Also, Kevin found the stylus to be too short to reach all the way across the Palm screen from his normal hand position. We tried a longer stylus but discovered another problem. Kevin holds the stylus at a very low angle (see Figure 1), and the side of the stylus, rather than the tip, occasionally hit the screen. Therefore, we needed a long, light, tapered stylus. Dan liked a souvenir stylus we received at a conference, and Handango.com graciously donated 60 more to this research. These worked well for our subjects, and can be seen being used in Figure 1.

An environmental problem is that all three of our local subjects often used their computer in a dark part of the room. Normally, this isn't a problem since the monitor gives off light. However, it was difficult to see the Palm screen since it is reflective. This problem was made worse by the position of the Palm screen which was normally lying flat on the table and therefore at an angle from the viewer. Turning on overhead lights helped, but we had to bring an extra table lamp next to the computer so Kevin could adequately see the Palm screen.

(a)

(b)

Figure 6. (a) The standard Palm IIIx stylus and (b) the stylus donated by Handango.

A final physical problem is powering the devices. Normally, a Palm is not expected to be used continually for long periods of time. Furthermore, running the serial port drains a noticeable amount of extra power. The Palm III runs on two AAA batteries, and the subjects frequently needed to replace them. Dan measured that he got about 17 hours of use over four days before his batteries needed replacing. Originally, he had tried running the Palm with the backlight on, to solve the brightness problem mentioned above, but then the batteries only lasted a few hours. The best solution would probably be to switch to a Palm device that can be plugged in to run on wall-power.

Software Changes

In order to make our applications more useful for people with physical disabilities, we have continually evolved the software based on our observations and feedback. While some of these issues are well known for PC accessibility software, others are specific to our application.

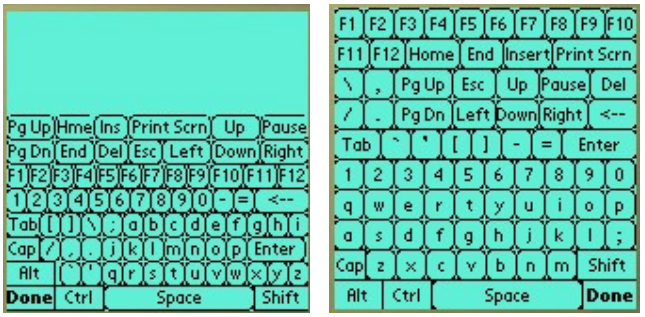
In the original Remote Commander, the physical buttons (which normally switch to different Palm applications) are used as modifiers or to "press" the mouse buttons. This proved very useful for most people, but it was difficult to remember what each button does, especially since they are assignable. Therefore, we added labels, as shown in Figure 2-a. Our subjects with disabilities were not able to push the physical buttons, however, so we made the labels into on-screen soft buttons that could be tapped.

An interesting issue arose about how the modifiers should work. With the physical buttons, users can hold down the button while they make a Graffiti stroke. For example, with the configuration in Figure 2-a, holding down the phone button while making the Graffiti "z" stroke will send control-z to the PC. For the soft buttons, you cannot hold the stylus on the soft button and at the same time perform a Graffiti stroke, so all the soft buttons were made into toggles. The tradeoff is that sometimes users forget to turn off the modes, so extra characters become shifted. As a compromise, possibly some of the modifiers should be automatically turned off after the next character, in the same way as the Palm's built-in keyboard, but this has not been implemented yet.

As an alternative to the soft buttons, tapping on the blank part of the screen that serves as a touchpad will send a left mouse click. The subjects, however, had difficulty getting this to work because they would tap too slowly or move slightly while trying to tap, and the software would recognize the input as a regular move instead of as a tap. An option was therefore added to make the tap recognition more tolerant, and then all the subjects were able to generate taps without problems. We were worried that this might increase the occurrence of accidental taps (sending a left click event when the user just meant to move the cursor), which could cause unexpected operations on the PC, but this did not appear to be a problem in practice.

Early on, we noticed that the subjects often triggered multiple characters when using the on-screen keyboard, due to the built-in auto-repeat of the keys. An option was therefore provided to turn key-repeat off. When Kevin tried using Remote Commander to play some PC games, we discovered another problem. We were originally sending key down and key up events together, so there was no way to hold a key down using Remote Commander. Many games depend on the keyboard keys being held down, so we modified the software to fix this.

Three of our subjects were experienced keyboard users before coming to Remote Commander, but Kevin was not. Consequently, he found it difficult to locate the desired letters on the default keyboard (Figure 2-b). Therefore we added an alphabetic keyboard (Figure 7-a). We pushed all the letter keys to the right, along with Shift and Enter, to make them easier for Kevin to reach, since he found it more difficult to hit letters on the left of the screen. We also added larger versions of both the QWERTY and alphabetic keyboards (Figure 7-b) because some of the subjects initially had trouble hitting the tiny keys. However, with experience, the subjects were better able to hit the keys, and they liked not having to move as far. They also liked the ability to use the blank area at the top to control the mouse while the keyboard is displayed.



(a) (b)

Figure 7. New keyboards created for Remote Commander.

In the future, we plan to investigate other keyboard configurations, such as those reported in the literature [8, 20]. Adaptive keyboards, either automatically or user-defined, and predictive input techniques (see next section) may also be useful.

We encouraged the subjects to experiment with the Palm on their own, and Dan became quite facile with the Graffiti gestures. However, Kevin could not perform Graffiti very accurately, because his gestures were too slow and wiggly. He therefore decided he would like to always use the keyboard. Since he had trouble hitting in the small "abc" area that pops up the keyboard, we provided an option where tapping anywhere in the Graffiti area pops up the keyboard.

The Palm screen is only 160 pixels across, but PC screens can be 1280 pixels across or even more. Therefore, the mouse control in Remote Commander operates in relative mode, like a laptop's touchpad. Movements across the Palm screen move the PC's cursor an equivalent amount across

the PC's screen. We built a little acceleration into Remote Commander, so faster movements on the Palm would move further on the PC. However, our subjects with disabilities still found it tedious to move large distances, since they tended not to move the stylus quickly, and were not able to make large strokes. Therefore, we added additional levels of acceleration so smaller movements on the Palm would make bigger movements on the PC.

A final fix to Remote Commander was an option to disable the auto-power-off feature of the Palm. To save batteries, the Palm allows users to pick whether it turns itself off after 1, 2, or 3 minutes of non-use. Some of our subjects with disabilities were not able to turn the Palm back on by themselves, so having it turn off was a big problem. In some other Pebbles applications, we disable the auto-off completely, but we were worried that here the users would end up leaving the device on until the batteries ran out. Therefore, we made an option where it would go off after 30 minutes of non-use, which seemed long enough to indicate that they were not at the computer.

CURRENT WORK

One clear problem is that entering text with Remote Commander is slower than using the regular keyboard. We are investigating the use of word prediction to address this. Word prediction, here meaning both *word completion* and *next word prediction*, has been studied for many years (see, e.g., [3]) and has been applied to the Palm [9] and to help the disabled use a PC [6], but we are not aware of any hybrid systems such as described here.

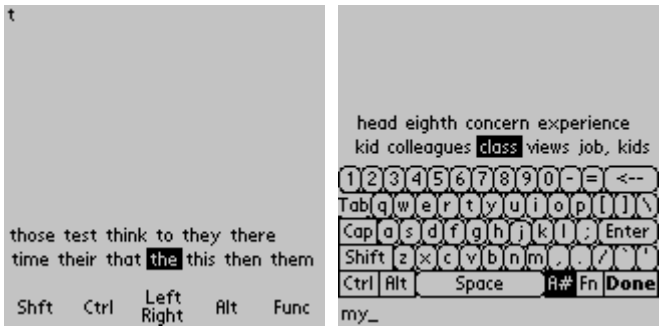
Word prediction systems have been built using different approaches. Our strategy is to combine the predictions from three sources, visited in this order: a word cache, a bigram list, and the alphabetical array of words in the vocabulary.

The word cache contains the most recent words entered by the user, whether or not the words are in the vocabulary. In this way, the system can adapt to the user's in-vocabulary words and out-of-vocabulary words.

Bigrams are pairs of words used to generate predictions. We get our bigrams by using the CMU-Cambridge toolkit [2] to parse a corpus of text on the PC, and then by building a vocabulary file (a list of the most common words) and a bigrams file (a list of the most common word-pairs). Our desktop application transforms these files into a Palm database file, which is downloaded to the handheld and accessed by the word prediction system at run time. We use the bigrams and word array to make predictions based on the previous word and, optionally, any letters the user has entered in the current word.

The Palm's processor is slow and its memory is limited, so the algorithms involved are optimized by pre-computing lookup indices on the desktop when writing the Palm database. For a vocabulary of 10,000 words, the word prediction database is roughly 270 KB on the Palm.

Developing a word prediction system is more than a technical challenge, it is a design challenge made difficult by the Palm's small screen space. Currently, we are exploring different layouts for displaying the predictions. Two current layouts we're exploring are shown in Figure 8. A key issue in these designs is to minimize the distance the hand must travel. The predictions in (a) are placed above the onscreen toggles for the hard buttons. These predictions appear while the user is entering Graffiti. The word in progress ("t") is shown in the upper left corner for feedback. When the keyboard is up (b), the predictions are placed just above it, and the letters already typed, in this case "my_", are shown below. Note here we are predicting the next word after "my". The space below the keyboard can be used to start mouse movements that extend up the full height of the screen, across the keyboard. In both designs, the most likely prediction is placed in the center to minimize travel distance, and is highlighted to provide a visual reference point. The next-most-likely predictions surround it. The least likely predictions are at the edges.



(a) (b)

Figure 8. Layouts for word prediction when the keyboard is hidden (a) and displayed (b). The keyboard is in alphanumeric mode ("A#"), and the function keys and common control sequences are accessible in function mode ("Fn").

FUTURE WORK

We have just begun our study of how handhelds can be used to help people with neuromuscular disorders, and there is much work to be pursued in many areas. We will continue to follow-up with our initial subjects, to see what further modifications would be useful. Soon we hope to try out our word prediction software with them. Occupational therapists have told us that our software may be useful for some people with ALS, cerebral palsy, and arthritis, so we hope to test the system on people with these disorders and see what further modifications would be useful. We plan to directly compare the Palm software to other assistive technologies now on the market, not just to the general input devices that our subjects already had.

We are developing other applications as part of the Pebbles project that might be useful for people with disabilities. We are working on having the handheld serve as a "Personal Universal Controller" (PUC) for appliances and devices [14]. We might even be able to use the Palm to control the wheelchair, but there are significant safety and liability issues with this idea. The Shortcutter application already

supports the standard X-10 protocol for sending signals through the power wires, but much more work is needed. This effort is related to the NCITS-V2 initiative on an Alternative Interface Access Protocol [13].

We developed a new interaction technique called "semantic snarfing" [12] where the PC's screen contents are sent to the handheld where detailed interaction is performed. This might prove useful for people with disabilities. For example, the handheld might reformat the PC's screen to be much larger or the text to use a larger font to help people with limited eyesight. It might also be useful for the text that is snarfed to the handheld to be read by a text-to-speech engine.

CONCLUSIONS

The programs discussed above are available for free use and can be downloaded from our web site: http://www.pebbles.hcii.cmu.edu/assistive. We hope that this software will be useful to a wide range of people, and that we can continue to investigate new ways that handhelds can help people with disabilities. Furthermore, working with people with disabilities is beneficial to the project in general, because the observations and changes we have made to our software as a result of these studies will improve the software for all the users of our applications.

ACKNOWLEDGMENTS

For help with this research, we would like to thank Drew Rossman, Amy Lutz, John Lee, Susan E. Fridie, Ronald Rosenfeld, and our subjects. Thanks also to Chris Long, Bernita Myers, Tim Carey, and John Lee for comments on earlier drafts of this paper.

The research reported here is supported by equipment grants from Palm, IBM, Synergy Solutions and Handango. The Pebbles project is supported by grants from DARPA, NSF, the Pittsburgh Digital Greenhouse, and Microsoft.

REFERENCES

1. Card, S.K., Moran, T.P., and Newell, A., *The Psychology of Human-Computer Interaction.* 1983, Hillsdale, NJ: Lawrence Erlbaum Associates.
2. Clarkson, P. and Rosenfeld, R. "Statistical Language Modeling Using the CMU-Cambridge Toolkit," in *European Conference on Speech Communication and Technology (Eurospeech '97).* 1997. Rhodes, Greece: pp. 2707-2710.
3. Darragh, J.J., Witten, I.H., and James, M.L., "The Reactive Keyboard: A Predictive Typing Aid." *IEEE Computer*, 1990. pp. 41-49.
4. Fichten, C.S., Barile, M., Jennison, M.S.W., Asuncion, V., and Fossey, M., *What Government and Organizations Which Help Postsecondary Students Obtain Computer, Information And Adaptive Technologies Can Do To Improve Learning and Teaching: Recommendations Based On Empirical Data.* Ev-NET, Working Papers #3, August, 1999. Montreal, Quebec, Canada. http://evnet-nt1.mcmaster.ca/network/workingpapers/Adaptive/adaptive_recomend.htm.
5. Fuhrer, C.S. and Fridie, S.E. "There's A Mouse Out There for Everyone," in *Calif. State Univ., Northridge's Annual International Conf. on Technology and Persons with Disabilities.* Los Angeles, CA: http://www.csun.edu/cod/conf2001/proceedings/0014fuhrer.html.
6. Garay-Vitoria, N. and González-Abascal, J. "Intelligent Word-Prediction to Enhance Text Input Rate," in *Proceedings of the ACM's Conference on Intelligent User Interfaces (IUI '97).* 1997. Orlando , FL: pp. 241-244.
7. Holtzblatt, K. and Jones, S., "Conducting and Analyzing a Contextual Interview," in *Readings in Human-Computer Interaction: Toward the Year 2000, 2nd ed.*, R.M. Baecker*, et al.*, Editors. 1995, Morgan Kaufman. San Francisco. pp. 241.
8. MacKenzie, I.S. and Zhang, S.X. "The design and evaluation of a high-performance soft keyboard," in *Proceeding of the CHI 99 Conference on Human Factors in Computing Systems.* 1999. Pittsburgh, PA: pp. 25-31.
9. Masui, T. "An efficient text input method for pen-based computers," in *CHI'98: Conference on Human Factors and Computing Systems.* 1998. Los Angeles, CA: pp. 328-335.
10. MDA, "The Muscular Dystrophy Association," 2001. http://www.mdausa.org/index.html.
11. Myers, B.A., "Using Hand-Held Devices and PCs Together." *Communications of the ACM*, 2001. **44**(11): pp. 34-41.
12. Myers, B.A., Peck, C.H., Nichols, J., Kong, D., and Miller, R. "Interacting At a Distance Using Semantic Snarfing," in *ACM UbiComp'2001.* 2001. Atlanta, Georgia: pp. 305-314.
13. NCITS-V2, "National Committee for Information Technology Standards Technical Committee V2: Alternative Interface Access Protocol (AIAP)," 2001. http://www.ncits.org/tc_home/v2.htm.
14. Nichols, J.W. "Using Handhelds as Controls for Everyday Appliances: A Paper Prototype Study," in *ACM CHI'2001 Student Posters.* 2001. Seattle, WA: pp. 443-444.
15. Perlin, K. "Quikwriting: continuous stylus-based text entry," in *palm: UIST'98: Proceedings of the 11th annual ACM symposium on User interface software and technology.* 1998. San Francisco, CA: pp. 215-216.
16. Scott, N., "Communicator Project of the Archimedes project; Research," 1997. http://archimedes.stanford.edu/research97.html.
17. Soukoreff, R. and MacKenzie, I. "Measuring Errors in Text Entry Tasks: An Application of the Levenshtein String Distance Statistic," in *Extended Abstracts of CHI 2001.* 2001. Seattle, WA: pp. 319-320.
18. Stack, J., "Palm Pilot Connects Girl with Classroom." *QUEST*, 2001. **8**(1): pp. 48-49. http://www.mdausa.org/publications/Quest/q81palmpilot.cfm. Magazine of the Muscular Dystrophy Association.
19. Trewin, S. and Pain, H. "A model of keyboard configuration requirements," in *ASSETS'98: Proceedings of the third international ACM conference on Assistive technologies.* 1998. Marina del Rey, CA: pp. 173-181.
20. Zhai, S., Hunter, M., and Smith, B.A. "The Metropolis Keyboard – An Exploration of Quantitative Techniques for Virtual Keyboard Design," in *UIST'2000: Proceedings of the 13th annual ACM symposium on User interface software and technology.* 2000. San Diego, CA: pp. 119-128.