ISSN: 2321-7782 (Online)
Volume 3, Issue 5, May 2015
International Journal of Advance Research in Computer Science and Management Studies
Research Article / Survey Paper / Case Study
Available online at: www.ijarcsms.com

Hybrid Method for Hand Gesture Recognition

Aneer P Imunny
Dept. Electronics And Communication
College Of Engineering Cherthala
India

Abstract: ***This paper proposes two new approaches of hand gesture recognition which will recognize sign language gestures in a real time environment. A hybrid feature descriptor, which combines the advantages of SIFT and SURF methods, is used as a combined feature set to achieve a good recognition rate along with a low time complexity. To further increase the recognition rate and make the recognition system resilient to view-point variations, the concept of derived features from the available feature set is introduced. K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) are used for hybrid classification of single signed letter. Comparative study of these methods with other popular techniques shows that the real time efficiency and robustness are better.***

Keywords: Finger Spelled Word Recognition, Hu Moment Invariant, Hidden Markov Model (HMM), Sign Language, Speeded Up Robust Features (SURF), Support Vector Machine (SVM).

I. INTRODUCTION

Sign language is used as a communication medium among deaf and dumb people to convey the message with each other. A person who can talk and hear properly (normal person) cannot communicate with deaf and dumb person unless he/she is familiar with sign language. Same case is applicable when a deaf and dumb person wants to communicate with a normal person or blind person. In order to bridge the gap in communication among deaf and dumb community and normal community, Video Relay Service (VRS) is being used nowadays. In VRS a manual interpreter translates the hand signs to voice and vice versa to help communication at both ends. A lot of research work has been carried out to automate the process of sign language interpretation with the help of image processing and pattern recognition techniques.

The long-term goal of our research is to enable communication between visually impaired (i.e., blind) people on the one hand and hearing and speech impaired (ie, deaf and dumb) people on the other. Since the former cannot see and the latter use sign language, there is currently no means of communication between such people who are unfortunately in significantly large numbers in a country such as India. Sign Languages (SLs) are made up of thousands of different signs; each differing from the other by minor changes in motion, hand shape, location or Non-Manual Features (NMFs). While Gesture Recognition (GR) solutions often build a classier per gesture, this approach soon becomes intractable when recognizing large lexicons of signs, for even the relatively straightforward task of citation-form, dictionary look- up.

Our project aims to bridge this gap by introducing an inexpensive computer in the communication path so that the sign language can be automatically captured, recognized and translated to speech for the benefit of blind people. In the other direction, speech must be analyzed and converted to either sign or textual display on the screen for the benefit of the hearing impaired. An important research problem in such a solution is the automatic recognition of sign language through image processing. In recent years, there has been a tremendous amount of research on hand gesture recognition. Some of the earlier gesture recognition systems attempted to identify gestures using glove-based devices that would measure the position and joint angles of the hand. However, these devices are very cumbersome and usually have many cables connected to a computer. This has brought forth the motivation of using non-intrusive, vision-based approaches for recognizing gestures.

The purpose of this document is to provide a broad introduction to the field of hand posture and gesture recognition. It also gives a critical review of the information presented so as to point out the general advantages and disadvantages of the various recognition techniques and systems. Although hand postures and gestures are often considered identical, there are distinctions between them. A hand posture is defined as a static movement. For example, making a first and holding it in a certain position is considered a posture. With a simple posture, each of the fingers is either extended or flexed but not in between; for example a fist, pointing, and thumbs up. With a complex posture, the fingers can be bent at angles other than zero or ninety degrees. Complex postures include various forms of pinching, the okay sign and many of the postures used in finger spelling.

A gesture is defined as a dynamic movement, such as waving good-bye. Simple gestures are made in two ways. The first way involves a simple or complex posture and change in the position or orientation of the hand. The second way entails moving the fingers in some way with no change in the position and orientation of the hand, for example, moving the index and middle finger back and forth to urge someone to move closer. A complex gesture is one that includes finger movement, wrist movement and changes in the hands position and orientation. Many of the signs in American Sign Language are examples of this type of gesture.

In human communication, the use of speech and gestures is completely coordinated. Machine gesture and sign language recognition is about recognition of gestures and sign language using computers. A number of hardware techniques are used for gathering information about body positioning; typically either image-based (using cameras, moving lights etc) or device-based (using instrumented gloves, position trackers etc.).

This paper is organized in the following manner: Section 2 explains the proposed approaches. Section 3 explains finger spelled word recognition approach. Experimental results and discussions are shown in section 4. Section 5 summarizes the paper with conclusion.

In this paper, two new approaches (SIFT and SURF) for real time hand gesture recognition which can identify different hand postures in a robust and faster way are introduced. American Sign Language (ASL) alphabet signs are used for recognition process.

Video or Real time images data-set of ASL alphabets is taken in different background and environmental conditions. Considering the tradeoff between recognition rate and processing time, proposed approach uses optimized feature set with combined output from hybrid classifier architecture i.e., KNN and SVM classifiers for single letter recognition is also proposed in this paper.

II. HAND GESTURE RECOGNITION SYSTEM

Hand gesture recognition system consists of the following steps (a) Pre- processing and hand segmentation, (b) Hand detection and tracking, (c) Hand posture recognition and (d) Hand gesture classification as shown in figure below.

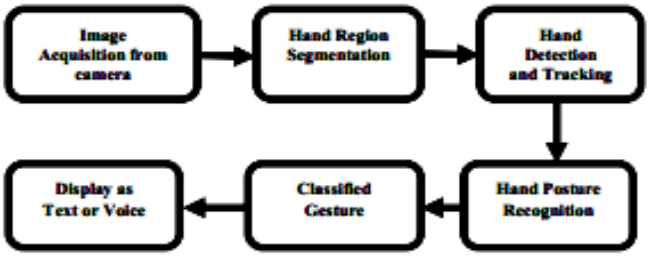
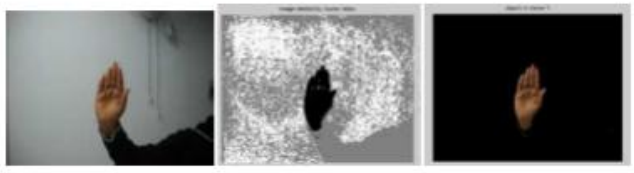


Figure 1. Block diagram of Hand Gesture Recognition System

a) Hand Segmentation

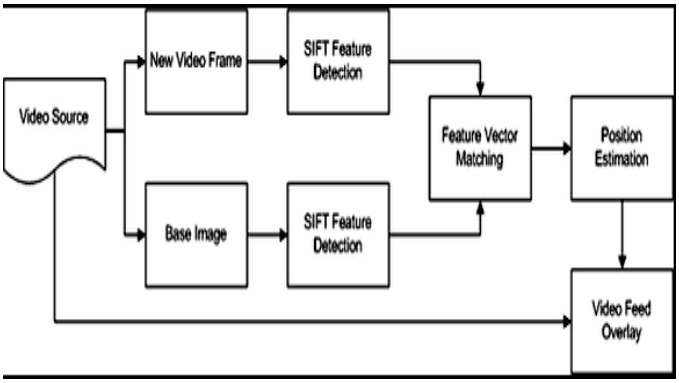
Skin colour segmentation is performed using k-means clustering method or Skin colour detection method. RGB colour frames I(m,n,p), (where m, n and p are number of rows , number of columns and number of colour planes) are converted into three 1-dimensional feature vector X of single column and m*n rows. Then I = X (1), X (2),.. X (N) (N=3) is obtained. Through experimental observation classification of colours using 3 clusters (K) under Euclidean distance measure provides better performance. For every pixel in the input, k-means returns an index corresponding to a cluster. Skin pixel region is identified from the different color regions using a thresholding method in RGB color space where the threshold value is selected experimentally. Repeat the cluster for 3 times to avoid local minima. Figure shows example results of the segmentation algorithm.



(a) Original frame, (b) cluster-labelled image and (c) Skin region segmented image

b) Hand Detection using Invariant Feature Descriptors

After obtaining skin segmented RGB image, it is converted into gray scale. The converted gray scale image is normalized. Invariant features are extracted using Scale Invariant Feature Transform SIFT method. The basic idea is to extract the invariant key point which represents/identifies hand from the segmented image. For this purpose of hand detection, SIFT features are first extracted from a set of reference images and stored in a database. An image frame is matched by individually comparing each feature from the image frame to this previous database and finding candidate matching features based on Euclidean distance of their feature vectors.



Block diagram for SIFT Algorithm

Four major steps are followed to find out invariant key points: scale-space extreme detection, key point localization, orientation assignment and defining key point descriptor. Detecting locations that are invariant to scale change of the image can be accomplished by searching for stable features across all possible scales, using a continuous function of scale known as scale space. The scale space of an image is defined as a function L(x,y) that is produced from the convolution of a variable-scale Gaussian, G(x,y). With an segmented input image SegI (x,y).

$$L(x, y, \sigma) = G(x, y, \sigma) * Segl(x, y)$$

$$G = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

Where '*' is the convolution operation in x and y.

To efficiently detect stable key point locations in scale space, the difference of Gaussian function can be computed from the difference of two nearby scales separated by a constant multiplicative factor k.

$$\begin{aligned} D(x, y, \sigma) &= \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * Segl(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned}$$

The initial image is incrementally convolved with Gaussians to produce images separated by a constant factor k in scale space. Each octave of scale space is divided into an integer number, s, (s=1 at start), of intervals. In the stack of blurred images, s +3 images are produced for each octave, so that final extreme detection covers a complete octave. Adjacent image scales are subtracted to produce the difference-of-Gaussian images. Once a complete octave has been processed, resample the Gaussian image that has twice the initial value of σ (it will be 2 images from the top of the stack) by taking every second pixel in each row and column. The accuracy of sampling relative to σ is no different than for the start of the previous octave, while computation is greatly reduced.

In order to detect the local maxima and minima of D(x, y, σ), each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below. It is selected only if it is larger than all of these neighbors or smaller than all of them. Once a key point candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. The principal curvatures can be computed from a 2×2 Hessian matrix, H, computed at the location and scale of the key point,

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

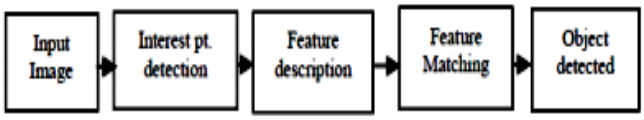
The derivatives are estimated by taking differences of neighboring sample points. The Eigen values of H are proportional to the principal curvatures of D. Key point location is obtained. For each image sample, L(x, y), the gradient magnitude, m(x, y), and orientation (x, y), is pre-computed using pixel differences, to assign orientation to these key points. An orientation histogram is formed from the gradient orientations of sample points within a region around the key point. The orientation histogram has 36 bins covering the 360 degree range of orientations. Peaks in the orientation histogram correspond to dominant directions of local gradients. A key point descriptor is created by summarizing the contents over 8×8 sub regions, from a 16×16 sample array. At last we obtain 4 × 4 array of histogram in 8 orientation bins, 4 × 4 × 8 ≡ 128 element feature vector for each key point.

This 128 × 1 feature vector is called invariant feature descriptor. The database contains both positive (contains hand) and negative images (non-hand image). SIFT key point descriptors are calculated for all the sample images and classified as hand (+1) and non-hand (-1).

Calculated SIFT features are compared with the database feature vectors through Ad boost classifier. If the features are matched, the classifier outputs +1 and sub-window coordinates are stored. If the classifier output is -1, then the sub window is left. By taking average of the matched sub-window coordinates, hand is detected in the segmented image.

c) Recognition Of Letters

The bounding box of the detected hand in each frame is obtained from the previous section. To recognize the posture of detected hand, a combined feature extraction methodology using Speeded Up Robust Features (SURF) and Hu Moment Invariant features is incorporated. Bounding box, BBIm (x, y) is taken as test image. Features are calculated and compared with the database features. Minimum Euclidean distance between the feature vectors recognizes particular hand posture/letter.



Block diagram for SURF Algorithm

Algorithm consists of four major parts.

» Integral image generation
» Interest point detection
» Descriptor orientation assignment (optional)
» Descriptor generation

1. SURF Features

Given an image BBIm(x, y), integral image ii(x, y) is calculated using,

$$ii(x,y) = \sum_{\substack{x1\le x \\ y1\le y}} BBIm(x1,y1)$$

To find out the interest points from the integral image, Fast Hessian Detector is used. Given a point X = (x, y) in image ii(x, y), the Hessian matrix H(X, σ) in X at scale σ is defined as

$$H(X,\sigma) = \begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix}$$

To localize interest points in the image and over scales, non-maximum suppression in a 3 × 3 × 3 neighborhood is applied. The maxima of the determinant of the Hessian matrix are the interpolated in scale and image space. In order to be invariant to rotation, Haar wavelet responses in x and y direction, within radius 6s around interest point is calculated. For the extraction of the descriptor, the first step consists of constructing a square region centered on the interest point. The region is split up regularly into Smaller 4 × 4 square sub-regions. This keeps important spatial information in. For each sub-region, a few simple features at 5×5 regularly spaced sample Points are computed. dx the Haar wavelet response in horizontal direction and dy the Haar wavelet response in vertical direction (f iltersize2s). The wavelet responses dx and dy are summed up over each sub-region and form a first set of entries to the feature vector. Absolute values of the responses |dx| and |dy| provide polarity information.

Each sub-region has a four dimensional descriptor vector. This results in a descriptor vector for all 4 × 4 sub-regions of length 64.

2. *Hu Moment Invariant Geometric Features*

Two-dimensional moments of detected hand image BBIm(x, y) of size m×m is given as,

$$m_{pq} = \sum_{x=0}^{x=m-1} \sum_{y=0}^{y=m-1} (x)^p . (y)^q BBIm(x, y)$$

$$p, q = 0,1,2,3\ldots..$$

The moments BBIm(x, y) translated by an amount (a, b), are defined as

$$\mu_{pq} = \sum_x \sum_y (x + a)^p . (y + b)^q BBIm(x, y)$$

Thus the central moment's mpq or μpq can be computed. Hu defines seven values, computed by normalizing central moments through order three, that are invariant to object scale, position and orientation. Now a 1 × 64 feature vector from surf and 1×7 feature vector from moment invariant are obtained for all the reference (posture/ letter) images and stored in a database.

3. *Classification using K-Nearest Neighbour*

Combined feature vectors of database images are stored in a database. These feature vectors are classified using KNN classifier. Feature vectors of detected hand image from subsequent frame are compared with the stored feature vectors by means of a Euclidean distance measure in KNN.

In pattern recognition, the k-Nearest Neighbors algorithm (or k-NN for short) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. The output depends on whether k-NN is used for classification or regression.

In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. In k-NN regression, the output is the property value for the object. This value is the average of the values of its k nearest neighbors. k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. The k-NN algorithm is among the simplest of all machine learning algorithms. Both for classification and regression, it can be useful to weigh the contributions of the neighbors, so that the nearer neighbors contribute more to the average than the more distant ones. For example, a common weighting scheme consists in giving each neighbor a weight of 1/d, where d is the distance to the neighbor. The neighbors are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required. A shortcoming of the k-NN algorithm is that it is sensitive to the local structure of the data. The algorithm has nothing to do with and is not to be confused with k-means, another popular machine learning technique.

4. *Classification using Support Vector Machines*

Simultaneously the feature vectors of the dataset are given to SVM classifier for training. The basic principle of SVM is to find an optimal separating hyper plane (OSH) which can separate different classes in a feature space, that is, the distances

between these classes should be the furthest. To perform the classification between two classes, a nonlinear SVM classifier is applied by mapping the input data (xi,yi) into a higher dimensional feature space using a non-linear operator φ(x).

The OSH can be computed as a decision surface:

$$f(x) = sign(\sum_i \alpha_i y_i K(x_i, x) + b).$$

Where sign () is the sign function and K (xi, x) = φ(xi) τφ(x) is the predefined kernel function. The coefficients αi and b in can be determined by the quadratic problem. This procedure is carried out for the sequence of detected hand from video frames. For each frame classifier recognizes a single letter as output. The results given by both the classifiers are taken as a combined feature vector for gesture classification.

5. Improving accuracy through derived features

To further increase the recognition rate and speed of processing, prominent features are derived from the available data set of features using forward selection algorithm. To improve the performance of classifier on a dataset, it is possible to evaluate each features deviation. The deviation is computed Per feature xj in the set of N features x = x1, ..., xN by calculating the sum of all differences between the calculated resultant feature vector when feature xj is left out and the actual resultant feature vector oi of sample si in the dataset D=N×P, containing P samples, where each sample si contains N features. For clarity, well define a new feature set y that excludes xj.

» The algorithm run as follows on feature xj, with feature set y that excludes xj :
The classifier resultant feature vector over the feature set y for the given test sample (detected hand from the video sequence) is calculated.

» The forward selection computes the best features of the data set, i.e. features whose feature deviation is minimal and prominent for best recognition rate. The resultant feature vector along with its deviation parameter is appended to the already available surf and moment invariant feature database for single letter recognition using hybrid classifiers.

III. CONCLUSION

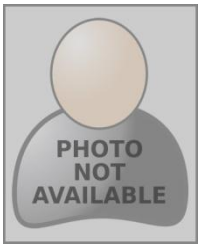
It is observed from the experimental results that SIFT and SURF, is robust against multiple variations like rotation, scale, lighting and view-point and provides good real time performance. Both approaches make use of hybrid classifier architecture such as KNN and SVM. The tradeoff between accuracy and speed of processing is maintained by the methods. . In future, research work will be focused on automatic Indian sign language (ISL) interpretation as text or voice. As ISL uses both hands for signing it involves both local and global hand movements thus the concept of gesture spotting, inter-hand occlusion will be investigated deeply in near future.

References

1. G.R.S.Murthy & R.S. Jadon,"A Review of vision based hand gestures recognition", International Journal of Information Technology and Knowledge Management, Vol-2,pp. 405-410, July-December 2009.
2. Y.Fang, K.Wang, J.Cheng & H.Lu, "A Real Time hand gesture recognition method", IEEE 2007.
3. Salleh, Jias, Mazalan, Ismail, Yussof, Ahmad, Anuar & Mohamad, "Sign Language to Voice Recognition: Hand Detection Techniques for Vision-Based Approach", In Conference Current Developments in Technology-Asssisted Education, FORMATEX 2006.
4. Paul Viola & Michael Jones, "Robust Real-Time Object Detection", Second International Workshop on Statistical and Computational Theories of Vision-Modelling, Learning, Computing and Sampling, Vancouver, Canada, July 13 2001.
5. H.Zhou, D.Lin & T.S.Huang, "Static Hand Gesture Recognition based on Local Orientation Histogram Feature Distribution Model", In IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops CVPRW'04 2004.
6. Chan Wah Ng, Surendra Ranganath, "Real-Time Gesture Recognition system and application", Image and Vision Computing 20 (2002) 993-1007.
7. T.Dinh, V.Dang & D.Duong, "Hand gesture classification using boosted cascade of classifiers", IEEE 2006.
8. T.Starner & A.Pentland, "Real-Time American Sign Language Recognition from video Using Hidden Markov Models", IEEE 1995.

9. G.Rigoll, A.Kosmala & S.Eickeler, "High Performance Real-time Gesture Recognition Using Hidden Markov Models" Gerhard-Mercator-University, Duisburg-Germany.
10. R.Liang & M.Ouhyoung, "A Real Time Continuous Gesture Recognition System for sign language", Shih-Chien University Taipei-Taiwan.
11. S.Mitra & T.Acharya, "Gesture Recognition: A survey", IEEE Transcations on Systems, Man and Cybernetics, part Applications and Reviews, Vol.37, No.3 May 2007.
12. K.S. Ravichandran & B. Ananthi, "Color Skin Segmentation Using K-Means Cluster", International Journal of Computational and Applied Mathematics ISSN 1819-4966 Volume 4 Number 2, pp. 153–157 2009.
13. Chieh-Chih Wang & Ko-Chih Wang, "Hand Posture Recognition Using Adaboost with SIFT for Human Robot Interaction", In Springer-Verlag Berlin- Recent Progress in Robotics, LNCIS 370, pp.317-329, 2008.
14. M.Elmezain, A.Al-Hamadi & B.Michaelis, "Hand Gesture Recognition based on combined features extraction", World Academy of Science, Engineering and Technology 60 2009.
15. H.Bay, T.Tuytelaars & Luc V.Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding (CVIU), Vol.110, No. 3, pp. 346—359, 2008.

AUTHOR(S) PROFILE



Aneer P Imunny, received the B.Tech degree in Electronics And Communication from KMEA Engineering College Aluva, in 2012. Currently pursuing the M.Tech degree in Signal Processing from College Of Engineering Cherthala affiliated to CUSAT, Kerala.

