PalmType: Using Palms as Keyboards for Smart Glasses

Cheng-Yao Wang, Wei-Chen Chu, Po-Tsung Chiu, Min-Chieh Hsiu,
Yih-Harn Chiang, Mike Y. Chen
National Taiwan University
{r00944052, r01922002, b99901030, r03922073, b99902104, mikechen}@ntu.edu.tw

ABSTRACT

We present PalmType, which uses palms as interactive keyboards for smart wearable displays, such as Google Glass. PalmType leverages users' innate ability to pinpoint specific areas of their palms and fingers without visual attention (i.e. proprioception), and provides visual feedback via the wearable displays. With wrist-worn sensors and wearable displays, PalmType enables typing without requiring users to hold any devices and does not require visual attention to their hands. We conducted design sessions with 6 participants to see how users map QWERTY layout to their hands based on their proprioception. To evaluate typing performance and preference, we conducted a 12-person user study using Google Glass and Vicon motion tracking system, which showed that PalmType with optimized QWERTY layout is 39% faster than current touchpad-based keyboards. In addition, PalmType is preferred by 92% of the participants. We demonstrate the feasibility of wearable PalmType by building a prototype that uses a wrist-worn array of 15 infrared sensors to detect users' finger position and taps, and provides visual feedback via Google Glass.

Author Keywords

Palm-based interaction; Smart glass; QWERTY keyboard; Text input; Wearable

ACM Classification Keywords

H.5.2. Information Interfaces and Presentation (e.g. HCI): User Interfaces - Input devices and strategies.

INTRODUCTION

Smart wearable devices such as smart watches and smart glasses are pushing the boundaries of mobile computing. Their mobility and comfort requirements necessitate form factors that have reduced display size, input area, and battery life. Compared to handheld devices such as smartphones and tablets, these wearable devices provide much quicker access to information and also free users from having to hold devices in their hands.

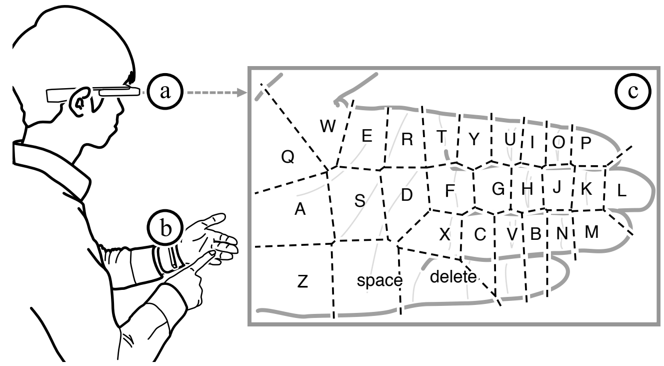

Figure 1. PalmType enables text input for (a) smart glasses using a QWERTY keyboard interface, by using (b) wrist-worn sensors to detect the pointing finger's position and taps, and (c) displaying a virtual keyboard with highlighted keys via the display of the glasses.

However, the form factors of wearable devices limit their input functionality. For example, Epson's Moverio smart glasses [2] uses an external, handheld touchpad to enter text, requiring users to first locate and hold the touchpad before using its virtual keyboard. Google Glass [3] only supports text input via voice, which cannot be used in public settings when privacy is a concern, nor can it be used when voice input is not socially appropriate, such as in meetings and lectures. Although Google Glass has a touch-strip on its right side that supports four touch gestures (tap, back/forward, down), no text input is supported – possibly because repeated touching of Google Glass may be socially awkward.

We present PalmType, which uses palms as interactive surfaces to enable an intuitive and efficient keyboard interface for smart glasses. PalmType leverages humans' innate ability to pinpoint different regions of their palms and fingers without visual attention (i.e. proprioception), and overlays a familiar keyboard layout, QWERTY, onto users' non-dominant hands.

As shown in Figure 1, wrist-worn sensors can be used to detect the pointing finger's position and taps, and a virtual keyboard that highlights the keys being selected is shown via the display of the smart glasses. This visual feedback eliminates the need for users to memorize keyboard layout and enables users to type in a natural posture without looking down at their hands.

To better understand how proprioception can be leveraged to optimize the QWERTY keyboard layout, we conducted design sessions with 6 participants. As shown in Figure 1(c), participants utilized the entire width of the non-dominant

MobileHCI '15, August 24 - 27, 2015, Copenhagen, Denmark.
 ISBN 978-1-4503-3652-9/15/08..$15.00.
http://dx.doi.org/10.1145/2785830.2785886

hand. Also, they adjusted the size and positions of the keys to reflect the different regions of the palm and fingers that can be accurately pointed upon without visual attention. For example, the 3 segments and 2 joints/creases of each finger are mapped to 5 different keys.

We conducted a 12-person user study to compare typing performance and user preference for three types of keyboards for smart glasses: 1) virtual keyboard using touchpad with relative pointing, 2) PalmType with rectangular QWERTY layout, and 3) PalmType with optimized QWERTY layout. Google Glass is used to display the virtual keyboards and visual feedback, and camera-based Vicon 3D motion capture system is used to track users' palms and fingers in real-time.

Study results showed that participants achieved 7.7 words per minute (WPM) using PalmType with optimized layout, which was 39% faster than touchpad, and 21% faster than the rectangular QWERTY layout. Also, participants were able to type 10.0 words per minute in word repetition (i.e. expert) mode. In terms of user preference, 92% preferred PalmType, among whom 81% preferred the optimized layout.

We prototyped a wearable version of PalmType using Google Glass and a wrist-worn array of 15 infrared proximity sensors, and applied machine learning to identify the keys selected by users. The IR sensor array supported higher accuracy closer to the sensors, and lower accuracy for keys toward the fingertips. A preliminary study with 3 users showed that their typing speed was 4.6 words per minute using this prototype.

The rest of the paper is organized as follows: we first discuss related work and present our design sessions to optimize QWERTY layout for PalmType. We then describe our evaluation using Google Glass and the Vicon motion capture system, and our wearable prototype using Google Glass and IR sensors. We discuss our observations from the studies and experience with the prototypes, then conclude with our contributions and future work.

RELATED WORK

Palm- and Arm-based Interfaces

Because of humans' high proprioceptive sense of palms, hands, and arms and convenience in interacting with these surfaces, a number of projects have appropriated them for always-available input interface. PalmRC [9] used palm as a remote control for TVs with a small set of buttons, and numeric pads has also been proposed [10]. Gustafson et al. designed an imaginary phone interface that directly mapped a phone's UI to users' palms [12], which enabled users to interact with their mobile phone by recalling and touching locations on their palms that corresponded to the app icons on the phones. In addition, studies on palm-based imaginary interfaces showed that people can interact effectively without visual feedback [13].

Skinput [15] uses an array of acoustic sensors mounted on the upper arm to enable the forearm, palm, and fingers to be used as input surfaces. The location of a finger tapping the arm and hand can be identified by analyzing the acoustic waves propagating through muscles and bones. User interface elements can be projected onto these body surfaces to provide targets and visual feedback. OmniTouch [14] combines shoulder-mounted depth-sensing cameras and projectors to project UI elements and to detect touch events on any surface. This computer vision-based technique can be utilized for interactive touch applications on arbitrary surfaces, including users' bodies.

Compared to previous projects that focused on target selection from a small set of targets, PalmType focuses on text input using a familiar keyboard layout that has a large number of keys, and optimizing its layout for users' hands.

Mid-air Text Input

Mid-air text input techniques that were proposed for large displays could also be used for smart glasses, because its input space is separate from its output space [1]. GesText [18] detected mid-air pointing and gestures using the accelerometers in handheld Wii remotes and displayed different types of virtual keyboard layouts for text entry. Users achieved 3.7 words per minute (WPM) in first time use and 5.4 WPM after 4 days of practice. AirStroke [23] evaluated mid-air stroke-based text entry using gloves with fiducial markers, and users were able to reach 6.5 WPM after 2 weeks of practice. Compared to these techniques, PalmType's interaction is subtler and thus more socially acceptable. PalmType is also faster without training.

Virtual QWERTY Keyboards

QWERTY keyboard has been shown to perform well as a virtual keyboard on handheld devices [24]. Researchers have tweaked the QWERTY layout not only to leverage users' existing knowledge but also to bring subtle usability improvements. For example, quasi-QWERTY [5] and QWERTY-like [16] layouts allowed keys to be relocated a limited distance from their traditional positions. The 1line keyboard [21] compressed columns of keys on a QWERTY virtual keyboard down to a single row of 8 keys, to take up less screen real estate while retaining the horizontal layout and familiarity of QWERTY. PalmType explores how QWERTY can be optimized for palm-based interaction.

Wearable Sensing Techniques

KITTY [20] used a glove instrumented with electronic contacts to enable the detection of finger touches. Mobile Lorm Glove [11] also used the palm for elaborate input/output for the deafblind. However, users are required to wear gloves which affect how they interact with everyday objects.

Han et al. showed that it is possible to calculate a magnet's 2D position using a pair of magnetometers if one imposes some constraints on movement [29, 30]. uTrack [8] uses a pair of magnetometers on the back of a finger to senses a magnet that is affixed to the thumbnail for continuous 3D input. While magnetic-based sensing does not require line-of-sight, the tradeoff is that users are required to affix a magnet to the pointing thumb/finger in addition to wearing the magnetometers.

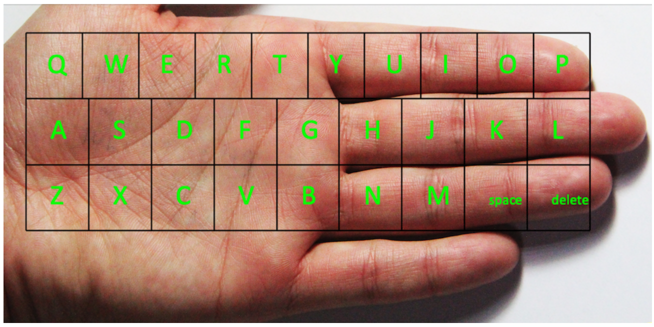

Figure 2. Mapping a rectangular QWERTY keyboard onto users' hands.

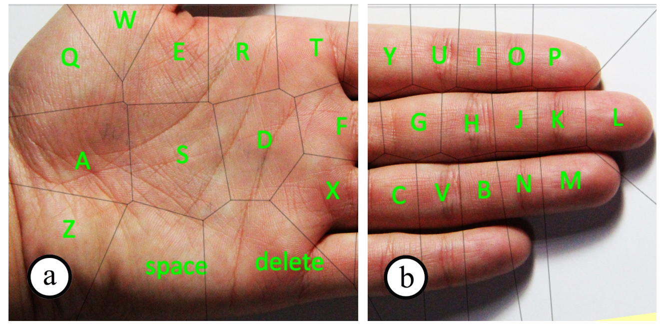

Figure 3. Mapping the user-designed, optimized QWERTY keyboard onto users' hands.

Optical and computer vision techniques have also been explored for hand input. Brainy Hand [26] used a head-mounted camera to capture images of users' hands to recognize its movements as input gestures, users had to keep their hands within the camera's field of view.

Infrared (IR) proximity sensors have been used to support virtual multi-touch interactions around the body of a small mobile device [6]. Gesture Pendant mounted an IR camera in a pendant form factor to track users' hands [27]. Digits mounted an IR camera on the wrist with a IR laser line to sense the finger movements [19]. Harrison et al. used a depth camera mounted on the shoulder [14]. Nakatsuma et al.[22] detected the touch position on the back of the hand by measuring the IR reflection from the finger. In our paper, we demonstrate the feasibility of PalmType by using wrist-worn infrared sensor arrays to detect users' finger position and taps, and providing visual feedback via Google Glass.

PALMTYPE DESIGN

Our design goals for PalmType are to provide intuitive and efficient text entry for smart glasses. QWERTY keyboard is the most popular keyboard layout, and most users already have significant experience from using virtual keyboards and physical keyboards on smartphones, tablets, and PCs.

While previous studies PalmRC [9] have shown that there are 9 landmarks on the palm based on users' proprioceptive abilities, a full keyboard has significantly more keys including 26 English characters plus at least the Space and Delete keys. As shown in Figure 2, a direct mapping of a rectangular QWERTY keyboard with equally sized and spaced keys is one possible layout, but it does not leverage proprioception.

Design Sessions

We conducted design sessions with 6 participants (3 males/females) to understand how users mapped a QWERTY keyboard to their hands. Participants previous experiences with QWERTY keyboards were between 4 and 6 (M = 5.17) on a seven-point Likert scale from "1 - No experience" to"7 - Expert". Participants' age ranged from 20 to 23 (M = 21.67).

Firstly, we took a photo of each participant's non-dominant hand and imported it into a Apple Keynote presentation slide as the background. Secondly, participants were informed of the concept of proprioception and the 9 salient regions reported in PalmRC [9]. Thirdly, we asked participants to think aloud and to position the four corner keys on their hands (Q, Z, P, and M), followed by the remaining keys. Lastly, as participants determined the center position of each key, we overlaid that key on top of the hand image in the Keynote slide and recorded any comments they made.

After all participants finished placing all the keys, they were asked to describe their own keyboards to other participants. During their discussion, they could adjust the keyboard layout iteratively on the Keynote slide until they were all in agreement on a feasible layout. When a possible layout of keyboard was proposed, all participants were asked to type random words by pointing to the corresponding key positions on their hands to test the performance of the keyboard. Although participants had varied opinions on QWERTY layout, the informed design process took about 90 minutes until participants were satisfied with all the keys on the final layout of the keyboard.

Results

All participants used the full hand width including their palm and fingers as the typing surface, as there were too many keys to fit onto only the palm or only the finger areas. As shown in Figure 2, participants mapped the rectangular keyboard using the full width of the hand and aligned the character keys along the left edge of the palm.

The optimized layout is shown in Fig.3a), P and M keys were positioned on the fingertips of index finger and ring finger respectively, and Q and Z were positioned on the top left corner and bottom left corner of the palm.

For the palm area, Q, Z, T keys were located at the convex corner landmarks. However, most of the participants placed the X key on the convex next to the ring finger rather than next to the little finger. The rest of keys on the palm were placed in the space order of QWERTY. Moreover, several participants stated that it was hard to distinguish W from E when W was placed in the same row, and they therefore suggested moving W to a higher position relative to the adjacent Q and E.

For the finger area, participants could accurately point to the creases over the interphalangeal joints even without looking at their hands. As shown in Figure 3b), all participants placed 5 keys for each finger that corresponded to the 3 finger segments and the 2 creases/joints.

Figure 4. User study setup for (a) PalmType and (b) touchpad, with fiducial markers used by the Vicon 3D motion tracking system that tracks finger position and touch events in real-time.

EVALUATION

We conducted a 12-person study to compare typing performance and user preference for 3 types of keyboards for smart glasses: 1) virtual keyboard using touchpad with relative pointing, which is used by the Epson Moverio glasses, 2) PalmType with rectangular QWERTY layout, and 3) PalmType with optimized QWERTY layout.

Implementation using Motion Tracking System

Our goal is to evaluate the performance of the interaction techniques, without being affected by noise introduced by sensors. Therefore, we developed PalmType keyboards using high-precision Vicon 3D motion tracking system that tracks reflective markers to locate users' touch events in real-time. Three markers were placed on the same plane as the non-dominant hands of the participants, and one marker was placed at the tip of their pointing index fingers. Each participant first calibrated the system with 3 points in order to find the plane of their non-dominant hands, then with 2 sets of 28 points corresponding to all the keys on the two QWERTY layouts.

We recorded the precise positions as well as the vertical distances from the pointing fingertip to the palm for all 28 keys. We used a 3mm vertical distance threshold to detect a touch event, and users had the flexibility to adjust finger position while in the touch-down state. A key was typed at the moment users lift their fingers above the threshold. For the touchpad keyboard, we used the touchscreen of a Google Galaxy Nexus phone to track the touch events because Vicon could not track the rapid clutching when using the touchpad.

The touch events from Vicon for PalmType and from the Nexus phone for the touchpad are processed then sent to Google Glass via WebSocket and displayed.

Design

We used a within subjects 3x2 factorial design with two independent variables: input method (Touchpad, rectangular QWERTY layout, and optimized QWERTY layout) and input task (phrase and word). The order of input methods and input tasks were counter balanced.

There were two phases in the input tasks: phrase input and word repetition [4]. The former simulated rectangular text entry tasks, while the latter investigated the learnability and

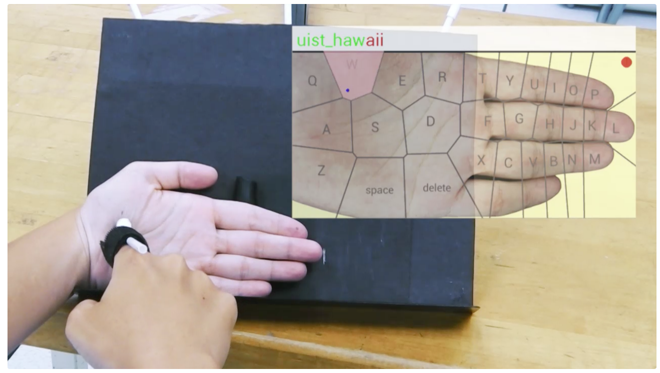

Figure 5. Example view when using PalmType with optimized QWERTY layout during the study.

expert input speed of each input method. The main measures were Input Speed (WPM), Not Corrected Error Rate, and Corrected Error Rate [25].

Participants practiced each input method before starting the actual trials. Participants started each trial by swiping the touch-strip located on the right side of Google Glass. When the trial was completed, participants swiped again to stop the timer and waited for the next phrase or word. Participants were asked to wear a mask to obscure their view of their hands but not of the view in front of them.

Phrase Input

Participant entered 8 phrases for each of the 3 input methods. The phrases were chosen from mobile text [28] , a collection of mobile email sentences written by actual users on actual mobile devices. We randomly chose 40 phrases as the test set, all of which contained only the 26 English characters and the space key. Participants were allowed to correct the current word by using the delete key.

The input speed was calculated as:

$$WPM = \frac{|T|}{S \times 1000} \times 60 \times \frac{1}{5}$$

where T was the final transcribed string and S was the elapsed time in milliseconds as recorded by Google Glass. Note that the numerator was $|T|$, instead of $|T| - 1$ [17], because the time of inputting the first character was also included.

The error rates were:

$$Not\ Corrected\ Error\ Rate = \frac{INF}{C + INF + IF}$$

$$Corrected\ Error\ Rate = \frac{IF}{C + INF + IF}$$

where C is the total number of correct words, IF is the number of incorrect but fixed (deleted) words, and INF is the number of incorrect (but not fixed) words. The error rate were calculated at the character-level.

Word Repetition

Participants entered 4 words for each of the 3 input methods, and repeated the same word 6 times. Examining how input speed progressed as a word was repeatedly entered revealed

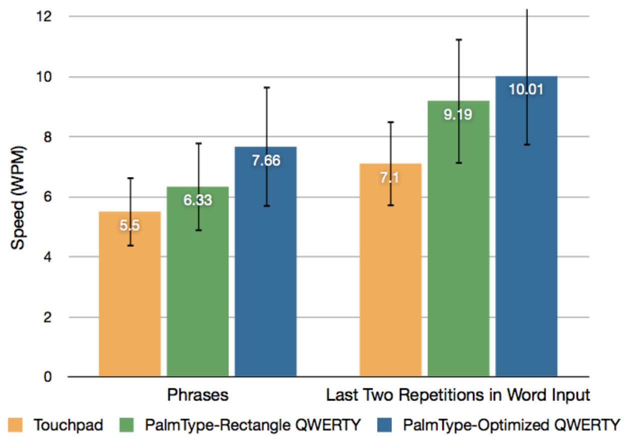

Figure 7. Average input speed (in words per minute) for the 3 input methods for phrases and for simulated expert speed. The two PalmType QWERTY layouts are 15% and 39% faster than touchpad for phrases, and 29% and 41% faster than touchpad for experts (bar represent standard deviation).

the learnability of a input method. Also, assuming that a user became familiar with the gesture pattern of the word after repeating it for multiple times, the last one or two repetitions simulated the expert input speed a user can achieve [5, 4]. Also, they were allowed to correct the current word using the delete key.

The input speed was calculated as:

$$WPM = \frac{|W|+1}{S \times 1000} \times 60 \times \frac{1}{5}$$

where $|W|+1$ was the length of the target word plus a space character. S and the error rates were the same as those in Phrase Input.

Participants

We recruited 12 participants (5 male, 7 female) between ages of 18 and 22. Each participant entered 3x8=24 phrases and 3x4x6=72 words. The study lasted around 1.5 hours for each participant.

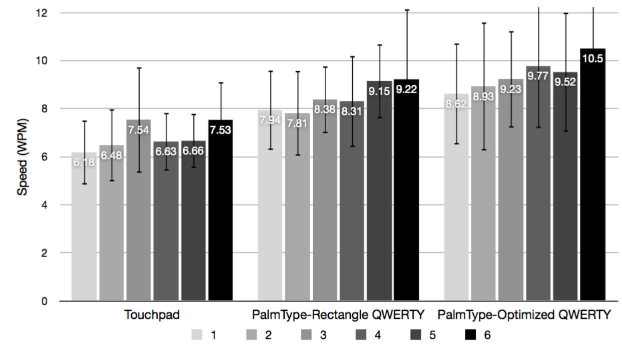

Figure 8. Average input speed (in words per minute) for each of the 6 rounds of word repetition for the 3 input methods (bar represent standard deviation).

Results

As shown in Figure 7, PalmType with rectangular and optimized QWERTY layouts were 15% and 39% faster than touchpad for phrases, respectively. ANOVA showed that input method had a significant effect on the input speed for both phrase input and word repetitions ($F(2, 30) = 5.978$, for phrase, $F(2, 30) = 7.573$, for last two repetitions; $p < 0.01$). Pairwise comparisons (Tukey's) showed significant differences between optimized and rectangular QWERTY ($p < 0.05$) and between optimized QWERTY and touchpad ($p < 0.01$). With Optimized QWERTY, the typing speed of participants was 39% faster than touchpad and 21% than rectangular QWERTY.

For the last two word repetitions, pairwise comparisons (Tukey's) showed significant differences between both PalmType layouts and touchpad ($p < 0.01$ for optimized QWERTY, $p < 0.05$ for rectangular QWERTY). Participants achieved 10 words per minute (WPM) using PalmType with optimized layout, which was 41% faster than touchpad, and the rectangular layout was 29% faster.

To understand the learnability of each input method, we evaluated the input speed of each round of repetition, for each type of keyboards separately, as shown in Figure 8. ANOVA

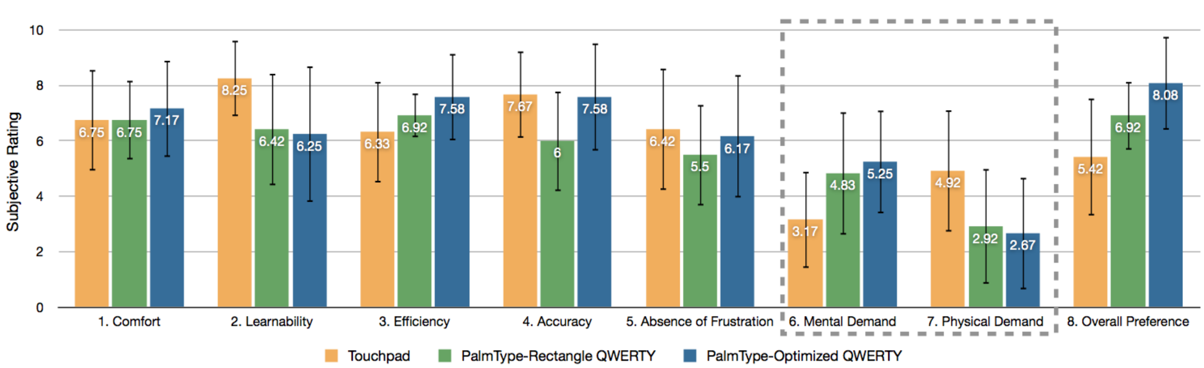

Figure 6. Mean (SD) of subjective rating on comfort, learnability, efficiency, accuracy, absence of frustration, mental, physical demands, and overall preference, using a scale of 1-10. For measures 1-5, and 8, higher rating is better. For measures 6 and 7, lower rating is better (bar represent standard deviation).

showed significant improvement in speed for optimized QWERTY layout ($F(5, 50) = 7.74, p < 0.01$).

Not Corrected Error Rate and Corrected Error Rate

ANOVA showed that the type of keyboards had no significant effect on Not Corrected Error Rate ($F(2, 30) = 2.768, p = 0.079$, for phrase, $F(2, 30) = 2.365, p = 0.111$ for last two repetition), nor on Corrected Error Rate was shown ($F(2, 30) = 0.024, p = 0.976$ for phrase, $F(2, 30) = 1.409, p = 0.260$ for last two repetition), indicating that the accuracy of text input by participants with different types of keyboards was at the same level for both phrases and word repetition tasks.

The mean (SD) of Not Corrected Error Rate for phrases was 1.58% (1.65%) for optimized QWERTY, 0.87% (1.22%) for rectangular QWERTY, and 0.61% (0.57%) for touchpad.

The mean (SD) of Corrected Error Rate for phrases was 1.74% (1.51%) for optimized QWERTY, 1.79% (1.10%) for rectangular QWERTY, and 1.97% (1.31%) for touchpad.

Subjective Measures

As shown in Figure 6, participants rated each type of keyboards with level of comfort, learnability, efficiency, accuracy, absence of frustration, mental, physical demands, and overall preference using 1-10 scales. ANOVA showed that the type of input methods had significant effect on the rating of accuracy ($F(2, 33) = 3.372, p < 0.05$), mental demand ($F(2, 33) = 3.832, p < 0.05$), physical demand ($F(2, 33) = 4.168, p < 0.05$), and overall preference ($F(2, 33) = 7.314, p < 0.01$). Although PalmType with optimized QWERTY had the lowest scores in learnability and mental demand, participants rated it as the most comfortable and efficient type after they got familiar with it.

As shown in Figure 9, when asked about the most preferred type of keyboards for smart glasses, 92% of the users chose PalmType, among whom 81% preferred the optimized layout. In terms of overall preference, the mean rating (SD) of PalmType with optimized QWERTY was 8.08 (1.68), PalmType with rectangular QWERTY was 6.92 (1.24), and touchpad was 5.42 (2.11).

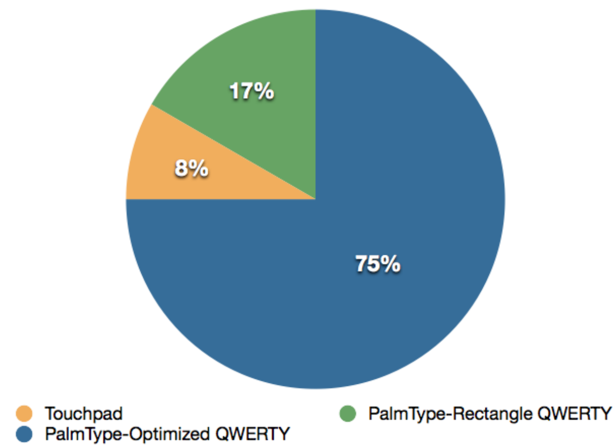

Figure 9. Participants' preference for the 3 input methods. 92% preferred PalmType vs 8% for touchpad.

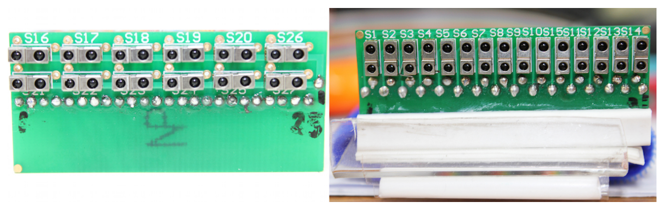

Figure 10. Array of infrared proximity sensors in a) two rows of 6 sensors in horizontal orientation, and b) 15 sensors in vertical orientation.

WEARABLE PALMTYPE PROTOTYPE

To demonstrate the feasibility of wearable PalmType, we developed two prototype using wrist-worn sensors, as shown in Figure 10. An array of discrete infrared (IR) proximity sensors face toward the direction of the palm, emitting infrared light from the IR LEDs and sensing the reflected IR with paired IR photodiodes to measure the distance between the sensors and the object on its optical path.

Sensor Board

The sensor board consists of Avago HSDL-9100-021 940nm IR proximity sensors. Each of the sensor contains an IR emitter and paired IR photodiode housed in a small SMD form factor. As shown in Figure 10, we experimented with two layouts including: a) 2 rows of 6 proximity sensors in horizontal orientation with 2mm spacing, and b) a single row of 15 proximity sensors in vertical orientation with 1mm spacing. An Arduino Mega board was used to generate the input PWM signal required for the proximity sensors and read the analog output sensor values.

The detection range reported in the official data sheet is from near zero to 60mm. Nevertheless, average hand size is around 180mm. We paired the sensors with Avago Signal Conditioning IC ADPS-9700 and also attached highly reflective stickers to users' index fingers to extend the detection range sufficiently. Because the light path of the IR LEDs have to overcome the height of the closest edge of the palm, we boosted the height of the sensors by 15mm, for a total height of 25mm. Our experiments showed that the array of 15 sensors in vertical orientation had less noise and interference.

Touch Event Detection Accuracy

We used LIBSVM [7], a Support Vector Machine library for key position recognition. We used 1-vs-1 multi-class classification and the radial basis function (RBF) kernel. We treated each sensor reading as a 10-bit input and combined them into a 15-dimension feature vector.

We conducted a preliminary evaluation with 3 participants. Overall, we collected 3 users 50 samples 29 (28 keys and lifting) classes = 4350 samples in total. We ran a 10-fold subject-independent (leave-one-out) cross-validation, and the overall accuracy was 74.5%. The sensors are more sensitive in closer range, with the accuracy within the palm region being 82.6% and within the finger region being 46.2%.

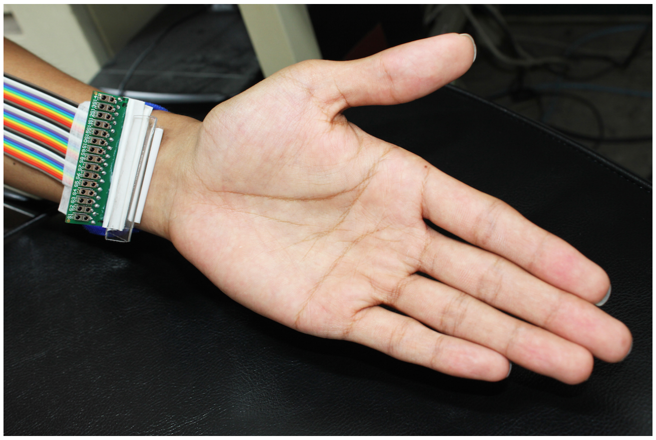

Figure 11. Wearable PalmType prototype with wrist-worn IR sensors.

Typing Performance

We used the same study design as our previous user study, and the average typing speed for phrases was 4.6 words per minute. To understand how detection accuracy affects typing performance, we additionally asked participants to type words that only used keys in the palm region and words that only used keys in the finger region. We observed significant difference in typing speed between the two regions, with 6.2 WPM for the palm region and 4.3 WPM for the finger region.

DISCUSSION

Additional Keyboard Layouts

We have evaluated two QWERTY layouts for PalmType, and there are many more possible keyboard layouts that we could explore. Participants have suggested multi-tap and T9 text entry, which were the standard text entry techniques for mobile phones with physical keypads before touchscreen smartphones became popular. These two techniques require 12 keys, which should map well onto just the palm region.

IR Proximity Sensor Sensitivity

There were two issues that we observed with the IR proximity sensors used in our wearable prototype. First was cross talk between the sensors due to imprecise alignment. The sensors were mounted onto the printed circuit board (PCB) using surface-mount technology (SMT), and the process we used was not precise enough and introduced variation in each sensor's alignment. We plan to remove the spacing between the sensors, which should improve the sensor alignment and also increase sensor density.

Second, the proximity sensors were not designed for our purposes and had a maximum range specification of 60mm, which was significantly shorter than the average palm width of 180mm. Although we resorted to boosting the current and adding reflectors to increase range, the gradual sensitivity curve of the photodiode led to noisier readings when pointing in the finger region. Using photodiode that have wider dynamic range and improved sensitivity curve should help reduce system error and improve typing performance and experience.

LIMITATION AND FUTURE WORK

Our current wearable prototype with IR sensors has limited range and lower detection accuracy in the finger region. We plan to experiment with stereoscopic IR cameras, similar to Leap Motion, that should provide more robust distance and location sensing than using arrays of proximity sensors. It should also help us reduce the PCB height from 25mm to the 10mm range, because we should be able to estimate the touch location as long as the upper half of the pointing finger is visible.

PalmType currently requires both hands to be available in order to enter text. We are exploring keyboard designs and sensing techniques for single handed input. We are also exploring adding word prediction and auto-correction, such as Swype, to PalmType to improve typing speed and user experience.

CONCLUSION

We have presented PalmType, which uses palms as keyboards to enable intuitive and efficient text entry for smart glasses. We conducted design sessions with 6 participants to see how they map familiar QWERTY keyboard onto their palms. Our 12-person user study showed that PalmType's two QWERTY layouts were 15% and 39% faster in typing speed, compared to touchpad-based virtual keyboard used by current smart glasses. In terms of user preference, 92% of the users preferred PalmType vs touchpad. We also implemented a wearable prototype using a wrist-worn array of IR proximity sensors to demonstrate the feasibility of PalmType.

REFERENCES

1. Book review: Handbook of human-computer interaction ed. by martin helander (north-holland, amsterdam 1988). 81.
2. Epson moverio bt-100. `https://www.epson.com/cgi-bin/Store/jsp/Moverio/Home.do`. Accessed: 2010-09-30.
3. Google glass. `http://www.google.com/glass/start/`. Accessed: 2010-09-30.
4. Bi, X., S. B., and Zhai, S. Multilingual touchscreen keyboard design and optimization, human computer interaction.
5. Bi, X., Smith, B. A., and Zhai, S. Quasi-qwerty soft keyboard optimization. In *Proc. ACM CHI'10 (2010)*, 283–286.
6. Butler, A., Izadi, S., and Hodges, S. Sidesight: Multi-"touch" interaction around small devices. In *Proc. ACM UIST'08 (2008)*, 201–204.
7. Chang, C.-C., and Lin, C.-J. Libsvm: A library for support vector machines. 27:1–27:27.
8. Chen, K.-Y., Lyons, K., White, S., and Patel, S. utrack: 3d input using two magnetic sensors. In *Proc. ACM UIST'13 (2013)*, 237–244.
9. Dezfuli, N., Khalilbeigi, M., Huber, J., Mller, F., and Mühlhäuser, M. Palmrc: Imaginary palm-based remote control for eyes-free television interaction. In *Proc. ACM EuroiTV '12 (2012)*, 27–34.

10. Goldstein, M., and Chincholle, D. Finger-joint gesture wearable keypad. In *Proc. ACM MobileHCI'99 (1999)*, 9–18.
11. Gollner, U., Bieling, T., and Joost, G. Mobile lorm glove: Introducing a communication device for deaf-blind people. In *Proc. TEI'12 (2012)*, 127–130.
12. Gustafson, S., Holz, C., and Baudisch, P. Imaginary phone: Learning imaginary interfaces by transferring spatial memory from a familiar device. In *Proc. ACM UIST'11 (2011)*, 283–292.
13. Gustafson, S. G., Rabe, B., and Baudisch, P. M. Understanding palm-based imaginary interfaces: The role of visual and tactile cues when browsing. In *Proc. ACM CHI'13 (2013)*, 889–898.
14. Harrison, C., Benko, H., and Wilson, A. D. Omnitouch: Wearable multitouch interaction everywhere. In *Proc. ACM UIST'11 (2011)*, 441–450.
15. Harrison, C., Tan, D., and Morris, D. Skinput: Appropriating the body as an input surface. In *Proc. ACM CHI'10 (2010)*, 453–462.
16. Hwang, S., and Lee, G. Qwerty-like 3x4 keypad layouts for mobile phone. In *Proc. ACM CHI EA'05 (2005)*, 1479–1482.
17. I. Scott MacKenzie, K. T.-I. In *Text Entry Systems: Mobility, Accessibility, Universality*,
18. Jones, E., Alexander, J., Andreou, A., Irani, P., and Subramanian, S. Gestext: Accelerometer-based gestural text-entry systems. In *Proc. ACM CHI'10 (2010)*, 2173–2182.
19. Kim, D., Hilliges, O., Izadi, S., Butler, A. D., Chen, J., Oikonomidis, I., and Olivier, P. Digits: Freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *Proc. ACM UIST'12 (2012)*, 167–176.
20. Kuester, F., Chen, M., Phair, M. E., and Mehring, C. Towards keyboard independent touch typing in vr. In *Proc. ACM VRST '05 (2005)*, 86–95.
21. Li, F. C. Y., Guy, R. T., Yatani, K., and Truong, K. N. The 1line keyboard: A qwerty layout in a single line. In *Proc. ACM UIST'11 (2011)*, 461–470.
22. Nakatsuma, K., Shinoda, H., Makino, Y., Sato, K., and Maeno, T. Touch interface on back of the hand. In *ACM SIGGRAPH 2011 Emerging Technologies*, 19:1–19:1.
23. Ni, T., Bowman, D., and North, C. Airstroke: Bringing unistroke text entry to freehand gesture interfaces. In *Proc. ACM CHI'11 (2011)*, 2473–2476.
24. Shoemaker, G., Findlater, L., Dawson, J. Q., and Booth, K. S. Mid-air text input techniques for very large wall displays. In *Proc. CIPS GI'09 (2009)*, 231–238.
25. Soukoreff, R. W., and MacKenzie, I. S. Metrics for text entry research: An evaluation of msd and kspc, and a new unified error metric. In *Proc. ACM CHI'03 (2003)*, 113–120.
26. Tamaki, E., Miyaki, T., and Rekimoto, J. Brainy hand: An ear-worn hand gesture interaction device. In *Proc. ACM CHI EA'09 (2009)*, 4255–4260.
27. Thad Starner, Jake Auxier, D. A., and Gandy, M. The gesture pendant: a self-illuminating, wearable, infrared computer vision system for home automation control and medical monitoring. In *Proc. IEEE ISWC '00 (2000)*, 87–94.
28. Vertanen, K., and Kristensson, P. O. A versatile dataset for text entry evaluations based on genuine mobile emails. In *Proc. ACM MobileHCI'11 (2011)*, 295–298.
29. Xinying Han, Hiroaki Seki, Y. K., and Hikizu, M. Wearable handwriting input device using magnetic field. In *SICE '07*, 365–368.
30. Xinying Han, Hiroaki Seki, Y. K., and Hikizu, M. Wearable handwriting input device using magnetic field: 2nd report: Influence of misalignment of magnet and writing plane. In *PRECIS ENG '10*, 37–43.