PAPER • **OPEN ACCESS**

The application of image processing for Human-Computer Interface (HCI) using the Eye

To cite this article: J K Tan *et al* 2021 *J. Phys.: Conf. Ser.* **2120** 012030

The application of image processing for Human-Computer Interface (HCI) using the Eye

J K Tan, W J Chew* and S K Phang

School of Computer Science and Engineering, Taylor's University, No. 1 Jalan Taylor's, 47500 Subang Jaya, Selangor, Malaysia.

*E-mail: WeiJen.Chew@taylors.edu.my

Abstract. The field of Human-Computer Interaction (HCI) has been developing tremendously since the past decade. The existence of smartphones or modern computers is already a norm in society these days which utilizes touch, voice and typing as a means for input. To further increase the variety of interaction, human eyes are set to be a good candidate for another form of HCI. The amount of information which the human eyes contain are extremely useful, hence, various methods and algorithm for eye gaze tracking are implemented in multiple sectors. However, some eye-tracking method requires infrared rays to be projected into the eye of the user which could potentially cause enzyme denaturation when the eye is subjected to those rays under extreme exposure. Therefore, to avoid potential harm from the eye-tracking method that utilizes infrared rays, this paper proposes an image-based eye tracking system using the Viola-Jones algorithm and Circular Hough Transform (CHT) algorithm. The proposed method uses visible light instead of infrared rays to control the mouse pointer using the eye gaze of the user. This research aims to implement the proposed algorithm for people with hand disability to interact with computers using their eye gaze.

1. Introduction

Human-Computer Interaction (HCI) has long emerged since the 1980s [1]. Despite the early existence, methods for HCI are still limited. Common methods of interaction include using the arm [2], fingers [3], voice [4] and eyes [5]. The eyes are especially useful to interact with the computer because human eyes contain various information which is a suitable candidate for data extraction and can even be use by people who have hand disability. Various method of eye gaze tracking for HCI is available such as using infrared rays to pinpoint the eye gaze on a computer screen.

By implementing the eye gaze tracking system with the use of infrared rays to pinpoint a person's gaze, the accuracy of the eye gaze tracking system will be high. However, by shooting the infrared ray on a person's pupil, it could cause eye damage. The near-infrared region exists between 760 and 1400 nm which are almost invisible to the cornea by 96%. Although the non-ionizing radiation of the infrared radiation is less than 12eV and is deficient for ionization, it could cause a varied biological effect of thermal origins. Moreover, it could stimulate molecular vibration and rotations which interacts with tissues that generate heat. As mentioned, the dominating of thermal effects and extreme exposure to the retina and choroid could cause enzyme denaturation. Also, the regenerative capabilities of the retina are quite limited, hence, any damage to the choroid or retina could cause severe loss of visual acuteness [6]. Therefore, an image-based approach has become a popular alternative.

Multiple methods of eye tracking are available such as the Starburst, neural network algorithm and more [7]. Neural network algorithm has been applied in various applications such as object recognition, autonomous robot driving and more. However, the neural network algorithm is extremely complicated and has to be extensively tuned to get the performance that is desired [8]. On the other hand, Starburst algorithm combines two approaches which are feature-based and model-based using infrared imagery. The objective of this algorithm is to obtain the center location of the pupil and the corneal reflection to compare the difference in vectors between those measurements to coordinates in the image [9].

The two aforementioned algorithms are widely used for image-based detection. To reduce the complexity and avoid the use of infrared ray as a medium to pinpoint a person's gaze, an image-based detection approach was proposed by combining two algorithms which are the Viola-Jones algorithm for facial feature detection and the Circular Hough Transform (CHT) algorithm for iris detection. The Viola-Jones algorithm was chosen as it is a popular method for object recognition which means that this algorithm can detect face features in a short amount of time. Secondly, the CHT algorithm was chosen since it was primarily designed to detect lines or circular objects. To avoid the use of infrared rays, this algorithm suits perfectly for the detection of a circular object where the color and shape of the iris of human eyes can be easily detected using this algorithm.

As the conventional system to control the mouse pointer is the mouse device, people with hand disabilities could not utilize the mouse as intended. Hence, the objective of this research project is to study the implementation of the Viola-Jones algorithm and CHT algorithm on human-computer interaction for people with hand disability using a person's eye gaze via the computer webcam. This paper describes the methodology of implementation of the two algorithms and the results of the proposed algorithm obtained after undergoing an accuracy and repeatability test.

2. Literature Review

Various methods are available to track a person's eye gaze such as electrooculography where the information of the position of the eyeball is collected using electrodes, a video-based eye-tracking system, and many more ranging from low cost to high cost [10]. Electrooculography (EOG) is an eye-tracking method that determines the resting potential of the retina. EOG usually measures an electrical signal or electrooculogram by sticking sensors around the eye. By moving the eyes, the retina will move toward one electrode while the cornea will move toward another opposing electrode. By observing the change in the electric potential field in the electrooculogram, eye movement can then be tracked [11]. However, using EOG as a method for HCI is quite intrusive as multiple sensors are needed to be attached around the eye region which is quite uncomfortable for users to use in a prolonged time.

Another method of eye-tracking that is already available is video-based eye-tracking. The working principle of video-based eye tracking is estimating the direction of gaze of a person from a picture that is captured frame by frame from a video. A way to detect the iris of the eye is by the contrast of the dark and bright area of the iris. Besides that, the eye-tracking system is also split into two different categories based on the type of light that was utilized such as infrared ray or visible light. The video-based eye-tracking system can be further split into two systems, invasive and non-invasive [12]. Invasive systems are eye tracker that requires the user to mount a system over their head and this might be uncomfortable to the user over a prolonged period of usage [5].

Non-invasive eye-tracking systems usually use a remote camera where users do not need to wear anything on their body such as the webcam approach for this research project. The non-invasive system usually captures the image of the human eyes by sectioning the features of the face as the non-invasive system capture the whole face of a person instead of just the image of a person's eyes. Despite the difference in approach to capture the image of the human eyes, the purpose of these systems is still the same which is to detect a person's point of gaze by detecting changes when the eye rotates [12].

Moreover, many video-based eye trackers work by illuminating an infrared ray on the cornea of the user's eyes. The infrared ray that is on the cornea of the eyes is used as a reference point for gaze estimation. This phenomenon is also known as "red-eye" which appears when taking photograph with flash [13]. This method of gaze estimation is also known as the corneal reflection method.

Another form of video-based eye-tracking system works by tracking the pupil movement and location. This research methodology was adopted by [14] and captures the image of the face via the webcam and then determine the person's eye gaze by taking the left and right corners of the eyes as a reference point to determine the shift of the iris as the person change focus. With the data acquired it can move the mouse pointer to the location of the user's focus.

A third type of method for eye tracking that is used in the medical field is infrared oculography and is commonly used for Magnetic Resonance Imaging (MRI). Infrared oculography also uses infrared rays as its basis for eye tracking detection. Although both the video-based eye-tracking system and infrared oculography also uses infrared ray, the method of detection is somewhat different. For this method, detection is obtained by comparing the intensity of the projected infrared ray and the reflected infrared ray from the sclera of the eyes [7]. The amount of reflected ray will determine the change of point of gaze of the eyes. However, this is an invasive system that requires the user to wear a spectacle. As this method depends on the intensity of infrared ray reflected, it is very sensitive to change due to external surrounding light source. Moreover, this eye-tracking method can only measure ± 35 degrees along the x-axis and ± 20 degrees along the y-axis. Due to its limitation of eye movement tracking and it is highly sensitive to changes in external light source, it is not suitable for a bright environment.

Based on the literature review, it can be said that infrared oculography is not viable as it has major limitations when it comes to controlling the mouse pointer due to its limited eye movement detection and being highly sensitive to the external light source. Also, with the risk that comes with the corneal reflection method as explained earlier, this method should be avoided if possible, despite having a better accuracy compared to image-based eye-tracking that solely utilizes the image of the human eyes under visible light. Therefore, an image based HCI system is proposed to help users capture their eye gaze which then can be used to move the mouse in a computer. This method is non-invasive and does not pose any risk to the user.

3. Methodology

The main approach proposed in this paper is tracking the iris location by using the reflection of the iris under natural visible light, as infrared ray could cause damage to the eyes. In general, to detect a person's eye gaze, detections on the face, eyes and iris need to be implemented first. Consequently, by comparing the difference between reference points which is the corner of the eyes and the current location of the iris in real-time, the eye gaze of a person can then be estimated. After knowing the eye gaze location, it can then move the mouse pointer to the selected location. The flowchart of the proposed program is as shown in Figure 1.

The webcam used for this project is the Logitech BRIO which can capture high-resolution images. After acquiring the image in real-time, it will then be converted from RGB to grayscale as RGB image is harder to process compared to grayscale. Hence, to shorten the computation process, the greyscale image is much preferred and is the reason why it was used for the proposed system.

After acquiring the pre-processed image in the image acquisition stage, Viola-Jones algorithm will be implemented to detect the face and then the eyes from the image. Consequently, the face is first segmented by cropping the image out from the captured area. Then, the eye image was segmented out from the face image. The reason to segment the image features by features is to reduce the false detection of eye movements.

3.1 Face and Eye Detection (Viola-Jones Algorithm)

To detect the face and eye of the user, an algorithm named Viola-Jones algorithm is introduced. Viola-Jones algorithm was known as a popular algorithm for face detection, as this technique allows for real-time facial detection and is reliable [15]. There is 4 ideation that was adapted and improved which are Haar-like features, integral image, Adaptive Boosting classifier learning and the attentional cascade.

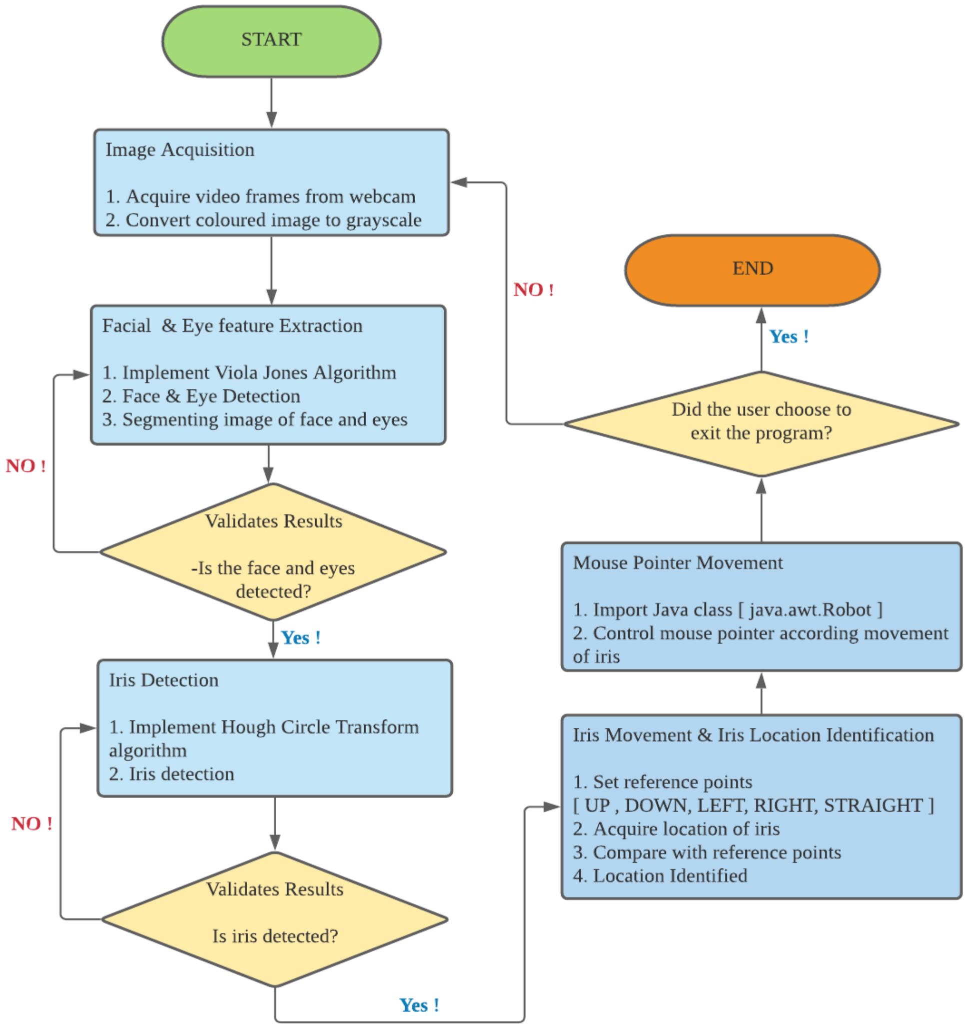

Figure 1. Algorithm flowchart to control the mouse pointer using the human eyes

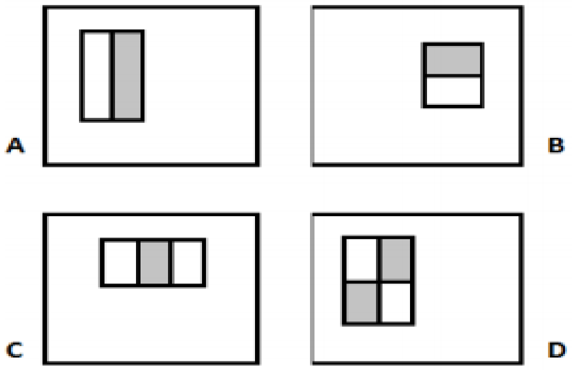

Figure 2. Rectangular features relative to the enclosing detection window [16]

The Viola-Jones algorithm uses rectangular features instead of pixels. As there is a limited quantity of data for training, it is not enough to encode ad-hoc domain knowledge. Hence, using a rectangular feature can solve this issue [16]. The rectangle features in Figure 2 represents the difference between pixels inside the rectangular zones. The group of pixels are useful to determine light and dark zones. Also, these rectangular features can be scaled according to the size of a pixel group that is to be examined. Therefore, with such scalability allowed, it can detect objects of different sizes.

By utilizing an intermediate representation of the image, rectangular features can be computed in a short amount of time and this method is known as integral image. The integral image at location x and y represents the sum of the pixel above, as shown in Figure 3.

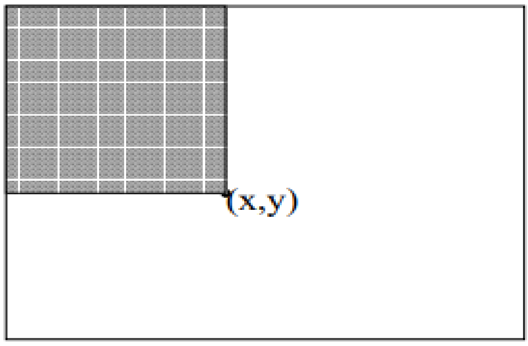

Figure 3. The shaded area represents the total sum at point (x,y) [16]

To compute the Haar-like feature, there is a need to sum up the pixels in the rectangular area. Besides, by using the integral image method, it can output four integral image value per clock cycle without any noticeable increase in the number of addition operations, hence, shortening the computation time [17].

Adaptive boosting algorithm then carries out feature selection and trains the classifier. The main idea of the AdaBoost algorithm is to train a weak classifier for the same training set and then assemble the weak classifier to become a strong classifier. Consequently, the AdaBoost classifier is applied in the cascade classifier architecture. The rectangular features are split into groups where each of the groups containing rectangular features are used in stages in the cascaded classifier. The stages in the cascaded classifier with the AdaBoost algorithm will decide if the input area is the image of a human face. However, if the input area of the image is not a human face, the area will be discarded. The overall flowchart of this classifier is shown in Figure 4 [18].

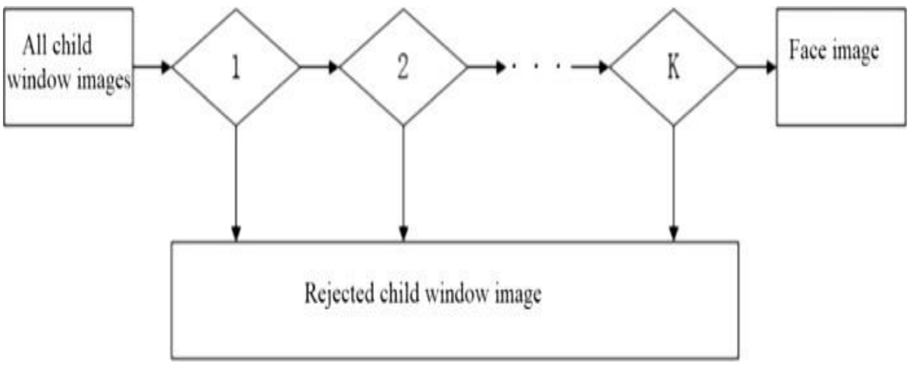

Figure 4. Flowchart of the cascaded classifier with AdaBoost algorithm [18]

3.2 Iris Detection, Movement, and Location Identification

Next, with the image of the eyes acquired from the previous stage, the CHT algorithm was implemented to detect the iris of the eyes. As the colour of the iris and the colour of the sclera is relatively different and with the CHT detection setting adjusted to detect only dark objects, the iris can easily be identified. By allocating a reference point, the movement of the iris can be easily estimated, hence, identifying the location of the eye gaze. Figure 5 will give an idea of how the location or direction of the iris is detected.

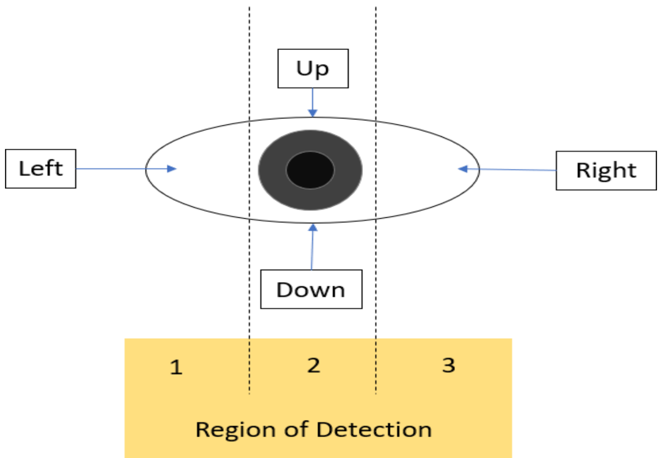

Figure 5. Region of iris detection by implementing the CHT algorithm

As shown in Figure 5, there are 3 regions of iris detection. The three regions are set to the left region, the right region, and the central region. The reasoning behind this method of data acquisition is to not complicate the overall system so that the margin of error will be lower. As an example, when the iris moves to either the right or left region of detection, the algorithm is programmed in such a way that it would only take account of the x-axis coordinates, hence, providing a more accurate result. Consequently, in the center region of detection, the detection region is set between the right and left region to allow the system to consider less factors so that it will provide a better result. In short, the location of the detected iris of the upper and lower region will only be detected in the central region while the left and right direction will only be either in the first and third region depending on the point of gaze. To detect the iris of the eye, the CHT algorithm was implemented. A circle is represented as equation (1):

$$(x - ax)^2 + (y - ay)^2 = R^2 \quad (1)$$

where (ax, ay) is the center of the circle and R being the radius of the circle [19]. By implementing this algorithm from the image processing toolbox from MATLAB, the center of the iris can be identified. Hence, by comparing the reference points of the corner of the eyes, the eye gaze of the user can be identified accordingly.

To ensure reproducibility of data, the dimension of the cropped image of the eyes will be standardized. Currently, the standardized dimension of the cropped image of the eyes is set at 50 x 70 pixel.

3.3 Mouse Pointer Control

Accordingly, with the estimation of eye gaze made, the coordinates of the destination of the mouse pointer can be set to move with respect to the eye gaze location. As MATLAB does not have the toolbox or the ability to move the mouse pointer programmatically, a function from java which is jave.awt.robot is needed to be imported to allow the control of the mouse pointer. The working principle of this function is just to automatically control the mouse from the data acquired in the iris detection stage. As the screen resolution of each user is different before the mouse pointer is moved programmatically, the screen size of the user must first be measured using MATLAB. Moreover, to ensure the mouse pointer moves smoothly, the last coordinates of the location of the mouse pointer is then stored into a variable and the content of the variable is refreshed whenever the mouse pointer moves. Furthermore, to allow the user to have better control of the mouse pointer, the sensitivity of the mouse or the speed of the movement of the mouse pointer can always be set before the program begins by selecting fast, medium or slow. According to the selection made, the program will then use the pre-set numbers to control the sensitivity of the mouse pointer.

4. Results and Discussion

The algorithm was constructed using the MATLAB software (Version 9.8, R2020a) and various inputs were tested to evaluate the consistency and the precision of the algorithm. The results obtained are plotted and a formula was used to calculate the consistency and the precision of the algorithm. The hardware setup to evaluate the performance of the algorithm is shown in Figure 6.

Figure 6. Experimental setup for algorithm test

The hardware setup consists of a Logitech BRIO webcam and a PC with a 2.5GHz Intel i5 7200U CPU with NVIDIA GeForce 940MX 2 GB graphics card and 8 GB RAM with Windows 10 as the operating system. The software setup for the tests was just a few lines of code in the MATLAB software to store and process the data obtained from the test and the LogiTune software to control the settings of the Logitech BRIO webcam.

$$\text{Consistency} = \text{True Positive} / (\text{True Positive} + \text{False Negative}) \tag{2}$$

In equation (2), false negative represents the number of times where the output should be categorized as true but was categorized as otherwise. This equation was being used to evaluate the overall performance of the algorithm. Consistency value represents the repeatability of the results produced by the algorithm.

As the experiences of every individual using the proposed algorithm is going to be subjective, hence. there is a need for a middle ground to evaluate the proposed algorithm. By collecting data from tests to analyze, the data collected are then applied into mathematical equation such as equation (2) for a fairer and logical evaluation. Examples of the test are as shown in Figure 7.

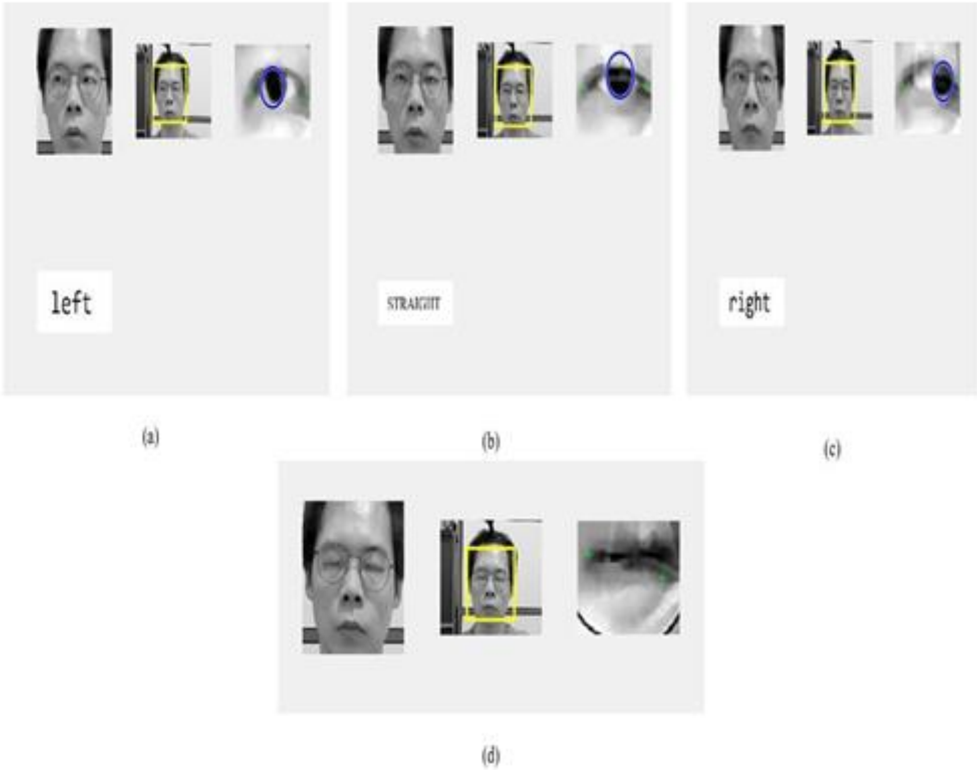

Figure 7. (a) Looking left, (b) Looking straight, (c) Looking right, (d) Blink

This test was performed for looking left, right and straight direction as well as for blink. Each of the looking test was 54 cycles long which was approximately 50.81 seconds, whereas the test for blink ends after 20 blinks. For every direction of the test, the test subject was required to look in a certain direction throughout the tests depending on the direction of the test. Whenever the correct conditions were met, a variable will store the results for further analysis using equation (2) above. The stored data were then plotted to present the performance of the overall system with every input that was input by the test subject. The results obtained were tabulated and shown in Tables 1 and 2 as well as in Figures 8, 9, 10 and 11.

Table 1. Results of algorithm test (Number of Cycles = 54)

Direction	**Right**	**Left**	**Straight**
True Positives	44	46	40
False Negatives	10	8	14
Consistency	81.48%	85.19%	74.07%

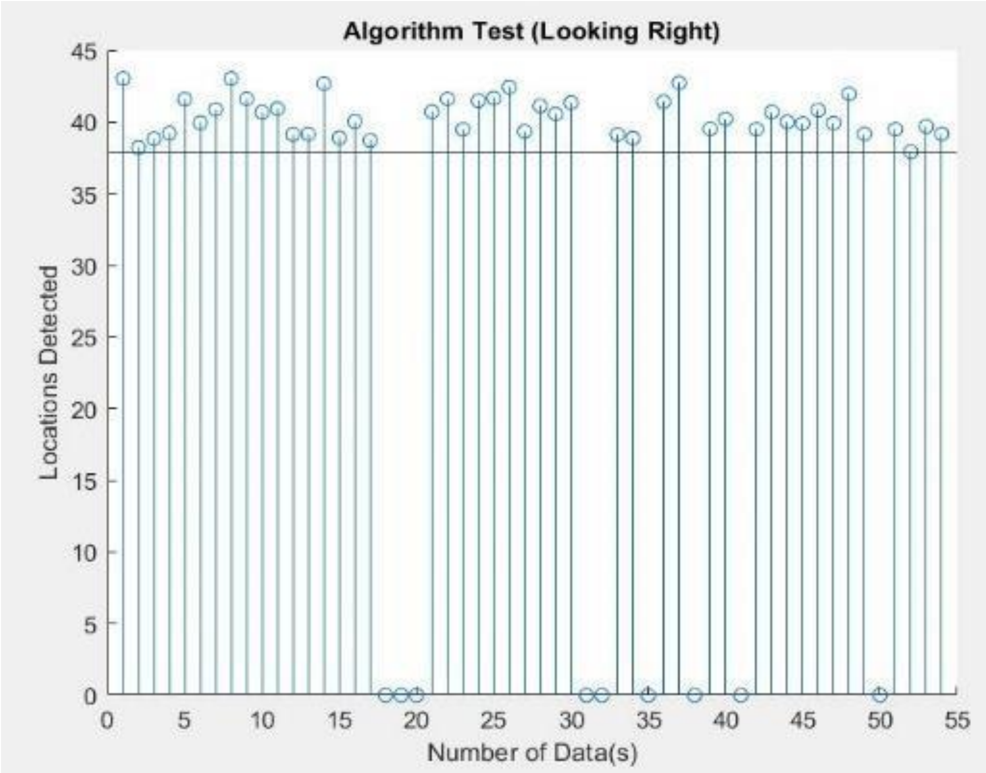

Figure 8. Algorithm test when the subject is looking right

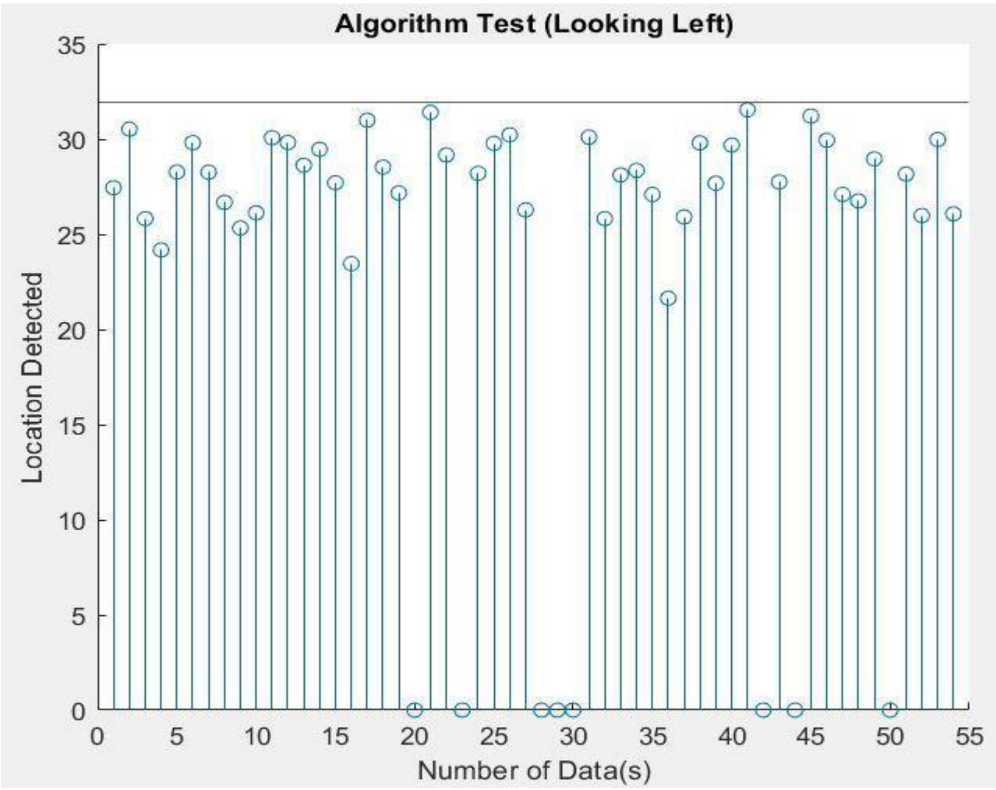

Figure 9. Algorithm test when the subject is looking left

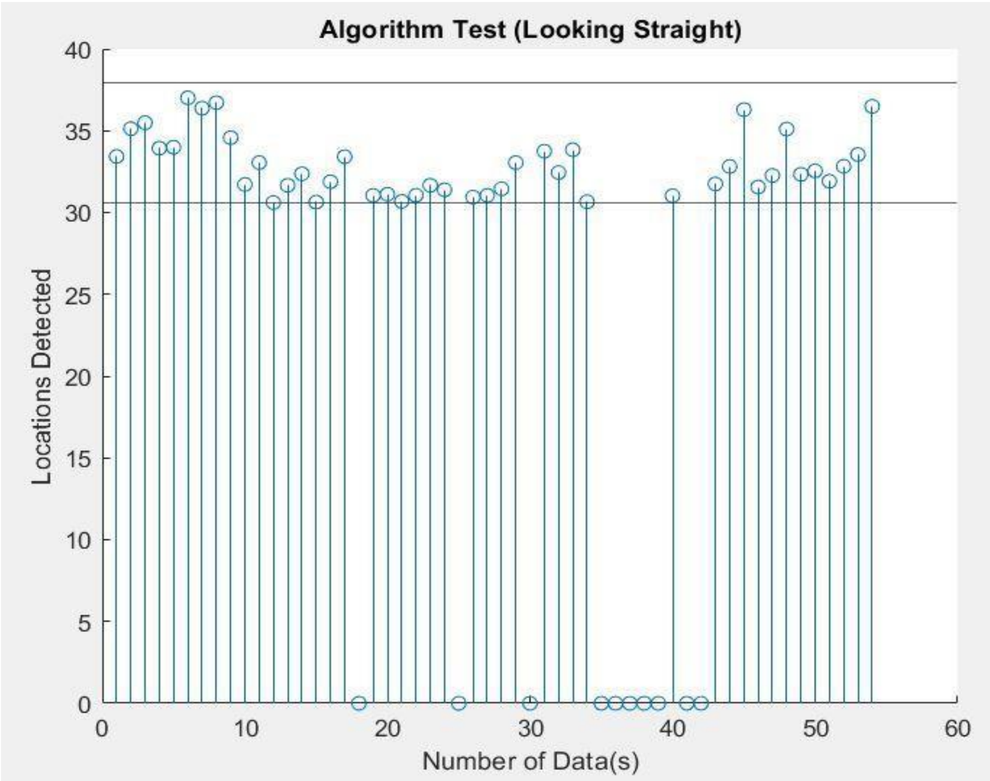

Figure 10. Algorithm test when subject is looking straight

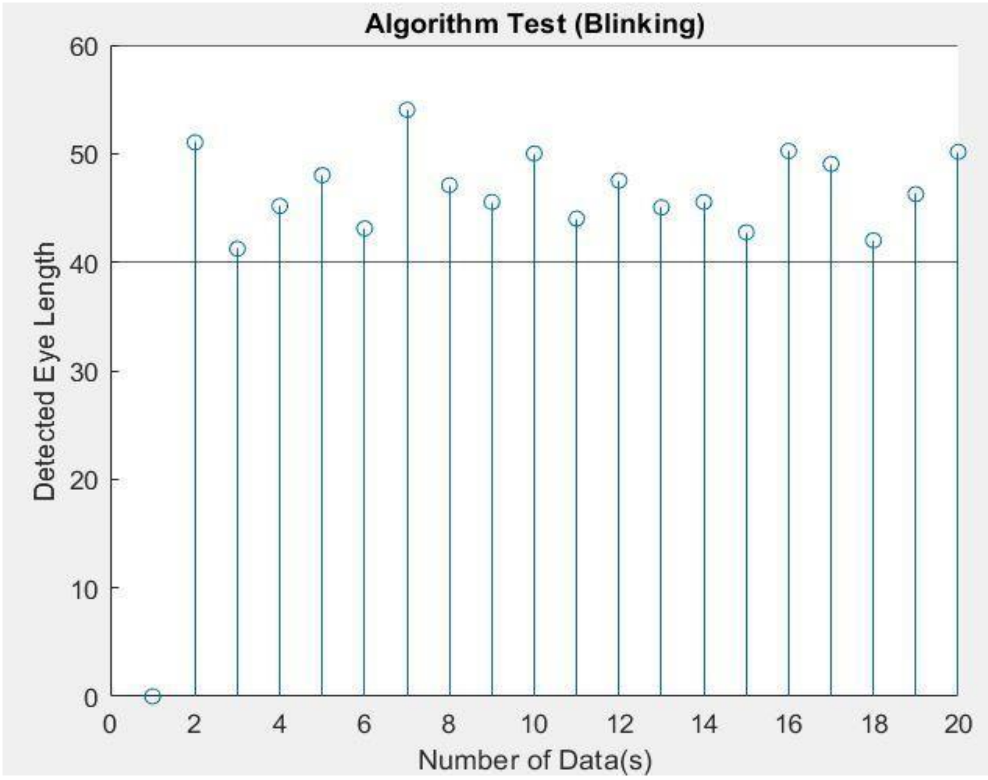

Figure 11. Algorithm test when the subject is blinking

Table 2. Algorithm test for blinking

Number of Blinks	**20**
True Positives	19
False Negatives	1
Consistency	95%

In Figures 8, 9 and 10, a horizontal line was drawn at specific locations where the lines are represented as thresholds. The circular points in these figures are detected location of the center point of the iris. In Figure 10, it can be observed that there are two threshold lines. As the threshold for looking straight is higher than when looking towards the left direction but lesser when looking towards the right direction, two threshold lines were drawn to show that any points that falls between the two lines are considered as looking straight. There are some 0 values at certain data points and those points are categorized as false negatives.

Furthermore, there is a part of the algorithm that detects line lengths for the purpose of detecting blinks. When the eyes are closed, there will be a long line formed around the eyelids, hence, a line detecting algorithm was used to measure the length of the line detected. Figure 11 consists of circular points which are the result of the measured length of line detected around the eyelid, where the length above the horizontal threshold line indicates that the eye was closed. Table 2 shows the consistency of the proposed eye blink algorithm for 20 blinks. As the proposed algorithm allows users to control the mouse pointer not only by moving it but also includes the clicking mechanism, hence, the performance of blinking detection is equally important as moving the mouse pointer. The threshold values were obtained after multiple trials to ensure consistency and precision as listed in Tables 1 and 2.

5. Conclusion

This paper proposes an image-based HCI system that incorporates the Viola-Jones and CHT algorithm to control the movement of a computer mouse by just using the eye gaze. The performance evaluation was tested quantitatively where some datasets are obtained manually. Moreover, the results obtained in this paper are proven significant and successfully demonstrates the workability of the proposed algorithm and avoids the eye gaze tracking method that uses infrared rays. The significance of this research project is the ability of this technique to extract data from the human's iris based on a person's eye gaze location. This can help people with disabilities interact with the computer in a more convenient way. Lastly, the eye gaze detection technique proposed in this paper could also be adopted in other fields of studies such as to determine the attention level of students and act as an examination surveillance system as popularity of online classes has been rising due to the current COVID-19 pandemic.

References

[1] MacKenzie I S 2013 *Human-Computer Interaction: An Empirical Research Perspective* (Amsterdam: Morgan Kaufmann)

[2] RB Widodo, AB Haryasena, H Setiawan, PLT Irawan, M Subianti, C Wada 2020 Design and evaluation of upper-arm mouse using inertial sensor for human-computer interaction. *Journal of Engineering Science and Technology* vol. 15, no.6, pp. 3673-3690

[3] V Ranga, N Yadav, P Garg 2018 American sign language fingerspelling using hybrid discrete wavelet transform-gabor filter and convolutional neural network. *Journal of Engineering Science and Technology* vol. 13, no. 9, pp. 2655-2669

[4] Shah SM, Memon M, Salam MHU. 2020 Speaker recognition for pashto speakers based on isolated digits recognition using accent and dialect approach. *Journal of Engineering Science and Technology* vol. 15, no. 4, pp. 2190–207

[5] Rashid, Mofeed & Rashid, Abdulmuttalib & Aldair, Dr Ammar & Rashid, Osama. 2020 A Robot arm control system for amputees using eye gazing techniques. *Journal of Engineering Science and Technology* vol. 15, pp. 905-918
[6] Kourkoumelis N and Tzaphlidou M 2011 Eye safety related to near infrared radiation exposure to biometric devices *Scientific World Journal* vol. 11, pp. 520–8
[7] Orman Z, Battal A and Erdem Kemer 2011 A Study on Face, Eye Detection and Gaze Estimation *Int. J. Comput. Sci. Eng. Surv*. vol. 2, pp. 29–46
[8] Yang M-H, Kriegman D J and Ahuja N 2002 Detecting faces in images: a survey *IEEE Trans. Pattern Anal. Mach. Intell*. vol. 24, pp. 34–58
[9] Li D, Winfield D and Parkhurst D J 2005 Starburst: A hybrid algorithm for video-based eye tracking combining feature-based and model-based approaches *2005 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Work*. vol. 03, pp. 79
[10] Zende S, Tambile V, Thakur A, Shendge M and Rathi S 2016 Mouse Pointer Movement using Gaze Tracking System *Int. J. Comput. Appl*. vol. 140, pp. 1–4
[11] Aungsakul S, Phinyomark A, Phukpattaranont P and Limsakul C 2012 Evaluating feature extraction methods of electrooculography (EOG) signal for human-computer interface *Procedia Eng*. vol. 32, pp. 246–52
[12] Chennamma H R and Yuan X 2013 *A Survey on Eye-Gaze Tracking Techniques* vol. 4, pp. 388–93
[13] Ball, L. J. 2014 Eye-tracking and reasoning: What your eyes tell about your inferences. *New approaches in reasoning research* pp. 51–69
[14] Nasor M, Rahman K K M, Zubair M M, Ansari H and Mohamed F 2018 Eye-controlled mouse cursor for physically disabled individual *2018 Adv. Sci. Eng. Technol. Int. Conf. ASET 2018* pp. 1– 4
[15] Paul T, Shammi U A, Kobashi S and Detection M F 2018 A Study on Face Detection Using Viola-Jones Algorithm in Various Backgrounds, Angles and Distances *Int. J. Biomed. Soft Comput. Hum. Sci. Off. J. Biomed. Fuzzy Syst. Assoc*. vol. 23, pp. 27–36
[16] Viola P and Jones M 2001 Robust Real-time Object Detection *Int. J. Comput. Vis.* vol. 57
[17] Ehsan S, Clark A F, ur Rehman N and McDonald-Maier K D 2015 Integral images: Efficient algorithms for their computation and storage in resource-constrained embedded vision systems *Sensors* vol. 15, no. 7, pp. 16804–16830
[18] Huang J, Shang Y and Chen H 2019 Improved Viola-Jones face detection algorithm based on HoloLens *Eurasip J. Image Video Process*. vol. 41
[19] Sit B and Quraishi M I 2016 A Review Paper on Hough Transform and it's Applications in Image Processing *Int. J. Innov. Res. Sci. Eng. Technol.* vol. 5, pp. 206–213