# Eye-based head gestures

Diako Mardanbegi
IT University of Copenhagen
dima@itu.dk

Dan Witzner Hansen
IT University of Copenhagen
Witzner@itu.dk

Thomas Pederson
IT University of Copenhagen
tped@itu.dk

## Abstract

A novel method for video-based head gesture recognition using eye information by an eye tracker has been proposed. The method uses a combination of gaze and eye movement to infer head gestures. Compared to other gesture-based methods a major advantage of the method is that the user keeps the gaze on the interaction object while interacting. This method has been implemented on a head-mounted eye tracker for detecting a set of predefined head gestures. The accuracy of the gesture classifier is evaluated and verified for gaze-based interaction in applications intended for both large public displays and small mobile phone screens. The user study shows that the method detects a set of defined gestures reliably.

**CR Categories:** H.5.2 [Information interfaces and presentation]: User Interfaces —Input devices and strategies, Interaction styles

**Keywords:** Head Gestures, Gaze interaction, Eye tracker, Interaction

## 1 Introduction

Gaze-based interaction has so far been restricted to interaction with computer screens using remote eye trackers. Gaze-based applications are still waiting to be investigated with improved principles that can even be used for gaze-based interaction in 3D environments as well as with virtual objects on screen. This paper proposes a novel method for enhancing gaze-based interaction through both voluntary head movements and vestibulo-ocular reflexes. Contrary to previous research this information is obtained only through eye and gaze information using an eye tracker. The method is shown to be useful for both gaze-based screen interaction and 3D environmental control.

Gaze interaction has been shown to be useful for many applications but eye information has been shown to be limited for interaction. The point of regard only posses information about position and does not provide sufficient information to make selections (a.k.a Midas touch). Extra information is needed to make convey other pieces of information such as clicks. Dwell-time selection, eye blinks and gaze-gestures [Jacob 1993; Isokoski 2000] have been typical ways of extending the capabilities of eye trackers with methods for communicating with interfaces (e.g. making selections on a screen).

Gestures are commonly used for interaction and are used to signify a particular command or intent. For eyes there are two types of gestures, namely eye and gaze gestures. Eye gestures such as wink and blinks make use of movements of the eyelid and eyebrows. However, interaction with eye gestures and blinking especially repetitive blinking for long-term use may create a feeling of nervous eye muscles [Drewes 2010]. Gaze gestures, on the other hand, are definable patterns of eye movements performed within a limited time interval [Istance et al. 2010]. Simple gaze gestures are not distinguishable from natural eye patterns and make unintended interaction similar to the Midas-touch problem. Complex gaze gestures consist of several simple gaze gestures are therefore needed for robust results. Such use may be considered unnatural as a perceptual channel is used for motor control [Zhai et al. 1999]. Besides, it may be physically straining and requires the user to memorize combinations of gaze gestures. This increases the cognitive load while forcing the eyes to be used actively, and therefore takes the focus away from the actual interaction task. In terms of interaction gaze gestures are facing severe limitations, for example, gaze gestures are not intuitively applicable for user interaction on e.g. Icons or objects in 3D space since the point of regard possibly leaves the object while interacting thus may confuse the user as well as significantly complicates the algorithmic design.

Head nods and shakes are widely used in our daily conversation as a gesture to fulfill a semantic function and as conversational feedback (e.g., nodding instead of saying yes) [Darwin 1872; Morris 1994]. People are more used to making deliberate movements of the head compared with similar patterns of eye movements. Basic head gestures such as nod and shake are relatively easy to measure from full-face images and have been also used for interaction with user interfaces [Toyama 1998; Kjeldsen 2001]. Methods for video-based head gesture recognition deal with three main problems: First localizing and identifying the face region in the image (which may have a cluttered background) using a fixed camera located in front of the head and works only when the face is in the field of view of the camera. The second problem is to extract the feature set that represents the head movements. And then classifying the feature set into a number of head gestures. These methods are not able to separate the head gestures from the natural head movements and most of them are only limited to detect some specific gestures like head nods and shakes. On the other hand, real time detection of head nods and shakes is difficult, as the head movements during a nod or shake are small, fast and jerky.

This paper suggests using head gestures measured by the gaze trackers, as a convenient way of interaction when using the gaze trackers. Eye image alone is not sufficient for detecting the head gestures, but by in combination with gaze information it is possible to measure the head gestures. This paper describes a novel approach for detecting head movements using only eye images and the point of regard. Having the point of regard allows for distinguish between the visual eye movements (eye-movements that are associated with vision) and the non-visual eye-movements that are associated with vestibulo-ocular reflex (VOR) and are caused by the head movements. This work is meant for gaze-based interaction and is related to gaze gestures in the sense that eye movements are used to signal gestures. However, the user does not move the eyes voluntarily, but eye movements are an effect of vestibulo-ocular reflexes when the user fixates on the interaction object and does head gestures.

This paper shows that it is possible to detect a relatively large amount of both large and small head gestures, using gaze trackers thus minimizing the need to make very complex gestures. The main advantage of this method is that attention remains fixed on the object of interaction while executing gestures.

The rest of the paper is organized as follows. Section 2 presents related work, and section 3 presents an overview of the method. The head gestures are introduced in section 4, and the algorithm used for recognizing the gestures are described in section 5. Section 6 describes the experimental applications in which the method is tested for interaction. Section 7 presents the experimental results and section 8 concludes the paper with future work.

## *2* Previous work

A comprehensive review on gaze gestures is given in [Møllenbach 2010]. Research on gaze gestures was initiated by Isokoski for text input using off-screen targets. The eye gaze has to visit the off-screen targets in a certain order to select characters. Off-screen targets force the gesture to be performed in a fixed location and with a fixed size [Isokoski 2000]. Drewes and Schmidt [2007] made a comprehensive research on gaze gestures and presented some scalable gaze gestures which could be performed in any location on screen, and used them for interacting with computers and devices with smaller displays. Wobbrock et al. proposed a similar idea to gaze entry of letters using Edge-Write gestures when the user could map out letters by combining the four corners of a square in various ways [Wobbrock et al. 2007]. The idea of using the gaze gestures for text input was continued later [Porta and Turina 2008; Bee and Andre 2008].

Many video-based methods for head gesture recognition have been proposed. Some attempts have been made to use eye information (e.g., eye location) for head gesture recognition. Davis and Vaks presented a prototype perceptual user interface for a responsive dialog-box agent. They used IBM PupilCam technology for only detecting the eye location in the image and used together with anthropometric head and face measurements to detect the location of the user's face. Salient facial features are then identified and tracked between frames to compute the global 2-D motion direction of the head. A Finite State Machine incorporating the natural timings of the computed head motions was employed for recognition of head gestures (nod=yes, shake=no) [Davis and Vaks 2001]. Kapoor and Picard introduced an infrared camera synchronized with infrared LEDs to detect the position of the pupils, and used it as the feature. A HMM based pattern analyzer was used to detect the nods and the shakes [Kapoor and Picard 2002]. Recognition of head gestures had been demonstrated by tracking eye position over time. They presented a real-time nod/shake head gesture detector. However, their system used complex hardware and software and had problems with people wearing glasses and with earrings. Nonaka [2003] used Eye-mark recorder and FASTRAK motion tracking system to track the eye movements and head movements respectively and proposed a communication interface working by eye-gaze and head gesture. Nonaka tried to use the eye tracker for detecting the fixed point of regard during the head gestures. Beside the complex hardware of the system, FASTRAK head motion tracker only worked in the range of its magnetic transmitter (max 3 meter). Fixation of eye gaze and also the gestures of "Shaking Head", "Nodding Head", and "Inclining Head" (assigned to "no", "yes" and "undo" respectively) are detected using successive dynamic programming (S-DP) matching method with their reference patterns. However this system was not always able to identify even these three gestures correctly.

## 3 VOR-based detection of head movements

Eye movements can be caused by the head movements while PoR is fixed (*fixed-gaze eye movements*) or by changing the PoR when the head is fixed (*fixed-head eye movements*). This paper investigates the fixed-gaze eye movements. When the point of regard is fixed and the head moves, the eyes move in opposite direction and with the same speed as the head movement. The eye movements are due to the vestibulo-ocular reflexes (VOR), which are used to stabilize the image on the retina. Figure 1 illustrates a user looking at an object but in two different situations, one when head is up and the other when head is down. The eye image is different in each posture even though the PoR is fixed.

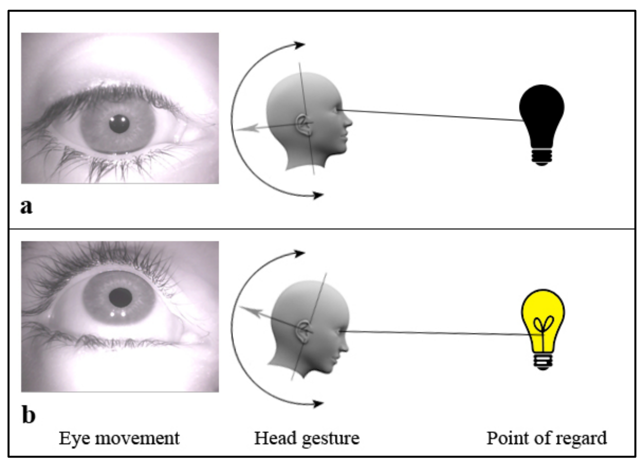



**Figure 1** *Eye image when POR is fixed and (a) head is up or (b) down*

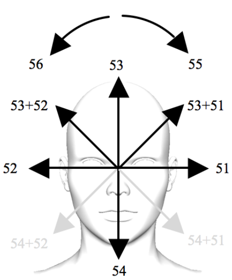
The eye trackers are able to distinguish between *fixed-gaze eye movements* and *fixed-head eye movements* since they measure both eye movements and estimate the point of regard. The term *eye-based head gestures* will in the following denote a predefined pattern of head movements measured through eye movements but where the PoR is fixed on a given object.

This paper focuses on measuring head gestures from a head mounted eye tracker. Head mounted eye trackers move with the head movements and there is therefore no information about the world reference frame in the eye image. Eye movements caused by VOR or by changing the gaze direction cannot be determined unless additional information is available. Head-mounted eye trackers have a scene camera that captures the user's field of view through which the PoR is determined. So, by the ability of recognizing a known reference point in the scene image, fixed-gaze eye movements can be recognized through the point of regard and the reference point when the user fixes the gaze and moves the head.

# 4 Head Gestures

This section describes head gestures, their relation to eye movements and how these can be measured in an eye tracker.

Ekman and Friesen [1978] developed a common standard to systematically categorize and encode human facial expressions. There are 44 action units (AU) that account for change in facial expressions and orientations. 8 action units correspond to head orientation (shown in figure 2). Some movements such as diagonal downwards movements (AU54+52 and AU54+51 in figure 2 (left) are uncomfortable to perform and are usually made in conjunction with head tilts (AU55-56). These head movements would not be suitable for interaction and are therefore disregarded in this paper.

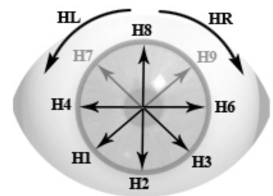



| AU Number | Movement |
|---|---|
| 51 | Head turn left |
| 52 | Head turn right |
| 53 | Head up |
| 54 | Head down |
| 55 | Head tilt left |
| 56 | Head tilt right |

**Figure 2** *The basic movements of head gestures and their AU number*

When a user keeps the gaze on a specific point in space, the vestibulo-ocular reflex makes it possible to measure head movements through eye movements, but where eye and head movements are in opposite directions. Consequently, head movements are measurable indirectly by eye trackers, even in close-up images.

The basic eye movements associated with a given head movement (when the PoR is fixed), is shown in figure 3. The VOR has both rotational (AVOR) and translational (TVOR) aspects [Panerai1998]. When the head tilts (AU55-56), AVOR can be seen as the iris rotates around LoS axis. These movements are termed as rotational eye movements. For the other head movements (AU51-54), we see a translation of the pupil center in the eye image, which is termed as linear eye movements. In the following, HL and HR denote left and right rotational eye movements and H1-H9 denote the linear eye movements




| Eye movement | Head movement |
|---|---|
| H1 | 53+51 |
| H2 | 53 |
| H3 | 53+52 |
| H4 | 51 |
| H6 | 52 |
| H8 | 54 |
| HR | 56 |
| HL | 55 |

**Figure 3** *Basic reflexive movements of the iris/pupil and the corresponding head movements*

Measuring sequences of eye movement are usually influenced by noise. We define a *character,* $C_i$, as a sequence of N eye movements where the majority of movements are the same e.g. $C_i = H_i^1 .. H_i^N$, (defined in figure 3). *Head gestures* are either *discrete* or *continuous.* A *discrete head gesture*, $G = C_1 .. C_T$, consists of a repeatable and recognizable sequence of characters, $C_i$. Discrete gestures and characters can conceptually be related to words and letters, when writing text. *Simple gestures*, $Gij = C_i C_j$ are 2 character words and *continuous gesture*, $G_H$, are sequences of eye movements $H$ along an axis.

There are in total (8*8) 64 simple gestures but only 14 of these are considered here since executing and distinguishing gestures that are orthogonal or neighboring is hard. A simple gesture is denoted *sweep gesture* when the characters in the gesture $Gij = C_i C_j$ are different ($i \neq j$) and is denoted *repetitive* when the characters are identical ($i = j$). In this paper repetitive gestures consist of two linear movements separated by a short break, $C_B$. Figure 6 shows examples of sweep gestures (top row) and a repetitive gesture (bottom row). Gestures are in this paper well described by regular expressions and thus by a finite state machine.

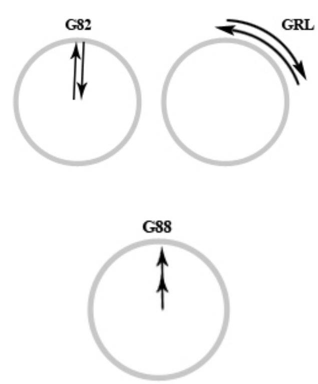



**Figure 4** *Examples of sweep gestures (top row) and a repetitive gesture (bottom row). The arrows indicate eye movement. The actual head movement is in the opposite direction.*

An example of continuous gestures is shown in Figure 5 where the gesture is used to continuously change the value e.g. the volume of a loudspeaker.

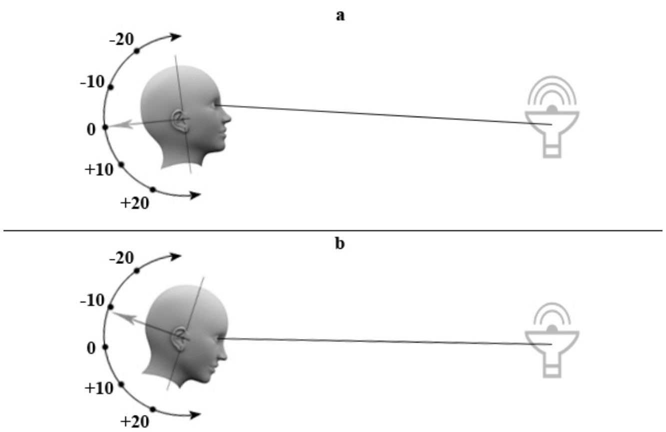



**Figure 5** *A continuous gesture moving the head downwards while the eyes move upwards.*

# 5 Gesture recognition

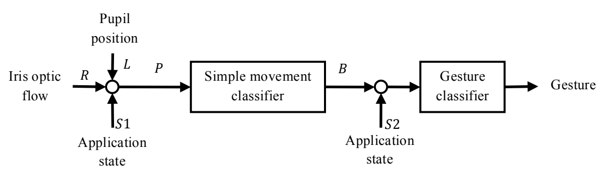



**Figure 6** *Overview of the gesture recognition method.*

The method consists of a classifier to detect the basic eye movements and a gesture classifier based on regular expressions and is shown in figure 6. The length of the sequences of head movements defining a character and the dictionary of gestures are found experimentally.

Basic eye movements $H_i^t$ at time t are estimated through feature vectors $\{P_1, \ldots, P_t\}$ measured from the images. Each feature vector is for clarity separated into 3 subsectors $P_i = [L, R, S_1]$, where $L = [l_1, l_2]$ are features needed for detecting the linear movements, $R = [r_1, .. r_8]$ are features needed to estimate rotational movements and $S_1$ is the current application state. The pupil center ($l_1$), and its velocity ($l_2$) between frames define the feature vector $L$. Feature vectors $r_1, .. r_k$ are sampled in regions $A_1, .. A_k$, where $r_i$ is the mean optic flow quantized into 8 directions. The regions $A_i$ and the corresponding feature vector $r_i$ are shown in figure 7. The location of each patch $A_i$ is defined relative to the pupil diameter to ensure the regions $A_i$ are stabilized within the normalized region between iris and pupil.

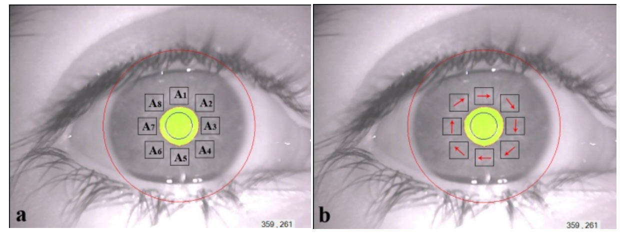



**Figure 7** *Measurements of rotational head movements with (left) the regions $A_i$ and the corresponding feature (right) measured during a rotation of the head.*

# 6 Experimental applications

The proposed method has been implemented for use with a head mounted eye tracker. Two experimental applications have been developed.

*iRecipe,* is an application to read and follow recipes when the hands are occupied or in a state that is not recommended for touching the computer. The second application is called *iiPhone* which is an iPhone emulator running on the screen that can be controlled by head gestures to show the potential of the proposed method for the mobile devices.

For both applications, the screen contour is detected and tracked within the scene image of the eye tracker. The eye tracker provides only gaze estimates $s = (x_s, y_s)$ in the scene image, but we need to determine where on the screen the user is looking. A homography [Harley and Zisserman 2000] from the screen corners $S_i$ to $M_i$ (figure 8) is estimated in each time instance. The gaze point in the scene image coordinates is then mapped to the screen coordinates through $(x_m, y_m) = H_s^m \cdot \mathbf{s}$. Figure 8 shows the mapping of the PoR (center of the red cross-hair) from the scene image to the screen plane and the real coordinates of the PoR in the screen by a black cross-hair (left image) [Mardanbegi and Hansen 2011].

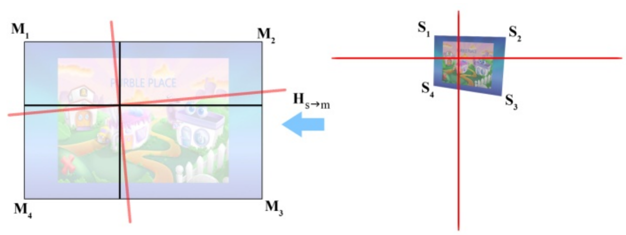



**Figure 8** *Mapping from the scene plane (right) to the real screen plane (left)*

### 6.1 iRecipe application

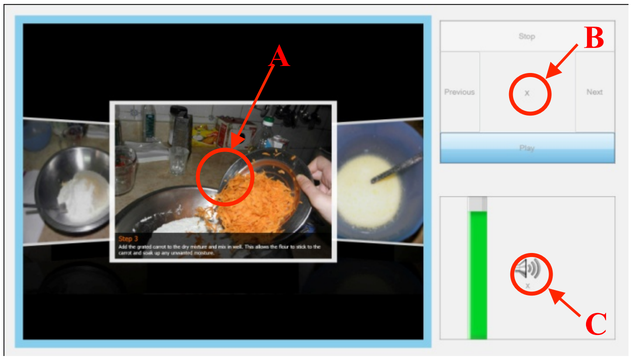



**Figure 9** *iRecipe interface with slides frame in the left side, music player at the top right corner and the volume frame at the below right corner. A, B and C are predefined regions in each frame that user should look at them while doing the gestures.*

The iRecipe application is intended for a hands-free interaction with a recipe when cooking. The user interface of iRecipe is shown in figure 9. The interface consists of three areas: the recipe slides frame, a simple music player and the volume frame. The interface is operated by looking at predefined regions (A, B or C) while doing the gestures. Each gesture is interpreted differently based on the gazed object. Therefore the same gestures might have different meanings depending on the PoR.

Four different sweep gestures including Up, Down, Left and Right ($G46, G64, G28, G82$) together with the continuous vertical head movements were used for controlling the application as below:

I. Changing the slides by looking at the region “A” and doing the right or left head gestures.

II. Changing the music files by Left/Right gestures and stopping and playing by Up or Down gestures when looking at the center of the player (B)

III. Changing the volume had 3 steps. First enabling the volume by looking at the icon(C) in the volume window for 1 second (dwell-time activation). The color of the icon will be changed when the volume is active, and then user can change the volume by vertical head movements (as it is shown in figure 8). Then the volume can be disabled simply by looking at another part of the screen or by closing the eye. Changing the volume will be indicated by showing a number (0-100) below the icon during the vertical movements and adjusting the volume.

There is no cursor shown on the screen but the user has visual feedback on the interface during the interaction as the regions become highlighted when the PoR is inside that.

### 6.2 iiPhone application

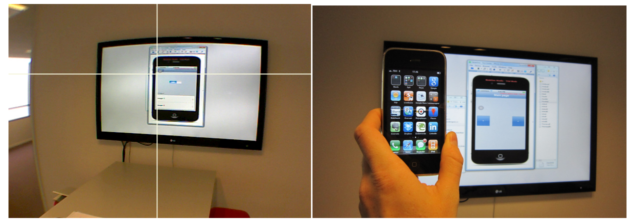


**Figure 10** *Interaction with iphone emulator. Left image shows the scene image with the gaze point (white cross), and the right image shows the visual appearance of a real iphone and the emulator in the user's field of view are about the same.*

The second application was to interact with an iphone emulator using the head gestures. The application had 4 different pages with different buttons and list-boxes. User is able to press or select an item by looking at the item and performing the corresponding gesture that was showing on the items. (e.g., left gesture for the back button).

## 7 Experimental results

A classifier test has been done before the applications. In this section the results of the classifier test and the performance of the users during the iRecipe and iiPhone applications will be presented.

The classifier test has been conducted for testing the accuracy of the implemented algorithm on a head mounted eye tracker. Simple gestures introduced in section 6, were tested in the classifier test, however we restrict the repetitive gestures ($Gii$) to only the linear movements ($i \neq RorL$). 14 simple gestures were shown on the screen by a simple figure, two times one by one and randomly. The shown gesture remains on the screen until the user performs the same gesture or pressing a key in the case when the user was not able to perform that gesture.

8 participants (6 male and 2 female, mean=35.6, SD=9.7) are used in the experiments. 7 participants were unfamiliar with this method. The method and gestures were introduced to participants and they had the chance of practicing the gestures for 10 minutes before the experiments. The experiments on each participant lasted about 50 minutes.

In all the experiments, a 55” LG flat panel screen was used as a display. The users wearing a head mounted eye tracker were able to move around the screen during the task and at the same time interact with the screen.

The head mounted eye tracker with the accuracy of about 1° made by the authors was employed in the experiments. The eye tracker consists of two webcams, and both eye and scene images with the resolution of 640x480 are processed at 25 frames per second in real time. A feature-based method has been used for pupil detection, and a homography mapping has been use for gaze estimation.

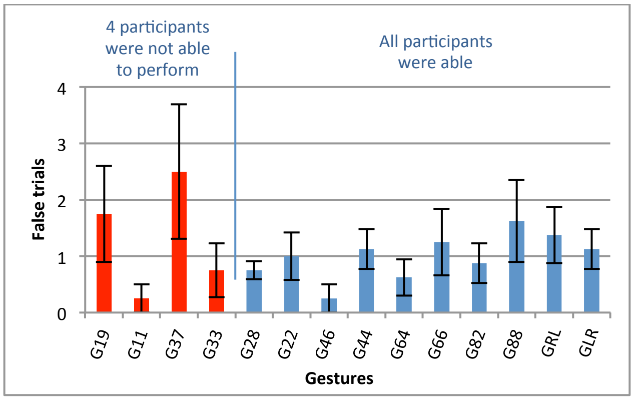



**Figure 11** *Average number of false trials per each gesture. Error bars show the standard error of the mean.*

Figure 11 shows the results of the classifier test and the average number of false trials of all participants for each gesture. Each gesture has been shown 16 times (2 times per participant). Each time that a participant performs a gesture but it is not recognized correctly, it will considered as a false trial. Ideally the number of the false trials should be 0, it means that the participant only performs the gesture one time after displaying the gesture on the screen which is detected correctly by the classifier. 4 participants were not able to perform the diagonal gestures ($G19, G11, G37, G33$), and these gestures are shown at the left side of the graph indicated by the red color.

The results show that the diagonal gestures were difficult for some of the participants. Among the other linear gestures which were more convenient for the participants, sweep down gesture ($G28$) had a more average of false trials. It means that it was not easy for participants to turn the head down. Repetitive down gesture ($G88$) is even more inconvenient and has the highest number of false trials in the right side of the graph, and it is because of the user needs to divide the down movement into two steps.

In the classifier test, it was also observed that even smallest movements of the head (<2°) can be detected by the system which is not possible to detect by the other methods introduced in the previous work.

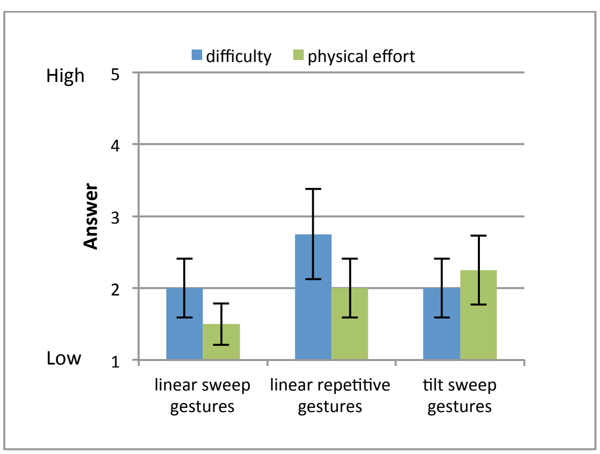



**Figure 12** *Results of the questionnaire, showing the physical effort and the level of difficulty for three types of gestures*

After the test, the participants were given a questionnaire consists of questions with the range of the answers from 1 to 5 to investigate the participants experience in terms of physical effort and the level of difficulty. The participant answered each question for three different gestures. Figure 12 shows the results of the questionnaire.

In the classifier test, the target was only a marker on the screen and sometimes the participants were moving the gaze point together with the head movements meaning that the gesture was not correct. However, it has been observed that it is easier for the users to keep the gaze point fixed during the head movements, when the object is something that they want to control it by a gesture. For example, during the iRecipe and iiPhone applications, participants were trying to press the buttons or controlling some items on the screen, and the average number of false trials of the simple sweep gestures, was less than classifier test. The false detection may also occur when the classifier is not able to detect the head gestures, meaning that the classification method requires some improvements. In addition, some of the defined gestures were difficult to perform and the users need more practice in order to be able to look at an object and do the movements.

All the participants were able to control iRecipe and iiPhone applications and do the tasks by head gestures. Each application took approximately 10 min. Some of the participant found the volume control more convenient than controlling the other parts of the recipe application. This was because of the real time feedback of the interface, both by showing the volume gain number at the gazed object and by hearing the changes of the volume. It shows that the small visual changes in the gazed object (e.g., changing the color) during the head movements can help the user to keep focus on the object during the gesture. However, any change in the appearance of the gazed area that leads to loos the visual attention from the object should not be done during the gesture. Using the sound as feedback, can also be a good choice in some cases. For example for controlling the objects in the 3D environment, when the visual feedback is not possible, sound feedback can be used during the gesture whenever the system detects a basic head movement, or before the gesture just to show that the gazed object is ready for control.

The accuracy of the eye tracker allowed the user to interact comfortably with 4×2 regions on the mobile display. The size of the emulated display shown on the 55" screen was about the same visual angle as real iphone display (figure 10). 5 of the participants were already using the iPhone and it was so easy for them to interact with the emulator by the head gestures.

Even thought the used gestures in the applications were simple (double characters), no unintended command was observed during the tasks.

# 8 Discussion & Future work

A novel method for detecting the head gestures in combination with gaze was suggested and tested on a mobile eye tracker. The proposed method shows that head gestures can be measured through eye image based on vestibulo-ocular reflex and by having the gaze point. Many video based methods have been used so far for detecting the head gestures. In contrast, the presented method in this paper, allows for identifying a wide range of head gestures even the small gestures accurately and in real time, by only using an eye tracker.

Head gestures together with fixed gaze point can be used as a method for gaze based interaction by eye trackers instead of complex gaze gestures. It can be used when the user is able to slightly move the head. The main advantage of this method with compare to the gaze gestures is that the user does not lose the visual attention on the object during the interaction.

This method has been implemented on a head mounted eye tracker for detecting a set of 14 simple gestures and the algorithm was evaluated. The method was also tested on two applications one to show the capability of the method for interacting with a screen at kitchen during cooking and when the hand is occupied. The other application was to interact with an emulated iPhone. The results showed the possibility of this method for interaction with the screens and even small displays like the mobile devices.

*Future work*

We have already shown that the presented method can be used for interaction with screens. This method has also a high potential to be a direct way of communication and controlling the objects (looking at the objects and doing a simple gesture). As a future work, we are trying to use this method and the developed head mounted eye tracker for interaction and controlling the objects in the home environment.

The proposed method allows for very simple and intuitive way of interaction that can be used either with head mounted gaze trackers or remote gaze trackers. However, how to determine whether the gaze is fixed during the head gestures differs for remote and head-mounted eye trackers. We are trying to implement this method on a remote eye tracker, since the mobile devices are predicted to embed increasingly capable eye gaze tracking technology. Eye-based head gestures can be used alone or together with finger gestures for operating the interfaces of tablets and mobile phones. Most of the remote video-based eye trackers use the Pupil-Corneal Reflection (P-CR)" to determine the point of regard (PoR) [Hansen and JI 2010]. The light sources are always in the field of view of the eye and the reflections in the eye image provide a reference at the world reference frame. It makes the detection of fixed gaze easier in the remote eye trackers with compare to the head mounted eye trackers.

When for some reason, the hands cannot be used, (e.g. due to the object being too far away; the hands are already occupied with other things; the hands cannot be adequately controlled due to disease or impairment) or even when the hands are free, proposed method can be used as a fast way of interaction with objects. Besides, in some applications that loosing the visual attention may increase the human risk (e.g., driving the vehicles, driving the wheelchair or in the high risk environments like the power plants control rooms), eye-based head gestures can be used for interaction without requiring the users to look away from their usual viewpoints. It can also be a way to interact with head-up displays in the automobile or aircrafts.

## 9 References


Bee, N. and Andre, E. 2008. Writing with Your Eye: A Dwell Time Free Writing System Adapted to the Nature of Human Eye Gaze. *In Perception in Multimodal Dialogue Systems, LNCS 5078/2008*, Springer, pages 111 – 122.

Darwin, C. 1872/1998. *The Expression of the Emotions in Man and Animals*, third edition. New York, Oxford University Press.

Davis, J.W., and Vaks, S. 2001. A Perceptual User Interface for Recognizing Head Gesture Acknowledgements. *In Proceedings Workshop on Perceptive User Interfaces.*

Drewes, H. 2010.Eye Gaze Tracking for Human Computer Interaction. PhD thesis, Faculty of Mathematics, Computer Science and Statistics, LMU München.

Drewes, H., and Schmidt, A. 2007. Interacting with the Computer using Gaze Gestures. *In Proceedings of Human-Computer Interaction - INTERACT 2007*, Springer LNCS 4663, pages 475 – 488.

Ekman, P., and Friesen, W., 1978. *Facial Action Coding System: A Technique for the Measurement of Facial Movement*. Consulting Psychologists Press, Palo Alto.

Gale A.G., 2005. Attention Responsive Technology and Ergonomics. *In Bust P.D. & McCabe P.T. (Eds.) Contemporary Ergonomics, Proceedings of the Ergonomics Society Annual Conference*, Hatfield University, Hertfordshire, pages 273-276.

Hansen, D.W., and Ji, Q., 2010. In the Eye of the Beholder: A Survey of Models for Eyes and Gaze. IEEE Trans. Pattern Anal. Mach. Intell, pages 478-500.

Hartley, R., and Zisserman, A. 2000. *Multiple view geometry in computer vision*. Cambridge University Press, Cambridge, UK.

Isokoski, P. 2000. Text Input Methods for Eye Trackers Using Off-Screen Targets. *In Proceedings of the ACM symposium on Eye tracking research & applications ETRA '00*. ACM Press, pages 15 – 21.

Istance, H., Hyrskykari, A., Immonen, L., Mansikkamaa, S., and Vickers, S.2010. Designing Gaze Gestures for Gaming: an Investigation of Performance. *In Proceedings of the ACM symposium on Eye tracking research & applications ETRA '10,* ACM Press, New York, NY March.

Kapoor, A., and Picard, R.W. 2002. A real-time head nod and shake detector. Technical Report 544, MIT Media Laboratory Affective Computing Group.

Kjeldsen, R. 2001. Head gestures for computer control. In *Proceedings Second International Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real time Systems*, pages 62–67.

Mardanbegi, D., Hansen, D.W. 2011. Mobile gaze-based screen interaction in 3D environments, Proceedings of the 1st Conference on Novel Gaze-Controlled Applications (NGCA2011), Blekinge Institute of Technology, Karlskrona, Sweden.

Møllenbach, E. 2010. Selection strategies in gaze interaction. PhD thesis, Innovative communication group, IT University of Copenhagen.

Morris, D. 1994. *Body talk: The Meaning of Human Gestures*. Crown Publishers, New York.

Nonaka, H. 2003. Communication interface with eye-gaze and head gesture using successive DP matching and fuzzy inference.

Journal of Intelligent Information Systems 21(2): pages 105–112.

Panerai, F. and Sandini, G. 1998. Oculo-Motor Stabilization Reflexes: Integration of Inertial and Visual Information. Neural Networks, 11, pages 1191–1204.

Porta, M., and Turina, M. 2008. Eye-S: a Full-Screen Input Modality for Pure Eye-based Communication. *In Proceedings of the ACM symposium on Eye tracking research & applications ETRA '08*. ACM Press, pages 27 – 34.

Shi, F., Gale, A.G. & Purdy, K.J. 2006.Eye-centric ICT control. *In Bust P.D. & McCabe P.T. (Eds.) Contemporary Ergonomics, Proceedings of the Ergonomics Society Annual Conference*, pages 215-218.

Toyama, K. 1998. Look, ma–no hands! Hands free cursor control with real-time 3D face tracking. *In Proceedings Workshop on Perceptual User Interfaces (PUI'98)*, pages 49–54.

Wobbrock, J. O., Rubinstein, J., Sawyer, M., and Duchowski, A. T. 2007. Not Typing but Writing: Eye-based Text Entry Using Letter-like Gestures. *In Proceedings of COGAIN '07*, pages 61 – 64.

Zhai, S., Morimoto, C., and Ihde, S. 1999. Manual And Gaze Input Cascaded (MAGIC) Pointing. *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems. CHI '99*, ACM Press, pages 246 – 253.