

My Own Private Kiosk: Privacy-Preserving Public Displays

Marc Eaddy[†]
eaddy@cs.columbia.edu

Gábor Blaskó[†]
gblasko@cs.columbia.edu

Jason Babcock[‡]
babcock@cis.rit.edu

Steven Feiner[†]
feiner@cs.columbia.edu

[†]Computer Graphics and User Interfaces Lab
Columbia University

[‡]Visual Perception Laboratory
Rochester Institute of Technology

Abstract

Ubiquitous, high-resolution, large public displays offer an attractive complement to wearable displays. Unfortunately, the inherently public nature of these public displays makes them unsuitable for displaying sensitive information. We present EyeGuide, a wearable system that allows the user to obtain information quickly from a public display without sacrificing privacy. To this end, EyeGuide employs a lightweight head-worn eye-tracker for hands-free object selection and an earphone for private communication.

Our system supports public displays that are dynamic (e.g., a large plasma screen) and static (e.g., a large printed map). In our printed map scenario, EyeGuide whispers verbal directions via earphone to a user, based on where they are looking on the map. Using a technique we call “gaze steering,” the system guides the user’s eye position to specific locations. In our dynamic public display scenarios, EyeGuide presents documents (e.g., maps) that contain sensitive data in a way that preserves privacy.

KEYWORDS: public display; privacy; eye tracking; gaze-contingent; gaze steering; audio augmented reality

1. Introduction

Public displays are commonly found in information kiosks, airports, campus and office buildings, and even in subways. They can be ideal for showing large, complex documents or high-resolution color graphics. Yet their public nature makes these displays inappropriate for displaying content that contains private data. To address this problem, we developed a proof-of-concept system called EyeGuide that uses the superior display capabilities of static and dynamic public displays, while maintaining the privacy and mobility of a wearable system.

Dynamic public displays can interact with the user and provide feedback. Examples are large plasma displays, CRT and LCD monitors, and video-projected displays. In cases where the wearable system can communicate with the dynamic display, it is possible to augment the display with visual markers, annotation, and other



Figure 1. EyeGuide tells user where to look on the map to get from Columbia University to JFK airport.

information. *Static* public displays do not support interaction with the user or provide feedback. They often take the form of large printed maps, blueprints, and diagrams, but can also include 3D architectural models, artwork, and static versions of the dynamic public displays mentioned above.

While printed materials allow for a much higher resolution than current digital displays, their static nature requires that any type of augmentation be done indirectly. For example, a see-through head-worn display could provide an indirect information overlay while still maintaining privacy. Unfortunately, see-through head-worn displays currently suffer from issues such as low brightness, low contrast, and, in the case of optical see-through displays, insufficient transparency [1]. Since these problems would diminish the benefits of using high-resolution printed material, we instead chose to use audio augmentation [2].

Figure 1 shows a user wearing our system and looking at a printed subway map. In this scenario, the user

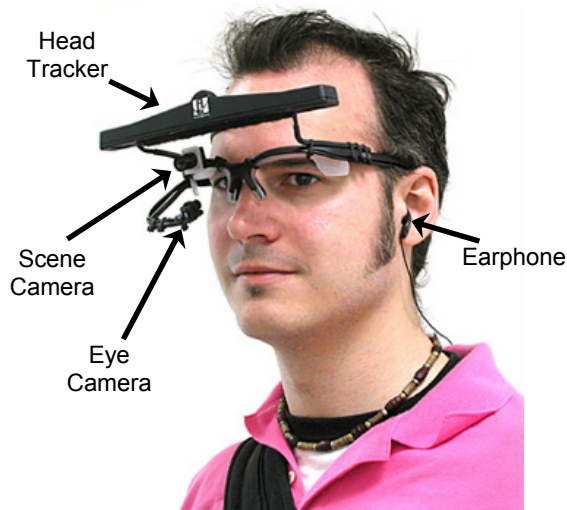


Figure 3. Custom eye- and head-tracking headgear

does not want curious onlookers to know what their destination is going to be or what route they are planning to take. EyeGuide audibly augments the map by providing gaze-steering instructions (e.g., “Look left”) using synthesized speech. To ensure privacy, the user receives the instructions from an earphone (worn in the left ear in Figure 1).

Although it is possible to display large or complex documents on a PDA or wearable display, it is not always convenient. The limited resolution of a typical PDA requires either continuous scrolling of the document or scaling it down to an unacceptable level of detail. Both techniques make the task of locating and following a route difficult. Our system allows the user to view the entire wall-size map so that they can trace their route or find other information at the full level of detail.

While a laptop or tablet PC provides adequate resolution, it may be difficult to access (if stowed in a bag, for example) and takes a long time to start up. In contrast, our system provides a quick, natural, and hands-free interaction with the map, once it has been calibrated.

2. Related work

Starker and Bolt designed a similar application [3] that combines eye movement and synthesized speech to augment the scene in real-time. Their system uses a remote eye-tracker that is not head-worn. In contrast, our system is a wearable eye- and head-tracking hybrid system. Starker and Bolt’s system verbally responds to fixations on specific locations, whereas our system verbally steers the user’s gaze to look at specific locations.

Our system is a *gaze-contingent display* similar to the work of [4] and [5], insofar as they use an eye-tracker as an input device that affects what is displayed. Our system is unique in that it employs oral communication for privacy and adapts to changes in accuracy.

Remote eye-tracking solutions can provide improved calibration procedures [6], require no wearable components, and allow for more head movement than our system. In comparison, our system’s chief advantage is that it is mobile. A significant limitation of our system, however, is that it currently requires the user to keep his or her head still while using the system. We are working on removing this limitation.

Several mobile eye-tracking systems have been developed, including one built by Duchowski and colleagues that is integrated with a virtual reality head-worn display [7] [8]. However, these systems perform offline analysis of the eye movement data to reconstruct gaze direction and point of regard. In contrast, our system analyzes the eye movement data in real-time.

3. System description

Figure 2 shows a close-up of the wearable system. It uses a lightweight eye-tracking headgear developed at the Rochester Institute of Technology [8]. The headgear is modified to hold an InterSense IS-900 VWT ultrasonic wireless head-tracker. The video image from the eye camera is connected to an ISCAN RK726BMP Portable Eye-Tracking processor. The ISCAN unit performs analysis of the eye image to locate the centers of the pupil and corneal reflection. For this system, we used only the pupil coordinates reported by the ISCAN processor.

The backpack in Figure 3 holds the ISCAN processor, the InterSense radio transmitter, and a Dell Inspiron 8000 laptop. The ISCAN unit and eye-tracking headgear are powered using two Sony NP-F960 rechargeable batteries regulated by a TRI-M Engineering power supply. The InterSense unit is powered by its own rechargeable power supply. The laptop has a 1.1 GHz CPU and 256 Megabytes of RAM. A small earphone (Figure 2) was attached to the laptop’s audio-out jack to allow the user to hear instructions privately. The Microsoft Speech SDK v5.1 was used for speech synthesis. Gandalf [9] was used for 3D matrix and homography calculations.



Figure 2. Wearable system housed in backpack.

4. Proof-of-concept applications

4.1. Static: Map navigation using gaze steering

We recreated in our lab a scenario in which the user is looking at a wall-sized subway map in order to find a route from Columbia University to John F. Kennedy Airport, as shown in Figure 1. To allow us to concentrate on the novel aspects of the project, we hardcoded the starting and ending destinations and the subway route. The route was encoded using a piecewise linear path made up of points we measured on the map.

Once calibrated, EyeGuide continually detects where on the map the user is looking and uses speech synthesis to tell the user where to look next. An example sequence is “Look for the red subway line to the far left,” “Look right for Columbia University,” “You are looking at Columbia University. Board the downtown 1 or 9 train,” “Follow red subway line down,” “Change here for the A train to Brooklyn,” and “Exit here for JFK airport.”

EyeGuide uses an empirically derived distance threshold to infer that the user is “lost” (i.e., not looking in the right place). It then provides instructions based on how far away the user’s eye position is. Examples are “Go back to the red subway line” (far away), “Look farther left” (close), and “Look left” (very close).

Our scenario assumes that the user is not familiar enough with the map for specific directions like “Turn right on Chambers Street” to be effective. We also do not use directions with absolute measurements such as “Look 51 cm from the right and 19 cm from the top.”

EyeGuide also makes it possible to find locations that are not visible on the map (e.g., 2960 Broadway or Tsampa Restaurant).

4.2. Dynamic: Information overlay

For this scenario, EyeGuide communicates with a dynamic public display to augment the display directly. To simulate a large dynamic public display, we use a computer that runs a simple server application and we project its display on a wall using a video projector.

This scenario involves a student who would like to arrange a meeting with her friends. The public display shows a large, detailed map with markers indicating friends’ locations (Figure 4). To maintain privacy, the friends’ names are not shown. When the user looks at a particular marker, a speech generator whispers the associated friend’s name in her ear through an earphone.

This scenario was first described by Raghunath and colleagues [10], who had the user look at a PDA to obtain the friends’ names. For example, the PDA could show a numbered list of friends, and the markers on the public display could be numbered accordingly. In contrast, our wearable system allows the user to look directly at the markers to identify them, thereby maintaining continuous eye contact with the display. By eliminating a visual context switch, we believe our system results in a

more seamless and natural interaction

5. Calibration

The user initiates calibration while standing in front of the public display. During calibration and, in the current version of our system, while the application is running, the user must keep his or her head still. This is vital to ensure a reasonable amount of eye-tracking accuracy. We are working on removing this limitation.

To perform calibration for the printed map, EyeGuide tells the user, via the earphone, to look at specific features on the map (e.g., “Upper left corner” or “The letter ‘A’ in Manhattan”). We intentionally chose features that were easily identifiable and did not require previous familiarity with the map. In comparison, for dynamic public displays the calibration points are shown directly on the display.

We experimented with a least-squares linear regression mapping function that treated x, y independently and a homography mapping function that treated x, y dependently. Based on the residual error, the homography mapping function was consistently more accurate [11].

6. Adapting to eye-tracking accuracy

Eye-tracking accuracy is affected by limitations of and errors with the ISCAN eye-tracking and InterSense head-tracking hardware. ISCAN states its resolution to be 1° of visual angle with a usable visual field of about 30° at a sampling frequency of 60Hz. To determine the map area subtended by 1° of visual angle, we must determine the distance of the user’s eye to the map. For this, we employ the InterSense head position and orientation tracker.

We approximated the gaze direction using the head position and orientation [7], as we need only the distance to determine the eye-tracking resolution. We can determine the angle subtended by two 3D points, **A** and **B**,

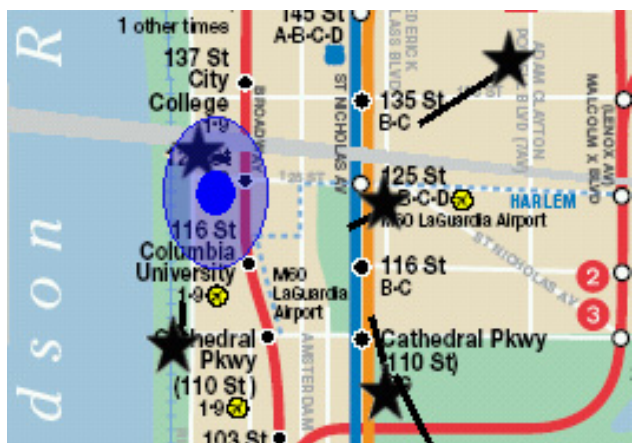


Figure 4. Dynamic display detail showing the current eye position and each friend denoted with a star and a leader line to their location.

using the visual angle equation:

$$\theta = \cos^{-1} \frac{(\mathbf{A} - \mathbf{E}) \cdot (\mathbf{B} - \mathbf{E})}{\|(\mathbf{A} - \mathbf{E})\| \|(\mathbf{B} - \mathbf{E})\|},$$

where \mathbf{E} is the eye origin. We also measure the residual error associated with the function that maps eye positions to real-world map coordinates.

The eye position circle in Figure 4 depicts the results of these measurements. (This is shown during testing only since users would probably not want to reveal their eye movements to onlookers.) The inner circle is the actual size of 1° of visual angle. It increases as the subject moves farther away from the display. The outer circle is the actual size of the mapping function's residual error in the x and y components.

Poor accuracy results in a large selection region, increasing the possibility of target ambiguity. This makes it difficult to select individual objects that are too close together. We attempt to alleviate this problem by dynamically "spreading out" markers that are within the range equal to the estimated accuracy. This can be seen in Figure 4 by noticing the markers have been spread out so that it is not possible for the selection region to include more than one.

7. Conclusions and future work

EyeGuide uses eye tracking in a novel way to guide the user by voice to look at specific points in the scene ("gaze steering"). Using voice as a feedback mechanism preserves personal and private information and allows the user to maintain constant visual contact with the scene. We believe this approach is faster than visually searching a map for a specific location or using a street index. Using EyeGuide, we were able to locate scene points within seconds. In addition, by incorporating measured and estimated accuracy, the system dynamically adjusts to make the selection process as accurate as possible. In practice, we were able to achieve an accuracy of about 1–2 degrees of visual angle.

The requirement that users keep their head perfectly still while using the system is a serious impediment to mobility. We plan to eliminate this restriction by incorporating the head position and orientation information from the head-tracker.

We are exploring repurposing the scene camera to perform vision-based pose estimation. This would eliminate the expensive head-tracking infrastructure, reduce headgear size, improve registration, and allow mobility outside of a lab setting. Unfortunately, our initial experiments with ARToolkit [12] were discouraging. We are looking at combining ARToolkit with Intel's OpenCV vision library [13] or using one of our lab's own vision-based tracking systems [14]. Ideally, we could also eliminate the need for the expensive eye-tracking processor. This could be accomplished using well-

documented [15] image analysis algorithms on the eye camera video feed to track the eye position. By performing video-based pose estimation and eye tracking, a complete mobile, wearable eye-tracking solution could be built for under \$500 USD.

Acknowledgements

This research was funded in part by Office of Naval Research Contract N00014-04-1-0005, and gifts from IBM and Microsoft.

References

- [1] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre. Recent advances in augmented reality. *IEEE Comp. Graphics and Appl.*, 21, 6, 2001. 34–47.
- [2] J. Rozier, K. Karahalios and J. Donath. Hear&There: An augmented reality system of linked audio. In *Proc. of the Conference on Audio Displays (ICAD 2000)*, 2000, 63–67.
- [3] I. Starker and R. Bolt. A Gaze-Responsive Self-Disclosing Display. In *Proc. SIGCHI Conf. on Human Factors in Comp. Sys.*, (Seattle, WA, 1990), pp. 3–10.
- [4] A. Khiat, Y. Matsumoto and T. Ogasawara. Task Specific Eye Movements Understanding for a Gaze-Sensitive Dictionary. In *Proc. of the Conference on Intelligent User Interfaces*, (Funchal, Madeira, Portugal), 2004, 265–267.
- [5] R. Danforth, A. Duchowski, R. Geist, and E. McAliley. A Platform for Gaze-Contingent Virtual Environments. In *Smart Graphics (2000 AAAI Spring Symposium, Technical Report SS-00-04)*, (Menlo Park, CA, 2000), AAAI, 66–70.
- [6] T. Ohno and N. Mukawa. A free-head, simple calibration, gaze tracking system that enables gaze-based interaction. In *Proc. of Eye Tracking Rsch. and Appl. (ETRA '04)*, 2004, 115–122.
- [7] A. Duchowski, E. Medlin, A. Gramopadhye, B. Melloy, and S. Nair. Binocular Eye Tracking in VR for Visual Inspection Training. In *Proc. of the Conference on Virtual Reality Software and Technology (VRST '01)*, 2001, 1–8.
- [8] J. Babcock and J. Pelz, Building a Lightweight Eyetracking Headgear. In *Proc. of Eye Tracking Rsch. and Appl. (ETRA '04)*, 2004, 190–114.
- [9] <http://gandalf-library.sourceforge.net>
- [10] M. Raghunath, C. Narayanaswami, and C. Pinhanez. Fostering a Symbiotic Handheld Environment. *IEEE Computer*, Sept. 2003, 56–65.
- [11] K.P. White, Jr., T.E. Hutchinson, and J.M. Carley. Spatially Dynamic Calibration of an Eye Tracking System. *IEEE Trans. on Systems, Man, and Cybernetics*, July/Aug. 1993, 23, 4, 1162–1168.
- [12] P. Malbezin, W. Piekarski and B.H. Thomas. Measuring ARToolkit Accuracy in Long Distance Tracking Experiments. In *Intl. Aug. Reality Toolkit Wrkshp.*, 2002.
- [13] <http://www.intel.com/research/mrl/research/opencv>
- [14] T. Okuma, T. Kurata, and K. Sakaue. A Natural Feature-Based 3D Object Tracking Method for Wearable Augmented Reality. In *Proc. of Advanced Motion Control (AMC'04)*, 2004, 451–456.
- [15] J.B. Mulligan. A Software-based Eye Tracking System for the Study of Air-traffic Displays. In *Proc. of Eye Tracking Rsch. and Appl. (ETRA '02)*, 2002, 69–76.