Eye-based Interaction Using Embedded Optical Sensors on an Eyewear Device for Facial Expression Recognition

Katsutoshi Masai
masai@imlab.ics.keio.ac.jp
Keio University
Yokohama, Kanagawa, Japan

Kai Kunze
kai@kmd.keio.ac.jp
Keio Media Design
Yokohama, Kanagawa, Japan

Maki Sugimoto
sugimoto@imlab.ics.keio.ac.jp
Keio University
Yokohama, Kanagawa, Japan

ABSTRACT

Non-verbal information is essential to understand intentions and emotions and to facilitate social interaction between humans and between humans and computers. One reliable source of such information is the eyes. We investigated the eye-based interaction (recognizing eye gestures or eye movements) using an eyewear device for facial expression recognition. The device incorporates 16 low-cost optical sensors. The system allows hands-free interaction in many situations. Using the device, we evaluated three eye-based interactions. First, we evaluated the accuracy of detecting the gestures with nine participants. The average accuracy of detecting seven different eye gestures is 89.1% with user-dependent training. We used dynamic time warping (DTW) for gesture recognition. Second, we evaluated the accuracy of eye gaze position estimation with five users holding a neutral face. The system showed potential to track the approximate direction of the eyes, with higher accuracy in detecting position y than x. Finally, we did a feasibility study of one user reading jokes while wearing the device. The system was capable of analyzing facial expressions and eye movements in daily contexts.

CCS CONCEPTS

• **Human-centered computing** → **Interaction devices**; **Mobile devices**.

KEYWORDS

eyewear computing, gaze gesture, wearable computing

ACM Reference Format:
Katsutoshi Masai, Kai Kunze, and Maki Sugimoto. 2020. Eye-based Interaction Using Embedded Optical Sensors on an Eyewear Device for Facial Expression Recognition. In *AHs '20: Augmented Humans International Conference (AHs '20), March 16–17, 2020, Kaiserslautern, Germany*. ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3384657.3384787

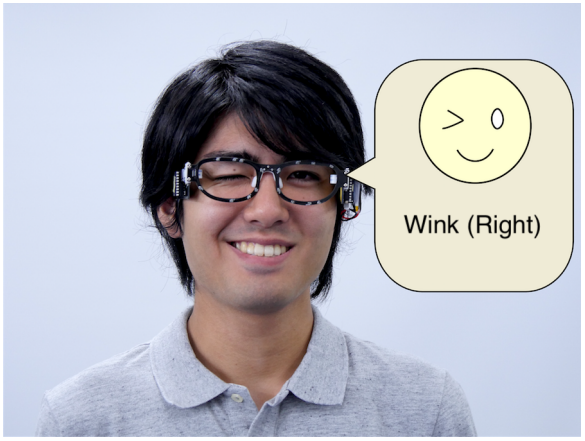

Figure 1: We investigated eye-based interaction using an eyewear device for facial expression recognition

1 INTRODUCTION

People communicate not only through language but also through nonverbal gestures, the tone of their voice, facial expressions, and eye movements. According to Knapp et al., people rely mostly on nonverbal cues in everyday communication [16]. Among these cues, the information from a face is most crucial. People can recognize others' emotional states through facial expressions [13]. Eye movements and blinks reveal information about people's minds; [6, 27]; both are therefore essential to understanding people's inner states and behavior.

In this paper, we explore explicit and implicit eye-based interaction using embedded optical sensors on an eyewear device (Figure 1). The device follows the same measuring principle as [22]. Optical sensors measure reflective intensity, which changes with skin deformation. It can classify basic facial expression states using a Support Vector Machine (SVM). We focused on the eye-based interaction, which could add new interactions to the eyewear device incorporating optical sensors.

The eye-based interaction is explored in the field of human-computer interaction [1, 21], virtual reality [25], and assistive technologies [33]. We envision to use the eye-based interaction with the device to improve work efficiency and enable stress-less input to computers in daily life. Implicit eye movement can convey the cognitive states of the user, which is useful to understand the user's behavior. For example, eye movements and blinks are used to drowsiness detection, such as [4]. If the device is combined with smart home systems and detects the drowsiness, it could ventilate the room automatically to work efficiently. The explicit eye gesture could make the input to computers quick and easy. Eye-gesture input allows users to engage in hands-free interaction, such as winking eyes to change music while working on cooking in the

AHs '20, March 16–17, 2020, Kaiserslautern, Germany

ACM ISBN 978-1-4503-7603-7/20/03…$15.00
https://doi.org/10.1145/3384657.3384787

kitchen. The input is also useful for people who cannot move their hands because of a disability. Eventually, natural interaction with the daily environments using the device integrates humans with their environments.

Since the system is wearable, the users do not need to set up cameras or any other device in their environments. They just wear the devices and input a command to the computer. We considered the social acceptability of the device when choosing to make it in the form of ordinary glasses. Also, the processing cost is much smaller than that of a camera system since the data from the sensors have lower dimensions (16-dimensional 10-bit values per reading).

The contributions of this paper are:

(1) Development of algorithms that can classify explicit eye gestures regardless of facial expression state. As classification algorithms, we used dynamic time warping (DTW) and one nearest neighbor threshold.
(2) Technical evaluation of classifying eye gestures. We recorded 210 gestures (the seven kinds of gestures on three different facial expression conditions, ten times) from each of nine participants in the experiment. The accuracy of classifying seven kinds of gestures is 89.1% with the user-dependent training.
(3) The evaluation of estimating the user's eye gaze position with a 5 X 5 matrix shown on a computer screen. We have five participants, and the result showed the potential of estimating eye gaze position.
(4) The feasibility study to measure the reading activity. The data from the system corresponded to the number of lines, blinks, and facial expressions. We were able to extract the individual data related to them using independent component analysis (ICA).

2 RELATED WORK

Our work is based on works in the field of wearable eye-tracker systems, interactive systems using eye gestures, and wearable approaches to recognize facial expressions.

A wearable eye tracker such as Pupil [12] can measure eye movements robustly using cameras. Recent wearable eye trackers consider a form factor in making it acceptable in the wild setting. For example, InvisibleEye uses four low pixel cameras embedded in the front frame of the device for gaze estimation [30]. However, as visual information from built-in cameras has a processing cost, the systems need the appropriate processors. It makes the devices heavy and bulky. Another sensing modality for wearable eye-tracking is electrooculogram (EOG). Wearable EOG glasses proposed by Bulling et al.can detect eye movements and allowed the wearer to play a desktop computer game using eye movements [1]. Manabe presented an earphone-based interface to detect eye gestures through EOG measurement and considered usage in daily life [21]. JINS MEME is commercial eyewear that measures EOG signals and detects eye movements and blinks. The appearance is almost the same as normal glasses. However, EOG electrodes are necessary to maintain stable contact to make a robust measurement. Additionally, electrodes on the face do not allow for everyday usage because of their appearance and because they are not comfortable to wear for long periods. On the other hand, Ishiguro et al. proposed Aided Eyes for human memory enhancement in daily life [9]. Their prototype sensed eye activities using small phototransistors and infrared LEDs. The entire system can be attached to glasses. The use of an optical sensor is promising, but their system must be placed in front of the eyes, which occludes the wearer's vision. Since we considered the device's mobility and daily usage, we preferred contactless low-signal sensors. We used photo-reflective sensors that are capable of estimating eye movements without occluding the view. We measured skin deformation around the eyes to predict the eye movements. Also, while previous research showed the potential to detect eye movements robustly in daily life, our method investigates simple eye-based interaction using an eyewear device with low cost, lightweight, low-dimensional, and compact sensors. In other words, we focused more on wearability and comfort than EOG methods [1] and camera-based methods [12, 30], which have higher dimensional information.

Eye-based interaction is explored in various fields. It is explored as commands to computers. Commercial EOG glasses are used to allow a user to select between options by tracking a cursor with the eyes [2]. Jota and Wigdor explored the design space of eyelid gestures using a commodity camera. They proposed various application cases such as answering or refusing a phone call by eyelid gestures [11]. Špakov and Majaranta explored the usability of a hands-free interaction system combining gaze pointing and head gestures as commands to computers [31]. Surakka combined two modalities, voluntary gaze direction, and facial muscle activation, for object pointing and selection [28]. They attached EMG electrodes to the user's face. Also, eye-gaze interaction has been explored to improve the user experience in virtual reality and augmented reality. Piumsomboon et al. investigated natural eye-gaze-based interaction for virtual reality [25]. Kytö investigated precise, multimodal selection techniques using head motion and eye gaze for augmented reaity [18]. Hirzle et al. presented the design space for gaze interaction on HMDs and two applications, such as training the eye muscles to help with eye fatigue and tension [8]. Assistive technologies are also a promising area of eye-gazed interaction. To improve communication for people with motor disabilities such as ALS, Zhang et al. presented low-cost and low-effort gaze-based interaction technologies using smartphone [33]. Their research suggests that measuring explicit and implicit eye information has the potential to support interaction with people and computers and allow for hands-free. Our research focused on eye-based interactions in daily life, such as operating appliances at home and reading detection using an eyewear device.

Researchers investigate wearable approaches to recognize facial gestures. Kimura et al. presented an eyeglass-based hands-free video-phone. The glasses have multiple fish-eye cameras to capture a wearer's face. They can yield his/her self-portrait facial expression image [15], but the system was bulky. Gruebler and Suzuki proposed an EMG signal based wearable device that can read positive facial expressions [7]. The system requires skin contact with sensors, which might not be comfortable. Nakamura et al. developed glasses with sensors attached. [24]. The photo-reflective sensor on the device detects the movement of the eyebrow. We narrow our eyebrows when we focus and stare at an object, for example, so these glasses measure our natural interactions to control the

amount of augmented reality (AR) information. Masai et al. developed smart eyewear that can recognize eight facial expressions in daily life by embedded photo-reflective sensors [22]. Those works measure facial expressions to improve interactions using AR or understand their emotional aspects. Our method follows Masai et al.'s sensing principle, yet our work focuses on eye gestures and movements, which could be additional interaction modality to their method. Additionally, we used time-series data to consider more generalizable parameters and subtle changes in sensor values, while Masai et al.'s work considers static data for recognition.

From our review of the related work, we decide to explore the potential to detect eye gestures unobtrusively using optical sensors on an eyewear device, which fits to use in daily life. Therefore, we aim at daily interaction using the device.

3 HARDWARE DESIGN

Our device aims at improving the interaction with daily environments. Figure 2 shows our device. The device follows the same design principle as [22]. We made customized printed circuit boards (PCBs) for sensor units (the front frame) and microcomputer units (temples). We used commercially available temple tips and made other parts, such as the nose pad, and hinges between the PCBs with a 3D printer (Form 2 from Form Lab). The nose pad can be replaced to fit the shapes of users' noses. We used a strap around the back of the head to stabilize the position of the devices.

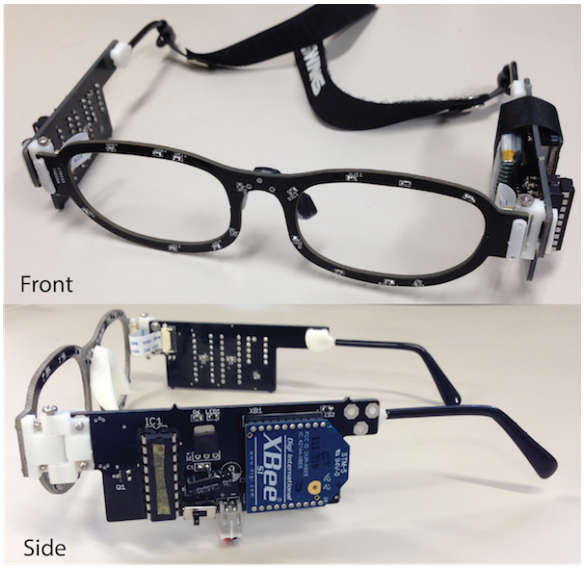

Figure 2: The appearance of our device. It includes 16 photo-reflective sensors on the front frame and microcontrollers placed on each side.

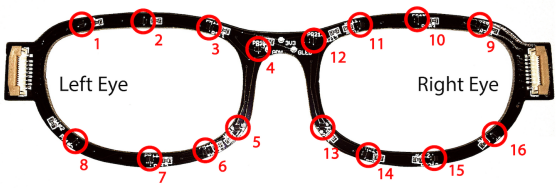

Figure 3: The layout of the sensors. The sensors are distributed all around the eyes.

We placed 16 photo-reflective sensors (NJL5901AR-1-TE1 produced by New Japan Radio Co., Ltd.) on the front frame of the eyewear prototype. Each photo-reflective sensor consists of an infrared LED and phototransistor. The sensors measure the proximity between objects and sensors through reflection intensity. The advantages of using the sensor are the small form factor (1.3 mm x 1.6 mm x 0.6mm), low cost, and fast processing. Figure 3 shows the sensor layout. We used phototransistors with different resistance values because the curvature of a face changes the distance range measured by the sensors. We used lower register values for the phototransistors of the sensors that measure close distance, such as the sensors close to the center of the front frame.

The device measures skin deformation around the eyes. Since eyeball movements cause deformation around the eyes and eyelids, such eye movements can change sensor values.

We placed one peripheral interface controller (PIC, 16F1827 produced by Microchip Technology) on each temple. Each PIC converts the voltage from each of the eight sensors to 10-bit digital value. For every PIC, we put one transistor to turn the infrared LED of the sensors on and off, which reduces the influence of ambient light. We define a data sample as 16 sensor values. Each dimension of the data sample is a subtraction of the sensor value with the LEDs on and off. XBee transmits data samples wirelessly to a laptop. The 3.7v lipo battery powers the PIC after the regulator adjusts the voltage to 3.3v.

4 EYE-BASED INTERACTION

We investigated the possibility of explicit and implicit eye-based interaction using embedded optical sensors on the eyewear device. We first evaluated the possibility of explicit interaction by classifying several eye gestures. Then we tested the feasibility of eye-tracking. Finally, we ran a feasibility trial collecting implicit eye movement data while the user was reading.

4.1 Eye Gesture Classification

Measurement of explicit eye-based interaction is fit to perform in daily life because it is subtle, easy, and hands-free. We suppose eye-based interaction is suited for applications with simple interfaces, such as turning the pages of an e-book or playing and pausing music. These daily-uses of the technique are compatible with our eyewear design. Also, the combination of eye movement inputs can be used for a command input like the music player application of Manabe et al. [21].

4.1.1 Gesture Set. Figure 4 shows the gestures we aimed to detect with the device. All seven kinds of gestures start from and end at a neutral eye position, shown on the top left of the figure. Among the gestures, people make only winks explicitly. The advantage of using it as an input is that it is possible to avoid unconscious inputs. We considered four basic directions of eye gestures for simplicity. For these four gestures, we asked that the user moves their eyes in a certain direction as much as the user can to clarify

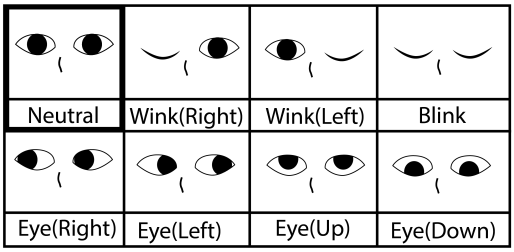

Figure 4: The Set of Gestures. All gestures start and end at the neutral eye position on the top left.

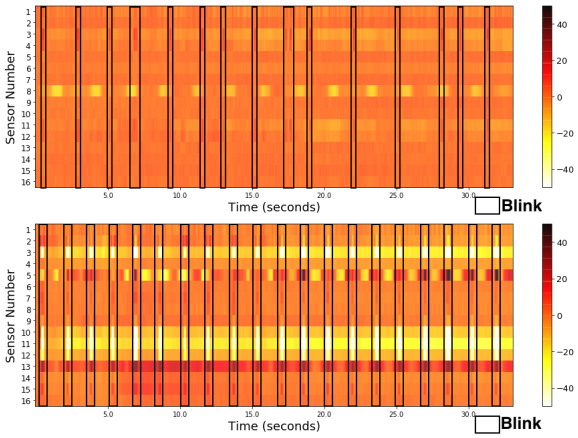

Figure 5: The sensor recordings of involuntary blinks (top) and strong voluntary blinks (bottom).

the difference between the four directional gestures. With this setup, the device is not able to detect the subtle movement while it can help to avoid false-positives of the eye gesture input. Since changes in facial expression can influence the sensor values, we asked users to make eye gestures in different facial expression states. We considered three states for simplicity's sake: positive (smile), neutral, and negative (anger). The categories are retrieved from Ekman's basic facial expressions [5].

4.1.2 Blink Detection. Blinks can be involuntary or voluntary. We investigated if there is a difference in their sensor values to avoid unexpected input. To see the difference between the sensor data of voluntary and involuntary blinks, we made two recordings of one participant holding a neutral face as a preliminary experiment. For involuntary blinks, we recorded 35 seconds of data samples while the user watched a neutral video. For voluntary blinks, we recorded 20 blinks for 35 seconds. For both recordings, we recorded videos of the user wearing the glasses. Figure 5 shows the heat map of the results. In the figure, we annotated the blinks manually by checking the video. The values on the heat maps are the subtraction of the initial data sample from time series raw data samples. Figure 5 confirms both blinks changed the sensor values in a certain pattern.

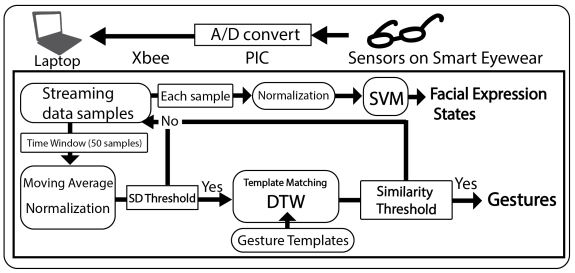

Figure 6: The overview of the system. The data samples from the sensors are applied to detect eye gestures with DTW and facial expressions with SVM.

We can see that strong voluntary blinks cause a bigger change to the values in the sensors that detect the deformation of the upper cheeks. Therefore, strong voluntary blinks can be differentiated from involuntary blinks. We can make use of voluntary blinks to input a command to computers as they can be stronger than involuntary ones.

4.1.3 Algorithm. Figure 6 gives an overview of our system. The eye gesture classification algorithm consists of two stages: data preprocessing and template matching.

Data Preprocessing. From the device, we acquire a 16-dimensional data sample per reading. The sampling frequency is 30 Hz. From the data streaming, we put the information from the data streaming into a buffer. The size of the buffer is 70 data samples. We calculated the standard deviation of the data samples for each sensor in the buffer. We compare the summation of the standard deviation with a threshold to determine whether there is a gesture in the buffer. If the summation is lower than the threshold, we classify it as no gesture. Otherwise, we regard it as a gesture. Then, we apply a simple moving average of 10 sequences to the buffer to smooth out the noise. Then, we normalize each sensor dimension of the time series samples in the array separately to a zero mean and unit variance.

Template Matching. If the gesture is detected, we compare the time series with matching templates of all of the seven gestures. If the time series array is similar enough to one of the templates, we regard the array as one of the seven gestures.

For the template matching, we applied one of the most standard time series similarity measures: DTW. This algorithm calculates the distance between two different time series. A shorter distance means the two are similar. Considering the possibility of real-time detection, we applied FastDTW [26]. This algorithm is an approximation of DTW that has a linear time and space complexity. As the signals are multi-dimensional, we used a distance measure as the summation of absolute difference in all sensor dimensions [29]. The formula for calculating the distance (D) between two K-dimensional time series, i-th sample of A and j-th sample of B, is as follows:

$$D = \sum_{k=1}^{K} |A_i(k) - B_j(k)| \quad (1)$$

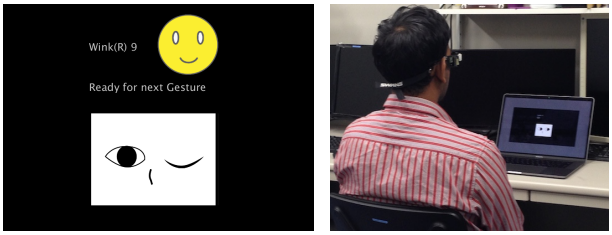

Figure 7: (left) The user interface used for the recording in the experiment. (right) The experiment setup.

By performing DTW on the first-order derivatives of the feature values, it is possible to consider the high-level feature of the shape of the time series [14]. We used the derivatives because, while the data sample at the starting point of the signal differs depending on the position of the device and facial expression states, how the data samples change over time is more consistent when users make eye gestures. Therefore, we compared the similarities between the derivatives of the buffer and the derivatives of all matching templates. We made matching templates by averaging the resized buffer to 70 samples for each kind of gesture in the experiment. Through the comparison, we found the cost matrix (CS) of the seven calculated distances. We classify the buffer signals as the closest gesture template ($argMin(CS)$) if $Min(CS)$ is lower than a threshold.The threshold rejects confusing gestures or different gestures. The bigger threshold helps to void the false positives of classification. However, if the threshold is too big, the true positives cannot be detected. For the evaluation, we did not set this threshold because all data contain gestures.

4.1.4 Evaluation. We evaluated the accuracy of the classification of eye gestures while users kept three facial expression states. Nine users participated in our study: eight of them are male, and they are all in their 20s. Eight of them are Japanese, and another is French. For this study, we developed the software to record the sensor data samples of the gestures with Processing, a Java-based language. Since the photo-reflective sensors are vulnerable to intense ambient light, we ran the study in a quiet room far from windows (Figure 7). We used the Python environment for the following analysis.

4.1.5 Procedure. Figure 8 shows a summary of the procedure. We collected 210 gestures (seven kinds of gestures x three facial expression conditions x ten times) for each participant.

Firstly, each participant was asked to sit in a chair in front of a laptop on a desk. They wore the prototype with eyewear band strap for stability. The observer introduced the software for the experiment to the participant. The observer explained that the participants would make seven different eye gestures ten times each for three different facial expression conditions (neutral, positive, and negative). When the positive expression is recognized, the zygomatic major muscle has been activated, while negative emotion activates the corrugator supercilii muscles [19]. As such, we asked the users to activate those muscles in the experiment. The observer told them that each gesture should start and end with a neutral eye position (starting at the center of the computer screen), and the order of the gestures was periodic so the participants would not

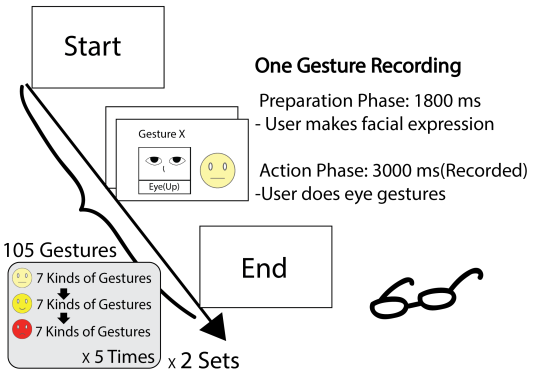

Figure 8: The summary of the experiment procedure.

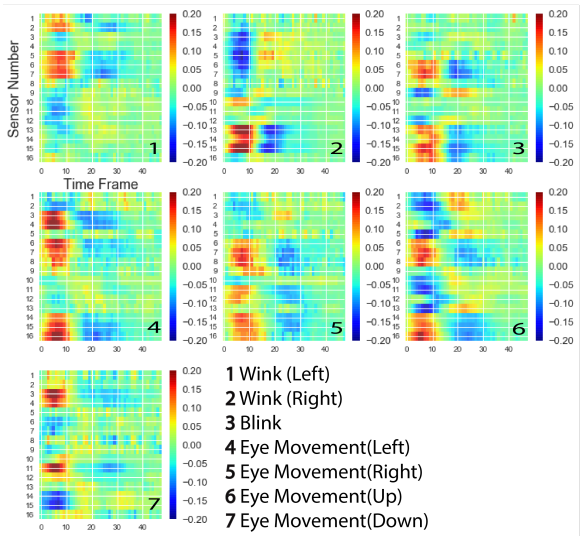

Figure 9: The average eye gesture templates of all 9 users.

make the wrong gesture. After giving the general instructions, the observer repeated the following process:

(1) The observer starts the software for the experiment and reminds the participant to keep specific expressions in the action phase.
(2) In the preparation phase (1800 ms), the software instructs the participant in which kind of facial gesture and expression they will make with text and images. Figure 7 (left) shows the screenshot of the software. In this phase, the user holds the instructed facial expression until the next preparation phase.
(3) The software asks the participants to make the instructed gesture in the action phase (3000 ms). The software records the data samples from the sensors in this phase.
(4) The Steps of (3) and (4) are repeated for seven gestures five times each.

(5) After the software stops, the observer gives the participant a short break before returning to step (1). Steps (1) to (5) are repeated twice, and the process takes 20 to 30 minutes in total.

We divided the recording of each gesture into two phases. In the preparation phase, the software instructed the user which gesture and facial expression to make next. In the action phase, the software told the user to make the gesture. By having two phases, the gestures could be recorded with almost the same timing, which helps to make matching templates. The software recorded the sensor data samples with 30 Hz only during the action phase.

The observer recorded a video of the experiment with the laptop's built-in camera. This recording was used to check manually if the participant held the right facial expression and made the right gesture.We had only 203 items of gesture data from one participant as our device did not work in the middle of one recording. Therefore, we collected 1883 (210 x 9 - 7) items of gesture data from nine participants for the eye gesture classification.

4.1.6 Result:Facial Expression. We assumed facial expression conditions and eye gestures could be simultaneously detected because we followed the same sensing principle as [22]. For the classification of facial expressions, we evaluated using the dataset acquired from each participant separately. We applied SVM (linear kernel, C = 100) as a classifier. Each dimension (each dimension includes the time series sensor values of one sensor from one participant's recorded gestures) from the experiment is normalized to zero mean and unit variance. Then, we used 10-fold cross-validation to the dataset with the SVM classifier. The micro-averaged accuracy of classifying three facial expression states are 90.9% with individual training.

4.1.7 Results: Eye Gestures. We used the first 70 data samples of each gesture to make and match templates (all gesture data have 70 X 16 dimensions for training and test). Since we collected two sections, we trained with one section of data and tested with another section of data for each user. To make matching templates, we used all of the recorded gestures' sensor values and averaged for each kind of gesture. The micro-averaged accuracy of classifying seven gestures from nine participants is 89.1% (89.4% on the neutral condition, 88.7% on the positive condition, and 89.0% on the negative condition) with user-dependent templates. Facial expression condition has no big influence on accuracy because our algorithm did not consider the sensor values on the initial state. Figure 10 shows the confusion matrix of the accuracy of classifying the seven kinds of gestures. Among the seven kinds, the system recognized blinks least robustly. The French male showed the lowest accuracy with 68.6% because, when he winked, he tended to close both eyes. Of his left-eye winks, 40.3% were classified as right-eye winks or blinks, and of blinks, 30.0% were classified as right-eye winks. Also, we could not control the start time of his gestures. He started the gestures according to his own arbitrary timing, which weakened the features of the template as we made the templates by averaging gesture data samples in time series.

We made the average templates for each kind of gesture using the data samples of all the participants (Figure 9). The sensor number corresponds to Figure 3. The different sensors on both the upper and lower sides of the frame react to the different gestures. We applied leave-one-user-out cross-validation. The micro-averaged accuracy of the templates is 70.5%. Figure 11 shows the confusion matrix. Compared to the individual templates, the accuracy is lower, especially for blinks. This is partly because the strength of the blinks was not stable within the trial and among the users. However, right eye wink, left eye movement, and down movement were recognized with more than 80.0% accuracy. The average processing time was 63.6 milliseconds per gesture with MacBook Pro (2.9 GHz Intel Core i7).

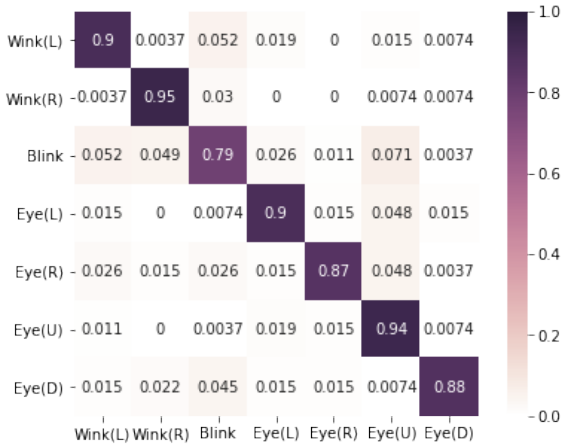

Figure 10: Confusion matrix of the accuracy when the system classified seven gestures using the user-dependent templates.

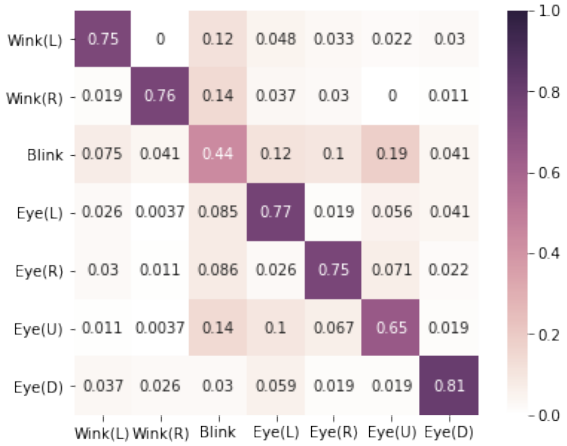

Figure 11: Confusion matrix of the accuracy when the system classified seven gestures using the averaged templates of all users.

4.2 Evaluation 2: Eye Gaze Position

To explore the potential of eye tracking using an eyewear device, we evaluated the accuracy of estimating eye gaze position. The

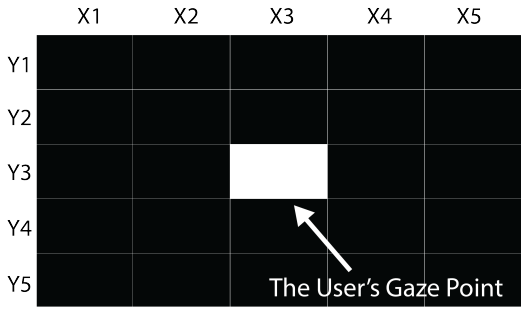

Figure 12: The screen shown to the participants

estimation of the position was based on the skin deformation caused by the directional change of the eyeballs.

Five students participated in the study. All of them were in their 20s, and one was female. The study was done one by one.

4.2.1 Procedure. First, the participants wore the device and sat at a distance of 60 cm from a 23-inch screen. On the screen, a 5 x 5 matrix was shown (Figure 12). Each class has approximately 5 degrees (vertical) and 10 degrees (horizontal) of the field of view. After the participants started the software, the colored square changed in order from (X1, Y1), (X2, Y1), ...,(X5, Y1),(X1, Y2), ..., to (X5, Y5). Whenever the position changed, the color of the square turned to gray for the first 500 ms to indicate the transition. The participants changed their gaze position to the gray rectangle. After the color changed from gray to white, the software recorded the sensor data samples (1000ms). This process made sure that we recorded only when the participant gazed at the correct position. Each person repeated the process of looking at 25 positions eight times: so the dataset of each participant includes the data samples of eight seconds for all 25 positions. The data samples of the 25 positions are labeled based on the position.

We asked the participants to do three things during the experiment to reduce noise data: (1) look at the center of a white rectangle on the screen: (2) hold a neutral face and blink only during the transition time to reduce the artifacts of the user's behavior (including facial expression change) to the sensor values: and (3) follow the white place with eyes only without moving their head (we did not consider the head pose). We also applied outlier rejection to reduce the noise caused by blinks. We used the following formula for outlier rejection in each position dataset. D is the data samples in each class: and d is a data sample that belongs to D. Mu and std are 16-dimensional values (average and standard deviation) calculated for each dimension of sensors within the class.

$$outliers = abs(d - mean(D)) < 2 * std(D) \; \forall d \subset D \qquad (2)$$

Later, we normalized each sensor dimension of the datasets to zero mean and unit variance. We merged the 25 classes of the data samples into five classes in two ways: horizontally and vertically (we merged the five classes in each column or row into a new class). Then we applied five-fold cross-validation using an SVM classifier (kernel = rbf, C = 1000) to each dataset. We repeated the process for every participant.

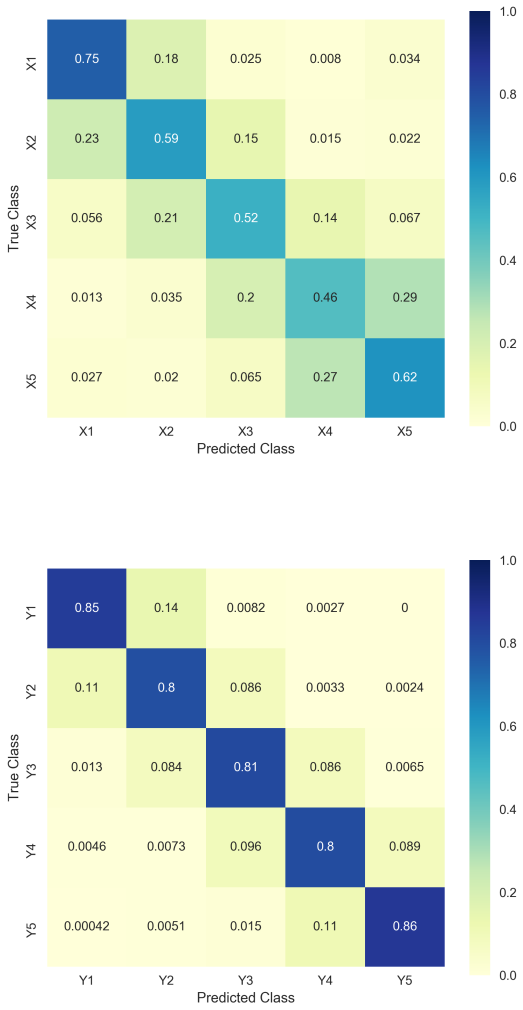

Figure 13: Confusion matrix of (above) horizontal gaze direction (below) estimating vertical gaze direction.

4.2.2 Results. Figure 13 shows the average accuracy of each participant's results with user-dependent classifiers. The figures indicate that the sensor data and eye gaze position are correlated. Vertical movements show a higher correlation (micro-averaged accuracy 82.4%) than the horizontal movements (micro-averaged accuracy 58.8%). This means that the vertical movements of eyes cause more skin deformation around the area measured by the device than horizontal movements. Most of the false predictions are classified as the area next to the area of the true classes. The accuracy is higher in the corner area compared to that of the central area. It is hard to identify the exact position because we estimated only based on the skin deformation measured by our device. The deformation was caused by the direction change of eyes, so our device cannot be used for eye pointing. However, our device can measure approximately

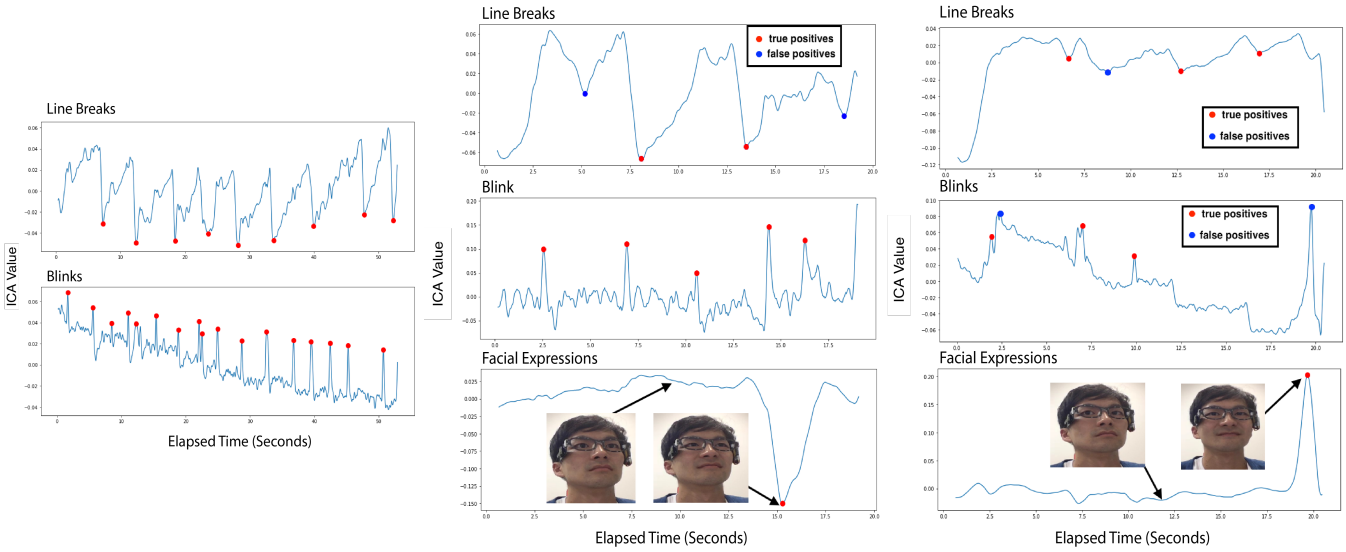

Figure 14: The time series data after applying FastICA to sensor data samples: while the user read with a neutral face (left), and while the user read and smiled in the end (center), while the user moved, read and smiled (right).

where the user looks at in an experimental condition. Information from the device can be combined with existing eye tracker information to improve the accuracy of measuring eye pointing with the eye tracker.

4.3 Feasibility Study: Reading Detection

To demonstrate the potential of our device for implicit eye-based interaction in daily contexts, we ran a feasibility study of reading detection. We collected data while the subject read, and we analyzed the sensor for facial expressions and eye movements. Since reading is vital for learning, reading detection is useful for quantifying and managing the activity to motivate users to read more [17]. Implicit tagging of facial expressions to the content could help users to search for their favorite content and could be beneficial for analyzing and recommending content.

One participant (a male in his 20s) read ten English jokes wearing the device. We chose jokes to induce non-neutral facial expression (positive). The jokes are retrieved based on [3]. The length of the jokes ranged from two lines to eleven lines. He read the texts shown on the screen in a text box of 900-pixel width on a 23-inch screen (1920 x 1080 p). Soon after he finished reading each joke, he pressed the keyboard in order to record the sensor data samples from only the reading activity. Then, the user evaluated each of the jokes by 1) how well the user understood the joke (1: not completely understand–9: completely understand) 2) how funny it was (1: not funny–9: very funny) with the 1-9 Likert scale. We also recorded videos of the wearer's face to count his eye movements and facial expression changes.

To the recorded data samples, we applied a simple moving average of five sequences. Later, we used FastICA from the Scikit-learn library to process the data samples into four-dimensional time-series data. We found that four dimensions of the data are categorized into 1) facial expression change 2) horizontal eye movements 3) blinks and the user's behavior and 4) the other factors such as ambient light noise. We manually categorized the data. We applied a moving average of 2-20 sequences depending on the category and the amount of the noise (for blinks: 2-5 sequences, for horizontal eye movements which correspond to line breaks: 10–20 sequences, and for facial expression: 20 sequences). We applied a peak detection algorithm (the Python implementation of "findpeaks function" in MATLAB Signal Processing Toolbox). We manually adjusted the parameters of the peak detection algorithm for each result.

From the recordings, we introduced three specific examples. One shows the data on reading activity only when the user kept a neutral face; another indicates the data on the facial expression change (neutral to positive); the third illustrates the data with facial expression change, head motion, and body movements. Note that the time scale of each figure is different depending on the length of the jokes.

The first example is the data on the nine and a half-line joke the user understood (8 points) and evaluated as a little funny (6 points)(Figure 14, left). We confirmed from the video that there is no facial expression change. The above of Figure 14, left) shows that the data of horizontal eye movements. Each peak corresponded with an eye movement that went from the end of a line of text to the new line. The red dots at the bottom of the figure show the blinks of the user. We confirmed that all blinks except the last one were successfully detected. The last blink was not detected because the recording ended in the middle of the blink. This example demonstrates the potential for quantifying how many lines or the words the user reads [17] using our device.

The second example is the data on the two and a half-line joke the user understood (9 points) and evaluated as funny (8 points)(Figure 14, center). We realized from the video that the false positives blue dots at 5 and 18 seconds) on the line break figure are backward saccades: the behavior of looking a couple of words behind. We successfully detected all blinks except for the last since the recording ended in the middle of it. The bottom figure shows that the ICA time series data correlated with the actual facial expression change. From this example, we can see it is still possible to detect line breaks and blinks while detecting the change in facial expressions.

The last example is the data on the three and a half-line joke the user understood (9 points) and evaluated as funny (8 points)(Figure 14, right). In the line break figure, we found the influence of the user's movement toward the screen and to look down on the keyboard. We detected all of the line breaks, but the influence of the blink caused a wrong line break detection (false positive). If blinks happened in the middle of the user's behavior, they were not detected. The user's behavior also caused the false detection of a blink, which means the ICA algorithm could not separate all the factors correctly for this example with the peak detection algorithm. The figure of the blinks on Figure 14, right) shows that the time series data included the blinks, the movements of the line break, and facial expression changes. The bottom of the figure demonstrates that we successfully detected the facial response.

Based on these examples, we think the device has the potential for implicit-interaction, such as automatic tagging and contents analysis, by making use of eye movements and blinks. If the device can classify an activity state like [10] by adding an inertial measurement unit (IMU), it would make the reading analysis more reliable with the device.

5 DISCUSSION

Because eyes move implicitly to look at people or the surroundings, the proposed system could not recognize whether the gesture input was intentional or not. This is a common problem of using eye gestures as an interaction technique and can be solved if we use an explicit gesture (e.g., wink) as a trigger command. Another solution would be to consider the head pose and body movement in conjunction with eye movement [20]. Detecting implicit eye gestures also opens up the possibility of an ambient interface that understands people's inner states. This interface could provide information and facilitate natural interaction with the environment or robots.

We chose the set of gestures by focusing on ones related to eye movement. However, the eyewear device with optical sensors can recognize hand-to face gestures and facial gestures since the device was able to measure the skin deformation caused by hand-to-face input and facial expressions such as [23, 32]. We could also consider hand-to-face gestures as trigger commands because, as we realized in technical evaluation, not all users are good at winking their eyes, which is the only eye gesture that is always explicit. The performance of eye gestures could be culture-dependent and user-dependent. We need to test the performance level of the gestures such as effort or strain to show its usability and practicality.

Eye gestures changed the sensor values only subtly. Therefore, we asked the participants to make exaggerated gestures, and the experimental design avoided influences from noises. We used a simple algorithm, and it worked with a stable condition, which suggests that our system could capture valuable features using photo-reflective sensors. This is the first step towards usage in real life. However, there is a risk of classifying non-defined gestures as targeted gestures with the algorithm if the system is used in real-life settings. If there are other influences during the gesture, the system may not work well. Possible sources of noise are from head motion, facial expression change, device displacement, and ambient light. Accuracy might be improved if we use a deep learning approach, which can accommodate such noises with a large data set. On the other hand, the deep learning would reacquire the cost of collecting the large dataset for training and have latency, which would prevent to use the gestures in real-time. Additionally, integrating IMU with the device could compensate for the photo-reflective sensor readings, which were influenced by head motion and the user's behavior. Because head motion is also related to non-verbal communication, the integration could deepen the analysis of implicit user's behavior.

6 LIMITATIONS AND FUTURE WORK

For the first experiment, we only considered the classification of the gestures. The result showed that our system worked in experimental settings. To ensure the system can be used in the wild setting, we would consider the detection problem in the future.

Direct sunlight should be avoided. Since the phototransistors of the sensors are easily influenced by ambient light, the sensor value can be saturated under the sun. A light-shielding cover can lessen this effect but represents a trade-off between the function and the appearance of the device.

The demographics of the experiments are biased. Most of the participants are male. Although the shape and features of a face are different depending on the nationality, gender, and age, we assume our method can work as long as it can measure skin deformation around eyes at a close distance. We can adjust the register values for the phototransistors of the sensors to avoid sensor saturation. We can also control the distance by changing the design of the nose pad, which can be replaced. To test the hypothesis, we are planning to gather more participants from various demographics.

7 CONCLUSION

We have presented a system that enables explicit and implicit eye-based interactions using an eyewear device with optical sensors. The device took the form of everyday glasses and incorporated 16 optical sensors. We used DTW to classify eye gestures. The average accuracy of detecting seven different eye gestures was 89.1% with user-dependent training. We demonstrated the possibility of estimating gaze positions in experimental conditions. We also showed the feasibility of reading detection in experimental conditions. Although user behavior caused false positives with the simple peak detection algorithm, we were able to detect blinks and line breaks in addition to facial expression changes by applying ICA.

ACKNOWLEDGMENTS

This research was partially supported by JST CREST JP-MJCR14E1, and Keio KLL research grant.

REFERENCES

[1] Andreas Bulling, Daniel Roggen, and Gerhard Tröster. 2008. It's in Your Eyes: Towards Context-awareness and Mobile HCI Using Wearable EOG Goggles. In *Proceedings of the 10th International Conference on Ubiquitous Computing (UbiComp '08)*. ACM, New York, NY, USA, 84–93. https://doi.org/10.1145/1409635.1409647

[2] Murtaza Dhuliawala, Juyoung Lee, Junichi Shimizu, Andreas Bulling, Kai Kunze, Thad Starner, and Woontack Woo. 2016. Smooth Eye Movement Interaction Using EOG Glasses. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction (ICMI '16)*. Association for Computing Machinery, New York, NY, USA, 307–311. https://doi.org/10.1145/2993148.2993181

[3] R. I. M. Dunbar, Jacques Launay, and Oliver Curry. 2016. The Complexity of Jokes Is Limited by Cognitive Constraints on Mentalizing. *Human Nature* 27, 2 (01 Jun 2016), 130–140. https://doi.org/10.1007/s12110-015-9251-6

[4] P. Ebrahim, W. Stolzmann, and B. Yang. 2013. Eye Movement Detection for Assessing Driver Drowsiness by Electrooculography. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*. 4142–4148. https://doi.org/10.1109/SMC.2013.706

[5] P. Ekman. 1989. The argument and evidence about universals in facial expressions. *Handbook of social psychophysiology* (1989), 143–164.

[6] Roger P.G. Van Gompel, Martin H. Fischer, Wayne S. Murray, and Robin L. Hill. 2007. Chapter 1 - Eye-movement research: An overview of current and past developments. In *Eye Movements*, Roger P.G. Van Gompel, Martin H. Fischer, Wayne S. Murray, and Robin L. Hill (Eds.). Elsevier, Oxford, 1 – 28. https://doi.org/10.1016/B978-008044980-7/50003-3

[7] A. Gruebler and K. Suzuki. 2014. Design of a Wearable Device for Reading Positive Expressions from Facial EMG Signals. *Affective Computing, IEEE Transactions on* 5, 3 (July 2014), 227–237. https://doi.org/10.1109/TAFFC.2014.2313557

[8] Teresa Hirzle, Jan Gugenheimer, Florian Geiselhart, Andreas Bulling, and Enrico Rukzio. 2019. A Design Space for Gaze Interaction on Head-Mounted Displays. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*. Association for Computing Machinery, New York, NY, USA, Article Paper 625, 12 pages. https://doi.org/10.1145/3290605.3300855

[9] Yoshio Ishiguro, Adiyan Mujibiya, Takashi Miyaki, and Jun Rekimoto. 2010. Aided Eyes: Eye Activity Sensing for Daily Life. In *Proceedings of the 1st Augmented Human International Conference (AH '10)*. ACM, New York, NY, USA, Article 25, 7 pages. https://doi.org/10.1145/1785455.1785480

[10] Shoya Ishimaru, Kai Kunze, Koichi Kise, Jens Weppner, Andreas Dengel, Paul Lukowicz, and Andreas Bulling. 2014. In the Blink of an Eye: Combining Head Motion and Eye Blink Frequency for Activity Recognition with Google Glass. In *Proceedings of the 5th Augmented Human International Conference (AH '14)*. Association for Computing Machinery, New York, NY, USA, Article Article 15, 4 pages. https://doi.org/10.1145/2582051.2582066

[11] Ricardo Jota and Daniel Wigdor. 2015. Palpebrae Superioris: Exploring the Design Space of Eyelid Gestures. In *Proceedings of the 41st Graphics Interface Conference (GI '15)*. Canadian Information Processing Society, Toronto, Ont., Canada, Canada, 273–280. http://dl.acm.org/citation.cfm?id=2788890.2788938

[12] Moritz Kassner, William Patera, and Andreas Bulling. 2014. Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction. In *Adjunct Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14 Adjunct)*. ACM, New York, NY, USA, 1151–1160. https://doi.org/10.1145/2638728.2641695

[13] D. Keltner, P. Ekman, G. C. Gonzaga, and J. Beer. 2003. Expression of emotion. *Handbook of Affective Sciences* (2003), 411–414.

[14] Eamonn J Keogh and Michael J Pazzani. 2001. Derivative dynamic time warping. In *Proceedings of the 2001 SIAM International Conference on Data Mining*. SIAM, 1–11.

[15] Shinji Kimura, Masaaki Fukuomoto, and Tsutomu Horikoshi. 2013. Eyeglass-based Hands-free Videophone. In *Proceedings of the 2013 International Symposium on Wearable Computers (ISWC '13)*. ACM, New York, NY, USA, 117–124. https://doi.org/10.1145/2493988.2494330

[16] L. Knapp, M., A. Hall, J., and G. Horgan, T. 2013. *Nonverbal communication in human interaction*. Cengage Learning.

[17] Kai Kunze, Katsutoshi Masai, Masahiko Inami, Ömer Sacakli, Marcus Liwicki, Andreas Dengel, Shoya Ishimaru, and Koichi Kise. 2015. Quantifying Reading Habits: Counting How Many Words You Read. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '15)*. ACM, New York, NY, USA, 87–96. https://doi.org/10.1145/2750858.2804278

[18] Mikko Kytö, Barrett Ens, Thammathip Piumsomboon, Gun A. Lee, and Mark Billinghurst. 2018. Pinpointing: Precise Head- and Eye-Based Target Selection for Augmented Reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. Association for Computing Machinery, New York, NY, USA, Article Paper 81, 14 pages. https://doi.org/10.1145/3173574.3173655

[19] Jeff T. Larsen, Catherine J. Norris, and John T. Cacioppo. 2003. Effects of positive and negative affect on electromyographic activity over zygomaticus major and corrugator supercilii. *Psychophysiology* 40, 5 (2003), 776–785. https://doi.org/10.1111/1469-8986.00078

[20] Juyoung Lee, Shaurye Aggarwal, Jason Wu, Thad Starner, and Woontack Woo. 2019. SelfSync: Exploring Self-Synchronous Body-Based Hotword Gestures for Initiating Interaction. In *Proceedings of the 23rd International Symposium on Wearable Computers (ISWC '19)*. Association for Computing Machinery, New York, NY, USA, 123–128. https://doi.org/10.1145/3341163.3347745

[21] Hiroyuki Manabe, Masaaki Fukumoto, and Tohru Yagi. 2015. Conductive Rubber Electrodes for Earphone-based Eye Gesture Input Interface. *Personal Ubiquitous Comput.* 19, 1 (Jan. 2015), 143–154. https://doi.org/10.1007/s00779-014-0818-8

[22] Katsutoshi Masai, Yuta Sugiura, Masa Ogata, Kai Kunze, Masahiko Inami, and Maki Sugimoto. 2016. Facial Expression Recognition in Daily Life by Embedded Photo Reflective Sensors on Smart Eyewear. In *Proceedings of the 21st International Conference on Intelligent User Interfaces (IUI '16)*. ACM, New York, NY, USA, 317–326. https://doi.org/10.1145/2856767.2856770

[23] Katsutoshi Masai, Yuta Sugiura, and Maki Sugimoto. 2018. FaceRubbing: Input Technique by Rubbing Face Using Optical Sensors on Smart Eyewear for Facial Expression Recognition. In *Proceedings of the 9th Augmented Human International Conference (AH '18)*. Association for Computing Machinery, New York, NY, USA, Article Article 23, 5 pages. https://doi.org/10.1145/3174910.3174924

[24] Hiromi Nakamura and Homei Miyashita. 2010. Control of Augmented Reality Information Volume by Glabellar Fader. In *Proceedings of the 1st Augmented Human International Conference (AH '10)*. ACM, New York, NY, USA, Article 20, 3 pages. https://doi.org/10.1145/1785455.1785475

[25] T. Piumsomboon, G. Lee, R. W. Lindeman, and M. Billinghurst. 2017. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*. 36–39. https://doi.org/10.1109/3DUI.2017.7893315

[26] Stan Salvador and Philip Chan. 2007. Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intell. Data Anal.* 11, 5 (Oct. 2007), 561–580. http://dl.acm.org/citation.cfm?id=1367985.1367993

[27] Daniel Smilek, Jonathan S.A. Carriere, and J. Allan Cheyne. 2010. Out of Mind, Out of Sight. *Psychological Science* 21, 6 (2010), 786–789. https://doi.org/10.1177/0956797610368063 arXiv:https://doi.org/10.1177/0956797610368063 PMID: 20554601.

[28] Veikko Surakka, Marko Illi, and Poika Isokoski. 2004. Gazing and Frowning As a New Human–computer Interaction Technique. *ACM Trans. Appl. Percept.* 1, 1 (July 2004), 40–56. https://doi.org/10.1145/1008722.1008726

[29] Gineke A ten Holt, Marcel JT Reinders, and EA Hendriks. 2007. Multi-dimensional dynamic time warping for gesture recognition. In *Thirteenth annual conference of the Advanced School for Computing and Imaging*, Vol. 300.

[30] Marc Tonsen, Julian Steil, Yusuke Sugano, and Andreas Bulling. 2017. InvisibleEye: Mobile Eye Tracking Using Multiple Low-Resolution Cameras and Learning-Based Gaze Estimation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 3, Article 106 (Sept. 2017), 21 pages. https://doi.org/10.1145/3130971

[31] Oleg Špakov and Päivi Majaranta. 2012. Enhanced Gaze Interaction Using Simple Head Gestures. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12)*. ACM, New York, NY, USA, 705–710. https://doi.org/10.1145/2370216.2370369

[32] Koki Yamashita, Takashi Kikuchi, Katsutoshi Masai, Maki Sugimoto, Bruce H. Thomas, and Yuta Sugiura. 2017. CheekInput: Turning Your Cheek into an Input Surface by Embedded Optical Sensors on a Head-mounted Display. In *Proceedings of the 23rd ACM Symposium on Virtual Reality Software and Technology (VRST '17)*. ACM, New York, NY, USA, Article 19, 8 pages. https://doi.org/10.1145/3139131.3139146

[33] Xiaoyi Zhang, Harish Kulkarni, and Meredith Ringel Morris. 2017. Smartphone-Based Gaze Gesture Communication for People with Motor Disabilities. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2878–2889. https://doi.org/10.1145/3025453.3025790