FaceRubbing: Input Technique by Rubbing Face using Optical Sensors on Smart Eyewear for Facial Expression Recognition

Katsutoshi Masai, Yuta Sugiura, Maki Sugimoto
Keio University
Yokohama, Kanagawa, Japan
(masai,sugiura,sugimoto)@imlab.ics.keio.ac.jp

ABSTRACT

With the emergence of the wearable devices, the method to make use of the limited input space is required. This paper presents an input technique to a computer by rubbing face using optical sensors on smart eyewear. Since rubbing gesture occurs in daily life, our system enables a subtle interaction between the user and a computer. We used the smart eyewear based on the work by [5]. Although the device is developed for facial expression recognition, our method can recognize rubbing gesture independent from facial expression recognition.

The embedded optical sensors measure the skin deformation caused by rubbing on the face. We detect the gestures using principal component analysis (PCA) and peak detection. we classify the area of the gesture with a random forest classifier. The accuracy of detecting rubbing gesture is 97.5%. The classification accuracy of 10 gesture area is 88.7% with user-independent training. The system can open up a new interaction method for smart glasses.

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**;

KEYWORDS

Wearable Computing, Input Technique, Eyewear Computing

ACM Reference Format:

Katsutoshi Masai, Yuta Sugiura, Maki Sugimoto. 2018. FaceRubbing: Input Technique by Rubbing Face using Optical Sensors on Smart Eyewear for Facial Expression Recognition. In *AH2018: The 9th Augmented Human International Conference, February 7–9, 2018, Seoul, Republic of Korea.* ACM, New York, NY, USA, Article 4, 5 pages. https://doi.org/10.1145/3174910.3174924

1 INTRODUCTION

In recent years, various wearable devices became available such as smart watches, smart headsets, smart clothes and so on. Among wearable devices, smart glasses are attracting attention [1]. Many companies develop optical see-through displays (OST) for augmented reality application. JINS made a smart eyewear (JINS MEME) that can measure the physical condition of the user. However, the wearable devices have only the limited input space because of the small size. It is cumbersome to operate the device from additional devices. Also, in the situations where OST is used, it is preferable for the user to control the information without being noticed by the other party. For example, people would like to cut off unexpected notifications during meetings or access manuscripts during a presentation while keeping eye contacts on the audiences. However, current input technologies like tapping or flicks use only a limited input space and mid-air gestures are noticeable and not subtle.

AH2018, February 7–9, 2018, Seoul, Republic of Korea

ACM ISBN 978-1-4503-5415-8/18/02…$15.00
https://doi.org/10.1145/3174910.3174924

Figure 1: Left: the eyewear device used for the recognition. Right: the user makes a rubbing gesture on face.

In this paper, we introduce a new input method of hand-to-face gesture using optical sensors on the smart eyewear (Figure 1). We used the redesigned device proposed by [5]. Although the device is developed for facial expression recognition, we make use of the sensors for an input method to a computer. This method allows users to input various commands by rubbing the different areas of the facial surface. The technology can be integrated not only into the proposed device but more general eyewear computing devices such as an OST because of the small factor of the optical sensors. The advantages of adopting the rubbing gesture are as follows. First, we can recognize the rubbing gestures independent from facial expression because there is no periodic motion change in facial expression change. It means we can use our gesture recognition method while we measure facial expressions with the device. Although a directional touch on the face could be recognized using optical sensors on the device, there has the risk of misclassification as a facial expression change. Second, it allows a subtle interaction. The gesture is less obvious than a mid-air gesture or touching gesture to a device since rubbing a face is one of the physiological behaviors that can occur in daily life.

The contributions of our research are 1) Algorithm development for recognizing rubbing gestures on the face. We describe three phases for the recognition of the rubbing gesture: pre-processing,

Figure 2: The sensors on the smart eyewear

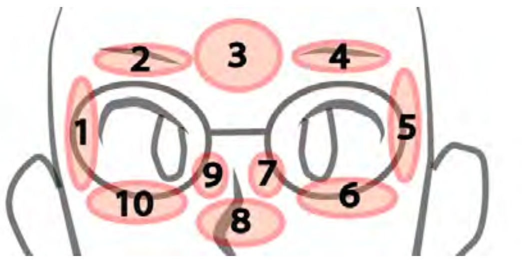

Figure 3: The input space for rubbing gestures

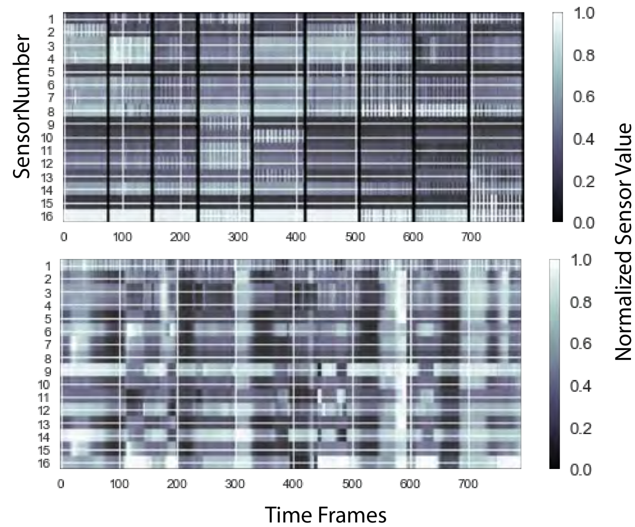

Figure 4: The sensor recordings of 1)Top: when rubbing various areas on facial surface 2)Bottom: when moving the facial muscles randomly

gesture detection and gesture classification. 2) Technical Evaluation of the gesture recognition. We run the study with five participants. we describe the results and the implications.

2 RELATED WORK

Researchers proposed the interaction methods to overcome the limited input space of the devices with a small factor. Harrison et al. presented Skinput that uses the skin on the arm and hand for an input surface [2]. SkinWatch enabled an interaction modality for a smart watch [7]. GestureSleeve is capable of detecting different gestures on touch-enabled sleeves [8]. It allowed the user to control a smartwatch without touching it.

Not only the input method on arms, many researchers explored the hand-to-face input method. Serrano et al. focused on interacting with head-worn displays [9]. They explored the design space for input gestures to faces. They used an infrared optical tracking system with six cameras and a proximity sensor to detect the gestures. Kikuchi et al. proposed EarTouch [3]. The device is a sensor-equipped earphone to enable an input to a computer by touching ear. They used photo reflective sensors to recognize the directional touch to the ear. A similar input method of directional touch on cheek surface is developed by Yamashita et al [11]. It made use of photo reflectors to detect the cheek deformation of the user. Hairware is a capacitive touch sensor integrated into a hair extension [10]. It detected a variety of touches to the hair extension for the triggering of different devices. Itchy Nose [4] detected various finger movements to a nose using EOG sensors in smart eyewear. These researches showed that the facial surface has a potential for input to a computer. In the gesture set of Itchy Nose, rubbing is included. We focus on rubbing gestures, and we consider various areas rather than the nose.

3 THE SYSTEM

We used the redesigned version of the smart eyewear developed by [5]. The device includes 16 photo reflective sensors (NJL5901AR-1 produced by New Japan Radio Co., Ltd). Figure 2 shows its layout. The photo reflective sensor consists of an infrared LED and a phototransistor. The sensors on the device collect the information about periodic skin deformation caused by rubbing gestures. Since the sensors are scattered on the front frame, the system can distinguish the rubbing gestures among different areas on the face. The input space considered in this work is shown in Figure 3.

3.1 The Algorithm

Firstly the user recorded the sensor data from the devices to see the difference between the changes caused by rubbing gestures and facial gestures (Figure 4). The top figure shows sensors values changed while the user rubbed various areas on the facial surface. The bottom figure displays when the user made facial gestures randomly. We normalized each sensor value to the range from 0 to 1 for a better visualization. The rubbing gestures caused the periodic sensor value changes in a short period while the facial gesture changed sensor values over longer time. Based on the characteristics, we propose a gesture recognition algorithm as follows. The algorithm consists of three phases; pre-processing, gesture detection, and gesture classification. We confirmed that our algorithm did not misdetect any rubbing gesture from the recording of facial gestures in Figure 4 despite the various change of sensor values. We implemented the algorithm in Python.

3.1.1 Pre-Processing. From the device, we acquire a 16-dimensional data sample per reading. The data sample is a subtraction of the sensor values when the infrared LEDs of the sensors are on and off. All the sensors are actuated at the same time. The sampling frequency is around 30 Hz. We apply a simple moving average of three sequences to the data samples. We use a sliding window of 40 samples (i.e. around 1.3 seconds). We run the algorithm every ten frames for the gesture detection.

3.1.2 Gesture Detection. We apply PCA to the data samples. PCA reduces the dimensions to one-dimensional time-series data. It includes the most dominant trend from the data samples. After normalizing the time-series data to fit in the range from 0 to 1, we apply peak detection algorithm (the python version of MATLAB peak detection algorithm). We used the threshold determination to detect the rubbing gestures. We count the number of upper and lower peaks. If there are more than five peaks in the data, we regard the data as the rubbing gesture to a certain area. It means the algorithm detects the gesture if there are the three rubbings in the time window. Since there is a slight chance that the noise from ambient light can also cause many tiny peaks when the user doesn't move and make any gesture, we excluded the data from the detection algorithm if the summation of absolute values of the derivative of the data samples in the window is less than a threshold.

3.1.3 Gesture Classification. If the rubbing gesture is detected, we create features from the data samples in the window. The features are 16-dimensional: the summation of the absolute values of the derivatives of each dimension of the data samples. We normalize the features so that the summation of the feature values is 1. Then, we apply a random forest classifier to the features in order to classify which area on the face is rubbed.

4 TECHNICAL EVALUATION

The goal of this evaluation is to investigate the accuracy of the proposed method. The five users (all of them are male in the 20s) participated in the study. For this evaluation, we developed the software for recording the sensor data samples of the rubbing gestures. We used Processing language. To avoid the influence of intense ambient light, we conducted the study in a quiet room far from windows. We used Python environment for the following analysis. We assumed the user makes the gestures only with a neutral face. Therefore, the participants held neutral face during the experiment.

Firstly, each participant sat on a chair in front of the laptop on the desk. They wore the prototype with an eyewear band strap for stability, and the observer started the software after explaining the experiment detail.

In the first stage, we recorded 100 gestures (Rubbing gestures to 10 areas X 10 times) each by each from every participant. The numbers of the input areas are corresponded to Figure 3. The order of the gestures the participants make is periodic (1 to 10). The software showed the figure of the input space and the number of the gesture area so as the participant can easily understand which area to rub. In order not to limit how to rub, we did not instruct which hand the user should use for the rubbing, the speed, and direction of the rubbing during the recordings. We divided each recording of the gesture into two phases. In the preparation phase (2000 ms), the user settled the rubbing position and started rubbing on a particular area. In the recording phase (3000 ms), the software recorded the sensor data samples while the user was rubbing.

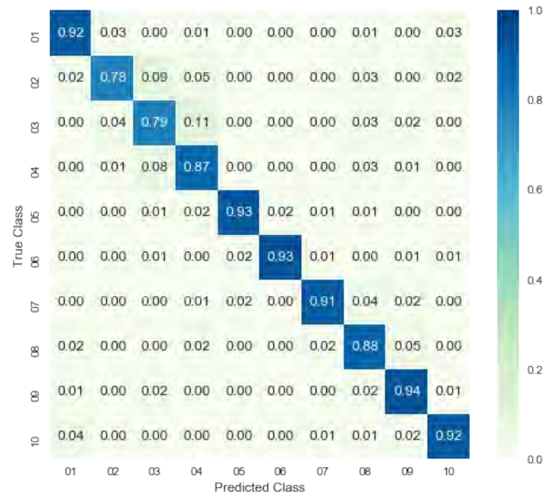

Figure 5: Confusion matrix of accuracy detecting rubbing gestures to ten different spaces

In the second stage, we recorded 20 gestures (10 areas X 2 times) from each participant. The recording was divided into four trials. The observer told each participant to make the gestures in specific ways (the number of the gesture area: 1-2-3-4-5, 6-7-8-9-10, 9-10-1-2-3, 4-5-6-7-8). In one time of the trial, the user made five gestures in 25 seconds. Every time the user made a gesture to a certain area, the user put their hands on their kees. This procedure created "no detection time" between the gestures. We gave 5 seconds break between the trials.

4.1 Results

First, we analyzed the data from the first experiment. We extracted first 80 data samples from the recording of each gesture. We split the samples into the two (first 40 samples and last 40 samples), i.e. we got two gesture data from one rubbing gesture. After excluding the outliers due to a lack of enough recording samples or a strong noise, we acquired the dataset of 937 gesture data from the recordings of five participants in total. We shuffled the dataset and applied five cross-validation method to the dataset. Overall, the accuracy of classifying ten gestures is 88.7% with user-independent datasets. Figure 5 shows the confusion matrix. Most of the false positive come from the adjacent area of the true positives. With user-dependent datasets, the average accuracy is 91.7%. Table 1 shows the result of each user. The accuracy of user A and B are around 80% that is about 10 % lower than the accuracy of the others. It is because some specific areas of the gestures of user A and B showed relatively less accuracy. For example, 38% of the gestures of the User A to the area 2 is predicted as the gesture to area 3. Also, 41% of the gestures of the User B to the area 4 is predicted as the gesture to area 3. In order to hold higher accuracy, we can use only the areas not close to each other. For example, the accuracy of classifying five areas(1,3,5,7,9) is 95.6% with user-independent datasets.

Regarding the dataset acquired in the second experiment, we merged the recording of each participant into one time-series data.

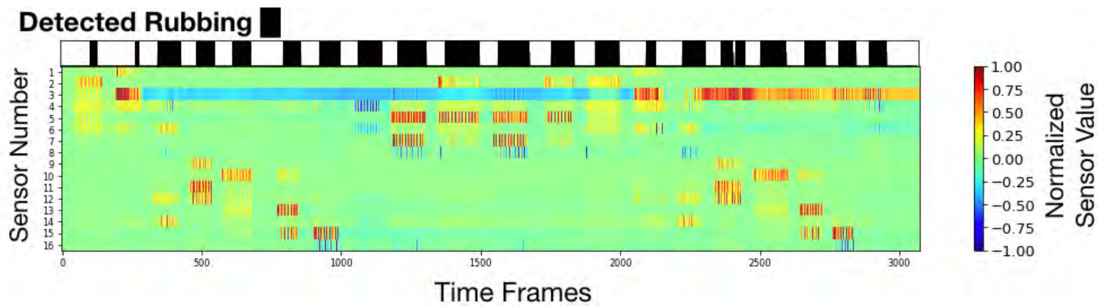

Figure 6: The time-series recordings: the heat map of the sensor data and the result of the gesture recognition of User D

Table 1: The accuracy of classifying 10 gestures with user-dependent datasets

User	A	B	C	D	E	Average
Accuracy(%)	82.2	82.7	96.7	98.5	98.4	91.1

Then, we applied our algorithm to the dataset. We trained the classifier individually using each participant's dataset from the first experiment. From each detected gesture, we have the sequences of the predicted results because the length of the gestures the users made is different every time. We used the dominant prediction result for estimating accuracy. We eliminated the dataset of User A since the strong noise was measured. As the device stopped for a short time while the user E used, the one gesture was not recorded. Table 2 illuminates the summary of recordings from the second experiment. The average accuracy of detecting the gesture is 97.5%. The F1-Score is 0.987. The average accuracy of classifying the gesture of the true positive is 91.0%.

Table 2: The summary of recording from the second stage. TP:True Positive, FP: False Positive, FN: False Negative.

User	B	C	D	E	ALL
Detection(TP)	20	20	20	18	78
Detection(FP)	1	0	0	0	1
Detection(FN)	0	0	0	1	1
Detection(F1-Score)	0.976	1.00	1.00	0.973	0.987
Classification Accuracy(%)	80.0	90.0	100	94.4	91.0

Figure 6 shows the heat map of the sensor data and the result of the gesture recognition of User D who showed the best performance among the participants. To make a better visualization, we subtract the initial values of the time-series data samples from all the data samples and normalize with the maximum absolute value for each sensor dimension. Although our system detected all the gestures, there was an unrecognized point in the series of the detected gesture. In those cases, we ignored the gap between the gestures if it was short enough.

5 DISCUSSION AND LIMITATION

We focused on the area around eyes for the input since rubbing on the area caused a big change of sensor values. However, rubbing on the cheek can also cause the skin deformation on the areas covered by the sensors. It means that we can use the larger area of the face. It can expand the input areas, but it also could reduce the accuracy of the gesture recognition.

There is a trade-off between the length of rubbing and robustness of the gesture recognition. We used the window size of 40 for the gesture recognition. If the size is too big, our algorithm may not detect the gestures properly. Also, rubbing gesture detection may take a long time. On the other hand, if the size is too small, it causes false detections. For example, when the user blinks continuously in short time, our system may recognize as a rubbing gesture.

Ambient light may inhibit from detecting gestures because intensive light makes sensor values saturated. Therefore, direct sunlight should be avoided when users use the proposed method.

An intentional rubbing gesture may cause some problems. If the user rubs too much on one certain area, this may lead to a rough skin. In addition, the user with a makeup may hesitate to make a rubbing gesture because the makeup comes off by the gesture.

6 CONCLUSION AND FUTURE WORK

We presented the input method by rubbing a facial surface using photo reflective sensors on smart eyewear. The input is subtle since rubbing a facial surface can occur as a physiological behavior in daily life. Although the system is developed for facial expression recognition, our system can recognize rubbing gestures to various areas independent of facial expression change. The accuracy of classifying rubbing gestures on 10 different areas of the face is 88.7% with user-independent datasets and 91.7% with user-dependent datasets. F1-Score of the gesture detection is 0.987.

In the future, we would like to explore the applications based on the recognized gesture patterns. We also consider the possibility of hand-to-face input by implicit movements. Implicit movements are an effective means to know the state of the person. According to [6], the hand-to-face gesture can reveal the mind of the person. it is useful to analyze daily user behaviors.

7 ACKNOWLEDGMENTS

This research is supported by JST CREST, the grant number: JP-MJCR14E1.

REFERENCES

[1] Andreas Bulling and Kai Kunze. 2016. Eyewear Computers for Human-computer Interaction. *interactions* 23, 3 (April 2016), 70–73. https://doi.org/10.1145/2912886

[2] Chris Harrison, Desney Tan, and Dan Morris. 2010. Skinput: Appropriating the Body As an Input Surface. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 453–462. https://doi.org/10.1145/1753326.1753394

[3] Takashi Kikuchi, Yuta Sugiura, Katsutoshi Masai, Maki Sugimoto, and Bruce H. Thomas. 2017. EarTouch: Turning the Ear into an Input Surface. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '17)*. ACM, New York, NY, USA, Article 27, 6 pages. https://doi.org/10.1145/3098279.3098538

[4] Juyoung Lee, Hui-Shyong Yeo, Murtaza Dhuliawala, Jedidiah Akano, Junichi Shimizu, Thad Starner, Aaron Quigley, Woontack Woo, and Kai Kunze. 2017. Itchy Nose: Discreet Gesture Interaction Using EOG Sensors in Smart Eyewear. In *Proceedings of the 2017 ACM International Symposium on Wearable Computers (ISWC '17)*. ACM, New York, NY, USA, 94–97. https://doi.org/10.1145/3123021.3123060

[5] 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

[6] Mark Nicas and Daniel Best. 2008. A Study Quantifying the Hand-to-Face Contact Rate and Its Potential Application to Predicting Respiratory Tract Infection. *Journal of Occupational and Environmental Hygiene* 5, 6 (2008), 347–352. https://doi.org/10.1080/15459620802003896 arXiv:http://dx.doi.org/10.1080/15459620802003896 PMID: 18357546.

[7] Masa Ogata and Michita Imai. 2015. SkinWatch: Skin Gesture Interaction for Smart Watch. In *Proceedings of the 6th Augmented Human International Conference (AH '15)*. ACM, New York, NY, USA, 21–24. https://doi.org/10.1145/2735711.2735830

[8] Stefan Schneegass and Alexandra Voit. 2016. GestureSleeve: Using Touch Sensitive Fabrics for Gestural Input on the Forearm for Controlling Smartwatches. In *Proceedings of the 2016 ACM International Symposium on Wearable Computers (ISWC '16)*. ACM, New York, NY, USA, 108–115. https://doi.org/10.1145/2971763.2971797

[9] Marcos Serrano, Barrett M. Ens, and Pourang P. Irani. 2014. Exploring the Use of Hand-to-face Input for Interacting with Head-worn Displays. In *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3181–3190. https://doi.org/10.1145/2556288.2556984

[10] Katia Vega, Marcio Cunha, and Hugo Fuks. 2015. Hairware: The Conscious Use of Unconscious Auto-contact Behaviors. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI '15)*. ACM, New York, NY, USA, 78–86. https://doi.org/10.1145/2678025.2701404

[11] 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