공학박사학위논문

C.E.E. : Command Everything with Eyes, Multi-modal gaze-based interface for everyday Interaction

착용 형 시선 추적 기반 복합 입력 장치 개발 및 다수

디바이스와의 상호 작용 방법에 관한 연구

FEBRUARY 2015

서울대학교 대학원

기계항공공학부

송 준 봉

C.E.E. : Command Everything with Eyes, Multi-modal gaze-based interface for everyday Interaction

지도교수 이 건 우

이 논문을 공학박사 학위논문으로 제출함

2014년 12월

서울대학교 대학원

기계항공공학부

송 준 봉

송준봉의 박사학위논문을 인준함

2014년 12월

위 원 장	고 상 근	(인)
부 위 원 장	이 건 우	(인)
위 원	김 종 원	(인)
위 원	안 성 훈	(인)
위 원	방 현 우	(인)
위 원	최 성 용	(인)

Abstract

Although gaze interface has high potential to be a pervasive interface, previous researches focused on enhancing performance to control single display device and researches of gaze interaction method with multi devices are lacking. To improve the gaze interface as an interface for everyday interaction, this paper proposes a multi-modal wearable gaze interface optimized for multi devices control. First, we developed high performance wearable gaze interface which can be fully customized for multi device control. A manual input modality for confirmation was added to improve accuracy and avoid Midas touch. Second, an infrared boundary+ID markers and its novel recognition algorithm are suggested and they enabled immediate device switching maintaining high robustness in various light condition. To verify the appropriateness of the developed gaze interface for multi displays control, user tests controlling three display devices were conducted and we compared performance with conventional interfaces, touch and mouse. The performance is similar to touch and better than mouse. In user interviews, the developed gaze interface showed better scores in most factors than other interfaces. Finally, by developing gaze sound ID recognition method, we expand the usability of gaze interface to control non-display device.

Keywords: *eye tracking, gaze tracking, wearable, user interface, multi-modal*

Student Number: 2012-30177

Contents

Chapter 1

Introduction

1.1 Overview

1.1.1 Gaze and gaze tracker

Gaze is known as a fast, intuitive and natural input modality since humans direct their visual attention by means of eye movements [3,10,12]. Many previous studies show the high potential of gaze interface. Especially, gaze control was found to have high potential and even has better performance in pointing than conventional interfaces like the mouse [3,12,13]. Also, in research of Bieg, Hans-Joachim, et al, the gaze interface was proved to be fast, low effort, and very intuitive as it reaches a target before a manual pointer [7]. Accordingly, after an early work by Robert Jacob [5], a number of studies for gaze supported interfaces had been researched and its performance had been improved solving many of problems. Now, we have sufficient levels of hardware specification and processing power for gaze tracking. As a result, the gaze tracking system is used successfully as a measurement interface in the laboratory, marketing, and automotive usability studies, that is, in research fields.

Fig.1 Various type of gaze trackers

1.1.2 Popularization of gaze based interface

In spite of the partial success of gaze tracker in research fields, the gaze tracking system had not been evaluated as a popular interface because of its extremely high price and inconvenience in use. But recently, thanks to low price but powerful processor, cameras and hardware, gaze interfaces have been transferring from researchers to consumers with low price and high performance products [49, 50]. The goal of the current gaze interface for consumers is to control all display devices like monitors, TV sets, mobile phones, and even wall-sized public displays with eyes [44]. But current gaze tracking interface attempts to replace a role of mouse, that is, eye mouse [10]. Eye mouse could be helpful for handicapped person who cannot move their hands because of amyotrophic lateral sclerosis (Lou Gehrig's disease) [61, 51]. In fact, in 'Eye writer project' [48], a graffiti artist who is suffered from Lou Gehrig's disease can continue his artwork by the help of open source gaze tracking algorithm and low cost hardware. However, for general users, mouse or touch is still more comfort and easier way to interact with computer than eye mouse.

Actually, Jacob et.al insists that the barrier to exploiting the gaze interface has not been eye-tracking technology but the study of interaction techniques that incorporate eye movements into the user-computer dialogue in a natural and unobtrusive way. [5] This means, for the popularization of gaze based interface, not only improving the performance of gaze tracker, but also proper interaction method for human gaze needs to be developed accordingly.

Fig.2 Eye writer project, open source gaze tracker for handicapped people [48]

1.1.3 Wearable gaze tracker

Now a days, a pervasive gaze interface is a fixed type, not a mobile type. Since most previous wearable (head mounted) gaze trackers were large, heavy and sometimes looks geek, they were not suitable for long term use and users had resistance to use it.

But, in case of using fixed type gaze tracker, the user has to pay for each gaze interface installed in each device. As the fixed type gaze trackers cannot detect and track multiple users' gazes at a time due to a limited processing capability, it cannot be controlled by multiple users, which is essential for public displays [44].

To avoid these problems which fixed type gaze tracker has, the wearable gaze tracker has to be designed as small, light, and fancy as possible as well as maintaining sufficient computing power. Fortunately, due to the rapid advances in computing power and decrease in cost, tiny and high performance micro computers [52] have been introduced these days and enable highly miniaturized wearable devices with real-time processing capabilities, like google glass [14].

In other words, although current research and commercial gaze interfaces focus on fixed type, rapid advances in computing power and decreasing costs enable well-designed and high performance wearable gaze interfaces.

1.1.4 Eye tracker for everyday interaction

Andreas et.al mentioned that though current research on eye-based interfaces mostly focuses on stationary and fixed settings, advances in mobile eye-tracking equipment and automated eye-movement analysis will allow for investigating eye movements during natural behavior and promise to bring eye-based interaction into people's everyday lives. [1] Therefore, ongoing developments in mobile gaze tracking systems can be a pervasive interface soon and, in the future, the wearable gaze interface will brings eye-based interaction into everyday life [1,4]. If every user wears his or her own gaze interface, ‘Always on’ and ‘Real-time’ gaze data will be collected. This can change the human computer interaction paradigm from current interaction methods such as button, touch, visual code and remote controller to ‘Just Look at’ interaction. In other words, a wearable gaze interface will be developed to be the one and only integrated interface and bring eye-based interaction into everyday life.

1.2 Motivation and objectives

1.2.1 Problems

Until now, most researches focused on improving the accuracy and robustness of the gaze interface itself. However, the obstacle for the popularization of gaze based interface is not the performance of gaze tracker but how to design gaze-based human computer interaction dialogue in a natural and unobtrusive way for everyday gaze-based interaction as Jacob et.al mentioned [5]. In this paper, we set three problems to solve as following.

1. Gaze-based interaction with multi display devices
2. Gaze-based interaction with non-display devices

Previous researches and problems will be reviewed in next sections.

1.2.2 Multi display device control using gaze interface

Most previous research focused on improving the accuracy and robustness of the gaze interface for single device, and that is one of the reasons that a fixed type gaze interface becomes pervasive in commercial gaze tracker industry. However, studies for multi device control using gaze were still lacking. If the wearable type gaze interface becomes popular and the gaze data of every user starts to be collected, the researches of gaze interaction methods for multi devices control need to be developed. Fortunately, thanks to the rapid growth of computing power and processing algorithms, the researching focus of the wearable gaze interface has gradually moved away from single device control to multi devices control.

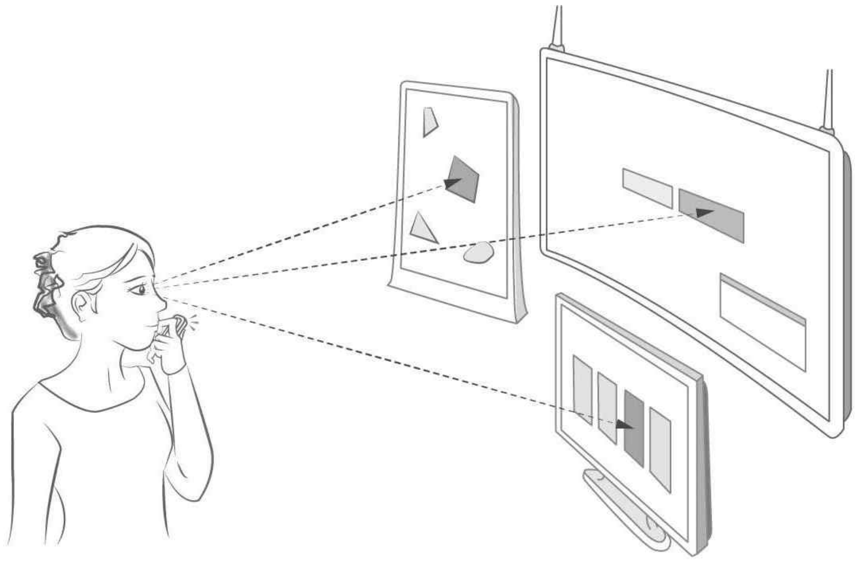


Fig.3 Concept art of wearable gaze interface for multi display device control

Mardanbegi et.al [8] and Hales et.al [11] studied methods of interacting with multi devices using the gaze interface and suggested reasonable systems. Mardanbegi et.al developed a control system using gaze-based head mounted interface for interaction with multiple screens at home. A QR code temporary displayed on screen was used to differentiate displays and acquire extra information. Turner et.al suggested a head mounted gaze interface that could control both hand held devices and public displays [19]. He also enhanced user experience and decreased user fatigue in switching devices by implementing a dual scene camera [24].

Fig.4 Previous researches for multi display device control with gaze [8,19]

These studies proved that a head mounted, wearable gaze interface has sufficient potential for multi device control. However, these works were close to concept works and not applicable for practical applications due to the low robustness for variable lighting condition and user situations. In addition, they took too much time switching to the interested device because of its long dwelling time used for discerning the ID of the display device.

1.2.3 Non-display devices control using gaze interface

As mentioned before, studies for multi display device control using gaze are rare to be found, and it is harder for non-display device control. Mardanbegi et.al [45] suggested gaze based wearable interface to control vacuum cleaner. He put the visual tag, QR code, on the top of the robot vacuum cleaner. By analyzing the image captured by the scene camera, he could recognize current position of the vacuum cleaner and steer the direction of the cleaner gazing the target position. In the same way, by attaching different image tag on the other non-display devices such as light and fan, he could choose device to control and turn it on and off with only his gaze. It could be considered as a meaningful research, but the idea of attaching unnecessary visual code on the device was far from practical application.

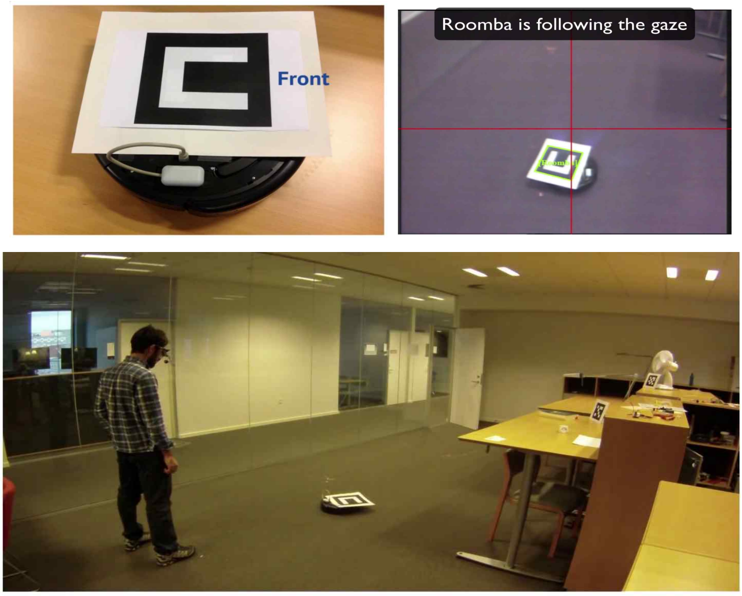


Fig.5 Previous researches for non-display device control with gaze [45]

1.3 Our approach : Eye tracker for everyday interaction

Our goal is to bring eye-based interaction into people's everyday lives, that is, C.E.E : Command Everything with Eyes. For this main purpose, three specific approaches are set as following.

1.3.1 Developing wearable multi-modal gaze interface

Most of all, it needs to be developed high performance wearable gaze interface appropriate for multi device control. Developed gaze interface has to be optimized for multi device control that shows high accuracy, speed, and robustness while enabling immediate device switching. We can use commercial wearable gaze trackers [49, 50]. However the SDK being supplied are limitedly opened for developing new concept of interface and the design was still bad causing resistance to use it. Extremely high price was another obstacle. To achieve these goals, we set our target performances as Table.1 first.

Table.1 Target performances for single display device control

Accuracy	±1.0 degree error @800mm distance from display
Real-time processing	operating at least 30 fps
Head free	±30 degree tolerance with respect to display @800mm distance

As a result, a multi-modal wearable gaze interface was developed. The developed gaze tracker was shaped like the google glass [14] and was designed to be light and fancy. For immediate device switching and high robustness in various light conditions, the infrared boundary+ID marker was designed and installed to the devices and a novel recognition algorithm was suggested. To enhance accuracy and avoid Midas touch [4, 9], we used the gaze only for pointing and added a manual input modality for confirmation. The converted pointing data is transferred to the display device wirelessly by router based data transfer (UDP).

1.3.2 Gaze based multi display devices control

At this point, the developed gaze interface is not able to control multi devices, because it cannot distinguish which device is selected by the gaze. This means for multi devices control, developed wearable gaze tracker has to recognize the identification of device user gazes, that is, device awareness. As our goal is Command Everything with Eyes, we segmented 'Everything' into two parts, display device and non-display device. Therefore, for our second approach, we decided to control multi display devices with wearable gaze tracker (see Fig 6). To achieve it, we first set our target performances as Table 2.

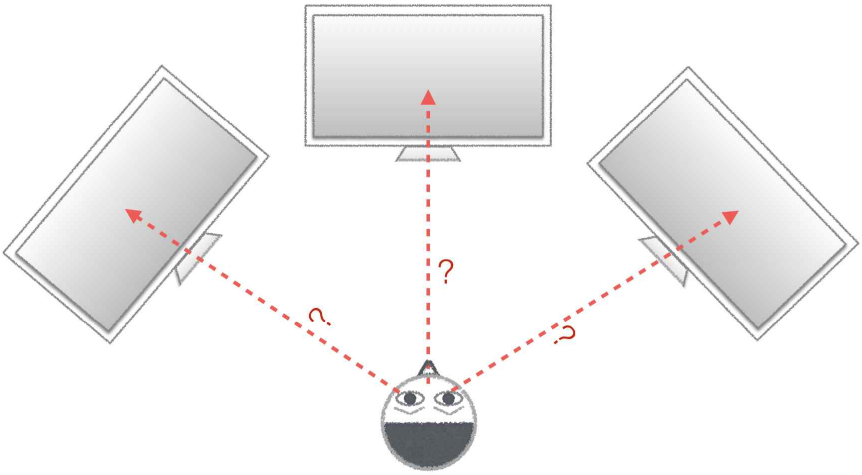


Fig.6 Gaze based multi display devices control interaction

Table.2 Target performances for multi display devices control

Accuracy	±1.0 degree error @800mm distance from display
Real-time processing	operating at least 30 fps
Head free	±30 degree tolerance with respect to display @800mm distance
Immediate response	Immediate boundary recognition and interested device switching : under 1/30 sec.
Target performance	Equivalent performance and user experience as widely used interfaces, touch and mouse

There could be various kinds of device awareness methods. In this paper infrared boundary+ID marker recognition method was designed and novel recognition algorithm discerning the device was developed. The infrared LED markers guaranteed high robustness in variable light conditions and the developed boundary+ID recognition algorithm enabled real-time response and action. To improve accuracy and speed, the gaze was only used for pointing and added a manual input, the wireless button, for confirmation. As a result, the multi-modal wearable gaze interface was developed. Detailed system instructions and implementation are described in the following section.

To verify the developed interface, a series of tests were conducted and to prove applicability, results were compared with the two most widely used interfaces: mouse and touch. User experience factors were also rated through user interviews. (See Chapter 3)

1.3.3 Gaze based non-display devices control

The other approach of Command Everything with Eyes is to control non-display devices with eyes (See Fig 7). Different from display device generally rectangle shape, non-display devices have enormous kind of form and design. So the boundary IR marker method applied for recognizing display device is not appropriate for non-display devices. To solve this problem, we referred to the nature, bat. Although bat has very bad eye sight, using ultrasonic waves, it can find the way, obstacles and insects for food.

In the same way, we designed gaze sound matching ID recognition method. In this method, what user doing is just look at non-display device to control and makes command. Once command emerges, virtual ultrasonic wave is transferred through router and each non-display devices make their own frequency's sound. By sound localization algorithm and frequency analysis using several microphones on the wearable gaze tracker, user can find where each device's sound comes from. Finally, gaze tracker finds a sound in the same direction of the gaze and sends final commands.

In this paper, for verifying the concept of gaze sound matching ID recognition method, two microphones are added to wearable gaze trackers. Other systems are just same as gaze interface for multi display device control. (See Chap. 4)

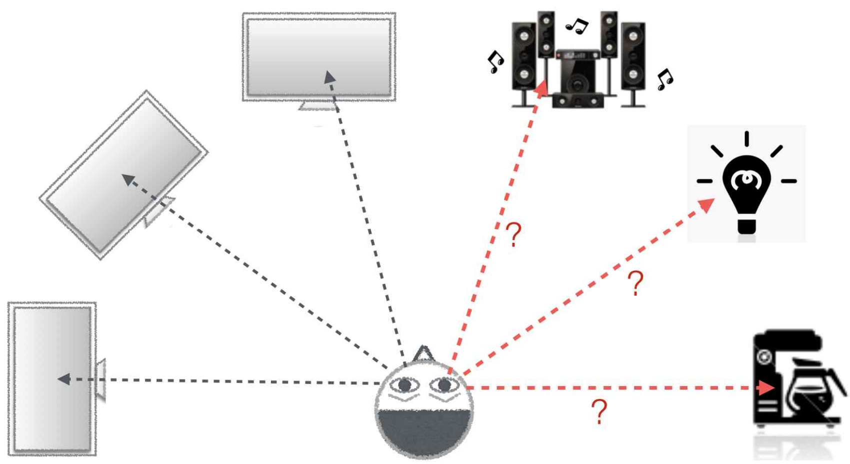


Fig.7 Gaze based every devices control interaction including non-display devices

Chapter 2

Implementation of wearable gaze interface for single display control

2.1 Overview

At first, we developed high performance wearable gaze interface appropriate for single device control. Of course, commercial wearable gaze trackers [49, 50] can be used for this research. However, we need this intermediate step to make the most appropriate gaze interface for multi device because it gives the ability of unlimited customizing to the rawest level. Developed gaze interface should have high accuracy, speed, and robustness while enabling immediate device switching. In following sections, we describe the implementation process which consists of four parts: Hardware, Software and independent input modality.

2.2 Hardware

The hardware of developed gaze interface mainly consists of four parts: a goggle shaped frame, scene camera, eye camera with IR lighting and main processor (See Fig 8).

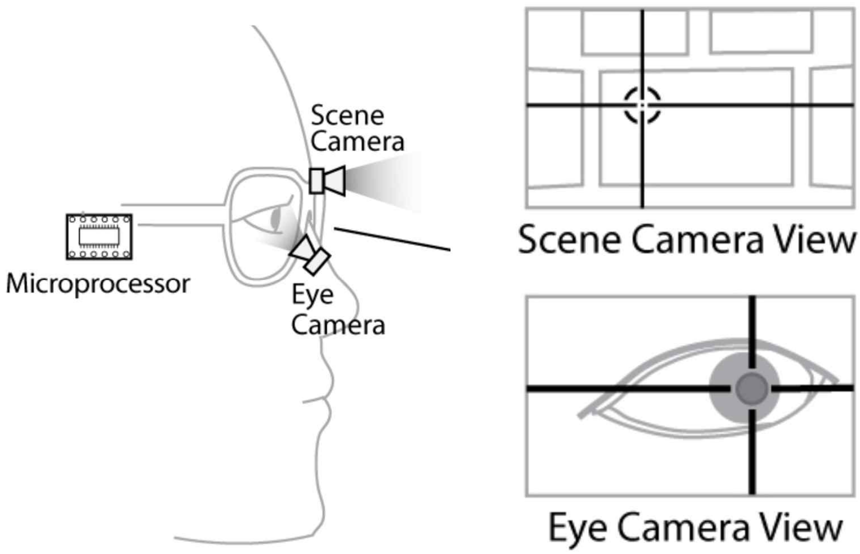


Fig.8 Hardware overview

2.1.1 Frame

Since most previous head mounted gaze trackers were large and heavy, they were not suitable for practical use and users resisted to it (See Fig.9). To improve usability, we attempted to design small, light and fancy wearable gaze tracker as possible.

Fig.9 Wearable type gaze trackers

The shape of gaze tracker is based on goggle type wearable device like google glass [14]. To minimize the whole weight, additional parts not related to the scene camera and eye camera are excluded in design process. Main frame was selected among commercial goggle manufacturers and additional brackets used for fixing cameras were designed from CAD program, Solidworks [46]. To shorten the time for making prototype, fused deposition modeling (FDM) type 3D printer was used [40] and PLA was used for printing material. Every version of prototype gaze tracker is shown in Fig 10.

Fig.10 Frame designs and development

2.1.2 Cameras and lighting

Two cameras are installed in gaze tracker, eye camera and scene camera. All of them are specially designed and manufactured customized for wearable gaze tracker by Huentek [39].

For the eye camera, 35x10x5 (WxHxD, mm), 640x480 resolution and 30fps infrared camera is used. As a dark pupil extraction algorithm is utilized to extract the pupil [47], 850nm infrared LED lights are installed on the camera. The eye camera position can be easily adjusted by the user using a flexible wire (See Fig 12).

A scene camera is installed to capture the front side image including the IR boundary+ID markers (IR markers will be introduced in Chapter 3). It has same physical size and frame rate but, the resolution can be changed from 640x480 to 1920x1080. The higher resolution scene camera has, the more accurate gaze position is determined. But it has to be adjusted properly due to increasing processing burden for higher resolution. To filter out all ambient light except infrared markers, a physical infrared pass and visible ray filtering film is attached in front of the lens. In particular, to minimize parallax error [25], a scene camera is equipped as close and parallel to the tracked eye as possible. Image of each camera is shown in Fig 11.

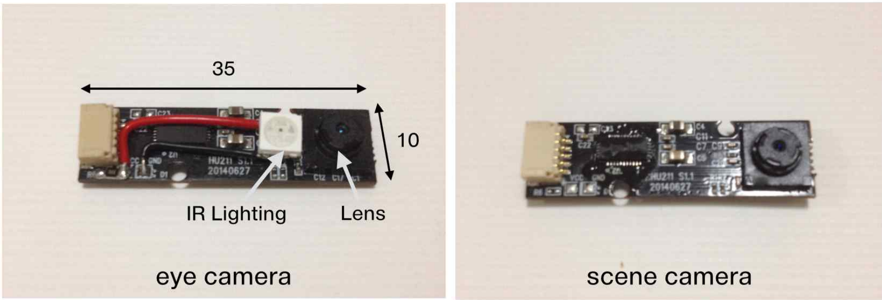


Fig.11 Camera specification, eye camera (left) scene camera (right)

2.1.3 Processor

Recently, as well as the powerful mobile processors of smartphone, open source mobile processors are introduced like Arduino, Raspberry pi and Edison [52,54,38] etc.. In this paper, we used the laptop computer as processing module for guaranteeing processing power. Macbook Pro 2013 is used and its specification is Intel i7 2.3Gh processor for CPU and 16 Gbyte Ram. Although laptop computer is used instead of the mobile processors for guaranteeing processing power, but laptop can be replaced easily as ongoing developments in wearable technology can allow low cost and high performance processors.

2.1.4 Hardware integration and result

Through verifying the hardware design with several tests and user interviews, final version of wearable gaze tracker is designed as Fig 12. The weight of wearable gaze tracker is under 150g.

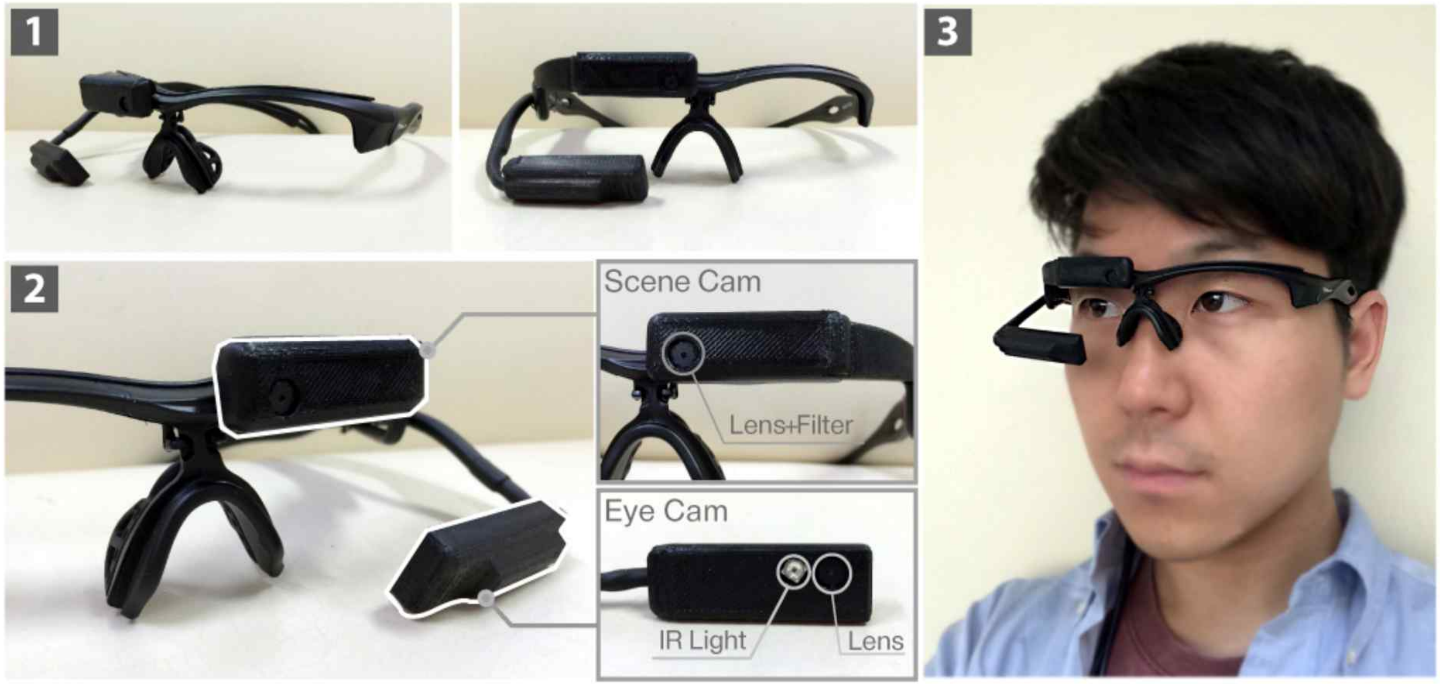


Fig.12 Wearable gaze tracker, 1) appearance, 2) detail view, 3) wearing,

2.2 Software

For the gaze estimation, not only tracing eye position, pupil extraction is necessary but head pose estimation also needs to be developed (Fig.13) [57]. Implemented pupil extraction and head pose estimation algorithm will be reviewed in following section

.

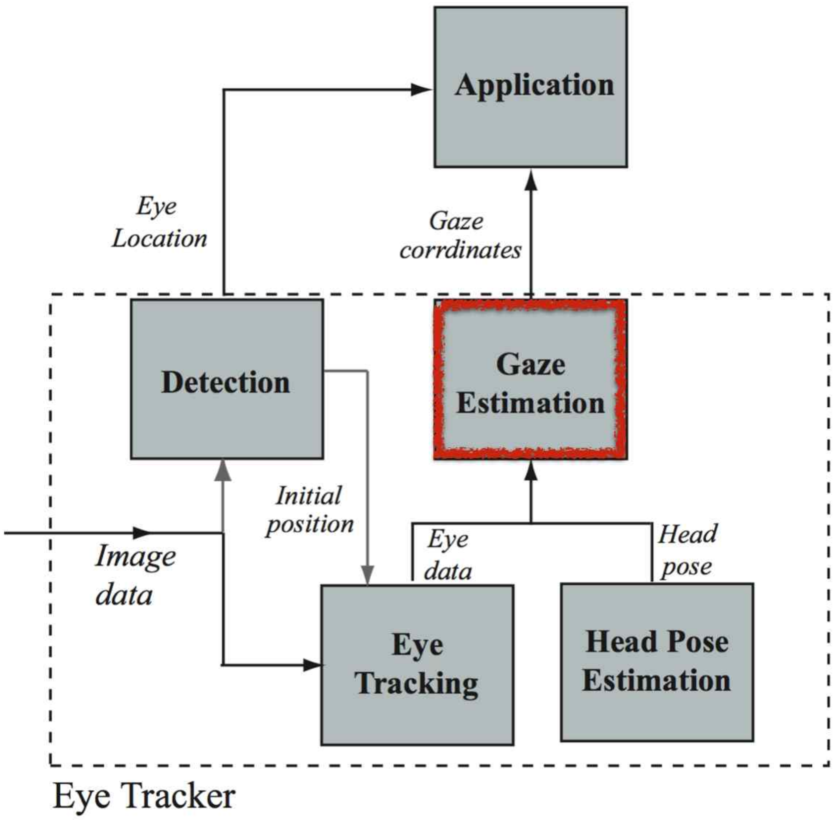


Fig.13 Schematic diagram of gaze estimation algorithm [57]

2.2.1 Pupil detection

2.2.1.1 Theories

Generally pupil detection algorithms are classified into two approaches, feature based method and model based method. The main difference between feature based method and model based method is that a criteria (e.g., a threshold) is needed or not to detect pupil [58].

Feature based method [53,42,43] extracts feature images of pupil by thresholding. That means, whether the specific parameters of one pixel are over (or under) a threshold decides the point is the feature (=pupil) or not. The determination of an appropriate threshold is typically left as a free parameter that is adjusted by the user. The parameters used for threshold vary widely across algorithms but most often rely on intensity levels or intensity gradients [58].

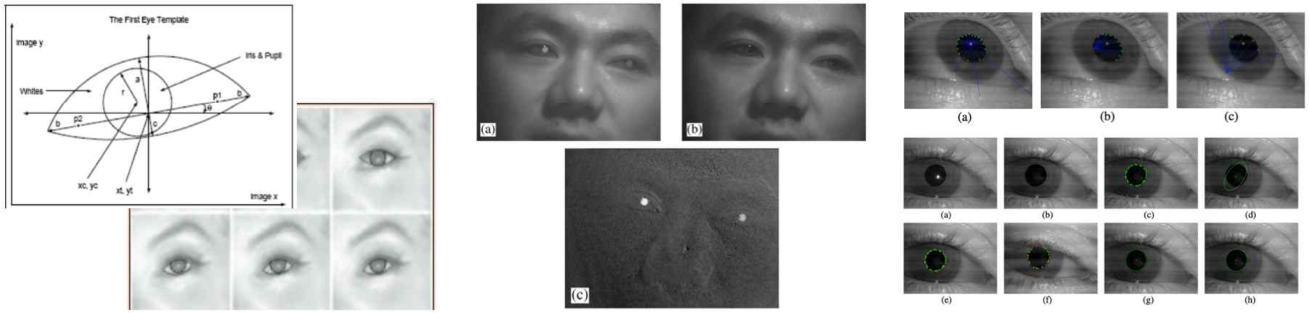
On the other hand, model based approaches [59,60,41] do not detect features by thresholding but rather find the best fitting model that is consistent with the image. In model based method, the eye models such as the limbus or pupil are designed shaped of ellipse or circle first, and the best fitting position in captured eye image is extracted by integro differential operators [58].

Generally, the model based method can detect the pupil center more precisely than a feature based method, but the model based method requires more searching time and processing power than the feature based method.

Finally, there is hybrid pupil extraction method [35,36,58] which combines the advances of feature based and model based method to achieve a proper tradeoff between accuracy and run time performance. One of the useful hybrid pupil extraction algorithms is 'Starbust'[58]. The algorithm begins by locating and removing the corneal reflection from the image. Then the pupil edge points are located using an iterative feature based technique. An ellipse is fitted to a subset of the detected edge points using the Random Sample Consensus (RANSAC) paradigm. The best fitting parameters from this feature- based approach are then used to initialize a local model- based search for the ellipse parameters that maximize the fit to the image data [58]

In Fig 14, the comparison of various pupil extraction method are described and representative model based, feature based and hybrid based pupil extraction method is introduced.

Method	Pre-Requirements	Approach	Uses Learning	Used Feature	Used Model/Learning Scheme
Asteriadis [2]	Detected face	Feature Based	-	Edges	Eye model for init + edge crossing count
Jesorsky [27]	Converged face model	Model Based	X	Edges	Hausdorff distance on eye model
Cristinacce [13]	Detected face	Model Based	X	Pixels	PRFR + AAM
Türkan [42]	Detected face	Hybrid	X	Edges	SVM
Bai [3]	Detected face	Feature Based	-	Gradient	-
Wang [47], [48]	Detected face	Model Based	X	RNDA	Boosted classifiers cascade
Campadelli [7]	Detected face	Hybrid	X	Haar Wavelets	SVM
Hamouz [24]	Correct constellation	Model Based	X	Gabor filters	Constellation of face features + GMM
Kim [30]	Normalized face images	Model Based	X	Gabor jets	Eye model bunch
Niu [36]	Detected face	Model Based	X	Haar Wavelets	Boosted classifiers cascade
Wang [46]	Both eyes visible	Hybrid	X	Topographic labels	SVM
Huang [26]	Detected face	Hybrid	X	Mean, std, entropy	Genetic Algorithms + Decision trees
Reale [38]	Detected face	Model Based	-	Pixels	Circle Fitting
Asadifard [1]	Detected face	Feature Based	-	Pixels	CDF filtering
Timm [41]	Detected face	Feature Based	-	Gradient	-
Kroon [32]	Detected face	Model Based	X	Pixels	Elastic bunch graph + LDA



1) Model Based detection 2) Feature based detection 3) Hybrid detection

Fig.14 Pupil extraction theories [36,59,58,42]

2.2.1.2 Algorithm development

For pupil extraction, we combined feature based method and model based method, which means applying hybrid method.

The process of gaze tracking starts by capturing raw eye image. The outside region of the eye image is masked with a gray color. Then, the candidates of pupil position are extracted by feature based method. Generally, to detect pupil candidate region, dark pupil extraction or bright pupil extraction is applied. The dark pupil technique [15,64,47] can be achieved to give IR light inconsistent to gaze. In this case, the intensity of the pupil region is darker than surroundings because the reflected IR light cannot come out through pupil. On the other hands, in the bright pupil technique [15], by giving IR light parallel to gaze, the intensity of the pupil region is brighter because of reflected IR light. Appropriately set intensity threshold determines the region of the pupil candidates and the binary pupil candidate image is achieved. After this feature based process, an ellipse fitting process, model based method, then be applied to find final pupil region.

We refer to the libraries suggested by John H [18] for the pupil extraction algorithm and modified it proper to our system. We also applied several image filters to improve accuracy, for instance, the median and average filters to eliminate noise blobs like eye blows. The pupil extraction process is shown in Fig.15.

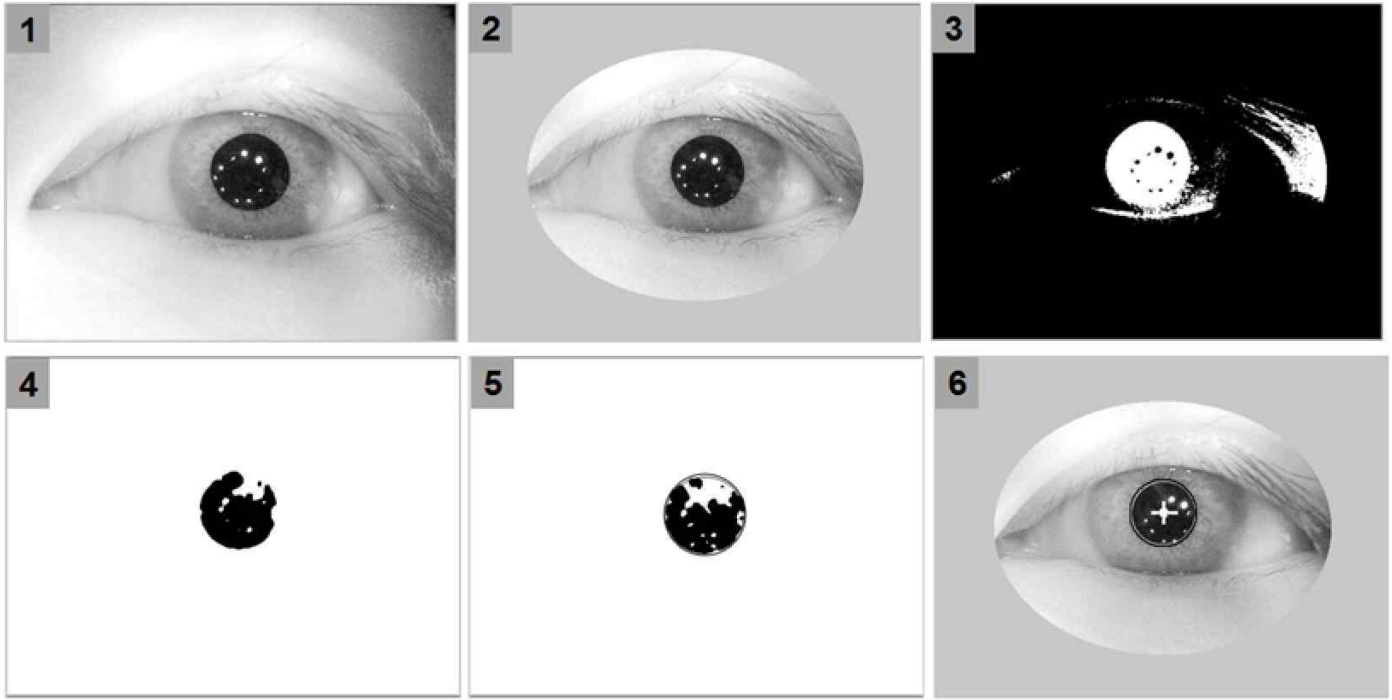


Fig. 15 Pupil extraction process: 1) original image, 2) masking, 3) blob extraction, 4) filtering, 5) ellipse fitting, 6) extracting center of pupil

2.2.1.3 Position mapping between eye and scene image

The user gaze position in a scene image is finally determined by coordinate matching between the pupil moving boundary and the scene image boundary (See Fig 16). For coordinate matching, a bilinear coordinate transform method is used and the user needs to calibrate at least four boundary points. In this paper, we applied 9 boundary calibration points. The bilinear coordinate transform method and its equation is shown in Fig 17.

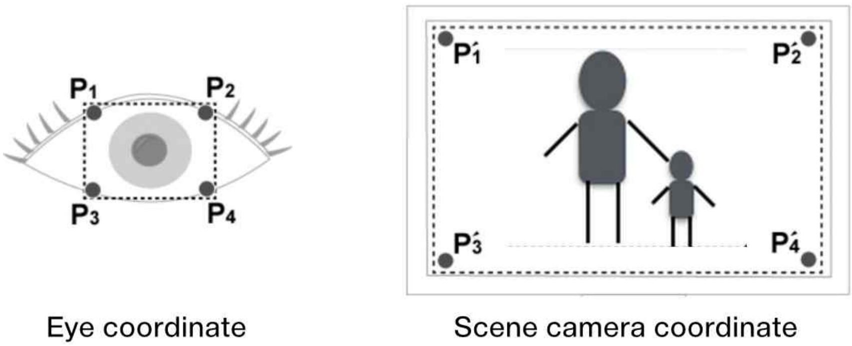
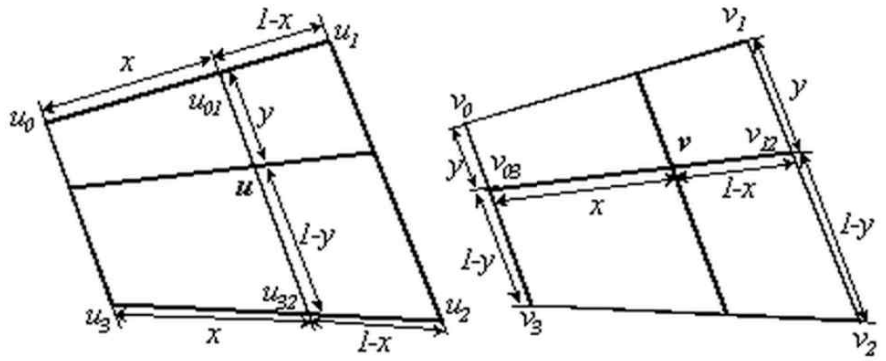


Fig.16 Coordinate matching between eye coordinate and scene camera coordinate



$$\begin{bmatrix} x & y \end{bmatrix} = \begin{bmatrix} uv & u & v & 1 \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ a_{41} & a_{42} \end{bmatrix},$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \\ a_{41} & a_{42} \end{bmatrix} = \begin{bmatrix} 1 & -1 & -1 & 1 \\ -1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_0 & y_0 \\ x_2 & y_2 \\ x_1 & y_1 \\ x_3 & y_3 \end{bmatrix}.$$

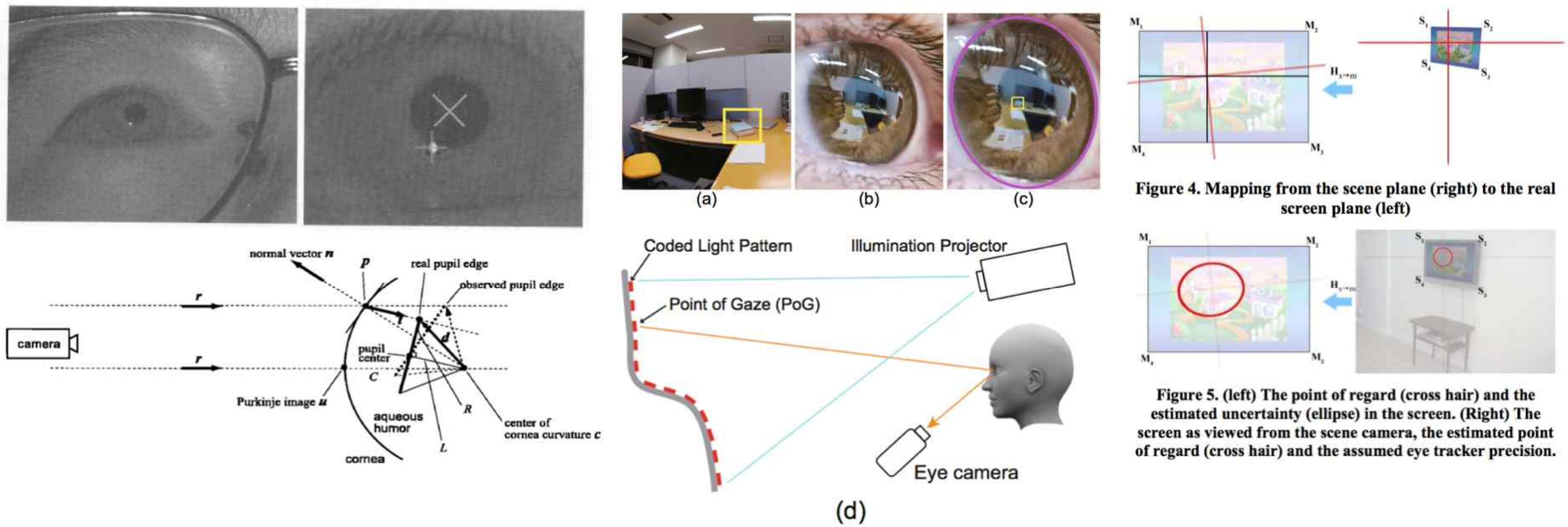
Fig.17 Coordinate matching equation : Bilinear coordinate transform

2.2.2 Head pose estimation

2.2.2.1 Theories

With the pupil extraction and coordinate mapping with scene camera, we can only tracking the gaze position in scene image. However, to manipulate the pointer on a display coordinate using a wearable gaze tracker while allowing the user to move his head freely, another essential process, head pose estimation method needs to be developed.

Various head pose estimation methods were suggested before, but most researches are developed for fixed type gaze trackers such as purkinje reflection tracing calibration [5,23] or head rotation recognizing using depth sensing camera [7]. Although some methods for wearable gaze tracker to detect the display boundary were studied such as corneal reflection [34,37], geometric approach [8] or fixed visual tags [20], it is considered not to be applicable to practical applications (See Fig 18).



1) Purkinje reflection tracing calibration 2) corneal reflection gaze estimation 3) Display recognition method

Fig.18 Representative head pose estimation methods [56,37,8]

2.2.2.2 Algorithm development

One of the useful solutions tracking users' head pose is to recognizing and tracing display boundary region using scene camera. But, for practical applications, the display boundary should be traced in real time minimizing processing burden.

To achieve this goal, four boundary markers are installed on the corners of the display device. The ideal gaze interface is able to detect and identify the screen in variable light conditions and even when the screen is turned off. It should also be fast enough to allow for real-time processing [8]. Therefore, as boundary marker, we applied infrared boundary LED markers. Infrared makers have high robustness in variable lighting circumstance. As infrared LED markers can always be on with low energy consumption, the display can be controlled and classified even when the display is off. The Infrared marker installation and captured image by scene camera are shown in Fig.19.

Once the gaze position enters the display boundary, the gaze position related to the boundary markers is converted to the position related to the display resolution by bilinear transform.

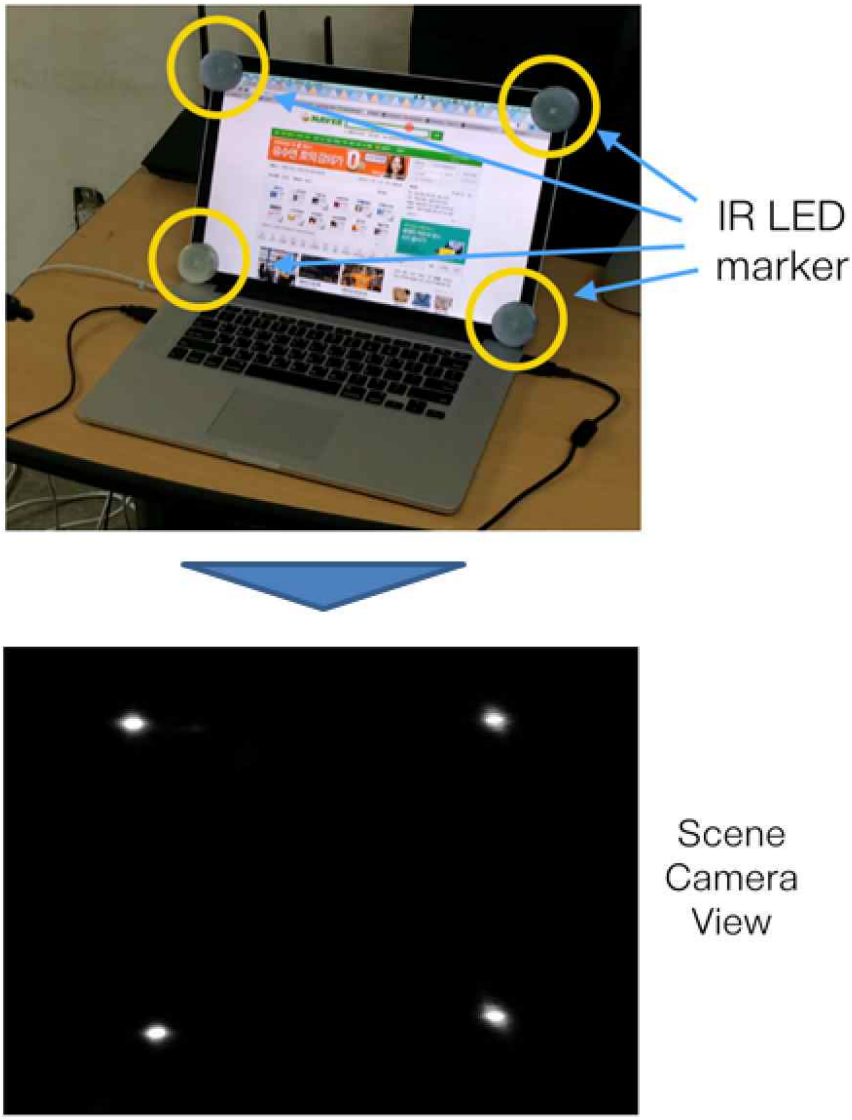


Fig. 19 IR boundary markers and captured image by scene camera

2.3 Independent input modality

In general, the gaze interface outperforms in the pointing task, but it is not appropriate for confirmation. In previous studies, eye gestures like blinking [26], gaze dwelling time [27], and head gestures [6] were used to confirm behavior in gaze interface. However, these take too much time to be practical when compared to click and touch. Midas touch, which is an unintentional action, is another problem [4]. This means that the gaze interface by itself has difficulty being a practically applicable interface. Instead, Zhai et.al suggested the gaze interface add a manual input button to improve speed and accuracy [9]. Stellmach et.al [16] and Jason Turner et.al [17] also suggested meaningful solutions for applying multi-modality combining gaze and manual input.

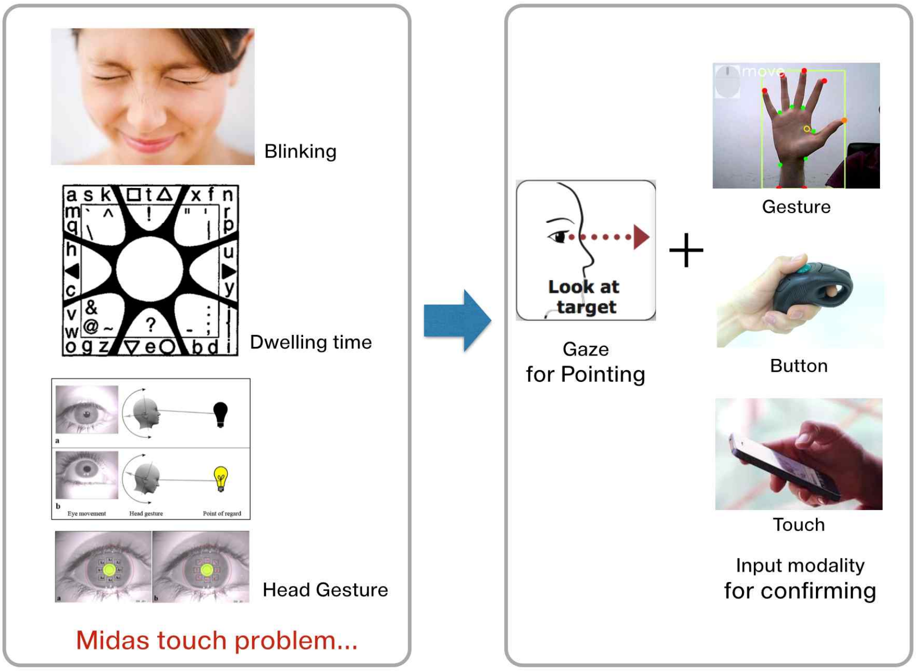


Fig. 20 Midas touch problem (left), independent input modality (right)

In other words, the gaze interface outperforms in the pointing task, but it is not suitable for confirming. In this respect, an extra manual input modality-customized wireless mouse button- is added for confirmation while gaze is only used for pointing. Therefore, a multi-modal gaze interface was developed using gaze for pointing and manual button for confirmation (See Fig.21).

Fig. 21 Multi-modal interface of gaze pointing and manual button confirmation

2.4 Performance test

To evaluate the gaze tracker accuracy, gaze pointing tasks for 25 reference points were conducted while the user was located 0.8 m from the 530 x 300 mm monitor. A coordinate matching calibration was done on 9 points and users can move their head freely during evaluation tests, which was repeated 20 times. The average error for distance was 9.14 mm and angle was 0.65 degrees. The standard deviation for distance was 4.15 mm and angle was 0.29 degrees. Fig.22 shows the results of the evaluation test and Table 3 shows the performance comparison with commercial gaze tracker.

Table. 3 Performance comparison with commercial gaze tracker [49]

	Tobii (X2-60 model)	**Developed interface**
Accuracy (80cm from display)	0.5 deg.	0.6 deg
Response (Hz)	60 Hz	60 Hz
Head free limit (WxH cm)	50 x 36 cm	**50 x 50 cm**
Max/Min distance (from display)	45 ~ 90 cm	**55 ~ 120 cm**

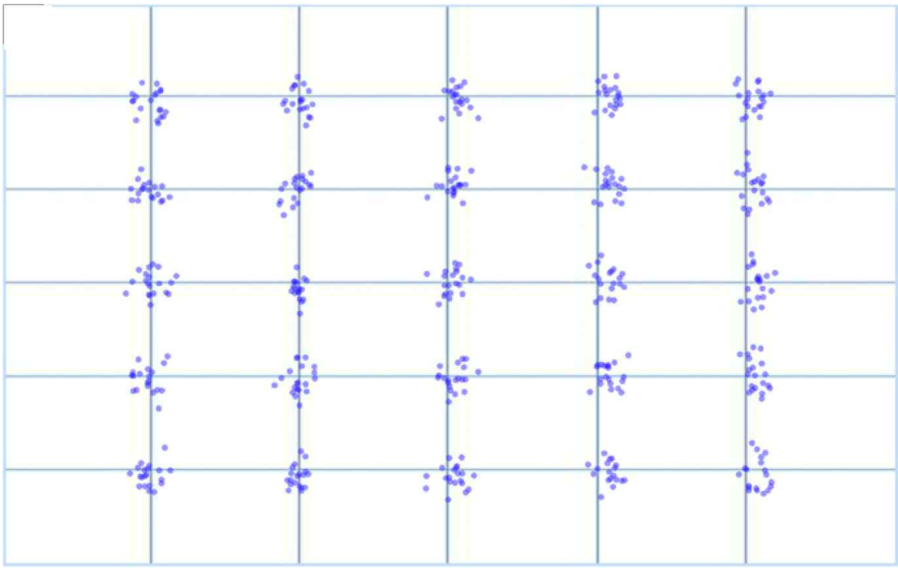


Fig.22 Accuracy evaluation result graph of developed wearable gaze tracker

2.5 Result and application

We developed high performance wearable gaze interface appropriate for single device control. Developed gaze interface can be fully customized for multi device control and it shows high accuracy, speed, and robustness tracking display device in real time. For practical application, we developed a multi-modal control interface GaFinC: Gaze and Finger Control interface [2]. GaFinC is basically finger gesture based interface for CAD application and it can track precise hand positions, recognizes several finger gestures. By utilizing an independent gaze pointing interface for setting the point of interest, it can have better intuitiveness than a conventional mouse interface while maintaining a usable level of comfort.

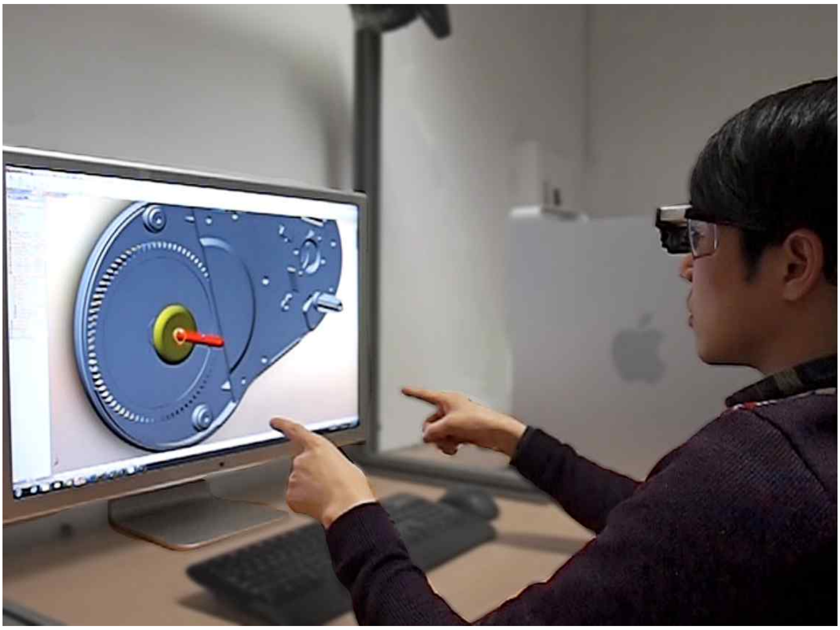


Fig. 23 Application - GaFinC : Gaze and Finger Control interface for 3D model manipulation in CAD application [2]

Chapter 3

Implementation of gaze and multi devices interaction

3.1 Previous works

At this point of the research, the developed wearable multi modal gaze interface is not able to control multi devices, because it cannot distinguish which device is selected by the gaze. In this section, we will expand researching focus to multi device control.

Many studies are suggested which prove the high potential for wearable gaze interface and it can be improved by combining the gaze and manual input to the multi-modal interface. However, most previous studies focused on improving the gaze interface for single device control while studies for multi device control using gaze were rare. For multi device control with gaze, we need to discern the device and an additional device awareness technology is required.

Mardanbegi et.al detected a screen through brightness binarization [8]. However, in his research, the detecting ability was affected by the ambient light condition and display brightness, and if multiple displays appeared and overlapped at the same time (which occurs frequently in practical situations), it was difficult to recognize the boundary rectangle.

Mandalbegi et.al also suggested using a temporary display identification tag [11]. When a display quadrilateral appears in the scene image, the eye tracker requests the server to show identification tags, like a QR code, and it appears for a short time on all the displays until the eye tracker has identified the tag. But, this means the user has to wait several seconds until the identification tag appears, recognizes, and disappears.

Turner et.al suggested a head mounted gaze interface that could control both hand held devices and public displays [19]. He also enhanced user experience and decreased user fatigue in switching devices by implementing a dual scene camera. But his suggestion is limitedly applicable for only two devices.

Fig. 24 Previous studies of multi devices control using gaze interface [8,11,19]

3.2 Algorithm development

For real-time interaction and high robustness, we added extra infrared markers to identify the device. Fig 25 shows system overview of multi display device control gaze interface applying ID markers. In this method, additional IR LEDs are installed on the top side of the display. In Fig 26, the side B1 to B2 is divided into certain equal parts and each part plays a role in one bit of binary ID. The display ID is decided by the combination of powered LED installed in each region. In this paper, we implemented a 3 bit ID marker that can recognize 7 different display devices, not including ID '0' for all markers off. Fig.27 also shows some examples. The variation of ID marker can be extended by combining the blinking frequency or intensity of LEDs. All of the identification process was completed under 1/30 sec. and users hardly felt any delays switching to interested devices.

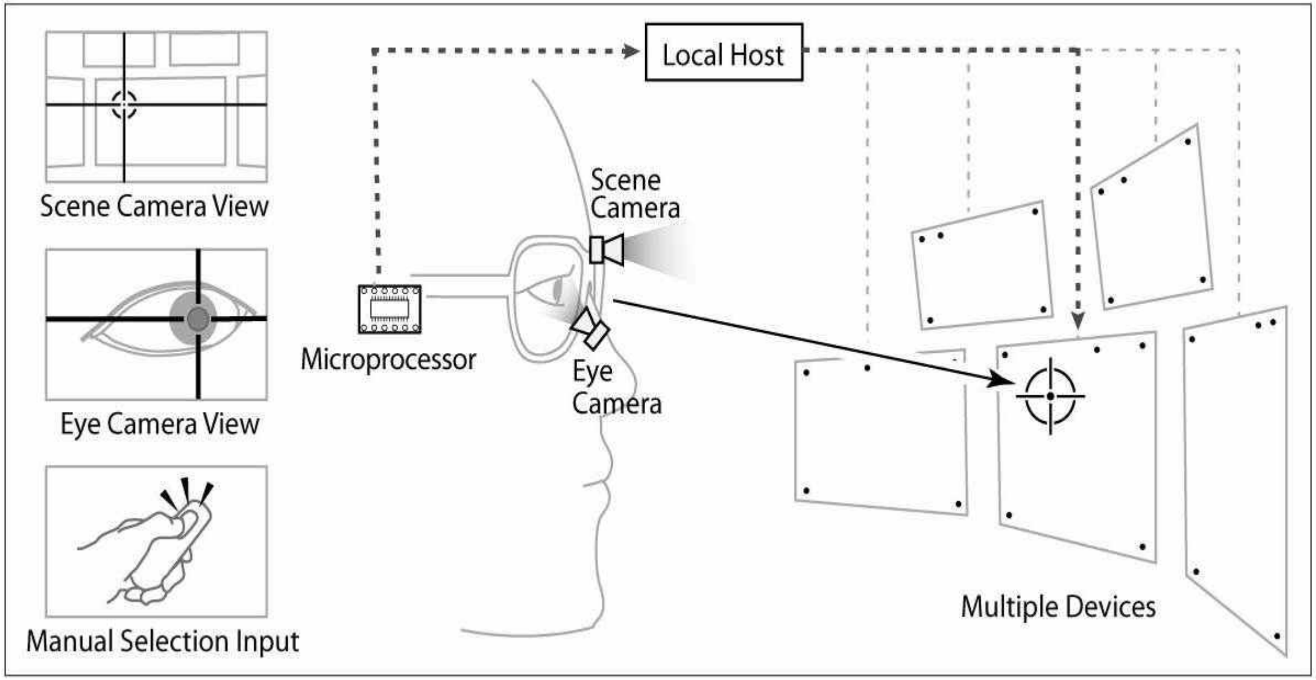


Fig.25 System overview and schematic diagram of the gaze interface for multi display device control

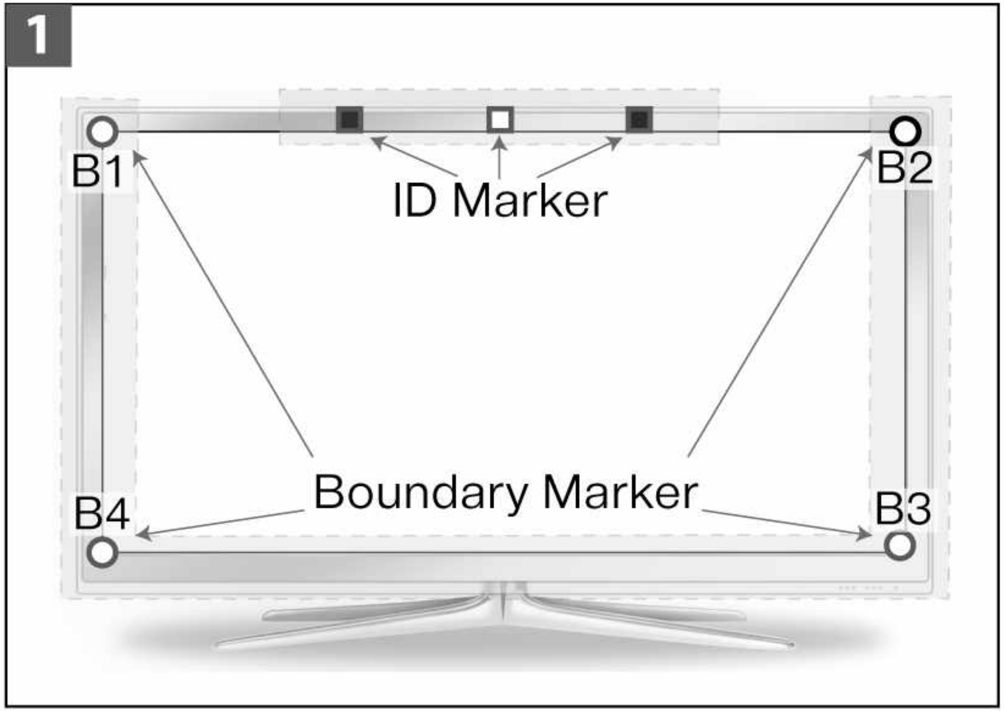


Fig.26 Installation of Infrared LED boundary and ID markers

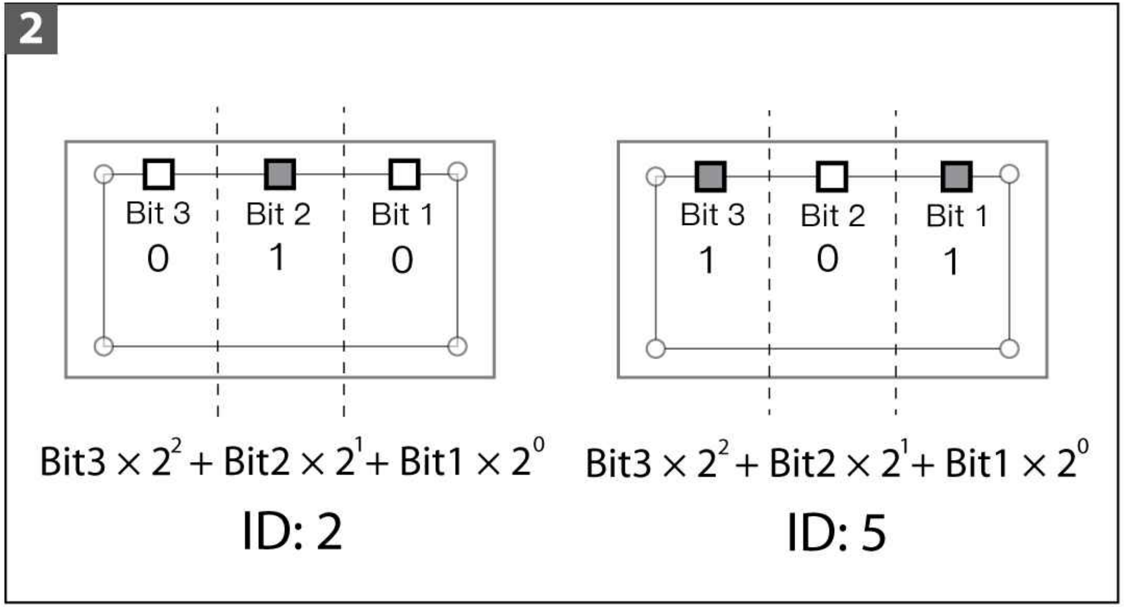


Fig.27 Examples of Infrared LED ID marker

3.3 Integration

As a physical infrared pass and visible ray filtering film is attached in front of the lens to filter out all ambient light except infrared markers, the image captured from scene camera filled with a number of bright points (See Fig 28). We have to recognize the actual boundary rectangle among a number of candidate boundary points and noise. To segment meaningful points, novel boundary extraction algorithm is developed. The boundary recognition process is described in Fig.29 and the extracting process is as following.

Input: scene image filled with bright points which is candidates of interested boundary and ID

Output: display's boundary region and ID

Procedure:

Iterate

Stage 1:

- Binalyze image : binalize image into point blob and background
- Making points list : Extract the list of points filtered by infrared ray passing and visible ray blocking
- Making quadrangles list: Create the list of candidate quadrangles by combining extracted points

Stage 2:

- Parallelogram filtering : Filter out quadrangles by the distance between two middle points of diagonal corner
- Rectangles filtering : Filter out parallelogram range of the internal angle less or over 90+-alpha
- ID marker filtering : Filter out rectangles without the ID marker

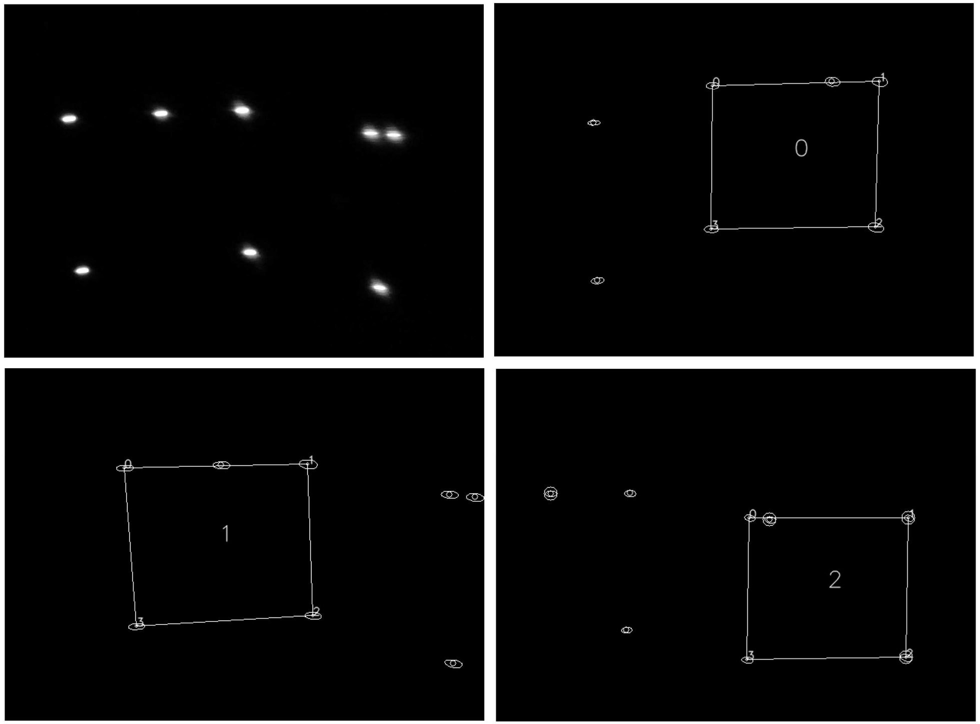


Fig.28, Scene image filled with IR boundary and ID marker candidates and its recognition result

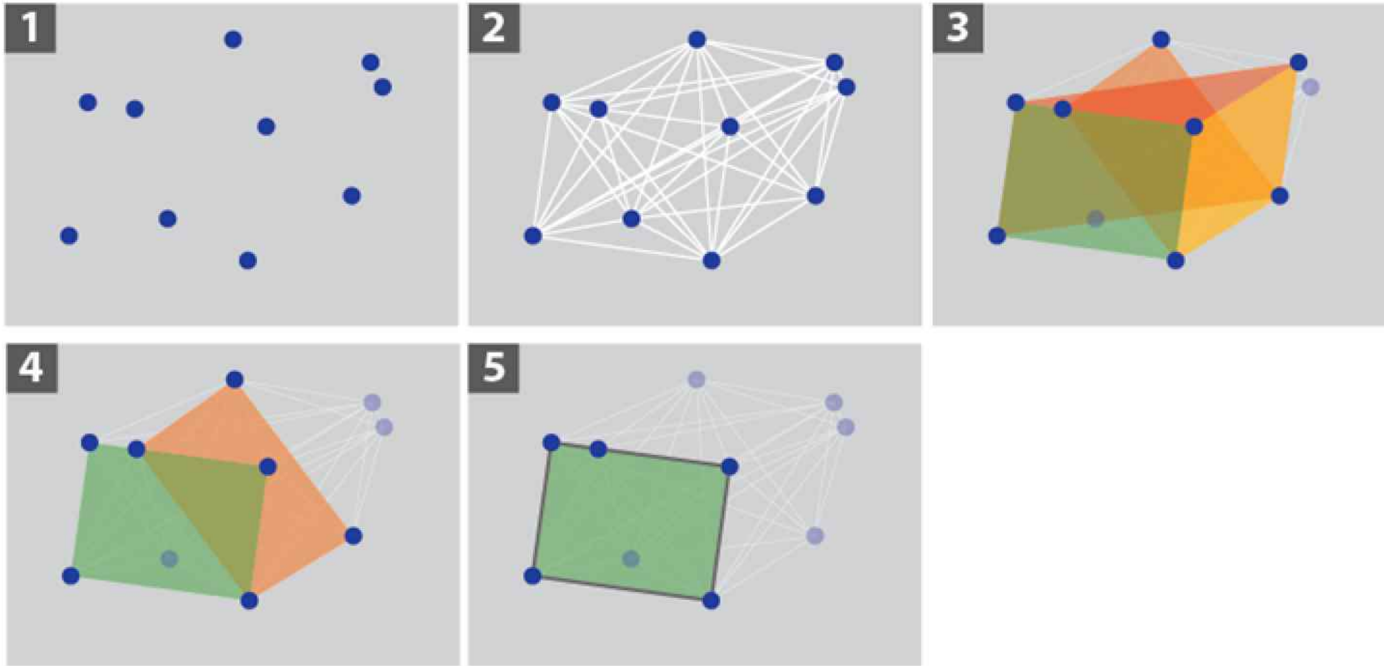


Fig.29 Boundary recognition process, 1) Making points list, 2) Making quadrangles list, 3) Parallelogram Filtering, 4) Rectangles filtering, 5) ID mark filtering : filter out rectangles that do not have the ID marker

3.4 Resolution adaptive position mapping

After recognizing the device ID, the gaze position in coordinates of the boundary of interested display is converted to the position in coordinates of the display resolution by bilinear transform. However, as the resolutions may be different from each other, the adaptive position mapping should be done according to device ID. For adaptive position mapping, each display resolution is predefined along with device ID. The final position P(X’,Y’) of the interested display coordinate is calculated by the ratio A:B of the calibration rectangle used in the coordinate matching of the pupil and the scene image, the ratio A’:B’ of interested display rectangle, and gaze position P(X,Y) related to boundary markers in equations (1). The mapping process is shown in Fig.30.

$$[X' \quad Y']^T = \begin{bmatrix} \frac{A'}{A} & \frac{B'}{B} \end{bmatrix} \cdot [X \quad Y]^T \qquad (1)$$

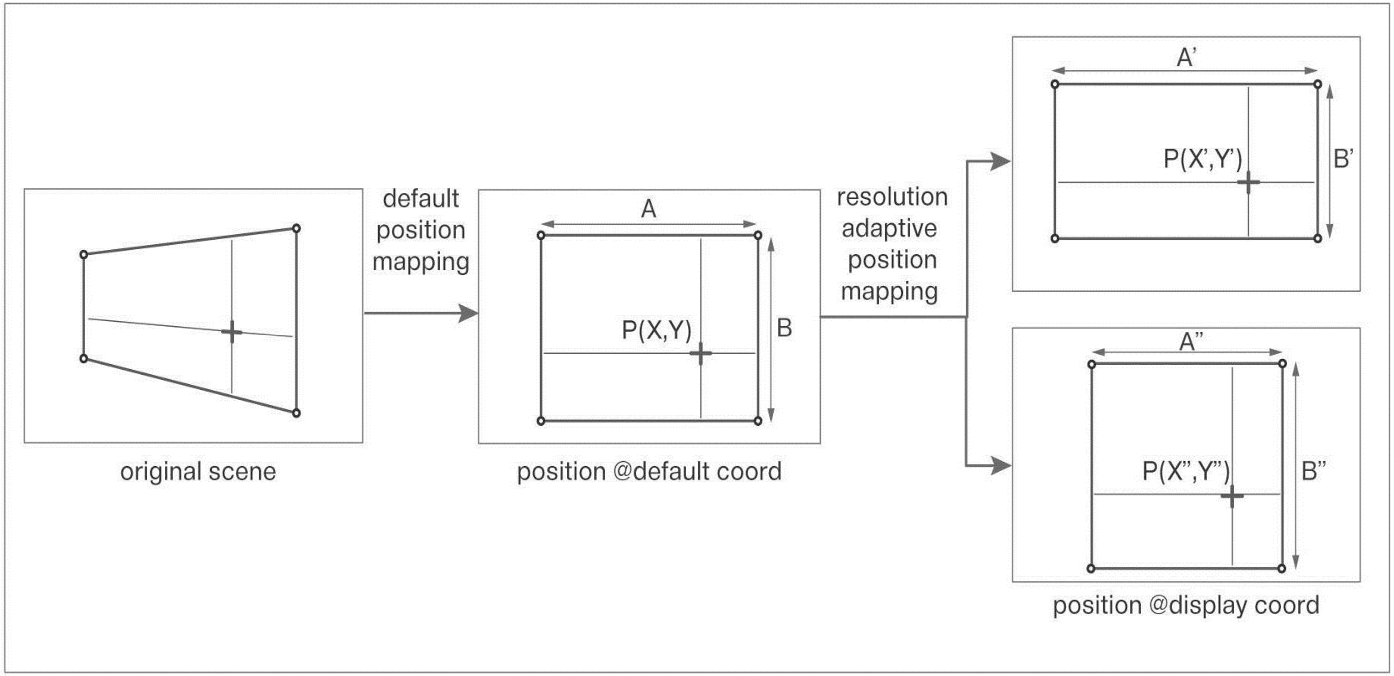


Fig.30 Resolution adaptive position mapping process

3.5 Implementation of system for IoT environment

The user's gaze position and manual input action are transmitted to the interested device wirelessly through router based data transfer by UDP (Fig. 31). For using the developed gaze interface, a gaze processing computer along with control devices have to be connected to the same Wi-Fi router. As you can find in Gartner emerging technologies hyper cycle [55], IoT environment technology is at the top expectation position now (See Fig. 32). This means, every device will be connected to router. In this environment, device ID will be changed as gaze interface connected to different router. For example, at home, device ID:2 is light while device ID:2 is switched to PC in office (See Fig.33). This means, in router based device ID environment, we can control infinite devices with limited device ID by combination of the router.

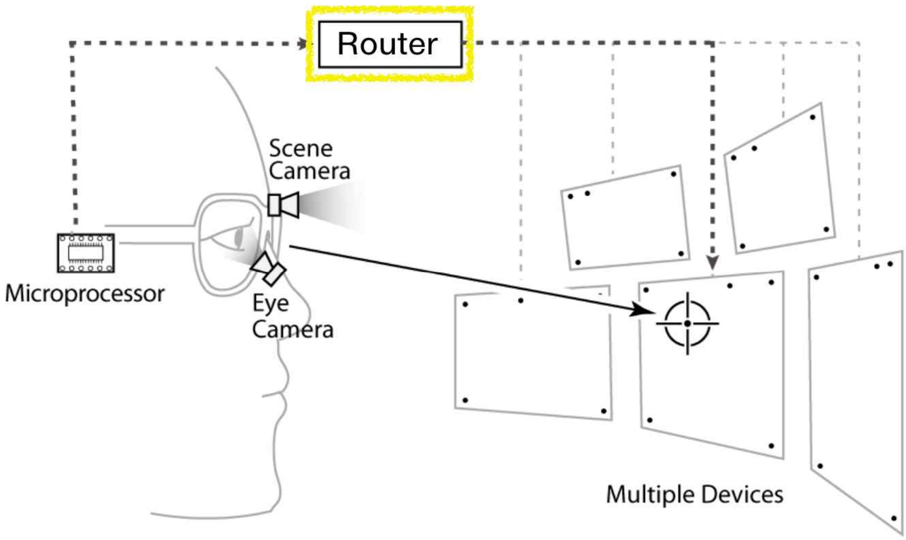


Fig.31 Router based command transfer environment

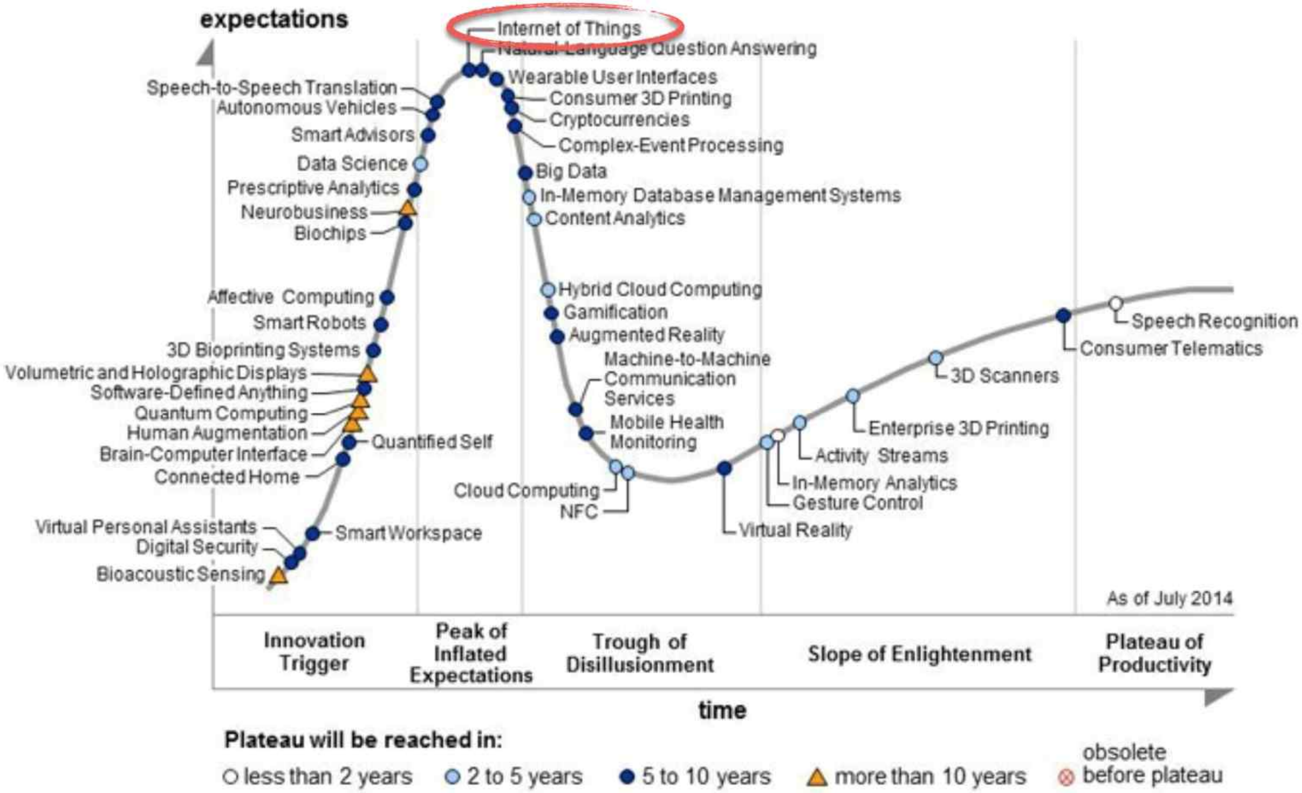


Fig.32 Gartner emerging technologies hyper cycle 2014 [55]

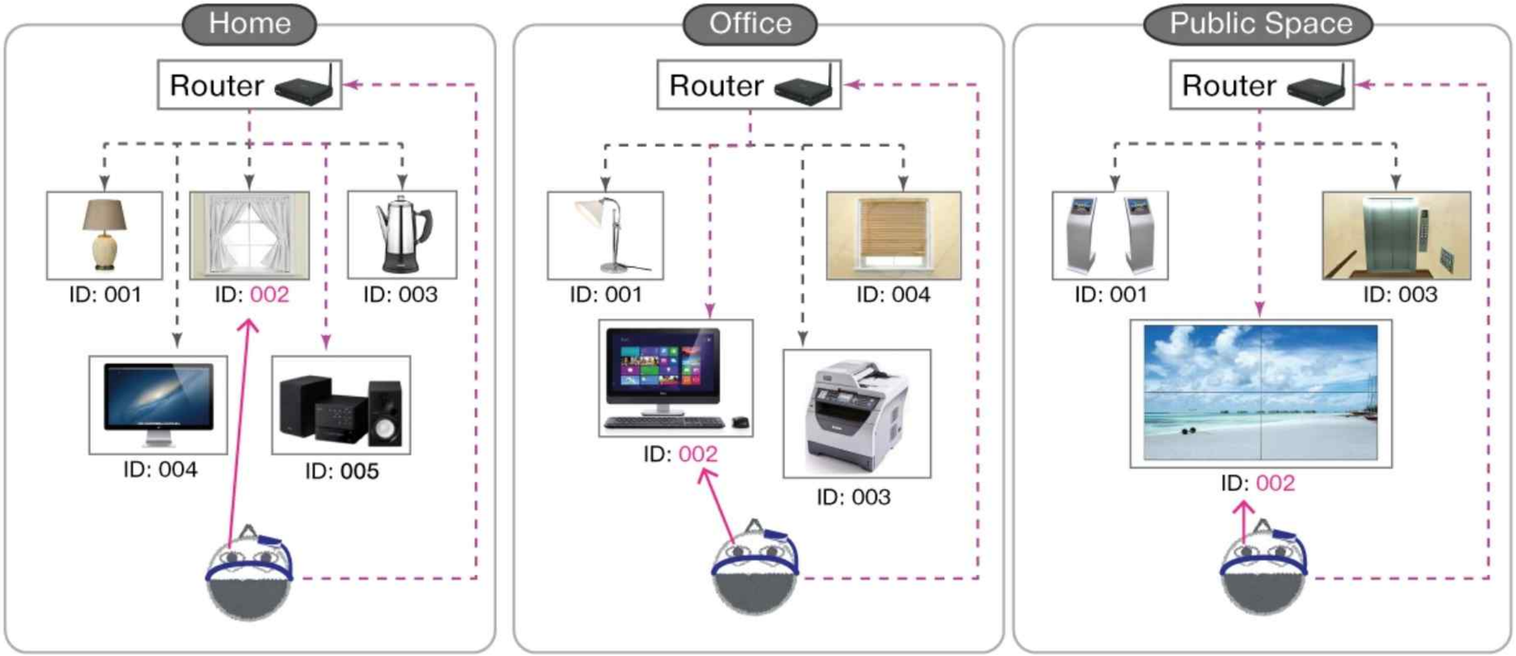


Fig.33 Designed command transfer system adaptive for IoT environment

3.6 User studies

Performance test and user interviews were conducted to verify the developed gaze interface.

3.6.1 Performance test

To verify the developed gaze interface, 'Number selecting' tests were conducted. The performance evaluation used the elapsed time and error counts to compare the results with widely used conventional interfaces like mouse and touch.

The absolute elapsed time and error counts for each multi device control were measured and compared. Since the developing interface cannot be compared to fully developed interfaces, the user needs more time to select the assigned number on multi devices than on a single device. If the developed gaze interface is appropriate for multi device control, the performance drops should be equivalent to other popular interfaces. Thus, two sets of tests, one for multi devices and the other for a single device, were conducted and the performance drops of each interface were compared.

The test set consists of four computers: one master and three slaves. The slave computers showed a 4x3 numerical keypad numbered from 1 to 12. The master computer showed the tasks participants had to complete like 'LEFT 5' or 'Middle 10'. The elapsed time and error counts are measured by the master computer. On the test, one set is the completion of 100 tasks that takes about 5 minutes without rest. Participants complete two sets: one for single device (tasks appeared only on the 'MIDDLE' computer) and another for multi devices (tasks appeared on 'LEFT', 'MIDDLE' and 'RIGHT'). Participants also complete the same sets of tasks using touch and mouse interface to compare performance. To avoid the learning effect, the interface order was randomly decided. In total, 9 males ranging in age from 26 to 34 years took part in the test with each participant practicing the developed gaze interface for about 3 min to become accustomed to it. The test program and process are shown in Fig.34 and example of accustomed user is shown Fig. 35.

The results with mean complete times and error counts for 100 tasks are shown in Fig.36 and 37 with error bars indicating 95% confidence intervals (CI) for the mean. By ANOVA analysis with 95% CI, p-values of all test interface combinations are $p < 0.05$, which means they are definitely different groups. The mean task complete times for single device control are 1.34 sec. for gaze interface (G), 1.25 sec. for touch interface (T), and 1.22 sec. for mouse interface (M). Thus, the mouse is the fastest while gaze is the slowest. The difference between the fastest and the slowest is 0.13 sec. (10.6% slower), so users can hardly feel the difference (discussed more in user interview). For multi device control, G is 1.61, T is 1.44, and M is 1.94; the touch is the fastest and mouse is the slowest. Using the mouse interface in multi devices, most participants faced difficulty in finding the mouse cursor's position when moving to another screen. The standard deviation of mouse control for multi devices was 0.65 sec., which is much longer than the other interfaces (G: 0.40 sec., T: 0.29 sec.). This is the reason for the sharp performance drops for multi device control with mouse. When analyzing results by the time difference of multi devices and single device, more time is needed at 0.27 sec. (20.5% performance drop) for G, 0.19 (15.2%) for T, and 0.73 (59.5%) for M. For multi devices control, the developed gaze interface shows a similar performance drop in the touch interface and is more suitable than the mouse. The error count results show that the mean error counts of each interface per 100 tasks were 3.67 for G, 0.56 for T, and 2.22 for M in a single device while it was 4.0 for G, 0.33 for T, and 1.89 for M in multi devices control. Although the absolute mean error counts for gaze interface is at least two times more than in other interfaces, the error increases only 0.33 times (8.9%) for multi devices with respect to single device.

The developed gaze interface is able to show appropriate performance in multi device control that is similar to touch and much better than mouse. The improved gaze interface performance for single device can lead to enhanced performance of multi devices control. Taking into account that users are much more familiar with a mouse and touch than with gaze interface, the gaze performance is expected to improve with increased familiarity. The fastest mean task completion time recorded by one developer was 1.19 sec. (27% faster than the average) with a standard deviation of 0.09 s.

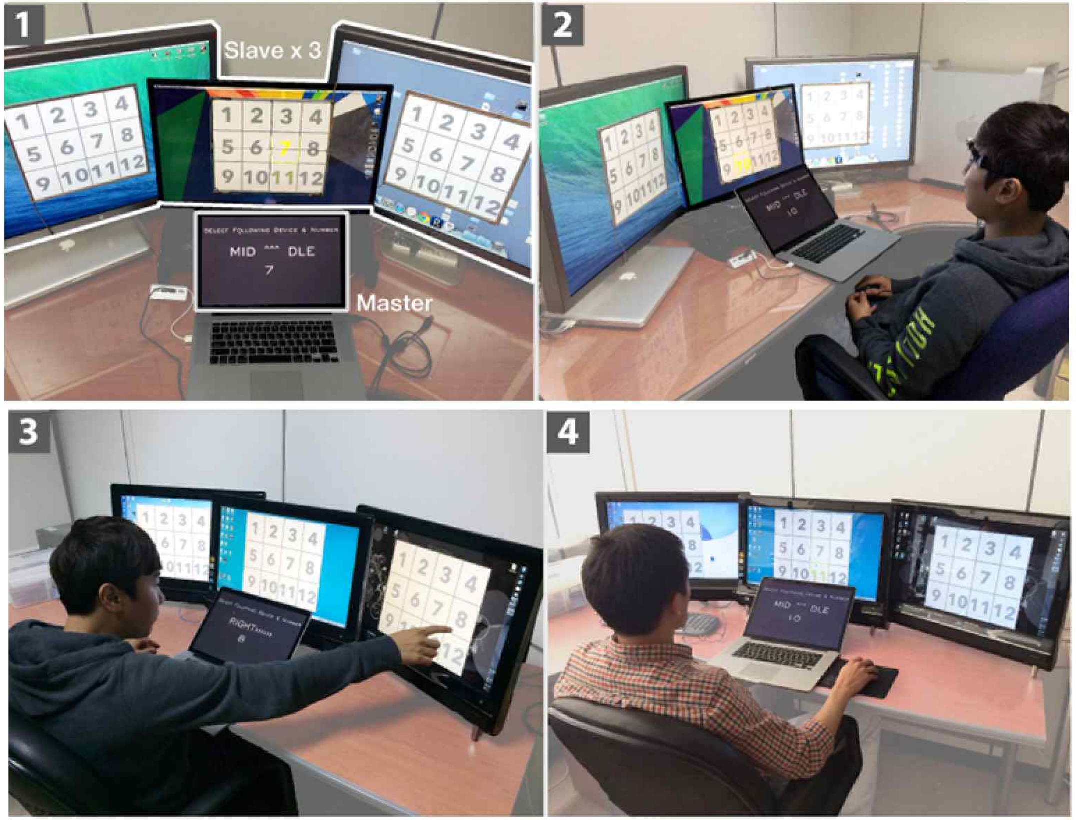


Fig.34 Performance tests: 1) test program, 2) gaze interface test, 3) touch test, 4) mouse test

Fig.35 Performance tests example of accustomed user

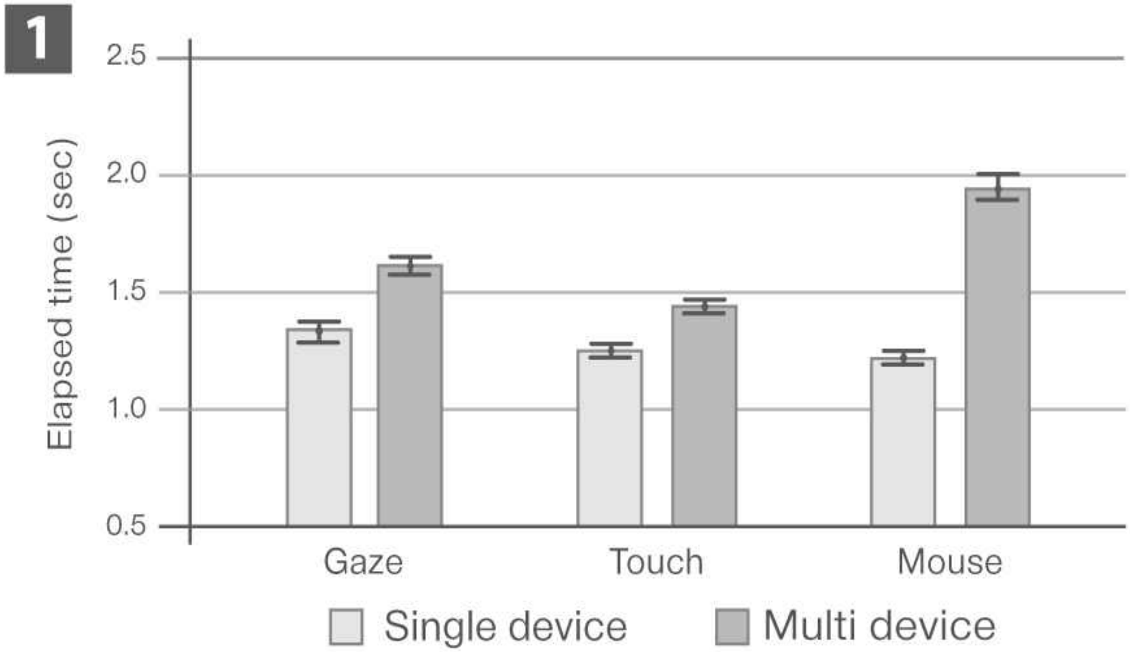


Fig.36 Performance test result graph, elapsed time

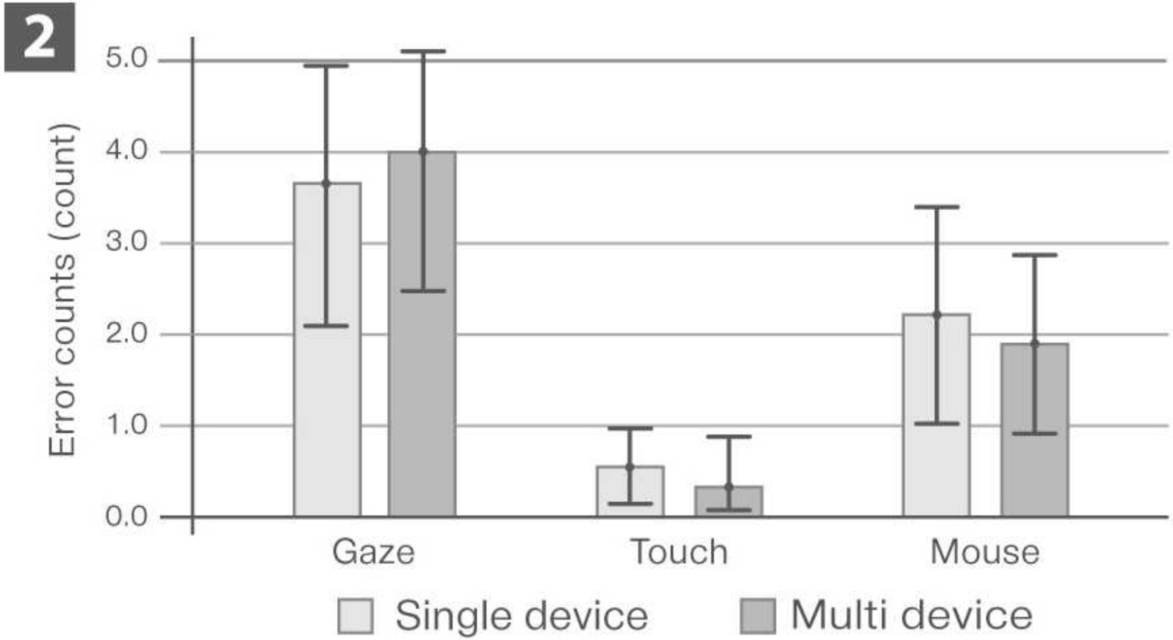


Fig.37 Performance test result graph, error counts

3.6.2 User interview

User experience factors for each interface in performance, usability, fatigue, intuitiveness, etc. were scored through user interviews. The participants who took part in the performance test scored on a scale of 1 (very bad) to 5 (very good). The results and ANOVA analysis are shown in Fig.38-1 and 38-2.

Users rated G as the best for speed, intuitiveness, fatigue, and overall satisfaction. They gave much higher scores in ‘Fatigue’ (Mean:4.55, StDev:0.72) than other interfaces T(Mean=1.88, StDev=1.26) and M(Mean=3.44, StDev=0.88). Users also evaluated G as very good (Mean:4.55, StDev:0.72) for ‘Intuitiveness,’ which is similar to T (Mean:4.5, StDev:0.52). Previous research proved that gaze has one of the most intuitive and low fatigue control method [22]. However, for multi device control, it was unnatural when switching to interested devices with the button or ID tag [11]. In the developed gaze interface, users merely look at the number of assigned devices and click to confirm without unnatural action. In particular, users felt that G had the highest rating in ‘Speed’(Mean=4.44, StDev=0.52), which is different from the results in the performance test. This means users could not feel about 0.2 sec. time difference. Bad ‘Fatigue’ affected the user’s sense of ‘Speed’. One of the participants stated, “The more tasks I attempted, the more I felt that time passed when completing an assigned task due to the pain in my arms”. Additionally, users felt G (Mean:4.77, StDev:0.44), like T (Mean:4.88, StDev:0.33), was easy to learn. Although users used the gaze interface for a relatively short 3 minutes, the gaze interface had almost equivalent speeds as T and M. This helps prove the 'Ease of learning' for the developed gaze interface. However, users felt G was less accurate (Mean:4.0, StDev:0.70) than other interfaces T and M, which scored over 4.33. The gaze interface accuracy can be improved in the future considering that the mouse and touch interface are well commercialized and popularized.

3.7 Result and application

We developed wearable gaze interface for multi devices. In Fig 39, developed gaze based interface and related system are described. Gaze interface for multi display control can be applied to various applications. In this paper, the application is developed for the automobile as an example. Drivers try to control several extra functions like music playing, volume adjustment and destination input to navigate, etc. These are behaviors that could possibly cause severe car accidents. By using wearable gaze interface, drivers can control these functions without taking hands off the steering wheel. The driver can just look at the desired function and push a button on the steering wheel to confirm selection. Fig.40 describes the automobile application, and a movie clip of the application can be seen at http://jbsong.com/c-e-e/.

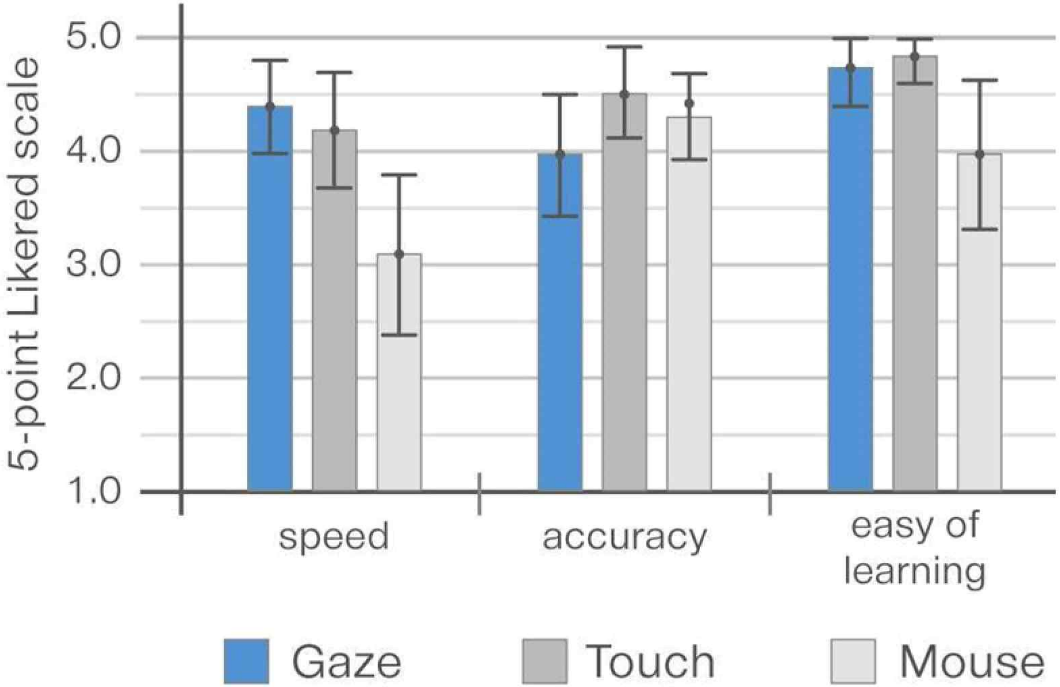


Fig.38-1 User interview results 1

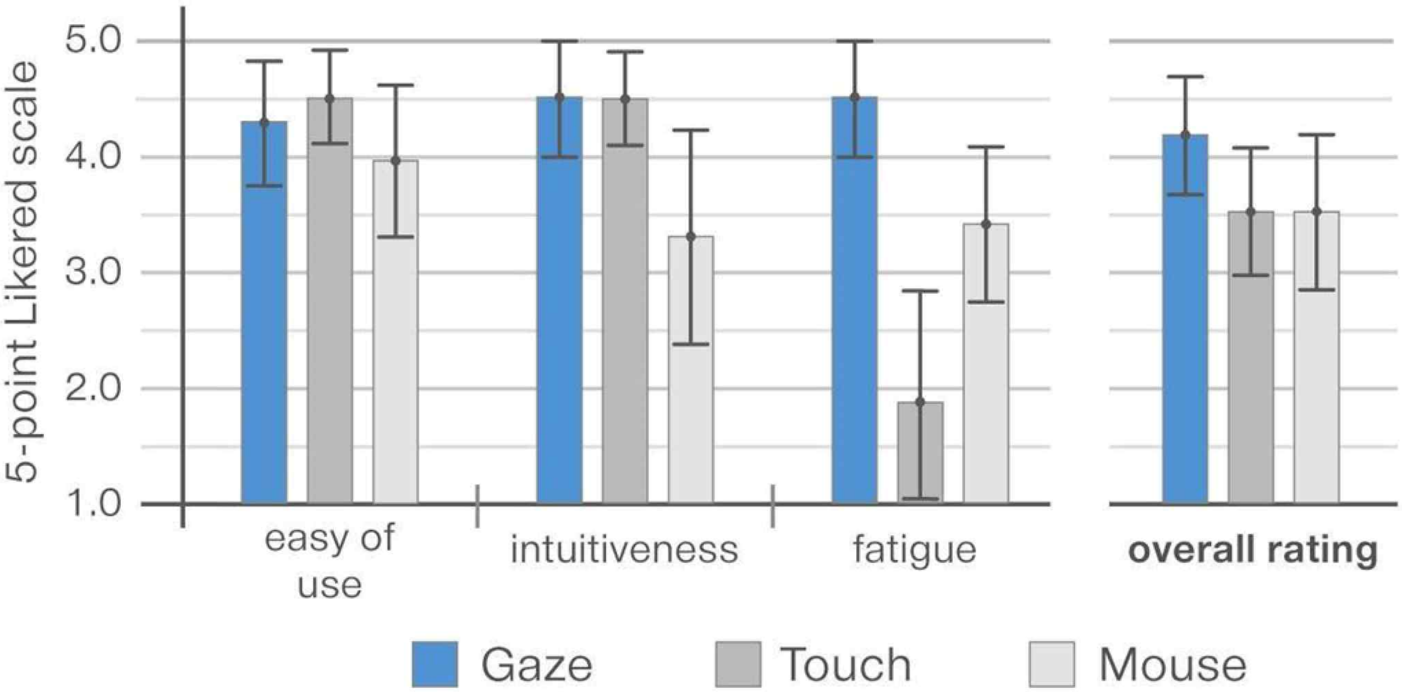


Fig.38-2 User interview results 2

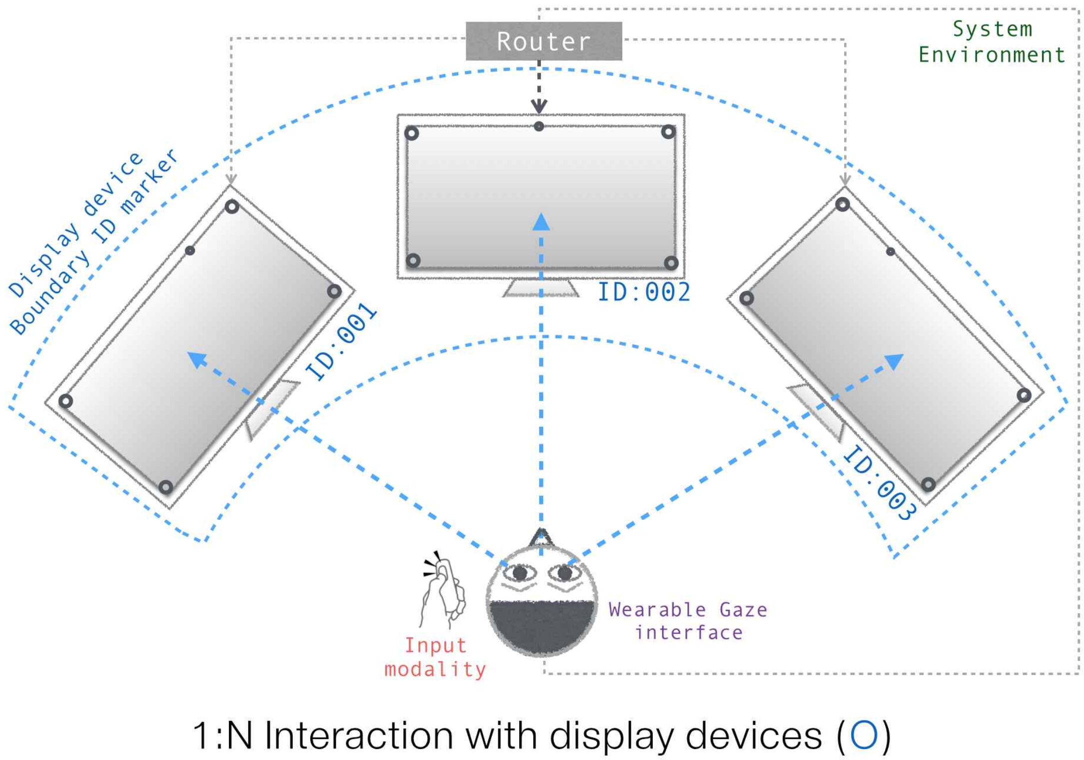


Fig.39 Implementation of gaze and multi devices interaction

Fig.40 Automobile application

Chapter 4

Implementation of gaze interface for non-display devices control

4.1 Introduction

We developed wearable multi modal gaze interface which can recognize the ID of multi display devices using IR LED markers. To complete our goal 'Command Everything with Eyes', we will expand researching focus to non-display device control using gaze (See Fig. 41).

As the shape and design of non-display devices are entirely completely various each other, the concept of attaching boundary+ID marker proper to rectangle shaped display device cannot be applied. So another concept of device awareness method should be studied for non-display devices.

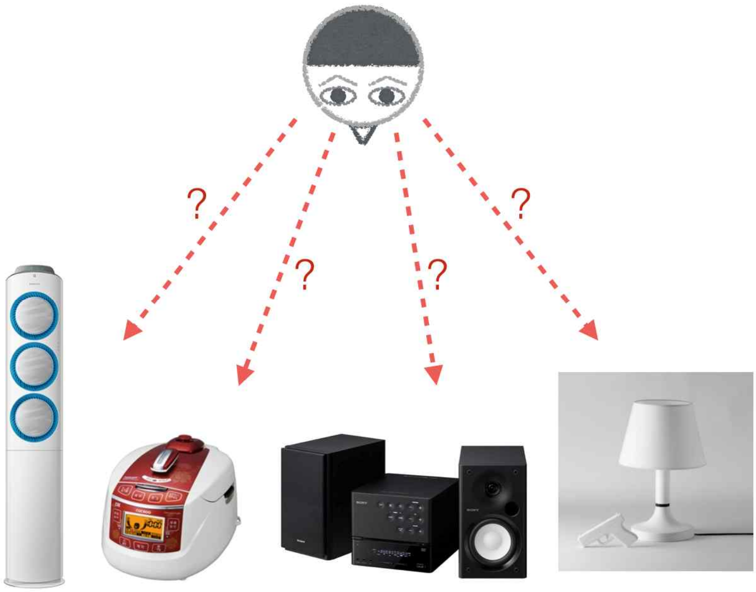


Fig.41 Gaze interface for non-display devices control

4.2 Previous works

As well as studies for multi display device control using gaze, researches for non-display device control using gaze is hardly found. Mardanbegi et.al [33] suggested gaze based wearable interface to control vacuum cleaner. He put the visual tag, QR code, on the top of the robot vacuum cleaner. By analyzing the image captured by the scene camera, he could recognize current position of the vacuum cleaner and steer the direction of the cleaner gazing the target position. In the same way, by attaching different image tag on the other non-display devices such as light and fan, he could choose device to control and turn it on and off with only his gaze. It was a meaningful research, but attaching unnecessary visual code on device was far from practical application.

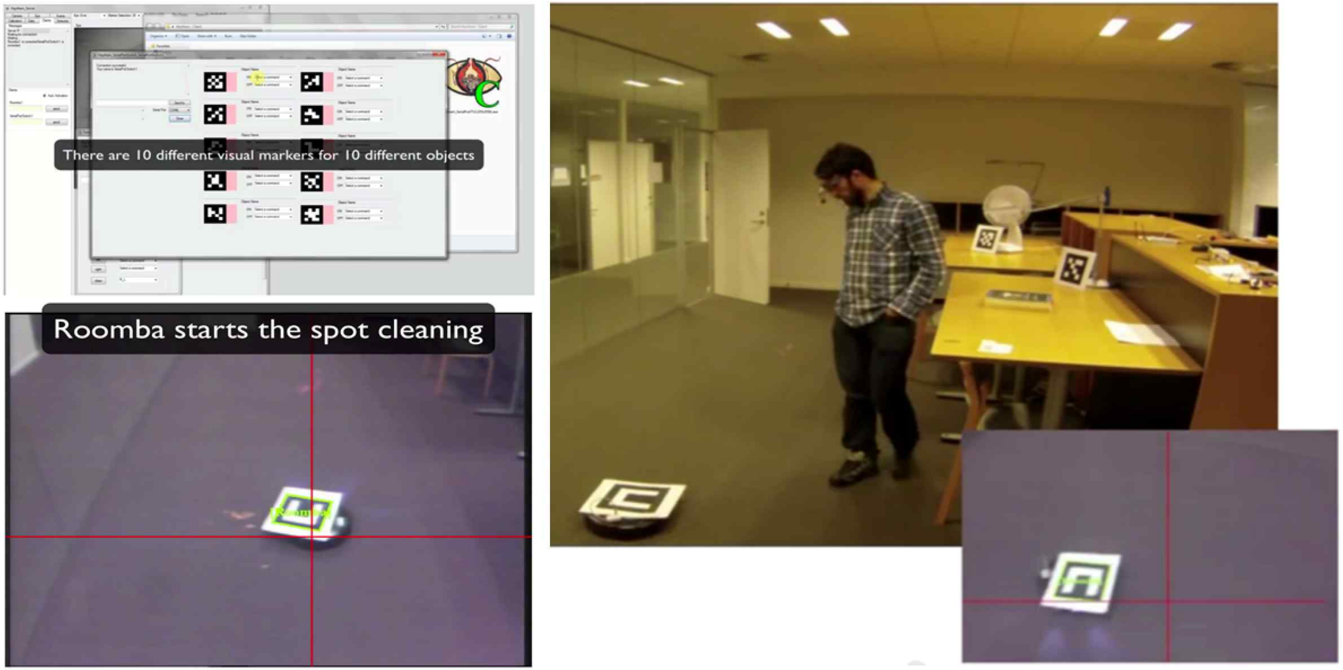


Fig.42 Previous works : visual code based non-display device control [33]

4.3 Theories

We referred to the nature and were motivated from the bat. Bat has bad eye sight, but they can avoid obstacles and find insects, using ultrasonic waves.

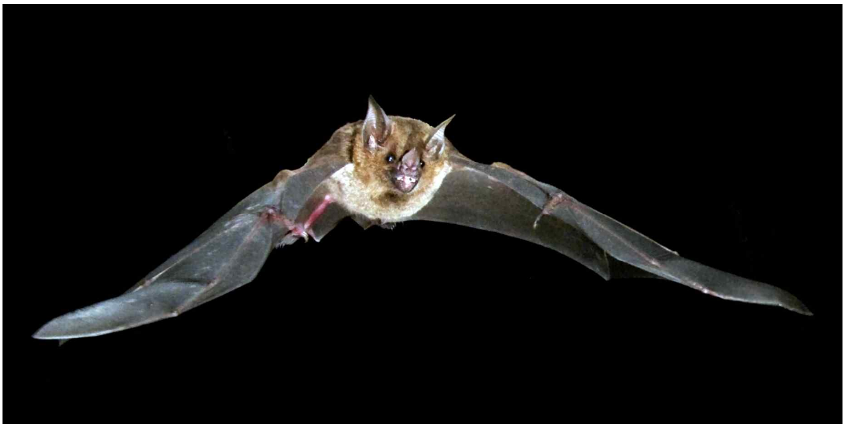


Fig.43 Motivation from nature: Bat, obstacle avoidance using sound

If we can recognize the sound made by device and track its directions, it is also possible to control the specific device. For related researches [28,29,30,31], John C. Murray et.al suggested sound source localization method (See Fig 44). If sound occurred at some point, its arrival time to two microphones apart is different unless sound source placed exactly on the intermediate between two microphones. As the speed of the sound is about 340m/sec, using normal sampling rate microphone, we can calculate the direction sound comes from. This algorithm is called as Time Delay Of Arrival (TDOA) method. Jean marc et.al [29,30] improved TDOA method to three dimension sound source localization using eight microphones and made it possible to recognize multi sound source by frequency analysis (Fig. 45).

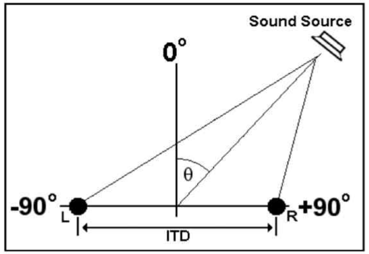


Figure 2. The angle calculated to determine the azimuth. Left of 0° i negative, right of 0° is positive.

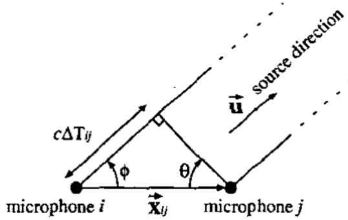
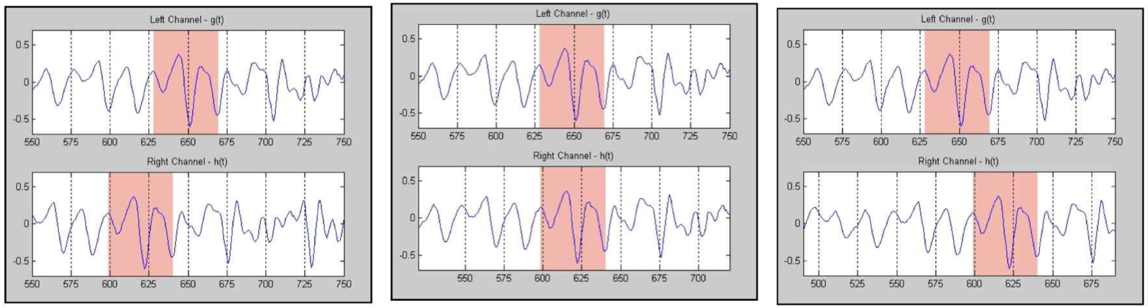
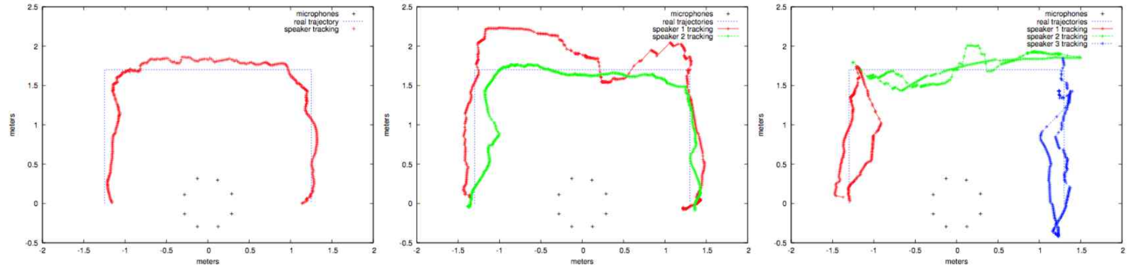


Fig. 5. Computing source direction from TDOA



$$\cos\phi = \frac{\vec{\mathbf{u}} \cdot \vec{\mathbf{x}}_{ij}}{\|\vec{\mathbf{u}}\| \, \|\vec{\mathbf{x}}_{ij}\|} = \frac{\vec{\mathbf{u}} \cdot \vec{\mathbf{x}}_{ij}}{\|\vec{\mathbf{x}}_{ij}\|}$$

Fig.44 Sound source localization algorithm [28]



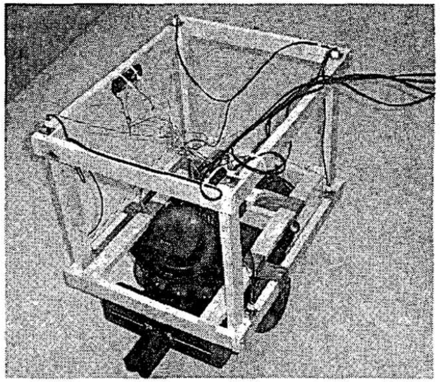


Fig. 3. Pioneer 2 robot with an array of eight microphones

TABLE I

DETECTION RATE AS A FUNCTION OF DISTANCE FOR DIFFERENT SOUNDS

Sound source	3 m	5 m	7 m
Hands clapping	92%	94%	84%
Speech ("test")	100%	90%	42%
Noise burst (250 ms)	100%	100%	100%

TABLE V. RESULTS OF TEST ANGLES

Test	Actual Angle	Calculated Azimuth	Accuracy %
Test 1	-90	-90	100
Test 2	-60	-55	94.44
Test 3	-20	-22	97.78
Test 4	0	2	97.78
Test 5	+30	+34	95.56
Test 6	+40	+45	94.44
Test 7	+70	+70	100
Test 8	+90	+90	100

Fig.45 Multi sound source localization in three dimension space [29,30,31]

4.4 Algorithm development

Motivated by obstacle avoidance of the bat and sound source localization method, we designed 'Gaze sound matching ID recognition' method. In this method, what user doing is just look at non-display device to control and makes command. Once command emerges, virtual ultrasonic wave is transferred through router and each non-display devices make their own frequency's sound. By sound localization algorithm and frequency analysis using several microphones on the wearable gaze tracker, user can find where each device's sound comes from. Finally, gaze tracker finds a sound in the same direction of the gaze and sends final commands. Fig.46 describes schematic diagram of the gaze sound ID recognition.

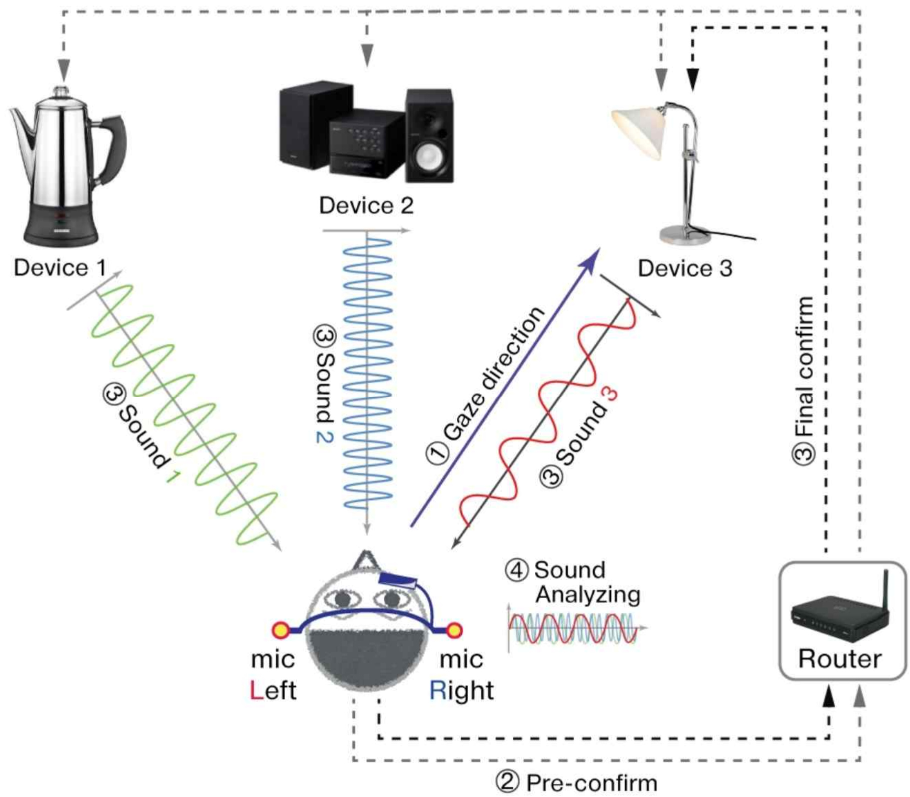


Fig.46 Schematic diagram of gaze sound ID recognition algorithm

4.5 Integration

For verifying the concept of gaze sound matching ID recognition method, microphone added wearable gaze tracker was developed. For gaze tracker, based system is same as that for multi display device control. Two microphones were installed on the left end and right end of gaze tracker and the distance between left and right microphone was about 300mm (Fig. 47). For sound data gathering, Arduino due is used [54]. Arduino due can gather sound data in maximum ADC speed 1MHz with 12 bit resolution. Gathered sound data transferred to PC for analysis by serial communication.

Fig.47 Microphone installed wearable gaze tracker

First, the feasibility test was performed whether current system could measure TODA of source sound. As described in Fig.48, TODA appeared clearly and the maximum arrival delay time was about 0.9 msec. To the separate directions of multi sound source, sound source localization and frequency analysis should be done concurrently. The sound direction tracing process consists of sound decomposition and sound localization is described in Fig.49.

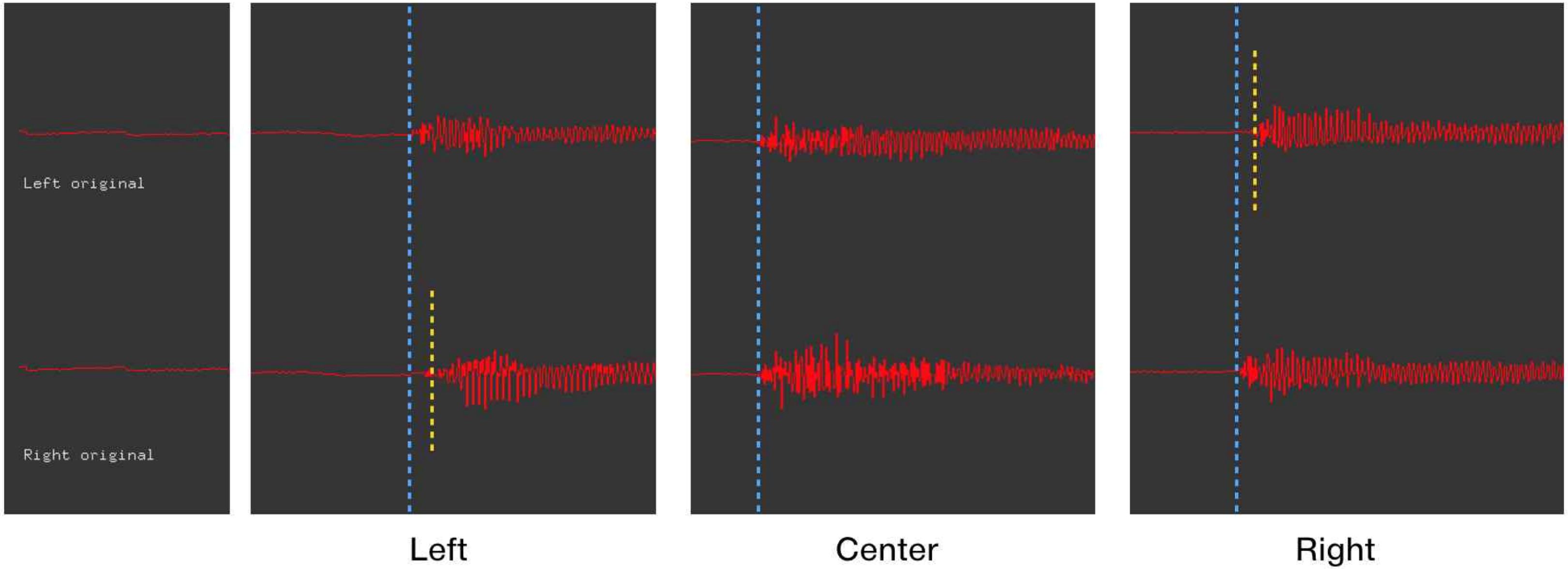


Fig.48 Feasibility test of TODA , when source sound comes Left, Center and right

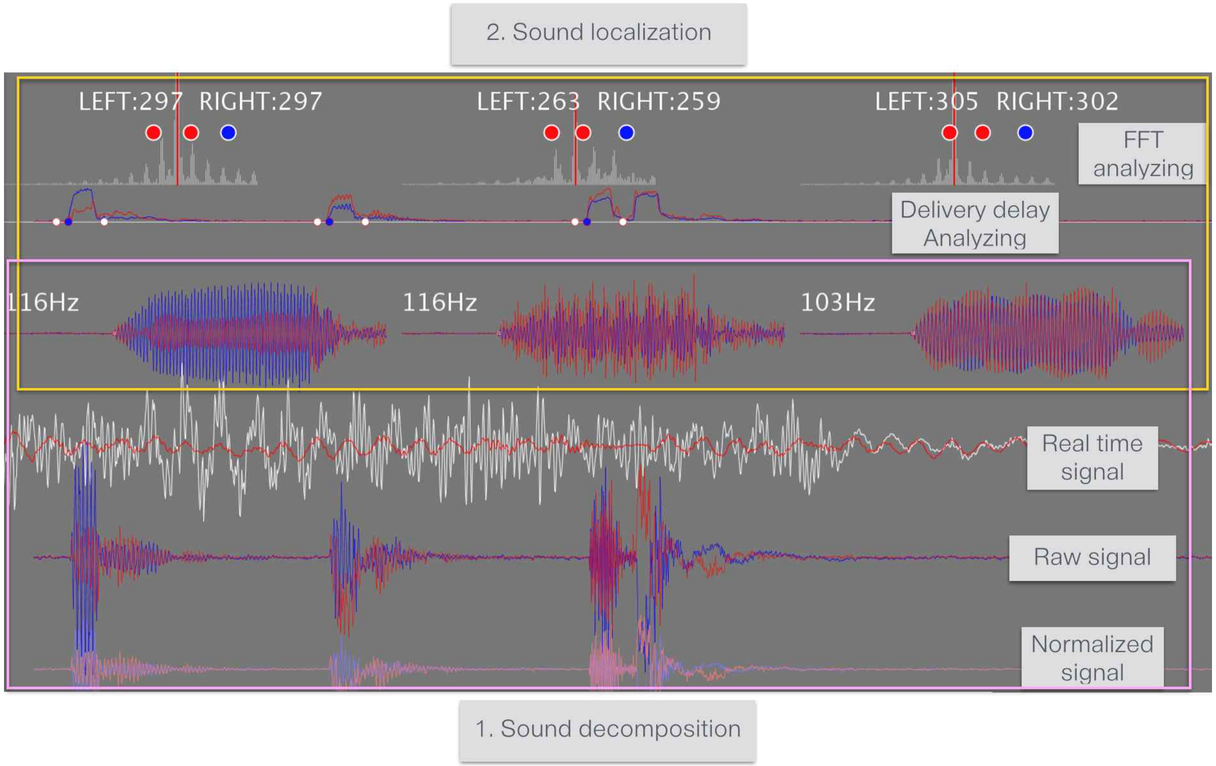


Fig.49 Developed sound source localization and frequency analysis process

4.6 Result and application

We developed wearable gaze interface for non-display device control using ‘gaze sound ID matching method. Gaze interface for non-display devices control can be applied to various applications. Fig.50 shows application controlling non-display devices, humidifier, light and audio. A movie clip of this application can be seen at ’http://jbsong.com/gaze-sound-id’.

Fig.50 Application of gaze sound matching ID recognition: control humidifier, light and audio

Chapter 5

Conclusions and future works

The purpose of this research is to command everything with eyes (C.E.E., Fig 51). To achieve this goal, we developed a multi-modal wearable gaze interface optimized for multi device control.

First, we developed high performance wearable gaze interface which can be fully customized for multi device control. A manual input modality for confirmation was added to improve accuracy and avoid Midas touch.

Second, an infrared boundary+ID markers and its novel recognition algorithm are suggested. By the help of this, we can switch to the interested device immediately only with 'just gaze'. Also developed gaze interface has high robustness due to the infrared. Through performance tests, we verified that the developed wearable gaze interface was appropriate for multi display devices control and found that its performance is similar to touch interface and better than mouse interface. In user interview, users rated the developed wearable gaze interface as 'Very good' for the user experience factors of 'Speed,' 'Ease of learning,' 'Intuitiveness,' 'Fatigue,' and 'Overall satisfaction'.

Finally, by developing gaze sound ID recognition method, we expand the usability of gaze interface to control non-display device.

For the future works, the infrared boundary+ID markers should be simpler so instead of adding extra binary ID markers, the blinking frequency or marker intensity can be used for identification [21]. The input modality should also be extended to various functions such as holding, dragging and scrolling in addition to enabling various behavior combinations [4]. Finally, There should be more feasibility test and applications for controlling non-display devices.

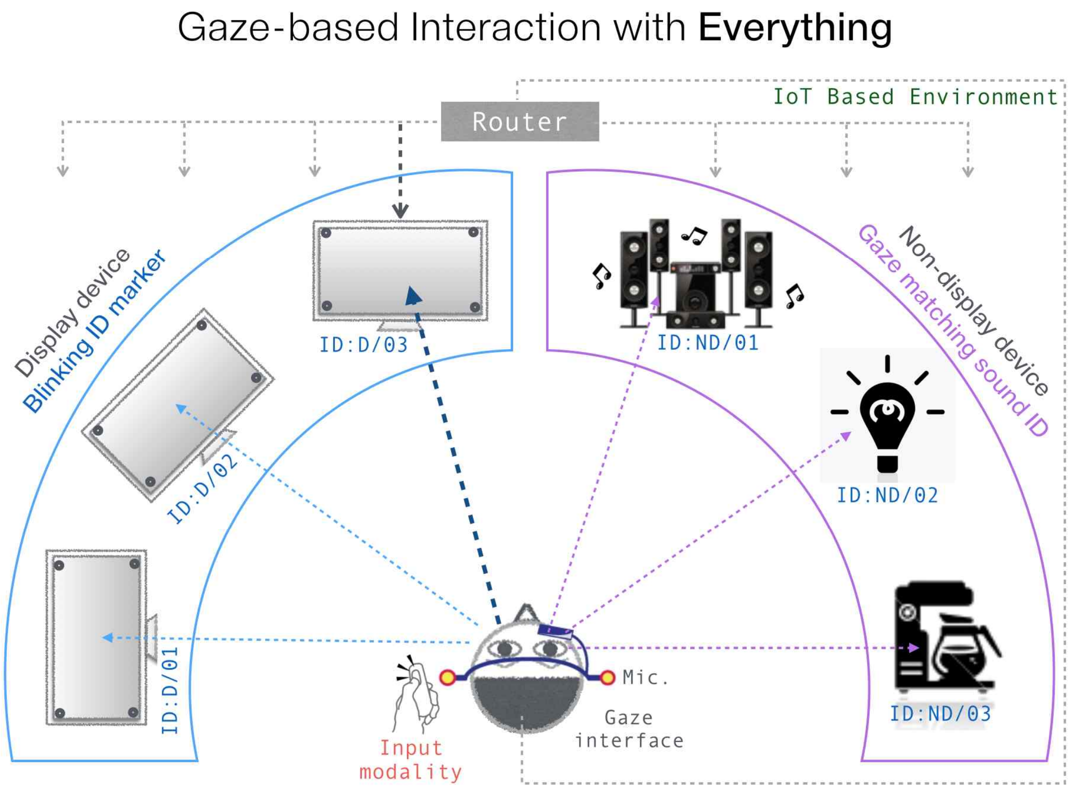


Fig.51 C.E.E : Control Everything with Eyes

References

[1] Bulling, Andreas, and Hans Gellersen. "Toward mobile eye-based human-computer interaction." Pervasive Computing, IEEE 9.4: 8-12, 2010.

[2] Song, Junbong, et al. "GaFinC: Gaze and Finger Control interface for 3D model manipulation in CAD application." Computer-Aided Design 46 (2014): 239-245.

[3] Ware, C., and Mikaelian, H. H. An evaluation of an eye tracker as a device for computer input. In Proc. of CHI'87, 183–188. ACM, 1987.

[4] Stellmach, Sophie, and Raimund Dachselt. "Still looking: Investigating seamless gaze-supported selection, positioning, and manipulation of distant targets." Proc. of the SIGCHI Conference on Human Factors in Computing Systems. ACM, 2013.

[5] Jacob, Robert JK. "What you look at is what you get: eye movement-based interaction techniques." Proc. of the SIGCHI conference on Human factors in computing systems. ACM, 1990.

[6] Mardanbegi, Diako, Dan Witzner Hansen, and Thomas Pederson. "Eye-based head gestures." Proc. of the symposium on eye tracking research and applications. ACM, 2012.

[7] Bieg, Hans-Joachim, et al. "Eye and pointer coordination in search and

selection tasks." Proc. of the 2010 Symposium on Eye-Tracking Research & Applications. ACM, 2010.

[8] Mardanbegi, Diako, and Dan Witzner Hansen. "Mobile gaze-based screen interaction in 3D environments." Proc. of the 1st conference on novel gaze-controlled applications. ACM, 2011.

[9] Zhai, Shumin, Carlos Morimoto, and Steven Ihde. "Manual and gaze input cascaded (MAGIC) pointing." Proc. of the SIGCHI conference on Human factors in computing systems. ACM, 1999.

[10] Bieg, H.-J., Chuang, L. L., Fleming, R. W., Reiterer, H., and Bu¨lthoff, H. H. Eye and pointer coordination in search and selection tasks. In Proc. of ETRA'10. 89–92, ACM, 2010.

[11] Hales, Jeremy, David Rozado, and Diako Mardanbegi. "Interacting with Objects in the Environment by Gaze and Hand Gestures."

[12] Stellmach, S., and Dachselt, R. Look & touch: gaze-supported target acquisition. In Proc. of CHI '12, ACM (2012), 2981–2990.

[13] Majaranta, Päivi, and Kari-Jouko Räihä. "Twenty years of eye typing: systems and design issues." Proc. of the 2002 symposium on Eye tracking research & applications. ACM, 2002.

[14] Google glass, http://www.google.com/glass

[15] Morimoto, Carlos Hitoshi, et al. "Pupil detection and tracking using multiple light sources." Image and vision computing 18.4: 331-335, 2000.

[16] Stellmach, Sophie, et al. "Designing gaze-supported multimodal

interactions for the exploration of large image collections." Proc. of the 1st Conference on Novel Gaze-Controlled Applications. ACM, 2011.

[17] Turner, Jayson, Andreas Bulling, and Hans Gellersen. "Combining gaze with manual interaction to extend physical reach." Proc. of the 1st international workshop on pervasive eye tracking & mobile eye-based interaction. ACM, 2011.

[18] Witt JHD. https://github.com/jhdewitt/eyetracker.

[19] Turner, Jayson, Andreas Bulling, and Hans Gellersen. "A Dual Scene Camera Eye Tracker for Interaction with Public and Hand-held Displays." 10th International Conference on Pervasive Computing (Pervasive 2012). 2012.

[20] Kato, Hirokazu, and Mark Billinghurst. "Marker tracking and hmd calibration for a video-based augmented reality conferencing system." Augmented Reality, 1999.(IWAR'99) Proc. 2nd IEEE and ACM International Workshop on. IEEE, 1999.

[21] Ouzts, Andrew D., et al. "On the conspicuity of 3-d fiducial markers in 2-d projected environments." Proc. of the Symposium on Eye Tracking Research and Applications. ACM, 2012.

[22] Sibert, Linda E., and Robert JK Jacob. "Evaluation of eye gaze interaction." Proc. of the SIGCHI conference on Human factors in computing systems. ACM, 2000.

[23] Lee, Hyeon Chang, et al. "Remote Gaze Tracking System on a Large

Display, Sensors. 2013.

[24] Turner, Jayson, Andreas Bulling, and Hans Gellersen. "Extending the visual field of a head-mounted eye tracker for pervasive eye-based interaction." Proc. of the Symposium on Eye Tracking Research and Applications. ACM, 2012.

[25] Mardanbegi, Diako, and Dan Witzner Hansen. "Parallax error in the monocular head-mounted eye trackers." Proc. of the 2012 acm conference on ubiquitous computing. ACM, 2012.

[26] Jacob, Robert JK. "Eye movement-based human-computer interaction techniques: Toward non-command interfaces." Advances in human-computer interaction 151-190. 1993

[27] Isokoski, Poika. "Text input methods for eye trackers using off-screen targets." Proc. of the 2000 symposium on Eye tracking research & applications. ACM, 2000.

[28] Murray, John C., Harry Erwin, and Stefan Wermter. "Robotics sound-source localization and tracking using interaural time difference and cross-correlation." Proceedings of NeuroBotics Workshop. 2004.

[29] Valin, Jean-Marc, et al. "Localization of simultaneous moving sound sources for mobile robot using a frequency-domain steered beamformer approach." Robotics and Automation, Proceedings. ICRA'04. 2004 IEEE International Conference on. Vol. 1, 2004.

[30] Valin, Jean-Marc, et al. "Robust sound source localization using a

microphone array on a mobile robot." Intelligent Robots and Systems (IROS 2003), Proceedings. 2003 IEEE/RSJ International Conference on. Vol. 2, 2003.

[31] Watkins, William A., and William E. Schevill. "Sound source location by arrival-times on a non-rigid three-dimensional hydrophone array." Deep Sea Research and Oceanographic Abstracts. Vol. 19. No. 10. Elsevier, 1972.

[32] Asoh, Hideki, et al. "An application of a particle filter to bayesian multiple sound source tracking with audio and video information fusion." Proc. Fusion. 2004.

[33] Gaze-based environment control,
https://www.youtube.com/watch?v=qgEsSEnd6M8

[34] Nishino, Ko, and Shree K. Nayar. "Eyes for relighting." ACM Transactions on Graphics (TOG). Vol. 23. No. 3. ACM, 2004.

[35] Hansen, Dan Witzner, et al. "Eye typing using markov and active appearance models." Applications of Computer Vision, 2002.(WACV 2002). Proceedings. Sixth IEEE Workshop on. IEEE, 2002.

[36] Valenti, Roberto, and Theo Gevers. "Accurate eye center location through invariant isocentric patterns." Pattern Analysis and Machine Intelligence, IEEE Transactions on 34.9 (2012): 1785-1798.

[37] Nakazawa, Atsushi, and Christian Nitschke. "Point of gaze estimation through corneal surface reflection in an active illumination environment." Computer Vision–ECCV 2012. Springer Berlin Heidelberg, 2012. 159-172.

[38] Raspberrypi, http://www.raspberrypi.org/

[39] Huentek, mobile camera manufacturer, http://huentek.en.ec21.com/

[40] Robox 3D printer, http://www.cel-robox.com/

[41] Valenti, Roberto, and Theo Gevers. "Accurate eye center location and tracking using isophote curvature." Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on. IEEE, 2008.

[42] Hansen, Dan Witzner, and John Paulin Hansen. "Robustifying eye interaction." Computer Vision and Pattern Recognition Workshop, 2006. CVPRW'06. Conference on. IEEE, 2006.

[43] Zhu, Zhiwei, Kikuo Fujimura, and Qiang Ji. "Real-time eye detection and tracking under various light conditions." Proceedings of the 2002 symposium on Eye tracking research & applications. ACM, 2002.

[44] Turner, Jayson. "Cross-device eye-based interaction." Proceedings of the adjunct publication of the 26th annual ACM symposium on User interface software and technology. ACM, 2013.

[45] haytham eye tracking software, http://eyeinfo.itu.dk/index.php/projects/low-cost-gaze-tracking

[46] Solidworks, http://www.solidworks.com/.

[47] Ji, Qiang, and Xiaojie Yang. "Real-time eye, gaze, and face pose tracking for monitoring driver vigilance." Real-Time Imaging 8.5 (2002): 357-377.

[48] Eye writer project, http://www.eyewriter.org/

[49] Tobii EyeX commecial eye tracker, http://www.tobii.com/en/eye-experience/eyex/

[50] EyeTribe commecial eye tracker, https://theeyetribe.com/

[51] Kühn, Michael, and Jorn Garbe. "Predictive and highly ambiguous typing for a severely speech and motion impaired user." HCI. 2001.

[52] Intel's Edison chip, http://www.intel.com/content/www/us/en/do-it-yourself/edison.html

[53] Ebisawa, Yoshinobu. "Realtime 3D position detection of human pupil." Virtual Environments, Human-Computer Interfaces and Measurement Systems, 2004.(VECIMS). 2004 IEEE Symposium on. IEEE, 2004.

[54] Arduino, http://www.arduino.cc/

[55] Gartner's 2014 Hype Cycle for Emerging Technologies Maps the Journey to Digital Business, http://www.gartner.com/newsroom/id/2819918

[56] Ohno, Takehiko, Naoki Mukawa, and Atsushi Yoshikawa. "FreeGaze: a gaze tracking system for everyday gaze interaction." Proceedings of the 2002 symposium on Eye tracking research & applications. ACM, 2002.

[57] Hansen, Dan Witzner, and Qiang Ji. "In the eye of the beholder: A survey of models for eyes and gaze." Pattern Analysis and Machine Intelligence, IEEE Transactions on 32.3 (2010): 478-500.

[58] Li, Dongheng, David Winfield, and Derrick J. Parkhurst. "Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches." Computer Vision and Pattern Recognition-Workshops, 2005. CVPR Workshops. IEEE Computer Society Conference on. IEEE, 2005.

[59] Yuille, Alan L., Peter W. Hallinan, and David S. Cohen. "Feature extraction from faces using deformable templates." International journal of computer vision 8.2 (1992): 99-111.

[60] Daugman, John. "How iris recognition works." Circuits and Systems for Video Technology, IEEE Transactions on 14.1 (2004): 21-30.

[61] Hornof, Anthony, Anna Cavender, and Rob Hoselton. "Eyedraw: a system for drawing pictures with eye movements." ACM SIGACCESS Accessibility and Computing. No. 77-78. ACM, 2004.

초록

시선 추적 기반 입력 장치의 앞으로의 높은 발전 가능성에도 불구하고, 기존의 연구들은, 단일 디스플레이 장치와의 상호 작용 상황에서의 성능을 높이는 방향에 집중되어 있는데 반해, 다수의 장치와 상호 작용에는 관련 연구가 거의 이루어지지 않고 있다. 시선 추적 기반 입력 장치를 일상 생활에 사용될 수 있는 대중적인 입력 장치로 적용하기 위해서, 본 논문에서는 다수의 장치들과 상호작용에 최적화 된 착용 형 다중 입력 시선 기반 입력 장치 및 관련 시스템을 제안한다.이를 위해 첫 번째로, 자유롭게 개조, 개선이 가능한 착용 형 고글 형태의 시선 추적 기반 입력장치를 개발하였다. 특히 개발된 시선 추적 장치는 시선을 포인팅의 용도에 한정하고, 결정의 기능은 독립적인 입력 수단을 부여함으로써, Midas touch 문제와 정확도 저하 문제를 해결하였다. 두 번째로, 다수의 디스플레이 장치와 상호작용을 위해서, 적외선 LED 로 구현된 외곽선 및 ID 설계하였고, 이를 통해 높은 다양한 조명 환경에서도 견실함을 유지하면서, 즉각적인 관심 장치의 변경을 가능하도록 구현하였다. 이와 같이 개발된 시선 추적 기반 입력 장치의 성능을 검증하기 위하여, 다수의 디스플레이 디바이스에서 발생하는 일련의 조작 업무를 부여하고, 이 과정을 기존의 마우스와 터치 입력 장치로 함께 수행하여 그 성능을 비교하였다 그 결과 개발된 시선 추적 기반 입력 장치는 마우스보다 우월하고, 터치와 유사한 수준의 성능을 나타내었다. 또한 사용자 경험을 인터뷰 한 결과에서도, 피로도 및 직관성을 비롯한 다수의 분야에서 가장 높은 점수를 기록하였다. 마지막으로, 시선 추적 입력 장치와 비 디스플레이 장치와의 상호 작용을 위해 소리 기반 ID 인식 방법을 제안하였고, 검증 시험을 수행하여, 시선 추적 입력 장치의 상호 작용 범위를 확장하였다.

주요어: 시선 추적, 착용형, 다중 입력 장치, 사용자 입력 장치

학번: 2012-30177