Recognition of Single Handed Sign Language Gestures using Contour Tracing Descriptor

Rohit Sharma, Yash Nemani, Sumit Kumar, Lalit Kane, Pritee Khanna, *Member, IAENG*

***Abstract*— Recently, many researchers have focused on building dual handed static gesture recognition systems. Single handed static gestures, however, pose more recognition complexity due to the high degree of shape ambiguities. This paper presents a gesture recognition setup capable of recognizing and emphasizing the most ambiguous static single handed gestures. Performance of the proposed scheme is tested on the alphabets of American Sign Language (ASL). Segmentation of hand contours from image background is carried out using two different strategies; skin color as detection cue with RGB and YCbCr color spaces, and thresholding of gray level intensities. A novel, rotation and size invariant, contour tracing descriptor is used to describe gesture contours generated by each segmentation technique. Performances of k - Nearest Neighbor (k-NN) and multiclass Support Vector Machine (SVM) classification techniques are evaluated to classify a particular gesture. Gray level segmented contour traces classified by multiclass SVM achieve accuracy up to 80.8% on the most ambiguous gestures of ASL alphabets with overall accuracy of 90.1%.**

***Index Terms*— Sign Language Recognition, Contour Trace, ASL, k-NN Classification, Multiclass SVM**

I. Introduction

SIGN language is a primary way of communication for deaf and dumb people that uses hand and arm movements, facial expressions and eyebrow movements to express the thought of a speaker without speaking it. Many countries have their own sign languages associated with their own grammars and rules. American Sign Language (ASL), British Sign Language (BSL), French Sign Language (LSF), and Indian Sign Language (ISL) are such languages to name a few. Sign Languages of some countries share a lot of similarities among them whereas some sign languages quite differ to each other. BSL and ISL [1] share a number of characteristics including dual handed alphabet representation which is in contrast to ASL that uses single handed representation for alphabets [2].

One-to-one communication between a hearing impaired and a hearing person simply does not exist in the present scenario. Researchers have come up with a temporary solution for this problem, called Video Relay Service (VRS) centre. A remote manual interpreter sitting in the VRS centre helps to establish conversation between hearing impaired (deaf/dumb) and hearing person by translating hand signs into voice and vice versa through videophones connected by internet. To eliminate this manual intervention, automated sign language interpretation is highly desirable. Sign language recognition is a multidisciplinary research area involving pattern recognition, computer vision, natural language processing and psychology. It is a comprehensive problem due to the complexity of the visual analysis of hand gestures and the highly structured nature of sign languages. To convey something, a signer makes use of a 3D space around her, called Signing space. Majority of the sign languages include manual as well as non-manual components. Parameters like hand's shape, orientation, position, and movements characterize a manual component whereas non-manual components are characterized by facial expressions, eye gaze, and head/body posture [3].

Gestures used to denote alphabets and digits are usually static, involving only a single key posture of the body. This is contrary to dynamic gestures where movement of the hands is also associated that leads to multiple key postures. Proposed work focuses manual static gestures of ASL. ASL, compared to dual handed sign languages, has higher number of ambiguous static gestures which makes its recognition a challenging task (Fig. 1).

Most of the gesture recognition systems work usually in following steps; gesture signal acquisition, processing of acquired gesture signal and descriptor extraction, and finally classification of descriptor to one of the probable gestures. Gesture recognition techniques for sign languages can broadly be categorized as sensor glove based, EMG (Electromyography) sensor based and vision based

Fig.1. ASL alphabet gestures A, E, M, N, S, and T, starting from top left to bottom right show an inherent ambiguity

Manuscript received March 05, 2013; revised March 28, 2013.

Rohit Sharma (rohit09103@iiitdmj.ac.in), Yash Nemani (yash09146@iiitdmj.ac.in), and Sumit Kumar (sumit09127@iiitdmj.ac.in) are undergraduate students in Computer Science and Engineering discipline at Pandit Dwarka Prasad Mishra Indian Institute of Information Technology, Design and Manufacturing (PDPM-IIITDM) Jabalpur, INDIA.

Lalit Kane is a PhD research scholar in Computer Science and Engineering discipline at PDPM-IIITDM Jabalpur, INDIA. (e-mail: lalit.kane@iiitdmj.ac.in).

Pritee Khanna is an Associate Professor in Computer Science & Engineering discipline at PDPM-IIITDM Jabalpur, INDIA (phone: +919425324241; e-mail: pkhanna@iiitdmj.ac.in).

techniques. In sensor gloves based approach, signer wears a pair of gloves equipped with electronic sensors to generate electric signals [4], [5]. Sensor gloves based techniques result in high reliability of recognition without any preprocessing overhead. However, they cause wearing inconvenience to signers and pose difficulties in natural movements. EMG (Electromyography) sensors measure electric signals generated by muscle cells and can be employed in manual as well as non-manual gesture recognition [5], [6]. Still, extracting and processing of features using EMG sensors is a complex and quite subjective process due to factors like random noise. In vision based gesture recognition techniques, one or more cameras capture the gestures performed by a signer in an image or video stream. Hand blobs are extracted from captured images or frames and processed further for descriptor (feature) extraction.

A number of works can be listed in the direction of single handed sign language recognition. Zaki and Shaheen [7] introduce an ASL recognition system that uses Principle Component Analysis (PCA), Kurtosis Position, and Motion Chain Codes as descriptors for sign gestures. Here PCA is claimed to retain configuration and orientation of hands while Kurtosis Position and Motion Chain Codes deal with dynamic gestures. The system evaluates these descriptors individually as well as in fusion.

A recognition system for single handed gestures of Persian Sign Language (PSL) by Karami et al. [8] employs 2D discrete wavelet transform with 9 levels of decomposition for each gesture image. The coefficient matrices obtained from wavelet decomposition tree are used as descriptors which in turn are given to a Multi Layer Perceptron Neural Network (MLP-NN) for classification into a fair accuracy.

In an ASL alphabet recognition work by Munib et al. [9], gesture images are converted from RGB color space to gray scale. Canny's edge detection algorithm is applied to detect sharp intensity changes. The descriptors are prepared by applying Hough transform which is customized to recognize arbitrary shapes. A three layer feed forward back propagation neural network is used for classification.

Another approach by Athitsos et al. [10] recognizes 20 hand shapes of ASL dataset by generating a large number of articulated 3D hand orientations for each hand shape. These 3D hand orientations are estimated using various viewpoints of the camera and stored in a database. A gesture is recognized by measuring chamfer distance between edge images of the input and each of the images stored in the database.

Though all the above mentioned recognition systems offer fair efficiency of recognition (90% or more) over entire gesture set, they do not emphasize much on the recognition of ambiguous gestures of the dataset under consideration. Novelty of the proposed work is driven by the emphasis on separation of closely resembling gestures and justified by fair results in recognition of such gestures. Proposed system works well with the single handed sign language ASL containing various ambiguous signs.

Proposed work uses contour trace which is an efficient representation of hand contours as hand shape descriptor and classifies the contour traces of input hand images using k-Nearest Neighbor and multiclass Support Vector Machine methods. The remaining text is organized in three sections; section II discusses the proposed methodology, experimental results are analyzed in section III, conclusions and further work is discussed in section IV.

II. Proposed Work

Proposed gesture recognition system works in three phases; preprocessing, descriptor preparation (preparing contour trace), and classification. Flowchart of this system is given in Fig. 2.

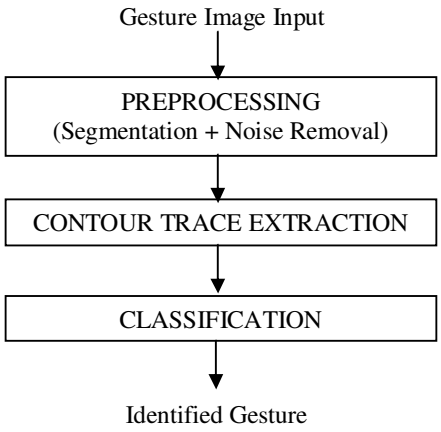

Fig. 2: Flow of the proposed system

A. Preprocessing

In preprocessing step, segmentation and noise removal operations (post processing) are carried out to separate background from hand blobs denoting gestures in a controlled environment. Popular RGB and YCbCr color spaces are considered here for detecting skin color from the whole image. Piecewise linear classifiers for these color spaces [11], [12], [13] given in Table I are tested. Experiments reveal that piecewise linear classifier for YCbCr color space performs better.

It is observed that although YCbCr based classifier yields better results than RGB based classifier, the contours extracted from these two methods are still erroneous, as depicted in Fig. 3. Faulty extraction greatly owes to varying illumination and inconsistent background.

As a consequence of Piecewise linear classifiers' performance, two different gray level based thresholding approaches; Otsu's algorithm [14], and a simple automated thresholding [15] are evaluated here for hand contour segmentation. Thresholding algorithm by Otsu attains optimum threshold by iterating through each gray level of the image. At each iteration, image is segmented into two regions and algorithm terminates when sum of the variances of two segmented regions is minimum (i.e. minimum within-class variance). One faster version of this algorithm attempts to maximize the between-class variance.

TABLE I
Piecewise Linear Classifiers (Color space ranges) for RGB and YCbCr Color Spaces [11], [12]

RGB	YCbCr
R > 95, G > 40, B >20, Max{R,G,B} - Min{R,G,B} < 15 |R-G| > 15, R > G, R > B	Cb: 77 –127, Cr: 133 – 173

ISBN: 978-988-19252-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

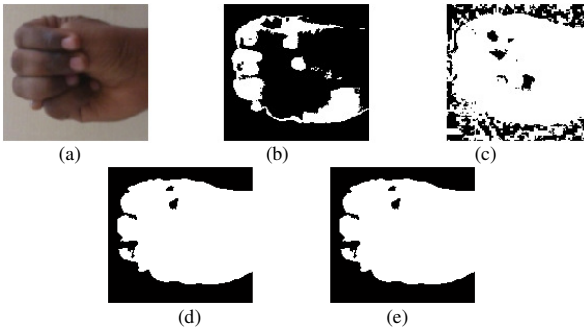

Fig. 3. Results of segmentation on (a) Original image; using (b) RGB linear piecewise classifier (c) YCbCr linear piecewise classifier (d) Otsu's thresholding (e) Simple automated thresholding

Automated thresholding approach gives comparable results at par with Otsu's algorithm. Algorithmic steps for this approach are as follows:

1. Select an initial threshold T (say, an average of minimum and maximum gray level intensities in the image).
2. Segment the image using T that yields two regions; background with gray level values $< T$, and foreground with values $\geq T$.
3. Compute average gray levels T_b and T_f for background and foreground regions respectively.
4. Compute a new threshold as

$$T = \left(T_b + T_f\right)/2 .$$

5. Iterate through steps 2 to 4 as long as the difference between successive thresholds is greater than some predefined parameter (a threshold greater than 5% of the penultimate threshold value is predefined criteria in this work).

In segmented binary images, noise is in the form of incorrectly segmented chunks which are considerably smaller as compared to hand region and must be alleviated. An 8-connected component analysis is conducted to label various dispersed regions. Largest region is kept for hand blob processing and other regions are discarded. This step results in an implicit removal of noise (Fig. 4).

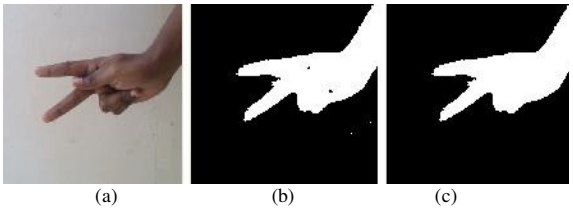

Fig.4. Preprocessing of gesture image 'K' (a) Original Image (b) Image after segmentation (c) Image after noise removal

B. Preparing Descriptor

In order to classify each gesture, it is mandatory to preserve its shape assimilated in the outline of the contour. The primary step here is to acquire outline of the contour extracted in preprocessing stage and then to give it an efficient representation. Steps of the *contour tracing algorithm* are as follows:

1. Examine each image pixel from top to bottom and left to right to locate first contour pixel P_{start}. Record spatial location of this pixel $P_{start} = P(i,j)$.
2. Determine next outline pixel P_{next}. The probable candidate pixels are $P(i+1,j)$, $P(i+1,j+1)$, $P(i,j+1)$, $P(i-1,j+1)$, $P(i-1,j)$, $P(i-1,j-1)$, $P(i,j-1)$, and $P(i+1,j-1)$. Select a pixel with intensity 1 and having at least one of its neighbors with intensity 0 (a pixel that belongs to the background). Record the spatial location of this pixel.
3. Set $P(i,j) = P_{next}$ and repeat step 2 until $P_{next} = P_{start}$.
4. Let X_i represent i^{th} point (x_i,y_i) in a sequence of N contour outline points where $i = 1,2,\ldots,N$. Find the bounding box (a rectangle that correctly fits the contour points in it).
5. Align the major axis of this box with horizontal axis (X-axis). This step, in effect, calculates an angle θ between horizontal axis and major axis of bounding box and transforms each point by an angle θ.
6. Measure and store the perpendicular distance of each point obtained in step 5 to horizontal axis. Normalize entire sequence by dividing it with the largest perpendicular distance obtained.

Let D_i be the i^{th} sample in a sequence of N perpendicular distances, where $i = 1,2,\ldots,N$ and $M = max\{D_i\}$. Normalized distance of sample i is given as

$$ND_i = D_i / M .$$

A set of ND_i thus obtained is referred to as *contour trace* and forms a unique descriptor for hand shape. Fig. 5 shows contour traces for two ambiguous ASL alphabets 'M' and 'N' obtained by Otsu's algorithm and simple automatic thresholding.

Step 5 in the contour tracing algorithm makes the hand gesture rotation invariant. First, an angle θ between major axis (represented by the line connecting two farthest points

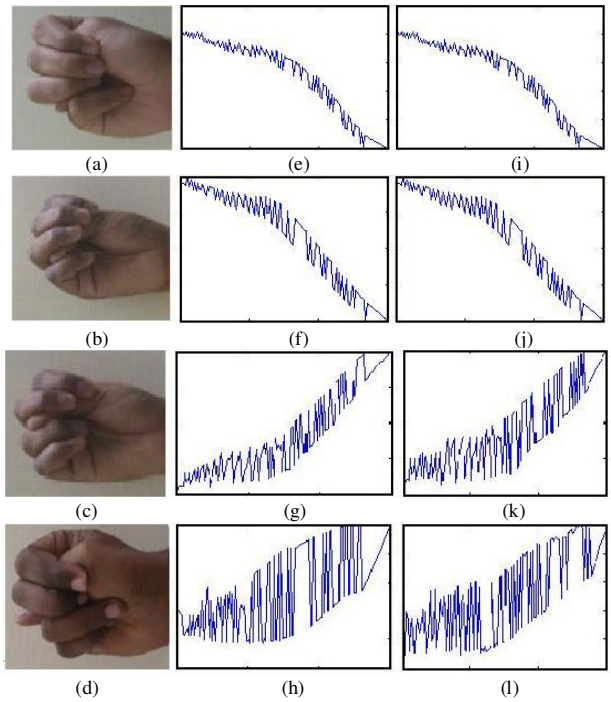

Fig.5. Gesture images of ambiguous ASL alphabet 'M' (a), (b) and alphabet 'N' (c), (d) in the first column
Corresponding contour traces of these gestures using Otsu's thresholding in the middle column, (e) - (h)
Contour traces using automated thresholding in the last column, (i) - (l)

ISBN: 978-988-19252-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

TABLE III
CONFUSION MATRIX FOR VARIOUS ALPHABETS OF ASL

	A	C	D	F	G	H	I	K	L	M	N	R	S
A	70	0	0	20	0	0	0	0	0	0	0	0	10
C	0	90	10	0	0	0	0	0	0	0	0	0	0
F	0	0	0	75	0	0	0	0	25	0	0	0	0
G	0	0	0	15	65	0	0	20	0	0	0	0	0
H	25	0	0	0	0	75	0	0	0	0	0	0	0
I	0	0	0	0	15	0	85	0	0	0	0	0	0
K	0	0	0	0	0	0	0	95	05	0	0	0	0
M	0	0	0	0	0	0	0	0	0	80	20	0	0
N	0	0	0	0	0	0	0	0	0	30	70	0	0
R	0	0	0	0	0	0	0	25	0	0	0	75	0
S	0	0	0	0	0	0	0	20	15	0	0	0	65

on the outline sequence) and horizontal axis is found. Afterwards, each point on outline undergoes a rotation transformation by θ° that causes major axis to coincide with horizontal axis. Step 6 in the contour tracing algorithm makes the gesture invariant to size by dividing each sample on the outline with maximum valued sample. Thus irrespective of the hand size of an individual or magnification or reduction factors, contour trace has a normalized value between 0 and 1. Total sample points on the contour trace are also restricted to 120.

C. Classification

Due to the inherent simplicity, *k*-Nearest Neighbor classification technique is used to recognize 26 ASL English alphabets. According to this algorithm, an unknown gesture pattern is classified to the class most common amongst its *k* nearest neighbor gestures, where *k* is a small positive integer. The *k* nearest gesture patterns are selected from the entire training set on the basis of the closest Euclidean distance to the test gesture. All relevant computations are carried out at the time of classification. In the training phase, algorithm stores descriptors (feature vectors of size 120) and class labels of the training gesture patterns.

However, driven by slight underperformance of k-NN classifier, multiclass Support Vector Machine (SVM) is applied for classification. The basic SVM scheme classifies input pattern to one of the two classes but multi class extension of SVM follows one to all differentiation to classify one class to all other classes. An input hand gesture image is to be assigned to one of the classes {a, b,…,z, ω}, where a-z are class labels for alphabet gesture A – Z and ω is the label for non-alphabets.

III. RESULTS AND DISCUSSION

Classification results obtained from various contour traces are depicted in Table II. Graph in Fig. 6 shows percentage of error occurred in each combination of segmentation and classification techniques. Training dataset comprises of 468 images of 26 ASL alphabets. Six different signers performed each alphabet gesture 3 times (6x3x26). The dataset is divided into two subsets – training subset (comprising of 6x 2x26=312) images and testing/validation subset (comprising of 6x1x26=156) images. Resolution of images is kept as low as 240x230 to compensate for large number of computations and enhance computation speed. All experiments are carried out using standard JDK 1.2.6 platform and its image processing APIs. Both, Color based and gray level based techniques are observed for segmentation, and it is observed that automated gray level thresholding outperformed other segmentation techniques. This is also observed that classification results yielded by multiclass SVM have distinct improvements over k-NN classification technique. Table III displays a confusion matrix showing efficiency of recognition by multiclass SVM. Rows and columns with no recognition confusion are omitted from this matrix.

TABLE II
RECOGNITION EFFICIENCIES BY K-NN AND SVM CLASSIFICATION SCHEMES FOR VARIOUS SEGMENTATION STRATEGIES

Segmentation Strategy	k-NN Classification Efficiency	Multiclass SVM Classification Efficiency
RGB Piecewise Linear Classifier	40	52.3
YCbCr Piecewise Linear Classifier	58.2	62.3
Otsu's Gray Level Thresholding	70.2	88.2
Simple Automated Thresholding	67.9	90.1

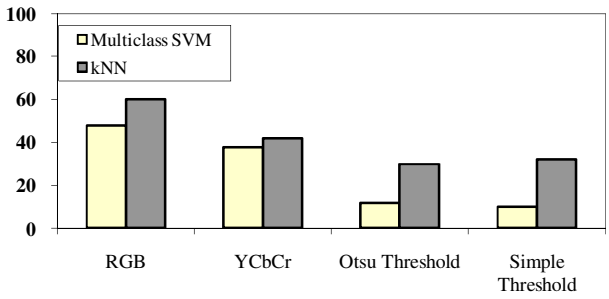

Fig.6. Graphical representation of recognition error for all segmentation schemes

The system performs fair for ambiguous gestures A, E, M, N, S, and T as shown in Table III. Alphabet 'A' has 10% confusion to alphabet S while it has no confusion to 'E', 'M', 'N', and 'T'. However, it is confused to 'F' 20% times. 'E' has not been confused to any alphabet and hence omitted from this table. 'M' is confused to 'N' 20% times and 30% in vice-versa case. From this matrix it is evident that the proposed system gives 80.8% recognition rate for ambiguous gestures. It is also evident from confusion matrix that gesture 'G' is misclassified to gestures 'F' and 'K' by 15% and 20%, respectively, whereas correct recognition rate of 'G' is 65%. Similar case is with gesture 'S' which gets misclassified to gesture 'K' by 20% and 'L' by 15% with correct recognition rate of 65%. However, considering average recognition rates of all gestures, overall efficiency of the ASL gesture recognition system computes to 90.1% using multiclass SVM classification.

ISBN: 978-988-19252-8-2
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

IV. Conclusions

In this work, four different strategies for segmentation of hand from constant background are evaluated towards building a robust static, single handed gesture recognition system. ASL is chosen for case study due to inherent ambiguity of various single handed alphabets. It is experimentally found that gray level based segmentation strategies yield better results. Rotation and size invariant contour trace feature set is used to describe various ASL alphabet shapes. This feature set is evaluated using *k*-Nearest Neighbor classification and multiclass SVM strategies with later one performing better than the former. In further work, efforts would be made to improve the accuracy of the current system. Improvement of accuracy with signer independent recognition will be emphasized.

Acknowledgment

The proposed work is done during the Project Based Internship (PBI) of the undergraduate students at PDPM IIITDM Jabalpur. We acknowledge the support provided by the Institute.

References

[1] A Guide to British Sign Language, www.britishsignlanguage.com, Accessed on January 30, 2013.

[2] American Sign Language University, www.lifeprint.com, Accessed on February 5, 2013.

[3] S. Ong and S. Ranganath, "Automatic sign language analysis: A survey and the future beyond lexical meaning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, issue 6, pp. 873–891, 2005.

[4] R. Xu, X. Zhou, W. Li, "MEMS Accelerometer Based Nonspecific - User Hand Gesture Recognition", *IEEE Sensors Journal*, vol. 12, issue 5, pp. 1166–1173, 2012..

[5] Y. Li, X. Chen, J. Tian, X. Zhang, K. Wang, J. Yang, "Automatic Recognition of Sign Language Subwords based on Portable Accelerometer and EMG Sensors", *ACM Conference ICMI-MLMI*, Beijing, China, 2010.

[6] M. Al-Ahdal and N. Tahir, "Review in sign language recognition systems", *IEEE Symposium on Computers and Informatics (ISCI)*, pp. 52–57, 2012.

[7] M. Zaki and S. Shaheen, "Sign language recognition using a combination of new vision based features", *Pattern Recognition Letters*, vol. 32, issue 4, pp. 572–577, 2011.

[8] A. Karami, B. Zanj, A. Sarkaleh, "Persian sign language (PSL) recognition using wavelet transform and neural networks", *Expert Systems with Applications*, vol. 38, pp. 2661–2627, 2011.

[9] Q. Munib, M. Habeeb, B. Takruri, H. A. Al-Malik, "American sign language (ASL) recognition based on Hough transform and neural networks", *Expert Systems with Applications*, vol. 32, pp. 24–37, 2007.

[10] V. Athitsos, H. Wang, A. Stefan, "A database-based framework for gesture recognition", *Personal and Ubiquitous Computing*, vol. 14, issue 6, pp. 511–526, 2010.

[11] J. Kovac, P. Peer, F. Solina, "Human skin colour clustering for face detection", *The IEEE region 8 EUROCON, Computer as a Tool*, vol.2, pp. 144–148, 2003.

[12] D. Chai and K. Ngan, "Face Segmentation using skin-color map in videophone applications", *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 551–564, 1999.

[13] K. Nallaperumal, S. Ravi, C. N. Kennady Babu, R. K. Selvakumar, A. L. Fred, C. Seldev, S.S Vinsley, "Skin Detection using Color Pixel Classification with Application to Face Detection: A Comparative Study", *International Conference on Computational Intelligence and Multimedia Applications*, pp. 436–441, 2007

[14] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms", *IEEE Transactions on Systems, Man, and Cybernencs*, vol. 9, issue 1, pp. 62–66, 1979.

[15] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. New Jersey: Prentice Hall, 2002, ch. 10.