EyeRing: A Finger-Worn Input Device for Seamless Interactions with our Surroundings

Suranga Nanayakkara[1], Roy Shilkrot[2], Kian Peen Yeo[1] and Pattie Maes[2]

[1]Singapore University of Technology and Design, 20 Dover Drive, Singapore, 138682

[2]MIT Media Lab, 75 Amherst Street, Cambridge, MA, 02142

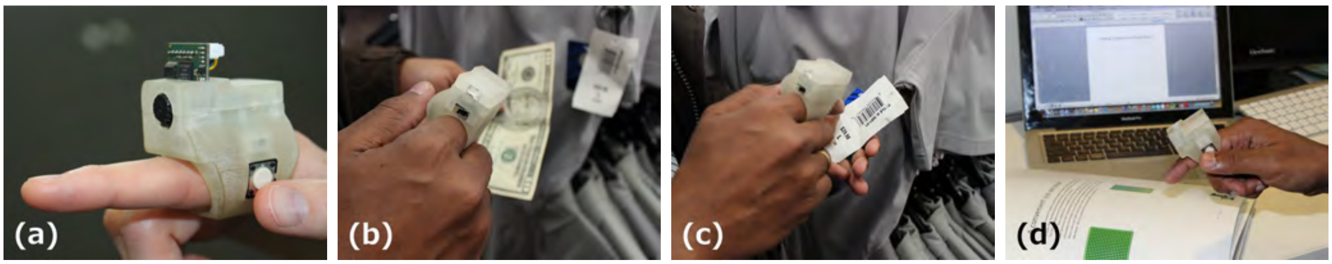


Figure 1. EyeRing: A finger-worn input device. (a) EyeRing prototype. (b) CurrencyDetector application. (c) TagDetector application. (d) Interaction with printed media.

ABSTRACT
Finger-worn interfaces remain a vastly unexplored space for user interfaces, despite the fact that our fingers and hands are naturally used for referencing and interacting with the environment. In this paper we present design guidelines and implementation of a finger-worn I/O device, the EyeRing, which leverages the universal and natural gesture of pointing. We present use cases of EyeRing for both visually impaired and sighted people. We discuss initial reactions from visually impaired users which suggest that EyeRing may indeed offer a more seamless solution for dealing with their immediate surroundings than the solutions they currently use. We also report on a user study that demonstrates how EyeRing reduces effort and disruption to a sighted user. We conclude that this highly promising form factor offers both audiences enhanced, seamless interaction with information related to objects in the environment.

Author Keywords
Pointing-based Interaction; Wearable Input Devices; Intuitive Interfaces.

ACM Classification Keywords
H5.2 Information interfaces and presentation: User Interfaces–input devices and strategies
K4.2 Social IssuesAssistive technologies for persons with disabilities

General Terms
Design; Human Factors; Algorithms; Performance

INTRODUCTION
The pointing gesture is fundamental to human behavior [13] and used consistently across cultures [14]. It begins at an early developmental stage [3] and lets humans reference proximal objects as well as abstract concepts in the world. Pointing gestures, which are part of our *gestural language*, are inherently used for *interaction*. This is a strong motivation for designing finger-worn devices for interaction: not only can we build upon a natural and universal human behavior, but we also benefit from the focus and intentionality in the gesture. Given the recent increased interest in using speech to interact with mobile devices [28], it is a logical next step to support a user in pointing at an object while stating a comment or asking a question about it.

The motivation for visually impaired to use pointing interfaces exists, although it stems from a different perspective. Turning to recent literature on interface design for the visually impaired, we note three desired qualities: assistive technology should be *socially acceptable*, work *coherently for disabled and non-disabled* alike, and also support *independent and portable interaction* [25, 26, 31, 19]. The finger-worn device presented here follows this design paradigm: it looks and offers the same affordances and mode-of-use to both sighted and blind users in a self-sufficient way. We deepen the discussion of design guidelines for sighted and visually impaired in the 'Design Considerations' section.

Among their many meanings, pointing gestures are perhaps most regularly used for referring to a place or a thing in space. In this paper, we propose a method for augmenting the pointing gesture for information retrieval tasks. Previous research work in the field of augmenting pointing gestures revolved around control [30] and information retrieval [15]. These works and others utilize a specialized sensor between the pointing finger and the target, such as an infrared

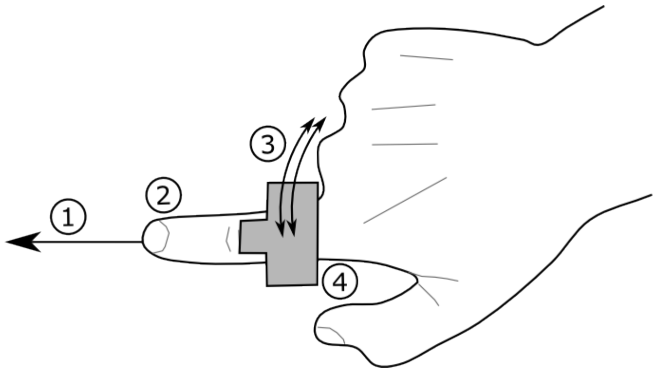


Figure 2. Types of ring-based interfaces. (1) Pointing. (2) Tapping. (3) Gesturing. (4) Pressing.

connection in the work of Merrill *et al*. [15]. This implies a pre-rigged environment, which inhibits natural interaction and limits the domain of use. We therefore chose to use a visible-light spectrum camera as the sensor. Naturally, this means the computational element of the system is more complex, as we use computer vision methods to extract information from images.

The EyeRing is an input device consisting of a camera mounted on a ring (typically worn on the index finger) with an embedded processor and wireless connection to a computation device (typically a smartphone). We employ a number of computer vision techniques to recognize objects or locations based on one or more images taken by the ring. The EyeRing system also has a speech recognition service to enable voice commands, and speech output (as well as screen output) as the means of communicating information. This platform enables a whole series of applications in which a user may acquire information or control objects/spaces in their proximity. For example a blind user could simply say 'currency' and point at a currency note to hear the value of it.

RELATED WORK

Finger-worn interaction devices received noteworthy attention over the last few years from HCI researchers and experience designers alike [10, 6]. Explorations of finger-worn interaction devices may be divided into a number of categories according to how they are operated, as illustrated in Figure 2. These finger gestures include (1) Pointing [15, 30]; (2) Tapping/Touching [8, 7, 32]; (3) Gesturing [12, 21, 34, 36]; (4) Pressing/Clicking On-Device [4, 9, 17].

Remotely controlling objects in the environment by pointing with a wearable device was implemented in the Ubi-Finger [30] and FieldMouse [27] projects. Efforts to attach and retrieve information from physical objects were implemented in [15] and recently in [32] using IR beacons and coded textures. However these applications often require the environment to be instrumented with sensors and markers, which limits the interactions to instrumented environments.

Handheld, see-through AR systems which present auditory feedback and overlay information on top of a user's field of view [1, 22, 23] are probably the closest related work to EyeRing's functionality. FingerSight [9] provides a visual-tactile substitution system by converting visual information into feedback, which is also embodied in the type of interaction suggested by EyeRing.

Generic multipurpose finger-worn input system were suggested by Chatterjee and Fumtoshi [4] who developed a device based on capacitive sensing, and more recently by Ogata *et al.* [17] who achieved the same with infrared sensors. Kim *et al.* [11] presented a similar interaction in the form of a wrist-worn device. These devices overcome several drawbacks of the common data-glove systems [18, 36], such as reduced physical contact between the hand and the environment, inconvenience of wearing and removal, etc. However, these works are not context-specific but rather focus on continuous interaction without referencing to the environment.

DESIGN CONSIDERATIONS

The design of Eyering follows guidelines set forth by Rekimoto [21] as well as by designers of assistive technology [25, 31].

Designing interaction with finger-worn devices

Rekimoto's work on *Augmented Interaction* proposed a number of guidelines for designing unobstructive wearable technology: *straightforward operation, using real-world situations as implicit input*, and ensuring that the technology is *socially acceptable*. These guidelines suggest that users should be able to operate the device without holding it and the device should allow for "quick changes between normal and operation modes." The goal of "[using] real-world situations as implicit commands" is embodied in our work through the use of a camera and computer vision methods. In addition, the device "should be as natural and (conceptually) unnoticeable as possible for use in various social settings." EyeRing is designed to look like a common wearable accessory, a wireless ring worn on the index finger (albeit a somewhat large one), to appear less conspicuous. In addition, we consider the following to be important design considerations: *leveraging the pointing gesture* and *minimal instrumentation of the environment*.

Leveraging the pointing gesture:

Pointing the index finger at something is a natural and universal deictic gesture [2] used to refer to an object and ask for or convey information about that object [14, 3]. EyeRing augments this natural behavior without obstructing it. We propose to leverage attributes of the pointing gesture, the focus of attention and the implied dialog, as guides for the device to support the interaction.

Minimal instrumentation:

Many input devices require the use of special purpose sensors, and they often only work in an instrumented environment (with markers, beacons, external sensors, etc.) [21, 30, 15]. A design that requires minimal instrumentation of the environment results in a more generic input device. However, a generic system requires additional features to make the interaction specific and focused, which we try to achieve in EyeRing by leveraging the pointing gesture to infer what the user is interested in.

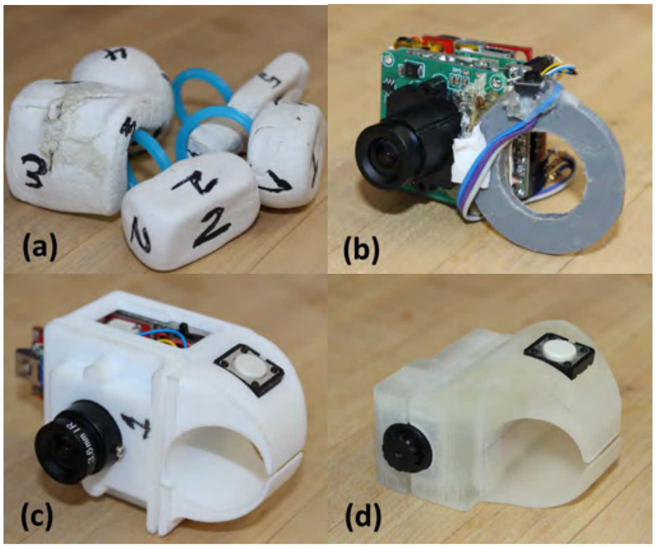


Figure 3. Prototypes of the EyeRing. (a) Clay models used for initial exploration. (b) First working prototype. (c) First prototype with plastic casing. (d) Second prototype.

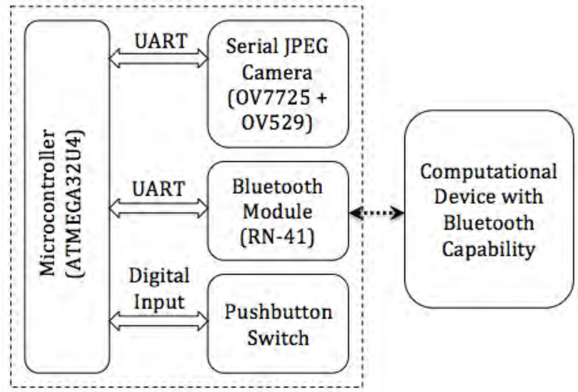


Figure 4. Overview of the EyeRing System.

Design of assistive devices for visually impaired

The design of EyeRing was informed by recent insights into the design of assistive technology for the visually impaired. The insights made by Shinohara through ethnographic research of assistive technologies struck the right chord with our intentions, most specifically their notions of *independence, portability and social acceptance*. “It is important to consider design ideas supporting cohesive socialization with .. people in [a] social sphere," Shionara claims [25] and goes on to say that “socially acceptable design might draw less unnecessary attention and change misperceptions about assistive devices." This is reiterated in Winberg’s work on collaboration using assistive technology: “Non-visual interfaces should be coherent with visual interfaces to enable collaboration" [31]. EyeRing is designed to support operation by both sighted and visually impaired users in the same fashion. It is worn on a finger and still allows one to use the hand and finger for feeling and holding. Even though it was not our primary concern, we also strive to make the device appealing for sighted people to wear, which inherently aligns with the claim that generic devices used both by sighted and non-sighted are far more successful [26].

Other design principles that resonated with us were *independence, portability* and *distinguishability of similars*. Applications we propose for visually impaired users are intended to increase self efficacy for blind users, and the small form factor of the ring ensures portability. By âĂIJdistinguishability of similarsâĂİ Shionara means “[ability] to distinguish among item with similar features" [25], which is why we focused on implementation of object recognition capabilities.

EYERING

EyeRing consists of a finger-worn device with an embedded camera and a computation element embodied as a smartphone or computer, which is also used for speech I/O. The finger-worn device is autonomous, wireless, and includes a single button to initiate the interaction. Information from the device is transferred via Bluetooth to the computing element where it is processed. An overview of the EyeRing system is shown in Figure 4.

Hardware design

The first working prototype of EyeRing used a JPEG Camera, AVR processor, Bluetooth module, polymer Lithium-ion battery, and push button switch. These components were attached onto a ring-shaped plastic piece (Figure 3b). This early working prototype enabled us to explore various application scenarios. Based on preliminary user reactions, we found a need to optimize the design, especially in terms of faster image acquisition and a smaller packaging. As a result, we came up with an improved hardware design for the second EyeRing prototype, which is discussed in detail below (Figure 3d).

Microcontroller:

We chose to use an Atmel 8 bit AVR (ATmega32U4) microcontroller because EyeRing only requires basic peripherals like digital input/output (I/O) and UART communication. A digital I/O pin is configured as an input and connected to a push button switch for user interaction. Two UARTs are used in the AVR. One is used for serial communication with an image acquisition module, and the other is used for setting up a Bluetooth communication channel.

Image acquisition module:

The EyeRing design uses an image acquisition module based on the OV7725 VGA CMOS sensor and the OV529 JPEG engine. It uses UART protocol to communicate with a microcontroller for setting up image properties and grabbing image data.

Wireless module:

In our design consideration, we require an always available, high speed wireless communication protocol. Bluetooth provides a good balance in terms of speed and availability compared to Wifi or Near Field Communication (NFC). Thus, the wireless communication between EyeRing and a mobile device is established using a Roving Networks RN-42 Bluetooth module with a baud rate of 115 kbps. This translates, under optimal conditions, to approximately 6 JPEG compressed images (with pixel resolution of 320 x 240) per second.

Software design

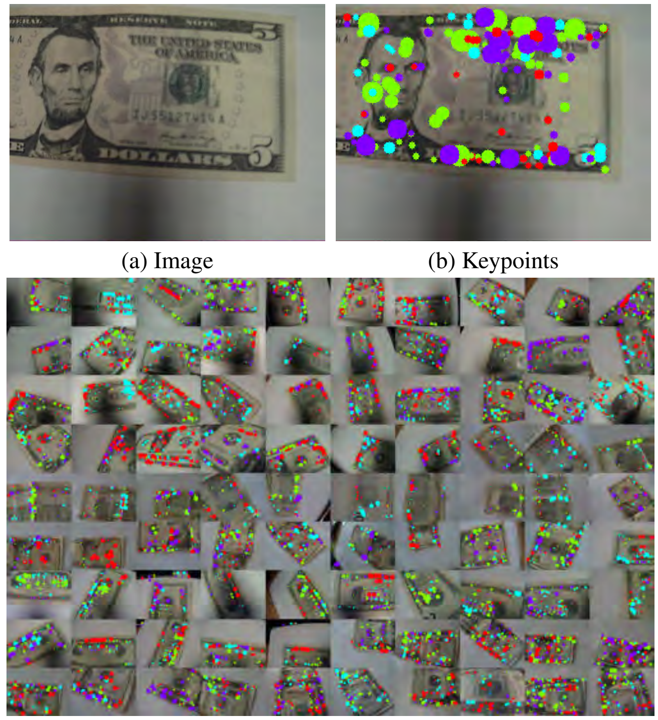
We developed a Bluetooth communication module that connects EyeRing with a smartphone running Android 2.2 or alternatively with a Notebook computer running Windows 7. These modules receive binary image data from the ring, as well as button click events. Some of the computer vision algorithms (e.g. currency recognition, tag recognition) were developed in-house, and we used a third party software [16] for general object recognition. At this point some of our computer vision software runs on the smartphone, while other applications run on the PC depending on the task at hand. We anticipate moving to an entirely smartphone platform for all applications in the near future. The software architecture (communication module and vision engine) allows for easy development of different applications.

Interaction flow

When used for the first time, EyeRing must be paired with the smartphone or PC application; however, this is done only once and henceforth a Bluetooth connection will be automatically established. Bearing in mind that the device should support both sighted and visually impaired users, we completely rely on non-visual interaction for all usage of the system. A typical interaction starts when the user performs a single click on the pushbutton switch located on the side of the ring using his or her thumb (Figure 3c, 3d). The type of analysis and corresponding response that follows depend on the selected application (currency, tag, insert, etc.) The user may change to a different application by double clicking the pushbutton and giving the system a brief verbal command that names the application, for example 'insert' (to insert some pictures previously taken into some online document), 'currency' (to recognize the value of a dollar bill), 'tag' (to recognize a price tag), and so on. The applications use a text-to-speech engine to provide audio feedback, hence providing a less disruptive interaction for sighted and visually impaired people alike.

EYERING ENABLED APPLICATIONS

The EyeRing system opens up the potential to build a great number of applications for people with vision impairments as well as for sighted people. In the following sections, we present detailed description of two proof-of-concept application scenarios: (1) a shopping assistant, to increase the independence of visually impaired persons in a shopping scenario; and (2) a desktop application providing a seamless copy-paste interaction for sighted people. On top of those, we experimented with a number of additional applications and use cases. One is an interactive application for children in a pre-reading stage that supports situated learning, by letting them read text on their own, before they can recognize alphabets or words. It has been shown that pointing at words while reading them aloud helps children learn faster [20]. When a text consists of many words, the EyeRing system assumes that the word they want read is the one at the tip of their finger. Another application currently in development builds upon the idea of I/O brush [24], where the ring is used as a 'paint brush' to capture a texture (for brush stroke) and to draw or paint around a screen or projected canvas. Many of these types of applications may exist for iPads, iPhones or similar devices, however the advantage of EyeRing is that it makes them instantaneous, requiring minimal effort and reducing shift of attention.



(a) Image (b) Keypoints

(c) Test-Set

Figure 5. CurrencyDetector process: (a) Scanned image. (b) Detected features. (c) Montage image showing our test set with detected features - different colors represent different classes.

A shopping assistant for blind users

CurrencyDetector:

Although currency detection applications for blind users already exist for smartphones [29], these applications require many steps to operate. Specifically, the user has to find the phone, unlock the screen, browse (using a sequential and hence slow auditory approach) to the right application, open the application, take a picture, listen for the answer, turn the phone off and put it away. In contrast, EyeRing requires a fewer number of steps, simply pointing to a currency note and clicking the button, while the other hand is free to hold the note. The system generates synthetic speech output indicating the monetary value of the note. If the currency is not recognized, an error message prompts the user to take another picture. Our EyeRing currency detector application is intended to help a user to identify USA currency bills ($1, $5, $10, $20, $100), although it is easily extendable to other currencies. A detection algorithm based on a Bag of Visual Words (BoVW) approach [5] makes a prediction on the type of note from the input image. Grayscale pyramid SURF features were used. Initially, the vocabulary was trained to be 1000 features long and then reduced by attribute selection to 170 features. A multi-class SVM (with RBF kernel) was used for classifying. The training dataset consists of 800 images under different lighting conditions and distances, 100 samples were held out for parameter tuning, and the rest were used in a 10-fold cross-validation scheme. For testing, an additional 270 images were used. The overall recognition rate is roughly 92% with a 0.905 kappa statistic. Figure 5 illustrates the currency detection process.

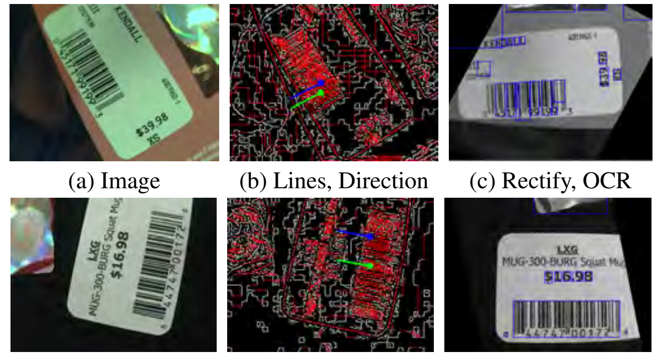
TagDetector:
This application intends to assist people with vision impairments in reading price tags on store products. It is based on searching the input image for cues of tag existence, and then extracting textual features or barcodes that allow for retrieving the price. Other applications such as the popular barcode scanner applications (similar to currency detection applications) on smartphones offer the same capabilities, however they too require the user to go through significantly more steps. Using the pointing gesture, we provide a more natural way of aiming the camera at the price tag and getting the result with a single-click operation.

Most product price tags include a UPC-type barcode, as it is a worldwide standard, and the price is usually indicated in a parallel or orthogonal alignment in relation to it (Figure 6a, 6d). We developed an automatic method to detect the orientation of the barcode in the image to extract the indicated price. A number of researchers recently used edge images and Hough Transform to locate barcodes in images [35, 33], therefore a similar approach was chosen. First a combination of second-order Sobel filters were performed, and then a probabilistic Hough line transform. A line-direction histogram is calculated for each cell of a regular grid over the image. The histogram has 32 bins for line angles, and the grid is of 10x10 cells. A scoring scheme is used to rank the cells:

$$Score(i,j) = \frac{\max_{\theta} Bin_{i,j}(\theta)}{\sum_{Bin_{i,j}(\theta)>0} 1}$$

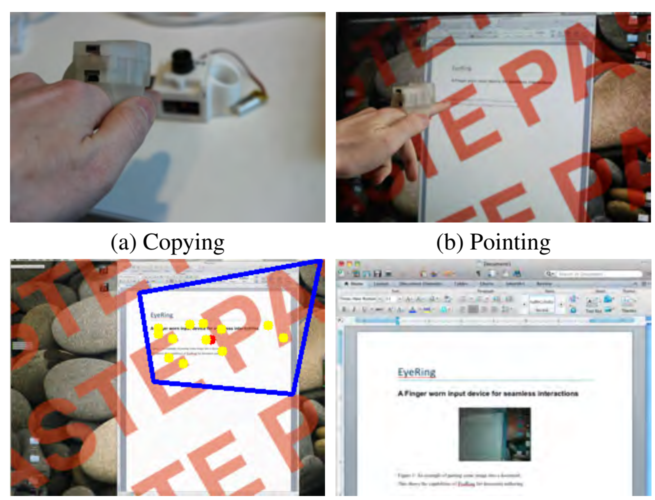
Where $Bin_{i,j}$ is the angles histogram vector, containing for each angle θ the number of lines in cell i, j agreeing with that angle. The score therefore highly ranks a cell with maximum agreeing lines and minimum possible angles. Unless there is significant interference, this corresponds to a lines-barcode with high probability in our test-set (Figure 6b, 6e). If fewer than 5 agreeing lines are found, the image is deemed not to contain a barcode. Finally the image is rotated about the center in 4 possible rotations. The next step is Optical Character Recognition (OCR) on the rectified images (Figure 6c, 6f) to recover any trace of a price mark. A '$' sign in the recovered text serves as an indicator, assuming the price is written to its right. If a price is found, it is spoken to the user, else an error message of either "No Barcode Found" or "No Price Extracted" is played back.

Initial reactions from visually impaired users
During initial development, we brought an EyeRing prototype to a group of people with vision impairments, who are particularly interested in assistive technology. They were given the EyeRing prototype and told how they could use



(a) Image (b) Lines, Direction (c) Rectify, OCR
(d) Image (e) Lines, Direction (f) Rectify, OCR
Figure 6. TagDetector process.

it. Some mentioned a few challenges they have when outside of a well-known environment: navigation, object recognition and reading printed text (which aids both in navigation and recognition) as the tasks which they need most help with. The idea of the shopping assistant was appealing to them because it helps distinguish objects. However, they raised a few concerns such as hesitation of using a camera as they have little to no experience taking photos, and using the ring with basic phones or different operating systems. When asked to comment on the EyeRing interactions, many users commented that "It (EyeRing) can't get any easier." We observed that the pointing gesture was intuitive to use; however a continuous feedback to assist in pointing at the right object might make it even better. We are in the process of incorporating continuous-auditory feedback into EyeRing. We also conducted a shopping scenario case study of EyeRing with a blind user who had never used the device before. He easily adapted to using EyeRing and managed to select the 'Cheez-Its' crackers box among many other morphologically similar products.



(a) Copying (b) Pointing
(c) Features detection (d) Pasting
Figure 7. CopyPaster process.

A desktop assistant for sighted users
CopyPaster:
Embedding an image of a proximal object into a digital document commonly involves numerous devices and operations:

a camera, smartphone or scanner for capturing, then email, USB drive or a cloud service for transferring, finding the correct place in the document, and lastly using the word processor's commands for embedding. EyeRing offers a simplified interaction to achieve all of the above steps with minimal effort and direct action. A user would simply point his or her EyeRing to capture any image or text in their environment, directly navigate it on the screen to the required position, and paste that information into a document authoring application. The user may optionally scale and rotate the data to the desired transformation, directly by rotating and moving the finger towards the screen. A button click commits the image to the document, similar to the ubiquitous copy-paste operation.

We implemented the interaction by continuously matching SURF features from the camera and those of the screen. A homography relationship is estimated via a robust estimator, and this allows for understanding the position, scale and rotation of the image. For improving the tracking of features on the screen, as well as giving visual feedback, a memory-resident application projects a semi-transparent pattern on the screen at the time of pasting, as well as a 'ghost' of the pasted image. See Figure 7 for an illustration.

EYERING VS SMARTPHONE EXPERIMENT

EyeRing is designed to be an 'immediate' interface. In other words, it should require a minimal number of steps to accomplish a task compared to a smartphone, where a user would have to browse to an application, launch it, point the camera, etc. As a result, the cost-benefit ratio of this input device is better. Thus we hypothesize that EyeRing may be a faster device to identify single objects, even for sighted people. We conducted a study to compare EyeRing currency recognizer against a state-of-the-art smartphone application, LookTel [29].

Participants and Apparatus

Twelve sighted participants (9 male subjects and 3 female subjects) took part in the study. Their median age was 22 years ranging from 17 to 34 years. EyeRing and a smartphone (iPhone 4S) were used to conduct the study. The studies were conducted in accordance with the ethical research guidelines provided by the Internal Review Board (IRB) of Singapore University of Technology and Design.

Procedure

The experiment was a 2×5 within-subjects factorial design. The two independent variables were: device type (EyeRing or smartphone) and number of currency notes (set of 1, 2, 3, 4 and 5). Participants were given a device (EyeRing or smartphone) and asked to identify currency notes ($ 1, 5, 10, 20, 100). The currency notes were given in 5 sets: {$1}, {$1, $5}, {$1, $5, $10}, {$1, $5, $10, $20} and {$1, $5, $10, $20, $100}, which were presented randomly. Half of the participants used EyeRing followed by the smartphone, the other half used the smartphone followed by EyeRing. With EyeRing, participants pointed at a currency note and pressed the trigger button; subsequently they heard audio feedback (value of the note if identified, otherwise an error message

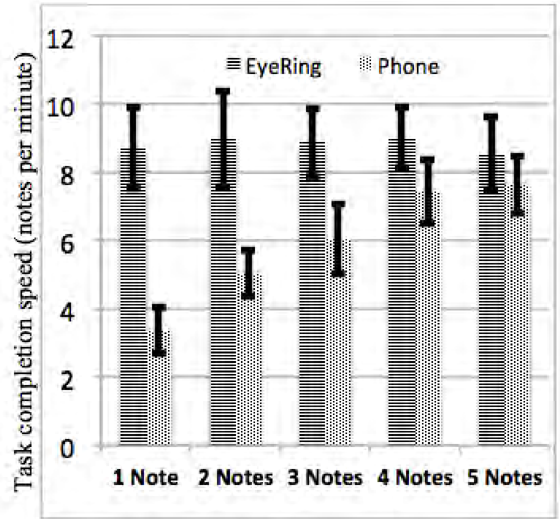


Figure 8. Task completion speed (notes per minute) across all experimental conditions (error bars show 95% confidence interval).

asking for a re-take). For the smartphone application, participants opened the application and scanned using the built-in camera on the phone until they heard the value of the note. The smartphone application had a continuous scanning feature meaning that once started, the application constantly looked for bills and reported their value [29]. For each set of notes, we measured the task completion time (i.e. from the time they were given the set of notes to the time they had identified all the notes). All the participants did a practice session to familiarize themselves with the devices and the experiment procedure. After the study, participants were asked to rate their experience by answering a questionnaire. Each participant took approximately 15 minutes to complete the study.

Results and analysis

Based on the time taken to complete the note recognition task, we calculated the speed (in notes per second) for all the different experimental conditions. As seen from Figure 8, it appears that participants were generally able to compete the task faster with EyeRing. A two-way repeated measures ANOVA analysis showed that there is a main effect of the 'device type' on speed of the task completion $(F(1, 110) = 76.08, p < 0.001)$. Also, there is a main effect of 'number of notes to detect' on speed of task completion $(F(4, 110) = 5.68, p < 0.001)$. Moreover, there is an interaction between 'device type' and the 'number of notes to detect' $(F(1, 110) = 5.64, p < 0.001)$. This combined with the results of Figure 8 implies that EyeRing is faster than the smartphone application as long as there are three or fewer notes to detect. In other words, there is an overhead for using the smartphone application (having to browse to the application, open the application, etc). This overhead contributes significantly to the total task time when there are only a few notes to detect. When there are four or more notes, the overhead cost is compensated due to the continuous scanning function of the smartphone application. In reality this

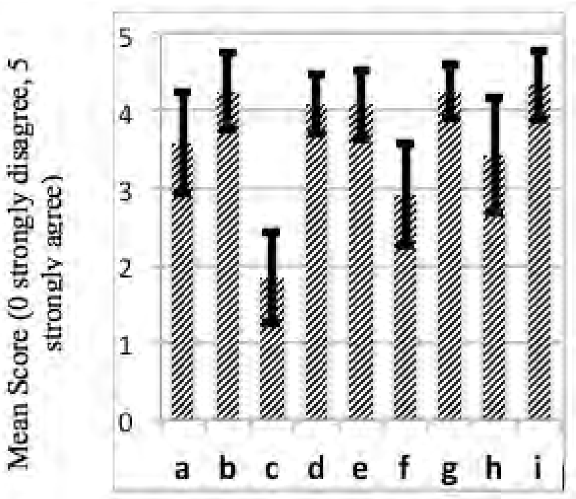


Figure 9. Summary of responses to the survey question (error bars show 95% confidence interval). (a) I like the smartphone form factor; (b) I like the EyeRing form factor; (c) Phone app required less effort compared to EyeRing; (d) EyeRing required less effort compared to Phone application; (e) EyeRing allowed hands-free operation; (f) I prefer to use smartphone application on a regular basis; (g) I prefer to use EyeRing on a regular basis; (h) Camera preview of the smartphone was helpful to frame the picture; (i) Pointing gesture of the EyeRing was helpful to frame the picture.

overhead is likely to be even greater than what we measured (since the user would have to find the phone, take it out, unlock the screen, etc). In contrast, for EyeRing, the speed of note detection doesn't depend on the number of notes. This is because there is no overhead, and for each note, the participants need to point-and-shoot to identify the value of the note.

Response to the questionnaire

Participants rated their experience using eight questions on a scale of 0 (strongly disagree) to 5 (strongly agree). Figure 9 shows the questions and the summary of responses. Although many people liked the smartphone form factor (question **a**), most of them indicated that the phone application requires more effort (question **c**). The difference in scores for questions **c** and **d** is statistically significant. This suggests that EyeRing required less effort compared to a smartphone application. Most participants agreed that EyeRing offers hands-free operation (question **e**) and that the pointing gesture is helpful to frame the picture (questions **i**). This is in line with our observation that participants touched the note with their finger and then backed off a bit to take a picture. The fact that EyeRing is more of a pointing device helped them to do the task more easily; however, a couple of participants mentioned that they prefer to get feedback about what they are pointing at. We are currently exploring options for doing so, for example, by using a laser pointer to indicate the location of the camera focus. In contrast, two participants mentioned that continuous scanning with smartphone made the task easier. In summary, we believe that decoupling the camera from a smartphone made EyeRing into an input device that is immediately accessible, requires less effort and is less disruptive.

CONCLUSION & FUTURE WORK

EyeRing, which leverage the pointing gesture, shows the potential of building seamless interactions for people with vision impairments and sighted alike. The nature and design of the ring apparatus is driven by lessons from established design frameworks for both natural interaction and assistive technologies. Preliminary user reactions suggested that the use of EyeRing applications is intuitive and seamless. A controlled user study indicated that EyeRing is an immediately accessible device that performs faster than a smartphone application for a single object detection task. Future applications using EyeRing (or even multiple EyeRings) rely on more advanced capabilities of the device, such as real-time video feed from the camera, increased computation to perform low-level computer-vision tasks such as edge detection or optical flow, and additional sensors like gyroscopes. These hardware capabilities and applications are currently in development for the next prototype of EyeRing.

ACKNOWLEDGMENTS

This work was supported by the International Design Center (IDC) of Singapore University of Technology and Design (SUTD) with IDC grants IDG31100104A and IDD41100102A. The authors would like to thank Fluid Interfaces group for their insightful remarks, Amit Zoran for the design of the prototype casing, and Shanaka Ransiri for conducting user studies.

REFERENCES

1. Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. Recent advances in augmented reality. *Computer Graphics and Applications, IEEE 21*, 6 (nov/dec 2001), 34 –47.

2. Buxton, W., Billinghurst, M., Guiard, Y., Sellen, A., and Zhai, S. Human input to computer systems: theories, techniques and technology. `http://www.billbuxton.com/inputManuscript.html`, 2011.

3. Carpendale, J. I. M., and Carpendale, A. B. The development of pointing: From personal directedness to interpersonal direction. *Human development 53*, 3 (2010), 110–126.

4. Chatterjee, R., and Matsuno, F. Design of a touch sensor based single finger operated wearable user-interface terminal. In *Proceedings of the SICE-ICASEÕ06* (2006), 4142–4147.

5. Csurka, G., Dance, C. R., Fan, L., Willamowski, J., and Bray, C. Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV'04* (2004), 1–22.

6. Design, Y. Thimble: There's a thing for that. `vimeo.com/groups/yankodesign/videos/17873025`.

7. Fukumoto, M. A finger-ring shapedwearable handset based on bone conduction. In *Proceedings of the ISWC '05*, IEEE Computer Society (2005), 10–13.

8. Fukumoto, M., and Suenaga, Y. Fingering: a full-time wearable interface. In *Proceedings of CHI '94*, ACM (1994), 81–82.
9. Galeotti, J., Horvath, S., Klatzky, R., Nichol, B., Siegel, M., and Stetten, G. Fingersight: fingertip control and haptic sensing of the visual environment. In *SIGGRAPH'08 new tech demos*, ACM (New York, NY, USA, 2008), 16–16.
10. Hyong-Suk, C., Ji-hye, J., and Park, Y.-J. Smart finger. `www.yankodesign.com/2010/09/13/my-fingers-are-so-smart-they-measure`.
11. Kim, D., Hilliges, O., Izadi, S., Butler, A., Chen, J., Oikonomidis, I., and Olivier, P. Digits: freehand 3d interactions anywhere using a wrist-worn gloveless sensor. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, ACM (2012), 167–176.
12. Lee, J., Lim, S.-H., Yoo, J.-W., Park, K.-W., Choi, H.-J., and Park, K. H. A ubiquitous fashionable computer with an i-throw device on a location-based service environment. In *Proceedings of the AINAW '07*, IEEE Computer Society (2007), 59–65.
13. Matthews, D., Behne, T., Lieven, E., and Tomasello, M. Origins of the human pointing gesture: a training study. *Developmental Science 15*, 6 (2012), 817–829.
14. McNeill, D. Catchments and context: Non-modular factors in speech and gesture. In *Language and Gesture*. Cambridge University Press, New York, NY, USA, 2000, ch. 15, 312–328.
15. Merrill, D., and Maes, P. Augmenting looking, pointing and reaching gestures to enhance the searching and browsing of physical objects. In *Proceedings of the PERVASIVE'07*, Springer-Verlag (2007), 1–18.
16. NEC. Imaging and recognition solutions. `http://www.nec.co.jp/techrep/en/journal/g11/n03/110302.html`.
17. Ogata, M., Sugiura, Y., Osawa, H., and Imai, M. iring: intelligent ring using infrared reflection. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, UIST '12, ACM (New York, NY, USA, 2012), 131–136.
18. Perng, J. K., Fisher, B., Hollar, S., and Pister, K. S. J. Acceleration sensing glove. In *Proceedings of the ISWC '99*, IEEE Computer Society (Washington, DC, USA, 1999), 178–180.
19. Pullin, G. *Design meets disability*. MIT Press, 2009.
20. Reese, E., and Cox, A. Quality of adult book reading affects children's emergent literacy. *Developmental Psychology 35*, 1 (1999), 20–28.
21. Rekimoto, J. Gesturewrist and gesturepad: Unobtrusive wearable interaction devices. In *Proceedings of the ISWC '01*, IEEE Computer Society (2001), 21–27.
22. Rekimoto, J., and Ayatsuka, Y. Cybercode: designing augmented reality environments with visual tags. In *Proceedings of DARE'00*, ACM (New York, NY, USA, 2000), 1–10.
23. Rekimoto, J., and Nagao, K. The world through the computer: computer augmented interaction with real world environments. In *Proceedings of the UIST'95*, ACM (New York, NY, USA, 1995), 29–36.
24. Ryokai, K., Marti, S., and Ishii, H. Designing the world as your palette. In *Proceesings of CHI '05*, ACM (New York, NY, USA, 2005), 1037–1049.
25. Shinohara, K., and Tenenberg, J. A blind person's interactions with technology. *Communications of the ACM 52*, 8 (2009), 58–66.
26. Shinohara, K., and Wobbrock, J. In the shadow of misperception: assistive technology use and social interactions. In *Proceedings of the 2011 annual conference on Human factors in computing systems*, ACM (2011), 705–714.
27. Siio, I., Masui, T., and Fukuchi, K. Real-world interaction using the fieldmouse. In *Proceedings of the UIST '99*, ACM (New York, NY, USA, 1999), 113–119.
28. Siri. `www.apple.com/ios/siri`.
29. Sudol, J., Dialemah, O., Blanchard, C., and Dorcey, T. Looktel: A comprehensive platform for computer-aided visual assistance. In *Proceedings of workshop on Computer Vision Applications for the Visually Impaired* (2010).
30. Tsukada, K., and Yasumura. Ubi-finger: Gesture input device for mobile use. In *Proceedings of the APCHI'02*, vol. 1 (2002), 388–400.
31. Winberg, F., and Bowers, J. Assembling the senses: towards the design of cooperative interfaces for visually impaired users. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, ACM (2004), 332–341.
32. Yang, X., Grossman, T., Wigdor, D., and Fitzmaurice, G. Magic finger: always-available input through finger instrumentation. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, ACM (2012), 147–156.
33. Youssef, S., and Salem, R. Automated barcode recognition for smart identification and inspection automation. *Expert Systems with Applications 33*, 4 (2007), 968–977.
34. Zhai, S., Milgram, P., and Buxton, W. The influence of muscle groups on performance of multiple degree-of-freedom input. In *Proceedings of the CHI '96*, ACM (1996), 308–315.
35. Zhang, C., Wang, J., Han, S., Yi, M., and Zhang, Z. Automatic real-time barcode localization in complex scenes. In *Image Processing, 2006 IEEE International Conference on*, IEEE (2006), 497–500.
36. Zimmerman, T. G., Lanier, J., Blanchard, C., Bryson, S., and Harvill, Y. A hand gesture interface device. In *Proceedings of the CHI'87*, ACM (1987), 189–192.