HAND GESTURES RECOGNITION WITH IMPROVED SKIN COLOR SEGMENTATION IN HUMAN-COMPUTER INTERACTION APPLICATIONS

ROMI FADILLAH RAHMAT[1*], TENGKU CHAIRUNNISA[1], DANI GUNAWAN[1], MUHAMMAD FERMI PASHA[2], RAHMAT BUDIARTO[3]

[1]Department of Information Technology, Universitas Sumatera Utara, Medan, Indonesia

[2]School of Information Technology, Monash University Malaysia Campus, Malaysia

[2]Department of Information System, Albaha University, Saudi Arabia

*E-mail: romi.fadillah@usu.ac.id

ABSTRACT

Hand gesture has significant roles in human's interaction and the hand gesture recognition itself nowadays becomes an active research area in human-computer interaction. Previous researches on hand gesture recognition used various techniques and tools such as Kinect and data glove. Hand gesture recognition area has many challenges, such as variation of illumination conditions, rotation problem, background problem, scale problem, and classification or translation problem. This research uses computer vision techniques to recognize hand gesture in human-computer interaction to control various apps, such as slideshow presentation, music player, video player, and PDF reader app for people with bare hand and in complex background of the image via web camera. Thus, a method is required to cope with background and skin detection problem. The proposed method combines two color spaces into HS-CbCr format for skin detection and uses averaging background for solving the background problem. The experimental results show that the proposed method is able to recognize hand gesture and reach up to 96.87% of correct results in good lighting condition. The accuracy of hand gesture recognition is influenced by lighting condition. The lower changing illumination on video occurs, the higher accuracy of hand gesture recognition is generated

Keywords: *Hand Gesture Recognition, Human Computer Interaction, Skin Detection, Average Background, Convexity Defects*

1. INTRODUCTION

Hand gesture has significant roles in human's interaction and its recognition nowadays becomes an active research area in human-computer interaction. One aim of the hand gesture recognition system is to avoid various disease, such as Parkinson's disease, trigger finger, gorilla arm, etc., caused by excessive use of mouse or keyboard [1].

From several survey studies on hand gesture recognition researches [2-6], the implementation of hand gesture system required various tools, such as Kinect [7], data glove [8] or high quality camera. Furthermore, there are some limitations of lighting or background condition, which mean that users have to be in particular place while running the system.

Hand gesture recognition had also been implemented for various application domains, such as sign language [9], robotic [10][11], virtual / augmented reality, games, driving assistance [12][13], television control, etc. There are two types of hand gesture recognition [14]:

1. Static Hand Gesture Recognition that recognizes predefined and fixed hand gesture which is detected in one frame at a time [9][16].
2. Dynamic Hand Gesture Recognition that recognizes motion-based gesture which is detected in sequence of images within a certain period [17][18][29].

Variation of illumination conditions which affects extracted hand skin region, rotation problem, background problem, scale problem and classification or translation of hand gesture problem are several problem in hand gesture recognition. Similar to some previous works, this research uses computer vision techniques as well to recognize static hand gesture and propose a method to solve problems in hand gesture recognition. The work focuses on developing hand gesture recognition approach which is easily used for people with bare hand in using daily applications, such as music and video player, PDF reader and slideshow presentation, which only with user's bare hand and is in complex background condition via web camera.

There are so many researchers conducted on hand gesture recognition with various tools and techniques, which mainly aim to build a system which will enable their device control to understand specific hand gesture or to improve the quality of human-computer interaction. Recently, it is also implemented for the other purposes such as sign language, robotic, virtual reality, games, etc.

A simple method for hand gesture recognition is implemented in [14], Blob Detection COG was used to extract feature and Template/Pattern Matching to recognize dynamic hand. The authors used a lower cost tool, namely the "green" footprint and implemented the method to type in notepad using hand gesture with extra facility in various languages, such as Japanese, Gujarati, Tamil, Canada, etc. The authors also used Blob Detection for feature extraction of hand skin region. The research focused on using hand gesture recognition for various functionalities, such as mouse control, single-pointer calibration, multi-pointer calibration, winamp control and gaming control. From the two works above, it can be implied that there were still limitations of background and lighting conditions.

Another research was conducted for recognizing static hand gesture using neural network [15]. The proposed method in the research was Wavelet Network and Artificial Neural Network (ANN) for feature extraction. In the testing step, 60 images of hand gesture with various Gaussian noise are used and divided into 6 classes. The experimental concluded that the overall accuracy boost up into 97%. Research work in [16] recognized static hand gesture for alphabetic sign language based on American Sign Language using Edge Oriented Histogram method for feature extraction and Multiclass Support Vector Machine (SVM) algorithm for classifying hand gesture. The authors concluded that Multiclass SVM could provide high accuracy in classifying the hand gestures up to 93.75% [16].

To train every input images for classifying using a machine learning requires so much time and computing resources. The more testing data used, the higher overall accuracy could be generated [19][20]. Finger counting method is used to extract features from computer vision techniques, specifically contour, convex-hull and convexity defects for static hand gesture recognition [21]. The approach is similar to our proposed work. From both research works, it can be concluded that computer vision techniques is very fast and easy to implement for hand gesture recognition, but still neither flawless nor properly applicable in different situation.

The authors in [22] also used Blob Detection for recognizing static hand gesture. The authors concluded that the efficiency of the system can be improved by background subtraction which removes the background so the system can detect the object (hand) easily. Another related research worked on detecting a moving object using average background method. This research concluded that the method could detect complex background with low-level computation [23].

Having done reviewing the related works, this paper proposes an improved computer vision technique using average background method to reduce the flaw and applicable in different situation. Based on the previous research, we define the novelty and contribution of the work is on improved skin color segmentation which combined with several image processing techniques including determined hand direction to perform hand gesture recognition for human computer application.

2. METHODOLOGY

The proposed method to recognize hand gesture via web camera for human-computer interaction in this research consists of several steps, namely image acquisition via web camera, resizing image, RGB to HSV conversion, RGB to YCbCr conversion, skin detection using Skin Color Bounding Rule method, skin segmentation, grayscaling, background accumulation, frame differencing, thresholding, inversion, binary image enhancement using erosion, dilation and Gaussian Blur, feature extraction using contour, convex-hull, and convexity defects, count detected finger and determine the hand direction. After all the above steps are done, then we can get the form of instruction or command to control the chosen application (slideshow presentation, video player, music player, or PDF reader application). Here in Figure 1 is our proposed method.

2.1 Skin Detection

Skin detection phase is used to accelerate the process of hand detection in every frame in video which is captured via webcam. Furthermore, the detection of hand skin region can simplify the system, so that the user does not require to wear any special device, such as Kinect or data glove which

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

hinder the nature of communication between human and computer. The method described here has been studied and experimented before [28]. The following steps are done in skin detection phase.

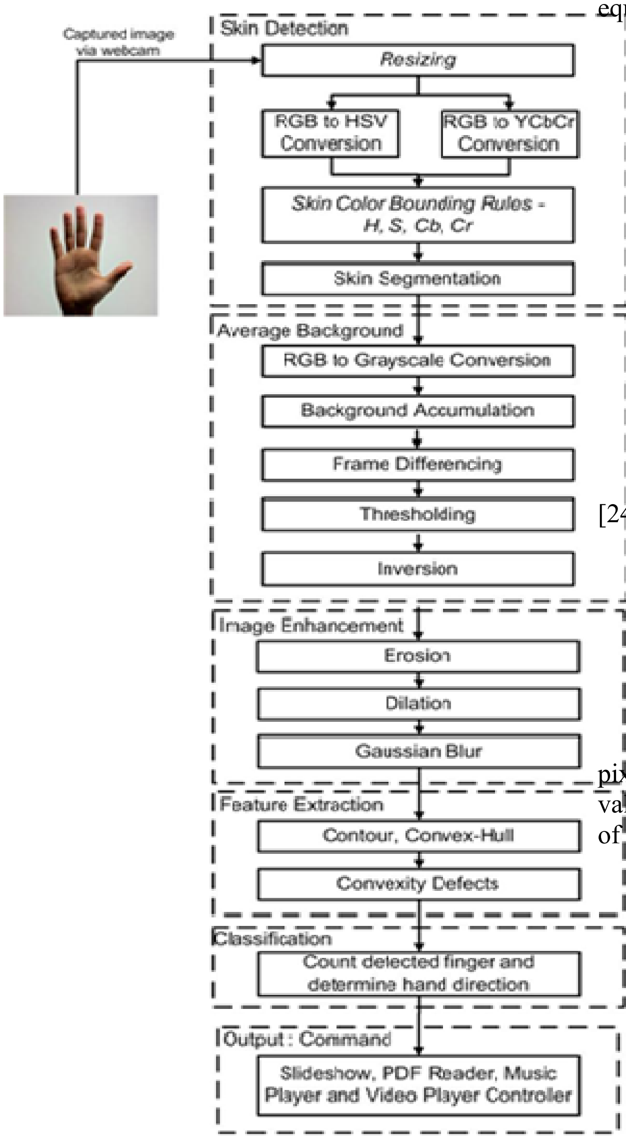


Figure 1: General Architecture

Step 1: Resize the captured image to become 400 x 300 (image(400,300)).

Step 2: Do RGB Normalization equation 1 until equation 3 as follows.

$$R' = \frac{R}{R+G+B} \quad (1)$$

$$G' = \frac{G}{R+G+B} \quad (2)$$

$$B' = \frac{B}{R+G+B} \quad (3)$$

Step 3: Do RGB to HSV conversion using all equations as follows.

$$V = \max(R, G, B) \quad (4)$$

$$S = \begin{cases} \frac{V - \min(R,G,B)}{V}, & if\ V \neq 0 \\ 0 & \end{cases} \quad (5)$$

$$H = \begin{cases} \frac{60(G-B)}{V - \min(R,G,B)}, & if\ V = R \\ 2 + \frac{60(B-R)}{V - \min(R,G,B)}, & if\ V = G \\ 4 + \frac{60(R-G)}{V - \min(R,G,B)}, & if\ V = B \end{cases} \quad (6)$$

$$if\ H < 0, then\ H = H + 360 \quad (7)$$

Step 4: Convert RGB value to YcbCr as follows [24].

$$Y = 0.299.R + 0.587.G + 0.114.B \quad (8)$$

$$Cr = (R - Y).0.713 + delta \quad (9)$$

$$Cb = (B - Y).0.564 + delta \quad (10)$$

Step 5: Check the value of HSV value in each pixel from HSV image and Cb (chrominance-blue) value and Cr (chrominance-red) value in every pixel of YCrCb image using the following algorithm [28].

FOR every pixels of image(x,y)
IF NOT ((Hue >= 0) AND (Hue <= 50)) AND
((Sat >= 0.23) AND (Sat <= 0.68)) AND
((Cb > 77) AND (Cb < 127)) AND
((Cr > 133) AND (Cr < 173))
THEN
SET RGB(x,y) = 0,0,0 (black)
END IF
END FOR

Step 6: Show the image (x,y).

2.2 Average Background

After the whole steps in skin detection phase are completed, then the next phase is removing the detected background which has the same color as human's skin using average background method. These following steps are done in average background method.

Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 1: Convert the skin region image (RGB) to grayscale image using equation 11 [24].

$$Y = 0.299.R + 0.587.G + 0.114.B \tag{11}$$

Step 2: Find the difference between the first frame and the next frame (frame differencing).

Step 3: Accumulate frames to determine a frame which is defined as a background image.

Step 4: Threshold the grayscale image into binary image. This step is also to separate the object from the noise pixels of frame differencing image.

Step 5: Invert the binary image to get the moving object region.

2.3 Image Enhancement

The binary image generated from average background method still contains much noise. To reduce it in captured image, we use 3 steps of image enhancement, namely erosion, dilation, and Gaussian Blur.

Step 1: Erosion. It aims to decrease or to reduce the unnecessary object.

Step 2: Dilation. After the unnecessary objects are removed by erosion, then the detected object should be bold by dilation.

Step 3: Gaussian Blur. It aims to reduce noise in dilated image so that the pixels will not be defined as an object.

2.4 Feature Extraction

After reducing noise in image enhancement phase, the next phase is feature extraction by detecting contour, convex-hull and convexity defects in detected object. This phase is aim to get information from the object for classifying or analyzing image.

2.4.1 Contour

Contour can be defined as the curve generated because of changing intensity of neighbour pixels. Due to the changing intensity, edge can be detected in the image. In this research, we detect contours in binary image generated from image enhancement phase before. Figure 2(a) shows the detected contour of a hand image [25].

2.4.2 Convex-hull

Convex-hull is an outer line which covers all the contour points. In this research, before detecting convex-hull, we should find the biggest contour area. It is aim to accelerate the detection of convexity defects in the next step. Figure 2(b) shows the detected convex-hull of a hand image [25].

2.4.3 Convexity Defects

Convexity defects is obtained from all contour points and convex-hull. It is represented in four vector elements, namely start_index, end_index, farthest_index and fix_depth. The detected convexity defects are shown in Figure 2(c) [25].

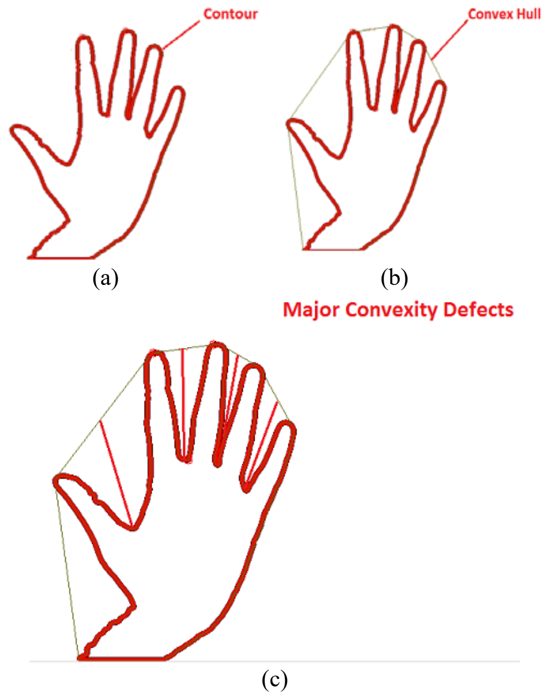


Figure 2: Feature Extraction Process; (a) Contour; (b) Convex-Hull; (c) Convexity Defects

2.5 Classification

From the extracted feature, we classify the hand gesture with the number of detected finger and hand direction. Instruction. The output of the classification is an instruction or command to control an application for simplifying the human-computer interaction. These following steps are performed for classifying the hand gestures.

Step 1: Find the center of gravity (CoG) point which is obtained from moments of every points in the biggest contour area. The process of determining the CoG point is described as follows:

Initialize array = list of points from feature extraction
Initialize variable moment, moment order (0,0), moment order (0,1), moment order (1,0)
moment = moments(array)
moment order (0,0) = momen.get_m00();
moment order (0,1) = momen.get_m01();
moment order (1,0) = momen.get_m10();
IF (momen order (0,0) != 0)
THEN

Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SET x coord of CoG point = momen order (1,0) / momen order (0,0)
y coord of CoG point = momen order (0,1) / momen order (0,0)
ENDIF

Step 2: Detect the fingertips in detected contour area from the vector elements of convexity defects. In this research, the minimum size of human's finger is set as 15. If the set value is less than the minimum size of fix_depth, then the object is not defined as a fingertip.

Step 3: Find the angle degree between center of gravity point and detected fingertip point using trigonometric function in equation 12 and equation 13.

$$\theta = \arctan \frac{CoG.y - fingertip.y}{CoG.x - fingertip.x} \quad (12)$$

$$angle = \begin{cases} \theta \\ \theta + 360, \ if\ \theta < 0 \end{cases} \quad (13)$$

Step 4: Determine the hand direction based on the obtained degree between center of gravity point and detected fingertip point. Figure 3 and Table 1 show the way to determine the hand direction using certain rules of degree. In this research, before determining the hand direction, we should find the average degree of all the detected fingertips.

Figure 3 shows that the red point represents center of gravity point and the yellow point represents detected fingertips. The degree can be obtained from those two points and used as reference of classifying the hand direction. The degree is measured based on counter-clockwise of the x-axis. Table 1 shows the rules of direction and average degree in this research.

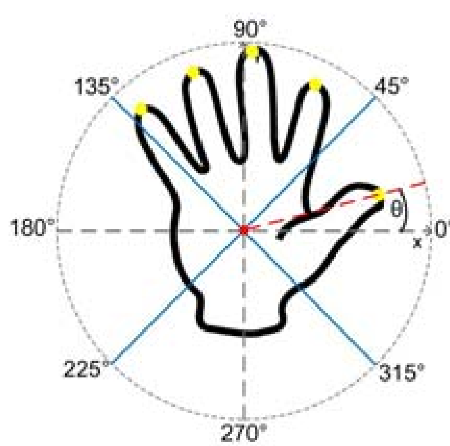


Figure 3: Determining Hand Direction

Table 1: Rules of Direction and Average Degree.

Direction	Rules of Average Degree (θ)
Up	46° - 135°
Right	136° - 225°
Down	226° - 315°
Left	0°- 45° & 315° - 360°

Step 5: Classify the number of finger and hand direction to run an instruction of chosen application (slideshow presentation, PDF reader, music player or video player application). The instruction is executed when the detected number of finger and hand direction is fit in 3 (three) sequence frames.

3 EXPERIMENTAL SET UP, RESULT AND DISCUSSION

The obtained results from the proposed method and practical applications will be discussed in this section.

3.1 Experimental Set-Up

The hardware and software specification used to build and run the system are as follows:

1. Processor : Intel ® CoreTM i5-2410M CPU 2.30GHz
2. Capacity : hard-disk 500GB
3. RAM : 3 GB DDR3
4. OS : Microsoft Windows 7 Ultimate
5. Presentation: Microsoft Office PowerPoint.
6. PDF Reader : Foxit Reader.
7. Music player : Winamp version 5.
8. Ext. video player : VLC Media Player
9. Web camera : 8 MP with 1280 x 720px resolution
10. Java IDE : Eclipse IDE Indigo version 1.4.1
11. Library : OpenCV 2.4.11.0

3.2 Result and Discussion

In result section, the experimental results of the two main phases, skin detection and hand gesture recognition, will be discussed and analyzed separately.

3.2.1 Skin detection

The proposed method for skin detection, HS-CbCr, has been experimented by comparing it with another 3 methods, namely HSV, YCbCr and H-CbCr using Hand Gesture Recognition (HGR) Dataset [26][27]. This dataset consists of three types of image data for hand gesture recognition

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

system in different background and lighting condition.

- HGR I :
  - Total : 61 image
  - Background : Random
  - Lighting : Random
  - Image Size : 640x480 pixel

- HGR II
  - Total : 85 image
  - Total Entity : 3 people
  - Total Gestures : 13 gestures
  - Background :44 images with gray background 41 images with random background
  - Lighting : Flashed
  - Image Size : 512x340 pixel

- HGR III
  - Total : 574 image
  - Total Entity : 18 people
  - Total Gestures : 32 gestures
  - Background : Green tone
  - Lighting : Flashed
  - Image Size : 339x512 pixel

It is shown in Table 2, HS-CbCr gives a better average percentage of accuracy than the other three methods(up to 84.43%).The HS-CbCr is able to detect skin color in different lighting condition, however does not give a high accuracy enough in condition of high light intensity.

Table 2: Skin Detection Experimental Result.

No.	Methods	Percentage Result(%)			Average Percentage
		HGR I	*HGR II*	*HGR III*	
1	HSV	90.04	67.71	92.6	83.45
2	YCbCr	84.18	69.78	93.65	82.54
3	H - CbCr	87.61	65.89	95.26	82.92
4	HS - CbCr	90.75	69.91	92.63	84.43

3.2.2 Experimental result of feature extraction

From all the processes described for the feature extraction in the previous section, the experimental results can be seen in Figure 4.

(a)

(b)

(c)

(d)

Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

(e)

Figure 4: Results; (a) Skin Segmentation Process; (b) Averaging Background Process; (c) Image Enhancement Process; (d) Detected Contour Area; (e) Feature Extraction Process

Figure 4(a) and Figure 4(b) are the processes of removing the detected background during the segmentation of the part of detected skin for a better feature extraction quality through the process of skin segmentation and averaging background while also relying on frame differencing process in advance to allow object movement detection. Other moving objects might be detected and contoured during the process and will decrease the accuracy of the hand gesture recognition. Therefore, the system cannot handle moving objects which have similar color to skin in the background.

After going through averaging the background, the image will go through image enhancement process as displayed in Figure 4(c). This process will remove unnecessary pixels which still stay after the averaging background process, so that it can enhance the feature extraction process in recognizing hand gesture better. Having done going through these processes, the detected contour area will be displayed and eventually the feature extraction process will show the correct hand gesture recognition and direction as seen in Figure 4(d) and Figure 4(e).

The convexity defects detection time which is the last phase of this process are calculated using this formula:

$$T = \frac{T2 - T1}{1000000} \tag{14}$$

where: T = detection time (ms)
T1 = time before the detection
T2 = time after the detection

The experiment results of convexity defects detection time using web camera with 1280 x 720 resolution to measure the detection time are shown in Table 3.

Table 3: Experimental Result of Convexity Defects Detection Time.

No.	Frame Name	Detection Time (ms)
1	Frame001	118.705
2	Frame002	60.266
3	Frame003	52.525
4	Frame004	51.744
5	Frame005	52.684
6	Frame006	52.285
7	Frame007	51.241
8	Frame008	52.567
9	Frame009	51.659
10	Frame010	51.898
11	Frame011	51.328
12	Frame012	51.216
13	Frame013	50.865
14	Frame014	51.836
15	Frame015	51
16	Frame016	52.258
17	Frame017	50.848
18	Frame018	49.876
19	Frame019	50.579
20	Frame020	51.512
21	Frame021	50.452
22	Frame022	50.119
23	Frame023	50.676
24	Frame024	50.504
25	Frame025	50.626

From the experiments with initial 25 frames, an average detection time is: 54,371 ms with fastest detection time is: 49.876 ms and slowest detection time is: 118,705 ms.

3.2.3 Hand gestures recognition

In order to experimenting the proposed method for hand gesture recognition, 9 hand gestures, which is defined as an instruction or command in two different rooms with different lighting conditions are used.

- Room A, downlight room with low-light intensity (dim).
- Room B, downlight room with high-light intensity (bright).

Figure 5(a),(b),(c),(d),(e),(f),(g),(h),(i) are the pictures of the experiments taken in room A with low-light intensity.

Table 4 below shows the hand gesture recognition testing result in Room B.

From the experimental results in Room B for 9 hand gestures shown in Table 4, we get the average percentage of accuracy as follows:

Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

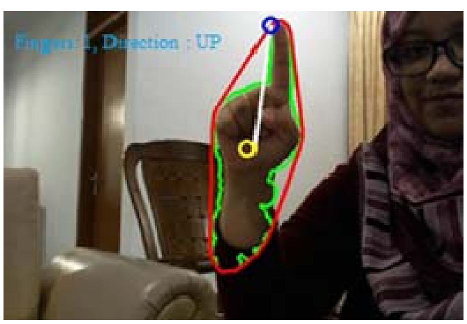
$$\bar{B} = \frac{96.51 + 100 + 99.21 + 99.03 + 98.2 + 100 + 98.68 + 98.06 + 98.72}{9}$$
$$= 98.71\%$$

Table 4: Hand Gestures Recognition Testing Result in Room B.

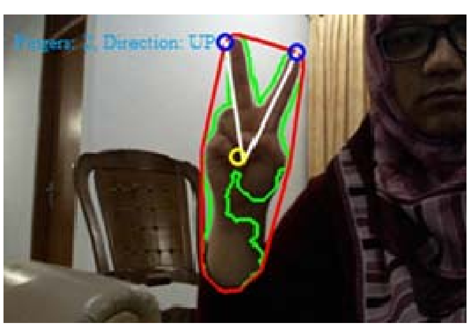
No. of Fingers	Hand Direction	No. of Testing (n)	Accuracy (%)	Error(%)
1	Up	86	96.51	3.49
2	Up	88	100	0
3	Up	126	99.21	0.79
4	Up	103	99.03	0.97
5	Up	111	98.2	1.8
1	Right	62	100	0
2	Right	76	98.68	1.32
1	Left	103	98.06	1.94
2	Left	78	98.72	1.28

From all of the experimental results above, we may conclude that lighting condition badly affect the overall accuracy, because it affects on skin detection result. When in dim lighting condition, the error that often happens is that system is unable to detect the skin color because the color of pixel is too dark. Whereas in bright lighting condition, all the hand skin region is well-defined. The overall accuracy of our proposed method for hand gesture recognition can be calculated as follows.

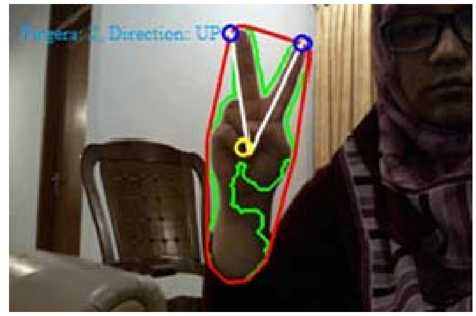
$$\text{Avg. of Overall Accuracy} = \frac{\bar{A} + \bar{B}}{2}$$
$$= \frac{95.03 + 98.71}{2}$$
$$= 96.87\%$$



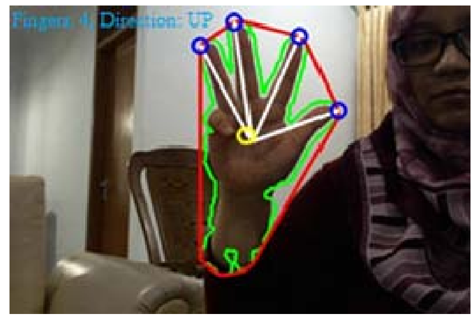
(a)



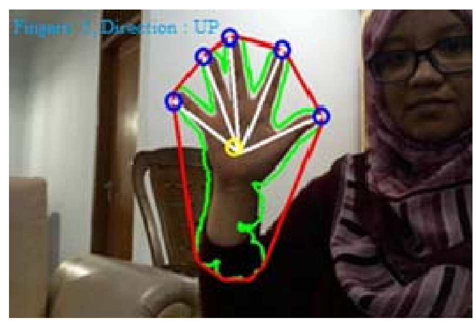
(b)



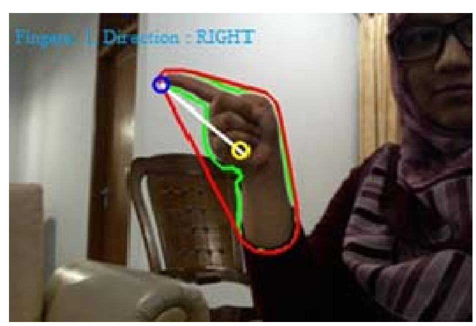
(c)



(d)

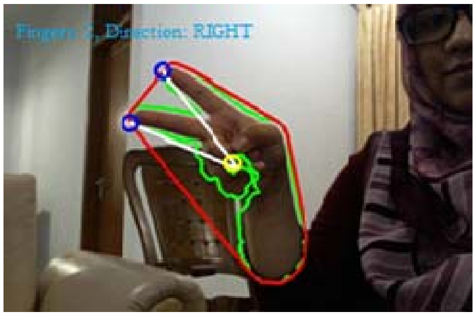


(e)

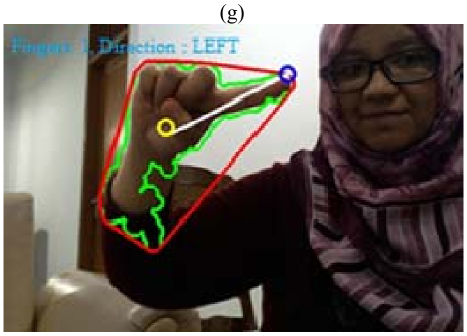


(f)

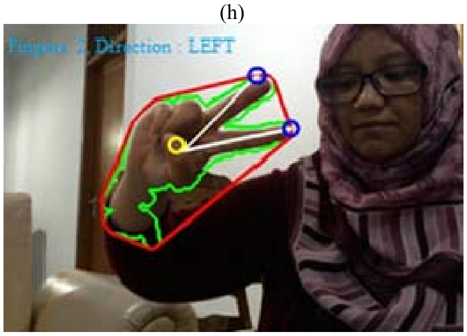
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



(g)



(h)



(i)

Figure 5(a),(b),(c),(d),(e),(f),(g),(h),(i): Hand Gesture Recognition With High-Light Intensity (Room B)

Table 5: Hand Gestures for Controlling Slideshow Presentation Application.

No.	Functions	Hand Gesture	
		No. of fingers	*Hand direction*
1	Slideshow	4	Up
2	Next Slide	2	Right
3	Previous Slide	2	Left
4	End Show	5	Up

Table 6: Hand Gestures for Controlling PDF Reader Application.

No.	Functions	Hand Gesture	
		No. of fingers	*Hand direction*
1	First Page	1	Left
2	Last Page	1	Right
3	Next Page	2	Right
4	Previous Page	2	Left
5	Zoom In	5	Up
6	Zoom Out	4	Up
7	Scroll Up	1	Up
8	Scroll Down	5	Up

Table 7: Hand Gestures for Controlling Music Player Application.

No.	Functions	Hand Gesture	
		No. of fingers	*Hand direction*
1	Play	1	Up
2	Pause / Resume	2	Up
3	Stop with fadeout	5	Up
4	Next Song	2	Right
5	Previous Song	2	Left

Table 8: Hand Gestures for Controlling Video Player Application.

No.	Function	Hand Gesture	
		No. of fingers	*Hand direction*
1	Play from the beginning	1	Up
2	Stop Movie	5	Up
3	Full Screen	4	Up
4	Pause/Resume	2	Up
5	Mute/Un-mute	3	Up
6	Volume Up	2	Right
7	Volume Down	2	Left

3.3 Practical Application

- Slideshow presentation (Microsoft Office PowerPoint). Table 5 shows all functions in the slideshow presentation application that can be controlled by hand gesture.
- PDF reader (Foxit Reader). Table 6 shows all functions in the PDF reader application that can be controlled by hand gesture.
- Music player (Winamp). Table 7 shows all functions in the music player application that can be controlled by hand gesture.
- Video player (VLC). Table 8 shows all functions in the video player application that can be controlled by hand gesture. One line space should be given above the sub section while no space should be given below the heading and text.

Based on the convexity defects experiments described in the previous sub-section, the eventual experimental results of hand gesture recognition to control slideshow presentation, PDF reader, music player and video player with hand gesture are displayed in Figure 6.

Figure 6(a) is the hand gesture to control the scroll up function in PDF reader and play function in music and video player. This gesture is the most

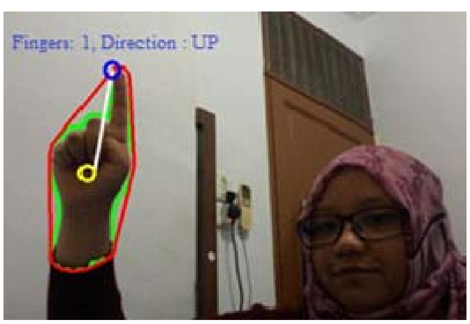
Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

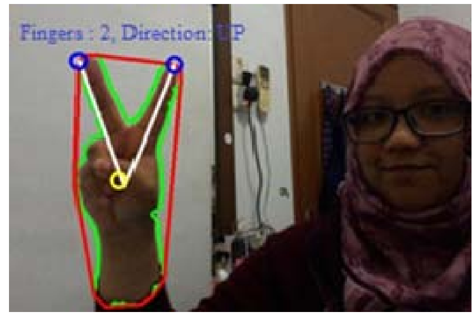
common gesture done in previous researches and the most common error in the recognition phase is when the palm is not detected because it is blocked by the folded fingers.

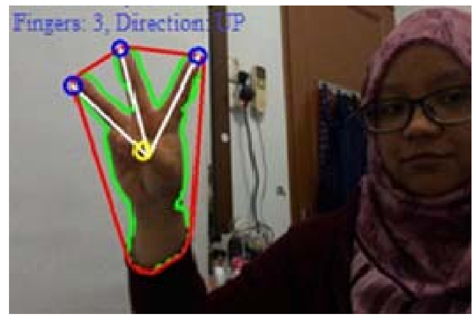
Figure 6(b) is the hand gesture to control the scroll down function in PDF reader and pause/resume function in music and video player. After several experiments, this gesture is one of the easiest recognized gesture because of the palm is seldom blocked by the folded fingers. Figure 6(c) is the hand gesture to control the mute function in video player, Figure 6(d) is the hand gesture to start the slideshow presentation in full screen mode, to zoom out in PDF reader and to do full screen mode in video player. Figure 6(e) is the hand gesture to end the slideshow and go back to the editor menu of the slide, to zoom in PDF reader and to stop in music or video player. Most of these gesture's errors are usually when the folded fingers are also detected as fingers due to the small depth point of the user's finger.

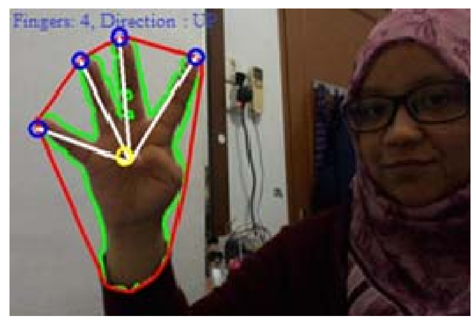
Figure 6(f) is the hand gesture to navigate to the last page in PDF reader, and Figure 6(g) is the hand gesture to move to the next slide page in slideshow presentation, to move to the next page in PDF reader, to select the next music in the playlist in music player and to increase the volume in video player. Figure 6(h) is the hand gesture to navigate to the first page of PDF reader, and Figure 6(i) is the hand gesture to move to the previous slide in slideshow presentation, to move to the previous page in PDF reader, to select the previous music in the playlist in music player and to lower the volume in video player.

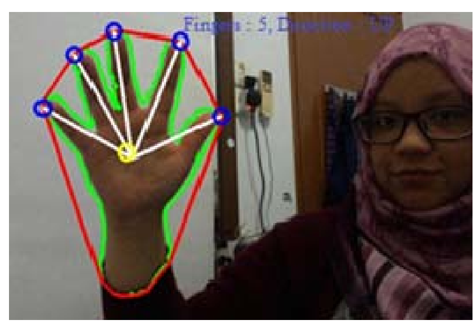


(a)


(b)

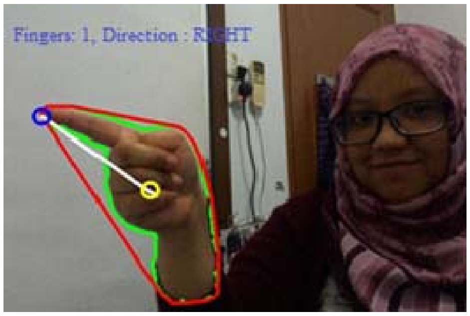

(c)


(d)

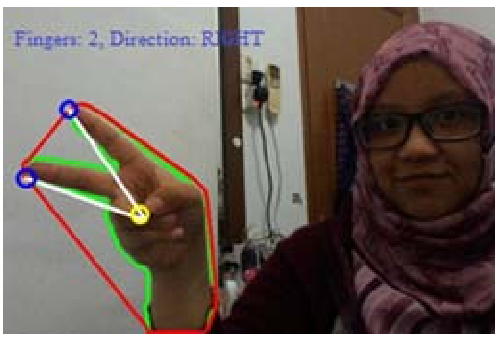

(e)

Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3

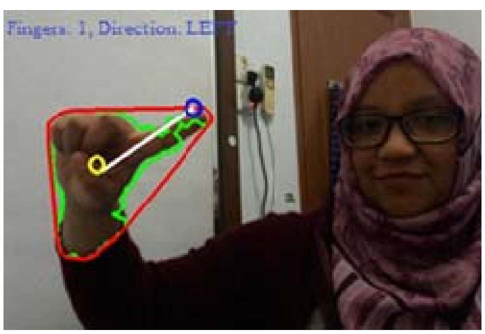
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195



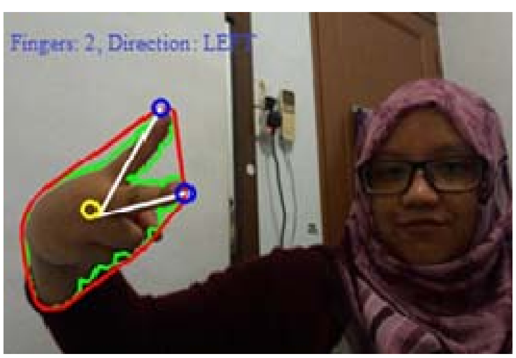
(f)



(g)



(h)



(i)

Figure 6(a),(b),(c),(d),(e),(f),(g),(h),(i): Detected Hand Gesture Recognition for Controlling Applications

3. CONCLUSION AND FURTHER WORK

Considering the experimental results, several conclusions can be drawn from this research. First, the proposed method can recognize hand gesture for human-computer interaction via web camera. The overall accuracy of the proposed method reached up to 96.87%. The accuracy is affected by lighting condition which affects on skin detection. The lower changing illumination on video occurs, the higher accuracy of hand gesture recognition is generated. From all phases of the proposed method for hand gesture recognition system, skin detection is the most important phase. Skin detection using HSCbCr provides up to 84.43% of average percentage of accuracy in different lighting conditions. The proposed method in this research is proved to recognize hand gesture easily, however the accuracy depends on how much accurate the object detection phase is implemented.

Future work for hand gesture recognition is to adding hand tracking method for dynamic hand gesture recognition. A machine learning may also be considered to give better accuracy for object recognition. Another recommendation for future skin detection stage is to use a method which can handle well the lighting condition problem. The obtained results from the proposed method and practical applications will be discussed in this section.

ACKNOWLEDGEMENT

The authors would like to thanks Lembaga Penelitian Universitas Sumatera Utara in supporting this research study through TALENTA Scheme and Grant.

REFRENCES:

[1] R. Bhatt, N. Fernandes, and A. Dhage, "Vision based hand gesture recognition for human computer interaction," *International Journal of Engineering Science and Innovative Technology (IJESIT)*, Vol. 2, No. 3, 2013, pp. 110–115.

[2] R. Z. Khan and N.A. Ibraheem, "Comparative study of hand gesture recognition system," in Proc. *International Conference of Advanced Computer Science & Information Technology in Computer Science & Information Technology (CS & IT)*, 2012, pp. 203–213.

[3] S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer

interaction: a survey," *Artificial Intelligence Review*, Vol. 43, No.1, 2012, pp. 1–54.

[4] A. R. Sarkar, G. Sanyal, and S. Majumder, "Hand gesture recognition systems: a survey," *International Journal of Computer Applications*, Vol. 671, No.15, 2013, pp. 26-37.

[5] J. Francis and B. K. Anoop, "Significance of hand gesture recognition systems in vehicular automation - a survey." *Significance*, Vol. 99, No.7, 2014, pp. 50-55.

[6] S. Mutha and K. S. Kinage, "Study on hand gesture recognition," *International Journal of Computer Science and Mobile Computing*, Vol. 4, No.1, 2015, pp. 51-57.

[7] F. Jiang, S. Zhang, S. Wu, Y. Gao, and D. Zhao, "Multi-layered gesture recognition with kinect," *The Journal of Machine Learning Research*, Vol. 16, No.1, 2015, pp. 227-254.

[8] F. Camastra and D. De Felice, "LVQ-Based Hand Gesture Recognition Using A Data Glove," *Neural Nets and Surroundings*, Springer Berlin Heidelberg (Germany), 2013, pp. 159-168.

[9] S. Nagarajan and T. S. Subashini, "Static hand gesture recognition for sign language alphabets using Edge Oriented Histogram and Multi Class SVM," *International Journal of Computer Applications*, Vol. 82, No.4, 2013, pp. 28-35.

[10] P. V. Kumar, N. R. V. Praneeth, and V. Sudheer, "Hand and finger gesture recognition system for robotic application," *International Journal of Computer Communication and Information System (IJCCIS),* Vol. 2, No.1, 2010, pp. 266-269.

[11] E. Aghajari, and D. Gharpure, "Real-time vision-based hand gesture recognition for robotic application," *International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE)*, Vol. 4, No. 3, 2014, pp. 133-142.

[12] P. Molchanov, S. Gupta, K. Kim, and K. Pulli, "Multi-sensor system for driver's hand-gesture recognition," *11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG),* 2015, pp. 1-8.

[13] E. Ohn-Bar and M. M. Trivedi, "Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 15, No.6, 2014, pp. 2368-2377.

[14] M. R. Ramjan, R. M. Sandip, P. S. Uttam, and W. S. Srimant, "Dynamic hand gesture recognition and detection for real time using human computer interaction," *International Journal of Advance Research in Computer Science and Management Studies (IJARCSMS)*, Vol. 2, No. 3, 2014, pp. 425-430.

[15] H. A. Jalab, "Static hand gesture recognition for human computer interaction," *Information Technology Journal*, Vol. 11, No. 9, 2012, pp. 1265-1271.

[16] S. Nagarajan and T. S. Subashini, "Static hand gesture recognition for sign language alphabets using Edge Oriented Histogram and Multi Class SVM," *International Journal of Computer Applications*, Vol. 82, No. 4, 2013, pp. 28-35.

[17] J. R. Pansare, M. Bansal, S. Saxena, and D. Desale, "Gestuelle: A system to recognize dynamic hand gestures using hidden markov model to control windows applications," *International Journal of Computer Applications*, Vol. 62, No. 17, 2013.

[18] V. Nayakwadi and N. B. Pokale, "Natural hand gestures recognition system for intelligent HCI: a survey," *International Journal of Computer Applications Technology and Research*, Vol. 3, No. 1, 2014, pp. 10-19.

[19] A. Williamson, "Vision based cursor control using hand gesture," University of The West Indies (United States), 2014.

[20] N. C. Kılıboz and U. Güdükbay, "A hand gesture recognition technique for human–computer interaction," *Journal of Visual Communication and Image Representation*, Vol. 28, 2015, pp. 97-104.

[21] S. U. Rahman, Z. Afroze, and M. Tareq, "Hand gesture recognition techniques for human computer interaction using OpenCV," *International Journal of Scientific and Research Publications*, Vol. 4, No. 12, 2014.

[22] S. Basak and A. Chowdhury, "A vision interface framework for intuitive gesture recognition using color based blob detection," *International Journal of Computer Applications*, Vol. 90, No. 15, 2014, pp. 36-40.

[23] Z. Yi and F. Liangzhong, "Moving object detection based on running average background and temporal difference," *International Conference on Intelligent Systems and Knowledge Engineering (ISKE),* 2010, pp. 270-272.

[24] B. C. Ennehar, O. Brahim, and T. Hicham, "An appropriate color space to improve human skin detection," *INFOCOMP Journal of Computer Science*, Vol. 9, No. 4, 2010, pp. 1-10.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[25] A. Dhawan and V. Honrao, "Implementation of hand detection based techniques for human computer interaction," *International Journal of Computer Applications*, Vol. 72, No. 17, 2013, pp. 6-13.

[26] M. Kawulok, J. Kawulok, and J. Nalepa, "Spatial-based skin detection using discriminative skin-presence features," *Pattern Recognition Letters*, Vol. 41, 2014, pp. 3-13.

[27] M. Kawulok, J. Kawulok, J. Nalepa, and M. Knyc. (2014) Database for hand gesture recognition. [Online]. Available: http://sun.aei.polsl.pl/~mkawulok/gestures/, 2014.

[28] R. F. Rahmat, T. Chairunnisa, D. Gunawan, and O. S. Sitompul, "Skin color segmentation using multi-color space threshold," *Proc. ICCOINS*, 2016, pp. 391-396.

[29] R. F. Rahmat, S. Purnamawati, E.P. Geonfi, O. S. Sitompul, M. F. Pasha, R. Budiarto "A Study on Dynamic Hand Gesture Recognition for Finger Disability using Multi-layer Neural Network" *Journal of Theoretical and Applied Information Technology*, Vol. 96, No. 4, pp. 3413-3452.