Hasan, J Comput Eng Inf Technol 2013, 2:2
http://dx.doi.org/10.4172/2324-9307.1000108



Journal of Computer Engineering & Information Technology

Research Article

A SCITECHNOL JOURNAL

New Rotation Invariance Features Based on Circle Partitioning

Mokhtar M Hasan[1*]

Abstract

Gesture system can be considered as the non-tongue language that enables the human to emphasize or clarify the importance of current speech when communicate with other human(s).This idea can be utilized to be the vision based language between the human and the human-made intelligent machines. In this paper, the researcher introduced a gesture system model for recognizing the hand posture, the main contribution of this paper is the presenting of a new feature extraction method based on circle division of hand posture, this new scheme gives some flexibility for posture rotation in some decided degree that depends on the number of applied circle portions which in turn reduces the number of gesture samples stored in the database which finally impacts the processing time. The researcher has applied the system with a very low number of training samples per posture that is just two samples and with a very remarkable recognition rate of 97.92 % using twelve circles of eight portions per circle when tested with eight samples per posture, and the achievable recognition time was less than 0.6 second.

Keywords

Gesture System; Feature Extraction; Euclidian Distance; Gesture Circle Partitioning; Recognition Algorithm

Introduction

The family gives the first lesson for their children, this lesson is how to communicate with the world that surrounds their children, and this lesson is formed by the means of gestures and the children replies back accordingly, gestures like "COME", "GO", "SIT DOWN", "GET UP" and "SHUT UP" are most used due to the crying nature of the babies.

Gestures are very important in our daily live for communication with other live objects; Recent study has revealed that the non-verbal communication occupied around 93 %, moreover, facial expression as well as body gesture can express motions [1] .

The traditional input devices like keyboard, mouse [2-4], keypad, light pen and trackball [3] limits the speed and naturalness of the entire system [2,3] and the new trend is to upgrade the system towards natural interface [5], this could help the robots as well by improving its visual skills which in turns enable the robots to manage more difficult and dynamic environment [6,7], this enable them to input the command and understand its meaning to be carried out correctly [8].

***Corresponding author:** Dr. Mokhtar M Hasan, Computer Science Department, College of Science for Women, Baghdad University, Iraq, Tel: +964-7716278174; E-mail: mmwaeli@gmail.com

Received: March 25, 2013 **Accepted:** July 07, 2013 **Published:** July 24, 2013

Those cumbersome devices such as the keyboard and mouse give annoying burden comparing with human nature communication; mistyping, out of mouse pad, and other limitations which restrict the intuitive interface, and also the glove based gesture adds a troublesome load that enforce the human to wear such sensors and connections, for that, the vision based technique considered to be a good alternative for overcome such limitations and smooth the relationship between the human and the intelligent devices made by him. Also the vision based represents an attractive alternative [9].

Some applications of the gesture system:

1. Vital role for children that cannot use utterance to express their demands.
2. Deaf people for communicating with other live objects through sign language [4,10].
3. As a secondary tool for the people with normal hearing ability with spoken words in order to attract more attention [11].
4. Very necessary between peoples with some distance in the same area.
5. Helicopter signaler, traffic police, and other professions are considered to be a gesture oriented careers.
6. In virtual reality environment [12].

A gesture can be triggered from any movement of the body especially the hand motion since can convey more information [13]. The hand gesture can provide a huge information especially the geometric features [14,15] as well as statistical ones; this information needs to be used in a correct and prefect manner. The techniques that used to exploit this information were simple in the past two decades, for now and as the advancing of the science, the information is used in a good manner to recognize the hand gesture under various circumstances and with different movements.

The recognition of static gesture considered very important [16] because it represents the key idea of recognition and can work with dynamic gestures as well; since the latter is a sequence of static gestures and actions can be triggered accordingly [17].

Many challenges facing this gesture system which can be classified as posture challenges and system challenges, the posture challenges can be summarized as any challenged involved in the posture such as rotation, scaling, translation, noise, illumination [18] and human skin color, the system challenges can be summarized as the response time for posture meaning, recognition accuracy should be acceptable and without ambiguous decisions, and should work under different ethnic groups [13,19].

Tracking is also can be achieved using gesture system since it is a multiple of static poses [20] that can combined logically to extract the tracked object, and this tracking process can be done as a prior to recognition process [21].

In this paper; the researcher has presented a new features that can be used to enhance and speed up the gesture system with a very and remarkable number of samples per posture, I have applied two

SciTechnol
International Publisher of Science, Technology and Medicine

All articles published in Journal of Computer Engineering & Information Technology are the property of SciTechnol, and is protected by copyright laws. Copyright © 2013, SciTechnol, All Rights Reserved.

doi:http://dx.doi.org/10.4172/2324-9307.1000108

samples per posture for training phase, I have used six different postures, and I have tested the system with an eight samples per posture and I have attained a noticeable recognition percentage as mentioned before, in this work, I have assumed that the hand posture is already extracted from a complex background since I have presented a feature extraction method in order to discover and explain this method in more details regardless of the hand extraction method which can be applied using any already-applied methods.

Related Work

Many researchers have applied different techniques for gesture system, template matching [22] was the simplest one, vector classifier [23] and fuzzy algorithm in conjunction with neural networks [24] have been applied as well to enhance the recognition percentage achieved by the final system, clustering technique [25] has some share for classifying the gesture system.

Agrawal and Chaudhuri [26] recognize the hand gesture by moment calculation, the hand gesture is tracked using the condensation algorithm, the output gesture is pre-processed for removing any possible noise by the filtering the data using the median filter and then average filtering of size 7, a feature vector called feature trajectory is built with size 6 for each of 8 gestures, these features are the moment values of zero, first, and second order moments, single feature trajectory is constructed per gesture to represent the gesture. The new presented gesture is treated the same and the minimum error is used to recognize this gesture.

The authors in [16] in this method applied static hand gesture recognition; they have applied the system with six different gestures represent the numbers (0-5), they have used the local variation for segmenting of hand region by stipulating that the distance between the regions should be greater than the distance within the same region, the minimum bounding circle has calculated to encircle the hand region and by using of morphological operations; the encircled hand is split up into two parts, fingers and palm and the ZMs & PZMs have calculated for each part, a distance measure is used with minimum outcome for recognizing the new presented gesture.

The authors in [27] applied a static gesture system with ASL gestures, they used normalized adapted RGB color model for segmentation process of hand gesture and then the hand boundary is detected using ordinary contour tracking algorithm, the extracted the features using the Peripheral Direction Contributively (PDC) with radial form division of posture to N regions by deciding a specific angle for division, 20 samples per gesture are used for training purpose and the same for testing, they used two recognition algorithms, Dynamic Programming Matching and Multilayer Perceptron Algorithm, the former achieved 98.8 % and the latter achieved 96.7% under best performance. As comparing with the rapid development of the technology, mobile learning [28] has been done using gesture system to enhance and simplify the interface for the majority of users.

I have applied this method using different number of circles for performance test and also to discover the best applicable number of circles, the next section describes this method.

Overall Approach

The gesture system is composed of several stages which can be packed into two main phases that are the features extraction phase and recognition phase, the features extraction phase is responsible for building the database of the system that contains the features of the whole gestures and their samples (different poses for same gesture to overcome gesture perturbation), those features should cover the entire corresponding postures and also should be indexed for recognition purposes, many algorithms for features extraction have been applied and the accuracy of the these recognition algorithm depends mainly on the correct extraction of a good features.

After the database has been built, the recognition algorithm must be trained using the extant extracted features, the purpose of the recognition algorithm during the training phase is to find a prominent, non-overlapped (or at minimum at possible) regions for each posture, so, the new tested posture can be easily attached to its closest region that interprets it, the posture samples in each region are collected together depending on some common factor(s) existed in their features extracted in the training phase which make them grouping in that region, Figure 1 shows an example of the classification algorithm task.

In latter figure; we can notice that there are three groups of gestures each with two samples, those three groups represent the feature space that form the entire system classification, upon presenting a new posture; the algorithm task should find the perfect matching of this posture among the space groups according to the extracted features or can be seen as a benchmarking for matching process. One important notice is that the rotation factor should be taken into consideration during this process to produce a unified feature regardless the posture perturbations; however, this parameter has been neglected by many researchers due to difficulty of modeling and compensated by providing multiple training patters which have a negative impact on the database size as well as processing speed.

Posture Mass Centralization

After the extraction of the hand posture from the background, the features must be calculated; I have applied an additional step before features calculation which reduces the number of samples per posture and overcome the problem of posture perturbation, this step is adopted in [29] as a part of producing a comparative study, this method is the Posture Mass Centralization which works as follows:

The mean of the image posture is calculated and the image is divided into four quarters centered with the calculated image center which is the hand center mass, after that the whole quarters are scaled independently and with different scaling ratios to create a new image, without the need to re-allocate the image center coordinates at (0,0).

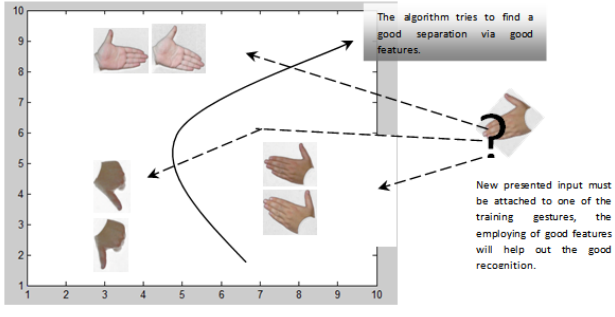


Figure 1: The behavior of the recognition algorithm for recognizing a new posture.

doi:http://dx.doi.org/10.4172/2324-9307.1000108

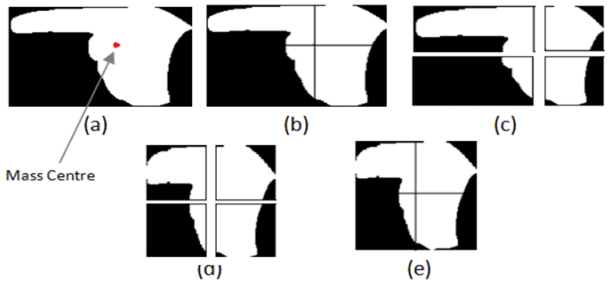


Figure 2: Application of Scaled Normalization. (a): the original input posture. (b): image division according to the mass centre. (c): image quarters. (d): equalize the four quarters. (e): image restoring.

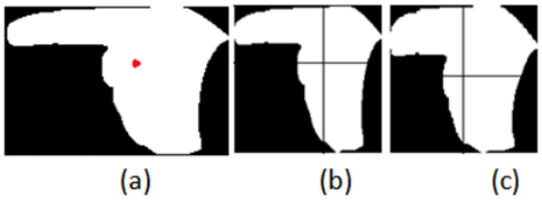


Figure 3: Comparison between the Ordinary and the Embodying Normalization. (a): the original input posture. (b): ordinary scaling which shifts the image centre. (c): scaling normalization which preserves the image centre.

In more details, in order to minimize the number of samples per posture in the database; we have to get rid of the perturbation transformation which is usually overcome by increasing the number of samples, my remedy for this problem is the scaled normalization, in this method, the center of the image is kept the same after the calculation of the original center for the image before scaling, and the scaling will be performed separately on each of the four image quarters with different ratios of scaling, so that, we will obtain a set of unified images with no overlapping in the quarters before and after scaling, the other already applied normalization method tries to unify the image sizes by converting them into an equal sized quarters and which leads to a formation of a new quarter features.

Figure 2 explains the scaled normalization which is useful to maintain the quarter features the same and can produce good features especially when applying with moments algorithms in which the center of the image is placed on (0,0) of the xy-coordinates.

As seen in the Figure 2, the center of gravity of the original image is kept the same after the normalization process which means that the scaling is done just at the quarters; the following Figure 3 shows the normalization operation that applied which is not preserve the center of the gravity.

Feature Division and Calculation

After application of the scaled normalization, now the image is ready for features extraction phase, the produced posture is divided into many circles depending of the wanted number of features which decides the recognition accuracy, I have applied the system with a different number of circles to examine and discover the best number of circles which means the best number of features are applicable in the system, this is the reason why I maintain the center location without any change by applying scaled normalization in order to unify all the circles with the same center which is the original image mass centre, Figure 4 shows an examples of an image posture divided into many circles.

As seen by the latter figure, the feature vector size was ten, this comes from the four circles with two portions which produce eight features, the remaining features which lies at the exterior area between the image border and first biggest circle border can be also used as a features because it is a part of the image posture and cannot be neglected, as another example to illustrates this division of image

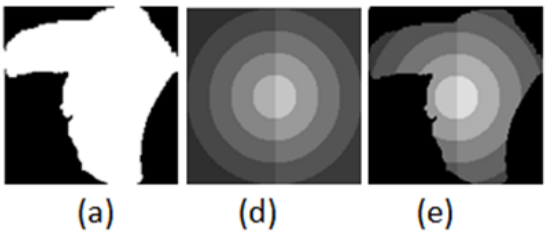


Figure 4: showing the feature extraction phase of input gesture with 4 circles divisions each of 2 portions. (a): the original posture image. (b): circle division with four circles, two portions per circle. (c): convolution of (a) and (b).

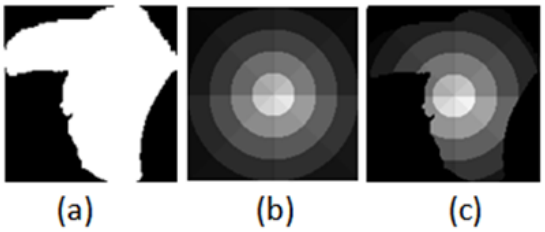


Figure 5: Features Extraction using 32 Portions. (a): the original posture image. (b): circle division with four circles, eight portions per circle. (c): convolution of (a) and (b).

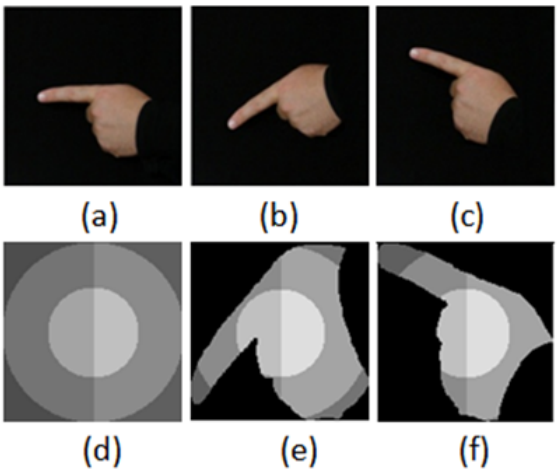


Figure 6: Rotation Invariance Example. (a): original posture. (b): downward rotation by 30 degree. (c): upward rotation by 20 degree. (d): circle features. (e): represents (b) after features classification. (f): represents (c) after features classification.

doi:http://dx.doi.org/10.4172/2324-9307.1000108



Figure 7: Training Set for One Posture.

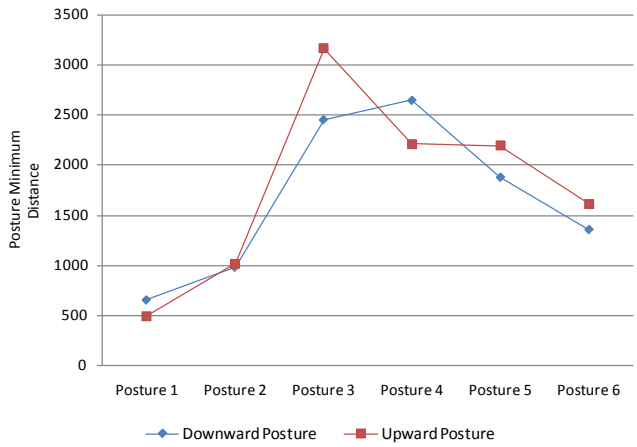


Figure 8: Classification Distance for the two Rotated Postures in Figure 7.

posture into feature vector can be found in Figure 5 which creates 32 features.

The calculation of Feature Vector Size (FVS) can be dependent upon two parameters, number of circles and number of portions per circle, this relation between these two parameters can be connected together in the Equation (1).

$$\text{FVS}=(\text{Number of Circles}+1)*\text{Parts Per Circle} \quad (1)$$

So, by using these feature division technique; we can overcome the problem of rotation, since, the suggested method allows some degree of rotation in the hand posture which falls in the same portion of the circle, Figure 6 explains this fact.

As seen by the this latter figure, the features produced by the two rotated hand postures are close and recognized correctly in spite of a significant degree of rotation of both postures (b) and (c) and in spite of the training postures were just two samples for this posture only along with other five different postures each with two training samples in the database, Figure 7 shows the training samples for this posture.

In order to clarify the advantages of the suggested features, I have made a sketch represents the distance of these two mentioned rotated hand postures in Figure 7 against all the database stored postures which are six different postures, and as told before, both rotated postures are classified correctly, Figure 8 shows this sketch which depicts that both rotated postures gained minimum error comparing with other database postures.

Feature calculation algorithm

The following algorithm calculates the feature vector of the input hand posture with NoC (Number of Circles) Circles and 8 portions per circle and the Edge parameter which represents the length of the hand posture, the size of the feature vector is calculated according to Equation (1).

Algorithm 1: Feature Vector Calculation

Input: Hand Posture Image I(x, y), NoC, Edge.

Output: Feature Vector V(n).

Method:

Step 1: [Initialization]

r = Edge/2, Distance = r / NoC, XC = YC = Edge/2,

FVS=(NoC + 1) * 8, Initialize all V(n) to zero, with all n=FVS.

Step 2: [radius array creation]

Create Radius_Array [s] with s circles radius and the distance between those circles radius is Distance parameter by using the following equation:

$$Radius_Array(i) = r - Distance * (i-1), foralli = 1toNoC$$

Step 3: [circle selection]

Iterate all white pixels of I(x, y) through step 6and attach each pixel to a

Specific Current Circle Number (CCN) with positive R radius by applying the following equations:

$$error = \sqrt[2]{(x-XC)^2 + (y-YC)^2}$$

$$R = Min\{error - Radius_Array(i)\}, foralli = 1toNoC$$

If no such R found, set R to zero, and i to NoC+1.

CCN = i which satisfies the above equation.

Step 4: [quarter selection]

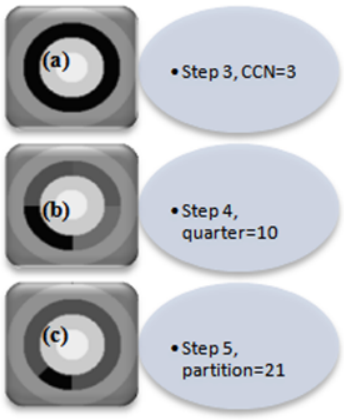


Figure 9: Seeking for the black area by selecting the circle, quarter, and portion respectively.

doi:http://dx.doi.org/10.4172/2324-9307.1000108

quarter = (CCN-1) * 4

If (x>XC) then quarter=quarter + 1

If (y>YC) then quarter=quarter + 2

Step 5: [partition selection]

Initialize partition to quarter * 2, since two portions per quarter

If (x, y) in the first quarter of the circle then

if (x, y) in the upper triangular then increment partition by one

If (x, y) in the second quarter of the circle then

if (x, y) in the upper triangular of the secondary diameter then

increment partition by one

If (x, y) in the third quarter of the circle then

if (x, y) in the upper triangular of the secondary diameter then

increment partition by one

If (x, y) in the fourth quarter of the circle then

if (x, y) in the upper triangular then increment partition by one

Step 6: [update feature vectorV(n)]

Increment V(partition) by one

Step 7: [output]

Output the Feature Vector V(n) with n=FVS, end.

As an illustration to the Step 3, 4 and 5 of Algorithm1, Figure 9 provides three reference images, (a) represents the circle selection step, (b) represents the quarter selection step by using the CCN parameter which is calculated in step3, and (c) represents the partition selection which multiplies the quarters by 2 since there are two partitions in each quarters, and also additional update which represents which partition inside the quarter should be.

Training Database

In this section, the training database that form the basic features via which all the new testing samples are matched is listed in Figure 10.



Figure 10: System database.

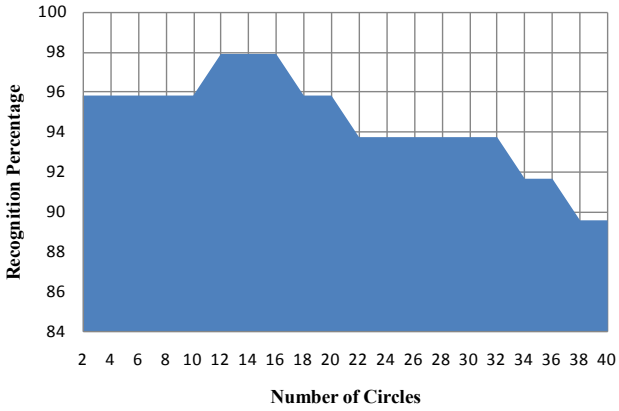


Figure 11: Recognition Percentages with Different NoC.

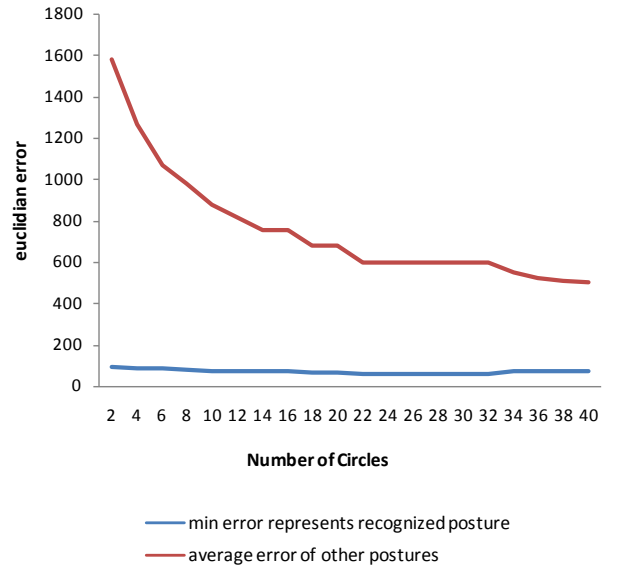


Figure 12: Recognition Percentages with Different NoC.

Experimental Results

I have tested the system with 48 different samples, 8 samples per posture out of six postures; furthermore, I undergone through all the different circle numbers by dividing the image posture into different number of circles at a time and testing the suggested method, Figure 11 show the recognition percentages depending on latter division.

I have discovered that the peak recognition points reside were at 14, 16, and 18 NoC, and this peak point has a value of 97.92 % which is very impressive in spite of minimizing the training set and the rotated samples include in testing set.

The extracted features can be distinguished easily by the recognition algorithm and the minimum error has a big gap comparing with next to value minimum error against the database training postures, this fact is illustrated in Figure 12 which depicts that the normal tested posture without a hard degree of rotation, this

doi:http://dx.doi.org/10.4172/2324-9307.1000108

sample is classified correctly with all adopted NoC and the minimum error does not exceed 93, while the other matching error reached 1600, this clear discrimination gives us a promising solution for vision based problems when adopting these kind of features.

The sample that presented in Figure 12 represents a normal sample as said, to accomplish the paper objective which is the production of rotation invariants features with a minimum number of training samples; I have tested the system again with a rotated sample and the minimum error maintains his distance from the average error as in Figure 13.

The two postures that are used in both of Figure 12 and Figure 13 respectively are shown in Figure 14; the reader can compare these two tested postures with the database postures in Figure 10 and imagine the system strength and classification robustness that imposed by the suggested method.

The following figure shows the behavior of recognizing two tested postures, the distance error that calculated for the recognized postures has a remarkable distance away from other database postures; and the error distance of the non-recognized posture is overlapped or too close with other database postures, as shown in Figure 15.

The recognized posture as noticed in latter figure has a minimum error distance of 72.43618 which was classified correctly, while the

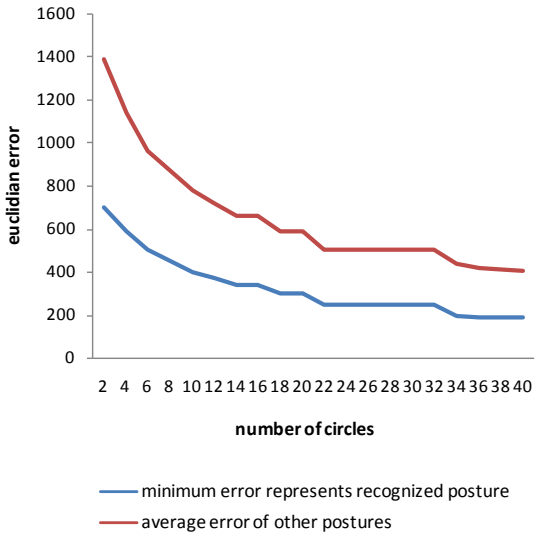


Figure 13: Recognition Percentages with Different NoC using rotated sample.

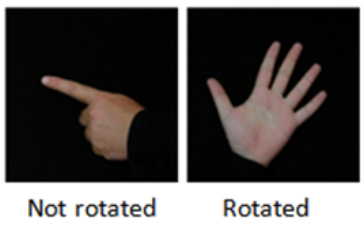


Figure 14: The two postures that are used for testing in latter two figures.

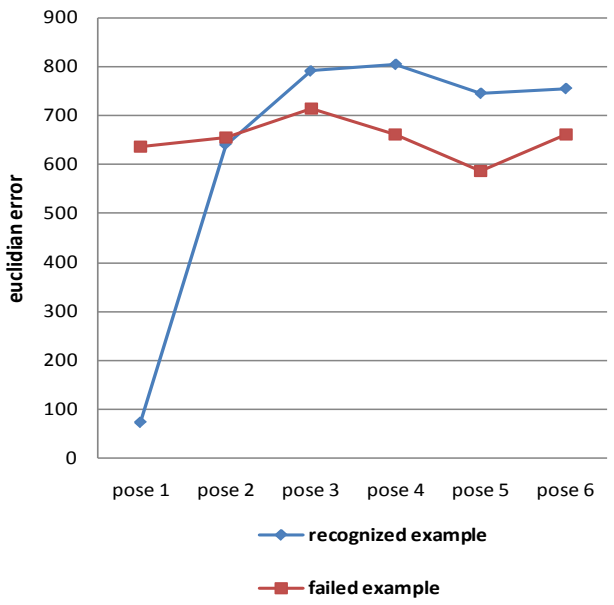


Figure 15: Two tested samples.

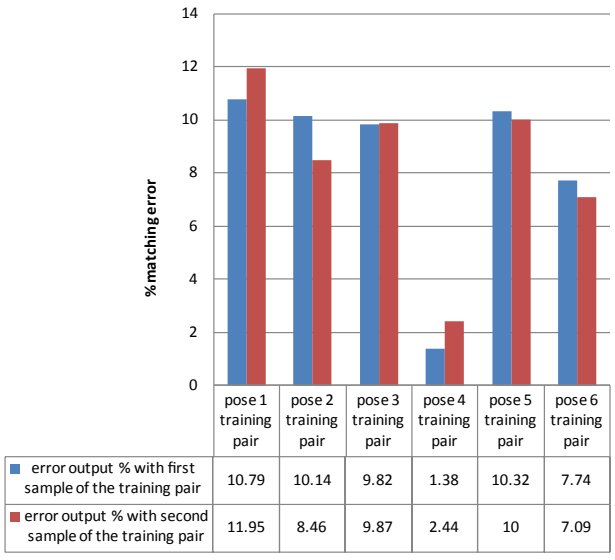


Figure 16: Matching score for one tested posture against all database postures.

matching score with other postures in the database are lied within (639.42865, 802.90845); in another hand, the non-recognized posture has a minimum distance of 586.9395 which refers to wrong database gesture; other database matching score are lied within (635.4337, 714.0714) which are closed to 586.9395.

Finally, I have presented in next Figure 16 the following fact which demonstrate the quality of the features and the separable property of these features, the NoC was 14 and the selected posture have recognized correctly as shown and was close to first pair of the pose 4 training pair with an error of 1.38 % out of the sum of errors of the database matching scores.

doi:http://dx.doi.org/10.4172/2324-9307.1000108

Table 1: Performance evaluation of the suggested algorithm as compared with other gesture algorithms.

Method name	Total gesture classes	Training samples as %	Testing samples as %	% recognition accuracy	Recognition time
[30]	6	N/A	N/A	80 %	2-4 seconds
[31]	7	N/A	N/A	90 %	2-5 seconds
[32]	6	50 %	50 %	71 %	N/A
[33]	8	25 %	75 %	94 %	N/A
Our Algorithm	6	20 %	80 %	97.92 %	0.6 seconds

Comparative Study

In order to unveil the powerful aspects of the proposed algorithm, a comparative study is presented in this section with some other gesture algorithms, the main parameters that can show the performance are the percentage of the training samples and testing samples as compared with the total number of samples used per single gesture, Table 1 shows this visually comparison along with the suggested algorithms.

As seen by latter table, the proposed algorithm achieved high recognition rate and low training set which is the main contribution of this paper.

Conclusion and Future Work

Gesture system is the second applicable language which used with deaf and people with normal hearing ability as well, the new trend which was started decades ago tries to bring this fact out with home appliances and human made machines to make the life easier and more convenient, therefore this trend kept developing continuously.

I have applied a new algorithm for deriving a new set of features which can be extracted from the input gesture, the reason why that is the human made machines can classify the gesture easily and more correctly, which is the circle division algorithm.

As noticed, I have applied different NoC and sketched a diagram shows the recognition percentages for each, as the number of circles increase, the recognition percentage increases as well, the peak point can be detected easily from Figure 11, and also the increasing of NoC reduces the calculated error and the distance error between the matched database gesture and the other non-matched database gestures is also expanded, I have examined the system with a new presented testing posture and NoC was 14 with 8 portions per circle and the error distance separation gained a significant ratio which is 1.38 % and 2.44 respectively as shown in Figure 16, this supports my theory.

I have achieved a maximum recognition of 97.92 % using just 12 training set of patterns composing of 6 different postures each of two samples which waste aim, I have reduced the number of samples per posture which; in turn; speed up the system, which is also proved by achieving a recognition time of less than 0.6 second per tested posture, and also the reducing the number of samples helps the system to include more different postures rather than reducing the number of postures to increase the number of samples per posture, so, the system will be able to recognize more different postures which enable the human-made machine for doing more tasks and different actions in each task.

As a future work, these features can be combined with any classification algorithm and also with complex background; in this paper I have presented this algorithm with a bare background since the objective was the demonstration of the new features.

References

1. GunesH, PiccardiM, Jan T (2007) Face and Body Gesture Recognition for a Vision-Based MultimodalAnalyzer. Computer Vision Research Group, University of Technology, Sydney (UTS).
2. Naidoo S, Omlin CW, Glaser M (1999) Vision-Based Static Hand Gesture Recognition Using Support Vector Machines. Department of Computer Science. University of the Western Cape, South Africa.
3. Symeonidis K (2000) Hand Gesture Recognition Using Neural Networks, Master Thesis. School Of Electronic And Electrical Engineering.
4. Triesch J, Malsburg C (1996) Robust Classification Of Hand Postures Against Complex Backgrounds, IEEE Computer Society, Second International Conference On Automatic Face And Gesture Recognition.
5. Freeman WT, Roth M (1994) Orientation Histograms For Hand Gesture Recognition, Mitsubishi Electric Research Laboratories, Cambridge, USA.
6. Swain M, Ballard D (1991) Indexing via Color Histograms. Intern. Journal of Computer Vision 7: 11-332.
7. Wachs J, Kartoun U, Stern H, EdanY (1999) Real-Time Hand Gesture Telerobotic System using Fuzzy C-Means Clustering. Department of Industrial Engineering and Management, Ben-Gurion University of the Negov.
8. Yang T, Xu Y (1994) Hidden Markov Model For Gesture Recognition. The Robotics Institute Carnegie Mellon University Pittsburgh, Pennsylvania.
9. Murthy GRS, Jadon RS (2009) Review of Vision Based Hand Gestures Recognition. IJKMIT 2: 405-410.
10. Ghotkar A S, Khatal R, Khupase S, Asati S. Hadap M (2012), Hand Gesture Recognition for Indian Sign Language. IEEE International Conference on Computer Communication and Informatics:1-4.
11. Shin JH, Lee JS, Kil SK, Shen DF, Ryu JG, et al. (2006) Hand Region Extraction and Gesture Recognition using Entropy Analysis. International Journal of Computer Science and Network Security 6: 216-222.
12. Cameron C R, DiValentin L W, Manaktala R, McElhaney A C, Nostrand C H, Quinlan O J, Sharpe L N, Slagle A C, Wood C D, Zheng Y Y,Gerling G J (2011), Hand Tracking and Visualization in a Virtual Reality Simulation. IEEE Conference of Systems and Information Engineering Design Symposium : 127 – 132.
13. Hasan MM, Misra PK (2010) HSV Brightness Factor Matching for Gesture Recognition System. IJIP 4: 456-467.
14. Brunelli R, Poggio T (1993) Face Recognition: Features Versus Templates, IEEE Transactions On Pattern Analysis And Machine Intelligence, Vol. 15.
15. Hasan M M, and Mishra P K (2012) Hand Gesture Modeling and Recognition using Geometric Features: A Review. Canadian Journal on Image Processing and Computer Vision 3(1): 12-26.
16. Chang CC, Chen JJ, Tai WK, Han CC (2006) New Approach for Static Gesture Recognition. J INF SCI ENG 22: 1047-1057.
17. Lee D., Lee S (2011), Vision-Based Finger Action Recognition by Angle Detection and Contour Analysis,, ETRI Journal 33(3): pp. 415-422,doi: 10.4218/ETRIJ.11.0110.0313.
18. HeiseleB, HoP, Poggio T (2001) Face Recognition with Support Vector Machines: Global versus Component-based Approach. Massachusetts Institute of TechnologyCenter for Biological and Computational LearningCambridge.
19. Hasan MM, MisraPK (2010) Robust Gesture Recognition using Euclidian Distance. IEEE International Conference on Computer and Computational Intelligence 3: 38-46, China.

20. Marcel S,Bernier O, Viallet J, Collobert D (1999) Hand Gesture Recognition Using Input–Output Hidden Markov Models. France Telecom Cnet2 Avenue Pierre Marzin 22307 Lannion, France.
21. Phu JJ, Tay YH (2006) Computer Vision Based Hand Gesture Recognition Using Artificial Neural Network. Faculty of Information and Communication Technology, University Tunku Abdul Rahman, Malaysia.
22. Jain AK, Duin RPW, Mao J (2000) Statistical Pattern Recognition: A Review, IEEE Transactions on Patterns Analysis and Machine Intelligence 22: 4-35.
23. SathiyaKS, ChapelleO, DeCoste D (2006) Building Support Vector Machines with Reduced Classifier, Complexity. Journal of Machine Learning Research 8: 1-22.
24. Hyun KJ, Wan RY, Hoon SJ, Seok HK (2006) Performance Evaluation of a Hand Gesture Recognition System Using Fuzzy Algorithm and Neural Network for Post PC Platform. Springer-Verlag Berlin Heidelberg, pp. 129 – 138.
25. LewYP, RamliAR, KoaySY, AliA, Prakash V (2002) A Hand Segmentation Scheme using Clustering Technique in Homogeneous Background.Student Conference on Research and Development Proceedings, Shad Alam, Malaysia.
26. Agrawal T, Chaudhuri S (2003) Gesture Recognition Using Position and Appearance Features. In Proceedings of, Catalonia, Spain 3: 109-112.
27. Simei G Wysoski, Marcus V Lamar, Susumu Kuroyanagi, Akira Iwata (2002) A Rotation Invariant Approach On Static-Gesture Recognition using Boundary Histograms and Neural Networks. In Proceedings of the 9th International Conference on Neural Information Processing (ICONIP) 4: 2137-2141, Singapore.
28. Sharma N, and Sharma H, HIM: Hand Gesture Recognition in Mobile-Learning. International Journal of Computer Applications 44 (16): 33-37, doi: 10.5120/6349-8695.
29. Mokhtar M. Hasan, Pramod K Mishra (2011) Brightness Factor Matching for Gesture Recognition System using Scaled Normalization. IJCSIT 3: 35-46.
30. Li X (2003) Gesture Recognition Based on Fuzzy C-Means Clustering Algorithm. Department of Computer Science, University of Tennessee, Knoxville.
31. Li X (2005) Vision Based Gesture Recognition System with High Accuracy. Department of Computer Science, University of Tennessee, Knoxville.
32. Stephan JJ, Khudayer S, (2010) Gesture Recognition for Human-Computer Interaction (HCI). International Journal of Advancements in Computing Technology, vol 2.
33. Bailador G, Roggen D, Troster G (2007) Real time gesture recognition using Continuous Time Recurrent Neural Networks. Proceedings of the ICST 2nd international conference on Body Area Networks, Belgium.

Author Affiliation

Top

[1]*Computer Science Department, University of Baghdad, Iraq*

Submit your next manuscript and get advantages of SciTechnol submissions

- 50 Journals
- 21 Day rapid review process
- 1000 Editorial team
- 2 Million readers
- More than 5000 facebook fans
- Publication immediately after acceptance
- Quality and quick editorial, review processing

Submit your next manuscript at ● www.scitechnol.com/submission