ISSN 2229-5518

Novel Algorithm for Multi Hand Detection and Geometric Features Extraction and Recognition

Mokhtar M. Hasan, Pramod K. Mishra

Abstract— The recent trends for house appliance devices are moving towards the natural interaction that ensures free-cumbering interaction with these human-made devices, we have presented herein our novel approach for features extraction template that enables such devices to interact in better way especially when gesturing to vision-based devices such as home TV, these vision-based devices preferred over tis competitions since its no-frills communication, robot and video game interaction without any sensors or extra hardware just like human communicates with his same species, our algorithm extracts the hand gesture structure which are palm, wrist and fingers with their corresponding features like their locations, fingertips locations, finger bases locations, wrist location and their order from left finger to right finger regardless the hand orientation even upside down or any other angles, these features are important for latter approaches for vision-based algorithms and hand/palm/fingers tracking algorithms, we have classified the finger(s) according to their five classes that are thumb, index, middle, ring, and pinkie; these fingers have been classified using Gaussian likelihood function as a classifier regardless which hand is presented left or right and without any prior assumption of the pose of the hand, any hand and any orientation, after the finger's classification we have proceeded with hand gesture recognition by preserving one binary bit for every finger and gesture indexing done perfectly, we have considered all the outcomes of such finger's raising that are 32 combinations, our system can detect multi-hands that contained in a single image frame as well as recognition step, we have applied circular templates by using dynamic template matching with two different radiuses for each of fingers and palm premises respectively, we have achieved a perfect classification of multi-hands with their corresponding palm/fingers features for different samples as well as the recognition results and we have listed the recognition percentages for finger-wise and hand-wise as well as their processing time.

Index Terms—Finger Tracking, Palm Tracking, Finger Detection, Palm Detection, Hand Gesture Recognition, Interactive Systems, Dilation Morphological Operations.

1 Introduction

The curious and interferences habit of the human kind makes him look always for imitation of his surrounding environment, the keyboard and mouse that are used for manual human computer interaction have become off the shelf and should be replaced by some other devices since they are no longer accepted compared to the level of technology the human achieved nowadays, the new device is the hand gesture [1] which is used for controlling and carrying out the issued command to the computer; but, however, this device needs a corresponding translation that can translate the meaning of current pose; which is the mathematical equations [2], and the cumbersome level of such environment of human computer interaction can be reduced dramatically especially when the user needs to bend his body for using of old devices like keyboard which needs direct touch, however, the requirements of providing natural interaction demanding by entertainment and augmented reality [3] can be accomplished as well.

Vision based and glove based are considered to be the main two devices for gesturing, vision based considered more affordable and has less complexity in which the user can wave his hand without considering the wires issue, while the glove based is based on sensors that attached to a glove and connected to computer through wires, these wires provides more cumbersome and awkward [1][4] and the level of neutrality is reduced [5] and we should restrict to wire length, such device is called Cyber Glove [1][5][6][7][8] regardless of the diseases that can infect the individuals due to multiple use of these gloves [7], there is another kind of vision based which depends on glove marker [7][8][9][10] technique, this technique enforces the user to wear a colored glove which simplifies the hand detection operation.

One more classification for vision based; appearance based and 3D model [1][4][7][8][11][12], appearance based can be defined as the matching of features between the input hand appearance and a set of predefined features extracted from trained appearance [1][4][7]. 3D hand model uses a set of joint angle parameters and the length of these segments to form the hand features [4][7].

We have applied a novel approach for the extraction of appearance based features from an input image using dynamic templates, we have used double templates with different radiuses to achieve our goal, we have classified the finger's class as well as the hand gesture, the finger classification done through the Gaussian likelihood function in which the features have been toned with statistical model, the hand gesture has been recognized using binary bit indexing method, furthermore, our system can classify an input image with multi-hand object with providing the features for each hand.

- *Mokhtar M. Hasan is currently pursuing his Ph.D. degree program in computer science department in Banaras Hindu University, Varanasi, Uttar Pradesh, India, PH-0091-9565198049. E-mail: mmwaeli@gmail.com*
- *Pramod K. Mishra is currently Associate Professor in Banaras Hindu University, Varanasi, Uttar Pradesh, India, he is supervising my research work, and he is interested in parallel and distributed computation, computational complexity, parallel and clustered data mining, high performance computing, VLSI algorithms, image processing, pattern recognition, and computer vision. E-mail: pkmisra@gmail.com*

2 Related Work

Appearance based model has been applied by [11][13][14][15], the authors in [11] used a single Gaussian function to formulate the hand color in HSV for segmentation purpose, and the palm and fingers of the hand have been located by the scale-space feature detection by finding the local maxima of the square of the normalized laplacian operator, in [15] the authors extract the fingers and the palm by using the Self-Growing and Self-Organized Neural Gas (SGONG), they cover the structure of the hand with neurons and they detects the fingers and palm as well. Some other techniques for extracting the fingers by looking for the peaks and valley for the hand contour [14], a review paper for geometric features extraction and modeling along with recognition can be found in details in [7].

3 System Structure

Our proposed algorithm consists of three main stages; these stages are shown as in Fig. 1.

The input image may contain one or more hand objects, these objects should not be occluded, the user can use any combination of multi-hand gesturing in the input image, and the system is responsible for clustering these hands using distance technique which is achieved by dynamic template matching for attaching each finger to its hand that belongs to.

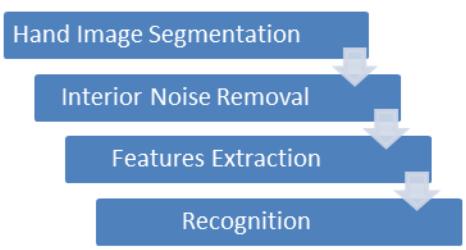


Fig. 1. Proposed system architecture.

4 Morphological Operations: Review

This term is originally came from the medical branch, which means there; ***"the branch of biology that deals with the structure of the organisms without any consideration for its function"***, when apply this term on image processing it gives the similar definition but the organs in this time be the objects inside the input image, image morphological operations refer to producing output image in which each output pixel of this image is a reference point for a shape- driven filter convolved with same size window of image object; usually; the reference point is the center of that shape-driven filter and that shape driven filter is called the structuring element (SE) which controls the relationship between the neighborhood of the pixels in this image window for production of this reference point. There are two main operations for morphological operations which are dilation and erosion that forms the root of any other composed operation.

4.1 Dilation Operation

Let us start with the formal definition of this term, we have taken the definition from the reference of English Language which is Oxford Dictionary [16],"***to become or to make something larger, wider or more open"***, this means that the expansion of the image object to a bigger size, but this expansion is controlled of course by the adopted SE, this operation is denoted by $\oplus$, Equation (1) shows the mathematical definition of the dilation operation for binary images.

$$A \oplus \mathrm{B} = \{r(B) \mid hit_B(A)\} \tag{1}$$

Where B is the SE applied and r (B) is the reference point of that SE which is B, and the $hit_B(A)$ means $\mathbf{A} \cap \mathbf{B} \neq \mathbf{\varnothing}$.

4.2 Erosion Operation

As we started in the definition of the dilation operation, we will do the same herein by starting with the definition of the erosion operation and also taken from Oxford Dictionary [16], ***"to gradually destroy the surface of something through the action of wind, rain, etc."***, herein, there is no wind nor rain but there is SE that controls the destroying process, this operation is referred by $\ominus$ which is translated mathematically as in Equation (2):

$$A \ominus \mathrm{B} = \{r(B) \mid fit_B(A)\} \tag{2}$$

Where $fit_B(A)$ means the coordinates of the convolution of A and B should completely done within the premises of A, in either meaning, $\mathbf{A} \cap \mathbf{B} \subseteq \mathbf{A}$.

For other mathematical definitions for these two operations, you can refer to [17][18][19][20] for binary and gray scale image morphological operations.

We have used modified erosion operation for filling and closing up the hand holes if existed since our template matching technique tries to fit the template with the hand object area's without any interior noise that may occur inside the hand object, for more details about this new algorithm for filling the hand object and removing the interior noise, please refer to [22].

5 System Implementation

As we mentioned in Fig. 1, there are four stages, the first stage and the most important stage and all the latter stages depends on is the segmentation operation, after this stage; some noises may creep inside the image during the first stage or there are some embedded noise from the original image [21], we have applied our modified approach for noise removing by filling the interior holes that lies within the hand premises, then, we have to apply our novel algorithm for fingers/palm/wrist locating and features extraction, finally, recognition stage should take place to fulfill the gesture recognition system task, we have labeled segmentation and noise removing as subsections since they are considered as preprocessing steps, the other stages have been given main headlines due to its importance.

ISSN 2229-5518

5.1 Segmentation Operation

This operation tries to segment the hand object and removes the background region by using of HSV color space, you can refer to [2] for skin color-based segmentation, Fig. 2 shows the application of the segmentation operation, we will not focus and spend more space for segmentation operation since it is already covered by many researchers and we will focus on our proposed algorithm.

5.2 Interior Noise Removing

We have removed the noise of the interior hand object by modifying the algorithm found in [17], the initial maker used in [17] is slow and consumes time for accomplishing its task, we have proposed a different marker as in [22] to speeding up the processing time as well as the accuracy of the output since there are some cases were missing by the original version of the algorithm, the following (3) and (4) show initial marker as set by [17] and [22] respectively.

$$F(x,y) = \begin{cases} 1 - A(x,y)\ , if\ (x,y) is\ a\ border\ pixel\ of\ A \\ 0 \qquad otherwise \end{cases} \quad (3)$$

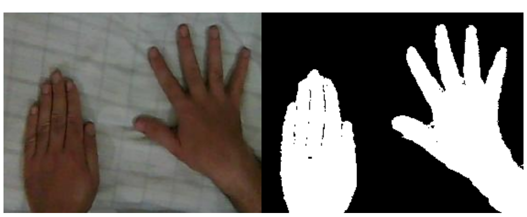
$$F(x,y) = \begin{cases} 1 \quad if\ there\ is\ a\ path\ from\ border\ to\ (x,y) \\ 0 \qquad otherwise \end{cases} \quad (4)$$

We have applied a horizontal path for (4), Fig. 3 shows the application of the modified algorithm with Fig. 2b as an input.

6 Hand Structure Extraction

The palm/fingers/wrist extraction consists of several stages; these stages will be illustrated herein below, Fig. 4 shows the steps of the feature extraction phase.

a b



a: input image. b: segmentaed image using HSV color model.

Fig. 2. Hand object segmentation with HSV color space.

As brief illustration of Fig. 4, we have used two different templates which are palm circle template (PCT) and finger circle template (FCT) , the first step is to distribute the PCT across the hand; second step is to distribute FCT across the hand object as well, after these two steps we will have two different areas filled with dots as a result of templates matching in which each template leaves its center as an output for matching which is a dot with its corresponding coordinates, so, we need to oust the finger dots that share the palm area with palm dots in a process called ousting operation that is step three, after this step we will see the palm dots in palm area and fingers dots in finger area, step four is the labeling operation in which each fingers dot belong to a one single finger are grouped together, so, the output of this step is a collection of groups as well as palm group (wrist area also included in the palm area so far), and some other groups may be produced incidental at palm corner as we will see later, these groups are not palm neither fingers area, so, step five removes these groups and step six detects the wrist area in which we have applied an algorithm that detects the wrist regardless the direction of the hand and with no restriction and can work under any hand pose unlike [15] which assumed that the hand should be in only one direction which is up righted and many others as well, the last step of feature extraction stage is to calculate the features of the hand object.

Fig. 3. Application of interior noise removing algorithm.

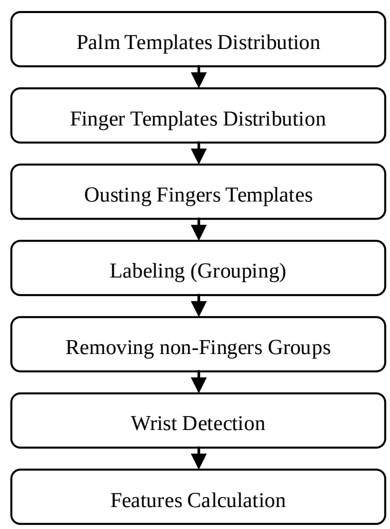


Fig. 4. Feature extraction steps.

6.1 Determining Template Size

Each one of the PCT and FCT has a different radius, we have used the advantage of being the finger premises has a narrower structure than in palm, and this characteristic enabled us to design an algorithm to trace the fingers as well as the palm by using of dynamic templates.

FCT radius can be any radius that fits in the finger premises and it is changeable, the change in size depends on the factor of processing time, the biggest the fastest since reducing the representative number of dots per finger reduces the processing time, however, the average number of dots for representation a single finger is less than 80 dots which has a significant processing time, the size of PCT should be chosen so that the corresponding template cannot fit into the finger premises, may a question arises here that is the FCT will fit the palm premises as well, that is true and the remedy will be explained later, we have applied the diameter of 11 and 70 for each of FCT and PCT respectively, however, after deciding the size of each we can keep them unchanged and in the case of depth distance is not fixed with computer camera we have to apply scaling operation to ensure the hand object fits within predefined scale.

6.2 Template Matching

We have modified the template matching in order to speeding up the processing of the extracted region; the new definition is as Equation (5).

$$matching(A,B) = \{\, z | fit_B(A) \& \, abs(z_i - z_j) \geq r, \forall i \neq j \,\} \tag{5}$$

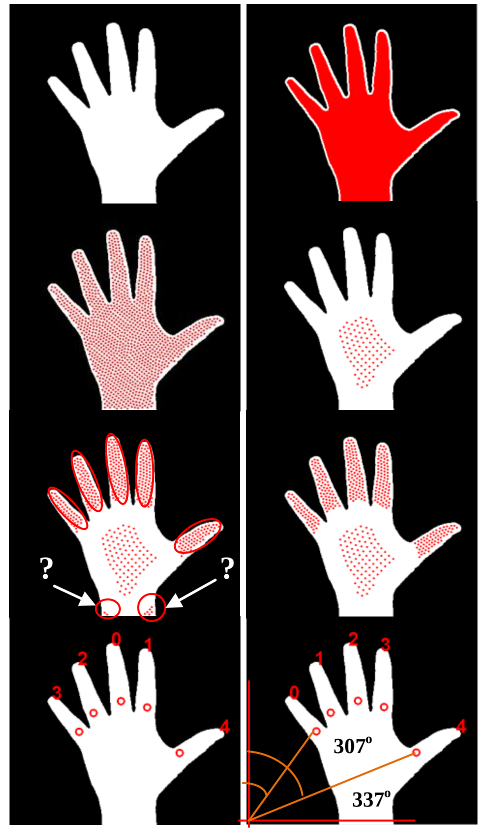
Where A is the hand image window and B is the template with radius r, the above equation can be interpreted as the resulting of template matching operation is the collection of circles with their centers touch the borders of each other which is called half-matching, the classical template matching is fully-matching that is ignored by us due to horrible time for matching, the same equation have been applied on the input segmented hand object with two different radius r used for each of FCT and PCT, Fig. 5b shows the traditional template matching and Fig. 5(c and d) show the application of Equation (5) with the two mentioned radiuses respectively, it is worth to mention that the radius respective for PCT will produce lower number of dots for representation palm/wrist area if we consider Equation (5) since a radius of 70 is applied, so, for producing more dots in palm/wrist area we have adopted eighth-matching, in either meaning, the r dedicated for PCT is divided by 4 (diameter by 8) and Equation 5 is applied successful and the output as seen in Fig. 5d.

6.3 Ousting

The solution of the problem mentioned in section 6.1 (second paragraph) is answered herein, the dots that produced from the PCT application dislodge any dots produced by FCT from their premises of radius r (radius of PCT and be increased with some threshold to ensure the dots of finger fit inside the finger premises completely), this will cause that the palm area and the finger(s) are clarified, as in Equation (6).

$$oust_{PCT}(FCT) = \{\, F | abs(f_i - p_j) \geq r, \forall i, j \,\} \tag{6}$$

Where f_i is the (x, y) coordinates of the finger dot and p_j is the palm dot for the matched circles, and F represents all the finger's dots that passed this filter, this means that the finger dots that their centers are away from palm dots will survive, Fig. 5e shows the ousting operation, after that the remaining FCT dots are grouped group considering the geometric distance, Fig. 5e shows 8 groups (including palm dots group).



a b
c d
e f
g h

a: segmented hand. b: fully-template matching. c: half-template matching. d: eighth-template matching. e: ousting process. f: removing of non-fingers groups. g: initial numbering. h: continuous numbering (finger indexing).

Fig. 5. The steps of our suggested system for feature extraction.

6.4 Removing the non-Fingers Groups

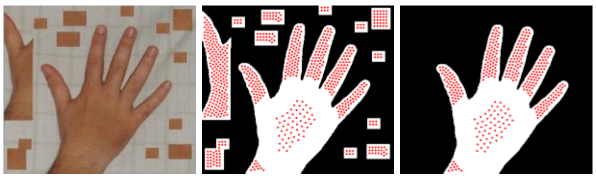
As noted in Fig. 5e, there are some groups of dots which are not fingers, those groups can be identified and removed by applying two different techniques:

1. Pixel Count Thresholding Technique (PCTT).
2. Distance Thresholding Technique (DTT).

We have applied those two techniques together, the former technique is obvious by removing the group that formed by a small number of dots less than the threshold of number of dots in a finger, the latter one employ the removing of the group of pixels that has a length less than a distance threshold (standard length of finger), which means in finger dots; we should find a dot has a distance greater than specific that distance threshold which is fingertip dot, if such dot spotted then the corresponding group will be declared as a finger group otherwise will be eliminated all these groups that survive should have a touch at some dot with palm area.

ISSN 2229-5518

One advantage of this method is the ability to remove any exterior noise since there is no palm area created elsewhere, and in the same technique we can cluster the second hand since it will has its own fingers that touch its palm premises, Fig. 6 shows the exterior noise removing as incidental outcome of our method, Fig. 5f shows the resulting of non-fingers group removing.



a: input image. b: dynamic templates matching. a b c
c: removing of non-fingers groups.

Fig. 6. Exterios noise removing applied incidentally by our algorithm.

As seen by Fig. 6, whatever the size and the place of exterior noises will be removed since they do not touch palm premises, and if there are noises with PCT size, we can set a threshold for number of dots in palm area and any area with less than this threshold will be neglected as well, Fig. 6c shows the hand area alone after removing the non-finger groups and maintain the palm area and finger's groups.

6.5 Finger Sorting

After the finger's groups have been decided, these groups should be sorted (indexed) with some sequence numbers, we have indexed these fingers from left finger to right finger regardless the direction of the hand object, as seen by Fig. 5g, these fingers have been assigned an initial sequence reflects its order of encountering during the template matching that applied raw major traversal, these fingers can be re-sorted correctly by applying several steps.

Our finger indexing algorithm based on finding the two neighbors of each finger, after obtaining those two neighbors we can track them to discover the two border fingers, those border fingers are not necessary to be thumb and pinkie since we can use two, three or four fingers of any combinations, however, after discovering those two border fingers we shall need one more clue about which one is on the left side and which one is on the right side, we should notice that in case of upside down hand, the left side in the image is our right finger normally and vice versa, however, we have to build a robust algorithm to find such distributions regardless the hand direction is, and this is what we have done.

6.5.1 Locating the Border Fingers

The first step is by calculating the distance matrix which represents the distance between all the bases of the extracted fingers, Table 1 shows these data that correspond to Fig. 5g, it is worth to mention that the fingertip and finger base can be obtained by adopting the closest and farthest finger dot to/from the palm area respectively.

Now, we need to apply one more step to finish the setting of the border fingers, we shall find the minimum value from distance matrix that not considered yet and of course non-zero value.

TABLE 1
DISTANCE MATRIX

Finger Number	0	1	2	3	4
0	0	43	47	83	125
1	43	0	86	115	89
2	47	86	0	37	152
3	83	115	37	0	164
4	125	89	152	164	0

Now, the shadowed cells in Table 1 represent the first neighbor of each corresponding finger, we shall refer to the neighborhood relationship as follows, (0, 1) means finger 0 has a neighbor of finger 1, (finger 0, neighbor 1)=(0, 1), however, from Table 1, we have obtained the following first neighborhood N1={ (0, 1), (1, 0), (2, 3), (3, 2), (4, 1) }, the second neighbor can be obtained from reversing that order and updating the neighborhood accordingly if not already there, follow our example, N2={ (1, 0), (0, 1), (3, 2), (2, 3), (1, 4) }, the four pairs have already there and the last one will change the set, the neighborhood matrix will be as follows in Table 2.

TABLE 2
NEIGHBORHOOD MATRIX

Finger Number	First neighbor	Second neighbor
0	1	-
1	0	4
2	3	-
3	2	-
4	1	-

Considering Table 1, the next minimum value is 47 which has the index (0, 2), by setting this neighborhood pair and its reverse which are N={ (0, 2), (2, 0)}, by this setting we have finished the border fingers locating as seen by Table 3 since the border finger has just one neighbor, we shall move to next step that is deciding the rightmost and leftmost finger, as an output of this stage, see Fig. 5h in which finger indexing has been done.

TABLE 3
NEIGHBORHOOD MATRIX

Finger Number	First neighbor	Second neighbor
0	1	2
1	0	4
2	3	0
3	2	-
4	1	-

6.5.2 Finding the Origin of the Coordinates

The next step is to locate the origin of coordinates that will be used as a reference point for calculating the angles of those extracted border fingers, we have adopted previously the palm center as origin of coordinates and adopting the minimum angle to be the leftmost finger, but some cases do not worked out especially if the hand was pointing to the right (horizontal hand direction), in this case the assumption has to be reversed, however, our new methodology uses dynamic origin of coordinates depending on input gesture, we have forms two vectors correspond to border fingers by subtracting finger base from fingertip of that border finger, after that we have calculated the sum of these two vector and according to its sign (x and y) the origin of coordinates has been decided perfectly, Fig. 7 shows the sign distribution for the final vector.

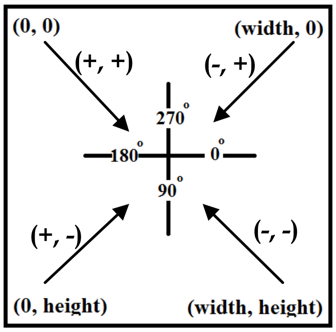


Fig. 7. Deciding the origin of coordinates.

As seen by Fig. 7, the arrow represents the direction of the vector with a label represents its signs, and the originated corner pixel is the origin of coordinates that will be used, the direction of angles is sketched in the middle of latter Fig., after calculating the angles of the border fingers in respect of the decided origin of coordinates, always the minimum angle is the leftmost finger and the maximum angle is the rightmost angle, in Fig. 5g, the origin of coordinates was (0, height) depending on the signs of the resulted vector, and the angles were 307°, and 337° respectively, according to our rule, the finger that corresponds to former angle is the leftmost finger since it has the minimum angle, however, this methodology has been applied on many gestures with different angles and always classifies correctly with no error.

6.5.3 Fingers Indexing

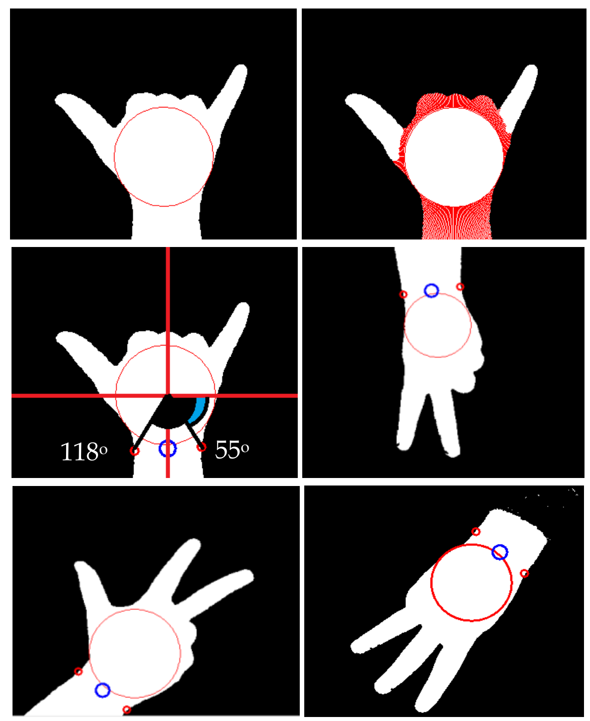
After correct discovering of the leftmost and rightmost fingers, and by using final neighborhood matrix, the indexing can be done, considering Fig. 5g, leftmost finger is 3 and right most finger is 4 (as calculated from previous steps), so, we have to trace the neighborhood matrix by starting at former finger and ending at latter finger, so, the sequence is:

$$\text{Sequence} = \{3 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 4\}$$

6.6 Wrist Detection

We have employed a novel approach for wrist detection so that the wrist can be detected regardless the orientation of the hand, this approach based on projection of the hand area, but, the projection is not like traditional projection it is a circular projection.

After deciding the fingers area as well as the palm center; we established a circle centered at the palm center with radius equal to the maximum radius that can be fit there; that means this circle touches the hand boundaries somewhere, after that we scan all the lines that emerged from the center of that circle onward to hand border and aggregates the number of white points (projection) as seen by Fig. 8b, the scanning goes around this circle and completes 360 ° so the projection covers all the area around this circle, Fig. 9 shows the histogram of that circular projection that corresponds to Fig. 8b in which the x-axis represents the angle and y-axis represents the value of the projection(number of white pixels on the line of that angle). As seen by Fig. 9, the peak point is the wrist area and that area is traced in order to decide the limits of the wrist area as shown in Fig. 8c, the threshold has been set to 20 which means that we consider the two angles that lie on the rotated histogram projection; these two angles are the starting and the ending of the wrist area and their average is the wrist center as seen in Fig. 9, for Fig. 8c that has Fig. 9 as its histogram, the starting angle is 55 ° and the end angle is 118° that represent the wrist boundaries as plotted in two small red circles in Fig. 8c.



a b
c d
e f

a: biggest circle deciding. b: rotated projection. c: locating the limits of the wrist as well as the wrist center. e: another sample with no sleeve. f: another sample with hand sleeve.

Fig. 8. Wrist locating.

6.7 Hand Parameters Detecting

The finger related parameters and palm related parameters can be detected by doing some calculation from the resulting data; these data can come from (7), (8), (9), and (10) as follows.

$$P_{center} = \sum_{j \in t_p} P(j)/t_p \tag{7}$$

$$F_i\,(tip) = \max_{j \in t_g} |P_{centre} - F_i(j)|\,,\ \forall\, i \in g \tag{8}$$

$$F_i\,(center) = \sum_{j \in t_g} F_i(j)/t_g\ ,\ \forall\, i \in g \tag{9}$$

$$F_i(base) = F_i(center) + \left(F_i(tip) - \widehat{F_i(center)}\right) * w,\quad \forall\, i \in g \tag{10}$$

ISSN 2229-5518

Where P is the palm and F is the finger, tp is the total number of pixels in palm p, g is number of groups, and Fi(j) is the pixel j of finger group i, and tg is the total pixels in group g, and $(\hat{B})$is the reflection operation, these formulas show the calculation of palm center, fingertips, finger centers and finger bases in (7), (8), (9), and (10) respectively in addition to the wrist center that we have already calculated hereinabove.

Note that in case of multi-hand object in a single image input, each group can be attached to specific palm by finding at least one palm pixel that touches this group as we mentioned before.

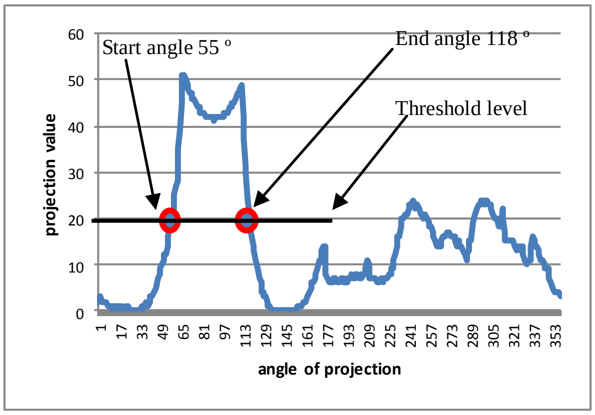


Fig. 9. Rotated histogram projection in which the peak point is the wrist area indication.

7 Hand Features Representation

After a correct calculation of the hand parameters, it is time to extract the features that will be used for finger classification as well as hand gesture recognition, we have adopted four kind of features in our algorithm that proved its robustness for correct classification, these features are grouped into two groups, collective group and selective group, each group has two features, however, the collective group has to be applied in full and depending on the outcome of that application one more feature should be chosen from the selective group, that means three features have to be applied for classification/ recognition, let us define the hand direction line (HDL) that is the line that represents the direction of the hand object and it calculated from wrist and palm centers, it worth to mention that Equation (7) has been applied for initial calculation of the palm center, and after calculation the wrist center we have calculated the final palm center that is the averaging of the palm border pixels as a resulted from rotated projection, however, the features can be listed below:

1. Perpendicular Casted Distance (PD): it is the distance between the casted finger's base on HDL and the palm center.
2. Base Angle (BA): it is the angle that existed between the line formed by finger's base to palm center and HDL, this feature along with PD have been used to distinguish thumb, middle and pinkie fingers since we are looking for robust algorithm that can recognize the presented hand without any prior assumption, so, index and ring fingers will have almost similar PD and BA for that reason we have designed the selective group.
3. Base(s) Angle (BsA): it is the same as BA but with one main different that is the replacing of HDL with the line formed by palm to finger's base.
4. Base Border (BB): let us simplify the definition, assume there is a line between the closest border pixel to the finger's base and a thresholding border pixel, this thresholding pixel is any quantity takes us to the sides of the hand border (46 for now came from two third of PCT diameter), now, BB is the distance between the perpendicular casting of a specific border pixel that has a maximum distance from that mentioned line and that closest finger's base, BsA and BB are designed especially for classifying index and ring fingers and used selectively according to some decision taking by applying collective features as we will see in section 8.2.

Fig. 10 presents a pictorial representation of our adopted features. As an explanation of Fig. 10a; the calculation of PD can be done from the distance of casting (a) on HDL to the (b), furthermore, (a) can take any finger's base, BA can be calculated from the angle that confined between the following sequence : (a) to (b) to opposite of (c), similarly, (a) is the base of any given finger, considering Fig. 10c; this feature is the angle that existed between this sequence: (a) to (b) to (c); it is clear that (a) can take just thumb or pinkie as we will discuss in section 8.2 and (c) can take other fingers as well as pinkie, the treatment of Fig. 10a is applied on Fig. 10d that is the distance between casted (a) and (b) on the line (b) to (c).

8 Classification-Recognition

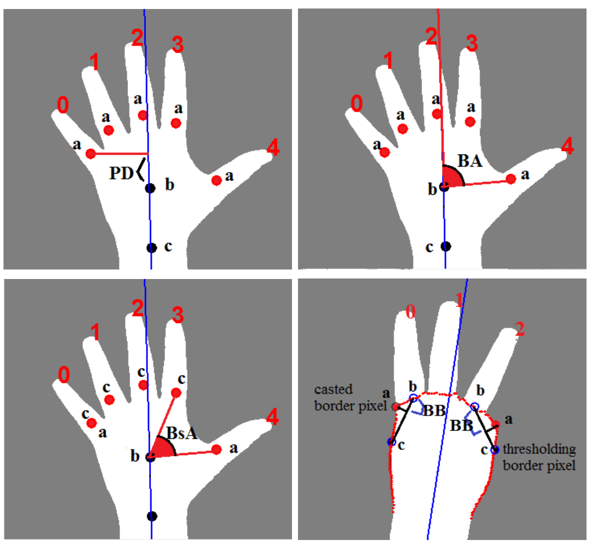
We have discriminated between two terms since we want to give better understanding on the kind of process have done, the classification related to distinguishing the finger name by attaching that finger to its class belongs since we have five predefined classes (thumb, index, middle, ring, and pinkie), recognition is a complementary process comes after the classification with an aim to recognize the shape of the final gestured hand which depends definitely on the classes that discovered for fingers.

8.1 Gaussian Classifier

Gaussian function has been adopted herein as a classifier, its purpose is to attach the extracted fingers to one of its predefined classes depending on the likelihood output for each finger according to the trained data, standard Gaussian has been employed herein since the maximum output of such function is one that represents the likelihood of specific finger belongs to certain class, the normal Gaussian can be also adopted but the maximum likelihood will be less than one which makes the likelihood output to be close to each other, however, Equation (11) shows the adopted function:

$$p_{i,j}(x) = e^{\frac{-(x-\mu_{i,j})^2}{2\sigma_{i,j}^2}} \tag{11}$$

Where i ϵ {1, 2, 3, 4} that represents the feature number (total four features as we mentioned) and j ϵ {1, 2, 3, 4, 5} that represents finger number, furthermore, μ and σ represent the corresponding mean and standard derivation, one more thing has to be mentioned herein, our features are separable and this is the main characteristic of features designing, Table 4, 5 and 6 show the μ and σ of PD, BA, and BB respectively in which the separability is explicit since the mean values are away from each other and standard derivation impacts on the full width at half maximum that means the size of covering area of probability.



a b
c d

a: PD feature. b: BA features. c: BsA feature. d: BB feature.

Fig. 10. Pictuorial representation of our adopted features.

TABLE 4
PD FEATURE

Finger	μ	σ
Thumb	14	10.9
Index	93	4.6
Middle	98	8.2
Ring	84	9.3
Pinkie	54	6.7

TABLE 5
BA FEATURE

Finger	μ	σ
Thumb	77	7
Index	18	4.9
Middle	2	4
Ring	22	6.3
Pinkie	51	5.5

TABLE 6
BB FEATURE

Finger	μ	σ
Index	16	3.36
Ring	31	4.44

As seen by these tables, the separability makes the discrimination of finger classes possible and gave us remarkable recognition percentages as we will see in experimental results.

8.2 Fingers Classification

After a perfect modeling of the extracted features, a classification should take place, according to our discipline of features grouping; we have applied the following steps for classification as follows:

1. Collective Group: the first step is to apply the collective group that has two features, this step extracts the thumb, middle, and pinkie fingers correctly without give any consideration for index and ring whatever their output were, see Fig. 11 and 12 how the statistical models correspond to their fingers can be discriminated explicitly and easily and imagine the discrimination resulted by combining these two models together.
2. Selective Group-BsA: depending on the output of Collective Group application, if thumb or pinkie has been found we shall proceed with first feature of this group that considers the thumb base as point (a) as seen by Fig. 10c and (c) point will take whichever finger available; this case is applied when thumb was there, if pinkie was there the same thing applied but point (a) will be the pinkie's base.
3. Selective Group-BB: in case of no thumb was there nor pinkie was there, we shall consider our last resort, this case covers 7 possibilities of the hand gesture (different combinations of availability of index, middle, and ring), we have mentioned in section 7 (point 4) that BB is obtained with a specific border point, this border point has the maximum perpendicular distance from the mentioned line in there, and should lie within the boundaries of finger's closest pixel and the thresholding pixel (see Fig. 10d, blue circles at both sides), the pictorial representation of the statistical models of these two fingers can be seen in Fig. 13.

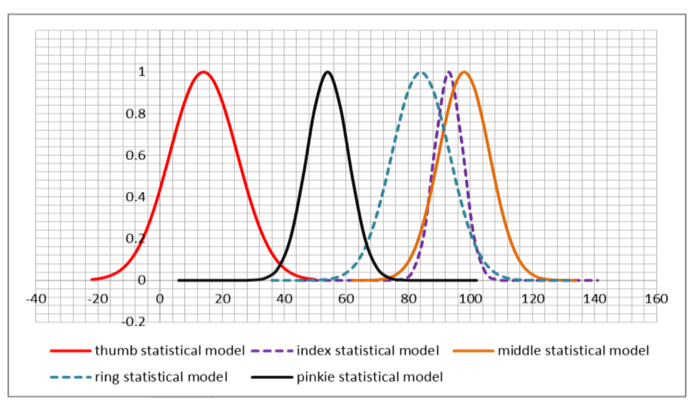


Fig. 11. Gaussian models for different fingers regarding PD feature.

ISSN 2229-5518

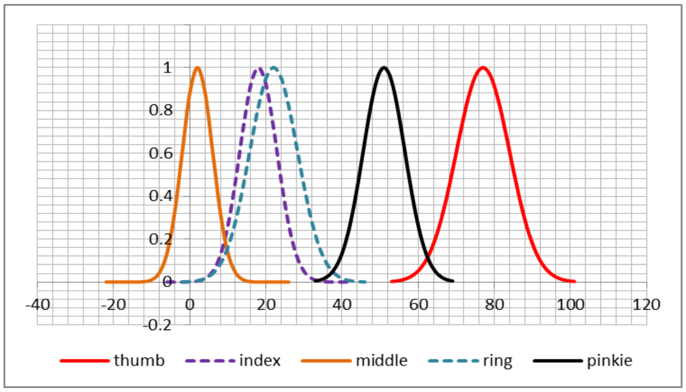


Fig. 12. Gaussian models for diffierent fingers regarding BA feature.

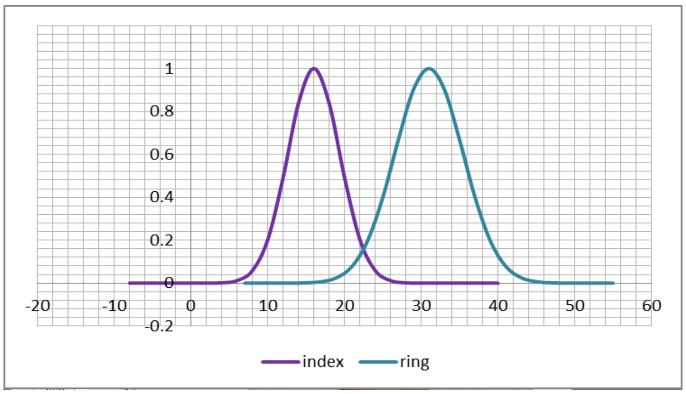


Fig. 13. Gaussian models for diffierent fingers regarding BB feature.

8.3 Hand Recognition

The hand recognition is applied after finger's classification, each finger has been assigned a fixed bit location in a 5-bits stream, Fig. 14 shows this assignment; after calculating the integer number represented by this stream we can get the index of the corresponding gesture immediately, other possibility is by building a DFA (deterministic finite automata) device for recognizing the hand gesture with 32 final nodes.

bit 4	bit 3	bit 2	bit 1	bit 0
pinkie	ring	middle	index	thumb

Fig. 14. Gesture database indexing.

9 Train and Test Sets

Considering the training set, we have trained our system with 26 different gestures, we have extracted the features associated with each finger and we have built the required model data for correct classification of testing process, we tried in training phase to choose some samples that cover the feature space; and also to be a representative for gesture database, we have toned these training data with statistical models for latter classification stage, the training set is arranged as follows:

Training Set= {(1, 10), (2, 5), (3, 5), (5, 6)}

Where (p, q) represents q different samples associated with a hand with p fingers; the statistical models have been instantiated from this set. We have tested our system using 100 different samples, total gestures are 32 of different combination of raising and non-raising fingers including non-finger's hand, some unrecognized gestures have been adjusted in scale for better recognition, we have used the following set as testing samples:

Testing Set= {(0, 2), (1, 17), (2, 25), (3, 24), (4, 24), (5, 8)}

10 Experimental Results

The major time spent is for hole filling that is ignored in many other papers like [15] and their total recognition time was 1.5 seconds per gesture, in case we adopt his segmentation, ours is 407 milliseconds (0.407 second), however, the hole filling time is 741 milliseconds (0.741 seconds) per gesture and in either cases we have achieved the minimum average time per gesture with or without hole filling application since the total with hole filling around 1148 milliseconds (1.148) seconds regardless of the recognition percentage that we have achieved that is 98.5%for finger-wise recognition and 96% for gesture-wise recognition as well as our implementation for hand gesture is no restriction implementation and can be with any angle unlike [15] that stipulated the up righted hand direction and right hand only, our approach can be implemented on any hand with no limitation, Fig. 15 shows the recognition percentages for each hand's finger count, we have applied PC i3 Intel CPU with 2.13 GHz processor speed and image size around 350x350 pixels, one more thing, the processing time will be declined dramatically if the image size reduced.

Now, Fig. 15 works as we have mentioned in Section 8.2 by locating the location of the thumb, middle and pinkie fingers by using collective group features and after that continues with selective group features according to the rule specified. Fig. 16 shows the recognition percentages of application of different combination of the collective group features. As seen there the finger-wise recognition has the highest recognition rate when both of PD and BA applied that is 98.5% and for gesture-wise recognition the application of PD got the highest score that is 97.8%.

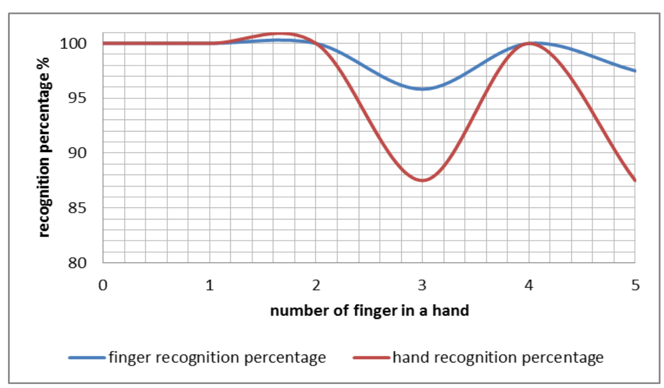


Fig. 15. Finger-wise and gesture-wise recognition.

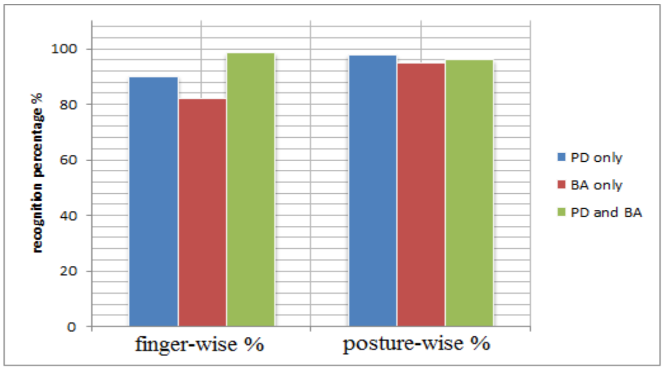


Fig. 16. Finger-wise and gesture-wise recognition.

10.1 Application Example

We have presented an example for understanding the likelihood and the distribution of these likelihoods over different fingers during the testing process, consider the following outcomes:

TABLE 7
PD FEATURES CORRESPOND TO FIG. 17A

Finger	Thumb	Index	Middle	Ring	Pinkie
0	0.90	0.00	0.00	0.00	0.00
1	0.00	0.00	0.04	0.30	0.05
2	0.00	0.22	0.27	0.99	0.00
3	0.00	0.00	0.03	0.70	0.00
4	0.01	0.00	0.00	0.00	0.36

TABLE 8
BA FEATURES CORRESPOND TO FIG. 17A

Finger	Thumb	Index	Middle	Ring	Pinkie
0	0.96	0.00	0.00	0.00	0.00
1	0.00	0.25	0.00	0.78	0.00
2	0.00	0.00	0.79	0.00	0.00
3	0.00	0.91	0.00	0.97	0.00
4	0.00	0.00	0.00	0.00	0.97

TABLE 9
PD AND BA FEATURES TOGETHER CORRESPONDS TO FIG. 17A

Finger	Thumb	Index	Middle	Ring	Pinkie
0	1.92	0.00	0.00	0.00	0.00
1	0.00	0.25	0.00	1.12	0.05
2	0.00	0.22	1.06	1.00	0.00
3	0.00	0.91	0.03	1.68	0.00
4	0.02	0.00	0.00	0.00	1.33

We care for Table 9 that has the addition of PD likelihood and BA likelihood together, we can read this table as follows: rows represent the finger indexing done by us and columns represent the system sequence for fingers, for finding the thumb for example, we have to see which finger has maximum likelihood regarding thumb columns, it seems finger 0 is the thumb, and the same for the rest fingers, Fig. 17a shows this hand, however, this Table gave us the correct decision about extracting the thumb, middle, and pinkie, by the way, the middle and pinkie will be neglected since by the presence of thumb and all other fingers will be attached new likelihood as seen by Table 10, for now, thumb is detected correctly and pinkie is also detected correctly as seen from the latter table but we care for thumb only since we will apply Fig. 10a and (a) point will be the thumb, in case of the hand does not has the thumb and a pinkie instead, we will apply the selective feature-pinkie based to discover other fingers, see Fig. 10c in which (a) will be the pinkie, in case of no thumb nor pinkie, we will apply selective features (BB feature) to discover the ring and index and considering the middle finger in this case will be located from Table 9.

TABLE 10
SELECTIVE FEATURE-THUMB BASED FOR FIG. 17A

Finger	Thumb	Index	Middle	Ring	Pinkie
0	-	-	-	-	-
1	-	0.11	0.00	0.00	0.00
2	-	0.00	0.46	0.00	0.00
3	-	0.00	0.00	0.75	0.00
4	-	0.00	0.00	0.00	0.99

As seen, all fingers have been classified correctly; Fig. 17 shows some gesture samples.

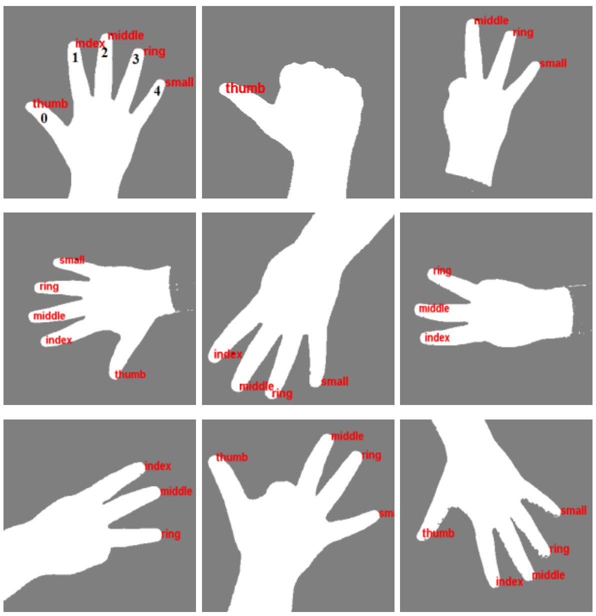
11 CONCLUSION AND FUTURE WORK

We have applied a novel approach for hand geometric features extraction and recognition using dynamic circle templates, we have achieved a multi-hand geometric features extraction that includes the locations of the palm and fingers as well, we have indexed the fingers from left finger to right finger whatever the hand orientation was, our algorithm can be used also for finger(s) and hand tracking as well.

The extracted parameters of our algorithm are palm center, wrist center as well as wrist boundaries, fingertips, centers, and bases, all these parameters can be used successfully for further analysis to extract different kind of features or can be used as a graph matching paradigm for gesture recognition system after constructing a graph comprising these features and all of these parameters can be extracted regardless the hand orientation, no matter with or without sleeve.

We have extracted four kinds of features for application of gesture recognition system and for correct raised and non-raised fingers classification as well as the hand object recognition, we have employed 32 different static gestures and we have achieved a remarkable classification/ recognition time as well as the recognition accuracy for finger-wise and gesture wise as well.

Our suggested four features proved their robustness for correct finger's classification regardless the hand orientation was, these features are grouped into two groups that are collective group and selective group, the collective group is applied completely that contained two kind of features which are PD and BA, after the correct implementation of that group we proceed further with one more feature out of selective group that has two features as well, this selected feature out of selective group is decided according to some output of the collective group application, three cases will emerge, if the hand object has a thumb we proceed with first feature of selective group that is BsA by considering thumb's base as reference point for calculating the features of other fingers, second case if there is no thumb but pinkie instead we proceed with calculation the BsA feature for other fingers by considering pinkie's base pixel as a reference pixel, if none of them existed we proceed with calculating the BB feature that designed especially for discrimination of index and ring fingers especially and the likelihood of middle finger can be taken from the application of collective group.



a b c
d e f
g h i

Fig. 17. Application of our algorithm on different samples with and without sleeve under different rotation angles.

Gaussian classifier has been employed for calculating the likelihood of finger's matching to a certain class out of five classes (thumb, index, middle, ring, and pinkie fingers), average recognition time is 0.407 second and average hole filling time is 0.741 second, overall processing time for single gesture is 1.148 seconds, 98.5%for finger-wise and 96% for gesture-wise as recognition percentages by using 350x350 average image size with 2.13 GHz CPU Intel i3, , and this time can be reduced dramatically by reducing the image size and removing the need for hole filling application.

The geometric features considered of major important for controlling the classification process since they represent live data, unlike the non-geometric features that are similar to blind data in which the pixels or group of pixels are gathering for processing as a wholesale data, so, we have prepared a framework for geometric features extraction as well as a finger and gesture recognition system.

ACKNOWLEDGMENT

The authors thank the editorial board for their effort toward the preparation of this final version of the paper and also thank the journal members for their assistance during the registration and preparation phase.

REFERENCES

[1] Qing Chen, Nicolas D. Georganas,and Emil M. Petriu, "Real-time Vision-based Hand Gesture Recognition Using Haar-like Features", Instrumentation and Measurement Technology Conference – IMTC 2007 Warsaw, Poland, May 1-3, 2007, doi: 10.1109/IMTC.2007.379068.

[2] Mokhtar M. Hasan, and Pramod K. Mishra, "HSV Brightness Factor Matching for Gesture Recognition System", International Journal of Image Processing (IJIP), vol. 4, no. 5, pp.456-467, 2010.

[3] Mokhtar M. Hasan, And Pramod K. Mishra, "Comparative Study for Construction of Gesture Recognition System", International Journal of Computer Science and Software Technology, vol. 4, no. 1, pp. 15-21, Jan.-Jun. 2011.

[4] Xingyan Li, "Vision Based Gesture Recognition System with High Accuracy", Department of Computer Science, The University of Tennessee, Knoxville, 2005.

[5] Agnes Just, Yann Rodriguez, Sebastien Marcel, "Hand Posture Classification and Recognition using the Modified Census Transform", In Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition (AFGR 2006), Southampton , UK, Apr. 2006, doi: 10.1109/FGR.2006.62.

[6] Kenji Oka, and Yoichi Sato , "Real-Time Fingertip Tracking and Gesture Recognition", IEEE Computer Graphics and Applications, vol. 22, no. 6, Nov. 2002, doi: 10.1109/MCG.2002.1046630.

[7] Mokhtar M. Hasan, Pramod K. Mishra, "Hand Gesture Modeling and Recognition using Geometric Features: A Review", Canadian Journal on Image Processing and Computer Vision, vol. 3, no. 1, pp.12-26, Canada, March 2012.

[8] Noor A. Ibraheem, Rafiqul Z. Khan, "Vision Based Gesture Recognition Using Neural Networks Approaches: A Review", International Journal of human Computer Interaction (IJHCI), vol. 3, no. 1, pp.1-14, Malaysia, Feb. 2012.

[9] Daeho Lee, and SeungGwan Lee, "Vision-Based Finger Action Recognition by Angle Detection and Contour Analysis", ETRI Journal, vol. 33, pp.415-422, Jun., 2011, doi: 10.4218/etrij.11.0110.0313.

[10] Kenji Oka, Yoichi Sato, and Hideki Koike, "Real-time Tracking of Multiple Fingertips and Gesture Recognition for Augmented Desk Interface Systems", IEEE International Conference on Automatic Face and Gesture Recognition, 2002, pp. 429-434, doi: 10.1109/AFGR.2002.1004191.

[11] Yikai Fang, Kongqiao Wang, Jian Cheng, and Hanqing Lu, "A Real-Time Hand Gesture Recognition Method", IEEE International Conference on Multimedia and Expo, 2007, pp. 995-998, doi: 10.1109/ICME.2007.4284820.

[12] Vassilis Athitsos, and Stan Sclaroff," Estimating 3D Hand Pose from a Cluttered Image", IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 432-9, 2003, doi: 10.1109/CVPR.2003.1211500.

[13] Mokhtar M. Hasan, And Pramod K. Mishra," Brightness Factor Matching For Gesture Recognition System Using Scaled Normalization", International Journal of Computer Science & Information

Technology (IJCSIT), vol. 3, no. 2, pp.35-46, Apr. 2011, 10.5121/ijcsit.2011.3203.

[14] Shahzad Malik, “Real-time Hand Tracking and Finger Tracking for Interaction”, Project Report, 2003.

[15] E. Stergiopoulou, N. Papamarkos , “Hand gesture recognition using a neural network shape fitting technique”, ELSEVIER Engineering Applications of Artificial Intelligence, vol. 22, pp. 1141–1158, 2009.

[16] Oxford Dictionary, 2011, doi: 10.1016/j.engappai.2009.03.008.

[17] Rafael C. Gonzalez, Richard E. Woods, “Digital Image Processing”, Pearson Prentice Hall, First Impression, Chapter 9, 2009.

[18] Gady Agam, Its'hak Dinstein,” Regulated morphological operations”, Pattern Recognition, vol. 32, pp. 947-971, 1999.

[19] Marc Van Droogenbroeck, and Hugues Talbot,” Fast Computation of Morphological Operations with Arbitrary Structuring Elements”, Belgacom New Developments, Belgium , 1996.

[20] Yee Yee Htun1, and Dr. Khaing Khaing Aye, “Fuzzy Mathematical Morphology approach in Image Processing”, World Academy of Science, Engineering and Technology, vol. 42, pp. 659-665, 2008.

[21] Elaine Cohen , Lavanya Sita Tekumalla , and Elaine Cohen,”A Hole-Filling Algorithm for Triangular Meshes”, School of Computing, University of Utah, USA, 2004.

[22] Mokhtar M. Hasan, and Pramod K. Mishra, “Improving Morphology Operation for 2D Hole Filling Algorithm”, International Journal of Image Processing, vol. 6, no. 1, pp.1-12, Feb. 2012.

[23] Mathias K¨olsch, and Matthew Turk, “Robust Hand Detection”, Department of Computer Science, University of California, Santa Barbara, CA, 2005.

[24] X. Li, “Gesture Recognition Based On Fuzzy C-Means Clustering Algorithm”, Department Of Computer Science The University Of Tennessee Knoxville, 2003.

[25] M.Swain and D.Ballard, “Indexing via Color Histograms”, International Conference of Computer Vision, vol. 7, pp. 390-393, Dec. 1990, Japan, doi: 10.1109/ICCV.1990.139558.

[26] J. Wachs, U. Kartoun, H. Stern, Y. Edan, “Real-Time Hand Gesture Telerobotic System using Fuzzy C-Means Clustering”, Department of Industrial Engineering and Management, Ben-Gurion University of the Negov, 1999.

[27] T. Yang, Y. Xu , “Hidden Markov Model For Gesture Recognition”, The Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania 15213, 1994.

[28] Jiong J. Phu, Yong H. Tay, “Computer Vision Based Hand Gesture Recognition Using Artificial Neural Network”, Faculty Of Information And Communication Technology, University Tunku Abdul Rahman (Utar), Malaysia, 2006.

[29] Heni B. Amor, S. Ikemoto, T. Minato, H. Ishiguro, “Learning Android Control Using Growing Neural Networks”, Department Of Adaptive Machine Systems Osaka University,Osaka, Japan, 2003.

[30] J. Triesch, C. Malsburg, “Robust Classification Of Hand Postures Against Complex Backgrounds”, IEEE Computer Society, Second International Conference On Automatic Face And Gesture Recognition, 1996, doi: 10.1109/AFGR.1996.557260.

[31] S. Marcel, O. Bernier, J. Viallet, D. Collobert, “Hand Gesture Recognition Using Input–Output Hidden Markov Models”, France Telecom Cnet 2 Avenue Pierre Marzin 22307 Lannion, France, 1999.

[32] The AF Research Laboratory, “Neural Networks, language and cognition”, Elsevier, Neural Networks vol. 22, pp. 247-257, 2009.

[33] H. Gunes, M. Piccardi, T. Jan, “Face and Body Gesture Recognition for a Vision-Based Multimodal Analyzer”, Computer Vision Research Group, University of Technology, Sydney (UTS), 2007.

[34] Y. Lu, S. Lu, F. Fotouhi, Y. Deng, Susan J. Brown, “A Fast Genetic K-Means Clustering Algorithm”, Wayne State University, Kansas State University Manhattan, USA, 2000.

[35] B. Heisele, P. Ho, T. Poggio, “Face Recognition with Support Vector Machines: Global versus Component-based Approach”, Massachusetts Institute of Technology Center for Biological and Computational Learning Cambridge, 2001.

[36] S. Sathiya Keerthi, O. Chapelle, D. DeCoste, “Building Support Vector Machines with Reduced Classifier, Complexity”, Journal of Machine Learning Research, vol. 8, pp. 1-22, 2006.