Project Glass 2.0 – Interaction with Physical Devices through Attention

Ben Zhang, Sean Chen
University of California, Berkeley
benzh@eecs.berkeley.edu, sean.yhc@berkeley.edu

ABSTRACT
With the increasing interest of assigning intelligence to everyday object, many approaches of interacting with them fall back to smartphone based applications. We argue about the inconvenience, overhead, and distraction in such an interaction paradigm, and propose a more natural solution. A glass-based controller which enables user to express their interest in interaction directly simplifies the process of selection in a smart-home or smart-office senario. We present our thoughts, design and evaluation of such a system in this paper to motivate researchers in considering "interaction in attention". In the meantime, we noticed that Google Glass might serve as the perfect enabling technology, though in our prototype we've adopted infrared approach for proof-of-concept. The evaluation and discussion of such system might be valuable for future work to explore on more mature hardware platform.

Author Keywords
HCI, Interaction, Attention, Infrared, Glass

INTRODUCTION
The vision of smart-home and smart office is being realized recently with the development of radio technology, sensors, and actuators. Nowadays, you can buy the basic components online [7, 5] and many hobbyists have already created their own smart environment (see [12] as an example). Though various challenges are ahead in the field to build a more robust, scalable, energy-efficient, and user-friendly systems, the general trend of having smart appliances around us is foreseeable. Those smart appliances each can be controlled separately (like what is existing now for television and heater), however, with the prevalence of smartphones, many systems [6, 3] are being designed as applications on mobile platforms so that the smartphone act as an universal bridge between human and appliances.

Admittedly, people have been used to accessing the Internet, conversing with friends, and even dealing with online business through their smartphones. And putting the control function of physical device into smartphones can significantly lowers the barrier for ordinary users. Also users can get other benefits by using smartphones – networking, display, gesture input, and the increasing computability.

However, we argue that traditional graphical user interface (GUI) on mobile platforms still imposes difficulty in interacting with physical objects. Imagine when we have all the control panel integrated into iPhone, users then need to browse a list of possible icons and select the right one. Even if you can create widget to simplify the browsing overhead, such as on Android platform, the problem still holds since squeezing physically distributed objects onto a-few-inch screen is not the right approach. In an ideal system, the user should be able to express their interest of interaction in a more direct way. On the other hand, taking out the phones (or even simply locating them when at home) is relatively time-consuming. According to a study in [8], 4 seconds are required on average just to get a phone out of the pocket or hip holster, not to mention the time to choose the target and initiate interaction. This project is trying to propose a new way of interacting with physical objects without losing attention, essentially solving the "addressing" problem in interacting with physical devices around.

Attention Aware System (AAS) [10] has been studied for a while in understanding how users' attention affects their interaction, and how system should affect or manage users' attention. This comes from some observations in human-human interaction. For people to direct interact with physical objects, their attention can be a strong indicator of their interest. When staring at the television, the user is more likely to change channels or do other TV related controls rather than turn off the heater. If they do want to control the heater, they will need to express this explicitly. Discussion of understanding human's thoughts through brain-computer interface is beyond the scope of this work. We utilize users' "gesture" to know their interest for interaction.

Some direct gesture include looking, pointing and reaching, and there is one interesting paper [14] which put sensors to users' ears and fingers to detect their intention. And this information is fed back to the system to assist user in browsing physical objects. Our approach borrows idea from this by putting sensors on users' body, however, we enable the direct interaction rather than just passive sensing. The other major difference is that we choose to use glass form factor for such attention expression. Firstly, line of sight is easier to detect on glasses, rather than earpiece. And this is also more intuitive to users that when they are looking at something, they might be interested in that, and this information is captured by a on-glass sensor. Second, with the release of Google Glass, we believe that it could serve as the enabling hardware platform. Though in this paper, we will still present some of our thoughts on "what if we have an advanced glass and various sensors", we focus most of the discussion on our prototype which uses IR to express users' interst, XBee radio for device communication, and a slider for users' gesture. We admit that improvements are huge to make the device more compact and compelling, the system more robust, and the size smaller and more convenient. But for this semester project, our goal (for us) is to explore the possibility of enabling direct interaction with physical objects, learn how to prototype novel interaction device on our own, and apply what we have learned and discussed in class about human computer interaction into a real project.

The contribution in this work is to propose a new way of interacting with physical device and simplify every task. Thoughts and discussion on this topic might also be beneficial to researchers on similar work.

The rest of this paper is structured as follows. We will first give an overview of some related projects, and discuss the similarities and different of our work from theirs. Then we present our approach in detail about the system design and implementation in the following two sections respectively. Evaluations and in-depth discussions will be provided before we concludes this paper and talk about potential future work.

RELATED WORK

In this section, we will summarize some prior work and how it relates to our project in three folds: Tangible User Interface (TUI), Glass-based device, and Attention Aware System (AAS), discussing each in turn.

Tangible User Interface

The concept of Tangible User Interface was proposed in [11], and according to the original paper, such system would enable user to "grasp & manipulate bits in the center of users' attention by coupling the bits with everyday physical objects and architectural surfaces". Thereafter, numerous TUI systems (such as [18, 20]) are developed. [14] is one that is closest to our system, which base users' interaction with physical objects on people's natural looking, pointing and reaching metaphor. However, these systems have limitations that they focus too much on sensing the physical world, while the actuation is mostly done in cyber world. Recent years, many startups [6, 5, 3] tries to "hack" the physical world to achieve the envisioned Internet of Things by enabling actuating devices from phone/web portal.

Our system combines some merits from prior work. As has been pointed in Sec. , we are also using people's natural looking as attention tracking as in [14], but we are not limited to browsing or searching physical objects. We provide such capability of interacting directly with physical devices. This also improves some of existing work that is based on smartphones or web portal [6, 5].

Glass Form-factor

[13], as the earliest wearable device, uses the form factor of glass to serve as a logging machine that records user's daily life. According to one review paper [16], there are many glass-based device developed for the past decades. Those who use glasses as input have primarily focused on gaze-tracking [21, 17], and the projects that uses glasses for available output device works hard to achieve virtual reality [4, 2].

The most recent and potentially most impactful project [2] integrate many crucial components – camera, projection, microphone, speaker, IMU and networking – into a single glass that can achieve virtual reality to an unimaginable extent. We do believe this would become a powerful platform for future interaction development. However, so far the vision is still being limited to digital world. In contrast, our project explores the potential of using glasses for direct interaction with physical devices, rather than performing "digital" tasks.

Attention Aware System

Microsoft Research's project "Attentional User Interface" [10] explores how attention can be employed to enhance human-computer interaction, from the lesson learned in human-human interaction. Attention detection can also assist interface design to avoid context shifting overhead, e.g. the peripheral displays and the notification level should be adjusted according to user's attention [19].

These work researched more on how the system should manage or adapt to users' attention. While for us, we want user to express their attention directly and use this to address the target for interaction.

DESIGN

In this section, we will talk about our design considerations in solving the "addressing" problem in the smart environment. We start by describing the example application, where in an ideal case, how the user should interact with physical devices. From that, we draw some design requirements. To explore the design space, we propose several possible solutions. Some of them are difficult to achieve within a shrot time. In next section, we will build a proof-of-concept system, where we have simplified several tasks.

Example Application

Our work is primarily motivated by the observation that in CS294-84 course at Berkeley, the instructor has to go to the switch to turn off the light during presentation. The separation of device and their controller makes it possible for people control things that are not reachable (such as the lights on the ceiling). However, such indirection introduces additional overhead since people still need to locate the controller and then interact. Remote control might mitigate such a problem, but one controller for one device is not a scalable solution. And squeezing controllers onto the smartphone display is also not the right approach. We will demonstrate this by giving the following Smart Home example.

Imagine in a smart-home environment that most devices are remotely controllable. When the users are watching TV shows, they want to turn off the lights and turn up the volume so they can enjoy this memorable time. Such ordinary-life tasks are happening everyday, and any simplification will result in significant time savings. One ideal approach is that the user expresses his interest of turning off the lights by simply staring at the light, plus a specific gesture. Then he turns his head back to the TV, and again simply "asks" the TV to turn up the volume. When transitioning, the disconnection with the light and the connection with TV happens seamlessly, with the correct feedback for the user. And all the magic are resolved by an always-available device; this might be the glasses, rings or watches.

Occasionally, the user needs to perform some advanced configurations to the TV, such as adjusting the white balance, or the brightness. Since the need of these complicated tasks are not frequent, and it will be less efficient to embed these features into the always-available device (which is designed to be small and easy-to-access, rather than too flexible). The smart-home falls back to a more generic approach which relies on touch screen or other rich input device. In this example case, since the user's attention is still on TV, his smartphone may pop up the control menu for advanced usage automatically. Such menu could also just pop up on walls or other large interactive displays [9, 15]. Even in this case, the attention system knows users' intention and saves time for user to address the device to interact.

System Requirements

From what we have described in the previous section, the example application is enabled by a few necessary requirements. We will discuss some important ones in this section.

The first and foremost is the ability to sense and control in an always-available fashion. Since the goal is to simplify everyday task in a smart-home or smart-office scenario, any cubersome solution is not desired. The control of the device would also have to be simple, not to confuse the user of using it.

Secondly, since screens might not always be available when users are interacting directly with physical device. Adequate feedback should assist user understanding what is going on. Even if screens are everywhere to assist in the future, the requirements of having proper feedback still hold.

As discussed before, we envision the simple always-available device would not be capable to cover all tasks. For those which are really complicated and less likely being executed, we will have to enable the system to fall back.

Besides the previous three, for any human-centric system, additional requirements like responsiveness would also be required, though not explicit discussed through this paper.

System Design

In this section, we will present several design consideration that fulfills the requirements. From [14], they tried the solution of putting IR into earphones, rings to enable looking, pointing and reaching metaphor. When we were thinking about how to capture user's attention, we are actually leaning towards the glasses solution. Part of the motivation is from Google Glass. Generally, glasses are more natural in align with human's light of sight. Therefore, sensors on glasses can reflect the users' attention in a more precise way. Then given the glass form factor, what might be the right sensor or

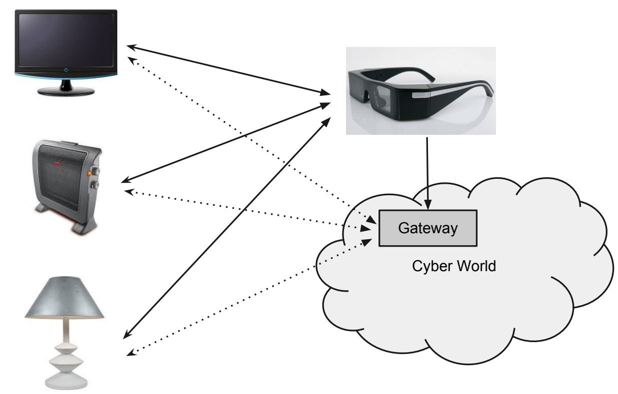

Figure 1. System Architecture

component to use to obtain users' attention remains a major question for us.

The first possible approach is a pure Google Glass solution. Since the glasses come with a camera, sophiscated computer vision algorithms might be able to detect what is in front of the user's sight. Following this thoughts, the camera would also be able to tell users' gestures. In fact, there are more features which would make this interaction cooler – the voice recognition is another important source of user input, the projection onto user's eyeballs can provide timely feedback to users' action, the networking capability provide seamless connection with other peripheral devices such as displays. And initially, we were also thinking about building a prototype that resembles Google Glass, but the complexity behind of achieving acceptable image processing is huge. Not to mention we don't have the Google Glass yet. This approach is more an exploration of design space.

Since our target is to validate the thoughts of interacting directly with physical objects through users' attention. We might break the initial proposal to several achievable tasks and build the proof-of-concept. Our second approach still adopts the glass form factor, but we've turned to other sensors to detect users' line of sight.

Specifically, we choose to use IR. When surrounded by tubes, the IR is pretty directional. And from our test, it can differentiate objects in 20 (do some experiments here) centimeters. User input can be captured by a slider on glass frame, where it's easy to access (though not hand-free). The networking capability is enabled by having an XBee module on the main board. We propose the interaction architecture in Fig. 1.

The dotted lines show some existing (even gradually commercialized) approaches that seek to provide access to physical objects through gateways from the broader Internet. In our project, we focus on the solid lines where the users can interact with them directly. Also as has been pointed out in the requirements, the fall-back scheme for complicated tasks are enabled by the communication between the glasses and the gateway. Though simple enough, we find it essentially useful to have this architectural figure in mind when considering users' interaction.

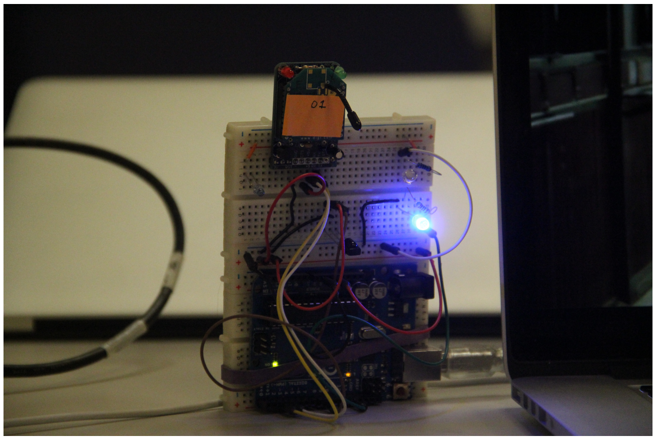

Figure 2. Prototype of the clients

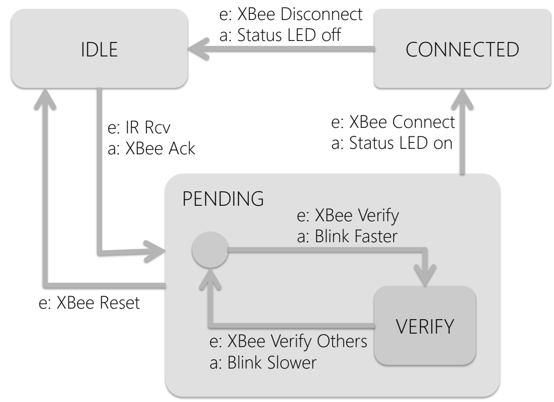

Figure 3. FSM model for client. For each transition, "e" stands for event, and "a" stands for action

IMPLEMENTATION

In this section, we detail our implementation on Arduino [1] platform. The whole system is comprised of a glass module, numbers of clients, and a gateway, as shown in the architecture figure (Fig. 1) in the last section. We will start from the clients, whose primary tasks are to interface between the Glass and the devices that are attached to it. Then we move on to the implementation of the Glass, which manages user's gesture recognition and communication with the clients. The functionality of the gateway so far is confined to only communicate with the clients, since it's not the primary focus of this project.

Clients

The clients are comprised of a main Arduino board, an IR receiver, an XBee radio, and various actuators. We used Arduino Uno, which has an ATmega328 microcontroller and suffice our purpose in this project. We use off-the-shell IR receivers (confirm it is PNA4602), which outputs 0V (low) on detection of 38 KHz carrier or 5V (high) otherwise. The XBee radio, accompanied with an XBee modems, takes charge of the communication with the Glass. LEDs are added as an indicator of the state of clients and visual feedback for users. Additional components are subject to the device being attached to. In our current system, we have a relay (need to know relay model) which can control the AC power plug, and we have a USB connected computer to control video playing. The prototype is shown in Fig. 2.

add clients photos here

The main function of the client is to respond IR signal and communicate with the Glass to receive commands from the user. Normally, the client stays in *IDLE* mode. When it receives an IR signal, suggesting that the user could be interested in interacting with it, the client responds with an XBee acknowledgement and goes into *PENDING* state. The glass, after seding the IR, will wait for a certain amoount of time. If there is only one client responding during the wait, the Glass will directly send a connecting message to the only client. When there are multiple clients responding, the Glass will broadcast a verify message with a client ID. The client with a matching ID is called the active client and its LED will blink faster then the others as a visual feedback to the user. After the user decide which one to connect with, the Glass will send out a connecting message. Client turned into *CONNECTED* state will be ready to take commands. To summarize the clients' behavior, we have a state machine in Fig. 3 for illustration.

Within *CONNECTED* state, the clients will react differently according to the device it is connected to. In our prototype, we have two different types of devices. The first is a relay, which can control the whole AC power supply. The interface to relay is simply an digital pin. We only need to set it to *LOW* or *HIGH*. Such command is transmitted remotely from the Glass through XBee using encoded packet. The second tyep is a computer that is used to play video. To send command to the computer, we have a USB adaptor and a python script taking charge of the translation. We use the *pyserial* and *appscript* library to trigger the play, pause command of an video, and system-wide volume control.

Glass

The Glass is complicated in two folds. First, we move the complexity of coordinating all clients to the Glass. Second, we have to handle user gesture on Glass. From an engineer perspective, we separate the gesture detection and XBee transmission into two customized library files in Arduino. And only relevant information is sent to the main program. We use the same Arduino Uno and XBee module as the client. An IR emitter is placed within a black tube in order to make the signal a straight beam of light. Then the the tube is attached to the Glass and adjusted in a proper direction. In addition, we have a membrane potentiometer (100 mm long) as the slider to take user's input. After we've put the components onto the Glass, we have the prototype shown in Fig. 4.

add glass photos here

The whole work flow on the Glass is as follows. When the user hasn't shown any interest for interaction, the Glass stays

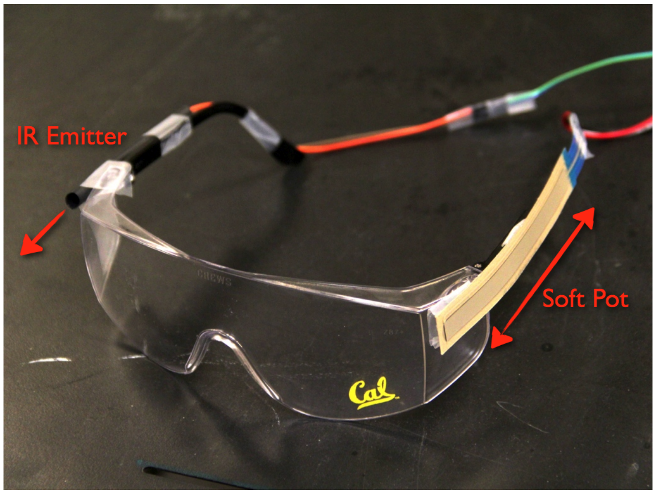

Figure 4. Prototype of the Glass

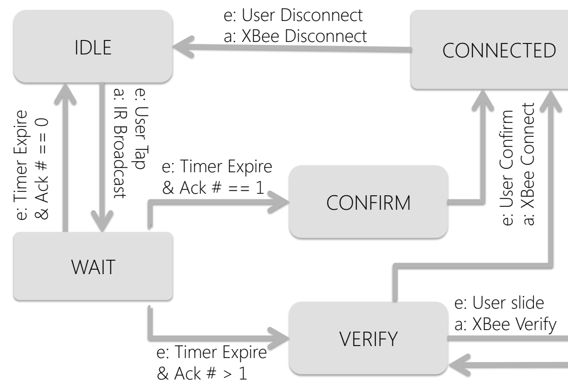

Figure 5. FSM model for Glass. For each transition, "e" stands for event, and "a" stands for action

in *IDLE* state. When "tapping" is detected, the Glass will send a beam of IR signal to inform the devices which are in front of the user. The Glass waited for a specific period of time and count the number of acknowledgement messages responded by the clients. To ensure responsiveness to end user, we set the timer in *WAIT* state to be 1 second. When the timer expires, Glass will go to *IDLE* if no client has responded, or *CONFIRM* if it receives just one message. However, the complicated case is when there are multiple clients that have received the IR signal (because they are near to each other) and responded. Though we expect this scenario rarely happens, there is still chance when the physical devices are put together and it is necessary to differentiate them. In this case, the system goes to *VERIFY* state, and sends out corresponding verifying message based on user's gesture. The user can slide back and forth to select the desired target. Once the user confirmed the selection and release his finger, it goes to *CONNECTED* state and now the Glass is ready to send out commands. We have the FSM in Fig. 5.

We've configured the XBee modules such that the Glass can talk in both direction with all clients, while clients cannot talk with each other. In this way, the Glass would serve as master of this personal area network (PAN), whose ID is 4321 in our scenario.

Gateway

The gateway provides access to this PAN for computers, and presumably this can be further open to the Internet. Since there have already been many products [5, 3] that essentially function in this way, we do not spend much time on this aspect in our project.

For completeness, we wrote a python script that controls a USB XBee adaptor. And this XBee module has the same configuration as the Glass, so it serves as the master of the network. This enables the remote control of the clients.

EVALUATION

Will fill this evaluation when we get to it.

DISCUSSION

Discuss the problems we have encountered and other random stuffs in this section.

Interaction in Attention

During our brainstorming and discussion of this idea, we have identified some tasks where user might not pay their attention to the device they are controlling. One such example is giving presentation. The attention of presenter is to interact with the audience, but this is achieved through controlling the slides and projection.

CONCLUSION

In this project, we explored a novel way of interacting with physical devices by capturing user's attention. The prototype we built demonstrates the possibility of identifying user's line of sight to select and control target appliances. Furthermore, we designed a mechanism to intuitively choose from multiple targets when they are close-by.

There are still a few more components that require enhancement: (A) Thorough verifications of the IR strength, angles, and reflecting effects. (B) A more robust communication protocol with better error handling. (C) A more stable and richer input method. (D) User study.

REFERENCES

1. Arduino. `http://www.arduino.cc/`.
2. Google glass. `http://www.google.com/glass/start/what-it-does/`. Accessed: 2013-05-03.
3. Lockitron. `http://lockitron.com/`. Accessed: 2013-05-03.
4. Lumus. `http://www.lumusvision.com/`. Accessed: 2013-05-03.
5. Ninja blocks. `http://ninjablocks.com/`. Accessed: 2013-05-03.
6. Smart things. `http://smartthings.com/`. Accessed: 2013-05-03.

7. Smarthome website. `http://www.smarthome.com/`. Accessed: 2013-05-03.
8. D. L. Ashbrook, J. R. Clawson, K. Lyons, T. E. Starner, and N. Patel. Quickdraw: the impact of mobility and on-body placement on device access time. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 219–222, New York, NY, USA, 2008. ACM.
9. B. W. R. Center. unpad and ewallpaper. `http://bwrc.eecs.berkeley.edu/research/unpad-and-ewallpaper`. Accessed: 2013-05-04.
10. E. Horvitz, C. Kadie, T. Paek, and D. Hovel. Models of attention in computing and communication: from principles to applications. *Communications of the ACM*, 46(3):52–59, 2003.
11. H. Ishii and B. Ullmer. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*, CHI '97, pages 234–241, New York, NY, USA, 1997. ACM.
12. D. Low. Berkeley ridiculously automated dorm. `http://lab.dereklow.co/brad/`. Accessed: 2013-05-03.
13. S. Mann. Continuous lifelong capture of personal experience with eyetap. In *Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences*, pages 1–21. ACM, 2004.
14. D. Merrill and P. Maes. Augmenting looking, pointing and reaching gestures to enhance the searching and browsing of physical objects. In *Proceedings of the 5th international conference on Pervasive computing*, PERVASIVE'07, pages 1–18, Berlin, Heidelberg, 2007. Springer-Verlag.
15. Microsoft. Microsoft future vision @ 2020. `http://www.youtube.com/watch?v=55p3vNCF4JQ`.
16. D. Morris. Emerging input technologies for always-available mobile interaction. *Foundations and Trends® in Human–Computer Interaction*, 4(4):245–316, 2010.
17. T. Nagamatsu, M. Yamamoto, and H. Sato. Mobigaze: development of a gaze interface for handheld mobile devices. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 3349–3354, New York, NY, USA, 2010. ACM.
18. S. Nanayakkara, R. Shilkrot, and P. Maes. Eyering: an eye on a finger. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, pages 1047–1050, New York, NY, USA, 2012. ACM.
19. S. Park, S. Park, Y. Lim, G. Lee, and M. Hahn. Designing attention-aware peripheral displays with gaze-based notification control.
20. S. N. Patel, J. Rekimoto, and G. D. Abowd. icam: precise at-a-distance interaction in the physical environment. In *Proceedings of the 4th international conference on Pervasive Computing*, PERVASIVE'06, pages 272–287, Berlin, Heidelberg, 2006. Springer-Verlag.
21. T. Selker, A. Lockerd, and J. Martinez. Eye-r, a glasses-mounted eye motion detection interface. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '01, pages 179–180, New York, NY, USA, 2001. ACM.