Low Cost Real-time Eye Tracking System for Motorsports

Yuanjie Xia
Microelectronics Lab
James Watt School of Engineering
Glasgow, UK
2289015x@student.gla.ac.uk

Andrew Lunardi
Glasgow School of Art
Glasgow, UK
2245578l@student.gla.ac.uk

Hadi Heidari
Microelectronics Lab
James Watt School of Engineering
Glasgow, UK
Hadi.Heidari@glasgow.ac.uk

Rami Ghannam
Engineering Design Research Group
James Watt School of Engineering
Glasgow, UK
Rami.Ghannam@glasgow.ac.uk

***Abstract*—Eye-tracking technology can be used to determine reaction time and cognitive abilities of a subject based on response time to visual changes and analysis of focus points and duration. In the context of motorsports, quantifying driver visual data adds to the wealth of information required for the marginal performance gains that make up motorsport competition. Specifically, the gaze of drivers can be used to analyze and assess driving abilities and improve driving performance. Despite eye trackers being perfectly suited for this purpose, their high cost may impede their penetration in the motorsports industry. Therefore, we demonstrate a low-cost wearable real-time eye tracking system for motor sport applications. This system was mounted on a racing helmet to monitor the driver's gaze in real-time. A dual-camera system was used to capture both eye image and front view. The eye image was processed using OpenCV-Python to determine the gaze coordinates. Moreover, the gaze position was mapped into the front view to determine the wearer's line of sight. The eye tracking helmet was tested and demonstrated good accuracy, where the deviation angle was 3.8 degrees. These promising results further add to the growing body of knowledge in low cost eye trackers.**

***Index Terms*—Eye tracking, Image processing, Wearable system, Sensors**

I. Introduction

Motorsport quickly evolved into one of the most technologically advanced sporting categories, and as such has developed into an extremely data driven industry. For example, a Formula One (F1) racing car has over 150 sensors on board which can produce up to 100 GB of data in car telemetry over the course of a race weekend [1]. Therefore, the value of a numerical data is clearly the route to gaining a competitive edge over rivals. Modern motorsport cars are extremely fast, and as such require incredibly strong reactions from a driver, who is taking in visual information at a much faster rate than ever before. Thus, a reliable eye tracking system is required to capture and assess drivers' driving skill and performance. The key qualitative information acquired from eye tracking technology is the ability to directly visualise what drew the subject's attention, and how they processed the visual information they were receiving [2]. Additionally, by putting numerical data to fixation duration, engineers could better determine how a driver is focusing on certain obstacles, or how they take in the flow of the track, and make suggestions on how this could be improved.

Fig. 1. A schematic diagram of the eye tracking racing helmet

There are a wide variety of techniques for eye tracking, with varying effectiveness and practicality. Video oculography is the most popular eye tracking method [3]. Camera sensors are placed near eyes to capture real-time image and then obtain the gaze position. This method is widely used in wearable eye tracking system design [4], [5]. Apart from the camera-based eye tracking system, researchers also purposed an eye tracking system using magnetic sensors. A contact lens with magnet was worn by user. The magnetic field would change as eye is moving, which could be detected by sensors integrated on the frame of glass [6], [7]. Electro-oculography is one of the older methods, which is mainly used in clinical applications. Eye movements change the orientation of the electrical fields, which are measured using electrodes fixed to the subject's skin around the eyes themselves [8], [9]. Infrared-occulography is another way to obtain gaze position [10]. An infrared (IR) emitter and sensor are mounted near the eye. The IR light could be reflected by the surface of eye and detected by the sensor. The intensity of IR light changes as the eye moves [11], [12]. Despite such IR systems consuming less power than video-based eye tracking systems, a camera based system was designed, since IR light from the surrounding environment may interfere with the systems accuracy [11].

Therefore, a camera-based eye tracking system was designed and mounted on a racing helmet, as demonstrated in fig. 1. Images were captured using camera modules and transmitted to a terminal device via local area network (LAN). Here, a computer was used as the terminal device, since it is capable of processing images in real-time. In the following sections we describe our approach in designing and developing the helmet, as well as the results obtained from our design.

II. Methodology

A. Hardware Design

The eye tracking system needed to be wearable, wireless and consume a low power amount of power. Therefore, the captured image data should be wirelessly transmitted to the computer. In comparison to Bluetooth, Wifi was used for image transmission since it enables higher data transmission speeds. Consequently, the ESP32 module was chosen since it is capable of both acquiring data from camera sensor and transmitting the data via Wifi. Regarding the camera sensor, two OV2640 cameras were chosen for its small volume, which have maximum 1632 x 1232 resolution. Moreover, each camera module has an IR filter to prevent IR light from the environment influencing the image quality. However, the IR filter was removed on the eye tracking camera, so that an IR LED could be used to illuminate the eye. In this case, the eye tracking camera was able to capture clear eye image in dark environment. Both cameras and IR LED were driven by ESP32 module. In addition, the ESP32 module was integrated with both WiFi and Bluetooth modules. Although the Bluetooth technology was not used in the eye tracker design. However, it enhances the practicality of the prototype by having the ability to wirelessly communicate with a computer and mobile phone simultaneously. For example, ESP32 can send log data to mobile phone or computer via Bluetooth, which will benefit both hardware developer and user.

The boards and cameras were powered by 5 V power bank to ensure the cameras to operate consistently and effectively. All electronic components were integrated into a racing helmet, as is shown in fig. 2a. In addition, a PCB was designed to integrate ESP32 surface mounted module (SMD), driver circuit and power management circuit. The PCB design is demonstrated in fig. 2b. The area of PCB board is $23mm \times 35mm$, which is ideal for wearable applications. Each PCB was designed to drive one OV2640 camera sensor and one ESP32 module, so that two boards were required in the system.

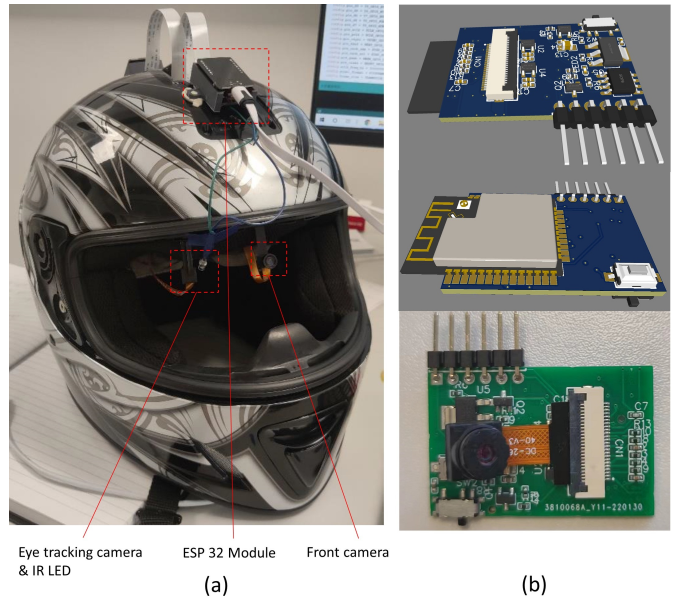

Fig. 2. (a) The appearance of eye tracking helmet (b) PCB for ESP32 and camera module

B. Eye tracking algorithm

The flow charts in fig. 3 explain how the system works. The coding for ESP32 was accomplished by using Arduino IDE. The underlying coding logic is demonstrated in fig. 3a. The first step is to initialize the GPIO ports. Secondly, the ESP32 module needs to establish connection with camera sensor. After that, the parameters such as resolution and contrast should be setup based on demand. For example, the resolution for eye tracking camera was set as 240×240 since it didn't require high quality to track eye movement, whereas the front view camera is set as 640×480. Although higher resolution is available, it was not selected since high resolution requires high transmission speed. High resolution is more suitable for picture mode rather than video mode. Therefore, 640×480 was chosen as the resolution to reduce image delay and ensure image quality. The next step is to setup the web server. After image is captured by camera sensor, it will be uploaded to a LAN address. The image data could be therefore accessed by terminal devices as long as ESP32 and terminal device connected to the same network. Since there are two cameras, the captured images would be sent to two different LAN address respectively.

Regarding the terminal device, the underlying coding logic is demonstrated in fig. 3b. Noticeably, a computer was used as terminal device since it is capable to process image in real-time. Firstly, the computer was connected to the same network as ESP32 modules. Here, the whole system was connected to hotspot from a mobile phone. After that, the computer sent request to two LAN address and fetched image data array simultaneously. It is noticeable that multi-thread method was applied to reduce network delay. If the terminal device request image data from two camera web server in sequence, the response time would be double and the video would lag. To be specific, each thread has a command to request hex image data and store it into a queue array, and two threads operate simultaneously. After that, the image data was processed by OpenCV-Python. The hex image data in the queue was decoded separately and displayed in sequence. The eye images were processed by median filter to remove noise points. Subsequently, the eye images were transformed into HSV figure, which has H, S and V three channels. The threshold value of each channel should be adjusted manually to make a mask for eye image. Additionally, dilate and erode algorithm was also applied to optimize the mask. After these process, only the area of iris and pupil was marked into white and all other area is black. The coordinate of gaze point could be therefore obtained by calculating the contours of the mask. The following step is calibration, where four point of gaze coordinate should be mapping with four relative coordinate in front view. With four coordinates on each image, the projection transformation matrix was calculated, and therefore other coordinates of eye image could be matched with front view. After all these steps, the eye tracking helmet is able to identify where people is looking at in real time. In following section, the gaze tracking result will be demonstrated and discussed.

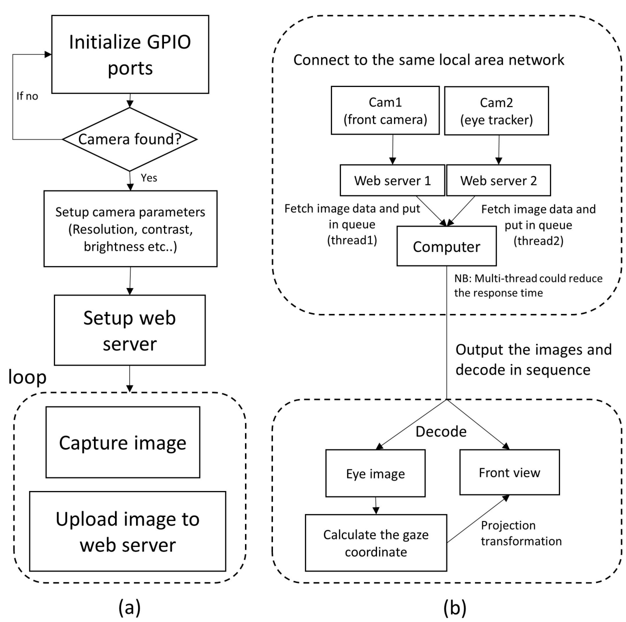

Fig. 3. Figure 3 demonstrates underlying coding logic for (a) ESP32 module and (b) terminal device (computer)

III. Result and Discussion

After assembling the helmet, the whole system was tested in a private car park to demonstrate that driver gaze can be traced, as shown in fig. 4a. During the test, both ESP32 modules and the computer were connected to the hotspot of a mobile phone, so that they could share image data using LAN addresses. Our results are demonstrated in fig. 4 b-d for driver gaze in three different directions. After calibration, the projection transformation matrix was obtained. Here, the thresholds of HSV channels were adjusted manually to obtain the mask of iris and pupil area. After that, the mask was transformed by the projection matrix as was demonstrated in the result. Thus, the gaze coordinate was calculated by finding the center point of white area. The gaze point was marked in front view by a large green crossing. Subsequently, the accuracy of the eye tracker was evaluated using the 'deviation angle'. Here, we invited users to sit approximately 0.5 meters away from a screen and asked them to stare at 9 different points in order. The error distance D_{error} could be calculated by estimated gaze coordinates and real gaze coordinates. The deviation angle could be obtained using the following:

$$\text{Deviation Angle} = tan^{-1}(\frac{D_{\text{error}}}{D_{\text{user screen}}})$$

The deviation angle was calculated using 5 groups of data, which were around 3.8 degrees. In comparison with previous low-cost eye trackers in the literature, our eye tracker has relatively good accuracy for a single camera eye tracking system [3].

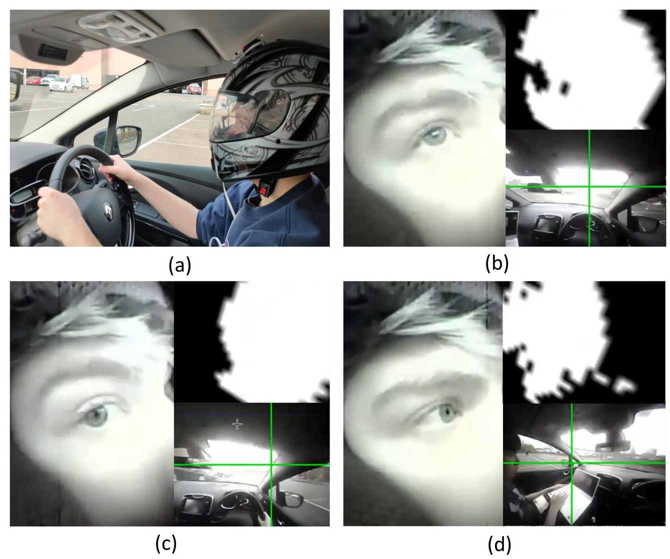

Fig. 4. (a) A driver wearing the eye tracking helmet (b-d) The driver is looking at center, right side and left wind mirror respectively. In each of the three figures, the image on the left is the raw eye image. The image on top right is the mask after projection transformation. The image at right bottom is the final result.

Fig. 4 showcases the gaze tracking results when the driver was driving in the car park. The result was accurate when the driver was looking in three different directions. Additionally, the driving performance could be evaluated and analyzed based on the real-time eye tracking result. For example, the drivers' safety consciousness could be assessed by observing if the driver uses wind mirror and backing mirror properly while turning and reversing. In future, we could improve our design by including Liquid Crystal Displays (LCDs) or other media to control the amount of light entering the eye, as suggested by [13]. Therefore, different liquid crystal alignment methods and electrical modulation methods could be investigated for that purpose [14], [15].

IV. Conclusion

Data acquiring and processing are highly demanded in motorsport area. Many sensors have been integrated into racing car to ensure drivers' safety and improve their driving skills. Eye tracking is one of the most essential way to evaluate drivers' driving abilities, including safety awareness and level of concentration. Therefore, the aim of this project is to design an eye tracking system for motorsport applications. In this paper, an eye tracking helmet is purposed to track the gaze of driver. A dual-camera system was utilized to capture both front view and eye image. These two camera sensor was driven by ESP32 Wifi module and transmitted hex image data array to a computer via LAN. Subsequently, the image data was decoded and processed by OpenCV-Python. The gaze position coordinate could be calculated and matched with the front view. After that, the prototype was tested in a private car park and able to monitor drivers' gaze in real-time.

Comparing with other eye tracking systems, this eye tracking system is low cost, small and low delay. The overall cost of this eye tracking system is less than 30 pounds, which is much lower than commercial eye trackers such as tobii and intel ones. The size of the driver board is $23mm \times 35mm$, which is suitable for wearable applications. This size is smaller comparing with previous designs [16], [17]. Additionally, the internet delay is reduced since multi-tread algorithm is applied, thus the internet delay is reduced by half. Apart from that, the deviation angle is 3.8 degrees, which is a relatively good accuracy among similar designs [3].

In current stage, the quality and fluency of the video image is limited by network bandwidth, which could be improved by connecting the whole system with a more stable and wider bandwidth network rather than hotspot from mobile phone. In addition to that, more advanced eye tracking algorithm could be developed improve the accuracy and recognition speed. In addition to traditional Human-computer interaction methods, for instance, hand gesture [18], this eye tracking system could also be integrated in IoT applications, which can improve and enhance the interaction of futuristic IoT designs.

References

[1] K. Collantine, "How one f1 team generates more data than it can handle," Feb 2018. [Online]. Available: https://www.racefans.net/2018/02/14/big-data-how-one-f1-team-creates-more-information-than-it-can-handle/

[2] A. T. Duchowski, "A breadth-first survey of eye-tracking applications," *Behavior Research Methods, Instruments, & Computers*, vol. 34, no. 4, pp. 455–470, 2002.

[3] A. J. Larrazabal, C. G. Cena, and C. E. Martínez, "Video-oculography eye tracking towards clinical applications: A review," *Computers in biology and medicine*, vol. 108, pp. 57–66, 2019.

[4] S. Y. Gwon, C. W. Cho, H. C. Lee, W. O. Lee, and K. R. Park, "Gaze tracking system for user wearing glasses," *Sensors*, vol. 14, no. 2, pp. 2110–2134, 2014.

[5] M. Mehrubeoglu, L. M. Pham, H. T. Le, R. Muddu, and D. Ryu, "Real-time eye tracking using a smart camera," in *2011 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)*. IEEE, 2011, pp. 1–7.

[6] A. Tanwear, X. Liang, Y. Liu, A. Vuckovic, R. Ghannam, T. Böhnert, E. Paz, P. P. Freitas, R. Ferreira, and H. Heidari, "Spintronic sensors based on magnetic tunnel junctions for wireless eye movement gesture control," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 14, no. 6, pp. 1299–1310, 2020.

[7] A. Tanwear, X. Liang, E. Paz, T. Bohnert, R. Ghannam, R. Ferreira, and H. Heidari, "Spintronic eyeblink gesture sensor with wearable interface system," *IEEE Transactions on Biomedical Circuits and Systems*, pp. 1–14, 2022.

[8] L. Chamberlain, "Eye tracking methodology; theory and practice," *Qualitative Market Research: An International Journal*, 2007.

[9] I. Käthner, A. Kübler, and S. Halder, "Comparison of eye tracking, electrooculography and an auditory brain-computer interface for binary communication: a case study with a participant in the locked-in state," *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, pp. 1–11, 2015.

[10] R. Li, E. Whitmire, M. Stengel, B. Boudaoud, J. Kautz, D. Luebke, S. Patel, and K. Akşit, "Optical gaze tracking with spatially-sparse single-pixel detectors," in *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2020, pp. 117–126.

[11] H. Chennamma and X. Yuan, "A survey on eye-gaze tracking techniques," *arXiv preprint arXiv:1312.6410*, 2013.

[12] Y. Xia, M. Khamis, F. A. Fernandez, H. Heidari, H. Butt, Z. Ahmed, T. Wilkinson, and R. Ghannam, "State-of-the-art in smart contact lenses for human machine interaction," *arXiv preprint arXiv:2112.10905*, 2021.

[13] Q. Zhang, Y. Huang, G. Chernyshov, J. Li, Y. S. Pai, and K. Kunze, "GazeSync: Eye movement transfer using an optical eye tracker and monochrome liquid crystal displays," in *27th International Conference on Intelligent User Interfaces*. ACM, mar 2022.

[14] R. James, E. Willman, R. Ghannam, J. Beeckman, and F. A. Fernández, "Hydrodynamics of fringing-field induced defects in nematic liquid crystals," *Journal of Applied Physics*, vol. 130, no. 13, p. 134701, oct 2021.

[15] R. Ghannam, Y. Xia, D. Shen, F. A. Fernandez, H. Heidari, and V. A. L. Roy, "Reconfigurable surfaces using fringing electric fields from nanostructured electrodes in nematic liquid crystals," *Advanced Theory and Simulations*, vol. 4, no. 7, p. 2100058, may 2021.

[16] K. Harezlak and P. Kasprowski, "Application of eye tracking in medicine: A survey, research issues and challenges," *Computerized Medical Imaging and Graphics*, vol. 65, pp. 176–190, 2018.

[17] A. F. Klaib, N. O. Alsrehin, W. Y. Melhem, H. O. Bashtawi, and A. A. Magableh, "Eye tracking algorithms, techniques, tools, and applications with an emphasis on machine learning and internet of things technologies," *Expert Systems with Applications*, vol. 166, p. 114037, 2021.

[18] Y. Xia, H. Heidari, and R. Ghannam, "Smart wristband for gesture recognition," in *2020 International Conference on UK-China Emerging Technologies (UCET)*. IEEE, 2020, pp. 1–4.