Raspberry Pi-based mobile system of the pupil size evaluation

A Ya Suranov
Department of Radiophysics and Theoretical Physics, Altai State University, 61 Leninave., Barnaul, 656049, Russia
E-mail: alexander.suranov@yandex.ru

Abstract. The article considers development of a human pupil measuring system on the Raspberry platform. The system is aimed at evaluating the variations of the human pupil diameter or area in the process of watching test images or video recordings. To reduce the interference from the eye surface the camera uses a band-stop color glass filter PS 13. In order to increase the pupil image contrast, IR LED backlighting of the eye is implemented. To provide the mobility of the system, battery power of the single-board computer was used while the registered image and the measurement results were transmitted via a Wi-Fi channel. The video camera and the single-board computer Raspberry Pi 4 with the battery bay are attached to the head-mounted flexible belt. The article gives the operation duration evaluation of the battery-supplied system.

1. Introduction

Studies of the eye response to light stimuli is widely applied in medical practice to diagnose various diseases [1-3]. Portable automatic pupillometers that provide optical isolation of the eye area from the external illumination are generally used (figure 1).

Figure 1. The design (a) and the application example (b) of a portable automatic pupillometer.

However, lately pupillometer techniques are becoming popular in terms of determining human psycho-emotional response to visually perceived test images or video records that are shown in
customary environment [4]. This environment excludes optical isolation of the eye area thus complicating image processing due to the reflection of the display with the test image and that of the surroundings which appear on the eye surface. Besides, when registered by standard color video cameras the pupil image may be of low contrast against the dark iris.

All that requires suppression of the visible light spectrum on the camera input and the use of IR radiation for backlighting which increases the contrast of the pupil image. This kind of lighting is easy to provide by means of LEDs. Thus when using color video cameras to register eye images one should remove the cut off IR filter and install a filter suppressing visible light spectrum and, besides, provide the eye IR backlighting.

The above-mentioned ideas were implemented on the basis of ASI120MC-S color video camera and were described in the article [5]. The camera with a telephoto lens was mounted on a tripod, and the testee was to be located at a defined distance from it. Meanwhile his head position could not be changed much during the experiment. The image from the camera was sent via an USB channel to the personal computer. The drawbacks of the described system are as follows: the heavy weight of the registration system and the cable connection to the computer for power and image transmission that limits the mobility of the testee.

However, works on pupillometer studies [6, 7] describe solutions on mounting the registration systems directly on the testee's head that solves the problem of the relative moving of the object and the camera. One can find more solutions on such systems in the works devoted to eye tracing systems [8, 9]. The examples of such systems are given in figure 2.

![Figure 2. The examples of head-mounted eye tracing systems.](image1)

To provide mobility one may use single-board computers Raspberry Pi and video cameras of the same family which are popular nowadays [10]. Nevertheless the application of these computers requires developing the corresponding software which should perform the real-time eye image processing and wirelessly transmit the results of processing.

The objective of the work is to develop hardware part and software for the head-mounted system of registering and processing the human pupil image.

2. The mobile system of registering and processing the human pupil image

The design of the developed head-mounted system model is shown in figure 3. One can see that a single-board computer Raspberry Pi 4 was fixed on a flexible rim and mounted on the head, while a video camera was fixed on the side of the rim. The camera was situated 3 – 4 cm lower than the eye gaze in order not to disturb the observation of the test image or video recording. IR LEDs were fixed on the sides of the lens to prevent the reflection from getting in the pupil image area. To provide its full autonomy the registration and processing system received power from batteries. The eye reference image and the processing results were transmitted to the PC by a wireless channel (Wi-Fi).
Figure 3. The method of attaching the registration and processing system to the head.

The eye image received in such configuration is shown in figure 4. Due to the vertical inclination of the camera the eye image becomes elliptically shaped that, nevertheless, does not prevent measuring relative variations of its area while the diameter can be measured on the horizontal axis.

Figure 4. Eye image.

Figure 5 shows image brightness histograms via three channels demonstrating that each channel has a distinctly localized peak in the area of lower brightness corresponding to the pupil area. The difference in brightness range values in the channels is apparently due to the notch filter partly admitting visible radiation in the blue and red area and more blocking in the green area.

Figure 5. Brightness histograms of camera channels.

It is possible to choose one of the channels e.g. the red one for eye image processing. To select the pupil it is advisable to binarize the eye image by a definite threshold, define the eye center, and calculate the number of "dark" elements on the line passing through the center. It is preferable to choose the
binarization threshold according to the above-given histogram in the central part of the "gap" between the first peak conditioned by the pupil and the following peaks connected with the brightness of the iris and other parts of the eye.

Besides real-time processing of the eye image in order to determine eye parameters, a single-board computer should be able to fulfill a number of other functions. In particular, before the beginning of the test it is desirable to see the eye image to set the optical part as well as in the test process in order to monitor its course. Herewith the image resolution may be lower than that of the camera. It is also advisable to see the brightness histogram via one of the channels to set the image exposure. While performing each operation it is necessary to monitor its duration.

3. The development of the battery-supplied system

To provide the autonomous operation of the mobile system of registering and processing the human pupil image it is necessary to provide it with battery power supply. The nominal supply voltage of the single-board computer Raspberry Pi should be at least 5 V. When using lithium-ion batteries capable of generating current sufficient for Raspberry Pi 4 two batteries with the nominal voltage 3.7 V are required. In order to reduce the output voltage to 5 V it is required to include a voltage stabilizer and a battery protection controller in the supply system. In this supply system (Fig. 6) we used a voltage stabilizer SCV0050-5V-3A and a controller FDC-2S-2.

![Figure 6. The supply circuit of the mobile system of eye pupil variation evaluation.](image)

To determine the time of the mobile system continuous operation it is required to study the relationship between the battery voltage and time at the given consumption current. Herewith the consumption current of the Raspberry Pi 4 depends on its computational load. Measurement using Digit Multimeter ABM-4141 [11] showed that the average consumption current for a single-board computer after switching on (at the nominal computational workload) equals 0.6 A and increases to 0.72 A when the video stream recording program is launched. The latter provides increased computational load for the computer. The curves of the battery discharge at the nominal and increased load are given in figure 7 and marked accordingly with numbers 1 and 2. On the x-axis the value of the discharge measured in amp hours was set. Provided that the maximum voltage limit equals 6.5 V the diagram shows that the operation time of the mobile system lies within the range of 1-1.5 hours depending on the computational load.
Figure 7. The curves of the battery discharge under the nominal (1) and increased (2) load.

4. The selection of an appropriate operating system and software development

The Raspberry Pi platform is capable of working under various operating systems [12]. To perform real-time image registering and processing tasks it may be feasible to use the most appropriate operating system. When comparing the characteristics of the operating systems proposed for Raspberry Pi ROS (Robot Operation System) [13] was opted for the purpose. ROS is a meta-operating system due to the fact that it operates "above" the installed system. ROS requires a base operating system Linux (e.g. Ubuntu distributive). Apart from the main Linux functions ROS also provides such options as working with libraries, data transmitting and reception for a range of devices, error planning and processing.

The ROS structure includes minimal units of performed processes called "nods". Herewith each nod is performed separately while interaction between different nods occurs only in process of message exchange. There are three types of message exchange: topics, services, and actions. Topics providing synchronous unicast transmission/reception are the most relevant for the eye image registering and processing task. Topics operation is based on the "Publisher" – "Subscriber" model of message transmission (figure 8).

Figure 8. Message transmission via the "Publisher" – "Subscriber" model.

Software for the eye pupil variation evaluation was developed on the basis of the selected operating system and the message transmission method. The structure of the software is given in figure 9.
Figure 9. The structure of the eye image registration and processing system software.

In the structure diagram nodes are shown as ovals while topics are shown as arrows. The nodes and the topics are named according to their functional purpose and the type of data transmitted. The names of the topics start with a slash. The functionalities of the nodes and topics can be seen in the table 1.

| Nod name       | Nod function                                   | Topic name   | Topic function                        |
|----------------|-----------------------------------------------|--------------|---------------------------------------|
| camera         | image registration                            | /camera      | transmitting the original image       |
| camera_preview | preparation of the reduced image 640х480     | /camera_p    | transmitting the reduced image        |
| pupil_size     | definition of the pupil size                  | /size        | transmitting the pupil size           |
| timer          | time measurement                              | /tick        | transmitting the time counts          |
| histogram_preview | reduced image histogram calculation         | /hist_p      | transmitting the values of the reduced image histogram |
| frame_interval | frame duration measurement                    | /data        | transmitting data with timestamps     |
| web_video_server | combining the image with the data and sending to the PC |

Python language was applied for the implementation of nodes and topics using the OpenCV library. Herewith the pupil size nod is the most time-critical one; it calculates the histogram of the original image of 1280x960, figures out the binarization threshold and evaluates the eye pupil size. Nevertheless, the first two functionalities of the nod are performed only in the beginning of the program operation, the found threshold value being used further, as the shooting conditions remain relatively stable. In these conditions the histogram calculation duration equals 80 – 100 ms, nod camera – 10 ms, nod histogram_preview – 38 ms, nod pupil_size – 12 – 36 ms.

The position of the eye pupil local area was fixed and its further movements were tracked in each frame for the further increase of the processing speed at the beginning of the program runtime. The suggested decisions allowed increasing the speed of the eye image registering and processing up to 25 frames per second that is sufficient to measure pupil size variations. The developed program also provides the display of the eye image as well as its brightness histogram, which simplifies the system parameters setting.

5. Measuring characteristics for the mobile system of pupil size evaluation

A visual stimulus created by a white LED similar to that described in the article [5] was used in order to measure the characteristics of the developed mobile system of pupil size evaluation. The pupil diameter was expressed as the number of pixels (figure 10). In this case the pupil diameter variation at
the area of abrupt brightness increase was approximated by the exponential function of the time constant (0.2 – 0.16 s), while at the area of abrupt brightness reduction the same function was used with the time constant 0.6 – 0.8 s. The resulting data were similar to the values received in the work [5]. Data processing was made in Logger Pro [14].

Figure 10. Pupil response to the visual stimulus of brightness change.

6. Conclusions
The mobile head-mounted system of eye size measurement based on the single-board computer Raspberry Pi 4 and Raspberry Pi HQ Camera described in the article makes it possible to measure pupil diameter variations in real time. For this task Robot Operation System (ROS) was opted, supporting simultaneous performance of the eye image processing function. Software was developed on the basis of nodes and topics. The system mobility is provided by Wi-Fi image and data transmission as well as by battery power supply.

Acknowledgments
The reported study was funded by the RFBR (project No. 18–47–860018 p_a).

References
[1] Velhover Ye S and Ananin V F 1991 Pupillodiagnostics (Moscow: UDN Publishing House)
[2] Kalnitskaya V E and Pogrebnov A I 2012 Binocular synchronous pupillometry in monitoring the functional state of highly qualified athletes Fizitcheskaya kul'tura, sport - nauka i praktika 2.
[3] Akhmetvaleev A M 2018 Neural network model and software complex for determining the functional state of human intoxication by pupillary response to light impulse exposure. dis. ... c. M. N. KNITU-KAI them. A.N. Tupolev. Kazan
[4] Boronenko M, Boronenko Y, Zelenskiy V and Kiseleva E 2020 Advances in Neuroergonomics and Cognitive Engineering. Proc. of AHFE 2019 Advances in Intelligent Systems and Computing (Springer, Cham) vol 953 doi: 10.1007 / 978-3-030-20473-0_43
[5] Suranov A Ya 2020 High-performance computing systems and technologies 4(2) 33 - 38
[6] Lanata A, Armato A, Valenza G and Scilingo E P 2011 Proc. of the 5th International ICST Conference on Pervasive Computing Technologies for Healthcare (Dublin, Ireland)
[7] Temel D, Mathew M J, AlRegib G and Khalifa Y M 2019 Proc. of International Symposium on Medical Robotics (ISMR) (Atlanta, GA, USA) p 1-7.
[8] Schneider N, Bex P and Barth E 2011 Proceedings of the 2011 Conference on Novel Gaze-Controlled Applications (Article 8) p 1-4
[9] EyeLink II User Manual. https://www.sr-research.com/eyelink-ii/
[10] The official Raspberry Pi camera guide. URL: https://raspberrypi.ru/files/camera_guide.pdf.
[11] ABM-4141 Benchtop multimeter. URL: https://www.aktakom.ru/kio/index.php?SECTION_ID=1042
[12] 20 Best Operating Systems You Can Run on Raspberry Pi in 2021. URL: https://www.fossmint.com/operating-systems-for-raspberry-pi/
[13] ROS/Concepts. URL: http://wiki.ros.org/ROS/Concepts.
[14] Logger Pro 3. https://www.vernier.com/product/logger-pro-3.