Kalpa Publications in Engineering
Volume 3, 2020, Pages 231–241
Proceedings of International Symposium on Applied Science 2019

Controlling External Devices By Using Eye Movements

Duy Long Dao[1], Thanh An Le[1], Thanh Tai Tran[1], Minh Huong Nguyen Thi[1]
[1]Ho Chi Minh City University of Technology,VNUHCM
1511797@hcmut.edu.vn

Abstract

In Vietnam, the number of people with limb defects or diseases that affect the ability of limbs is increasing [1]. They were reduced or incapable of living, unable to walk, to run, and utilities around them became useless. Therefore, it is imperative to build a patient support system in controlling external devices. Recognizing the problem, the author has studied, researched and built a wheelchair control system based on speech recognition and eye movement. In this study, the eye movement system is at the simple level: glance left and right. The system uses an image processing method to locate the pupil. External device for testing is a car model that control by Arduino connected via Bluetooth. Experimental evaluation results show that the eye movement system achieves high accuracy of over 70%. However, the eye movement system by image processing takes a lot of time to process for a command. All of the above shows the feasibility of developing an external control device for people with disabilities.

1 Introduction

In 2014, Imperial College London in the UK developed an eye-tracking software to help wheelchair users with disabilities [2]. In the same year, Dr. Prashant Pillai in the University of Bradford demonstrated new technology which allowed disabled people to control their electric wheelchairs by simply moving their eyes [3]. Besides, in 2016, a group of scientists from United Arab Emirates introduced a new system of eye tracking control for ALS people. The software of the Wheelchair control system is a combination of LabVIEW, MATLAB, C++ and C codes [4].

Particularly in our country, this field is quite new, not many research groups and apply in real life. In 2015, for example, Dr. Huynh Thai Hoang, from Ho Chi Minh City University of Technology, designed an eye-gaze controlled electric wheelchair. This is one of the most famous products in this field in Viet Nam [5]. Therefore, we should focus on researching and developing this field for teaching at universities, helping students better understand the new knowledge, which is a good way to promote student creativity in building community support systems.

The research of this topic is an opportunity to learn more about the concept of eye movements, the methods of image processing to identify the iris [6],... In education, the subject's products can be applied to teaching facilities at school, which is a practical resource for passionate students. For a further development, it can be applied in industrial production, in identifying identities and commands of competent persons to control and operate remote machines automatically.

In this article, I used the color of iris to build a motion recognition program. In our eyes, iris is the darkest place. When I turned the eye image to a binary image, color of the area around iris became the strongest contrast. I will then make a mark, circle darkness area, detect the motion of the iris, and locate its motion in 2-axis coordinates.

2 Method

2.1 Image Processing

In this research, I focus on image processing with a lot of algorithms. The whole process will be shown following to this diagram:

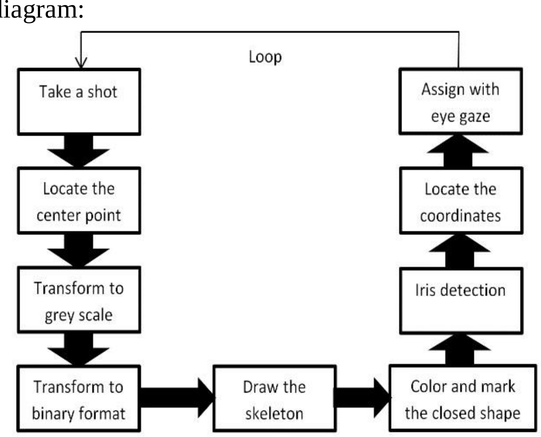


Figure 1: Diagram of algorithms

Now I will introduce some important algorithms and terms:

Gray scale image: each image is represented by a two-dimensional matrix, in which the value of each element indicates the brightness (or gray level) of that pixel. This matrix can be either uint8, uint16 or double. This called "black and white" photos.

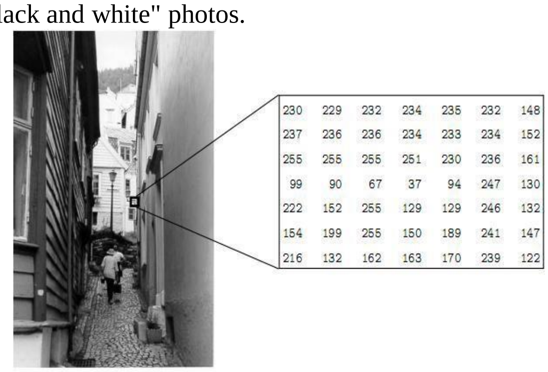
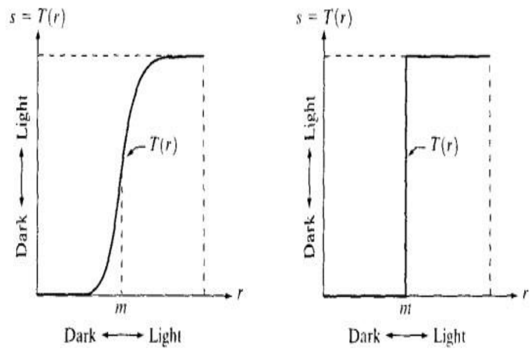


Figure 2: Gray scale image

Gray scale transformation: In this transformation, the value of g(x, y) depends only on the value of f(x, y), and T becomes the gray level transform function. We have the following simple formula:

$$s = T(r)$$

where r is the initial gray level at (x, y), s is the gray level after the transformation at (x, y). For example, we have 2 gray scale



$$s = \frac{1}{1 + (\frac{m}{r})^E} \qquad \begin{cases} s = 0, r < m \\ s = 1, r > m \end{cases}$$

Figure 3: Gray scale transformation

In the left figure, the transformation makes the post-processed image a higher contrast than the original image. The gray level values r < m through transformation are compressed to near 0 (darker), similar to r > m values but compressed near level 1 (brighter) to make the post-processing image high contrast. The transformation in right figure aims to convert a grayscale image into a binary image. We assign the threshold m, with r < m assigned to level 0, and r > m assigned to level 1.

Binary image: the image is represented by a two-dimensional matrix of logical type. Each pixel can only receive one value either 0 (black) or 1 (white).

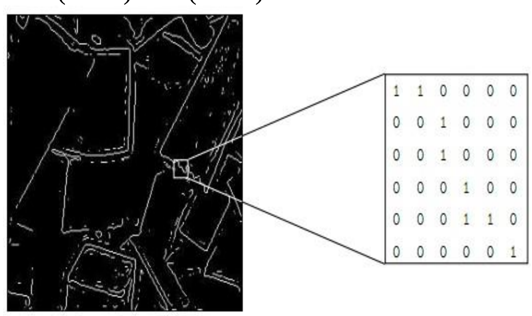


Figure 4: Binary image

Binary transformation: a binary image is the same as a thresholded gray image which is obtained using a threshold T for the original gray image. Thus,

$$B[i,j] = F_T[i,j]$$

Where for a darker object on a lighter background

$$F_T[i,j] = \begin{cases} 1 \; if \; F[i,j] \leq T \\ 0 \; otherwise \end{cases}$$

It is known that the object intensity values are in a range $[T1T2]$, then we may use

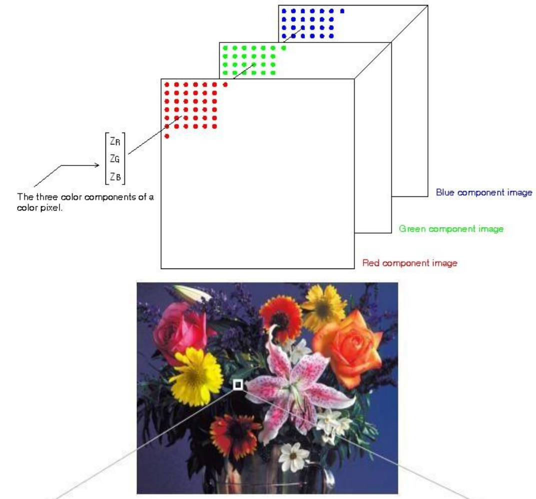
$$F_T[i,j] = \begin{cases} 1 \; if \; T_1 \leq F[i,j] \leq T_2 \\ 0 \; otherwise \end{cases}$$

A general thresholding scheme in which the intensity levels for an object may come from several disjoint intervals may be represented as

$$F_T[i,j] = \begin{cases} 1 \; if \; F[i,j] \in Z \\ 0 \; otherwise \end{cases}$$

Where Z is a set of intensity values for object components.

RGB image: which is also known as a "truecolor" image due to its honesty. This image is represented by a three-dimensional matrix of size m x n x 3, with m x n is the size of the image in pixels. This matrix defines red, green, and blue components for each pixel, whose elements may be of type uint8, uint16 or double.



49	55	56	57	52	53
58	60	60	58	55	57
58	58	54	53	55	56
83	78	72	69	68	69
88	91	91	84	83	82
69	76	83	78	76	75
61	69	73	78	76	76

Red

64	76	82	79	78	78
93	93	91	91	86	86
88	82	88	90	88	89
125	119	113	108	111	110
137	136	132	128	126	120
105	108	114	114	118	113
96	103	112	108	111	107

Green

66	80	77	80	87	77
81	93	96	99	86	85
83	83	91	94	92	88
135	128	126	112	107	106
141	129	129	117	115	101
95	99	109	108	112	109
84	93	107	101	105	102

Blue

Figure 5: RGB image

Skeleton: convert all objects in an image into lines (frames) without changing the original structure of the image.

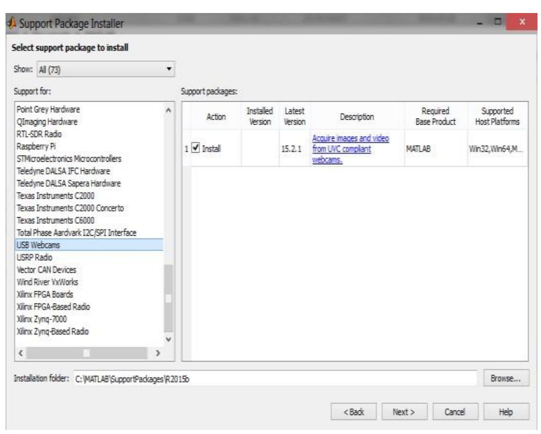


Figure 6: Before and after skelonize

The language code I use to do image processing is Matlab because of its wide application. This is a high-level language for engineering, integrating computing, system modeling and easy programming.

Therefore, I decided to use MatLab in my research, to conduct image processing. The combination of MatLab and Arduino tools to help problem solving is faster and more efficient than ever. Each tool has its own strengths, one tool is in mathematics and data processing; and the other tool is used to control external devices. 2.2. Arduino language

Arduino is a prototype (open source) platform based on easy-to-use software and hardware. It includes a board and a supporting software called the Arduino IDE (integrated development environment for Arduino), used to write and load from a computer code to a physical board. Moreover, the Arduino IDE software uses the simplified version of C ++, making programming learning a lot easier. So I used the board and the Arduino language in this research.

3 Record And Analyse Eye Ovements

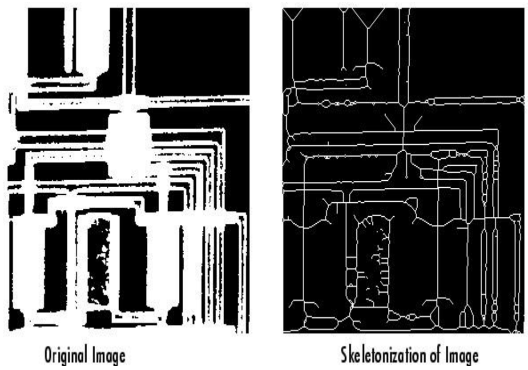


Figure 7. Steps taken

For the first process, after installing the camera, I used the “webcamlist” command to check the connection as well as the number of cameras connected to the laptop. I then used the “webcam” command to start retrieving the image. To use this command, you must download the Add-Ons USB Webcams in MatLab's library.

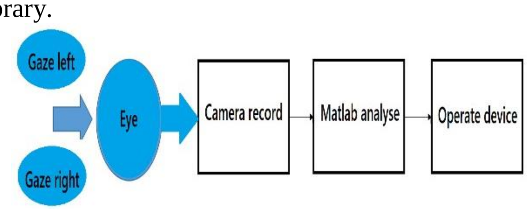


Figure 8: Install USB Webcam library

Next I used the “preview” command (orange) to open the camera figure and check the position of the eye to see if it was off the camera frame.

During the identification process, to increase image brightness, I installed 6 more infrared LEDs around the camera. Here I use infrared LED near 810nm, installed about 4-5cm from the eye position. In addition to increasing the brightness of images, according to a recent study of eeNewsLed magazine, infrared with a wavelength of 810nm will help create a model image with the most obvious contrast to identify iris in security applications [7].

Regarding the safety of cornea, avoiding corneal burns caused by heat from LED, I have also studied a number of documents. Scientists have demonstrated that irradiation levels below 10mW / cm2 are considered safe when exposed to infrared within 720 - 1400 nm. The LED bulb I use is IR-810-524N1 with a diameter of 5 mm. Formula to calculate the irradiance of 1 LED bulb:

$$E = \frac{P}{S}$$

Where P is capacity, S is surface area

According to the LED datasheet, the power is 15mW and the projection angle of the ball is 24o. Now I calculate the area S = 1.5 cm2

Then I calculate the radius of irradiation surface r ≈ 0.7cm. With the angle ϕ = 24o, the shape of the projected light is a cone, so I calculate the half of the projection angle ϕ1=ϕ/2=12o. Next I calculate the distance from the LED bulb to the eye:

$$d = \frac{r}{|\tan(\phi 1)|} \approx 1.1\ cm$$

After caculating carefully, the safe distance for each bulb is about 1.1cm. With 6 LED bulbs at the same position, the safe distance is about 6.6cm. But I setup 6 bulbs at 6 different place, so I can choose the distance from 4 to 5cm.

From here I began the process of identifying eye movement. First take a snapshot via the “snapshot” command, then calculate the camera frame size and determine the horizontal and vertical center lines of the camera frame. The intersection of these 2 lines is the central position of the camera frame. From that central position, I will extend the central area and assign the position when the pupil is normal, not glancing left or right.

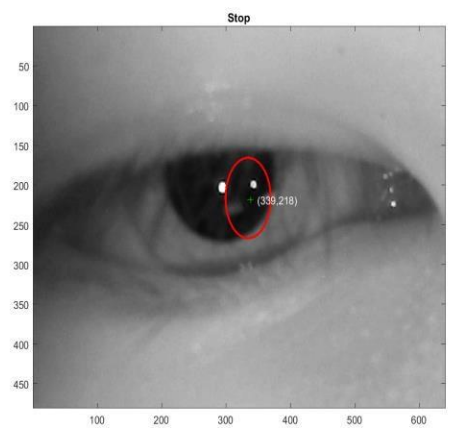


Figure 9: Normal eye after analysing

In the next step, I convert the original image into a black and white image through the command “rgb2gray”. This command converts RGB images into black and white by removing color and saturation information while preserving brightness. Then, with some more complex commands and algorithms, the program will mark and determine the coordinates of the darkest region, which is also the position of the iris. All the above steps are only for one shot. Therefore, in order to process in real time, we have to take many pictures in succession by putting the above algorithms into the loop.

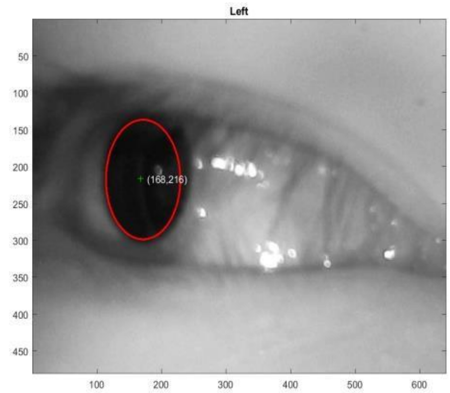


Figure 10: Left gaze regconition

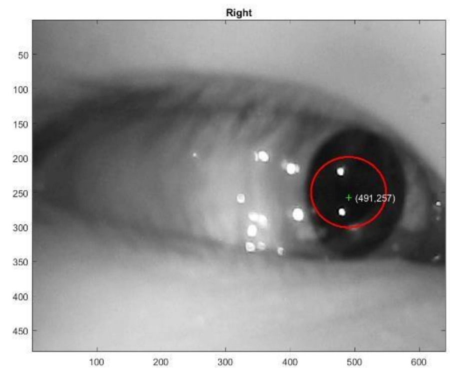


Figure 11: Right gaze regconition

After capturing images and recognizing realtime eye movements, I assign variables to the 3 commands above so that the it can control external devices.

4 Building Control Helmet And External Device

The control helmet includes a JW5530 webcam, 6 IR LEDs, 3.7V battery, Vietduino board. Thus, the total cost to build the product is about 500,000 VND. This is quite a reasonable price for a control device like this

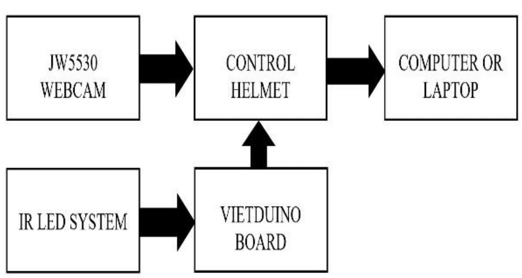


Figure 12: Diagram of control helmet

The control helmet after being designed will work according to the above diagram. IR LED system will be controlled by Vietduino board and operated in parallel with the JW5530 webcam.

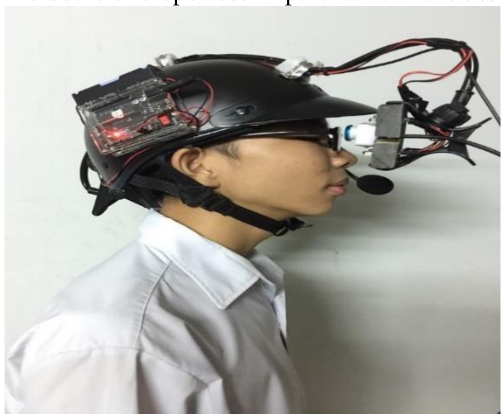


Figure 13: Tester is wearing the helmet

The external device that I use is a bluetooth controlled car model. It includes arduino R3, L293N bridge circuit, HC-06 bluetooth module, HC SRF-04 ultrasonic sensor and some other modules.

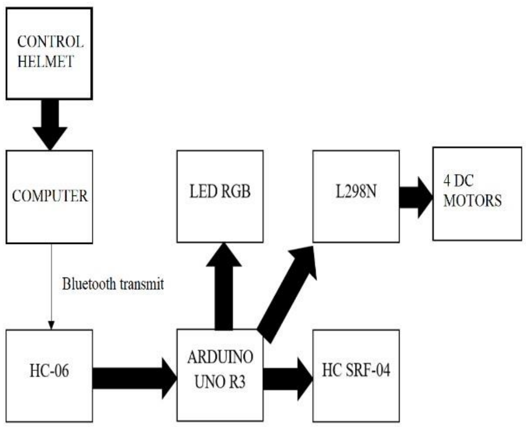


Figure 14: Car’s system block diagram

Following to this block diagram, control helmet will transmit commands to Matlab in computer. After that, Matlab will connect directly to the car model through bluetooth signal. HC-06 module will receive signal and transmit to arduino R3 module to control LEDs, DC motors and ultrasonic sensors.

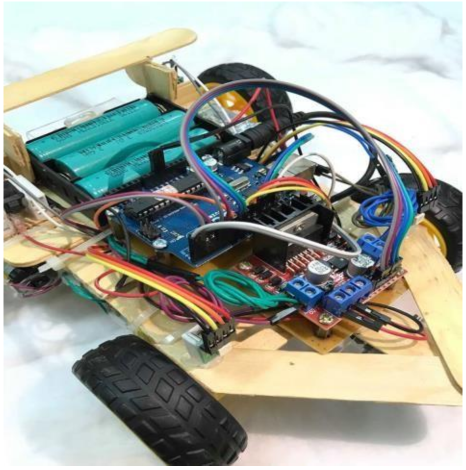


Figure 15: Bluetooth car model

5 Result

In this method, I calculated the appropriate distance for the webcam to receive only the image of the eye, not the image at the back of the head or the eyebrow - which is also the same dark color as the iris. Besides, I also use some extra tricks to increase the reflexes, so the success rate was almost 100%. Therefore, I only evaluate the response speed of the processing.

The experiment I use is to perform a left glance 50 times, then a right glance 50 times. To calculate time, I use the “tic” function to start and “toc” to end. Assign “tic” function to the word "Stop" and the “toc” function when the word "Left" or "Right" appear. Below is a table and a chart of statistics of the experiment.

	Time	Amount		Time	Amount
LEFT GLANCE	Under 1s	0	RIGHT GLANCE	Under 1s	0
	From 1-5s	39		From 1-5s	37
	From 5-10s	7		From 5-10s	11
	From 10s	4		From 10s	2

Figure 16: Statistic of response time in table

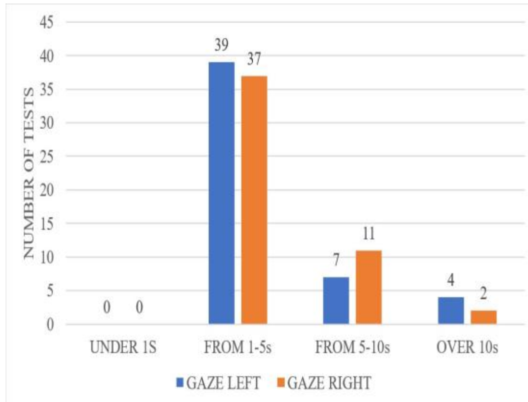


Figure 17: Statistic of response time in chart

Comment: the experiment shows that the response time for a command is mainly from 1-5 seconds. No samples responded less than 1s. As the end of the experiment, the system was no longer stable as the original operation, so the response time gradually increased to more than 5s, at a time the laptop had a lag, causing the screen to respond relatively long for more than 10 seconds. This can be overcome by upgrading the camera, which uses a better processor chip but at a higher cost. In addition, the experimental laptop is an old model, so the processing ability is not good because this is essentially an image processing process similar to graphics processing, requiring good hardware resources to use.

6 Conclusions

After doing researches, I have built a model of eye movement control system with a pretty good performance of over 70%. In order to achieve that, I myself and Ms. Nguyen Thi Minh Huong have worked together to identify the features and methods image processing, iris recognition, etc., and then successfully applied to the control helmet model. However, besides the advantages, there are still some shortcomings that have not been overcome in the research time.

Through this research, I hope to bring practical values to society, especially for disabled people. Thereby I can help them to have more excitement, optimism and more freedom in life. I realize that the cost for this project is quite reasonable, not too expensive, and it is possible to commercialize the product in the future. In addition, I also want to expand the applicability of this topic not only in car model control, but also in controlling all household devices, head to a modern and flexible life.

References

General Statistics Office and UNICEF, "Launch of Key Findings of Viet Nam's first largescale National Survey on People with Disabilities (2016)", January 10, 2019.

Martin Sayers and Caroline Brogan, "Selfdriving AI wheelchair edges closer to aiding people with disabilities", Mobility Unlimited Challenge Discovery Award, April 12, 2018.

Dr Prasant Pillai, "Eye-controlled wheelchair" Bradford University's Open days 2014, June 4, 2014.

Mohamad A.Eid, Nikolas Giakoumidis and Abdulmotaleb El Saddik, "A Novel Eye-GazeControlled Wheelchair Systemfor Navigating Unknown Environments:Case Study With a Person With ALS", New York University Abu Dhabi, March 8, 2016.

Dr Huynh Thai Hoang, “Design and implementation of an eye-gaze controlled electric wheelchair”, Ho Chi Minh City University of Technology, 2015.

Kamal Mitra, “Iris Regconition System Using Matlab”, West Bengal University of Technology, Kolkata, December 2014.

Julien Happich, “810nm IR LED delivers best contrast for iris recognition”, eeNews LED Lighting, July 30, 2017.