Hand detection and segmentation using smart path tracking fingers as features and expert system classifier

Khaled N. Yasen[1], Fahad Layth Malallah[2], Lway Faisal Abdulrazak[3],
Aso Mohammad Darwesh[4], Asem Khmag[5], Baraa T. Shareef[6]
[1]Department of Computer Science, Cihan University-Erbil, Erbil 4400, Iraq
[2]Computer and Information Engineering, College of Electronics Engineering, Ninevah University, Mosul, Iraq
[3]University Research Center, Computer Science Department, Cihan University Slemani, Slemani, Iraq
[4]Department of Information Technology, University of Human Development, Sulaimani, Iraq
[5]Computer system Engineering, Faculty of Engineering, University of Zawia, Azzawia, Libya
[6]Department of Information Technology.College of Information Technology, Ahlia University,
Manama, Kingdom of Bahrain

Article Info

Article history:

Received Feb17, 2019
Revised Apr 2, 2019
Accepted Jul 21, 2019

Keywords:

Expert system
Hand detection
Hand gesture recognition (HGR)
Human-computer interaction (HCI)
Segmentation

ABSTRACT

Nowadays, hand gesture recognition (HGR) is getting popular due to several applications such as remote based control using a hand, and security for access control. One of the major problems of HGR is the accuracy lacking hand detection and segmentation. In this paper, a new algorithm of hand detection will be presented, which works by tracking fingers smartly based on the planned path. The tracking operation is accomplished by assuming a point at the top middle of the image containing the object then this point slides few pixels down to be a reference point then branching into two slopes: left and right. On these slopes, fingers will be scanned to extract flip-numbers, which are considered as features to be classified accordingly by utilizing the expert system. Experiments were conducted using 100 images for 10-individual containing hand inside a cluttered background by using Dataset of Leap Motion and Microsoft Kinect hand acquisitions. The recorded accuracy is depended on the complexity of the Flip-Number setting, which is achieved 96%, 84% and 81% in case 6, 7 and 8 Flip_Numbers respectively, in which this result reflects a high level of finite accuracy in comparing with existing techniques.

Corresponding Author:

Fahad Layth Malallah,
Computer and Information Engineering,
College of Electronics Engineering,
Ninevah University, ORCID: 0000-0001-6067-7302, Mosul, Iraq.
Email: Fahad.malallah@uoninevah.edu.iq

1. INTRODUCTION

In order to enhance the quality of life of disabled people, human-computer interaction (HCI) must be developed to achieve the aforementioned [1]. Hand gesture recognition (HGR) is a major topic of HCI that attracts researchers in different fields of computer vision, pattern recognition, and machine learning. Hand and head gestures were the first modes of communication. Usually, the mode of communication is verbal and non-verbal. In terms of non-verbal communication, it can be used for many kinds of applications such as aviation simulation, 3D gaming, and surveying. On the most popular HCI tools is HGR techniques. HGR system is similar to the biometric system, biometric systems are consisting of basically the following stages: input data, preprocessing, feature extraction and selection, and classification stages, as dully explained in [2-4]. The basic stages of designing the HGR system are comprising of the following: data acquisition, detection, segmentation and tracking, feature extraction and selection with the final stage is the gesture recognition by using various classification algorithms [5]. In HGR, there is no need of a peripheral device to interact with the computer except the camera, so as to capture the fronted view to be analyzed by different image processing and artificial intelligent tools by a processor, then actions will be taken accordingly[6].

One of the challenges of HGR is the hand detection and segmentation precisely, especially if there are many random objects beside hand object. Moreover, detection operation will be more difficult in the case where the random object having color as the same as hand color. In this paper, a new algorithm is presented and explained on hand extraction and segmentation by using scanning the object image from left to right and from top to bottom in order to scan how many flips from zero pixel“0” as black to one pixel “1” as white. In this research, the hand is assumed to be detected must take a form or template as shown in Figure 1. Otherwise, the object will not be classified as a hand object. This is the assumption of the current algorithm, in which its accuracy depends on the Flip_Number, which will be set based on the required complexity. In other words, Flip_Number is considered as the degree of the complexity, larger Flip_Number is set, better hand object is predicted. More details of the algorithm will be illustrated in the methodology section and experiments are conducted as well, to test the correctness of the proposed algorithm.

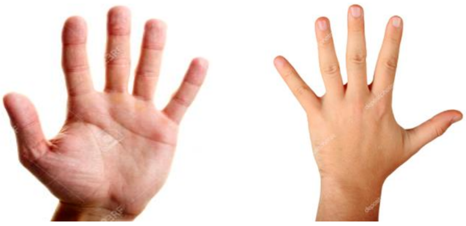


Figure 1. Hand template depended for the detection pointed out as 5-finger

It is worth to mention that, the idea of this paper as flips number based on the smart path has been adapted from [7], in which the origin work in this paper was exploiting smart path to count numbers by hand gesture as 0,1,2,3,4, and 5. Accordingly, a modification has been done to outcome a new version to be much more suitable for our proposed work as the following hypothesis: if there are five fingers, so it means it is a hand otherwise it is not a hand even if there are four fingers are pointed out. The modification will be explained in the methodology subsection. The proposed method does not need data training, which is an advantage that makes the system reliable for embedded systems and lightweight devices. However, the weak point of this algorithm that the hand detection works for only five fingers are pointed out, for instance, a person has a cut finger, it might not be working properly, or it requires to inform the system administrator, later on, to change the parameter of the detection from five to four by simply changing the filp number. The organization of this paper is as follows; Section 2 reviews literature related to hand detection, Section 3 explains the methodology of the proposed technique with testing and analysis, Section 4 describes the experiment; Section 5 presents the results and discussions. Finally, the conclusion of this research and its possible future work are presented in Section 6.

2. LITERATURE REVIEW

Hand detection ideas for previous works are listed in this section with their methodologies and attributes. For instance, in [8], hand detection is designed according to hand motion based on FIFO to detect foreground hand and non-hand information. This idea is based on the several consecutive difference images through the FIFO and path overlap, and then the output is combined with KCF (Kernelized Correlation Filter) on HOG in order to improve the tracking. Another work is presented in [9], this workconsists of 4 steps: head detection operation, back projection, hand rotation, andthen hand detection. Here, the human head information as color is used to be assistance of hand ROIs detection, by using the feature extraction as a histogram of oriented gradient (HOG) feature and Support Vector Machine (SVM) as a classifier. In 2016, a hand with wrist detection method for unobtrusive hand gestureis reportedin [10], the operation is implemented by using a head mounted display (HMD) where locates in upper body area of a user, and a depth camera under an HMD to extract the shape context features and SVM for the classifier. Another hand detection using facial information is presented in 2016 in [11], here detection of a face is the first step to pick up the face color so as to be used for regions of interest (ROI) extracting to detect hands, specially hand skin color and face skin color are almost similar, here,successful detection rate is up to 92%. Another hand detection presented recently in 2018 in [12], which is based on the convolutional neural network as a deep learning, This technique is based on the architecture of YOLOby utilizing the spatial-transfer connection (STC) between high-level layers and low-level layers, the multi-scale features from different layers can be aggregated for detecting the hands.Another work for hand detection based on statistical learningtraining way is introduced in [13], in which this idea was tested by Using Microsoft's Kinect sensordataset, which is the same database of the proposed work in this paper as well, here features for statistical learning whichapproximates with a Harr-like feature with the help of Adabooststatistical learning, gets the training model. Furthermore, idea of hand detection, which is used an extended histogram of oriented gradients (HOG) model named skin color histogram of oriented gradients (SCHOG) is presented in [14] to construct a human hand detector, firstly, features based on SCHOG are extracted by combining HOG with skin color cues, then support vector machine (SVM) algorithm is used for training the dataset and finally, this method is verified on the testing dataset for the SCHOG features. The hand is also detected in 2014 in [15], by employing a corner detector to figure out the problem of the finger fragment occurred during hand detection, the process of this detector is shrinking the ROI into a much smaller range while performing corner detection. Another work of hand detection utilized skin color filtering method based on skin color range modeled in YCbCr color space as in [3, 16]. It is worth to add that, the proposed method designed according to the color skin method as a first step then secondly, examining the object that depending on Flips_Numbers which will be explained in the next subsection.

3. THE PROPOSED METHOD

The process starts by extracting frames by frame from the video stream to be processed separately. The stages as a block diagram of the proposed hand detection are shown in Figure 2. After extracting frame, searching operation for skin color based on ranges of RGB color space will be started. The ranges of red, green and blues are modeled especially for white people. In other words, these ranges are not applicable to dark skin color people, thus searching operation based on the range of the skin color is implemented and illustrated as a bordered box around the hand as in Figure 3(1).

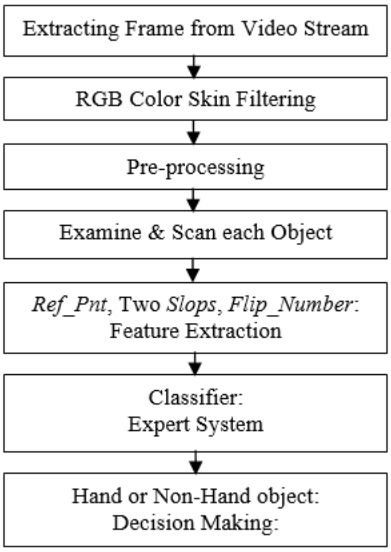
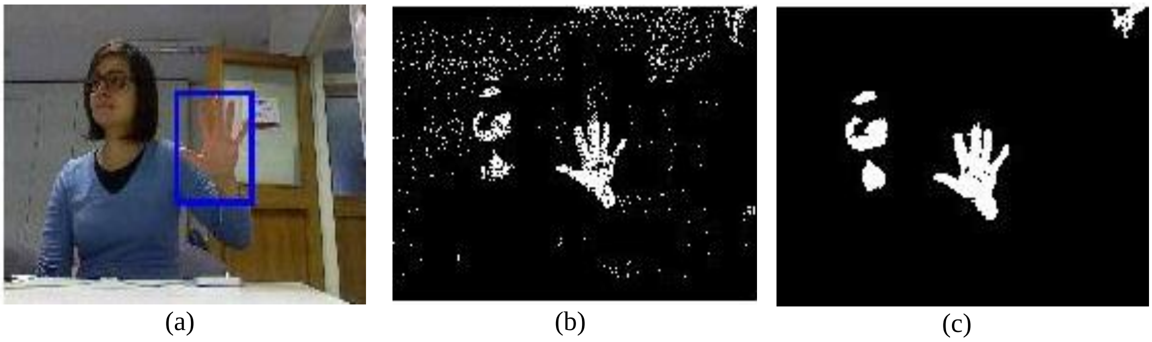

Extracting Frame from Video Stream
RGB Color Skin Filtering
Pre-processing
Examine & Scan each Object
Ref_Pnt, Two *Slops*, *Flip_Number*: Feature Extraction
Classifier: Expert System
Hand or Non-Hand object: Decision Making:

Figure 2. Methodology steps of hand detection and segmentation

The ranges for each color space (RGB) as red, green and blue are stated below, these rang are modeled especially for the color skin:

$$103 < \text{red\_color\_range} < 159 \qquad 74 < \text{green\_color\_range} < 103 \qquad 43 < \text{blue\_color\_range} < 98$$

Afterward,several substages of pre-processing in terms of image processing such as median filtering and removing some object that their areas are up to 300 pixels (as very small object area), then applying image dilate morphological filters in order to smoothly make the object connected. Figure 3(c) depicts the aforementioned steps of image processing, as it is shown that there are only 5 objects in Figure 3(c), one of them surely is the hand.Finally, a border is drawn around the hand for the original image frame to set the target region of interest (ROI) by extracting the four border points, then plotted to the original image as shown in Figure 1(a).



(a) (b) (c)

Figure 3. Pictorial illustration of the methodology steps as images

3.1. Proposed detected algorithm

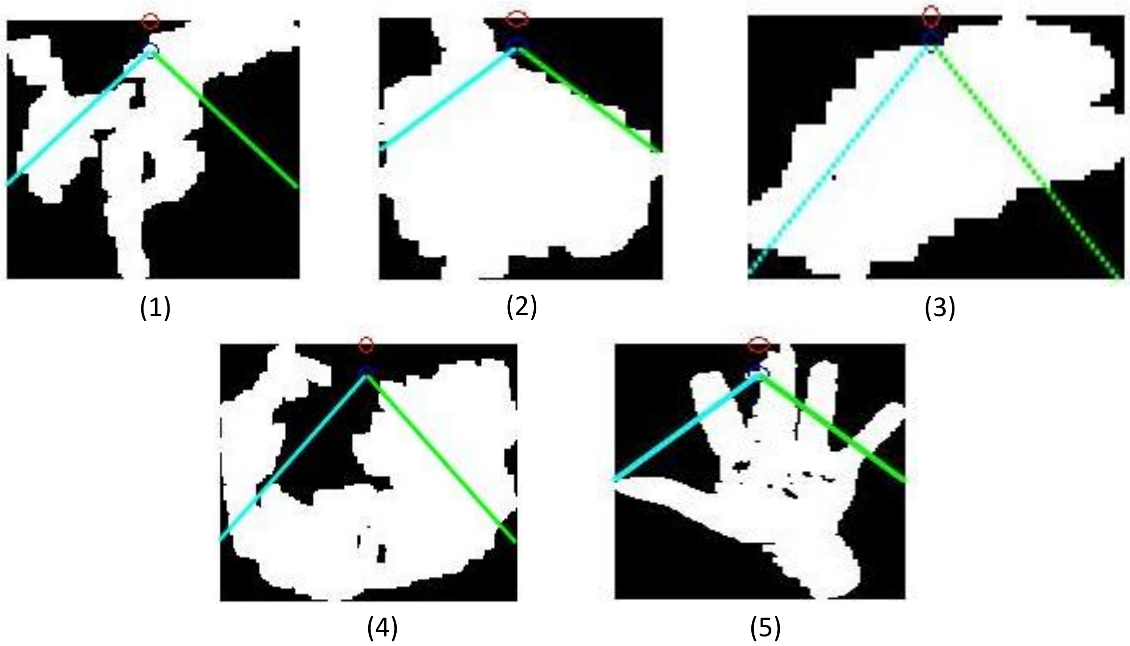
Once the ROI (object) is accessed, steps of the proposed algorithm are as following:

1. Get Point*(x,y)* in Top Middle of Object Image, and not at top middle of the object.
2. Sliding down of this point quarterly to be reference point (*Ref_Pnt*).
3. Draw right line slop from (*Ref_Pnt*).
4. Draw left line slop from (*Ref_Pnt*).
5. Merge left & right slops to be a smart scanned path.
6. Calculate *Flip_Number* based only on the scanned path

Pseudo-code describing 1 & 2 for extracting the *Ref_Pnt* is as following:

```
[x_maxy_max]=size(img);
top_ref_point_y=floor(y_max/2);
top_ref_point_x=1;
y_ref=top_ref_point_y;
qurt_top_x=fix((x_max/8));
   for i=1:qurt_top_x
    for j=1:y_max
      if ((img(i,j)==1))
x_ref=i;
     endendendref_pnt=[x_refy_ref];
```

where*Ref_Pnt* is the point that has the trajectories (*x)* and (*y)* to be considered as a reference of the branching the two slops left and right. Figure 4 depicts all the five objects that are possible to represent hand as these objects passed through the skin color filter. Next stage, examining operation is applied based on the proposed algorithm to extract the true hand object among the others. It is obvious that all objects have the red circle symbol, which is deemed as the top point then by getting down around half-quarterly, the blue circle symbol is considered as the *Ref_Pnt*. At this point, the two slops are branching left and right. The reason for branching is important to fully dissect the object type and to count how many flips the object has during the scanning operation. After that, point 3 of the algorithm is specifying the right slops. Firstly, the idea of extracting the right slop is by incrementing one (*y*) to both rows and columns to get new extracted scanned right slop as shown in Figure 4 with green color slop, the algorithm programming as pseudo-code is shown under the:***right_slop*pseudo-code**. Secondly, step 4 of the proposed technique is to draw a left slop starting from *Ref_Pnt* going down to the left end of the image as shown in Figure 4 of the pink color slop, the idea of extracting the left slop is by decrementing one to image columns and incrementing one to the rows to get new extracted scanned left slop, the algorithm programming as pseudo-code is shown ***left _slop* pseudo-code**.


(1) (2) (3)

(4) (5)

Figure 4. Depicting *Ref_Pnt* with left and right slops as smart path of an image object

right_slop **pseudo-code**

```
right_slop_x=0;right_slop_y=0;
  rx=0;ry=0;size_slop=1;
  for y=1:size_slope:(y_max-y_ref)-1
  y_new=y_ref+y;
  ry=ry+1;
  right_slop_y(ry)=y_new;
  x_new=x_ref+y;
  rx=rx+1;
  right_slop_x(rx)=  x_new;
  end
  plot(right_slop_x,right_slop_y);
```

left _slop **pseudo-code**

```
 size_slop=1;
 left_slop_x=0;left_slop_y=0;
 lx=0;ly=0;
 left_size=y_max-length(right_slop_y);
 while (left_size~=0)
 left_size=left_size-1;
 lx=lx+1;
 x_new=x_ref+lx;
left_slop_x(lx)= x_new;
ly=ly+1;y_new=y_ref-ly;
left_slop_y(ly)=y_new;
end
plot(left_slop_x,left_slop_y);
```

After specifying the right and left slop and merging them to be one scanned (smart) path, now it is ready to extract the *Flip_Number*based scanned path. Here, to guarantee drawing slops without missing any other finger of the hand to be detected. In this situation, the idea is extracting the *Flip_Number* from the right and left slopes, and then adds them together to be the final *Flip_Number*of the object. The *Flip_Number* is defined as the pixel brightness changing from “0” to “1” or “1” to “0” if any changing happened, then a counter will be increment by one, eventually, this counter will be represented as the *Flip_Number*. The two pseudo-codes of extracting the *Flip_Number* of both right and left slops are below:

Flip_Number **from right slop**

```
fst_value= obj_img(ref_pnt(1,2),ref_pnt(1,1));
  [x_num,y_num] = size(obj_img);flip_num_right=0;
   for p=1:length(right_slop_x)-2
   ifx_num>right_slop_x(p)&&y_num>right_slop_y(p)
   if (  fst_value ~=
obj_img(right_slop_x(p),right_slop_y(p)))
   fst_value= obj_img(right_slop_x(p),right_slop_y(p));
   flip_num_right=flip_num_right+1;
   end
  end
end
```

Flip_Number **from left slop**

```
flip_num_left=0;
 fst_value= obj_img(ref_pnt(1,2),ref_pnt(1,1));
 for k=1:length(left_slop_x)-2
 ifx_num>left_slop_x(k)&&y_num>left_slop_y(k)
 if (  fst_value ~= obj_img(left_slop_x(k),left_slop_y(k)))
 fst_value= obj_img(left_slop_x(k),left_slop_y(k));
flip_num_left=flip_num_left+1;
 end
end
end
```

Now, summation between *flip_num_left*and *flip_num_right* to produce *Flip_Number*feature as:

flip_number=flip_num_left+flip_num_right;

Next, classification is responsible to detect which object is hand and non-hand, as assumed the hand to be detected is shown in Figure 1, which has five fingers are pointed out. Accordingly, that image if this proposed algorithm is applied to it, the Flip_Number must equal to 10 flips. However, 10 flips are challenging because sometimes problem rises related to image processing filtering and skin coloring search. Therefore, decreasing this challenge 10 to 8 or 6 Flip_Number to be detected is preferable. However, once decreasing the challenged Flip_Number, the False Accept (FA) will be increased.

An expert system is defined as a computer system that emulates the decision-making ability of a human expert. In rule-based expert systems, forward chaining inference techniques is used in this research. The domain knowledge is represented by a set of IF-THEN in order to produce rules and the data is represented by a set of facts about the current situation, which is represented by feature named the Flip_number. The inference engine must decide when the rules must be executed. Forward chaining is used in this paper because of the similarity to the methodology that depends on data-driven reasoning. The reasoning starts from the known data and proceeds forward with that data. Each time, only the top rule is executed, and when executed, the rule adds a new fact to the database. Any rule can be executed only once. The pseudo-code of the expert system is shown below:

```
if (flip_number>=  10)
disp('Hand Detected');
else
disp(' No detection');
end
```

As it is shown in Figure 4, there are five objects generated during preprocessing and filtering the image, the algorithm should be applied to all generated objects to extract the *Flip_Number*feature and only one object is predicted by the expert system as the true hand among the others.

4. EXPERIMENT

To evaluate the proposed method, experiments have been conducted on 100 images, which is exploited a dataset named Dataset of Leap Motion and Microsoft Kinect hand acquisitions issued in University of Padova (Italy), 2014 [17]. The size of each image here is Kinect color map (1280 x 960). These images are rgb.png extension. This dataset contains 10 gestures in general, of different hand gesture, like the assumption of the current research is to detect a hand with a style of five fingers pointed out, so that, one gesture which is labeled gesture 9 in this database is selected only for the 10 individuals. In other words, ten rgb.png images have been used for each individual, totally 100 images exploited in the test. It is worth to mention that in this algorithm there is no need to train images. Just test any image, so that all the 100 images are used for testing to extract the total accuracy. Usually, in the verification or identification comparison, there are two possible errors to be measured: False Accept Rate (FAR) error, which results from the forged template that accepted by a system falsely during testing and the second error is False Rejection Rate (FRR), which results from the genuine template that the computer predicted it wrongly [18-20]. Generally, the overall accuracy of the proposed system is calculated by subtracting the average error rate from 100% as in (1):

$$Accuracy\% = 100\% - \frac{FAR + FRR}{2} \tag{1}$$

However, this research has no FAR error, since there are no forge hand images in this experiment to be tested. Therefore, FAR is considered to be zero. However, FRR is used for the testing to assess the recognition rate, because these hand images are considered as genuine templates. In case they are wrongly recognized by the proposed system, then the FRR increases. The equations that are used to measure the accuracy of this research are in (2) and (3):

$$Accuracy\% = 100\% - FRR\% \tag{2}$$

$$FRR\% = \frac{Total\_False\_\mathrm{Re}\,ject}{Total\_True\_Attempt} \times 100\% \qquad (3)$$

Matlab 2016b as a workstation has been used in this experiment installed into a computer, which has the following characteristics core2due, 4 G-RAM.

5. RESULTS AND DISCUSSION

The reported results in this research fall into two types: pictorial and statistical results. The pictorial result is depicted in Figure 5, in which it contains 5 images, the first one in Figure 5(a), which is the detected hand object as ROI successfully. In Figure 5(b) illustrates the output of the RGB color skin search based on RGB ranged as aforementioned in the methodology section. It is clear that contains many objects and noise, after removing noise using some preprocessing tools such as median and morphological filter the result is illustrated in Figure 5(c), here it is clear finally contains two objects only, in which surely one of them is the hand and the other is a non-hand object.

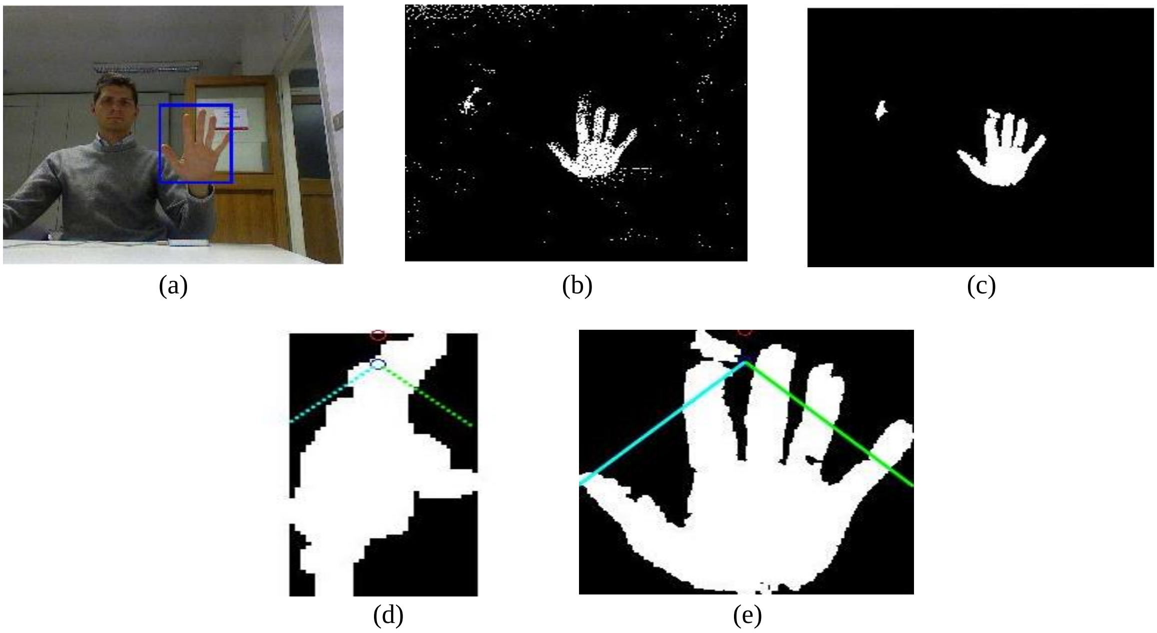


Figure 5. Applying the proposed algorithm of a dataset sample image and its pictorial results

After applying the proposed algorithm to count the *Flip_Number*and examine the objects, which are extracted in the image, results will be outcome either “No Detection”as shown in the Figure 5(d) because, it is clear the *Flips_Number* is 2, which does not satisfy the expert system condition to be announced as the object is non-hand. Or “Hand detected” result as depicted Figure 5(e) due to the clearness of the*Flips_Number* is 10 in which satisfies the expert system condition to be announced as the object is the hand that contains 5 fingers. In terms of statistical results as shown in Table 1, in case the *Flip_Number* is 6, which is indicated to the easy prediction, because due to the image capturing and filter that is not all the five fingers will be appeared, therefore, permissible is allowed to be 6, 7 or might be 8 as less permissible. However, in case of decreasing the rigid of the hand prediction that means False Rejection Rate will be increased,this means that objects might be hand truly,and the system wrongly rejects them due to the rigid condition of the test and decision making. However, 100 images with a cluttered background got accuracy 97% and 84% in case *Flip_Number* set to 6 and 7 respectively. In case *Flip_number* is set up to 8 and 10, the result is recorded in Table 1 as well. The accuracy is 81% and 48% respectively.

Table 1. Accuracy reported with four cases 6,7, 8 and 10 flip_number

	Flip_Number = 6			Flip_Number = 7			Flip_Number = 8			Flip_Number = 10		
User ID	FR	FRR	Accuracy %	FR	FRR	Accurac %	FR	FRR	Accuracy%	FR	FRR	Accuracy %
1	0	0	100	2	20	80	2	20	80	6	60	40
2	0	0	100	0	0	100	0	0	100	2	20	80
3	3	30	70	5	50	50	5	50	50	8	80	20
4	0	0	100	2	20	80	3	30	70	6	60	40
5	0	0	100	2	20	80	2	20	80	8	80	20
6	0	0	100	0	0	100	0	0	100	4	40	60
7	0	0	100	4	40	60	5	50	50	7	70	30
8	0	0	100	0	0	100	0	0	100	2	20	80
9	0	0	100	0	0	100	0	0	100	5	50	50
10	0	0	100	1	10	90	2	20	80	4	40	60
	Total Accuracy =		97%	Total Accuracy =		84%	Total Accuracy =		81%	Total Accuracy =		48%

It is noticed that with larger Flip_Number is setting up, less accuracy is recording. However, the recognition rate is low especially with 10 Flip_Number, because the selecting specification of the hand object becomes rigid and challenging.As the author’s perspective, the most suitable Flip_Number in terms of simplicity is 7 or 8, as in the middle between severing and easy condition. Because sometimes users have a cut-finger or mis-counting as the slop is not passing through the fingers due to hand rotating. However, the opposite case that might an object is not a hand but it has Flip_Number as 10 or more, such this case is depicted in Figure 6, in which this object appeared during conducting the experiment as hand but it is indeed a non-hand object. This is considered as the weak point of the proposed algorithm. In the same time, the proposed algorithm is suitable for the lightweight devices, as there is no training data for the prediction.

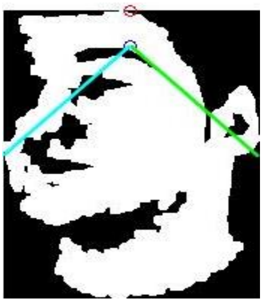


Figure 6. Object classified as a hand because of containing 10 flip_numbers

In terms of comparison with several published works regarding hand detection, Table 2 lists the recent works with their methodologies compared with the proposed work. It is clear that from the accuracies listed in Table 2, the proposed algorithm result can offer knowledge contribution by a new methodology with an acceptable recognition rate.

Table 2. The accuracy of the proposed methodology compared with recent several hand detection works

No.	Methodology	Accuracy	Citation/Year
1	Hand Segmentation by deep learning: CNN (Convolutional Neural Network)	96%	[21] / 2019
2	Hand Recognition: Skin and Wrist Detection by PCA + Euclidean distance.	93%	[22] / 2019
3	Hand Segmentation and Fingertip Tracking from Depth Camera Images Using Deep Convolutional Neural Network and Multi-task SegNet.	83%	[23] / 2019
4	Proposed method	96%	

6. CONCLUSION

With an improvement happened in computer vision and machine learning in the fields related to human-computer interaction, hand detection researches are becoming important among researchers. In this paper, a new algorithm has been proposed and tested so as to predict hand verses non-hand object in an image that contains a complex background. The operation is kicked off by searching on color skin objects, then examination operation for each object is performed by the proposed algorithm, which is predicting a reference point (Ref_Pnt) in the object then drawing a right line slop from Ref_Pnt and drawing a left line slop from Ref_Pnt then merging them to be a smart scanned path. Finally, computing the Flip_Number, which is based only on the scanned path, acts the feature of this system. The conducted experiments were performed by using 100 hand images originated from random 10-individual taken from dataset named Dataset of Leap Motion and Microsoft Kinect hand acquisitions. The performance of the proposed algorithm is up to 84% and 81% in case the Flip_Number feature is 7 and 8 respectively. For the future work, the proposed algorithm might be developed by enhancing the accuracy by adding another examining idea to boost the hand object result and including dark skin hand detection as well.

REFERENCES

[1] Patil, N.M. and S. Patil, "*Review on real-time EMG acquisition and hand gesture recognition system*," in Electronics, Communication and Aerospace Technology (ICECA), 2017 International conference of. IEEE, 2017.

[2] Malallah, F.L., et al., "A Review of Biometric Template Protection Techniques for Online Handwritten Signature Application,*"International Review on Computers and Software (I. RE. CO. S.),* 8(12),2013.

[3] Malallah, F.L., et al., "Hybrid Hand-Directional Gestures For Biometric Based On Area Feature Extraction And Expert System,"*Journal of Theoretical & Applied Information Technology*, 95(23),2017.

[4] Malallah, F.L., et al., "Irreversible Biometric Template Protection by Trigonometric Function,"*International Review on Computers and Software (IRECOS),*11: p. 1138-1146, 2016.

[5] Sonkusare, J.S., et al. "*A review on hand gesture recognition system*,"in Computing Communication Control and Automation (ICCUBEA), 2015 International Conference on, IEEE, 2015.

[6] Kaur, H. and J. Rani. "*A review: Study of various techniques of Hand gesture recognition*,"in Power Electronics, Intelligent Control and Energy Systems (ICPEICES), IEEE International Conference on, IEEE, 2016.

[7] Sulyman, A.A., et al., "Real-Time Numerical 0-5 Counting Based On Hand-Finger Gestures Recognition*," Journal of Theoretical & Applied Information Technology*, 95(13), 2017.

[8] Wang, Y.-R., W.-H. Lin, and L. "Yang. *A fast hand motion detection based on FIFO*,"in Machine Learning and Cybernetics (ICMLC), 2017 International Conference on. IEEE, 2017.

[9] Kim, J., J. Baek, and E. Kim. "*A part-based rotational invariant hand detection,"* in Fuzzy Theory and Its Applications (iFUZZY), 2013 International Conference on. IEEE, 2013.

[10] Oh, J.Y., et al. "*A hand and wrist detection method for unobtrusive hand gesture interactions using HMD*," in Consumer Electronics-Asia (ICCE-Asia), IEEE International Conference on. IEEE, 2016.

[11] Kaur, T., J. Gambhir, and S. Kumar. "*Arduino based solar powered battery charging system for rural SHS*," in Power Electronics (IICPE), 2016 7th India International Conference on. IEEE, 2016.

[12] Le, T.-H., et al. "*An efficient hand detection method based on convolutional neural network*," in 2018 7thInternational Symposium on Next Generation Electronics (ISNE), IEEE, 2018.

[13] Li, H., et al. "*Hands detection based on statistical learning*," in *Computational Intelligence and Design (ISCID), 2012 Fifth International Symposium on*, IEEE, 2012.

[14] Meng, X., J. Lin, and Y. Ding. "*An extended HOG model: SCHOG for human hand detection,"*in Systems and Informatics (ICSAI), 2012 International Conference on. IEEE, 2012.

[15] Wang, Y.-R., W.-H. Lin, and L. Yang. "*An improved hand detection by employing corner detector,"*in Machine Learning and Cybernetics (ICMLC), 2014 International Conference on, IEEE, 2014.

[16] Malallah, F.L., et al., "Vision-Based Control By Hand-Directional Gestures Converting To Voice,"*International Journal Of Scientific & Technology Research*. 7(7).

[17] Marin, G., F. Dominio, and P. Zanuttigh. "*Hand gesture recognition with leap motion and kinect devices*,"in Image Processing (ICIP), 2014 IEEE International Conference on. IEEE, 2014.

[18] Malallah, F.L., et al., "Online handwritten signature recognition by length normalization using up-sampling and down-sampling,"*International Journal of Cyber-Security and Digital Forensics (IJCSDF*), 4(1): p. 302-313, 2015.

[19] Malallah, F.L., et al., "Non-Invertible Online Signature Biometric Template Protection via Shuffling and Trigonometry Transformation,"*International Journal of Computer Applications*, 98(4),2014.

[20] Alkassar, S., et al., "Robust sclera recognition system with novel sclera segmentation and validation techniques,"*IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(3), p. 474-486, 2017.

[21] Rony, M.R.A.H. and M.M. Alam. "*Hand gesture recognition using image segmentation and deep neural network*,"in Eleventh International Conference on Machine Vision (ICMV 2018), International Society for Optics and Photonics, 2019.

[22] Yadav, K., et al., "Hand Gesture Recognition using Improved Skin and Wrist Detection Algorithms for Indian Sign," *Journal of Network Communications and Emerging Technologies (JNCET) www. jncet. org*, 9(2), 2019.

[23] Nguyen, H.-D. and S.-H. Kim, "*Hand Segmentation and Fingertip Tracking from Depth Camera Images Using Deep Convolutional Neural Network and Multi-task SegNet,"* arXiv preprint arXiv:1901.03465, 2019.