Features Fitting using Multivariate Gaussian Distribution for Hand Gesture Recognition

Mokhar M. Hasan\textsuperscript{#1}, Pramod K. Mishra\textsuperscript{#2}

\textsuperscript{#1, #2} Computer Science Department, Faculty of Science, Banaras Hindu University
Varanasi 221-005, Uttar Pradesh, India
\textsuperscript{#1}mmwaeli@gmail.com
\textsuperscript{#2}pkmisra@gmail.com

\textbf{Abstract}—Gesturing and posturing are two common tools used by the human to support his oral language during communicating with others which helped and proved its ability to deliver the message easily and correctly especially when those were on some sight distance, in this paper we have implemented a novel approach for providing such intuitive interface but this time will be used between human and human-made machine for human computer interaction purposes that helps the hearing impaired people as well, our approach based on feature distribution using multivariate Gaussian distribution function for finding a permanent remedy for translation, scaling, as well as rotation as one pack, we have focussed mainly on the rotation perturbation which solved normally by providing tenfold, hundredfold and even sometimes thousand fold of training gestures in training phase, we have headed a different direction by focusing on lowering this simmering number of training patterns and producing a unified set of features for modelling the hand class, these features are controlled by the direction of hand object that is extracted using our direction analysis algorithm, central moments are used herein for feature vector representation and static gesture recognition that proves its robustness, we have achieved a remarkable recognition rates especially with few number of training samples which is our aim of this study with a recognition time of 0.794 second.

\textbf{Keywords}—Multivariate Gaussian, Bivariate Gaussian, Central Moments, Feature Vector Extraction, Direction Analysis, Main Principal Components, Gesture System.

1. Introduction

The second language besides the oral language is the gesture language, this language is used as a primary language like sign language for deaf people to communicating with others, however, the hearing people use this language in their daily life for every detail even when they are sitting alone and thinking in some subjects.

The sign language is applied by many researchers [1][2][3][4][5][6][7] to imitate the human ability of recognizing the message concealed within the motion/view of the hand, American Sign Language (ASL) was implemented [1] for this reason, British Sign Language (BSL) was considered as well [8], Pakistan Sign Language [9], Japanese Sign Language [10], this study is expanded to include even the traffic sign language [11] for automation the meaning of these signs laid over the road.

Generally, hand gesture can be classified into a glove based approach [13][14][15] and appearance based [13][14][15][16], glove based considered of cumbered devices [17] that hinders the level of naturalness due to the wires attached to glove, and also these gloves should fit the human hand and we have to provide special size for different hands in different ages which adds additional difficulty for already exited drawbacks, and it is worth to mention that the gloves may transfer some skin diseases [18] especially in game parlours when everyone has to wear gloves for gaming; on the other hand, the mouse and keyboard which also considered as boring interfaces; and could transfer the diseases as well since its belong to the class of touchable devices [19], for all of these aspects; vision based is preferred and has an edge over them in which the communication or message delivering is applied with no distance limitations nor wearable stuff should be considered.

Many factors that improved speed and accuracy of gesture system due to the exponential growing of computational power [19] and the ability of processing a huge data set [19]; besides elaborate and diverse methods which growth rapidly [19] and the upgrade [17] and development of such methods for enabling new feature extraction as well as classification algorithms that can employ the advantages of the development of pattern recognition algorithms [17], machine learning algorithms [17].

The gesture system can be built by gathering computer vision [21], image processing [21] and artificial intelligence [21] all together to achieve one common goal which is how to imitate the human capabilities of understanding the meaning of human hand gestures which is done very easily by human
eye-brain system.

Many challenging have been addressed for successful implementation of gesture recognition system, these challenges are classified by [16] as posture challenges and system challenges, the former challenges are those have impact on the correct feature vector extraction and retaining in the system, system challenges are those that have impact on the system performance and recognition as well as the classification accuracy.

The solution of these challenges is addresses by [22] to be one of two approaches: train-based solution (TnBS) and test-based solution (TtBS), by the means of TnBS; an intensive training set should be provided to dismiss these challenges mentioned hereinabove, while a fewer training set could be employed in TtBS in which the system undertakes the recovering of these challenges. It is worth to highlight that most researchers embody TnBS for their systems, for example; 120 different samples were employed in [23] for covering of 12 different gestures, 40 samples is provided for each of 7 different gestures in [24] which means 280 different samples for training set, in [14]; four different postures with 450 samples per posture are taken into consideration which raise the total to 1800 samples plus to 500 additional samples with no hand posture; levied is 2300 samples, in [25] ; 46 gestures each with 100 samples are employed which raise the total to 4600 samples, this increasing in the number of samples can be reduced if we consider TtBS as employed herein.

In this paper, we have presented a novel approach for application of TtBS for developing a robust classification algorithm that achieves a high recognition rates using minimum number of training set in each class that used for instance the gesture model via perfect fitting of the hand gesture features using multivariate Gaussian distribution function dominated by the hand direction, these fitted features are used to extract the feature vector using geometric central moments, it is worth to mention that multivariate is called bivariate in case of two coordinates applied.

2. System Overview

Our gesture system consists of many stages, some of them are concealed inside others; Figure (1) highlights the important stages that have been employed herein.

3. Calculating of Hand Direction Parameters

The calculation of the hand object direction parameters plays a crucial role for implementing the pending steps, these parameters are hand centroid ($\mu_x, \mu_y$), and the angle of the hand slope formed by $\theta$, the former parameter can be calculated using Eq. (1), but, the latter parameter needs some attention since hand slope needs taking into consideration all the hand pixels into a single algorithm to get this parameter, one approach is PCA (principal component analysis) which requires intensive calculation, matrix and vector operations to get this parameter by the help of eigenvalues and eigenvectors, however, we have developed our own algorithm for object slope calculation and data trend as well, this algorithm called Direction Analysis Algorithm (DAA) [22] which based on finding the relationship between the variance and covariance statistical parameters calculated from given object/data, for more details please refer to [22], Figure (2) shows some statistical data using the this algorithm in which their data were: (194, 198), and (193, 205) represent the centroids for each of Figure (2a) and Figure (2c) respectively, $342^\circ$, $58^\circ$ are their $\Theta$ respectively as well.

4. Hand Object Locating

After successful calculating of the hand direction parameter as well as the center of mass parameter, the next step is to locate the borders of the hand object and confines this object, the normal method for achieving this is by scanning the left, right, top and bottom all together and heading towards the centre of the image for achieving the borders of the object that lie inside the image as in [21][26][27], this method consumes much computational power and time; and neglects the rotation factor of the hand object since we are trying to find a permanent solution especially for rotation perturbation, however, we have applied a different approach for locating of the hand gesture by tracing the main and secondary principals in two steps for each principal, the first step is to go directly from the image border to the first hand foreground pixel which is a line tracing and you can imagine how fast it is since it is a one pixel motion and tracing, and the second step is going back to cover the hand completely in this direction by constructing a perpendicular line across that traced line, Figure (3) shows this process.

As noticed in Figure (3), the perpendicular line can be constructed by reversing the dy and dx order in the line equation with negating the resulted term, and the line generation used was bresenham line drawing algorithm; and the processing time were 36 msec and 32 msec for locating and confining hand object in traditional method and were 11 msec and 6 msec by adopting our technique for Figure (2a) and Figure (2c) respectively which means faster processing time and orientation-preserve normalization.
5. Deciding the Centre of the Bordered Area

Now, we have to find out the center of that bordered area, we can consider the following Figure (4). In case of Figure (4a), the center can be calculated easily by application of Equation (1), but, however, Figure (4b) needs another approach which is by intersecting of each two adjacent lines, we have derived the equation for finding the intersection between two lines by considering those two lines are perpendicular for each other as Equation (2) and (3), where input \((x_1, y_1)\) is the border point that calculated from section 4, the output is the centered red circle as seen by Figure (4c).

\[
\frac{X + X'}{2} = m \begin{pmatrix} x_1 \ y_1 \end{pmatrix} + \begin{pmatrix} x_2 \ y_2 \end{pmatrix}
\]

(1)

\[
x = \frac{m^2 x_1 - m y_1 + x_2 + m y_2}{m^2 + 1}
\]

(2)

\[
y = m (x - x_1) + y_1
\]

(3)

We need to find out the distance between each two opposite points in each of the main and secondary principal components, which are called main principal distance (MPD) and secondary principal distance (SPD) respectively, for Figure (4c), MPD is the distance between two red points on the solid gray line and SPD is the distance between the red points on the dotted gray line.

6. Gaussian Multivariate Dist. Function

We have employed a bivariate Gaussian distribution since we are working on xy-plane coordinates and we have used the normal Gaussian since we are looking for a peak point in the middle of the Gaussian which is 1, our formula can be written as in Eq. (4):

\[
f(x_1, x_2) = k \exp \left(-\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right)
\]

(4)

Where \((x_1, x_2) = (x, y)\), \(\mu = (\mu_x, \mu_y)\), \(\Sigma^{-1}\) is the inverse of the covariance matrix, and \(k=1\), furthermore, Figure (5) shows a sketching for the Gaussian bivariate distribution for some hand samples.

7. Selecting the Likelihood Width

As known, the Gaussian likelihood for any input falls in the range \((0, 1]\), our aim is to divide these probabilities into a number of circles that called a feature terrace areas, each feature terrace area has a specific likelihood width out of the Gaussian likelihood output, an eleven different feature terrace
areas unified at Gaussian center can be selected, the reason behind this 11 feature terrace areas is taken from our study for circle features division to decide the perfect number of circles that the hand can be, we have done a study for a different number of circles and 11 circles had the maximum recognition percentage, so, we have divided each 0.1 probability of the Gaussian output to be a single terrace as seen by 3D plotting of these terrace areas in Figure (6).

8. Calculating Additional Terrace

As we mentioned that we need 11 terrace areas, 0.1 likelihood divisions gives us 10 terrace areas; one more terrace area needed, this is done by finding the distance between the pixels that lies on the edge of the terrace areas that have a maximum distance away from Gaussian center. After calculating these distances, we built a non-linear regression model for fitting these data by using half of these distances and the missing terrace area can be found easily by spotting the likelihood of the next estimated distance; this spotted likelihood will be the border of the additional terrace area as noted in Figure (7).

9. Adjustment Gaussian Parameters

To ensure that the bivariate Gaussian Distribution function covers the hand object especially after calculating the new terrace area since the Gaussian area should covers most parts of the hand object; we shall adjust the $\Sigma_{xx}$ and $\Sigma_{yy}$ as a final step before proceeding with the rest steps, we just need to make sure that the main principal distance of the Gaussian function (GMPD) adjusted with the MPD mentioned hereinafter, same for the Gaussian secondary principal distance (GSPD) should adjusted SPD as well, the following Figure (8) shows these two parameters in case of the border likelihood was (0.1).
As an example of the above adjustment, we can consider Figure (9), in which the data for this input hand is listed in Table (1).

![Figure 6. Feature terrace areas for 0.1 likelihood division](image1)

Figure 6. Feature terrace areas for 0.1 likelihood division

After a perfect modeling of the hand gesture within a bivariate Gaussian distribution, features of this hand gesture should be extracted, first of all, we have to show how the number of terrace areas as well as the number of subdivisions reflected on the hand object. Figure (10) shows the division of a single terrace area into 8 subdivisions and whole Gaussian area divisions into 88 feature areas.

As seen in Figure (10), we have a total of 10 Gaussian terraces after neglecting the inner most terrace area which is one pixel area as seen by Figure (6), plus one terrace area which is the exterior area cover the rest of the image, aggregates to 11 terrace areas, each terrace area is divided into 8 subdivisions called feature areas, so, we got 88 feature areas.

![Figure 7. Full example for fitting the bivariate Gaussian distribution function](image2)

Figure 7. Full example for fitting the bivariate Gaussian distribution function

10. Feature Area Division

![Figure 8. 3D and 2D plots showing the GMPD and GSPD parameters](image3)

Figure 8. 3D and 2D plots showing the GMPD and GSPD parameters

![Figure 9. Adjustment of Gaussian bivariate parameters](image4)

Figure 9. Adjustment of Gaussian bivariate parameters

a: input hand object, b: bivariate Gaussian function, c: eleven terrace area with 0.1 likelihood distance, d: estimation of the originated terrace area.
Table 1. Practical example showing the adjustment in the Gaussian parameters for Figure (9).

<table>
<thead>
<tr>
<th></th>
<th>MPD</th>
<th>SPD</th>
<th>GMPD</th>
<th>GSPD</th>
</tr>
</thead>
<tbody>
<tr>
<td>old $\sum_{xx}$</td>
<td>236</td>
<td>194</td>
<td>316</td>
<td>192</td>
</tr>
<tr>
<td>old $\sum_{yy}$</td>
<td></td>
<td></td>
<td>new $\sum_{xx}$</td>
<td>2702.797</td>
</tr>
<tr>
<td>old $\sum_{xy}$</td>
<td></td>
<td></td>
<td>new $\sum_{yy}$</td>
<td>1362.041</td>
</tr>
<tr>
<td></td>
<td>3619</td>
<td>1348</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**11. Geometric Central Moments**

Two moments have been taken for feature vector representations which are the first and second moments, the size of the local moments will be 88*2=176 moments, and the global moments will be two moments only, so, the total number of moments levied to 176+2=178 total moments, Eq. (5) shows the calculation of the central moments.

$$
\mu_{pp} = \Sigma_x \Sigma_y (x - \mu_x)^p (y - \mu_y)^p f(x, y) 
$$

(5)

As seen by Eq. (5); we have employed two geometric central moments which are $\mu_{00}$ and $\mu_{11}$ which are enough for our feature representation since these two moments can represent the shape of the given feature area as well as our feature division invariant to translation, scale and rotation, we can shorthand the local features as area geometric central moments (AGCM), and the global features as object geometric central moments (OGCM), so, Eq.s (6) and (7) show the calculating of each of AGCM and OGCM respectively where $(\mu_x^{(k)}, \mu_y^{(k)})$ are the local means for each feature area $(k)$, and $(x^{(k)}, y^{(k)})$ are the $(x, y)$ pixels belongs to feature area $k$ and $(\mu_x, \mu_y) = (4, 5)$ which is the mean values of (number of divisions, number of terraces), i.e.; 4 is the number of divisions and 5 is the number of terraces.

$$
\mu_{pp}^{(k)} = \sum_y \sum_x (x^{(k)} - \mu_x^{(k)})^p (y^{(k)} - \mu_y^{(k)})^p f(x^{(k)}, y^{(k)})
$$

$\forall k = 1, 2, 3, ..., 88 & \forall p = 0, 1$  

(6)

$$
\mu_{pp} = \sum_{y=1}^{11} (x - \mu_x)^p (y - \mu_y)^p \text{Intensity}(x, y)
$$

$\forall p = 0, 1$  

(7)

Where $\text{Intensity} (X, Y) = \sum_y \sum_x f(x^{(k)}, y^{(k)})$  

(8)

such that $k = (Y - 1) \times 8 + X$

In latter equation, $k$ is the resulted feature area number.

**12. Experimental Results**

We have tested our system with 60 samples as a total, 6 gestures each with 10 samples, we have gradually increased the number of the training set on the account of testing set so that the performance of the system will be discovered, Figure (11) shows the recognition percentages after adopting different number of training samples, starting from one training sample to 9 samples gradually, we have employed Euclidian distance for feature classification.

**13. Conclusion**

We have implemented a novel algorithm for gesture recognition system based on bivariate Gaussian distribution and the fitting of the input hand gesture has been done using this distribution, we have developed a robust algorithm for curing the translation, scaling, and rotation problem, translation and scaling has been solved by many researchers by shifting the hand object and fixing the hand for a
specific size and scale, but the rotation perturbation has been addressed by providing huge number of samples for model instancing which cause slowing down the recognition process since a lot of features should be taken into consideration during the classification process as well as the huge database size required for retaining all trained features, herein, we have introduced translation, scaling, as well as rotation invariant algorithm that we have applied with only one training sample and we have got 90% recognition percentage, when this number raised to 5 samples per gesture; the recognition percentage raised to 100% as seen by Figure (11) hereinabove.

We have extracted the features using geometric central moments in two kind of features that are local and global features, the local features is the features that are calculated from each feature area which produce a total of 88 values each with 2 moments, and global features are calculated from the converting the feature areas into a 2D matrix and applying equation (7), this produces two moment values which aggregates the total to be 88*2+2=178 feature vector size, these features have been classified using Euclidian distance for finding the nearest feature vector against database stored feature vectors.

We have applied one more algorithm using geometric features by extracting the locations of the hand fingers’ as well as the palm and wrist centers, our wrist detection is rotation-free and has no drawback since it can detect the wrist area whatever the hand orientation is; as well as all geometric features are rotation-free extraction algorithm, after that the recognition is done using Gaussian distribution modeling for the extracted features, for more details please refer to [28][29].

Acknowledgement

The authors would like to thank the editorial board for bringing the final form of this paper as well as to thank the editor for his coordination and preparing the final template of the paper.

References


[26] Xingyan Li, “Gesture Recognition Based on Fuzzy C-Means Clustering Algorithm”, Department of Computer Science, The University of Tennessee, Knoxville.

