Real-Time Hand Gesture Recognition for Embedded System

Farah Farhana Mod Ma'asum, Suhana Sulaiman, Azilah Saparon

Abstract— **A system with low-cost hardware computer webcam as the replacement of mouse click is being applied in this research. In order to capture good image of hand in vision based system, various segmentation techniques proposed by other researchers are combined and tested to enhance the quality of segmentation image. Canny edges and Otsu threshold technique are used to segment the hand image while convex hull and convexity defects algorithm are used to extract the image of hand features. Embedded hardware (Arduino) board is employed for validating the signal sent using hand gesture to replace LEFT CLICK, RIGHT CLICK, MOVE cursors. An experiment is set up to determine the accuracy in percentage of this work with ten test subjects. They were prearranged for five minutes to become familiar with the hand tracking system after the initial attempt. The findings revealed that users are better trained in the second trial after five minutes training. The results significantly improved from 33.3 % to 52.6 % for LEFT CLICK, 46.7% to 61 % improvement for RIGHT CLICK while 56.7% to 77.3% for MOVE cursor.**

Index Terms— **segmentation image, convex hull, convexity defects algorithm, Embedded hardware,**

I. Introduction

Development of computer technology has progressed tremendously since it has become an essential component in human daily life. Personal computers were invented in the 80s where they were equipped with keyboard, mouse and panel display. However, as time goes by, the size and weight of computer have decreased and reduced. Even when touch screen panel has been developed as the combination of display panel, keyboard and mouse, there are still people who find this unbeneficial for them. For example, people who suffer hyperhidrosis will find it is hard for them to fully control the movement of cursors due to excessive sweats produced from the skin. Not only that, touch screen panel is also unfriendly for disabled people since they required some assistance to perform some activities if they want to use it.

Various approaches of Human Computer Interface, or known as HCI have been proposed as the replacement of input devices – keyboard and mouse. Wired gloves, depth cameras and web camera are few tools of vision-based HCI which could be used to interpret human gestures, have their own advantages and disadvantages. Wired gloves, for example, provide input to the computer where the position and rotation of hand are being captured and then being analyzed based on the degree of finger's flexion [1]. However, wired gloves do have some disadvantages since they hinder naturalness behavior between human and computing device. The glove requires wearing load of cables in other to transmit information of hand position. According to S.S. Rautaray *et. al* [2], by using data glove for example, it created factors such as awkwardness, unintuitive, rigidity and prone to distortion from the physical environment. This contact based device are usually used by experienced users since it does not provide much acceptability [3], due to its highly cost [2] .

Another example of HCI which is being used by other researchers is depth camera. Depth camera comprises of structured light or time-of-light (TOF) camera which generates a depth image for a short distance range [4]. Therefore, it has the advantage of producing better segmentation image compared to others. Example of device which instilled depth camera is Microsoft Kinect, where it consists of RGB-D sensor which acts as a controller in Xbox game console. Combinations of RGB camera, depth sensor and four-microphone array as depicted in Figure 1, produced depth signals which enable accurate image segmentation. Kinect does have some key features which is useful in a larger scale compared to other gaming consoles such as facial recognition system that uses a person's physical features for security purposes. Imagine a replacement of touch screens concept employing Kinect concept. It will provide advantages to users as this will be a natural user interface (NUI). However, Kinect device is impossible to be installed on the hand phone due to its size, dimension and weight. In addition of its price, depth camera is quite expensive compared to normal web camera.

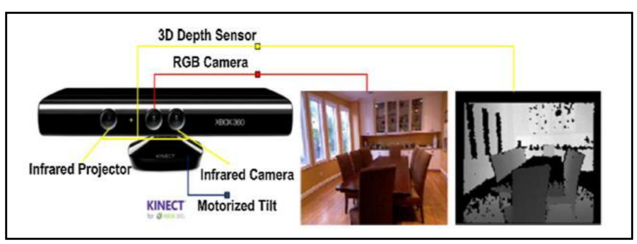


Figure 1 : Hardware configuration of Kinect and images captured by RGB camera and depth sensor [5]

This manuscript is submitted on 3 January 2018 and accepted on 26 April 2018. The work is partly the MSc tesis of Farah Farhana who is awarded the Young Lecturer Scheme scholarship from Universiti Teknologi MARA Shah Alam, Malaysia. This work was supported in part by Ministry of Higher Education, Malaysia. Farah Farhana Mod Ma'asum is currently with Intel Penang as Validation Engineer. (email: farah.farhana88@yahoo.com).Suhana Sulaiman is with Facuty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Malaysia (email: suhana832@salam.uitm.edu.my).Azilah Saparon is with Facuty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Malaysia (email: azilah574@salam.uitm.edu.my)

This work is aimed to demonstrate an alternative system where robustness of Microsoft Kinect can be reproduced using low-cost hardware. In short, the system does not require any other hardware device except built-in camera itself. To capture good image of hand in vision based system, various segmentation techniques have been proposed by other researchers - Canny edges and Otsu threshold technique. These techniques are used to segment the hand image while convex hull and convexity defects algorithm are used to extract the image of hand features. The authentication of signal sent by the hand gesture for LEFT CLICK, RIGHT CLICK, MOVE cursors are validated with embedded hardware (Arduino) board.

II. LITERATURE REVIEW

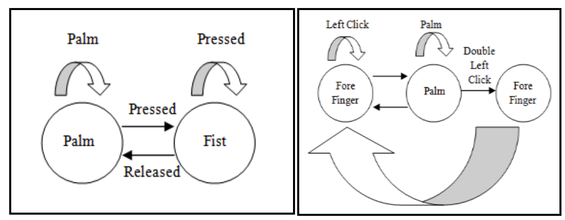
Recently, researchers have shown an increase of interest in hand gesture recognition in vision-based interaction. Vision has the potential of carrying wealth information in a non-intrusive manner at low cost, which helps in developing good modality of hand gesture recognition [6]. Various techniques have been used in image acquisition, image segmentation and feature extraction respectively.

Skin color model works as an image acquisition method where it consists of different types of model. H. Jalab *et. al* [7] employed conversion of RGB to Lab color Space (CIE mode) on the image data set of hand gesture images. M. K. Ahuja *et. al* [8] on the other hand preferred to apply skin color conversion from RGB to YCbCr on 20 images per gestures obtained during image acquisition process. All skin color model has its own benefits, based on its purpose. For example, YCbCr works best in video processing since it can solve instability issue faced by RGB color space.

After an image acquisition and segmentation process, location of hand and fingertips are determined by feature extraction process. Most of the researchers preferred to create their own algorithm in order to distinguish which one are hand, palm and fingertips [9], [10], [11]. T. Ikai *et. al* [11] creates algorithm in determining the finger direction in 3D space background. Using his Finger Direction Recognition (FDR) algorithm, estimation of two finger directions is determined: finger direction on photographic surface and finger direction of depth. T. Ikai also defines a pointing finger as "the farthest finger from gravity center of hand region".

H. S. Yeo *et. al* [12] and A. Dhawan *et. al* [13] on the other hand applied convexity defect technique in determining fingertips contours. Yeo [12] has proved that user can interact and play with games or PC applications by hand gestures instead of relying with gaming physical controllers. In his research, YCrCb threshold has been used to overcome low light limitation in the usage of single camera. Advantage of using Kinect with depth sensor is proven in his research where robustness of hand image is utilized. However, Kinect has the disadvantages of sizes and price, which it is actually a limitation to users. Therefore, implementing the concept of Kinect by using low-cost hardware is the main objective in this paper.

Finite State Machine (FSM) is another type of feature vector representation which frequently used by other researches. FSM is basically uses string matching between a data sequence and state sequence of an FSM [14] [15]. There are 4 different postures that are needed to be recognized in this research, and leads to 3 mouse actions which are "mouse move", "left click" and "right click". The transitions between these 3 mouse actions are important so that correct mouse actions are recognized. Figure 2 exhibits the FSM graphic of the mouse actions in Aksac *et. al* [15] research. Figure 2(a) shows when the posture turns from palm to fist or remain as fist, it is defined that the mouse is pressed. On the other hand, if posture returns to palm from fist it is described as mouse released action. For mouse left click action, it is described when the oblique forefinger posture is detected in a frame. There is timer being triggered during this action. But if the action is repetitively executed within less than 500 milliseconds, the action is labeled as double left click action. Similar concept with different fingertip is employed to determine both right click and double right click action.



(a) (b)

Figure 2: Finite State Machine (FSM) in [15]

Combinations of both methods with little advancement are considered in this research. Furthermore, the verification of the feature extraction method and classifier was completed with the software-based technique. One of the issues that emerged from these findings is to integrate the gesture recognition action with hardware-based effect. Therefore, this work provides a framework for the exploration of the integration of hand gesture recognition with hardware.

III. RESEARCH METHODOLOGY

This system consists of two parts: software development and hardware implementation. For the software development, Microsoft Visual Studio 2012 (OpenCV) is used in image processing phases: Image acquisition, Segmentation module and Feature representation module as depicted in the flowchart in Figure 3. While Matlab R2010a is used in comparing which techniques produced the best results in segmentation module. The hardware implementation Arduino is used to validate the hand gesture by employing LED for the gesture representation.

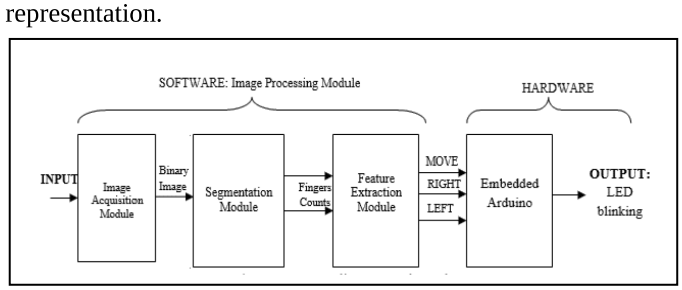


Figure 3 : Flowchart for the overall system

A. Image Acquisition

Image acquisition is an important module to obtain hand gesture image even in poor illumination and different lighting conditions[2]. This is to ensure that the hand gesture image is acceptable for further examination in following modules.

Furthermore, this allows the system to operate within the real-time constraints. For this work, a single built-in camera based method provides as the main objective which is to achieve robustness in capturing images of hand gesture as depicted in Figure 4.

Figure 4: Image of hand gestures is being captured by built in camera [16]

B. Image Segmentation

Image segmentation is the process of partitioning an image into understandable non-overlapping regions. Based on skin information, the input colored image is classified into skin and non-skin pixels [17]. This method is particularly useful in converting gray scale image into binary image so that there will be only two objects in image: one is the hand and the other one is the background [18]. The segmentation starts by converting the input image, which has been captured and stored in the acquisition module, to other color space representation module.

i. Color Space Module

There are many types of color space representation module, but in this research, we choose to convert RGB to HSV color module. The conversion of RGB to HSV color module is made because of its attributes correspond directly to the basic color concepts. This makes it conceptually simple for segmentation module since it can discriminate color information distinctively.

ii. Classification of Segmentation Methods

A good segmentation is when there is no part of hand is distorted during extraction process. Types of segmentation techniques are depending on the type of images and application areas. Pixel based segmentation (thresholding) is a technique where intensity values are assigned for each pixel depending on either object pixel or background pixel. Output of threshold method is a binary image with all pixels 0 belong to the background while pixels with 1 represent the hand image.

Fixed threshold is defined as (1):

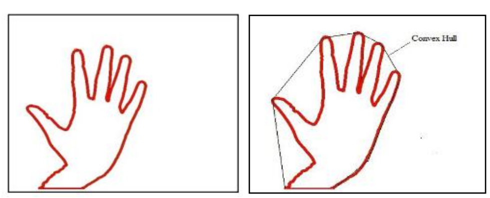
$$g(x,y) = \begin{cases} 0, f(x,y) < T; \\ 1, f(x,y) \geq T \end{cases} \qquad (1)$$

where T is the threshold value of the image pixels, *f(x,y)*

Canny operator with Otsu threshold is further applied on this segmented hand image which will enhance the quality of the image. This is because of the threshold-based segmentation results are usually less than perfect. Furthermore, Canny operator has the ability to detect true weak edges since it does not susceptible to noise interference [19]. Consequently, three criteria must be fulfilled to determine an optimal edge *i.e.* by minimizing error rate, marking edges as closely as possible to the actual edges to maximize localization, and marking edges only when a single edge exists for minimal response [20]. Otsu method is global thresholding which depends only on the gray value of the image, and it is widely used because of its simplicity and effectiveness in segmenting image [21]. Not only that, Otsu algorithm is also a nonparametric and an unsupervised method of automatic threshold selection [22] where most histogram of an image consists of a valley between two peaks representing objects and backgrounds respectively. Comparison between all these three segmentation images are elaborated in the next section for further understanding.

C. *Contour Extraction*

Contouring the image line is the next step to obtain good extraction of hand region. Contour is a list of point representing a curve in an image which represented by lines and displaying the steepness of sloped, valleys and hills [12][13]. Hand contour model is used to simplify the gesture matching process, which helps in reducing the computational complexity of gesture matching. In this research, segmented hand is contoured where user's palm curve is represented by lines. These lines are then being applied with convex hull technique, where a set of points in the Euclidean space is the smallest convex set that contains all the set of given points [13]. This can be further visualized as the shape formed by the rubber band stretched around a set of points. Results gained from convex hull technique are then implemented with convexity defects which produced points of fingertips to determine hand gesture feature vector.



)

(a) (b)

Figure 5: (a) Image of contoured hand; (b) Image of convex hull being applied on contoured hand

D. Convexity Defects

Once the convex hull is drawn around the contour of hand image, convexity defects are then being created which produced a set of values in the form of vector. This vector contains the start and end point of the line defect in the convex hull. This research is implementing the similar technique used by H.S. Yeo [12] in his hand tracking research to locate all the important nodes and points.

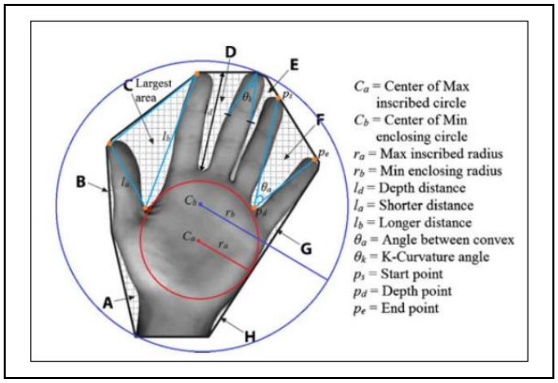


Figure 6: Convex hull, convexity defects of hand shape used by H.S. Yeo [12]

Fingers' locations can be determined by implementing convexity defects method in this research, using following steps:

a) Depth of each defects (ld) must be longer than palm radius (ra), but shorter than minimum enclosing circle radius (rb)
b) Angle (θa) between start point (ps) and end point (pe) must be less than 90°

E. Feature Classifier

All data obtained from convexity defects are then categorized based on its specific feature vectors to determine the gesture functions. For instance, angle of index finger's tip and middle finger's tip would create a depth angle. Initially, to identify which finger are detected during experiment, the angle between these two tip points are calculated. If the angle is between 45˚ to 90˚, it can be identified as index finger whereas if the angel in between 90˚ to 140˚ is classified as middle finger.

Once the criteria and location of fingers are detected, classifier then takes part in determining the finger's criteria. The followings are few criterions of gestures that are classified based on the location of fingers in this research:

a) MOVE CURSOR – Two fingers (index finger and middle finger) are pointed for scrolling
b) RIGHT CLICK – Only middle finger is closed after move position, pointing only index finger
c) LEFT CLICK – Only index finger is closed after move position, pointing only middle finger

These criteria show that only two fingers will play the most roles in this research in determining the gesture indicated by user's hand. We only use these two fingers because we would like to reduce the processing time of the classifier algorithm in this research. If all fingers locations are considered, the processing time during classifying will be longer.

IV. Results and Discussion

Throughout this research, static background is being used to eliminate noises from the hand image. Major challenges in gesture recognition is to cope with a wide range of variations in the human skin color. Since there are many ethnicities in the world, segmentation technique by using track bar is the most suitable one to produce better results. Users can slide the track bar per their own preferences.

A. Experiment in Image Segmentation Technique

Figure 7 shows the comparison of hand images after implementing three methods, Sobel method, Canny method with fixed threshold value and Canny method with Otsu threshold value respectively.

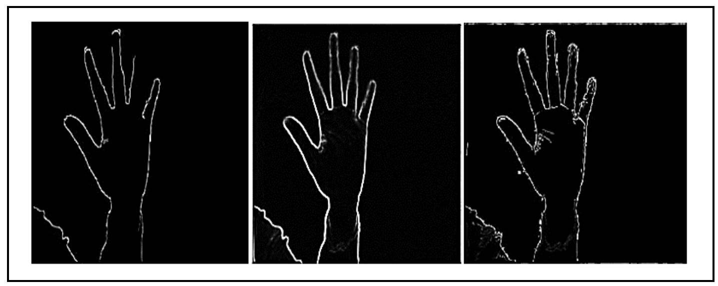


Figure 7: Comparison between edge segmentation on HSV image (a) Canny operator using fixed threshold value 50 ; (b) Sobel method ; (c) Canny using Otsu method

As we can observe from Figure 7, Sobel method produces blurry image which contains unrefined lines between edges. This shows that Sobel technique is not suitable in our analysis since hand image edges will be hard to be defined. When we compared between Canny and Sobel technique, Canny technique produces better result since this technique able to distinguish fine lines either by applying normal threshold value or Otsu threshold value. The result of Otsu threshold managed to detail the fine lines compared to normal threshold image. There is missing boundary lines in normal threshold (Figure 7(b)) compared to image in Figure 7(c) - Otsu threshold. Detailed boundary fine lines are important in our research to eliminate any unwanted noises from the background image.

Hence, to determine which technique is best applied in segmentation method, a comparison based on similarity index has been performed on different segmentation techniques using Jaccard Similiarity Index (Equation 1). This calculation can be further understood with the representation of Figure 8.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \qquad \text{Equation 1}$$

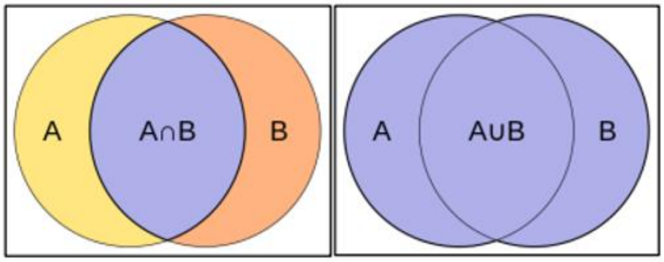


Figure 8: Intersection and union of two sets A and B

The results obtained from similarity between sample sets based on Equation 2 is used and in this calculation, we aimed to gain results which produced higher similarity index with respective segmentation images. Results obtained from this calculation will then be used as an input for contouring hand image.

$$0 \leq J(A,B) \leq 1 \qquad \text{Equation 2}$$

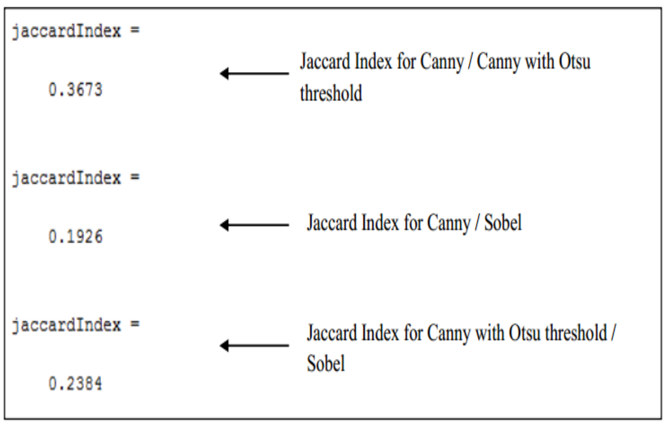


Figure 9 : Results obtained after implying Jaccard Similarity Index using Matlab

As shown in Figure 9, results obtained from Jaccard index shows that similarity index value of Canny segmentation using Otsu threshold is much higher compared to Sobel technique.

Jaccard index obtained between Canny with threshold value: 50 and Canny with Otsu threshold is 0.3673, while Jaccard index obtained between Canny and Sobel image is 0.1926. This proves that boundary lines between Canny with threshold value 50 is better enhanced by using Otsu threshold technique due to relatively small binary overlap between the pixels. To support the discussion obtained above, Fang and Yue [69] also has done similar experiments by applying Otsu operator on Canny edge detection. In their paper, the image of contour line can be clearly obtained when applying Otsu threshold on Canny algorithm.

B. Feature Classifier Module

Three gestures of hand were tested in this study which are MOVE, RIGHT CLICK and LEFT CLICK. In order to capture these gestures, only 2 fingers are required to be defined – middle finger and index finger. Index finger is in bound within less than 90˚, while middle finger angle is more than 90˚. Figure 10 shows the point of finger needed to define it as 'MOVE' gesture. For both RIGHT CLICK and LEFT CLICK require the transition from MOVE action (both index and middle finger) to either only one finger left as shown in Figure 11 and Figure 12 respectively. Good detection of fingertips is important in this step to identify the right action of hand movements. Not only that, the transition of fingertips must be within few seconds so that unnecessary movements can be discarded.

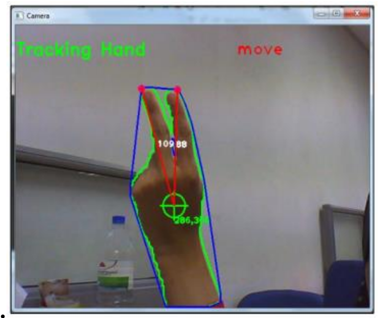


Figure 10: Middle finger and index finger angle for MOVE CURSOR

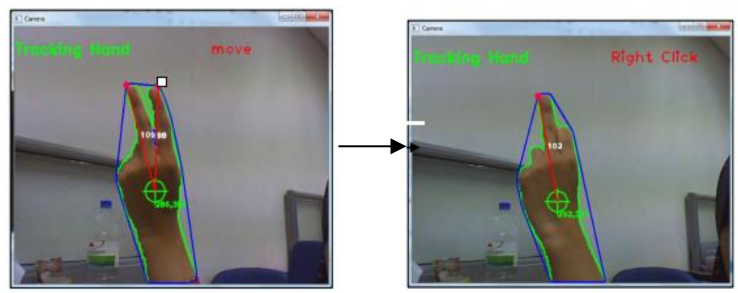


Figure 11: Transition of fingertips to create LEFT CLICK action

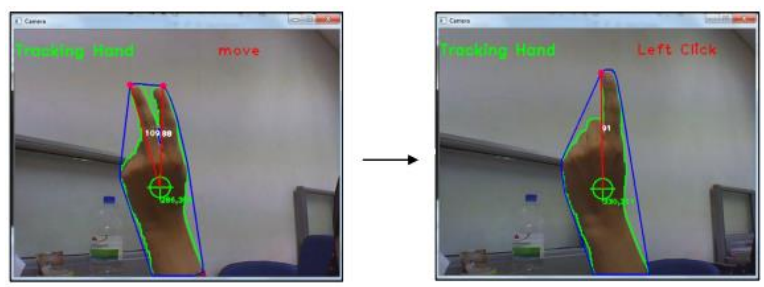


Figure 12: Transition of fingertips to create LEFT CLICK action

C. Accuracy Performance of Feature Classifier

In order to ensure that the system could effectively detect the gesture, an experiment was set up in which ten subjects were required to maneuver their fingers on air with certain task gestures at random locations for 5sec at a time. Their gestures were then displayed on the screen to demonstrate whether they had been represented correctly or not. Each subject had to perform three gestures i.e. MOVE CURSOR, RIGHT CLICK and LEFT CLICK as shown in the previous section, for 30 times without any training. Table 3.1 shows the number of accurate gestures detected during the experiment and it is about 45.6% of accuracy. After that, the subjects were given 5 minutes to get familiar with the hand tracking system and they tried the system again. The results obtained after the training, are improved as depicted in Table 3.2. It gives 63.7% accuracy. It clearly shows that the system could give better performance if users know the correct gesture to be used.

Table 3.1: Number of Accurate Gesture Detected (in percentage)

User	*Move*	*Right Click*	*Left Click*
1	*17/30*	*9/30*	*8/30*
2	*14/30*	*8/30*	*9/30*
3	*20/30*	*19/30*	*7/30*
4	*1/30*	*3/30*	*0/30*
5	*27/30*	*14/30*	*4/30*
6	*15/30*	*28/30*	*20/30*
7	*8/30*	*20/30*	*14/30*
8	*27/30*	*17/30*	*28/30*
9	*25/30*	*13/30*	*5/30*
10	*16/30*	*9/30*	*5/30*
Total Count	*170/300*	*140/300*	*100/300*
Percentage (%)	***56.6667***	***46.666667***	***33.33333***

Table 3.2: Second accuracy test after training (in percentage)

User	*Move*	*Right Click*	*Left Click*
1	*20/30*	*20/30*	*19/30*
2	*24/30*	*17/30*	*18/30*
3	*24/30*	*16/30*	*19/30*
4	*9/30*	*9/30*	*5/30*
5	*30/30*	*23/30*	*13/30*
6	*17/30*	*28/30*	*19/30*
7	*23/30*	*10/30*	*18/30*
8	*30/30*	*25/30*	*25/30*
9	*30/30*	*20/30*	*10/30*
10	*25/30*	*15/30*	*12/30*
Total Count	*232/300*	*183/300*	*158/300*
Percentage (%)	***77.3333***	***61***	***52.66667***

The finding also reveals that 9 out of 10 users were having problem in producing left click. This implies that during the first trial, the algorithm may mistakenly interpret the gesture as right click gesture due to similar angle produced by the index and the middle finger. However, after being trained for 5 minutes, users were able to control the angle of index finger, hence, they were able to produce left click signal easily.

V. CONCLUSION

As for conclusion, this research is to produce a low-cost hand gesture recognition where no added hardware is needed. The hand image which are being captured using built-in camera laptop, are converted into binary using RGB-HSV color space conversion based on tracking bar thresholding technique. Segmentation image is done next to ensure that the image of hand is fully captured and analyzed. Comparison between 3 techniques: Sobel, Canny with normal threshold value 50 and Canny with Otsu threshold value; has proved that by implying Canny with Otsu threshold will enhance the quality of hand segmentation. By applying Jaccard coefficient on these three techniques, it has been proven that Canny with Otsu threshold has higher correctness and stability with respect to the original image. Once after the image of hand has been segmented, convex hull technique and convexity defect technique are applied on contoured hand palm to determine the locations of each finger by calculating angle between fingertips. Angles gain from the fingertips will then determine the gestures pointed by the user. In this research, three types of gestures types are tested such as moving the mousecursor, right click cursor and left click cursor.

REFERENCES

[1] P. Kumar, J. Verma, and S. Prasad, "Hand Data Glove: A Wearable Real-Time Device for Human-Computer Interaction," *Int. J. Adv. Sci. Technol.*, vol. 43, pp. 15–26, 2012.

[2] S. S. Rautaray and A. Agrawal, "Real Time Gesture Recognition System for Interaction in Dynamic Environment," *Procedia Technol.*, vol. 4, pp. 595–599, 2012.

[3] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: a survey," *Artif. Intell. Rev.*, vol. 43, no. 1, pp. 1–54, 2015.

[4] P. Krejov and R. Bowden, "Multi-touchless: Real-time Fingertip Detection and Tracking using Geodesic Maxima," *IEEE Int. Conf. Work. Autom. Face Gesture Recognit.*, pp. 1–7, Apr. 2013.

[5] J. Han, L. Shao, D. Xu, and J. Shotton, "Enhanced Computer Vision with Microsoft Kinect Sensor : A Review," *IEEE Trans. Cybern.*, vol. 43, no. 5, pp. 1318–1334, 2013.

[6] G. Murthy and R. Jadon, "A review of vision based hand gestures recognition," *Int. J. Inf. Technol. Knowl. Manag.*, vol. 2, no. 2, pp. 405–410, 2009.

[7] H. A. Jalab and H. K. Omer, "Human computer interface using hand gesture recognition based on neural network," *5th Natl. Symp. Inf. Technol. Towar. New Smart World*, pp. 1–6, 2015.

[8] M. K. Ahuja and A. Singh, "Static Vision Based Hand Gesture Recognition Using Principal Component Analysis," *IEEE 3rd Int. Conf. MOOCs, Innov. Technol. Educ.*, pp. 402–406, 2015.

[9] J. L. Raheja, A. Chaudhary, and K. Singal, "Tracking of Fingertips and Centers of Palm Using KINECT," *Int. Conf. Comput. Intell. Model. Simul.*, vol. 3, pp. 248–252, Sep. 2011.

[10] Y. She, Y. Jia, T. Gu, Q. He, and Q. Wu, "An Integrated Approach of Real-time Hand Gesture Recognition Based on Feature Points," *Int. J. Multimed. Ubiquitous Eng.*, vol. 10, no. 4, pp. 413–428, 2015.

[11] T. Ikai, M. Ohka, and H. Yussof, "Behavior Control of Robot by Human Finger Direction," *Int. Symp. Robot. Intell. Sensors*, vol. 41, pp. 784–791, 2012.

[12] H.-S. Yeo, B.-G. Lee, and H. Lim, "Hand tracking and gesture recognition system for human-computer interaction using low-cost hardware," *Multimed. Tools Appl.*, May 2013.

[13] A. Dhawan and V. Honrao, "Implementation f Hand Detection based Techniques for Human Computer Interaction," *Int. J. Comput. Appl.*, vol. 72, no. 17, pp. 6–13, 2013.

[14] O. Ozturk, A. Aksac, T. Ozyer, and R. Alhajj, "Boosting real-time recognition of hand posture and gesture for virtual mouse operations with segmentation," *Appl. Intell.*, vol. 43, no. 4, pp. 786–801, 2015.

[15] A. Aksac, O. Ozturk, and T. Ozyer, "Real-time Multi-Objective Hand Posture/Gesture Recognition by Using Distance Classifiers and Finite State Machine for Virtual Mouse Operations," *Int. Conf. Electr. Electron. Eng.*, pp. 457–461, 2011.

[16] Neurotechnology, "Neurotechnology Company Website." [Online]. Available: http://www.neurotechnology.com/npointer.html.

[17] N. Ibraheem, R. Khan, and M. Hasan, "Comparative Study of Skin Color based Segmentation Techniques," *Int. J. Appl. Inf. Syst.*, vol. 5, no. 10, pp. 24–38, 2013.

[18] S. Meena, "A Study on Hand Gesture Recognition Technique," 2011.

[19] L. Bin and M. Samiei, "Comparison for Image Edge Detection Algorithms," *IOSR J. Comput. Eng.*, vol. 2, no. 6, pp. 1–4, 2012.

[20] D. D. Han, T. C. Zhang, and J. Zhang, "Research and Implementation of an Improved Canny Edge Detection Algorithm," *Key Eng. Mater.*, vol. 572, pp. 566–569, 2014.

[21] H. J. Vala and A. Baxi, "A Review on Otsu Image Segmentation Algorithm," *Int. J. Adv. Res. Comput. Eng. Technol.*, vol. 2, no. 2, pp. 387–389, 2013.

[22] N. Otsu, "A Threshold Selection Method From Gray-Level Histograms," *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.

Farah Farhana Mod Ma'asum, has completed her MSc in Electrical Engineering in March 2017 in Universiti Teknologi MARA Shah Alam, Selangor and Degree in Electronic Engineering at Universiti Teknologi Malaysia in 2011. Currently, she has joined Intel Penang, Malaysia as System Validation Engineer working in FPGA field.

Suhana Sulaiman received her B.Sc in Electrical Engineering from University of Utah, Salt Lake City, USA in 1996 and M.Sc in Microelectronics from University of Newcastle Upon Tyne, UK in 2002. Her Ph.D. degree in Electrical Engineering from Universiti Teknologi MARA, Shah Alam Selangor, Malaysia in 2013. Currently, she is a senior lecturer in Faculty of Electrical Engineering, UiTM Shah Alam, Selangor. Her research interests include RFIC, microelectronics, electronics circuit and system and microfabrication.

Azilah Saparon, is an Associate Professor at Faculty of Electrical Engineering, UiTM Shah Alam, Selangor, Malaysia. She received her B.E.E from Gannon University, Erie, Pennsylvania, U.S.A in 1989, M.Sc in Electrical Engineering from Washington State University at Pullman,U.S.A in 1995 and a Ph.D in Electronic and Electrical Engineering from Loughborough University, Leicestershire, U.K 2006. Her research interests include reconfigurable system, micro-computer architecture and video coding.