Automatic Keyboard Switching by Gaze Tracking

Xiao Sha[1], Hualin Zheng[1], Zhihang Han[1]

***Abstract*—Gaze tracking technology has been widely used in research area and human daily life, such as leveraging user interactions with computer [1], psychological analysis [2], medical diagnosis and etc. Many algorithms have been proposed to track gaze, however, every algorithm needs very large database sets to train. A new strategy with high accuracy in this paper and none data training required, is proposed to construct a prototype which can switch output of keyboard according to users gaze. In this strategy, two different tracking algorithms are incorporated, and each of them has high accuracy at some certain condition. System selects a suitable algorithm for specific situation automatically, which relies on consumer-grade hardware, less computation and achieves relative high accuracy. This allows system to switch output in real time with little latency. The experimental shows that our accuracy reaches 98%.**

***Index Terms*—Gaze Tracking, Bluetooth Keyboard Switching, High Accuracy New Strategy**

I. Introduction

A Great deal of computer vision researches has been completed to detect users movements and gestures [1]. Because the humans gaze positions are important information for many applications such as human-computer interaction (HCI) [3], market analysis [2], medical diagnosis [4]and many other applications. There have been many researches using gaze tracking in real life such as helping disabled people operate computer, control mouse [5] and help people to input password with eyes movements [6].

Variable techniques about eyes tracking has been reported in the literature. There are many techniques currently used to track eyes and many open source bookstores, such as OpenCV which contains a great number of reliable detection algorithms. The problem with current gaze tracking systems is that they all require expensive hardware or artificial environments such as head mounted cameras and chin rests [7].

This paper constructs a prototype to switch output of keyboard by gaze tracking. As inllustrated in Figure 1, the system includes three parts: a camera, a Bluetooth keyboard and a Raspberry Pi 3b+. Two LED substitutes the bluetooth keyboard here to make a simple demo, which just requires a slight change to apply on the bluetooth keyboard. A new strategy is proposed to improve gaze tracking accuracy. A robust gaze-tracking system is supposed to have stable performance in different environments. We take advantage of different algorithms to deal different conditions. We first use algorithm to track gaze when users face is in front of the camera. When users face has some angle to the camera, the first algorithms accuracy decrease. Then we use the second algorithm to track gaze.

[1]X. Sha HuaLing Zheng, Zhihang Han was with the Department of Electrical and Computer Engineering, StonyBrook University, StonyBrook, NY, 11794 USA e-mail: (see xiao.sha@stonybrook.edu).

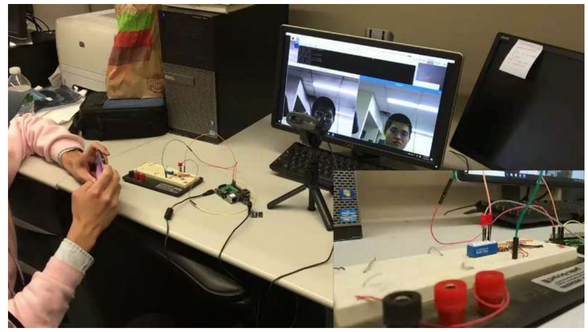

Fig. 1. The Actual Prototype of Automatic Keyboard Switching System

The outline of this paper is as follows. In Section 2 we first introduce some related work about people using gaze tracking to control mouse and operate computers. In Section 3 we develop a new system constituted by two algorithms and give the explanation of its working principle. We provide a performance evaluation in Section 4 considering different working situation, then conduct an analysis over variety of gestures. We conclude the paper with a summary of whole system and the future work which could make the system work better.

The main contribution of our research are as follows:

- We create a complete prototype of an efficient, easy-to-use, and inexpensive mobile gaze-tracking system. It computes gaze position in real time and switch the output of keyboard with the users gaze change.
- We propose a new strategy which incorporates two tracking algorithms. The system chooses a higher accuracy algorithm when people has different angle face to camera. The system does not need too much database to train the model.

II. Related work

Gaze-tracking systems have been used in many fields. Researchers have proposed many methods for their detection. And gaze tracking has been used into many applications. Many researches have worked on using gaze tracking for human-computer interaction. For instance, there has been a system working for disabled people to use eyes controlling mouse. [5] Their system contains four parts: face detection, eye detection, eyes tracking and mouse control. They use different color distribution to detect humans face. Then they utilize MLP to distinguish eyes regions and no-eye ones. For the gaze tracking part, the researchers iteratively shift each data point to the average of its neighboring data. After the system determines the center of eyes and sets it as the initial coordinates of mouse

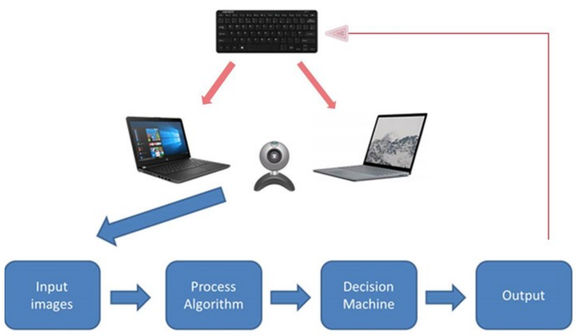

Fig. 2. The Prototype of System

and when the coordinates do not change over one second the system the process click motion.

III. DESIGN OVERVIEW

This paper is focused on the new strategy to improve the performance of right algorithm models for tracking gaze movement, which aims at accurately tracking gaze movement based on less calculation and classifier training. The strategy is based on the basic facts and the law when the human regularly moves head and gaze. By setting constraints and the threshold, it will help tracking system predict gaze movement much easier. This system composed of three critical modules: 1. System prototype, 2. Algorithm processing, 3. Decision machine.

A. System prototype

In this prototype, two laptops, a bluetooth keyboard, a camera and a Raspberry Pi 3b+ are utilized as shown in in Figure. 2. The camera takes photos at real time, and the images will send to the Raspberry Pi 3b+ which is set up with strategy program. OpenCV library and convex-hull method that built in the Raspberry pi 3b+,will be used to process the images, which consists of two different algorithm. A choice will be made in process algorithm according to the availability of input images data. The output of process algorithm is a rough judgment result indication the gaze's direction, that can't be sent to bluetooth keyboard directly. The decision machine will handle the rough judgment in the previous step by comparing it with the decision history tree to filter the noise from environment or error judgment result. A final decision, whether to transfer bluetooth keyboard signal or not, will be transferred to keyboard, by controlling the GPIO ports of Raspberry Pi 3b+.

B. Processing algorithm

A new strategy called Processing algorithm includes two parts, which are face and eye detection and eye centre detection. Face and eye detection aims at capturing the frame of human face and eye, which is based on face landmark library using 48 critical feature points in the face to reconstruct the face and eye frame. The eye centre detection is finished after the face and eye frame reconstruction, which means the eye centre localization is performed in the frame of eye after the first step. Several methods is employed to increase the accuracy of eye centre detection.

1) Face and eye segmentation: Face and eye segmentation are important prerequisite step required for successful eye gaze tracking system. The more accurate segmentation, the more likely recognition is to be succeed. The video captured using a web camera is composed. After that, the RGB color space into another color space which is not sensitive to light variation. In order to make the system achieve real-time processing and adapt to most of the environments, HSV [8]color space is being used to extract skin color regions. These skin color regions are converted to binary images in which morphological operations including erosion and dilation are performed. Noises are eliminated by erosion and the contour of skin color area is smoothed by dilation. A OpenCv library based on Viola-Jones algorithm and a classifier will be used to mark the face and eye region [9].

2) Eye centre localization: After eye region marked, the relationship will be described between a possible center and the orientations of all image gradients. Another classifier based on Kothari and Mitchells method [10] can be used to find the iris position. There are two scenarios to infer eye direction. First, small movement. When the head is not moving, the system will use iris direction to infer users' eye direction. Second, large movement. When the user's head is moving in another direction, the system will automatically use the distance between eye edge region and nose region to infer eye direction. These two scenarios based on the new strategy make the whole eye tracking process more efficient.

3) New strategy: The core idea of the strategy is to maximize the efficiency of different algorithm based on certain circumstances. The system will use two algorithms to realize the gaze tracking. Algorithm A can track the face direction, but it can't really tracking gaze. Algorithm B is the real gazing tracking method, but it only works properly when users face straight looking at the camera. The strategy can combine the two algorithms and realize the true gaze tracking. When users face direction changed, the system will automatically choose algorithm A to predict the gaze direction. When users faced straight to the camera, the algorithm B will start to track users gaze. Two threshold values will set to judge if the user is still looking the screen or not. The first threshold is an angle value which will set to judge whether the users gaze is looking the maximum edge of left or right screen. The second threshold is time value. To eliminate the possibility that the user may just have a glance to somewhere else, the constraint time set at 0.34 seconds. If the users gaze angle keep over the maximum angle value over 0.34 seconds, the program will send a signal to switch the Bluetooth keyboard to another device (if the user looking at the device over 3 seconds). When the user not looking at any device, the Bluetooth keyboard will still keep connecting to the same device.

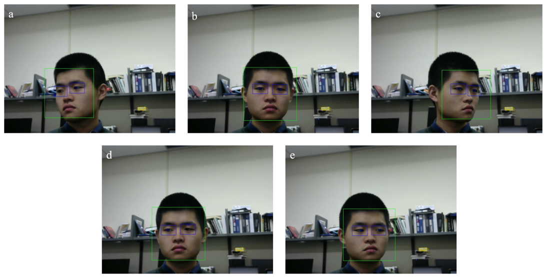

Fig. 3. Five Scenarios of Eyes direction. (a)Head turning left (b)Looking straight (c)Head turning right (d) Eyes turning left (e)Eyes turning right

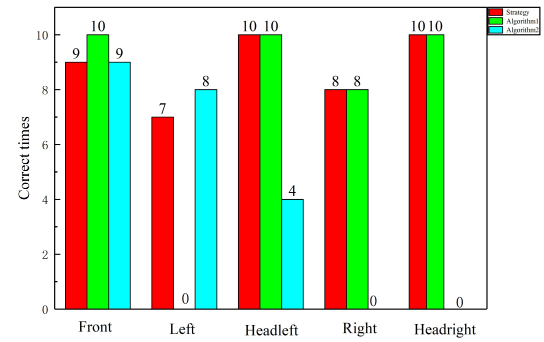

Fig. 4. The Accuracy of Different System

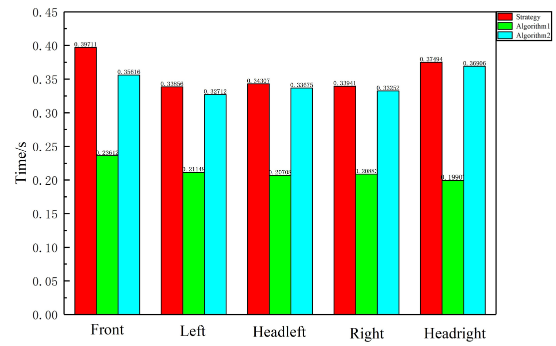

Fig. 5. The Time Evaluation of Different System

C. Decision machine

The decision machine keeps processing two sets of data. One is previous decision, the other is a new decision. The program based on OpenCV classifier will send new decision con- tinuously. If the user eye direction changed, the new decision will replace the old decision and output transfer signal. If the users eye direction not changed, the old decision will keep the same. This simple mechanism machine keeps the decision making efficient.

IV. Evaluation

The system is tested under two different situations: static situation and dynamic situation. At the static situation, we detect the accuracy of the system by inputting an image. When we just input an image to the system, our aim is to detect the real accuracy of the system without the camera low quality affect. Then we run the whole system in real time to detect the systems speed and accuracy. This time we operate the system in real time and our purpose is to obtain the users real experiences, which is related to the latency and accuracy of the system, with our limited hardware.

A. Static evaluation

We compared our strategy with other systems which only utilizes one algorithm. We input five different kinds of pictures of users normal gesture, which includes (a) looking straight at the screen, (b) looking at the left side with turning eyeballs, (c) looking at the right side with turning eyeballs, (d) looking at the left with turning head and (f) looking at right with turning head as shown in Figure. 3. Each type of pictures is 10 images set, which is captured by the same camera. Then we obtained the whole processing time and accuracy which show at Figure. 4 and Figure. 5.

The system accuracy is obviously higher than the other two systems. Since each algorithm has its own advantage for specific situations, our strategic task is choosing a better one. The result shows that during picture processes of system, when user turned their head at a large angle, the system with second single algorithm has a much higher accuracy for this situation. While the system with the first single algorithm exhibits high accuracy when user only moves their pupils to look at different screens. The reason of these appearances is that the first algorithm judges that the orbit of the user is still in the middle of the face when the users face has a large angle to the camera. This judgment is obviously outside of the natural science. Hence the accuracy decreases quickly. Here is another reason for the accuracy reduce. When people face is in front of the camera, the algorithm mainly catches the distance from the pupil to confuse which needs much coordinates to mark and much data to train.

The time consumption of our system is slightly more than any other two systems. This phenomenon is easy to understand. That's because at first the system needs time to initialize and when our strategic processes, the system primarily detects the angle of facing then select an appropriate algorithm which consumes a part of the time. However, our speed still can satisfy users' requirement, taking consideration of that human response time to visual stimulation is about 150 ms [11]. In this case, our processing time is quick enough that can't be felt the pause and transition, which means it will not affect users experience. The main influence factor of our system is the speed of image capture, which is real a hardware and time consumption. A long operation camera generates the plentiful heat and slow down the speed of CPU in Raspberry Pi 3b+.

B. Dynamic evaluation

We let the system work for 30 seconds. During the using time, the system took on hundred available images. Then we collect the accuracy and time consumption data. The system performs a high accuracy in real time. The accuracy of these images judgement is 98% which means that the system only makes two mistakes in one hundred images. The mistakes would not affect the experience of the user. Because the camera collects images continuously and decision machine will compare two consecutive judgements of the system. If a wrong judgement appears it will be correct after the next image is processed. The average time consumption of each decision is 0.283996s. The time consumption in dynamic situation is less than the static situation which can improve the experience of users much. Our system is qualified to track gaze and complete the switch keyboard output contrast with other single algorithm systems.

V. Conclusion

We have developed a prototype that is able to judge which side is human watching and switch the output of Bluetooth keyboard to the expected screen. This system consists of several parts: a camera, a Raspberry Pi and a Bluetooth keyboard which has reduced the cost of assistive device.

In order to improve the accuracy of the system we adopt a new strategy. Our strategy is incorporating two kinds of tracking algorithm. We deal different kinds of situation with relatively high accuracy algorithm. Hence our system has high accuracy. The experimental results show that our system with this strategy has improved much accuracy compared with other single algorithm approach.

For the future work, the raspberry pi and program can be integrated into the Bluetooth keyboard. Gaze tracking keyboard can bring users more convenience. And this papers new strategy not only limited in for the gaze tracking project. It can be applied to any other tracking algorithm based on different situation.

References

[1] M. H. Khosravi and R. Safabakhsh, "Human eye sclera detection and tracking using a modified time-adaptive self-organizing map," *Pattern Recognition*, vol. 41, no. 8, pp. 2571–2593, 2008.

[2] M. Wedel and R. Pieters, *Looking at Vision*. Routledge Abingdon, 2014.

[3] A. Lopez-Basterretxea, A. Mendez-Zorrilla, and B. Garcia-Zapirain, "Eye/head tracking technology to improve hci with ipad applications," *Sensors*, vol. 15, no. 2, pp. 2244–2264, 2015.

[4] F. Galgani, Y. Sun, P. L. Lanzi, and J. Leigh, "Automatic analysis of eye tracking data for medical diagnosis," in *Computational Intelligence and Data Mining, 2009. CIDM'09. IEEE Symposium on*. IEEE, 2009, pp. 195–202.

[5] M. Betke, J. Gips, and P. Fleming, "The camera mouse: Visual tracking of body features to provide computer access for people with severe disabilities," *IEEE Transactions on neural systems and Rehabilitation Engineering*, vol. 10, no. 1, pp. 1–10, 2002.

[6] Z. Lin, K. Cheng, Z. Tai, and Y. Zheng, "The Eye Control Password Keyboard Based on Gaze Tracking," in *PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON ADVANCES IN MECHANICAL ENGINEERING AND INDUSTRIAL INFORMATICS*, ser. AER-Advances in Engineering Research, Liu, M and Zhang, X, Ed., vol. 15, 2015, pp. 1413–1419.

[7] Y. Matsumoto and A. Zelinsky, "An algorithm for real-time stereo vision implementation of head pose and gaze direction measurement," in *fg*. IEEE, 2000, p. 499.

[8] H.-D. Yang, S. Sclaroff, and S.-W. Lee, "Sign language spotting with a threshold model based on conditional random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 7, pp. 1264–1277, 2009.

[9] http://picoledelimao.github.io/blog/2017/01/28/eyeball-tracking-for-mouse-control-in-opencv.

[10] R. Kothari and J. L. Mitchell, "Detection of eye locations in unconstrained visual images," in *Image Processing, 1996. Proceedings., International Conference on*, vol. 3. IEEE, 1996, pp. 519–522.

[11] M. Rehman, M. Abdul Mujeebu, Y. Cheng, and A. Abu Al Aish, "A microcontroller-based measurement system for human response to visual and hearing stimulations," *Experimental Techniques*, vol. 36, no. 6, pp. 5–12, 2012.