EMITTER International Journal of Engineering Technology
ISSN: 2443-1168, Vol. 8, No. 1, June 2020, pp. 67~85
DOI : 10.24003/emitter.v8i1.509

Developing Shooter Game Interaction using Eye Movement Glasses

Abdullah Iskandar, Achmad Basuki, Artiarini Kusuma Nurindiyani, Faris Rasyadi Putra, Mohamad Safrodin

Department of Creative Multimedia Technology
Politeknik Elektronika Negeri Surabaya
Jl. Raya ITS, Sukolilo, Surabaya, 60111, Indonesia
Telp: 6231 5947280 Fax: 6231 5946114
E-Mail : abdis.gametech@gmail.com, basuki@pens.ac.id, artiarini@pens.ac.id, farisputra11@gmail.com, safrodin@pens.ac.id

Received April 14, 2020; Revised May 7, 2020; Accepted May 25, 2020

Abstract

A quadriplegic is a paralysis that affects limitations in some physical movements and psychological disorders. They have limited media to interact with computers so a suitable solution is needed in the form of a media that can recognize other body parts movements which in this research uses eye movement. One of the solutions to this problem is to propose alternative technologies to interact and play games. We propose a simple technique by using a camera mounted on a glasses that will take the eye area. This technique will help reduce unnecessary parts of eye detection so that performance increases. The eyes movement will be processed using basic image processing and then determined the center of pupil using the Mean method. The system uses pupil movement as pointer as motion control and blinking of eyes for shooting. The performance test of this method toward the system, which has used 10 people with 7 experiments, shows an accuracy of 84.86 percent, the speed of movement with a duration of 2.22 seconds and the speed of response blinking with a duration of 0.026 seconds. In addition, we can distinguish between intentional blink and unintentional blink in which intentional blink has a duration of 0.30 seconds and unintentional 0.12 seconds. It can be concluded that by using this method and this technique is able to achieve good accuracy and also able to use intentional blink as shoot trigger.

Keywords: shooter game, eye controller, eye glasses.

1. INTRODUCTION

The quadriplegic is a paralysis caused by illness, injury, or imperfect growth that affects the functions of bones, muscles, and joints [1]. Many quadriplegic people also have the psychological disorder. They tend to feel embarrassed, sensitive, and inferior. Besides these characteristics, there are other problems caused by the tactile, kinesthetic, and emotional disorder. A quadriplegic person is also called as disabilities. The World Health Survey and Global Burden of Disease estimated that there is 15.3 percent people with disability around the world in 2004 [2]. People with mobility disabilities have limited capabilities in moving, performing manual task and taking participation in life activities. Since that mobility impairments can significantly make the activities feel tedious, even in some cases quite disable using of information technology, it is required to make adaptations that will enable full communication with computers for people with this kind of disabilities.

In the context of special education, there are four main components, such as physical environment, teaching procedures, teaching content, and the use of adaptive equipment. Therefore, one important component of learning that must be considered is using media that suits the needs of the children. The media is a simple or complex technology that can be used by the children with special need [3]. The proposed media should able to increase children's spirit and creativity especially to deal with the quadriplegic children limitation. Therefore, a special media is needed for them. One of the solutions is using a media in form of game [4]. the genre of game that will be used is a shooter game. Shooter game is a game where the player has to move the target to shoot any enemies around the virtual world.

This paper proposed a simple technique by using glasses embedded with a camera which will only capture the eye part. By this technique, it will make the algorithm process the eye detection faster because it does not have to remove the unnecessary part of the image. On the hardware that has been created, which is a pair of eye glasses, there are four points that can be adjusted. So, these eyeglasses will work for every user that want to play the game. And also, there is a calibration for the user before they play the game. This will make the algorithm adapt to every user. This calibration will compare the pupil movement area and the desktop screen.

This research is limited to create a 3D shooter game that is specialized for quadriplegic children. The game will be controlled by the pupil movement of the children. The game will be created for the desktop platform. Lastly, this research will create a hardware to help the algorithm for capturing the pupil. Eye tracking will be used as the main input of the game to help them move the target mark so, they will be able to aim and shoot the enemies. The game needs a special hardware that has a camera facing the eye of the user. The camera will capture the pupil of the eye to keep it tracked as the user plays the game. And when the algorithm says the pupil is not found, because the eye is blinking, it will be detected as a shooting command.

2. RELATED WORKS

Somaya Adwan and Hamzah Arof proposed an innovative technique for eye detection using Augmented Dynamic Time Warping (ADTW) algorithm.

The proposed technique overcomes some of the shortcoming associated with traditional dynamic time warping and it improves the alignment of the warping path of the sequences. The result of the implementation of the ADTW to detect eyes in facial image when either using or not using eye glasses showed improvement in detection accuracy when compared with the traditional DTW. The Detection rate increased from 60% to 70.67% when applying the image processing strategy and increased up to 95% when applying image processing and applying the proposed weighting scheme. [5]

Muhammad Usman Ghani, Sarah Chaudhry and Maryam Sohail and Mohammad Nafees Geelani proposed a low-cost real time system for eye-gaze based on human-computer interaction. The algorithm consists of three steps, facial feature extraction, eyes features extraction and Point of Gaze (PoG) calculation. Proposed system performance was analysed in different scenarios and some limitations were defined. The user head should be at the same altitude as the web-cam, and the distance should be in the range of 20-75 cm. This system can't be used with glassed on, lighting conditions should be good and head movements are not allowed. The results show that the efficiency of pupil detection was performed for 6-10 milliseconds, PoG computation was performed for 2-3 milliseconds. In other side, the accuracy for featured extraction algorithm depends upon image quality and lighting conditions. Algorithm performance drops down in the poor lighting environment. [6]

Tereza Soukupova and Jan Cech proposed a real-time algorithm to detect eye blinks in a video sequence from a standard camera. Recent landmark detector, trained on in-the wild datasets exhibit excellent robustness again a head orientation with respect to a camera, varying illumination and facial expression. They estimate the landmark position, extract a single scalar quantity-eye aspect ratio (EAR)-characterizing the eye opening in each frame. An SVM classifier detects eye blink as pattern of EAR values in short temporal window. The results show that the Landmarks could be detected precisely enough to reliable estimate the level of the eye opening. [7]

Chao Gou proposed a method for simultaneous eye detection and eye state estimation. Based on a cascade regression framework, the method iteratively estimates the location of the eye and the probability of the eye being occluded by eyelid. At each iteration of cascaded regression, image features from the eye center as well as contextual image features from eyelid and eye corners are jointly used to estimate the eye position and openness probability. Using the eye openness probability, the most likely eye state can be estimated. Since it requires large number of facial images with labeled eye related landmarks, so the proposed method is to combine the real and synthetic images for training. [8]

Pasarica presented two pupil detection algorithms (based on Starburst and circular Hough transform) for the eye tracking application. The operation of the eye tracker device depends mainly on the infrared video camera performances, PC processing power of the video signals and the precision of the pupil detection algorithm. Both algorithms used for the eye tracker implementation have been comparatively analyzed in different conditions for a set of subjects in the field of assistive technology. Both the algorithms had good performances for the desired application, but the Starburst algorithm offers better accuracy in cursor movement with higher noise levels due to the high disparity of the pupil center position from frame to frame. [9]

Ke-ke GU proposed new method for pupil detection on the basis of contour tracking. The method scans the binary pupil image to find the starting point of the contour, and then find the complete contour from the starting point. Calculate the contour area to judge if it is eyelash or pupil area. Continue to search until the pupil contour is found and then calculate its diameter and center. According to the pupil information of the previous frame, narrow the detection region for binarization and track the contour to get the pupil information of the current frame. Threshold for binarization can be fixed, in this case is the threshold of the first frame of the cropped image. Experimental results show that the method has high accuracy, stable performance, and can meet real-time detection speed. [10]

3. ORIGINALITY

This paper proposed a simple technique by using glasses with embedded camera which will only capture the eye part. By using this technique, it will make the algorithm process the eye detection faster because it does not have to remove the unnecessary part of the image. In the hardware that has been created, which is a pair of eye glasses, there are four points that can be adjusted. So, these eyeglasses will work for every user that want to play the game. And also, there is a calibration for the user before they play the game. This will make the algorithm adapt to every user. This calibration will compare the eye pupil movement area size and the desktop screen size.

4. SYSTEM DESIGN

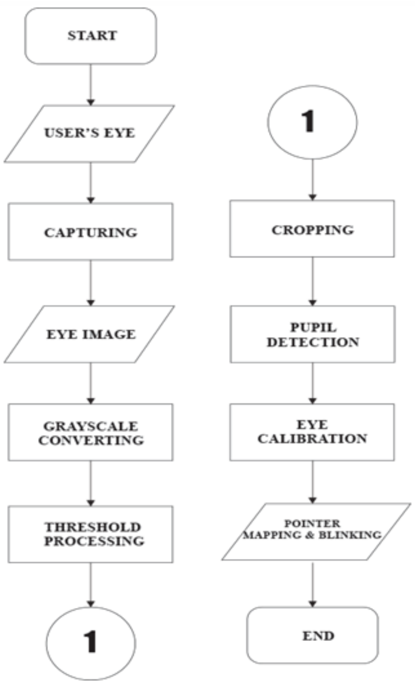

Figure 1. Flowchart of system design

The proposed system consists of 7 process, there are (1) eye image capturing, (2) grayscaling, (3) thresholding, (4) cropping, (5) pupil detection, (6) pointer mapping, and (7) blink detection. The overall design of the system is shown in Figure 1. Each part of the system will be described in Section 4.1 – 4.7.

4.1 Eye Image Capturing

Eye image captured by the mini camera that is mounted to the hardware. On the calibration scene, user will have the option to choose the right camera. The image that is captured by the camera is shown on Figure 2.

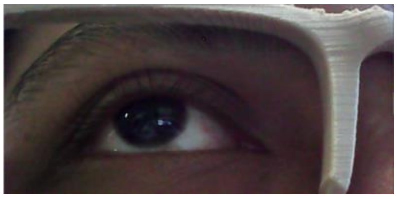

Figure 2. Eye region capturing result

In this research, a hardware is needed to be able to play the game. This hardware is a small camera that is mounted on a pair of glasses. There are four parts of the glasses that have an adjustment. These parts are adjusted to fit the camera position on any head size. The glasses parts detail can be seen in Figure 3.

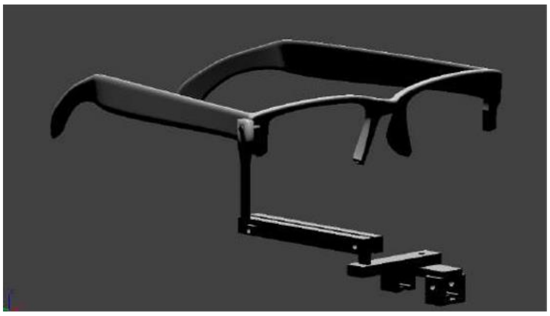

Figure 3. Eye glasses hardware design

The glasses have the dimension of 17.5 cm length, 16 cm width, and 5.5 height. The purpose of using these glasses is to help the camera to stay focus on the user's eye. It will eliminate most of the head parts that will not be used for the eye tracking. By wearing it, capturing pupil image will be clear without worrying that the user has to face to the front.

Table 1. Camera specification

Camera Type	USB 2.0 Endoscope Camera
Pixel	3.0 Megapixel VGA COMS
Resolution	640 x 480
Frame Rate	30 fps
Angle Degree	70 degrees
Camera Dimension	7 x 7 x 25 mm

The camera that has been mounted on the glasses is then integrated with the game that has been created by connecting the camera cable via the Universal Serial Bus (USB) port of the computer. Each frame of the connected camera frame is then read as input and then the image is processed using a method which is able to get center of pupil location changes which is used as the direction of the cursor's motion and the angle of the open eyelid as a shot command.

4.2 Grayscaling

In every pixel in the picture has RGB. The function of color difference is to show the different area, shape, contour, or the highlight and shadow of the picture. Every component has 8-bit value [11], if there is an image editing, it will take about 24-bit. It will make the process slower. Besides that, the color vibrance will make the system difficult to process. This statement underlies the uses of pixel conversion from 24-bit to 8-bit. In other word, every pixel is represented by one color. This method called grayscale process. The formula of this method is shown below.

$$x = \frac{r+g+b}{3} \tag{1}$$

$$x = a_r.r + a_g.g + a_b.b \qquad (2)$$

$$a_r + a_g + a_b = 1 \qquad (3)$$

Picture before grayscaling process and after grayscaling process can be seen on Figure 4.

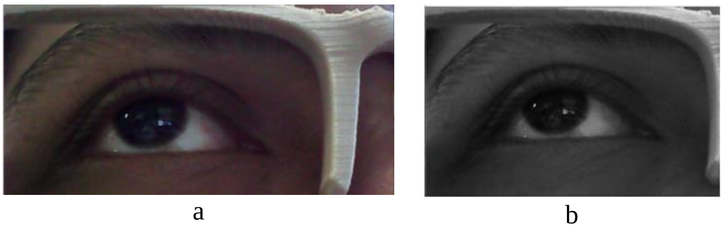

Figure 4. a. Image before grayscaling process and b. Image after grayscaling process

4.3 Thresholding

Segmentation involves the separation of images into regions (contours) that match the object. In this research, the segmentation process is usually done by identifying general or similar properties such as identifying contour differences between regions (edges) [12]. The simplest pixel data properties to process are intensity, so they can be segmented based on the intensity of using the thresholding process for the determination of light and dark areas. Threshold creates a grayscale image into a binary image by altering all pixels that are below the threshold value to 0 and all pixels above the threshold value to 1. This can be written as the following equation.

$$\begin{aligned} x_{bw} &= 1 \quad if\ \ x > threshold \\ &\ \ 0 \quad if\ \ x < threshold \end{aligned} \qquad (4)$$

The image before thresholding process and after thresholding process can be seen in Figure 5.

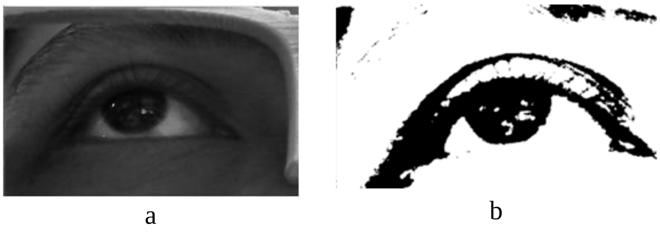

Figure 5. a. Image before threshold and b. Image after threshold process

4.4 Cropping

Image cropping is a general method to enhance the quality of the image by trimming unnecessary part from the image. In the Learning the Change for Automatic Image Cropping research [13], it proposed an automatic cropping technique for an image. Its approach that is done by the method is a data training that includes the image before and after cropping process to get an image of interest from the image. But, this research used a Mean method by sum the black pixels on every rows and column of the image. Then it analyzed based on histogram data. When the value of Mean result reach 20% before the peak position, it is called starting position. When the value of Mean result below 20% after the peak point, it is called ending position. Then, the image is cropped started on starting point and ended on ending point. The image before cropping showed as in Figure 6.

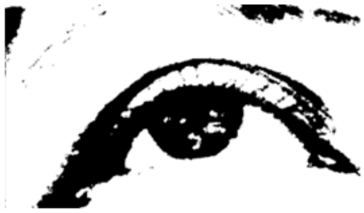

Figure 6. Eye image before cropping process

The horizontal histogram of eye image before cropping process can be seen in Figure 7.

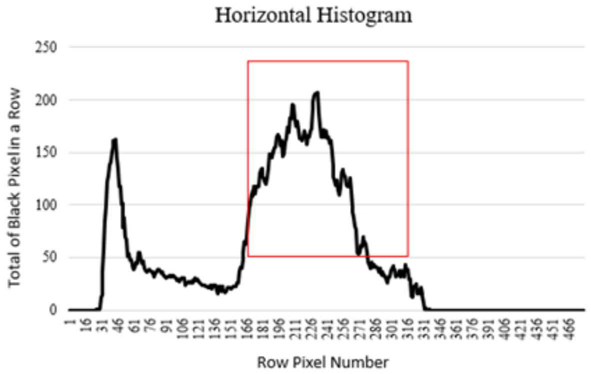

Figure 7. Histogram of eye image in horizontal

Based on the graph, the biggest region is the eye region. The eye region showed by red box. The result of cropping process can be seen in Figure 8.

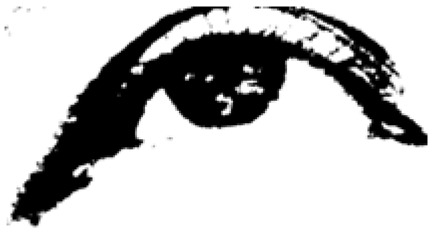

Figure 8. Eye region after cropping process

4.5 Pupil Detection

The pupil location will be determined based on the centroid using Mean method in which the middle value of the image symbolizes the pupil position. The pupil position consists of two positions, horizontal position and vertical position. Pupil position is detected according to the following equation.

$$Mx = \frac{\sum_x x.h(x)}{\sum_x h(x)} \tag{5}$$

$$My = \frac{\sum_y y.v(y)}{\sum_y h(y)} \tag{6}$$

Where Mx is mean value of x, My is mean value of y, x means current position of checked x pixel, y means current position of checked y pixel, h means the total of black pixel in each row and v means the total of black pixel in each column. Mx represents center pupil position in horizontal and My represents center pupil position in vertical. System will track the pupil position in real-time. The result of detection can be seen in Figure 9.

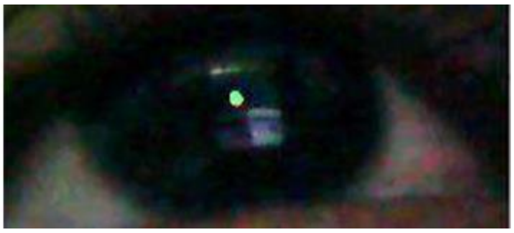

Figure 9. Eye pupil center detection

4.6 Pointer Mapping

Mapping is done by converting pupil coordinates in the eye to the pointer coordinates in the game. First, the user must perform calibration by looking towards the top left, then perform the calibration by looking to the bottom right. This calibration data symbolizes the wide area of eye coverage. When the pupil position is on the top left, then the pupil position will be calculated using Mean method so that the value of x and y are then called as x1 and y1. The pupil position can be shown by illustration in Figure 10.

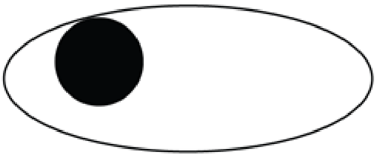

Figure 10. Top-left eye position calibration

Then when the pupil position is in the lower right, the pupil position will be calculated using the Moment Invariant method so that the x and y values are then called x2 and y2. The pupil position can be shown based on the illustration in Figure 11.

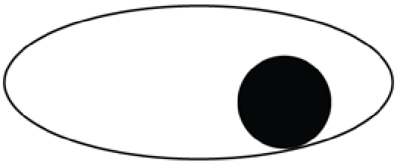

Figure 11. Right-bottom eye position calibration

Furthermore, the difference value between x1 and y1 with x2 and y2 are calculated to obtain the dx and dy values. After that, the positioning of the cursor on the game is determined using the following equation.

$$xPointer = \frac{(EyePositionX - X1).\,Detected\ Area\ Length}{dx} \tag{7}$$

$$xPointerGame = \frac{xPointer}{Detected\ Area\ Length}\ Game\ Area\ Length \tag{8}$$

$$yPointer = \frac{(EyePositionY - Y1).\,Detected\ Area\ Height}{dy} \tag{9}$$

$$yPointerGame = \frac{yPointer}{Detected\ Area\ Height}\ Game\ Area\ Height \tag{10}$$

After pointer has been mapped, the pointer will be used in game that had been designed. The game that is used for the research is an action game with first-person shooter (FPS) genre. The game design is a 3D game that has a view angle limit of about 180 degrees from the starting point. Movement towards other angles can be done while still looking to the right until the camera moves to the right corner or left until the camera moves to the left corner. Since the game is made specifically for quadriplegic children so there will be restrictions applied. The use of game genres in research only focuses on FPS games based on limited eye movements and aims to reduce eye fatigue due to excessive movement. The number of enemies that come and the frequency is limited to one enemy for every 10 seconds. In addition, the background selection in the game uses dark mode for eye comfort in viewing the screen for a long time. This game will run on the desktop. In this game, the player will stand still on one spot. The player has to look around to find any appearing enemies. To move the camera, the player has to move the eye's pupil to the intended direction. It can be left, right, up, or down. The player also has to blink in order to shoot the enemies. The game will be over when the player can withstand five waves of enemies or when the player's HP drops to 0. The preview of game can be seen in Figure 12.

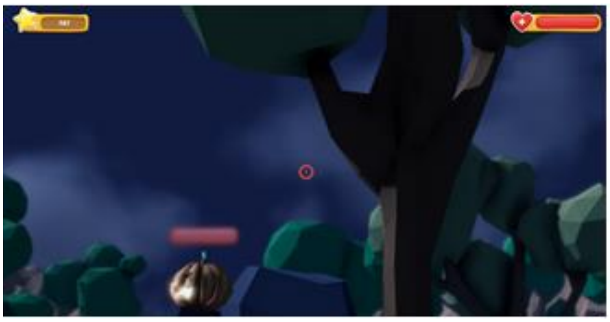

Figure 12. Game preview

4.7 Blink Detection

Blinking is a condition when the eye is temporary closed. There are two types of blinking, intentional blink and unintentional blink. Unintentional blink is a condition where the user blink normally as a reflex of natural eye behavior. The blink intensity of normal blink is affected by user condition based on the interaction that they are doing at the moment. For example, when the user is resting or when the user is reading [14]. Whereas, intentional blink is a condition when the user blinks on purpose. On this research, the intentional blink was used because the intensity can be set by the user. The intentional blink will be used as shooting trigger in game.

Blink detection is done by calculating the ratio of eye area when opening and closing. The way to calculate eye area is using histogram and determine the upper eyelid which means the first black pixel data in horizontal histogram of image and lower eyelid which means the last black pixel data in horizontal histogram of image.

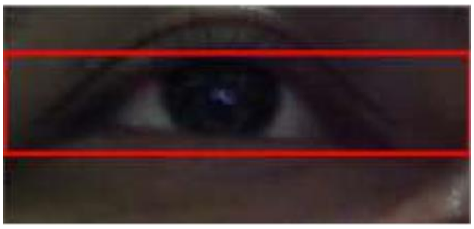

Figure 13. Eyelid Detection

The normal eye area is obtained when performing eye calibration by looking to the middle of the screen. This area is then referred as the maximum area. Normal eye or opening eye calibration illustration can be viewed in Figure 14.

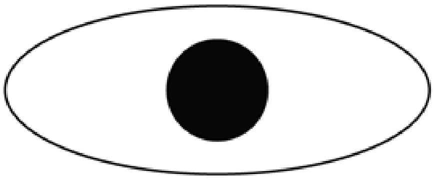

Figure 14. Opening eye illustration

The area of the closed eye is obtained when performing a closed eye calibration. This area is then referred as the minimum area. The closing eye calibration illustration can be viewed in Figure 15.

Figure 15. Closing eye illustration

Furthermore, the maximum area will be subtracted by the minimum area which is then called the blink area. The current area will be taken continuously. If the current area is less than 20% of the blink area then the duration of the blink will be started. When the current area has exceeded 20% of the blink area, the duration counting will be stopped then determined whether the duration is in the range of threshold. If so, it will be considered as a blink, otherwise it is not a blink. The duration threshold is the time obtained based on several experiments to get the duration range of some users. Further details can be seen in the following equation.

$$Blink\ Threshold = \frac{(Maximum\ Region - Minimum\ Region)}{100} x20 \qquad (11)$$

5. EXPERIMENT AND ANALYSIS

The experiment and analysis consist of (1) blink comparison, (2) user motion speed, (3) blink rate response and (4) eye movement accuracy. Experiments in this study were performed under even room lighting conditions so that the light illuminated the area around the eyes well.

5.1 Blink Comparison

The blinking difference is determined by the duration when the current eye area is below the 20% of the threshold until it has exceeded 20%. Blink duration data are obtained from 10 users with 7 experiments. Unintentional blink duration data in millisecond are presented in Table 2.

Table 2. Unintentional blink data

User	Average Duration (milliseconds)	Min Duration (milliseconds)	Max Duration (milliseconds)
1	155.77	82.95	218.20
2	101.13	82.82	116.38
3	130.85	82.85	215.42
4	108.74	82.83	215.11
5	104.29	82.45	199.01
6	96.39	82.53	115.96
7	139.63	81.94	182.23
8	118.15	79.34	148.12
9	128.78	95.88	183.41
10	133.26	92.32	168.14
Average	121.70	84.60	176.20

Based on the table 2, it can be seen that the average of unintentional blink duration is 120.70 millisecond with the fastest value is 84.60 millisecond and the slowest value is 176.20 millisecond. Furthermore, the data of intentional blink duration in millisecond are presented in Table 3.

Table 3. Intentional blink data

User	Average Duration (milliseconds)	Min Duration (milliseconds)	Max Duration (milliseconds)
1	402.94	215.36	596.48
2	230.3	182.74	513.57
3	296.53	281.56	314.67
4	269.99	199.39	612.97
5	358.00	198.53	596.40
6	220.74	175.97	397.60
7	288.14	229.14	336.76
8	324.58	223.82	416.84
9	302.82	218.79	546.24
10	309.53	207.98	582
Average	300.36	213.33	491.35

Based on the table 3, it can be seen that the average intentional blink duration is 300.36 millisecond with the fastest value is 213.33 millisecond and the slowest value is 491.35 millisecond. Based on these data, the range value of unintentional and intentional blink can be used as a reference of threshold duration.

5.2 User Motion Speed

The controller can not be separated with the speed of movement which is an important factor that should be measured for user interaction. This speed of motion will later determine how difficult a game is designed. It aims to make games that will not be too difficult or too easy for the user. This experiment was performed as much as 7 times experiment on 10 users which includes 4-way motions. Each of direction movement is performed from the middle position to the determined direction and towards the center back. there will be 8 movement in total of all directions. The average of each experiment value results then will be calculated separately. The results of these experiments are presented in Table 4.

Table 4. User motion speed data

User	Average Duration (miliseconds)	Min Duration (miliseconds)	Max Duration (miliseconds)
1	2584.14	1735.04	4575.07
2	2040.61	1357.41	3026.10
3	2466.08	1213.37	5429.62
4	2674.50	1536.51	4233.70
5	1692.40	1246.87	2814.09
6	2450.64	1901.76	3736.03
7	2611.69	1527.15	4921.01
8	1881.91	1477.38	2819.90
9	1679.74	1378.61	2205.93
10	2085.90	1555.22	2540.37
Average	2216.76	1492.93	3630.18

Based on the table above, it can be seen that the average player's movement requires 2.22 seconds, the fastest movement requires a duration of 1.49 seconds and the slowest motion requires a duration of 3.63 seconds. This minimum and maximum data can later be used as a reference for developing a game difficulty level. Further discussion will be explained as in Figure 16.

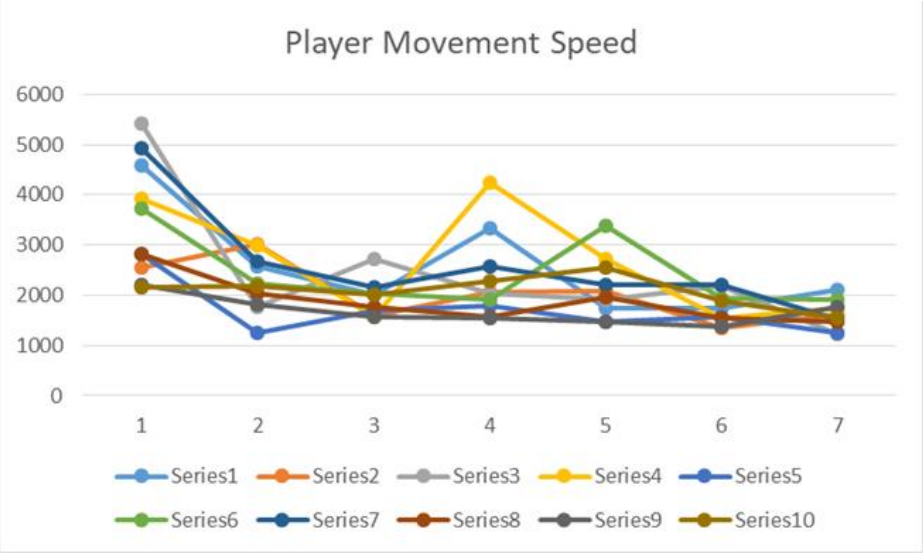

Figure 16. Player movement speed graph

Experimental data 1 to 7 are shown on the X-axis while motion speed is shown on the Y-axis in milliseconds. From the data, it can be concluded that the average player is likely to experience difficulties or need a fairly high time when first trying to make movements with the eyes, although this does not apply to user 4 during the fourth experiment which requires a longer duration than the first experiment. This is known to be caused by the player's lack of focus when making movements. Lack of concentration also caused a spike in the duration data on user 5 in data-4 and user 6 in data-5 , user 3 in data-3 and user 7 in data-4. On the other hand, the user showed a decrease in duration and began to stabilize in the sixth experiment. From the graph it can be concluded that each user has a different number of variations to adapt to the eye control system.

5.3 Blink Rate Response

Blinking is one of the features used in this study which serve as a reference for triggering shooting in the game that has been made. The player is marked as blinking when the current eye area drops to less than 20% of the blink area. Based on this data, the graph of the data from the blink response are presented in Table 5.

Table 5. Blink rate response data

Data	Time (ms)	Average (ms)	Min (ms)	Max (ms)
1	16.7	26.33	16.5	100
2	21.2			
3	16.7			
4	18			
5	16.6			
6	16.6			
7	16.5			
8	68.5			
9	16.7			
10	16.6			
11	16.5			
12	19.5			
13	100			
14	18.4			
15	16.5			

The response of the blink is calculated starting from the difference in time between the second maximum point and the time before the maximum point. The graph above shows that the average of the blinking response rate requires the fastest 16.5 milliseconds and the longest 100 milliseconds. The above experiment shows that the average response time towards system is 26.33 milliseconds

5.4 Eye Movement Accuracy

The accuracy of eye movement is measured by some experiments. The experiments are performed by calculating how many points of motion deviate from the motion path. We obtain the data from 4 movements, there are center to right, center to top, center to left and center to bottom. This research was conducted using four directions based on the eyes that has a tendency to move frequently to the right, left, top and bottom so that there are four movements and the rest are a combination of several major movements. The user moves the cursor to a straight point where when moving in a straight line according to the horizontal path, then the vertical deviation is an error and so is the opposite axis. When the cursor is on the line, the value of the position is zero, while when the cursor is at the farthest boundary of the track, the position is 100. The determination of the error value is determined using the following equation.

$$Error\ Percentage = \frac{cursorposition\ average}{wide\ area} * 100 \tag{12}$$

Each user takes 7 times experiment. The average error movement data of each direction are presented in Table 6.

Table 6. Eye movement accuracy data

Eye Movement Error Percentage (%)					
	Error Average For Each Direction				
User	Left	Right	Top	Bottom	Average
1	8.32	6.76	18.94	4.20	9.55
2	32.34	12.29	22.64	8.28	18.88
3	8.60	13.20	16.00	15.00	13.20
4	33.52	16.00	24.35	6.40	20.06
5	18.30	13.12	14.98	21.89	17.07
6	14.41	10.64	17.57	17.30	14.98
7	10.85	9.69	14.81	14.82	12.54
8	13.95	14.48	16.92	15.45	15.20
9	14.24	10.27	17.89	16.62	14.75
10	10.95	13.22	17.42	18.75	15.08
Average					15.13
Min					9.55
Max					20.06

The eye movement shows the minimum error is 9.55 percent and the maximum of error is 20.06 percent and the average of error is 15.13 percent which means the accuracy is 84.86 percent with the maximum accuracy is 90.45 percent and minimum accuracy is 79.94.

6. CONCLUSION

Based on the above experiments, we get several conclusions. Eye detection that was made successfully was used as eye controller which through experiments obtained user speed data with a duration of about 2.23 seconds. The data shows that new user needs adaptation for the first interaction, the user tends to consume more time than the average. The shooting response speed based on the blinking takes duration about 0.026 seconds. The accuracy of eye movement is 84.86 percent. The average intentional blink duration is 0.30 seconds and the average of unintentional blink duration is 0.12 seconds. The intentional blink takes more time than unintentional blink that can be distinguished and used as shooting interaction.

The future works of this research are to create the blink noise removal and the acceleration of eye movement speed. Blink noise removal will be used when a blink happens, the average position of pupil still considered as the eye position by algorithm, the cursor will move accidentally to another position and make the game will shoot unintentionally at last position after a blink happens. The acceleration of eye movement will be used when a player looks at less than maximum position or less than minimum position of calibration, the speed of cursor will be increased based on distance difference. Further research on the effect of light is also needed to discover how much the level of light sensitivity affects the accuracy of eye movement to the cursor and blink detection to the shots.

REFERENCES

[1] Nur Fadhilah Al-Karimah, **Subjective Well-Being pada Penyandang Tuna Daksa**, *Jurnal Penelitian dan Pemikiran Psikologi*, Vol. 13, No. 1, pp. 57-64, 2018.

[2] Word Health Organization, **World Report on Disability**, *WHO Press (Geneva)*, 2011.

[3] Dedy Ariyanto, **The Role of Learning Technology in Supporting the Implementation of Inclusive Education**, *International Conference on Special Education in Southeast Asia Region*, pp. 381-386, 2017.

[4] Linda A. Jackson, Edward A. Witt, Alexander Ivan Games, Hiram E. Fitzgerald, Alexander Von Eye, Yong Zhao, **Information technology use and creativity: Finding from The Children and Technology Project**, *Computers in Human Behaviour*, Vol. 28, pp. 370-376, 2011.

[5] Somaya Adwan, Hamzah Arof, **Modified Integral Projection Method for Eye Detection using Dynamic Time Warping**, *International Journal of Computer Applications*, Vol. 8, No. 1(A), pp. 187-200, 2012.

[6] Muhammad Usman Ghani, Sarah Chaudhry, **Gaze Pointer: A Real Time Mouse Pointer Control Implementation based on Eye Gaze Tracking**, *International Multi Topic Conference*, pp. 154-159, 2013.

[7] Tereza Soukupova, Jan Cech, **Real-Time Eye Blink Detection using Facial Landmarks**, *Computer Vision Winter Workshop*, pp. 1-8, 2016.

[8] Chao Gou, Yue Wu, Kang Wang, Kunfeng Wang, Fei-Yue Wang, Qiang Ji, **A Joint Cascade Framework for Simultaneous Eye Detection and Eye State Estimation**, *Pattern Recognition*, Vol. 67, pp. 10-13, 2018.

[9] A. Pasarica, R. G. Bozomitu, V. Cehan, R. G. Lupu, C. Rotariu, **Pupil Detection Algorithm for Eye Tracking Applications**, *International Symposium for Design and Technology in Electronic Packaging*, pp. 161-164, 2015.

[10] Ke-ke Gu, **Real-time Pupil Detection based on Contour Tracking**, *Current Trends in Computer Science and Mechanical Automation*, Vol. 2, pp. 31-40, 2017.

[11] Tarun Kumar, Karun Verma, **A Theory Based on Conversion of RGB image to Gray Image**, *International Journal of Computer Applications*, Vol. 7, No. 2, pp. 5-12, 2010.

[12] Khalil Ibrahim Alsaif, Raghad Hazim Hamid, **Study the Effect of Threshold Value on Object Detection**, *International Journal of Computer Applications*, Vol. 179, No. 16, pp. 10-13, 2018.

[13] Jianzhou Yan, Stephen Lin, **Learning the Change for Automatic Image Cropping**, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 971-978, 2013.

[14] Artem Lenskiy, Rafal Paprocki, **Blink Rate Variability during Resting and Reading Sessions**, *IEEE Conference on Norbert Wiener in the 21st Century*, pp. 90-95, 2016.