Shape, Texture and Local Movement Hand Gesture Features for Indian Sign Language Recognition

2 authors:

Rekha Jayaprakash
institute of laser and system technologies (iLAS), TUHH
6 PUBLICATIONS   **115** CITATIONS

SEE PROFILE

Somajyoti Majumder
Central Mechanical Engineering Research Institute
91 PUBLICATIONS   **817** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Tele-operated Mobile Robot for Underground Coal Mines View project

Design of a Tree Climbing Robot View project

Shape, Texture and Local Movement Hand Gesture Features for Indian Sign Language Recognition

J. Rekha[1], J. Bhattacharya[2], S. Majumder[3]
[1]Scientist, [3]Scientist & Head, Surface Robotics Laboratory,
Central Mechanical Engineering Research Institute (CMERI-CSIR)
Durgapur, India
[1]jayaprakash.rekha@gmail.com, [2]bjhilik@gmail.com, [3]sjm@cmeri.res.in

Abstract--**This paper proposes an automatic gesture recognition approach for Indian Sign Language (ISL). Indian sign language uses both hands to represent each alphabet. We propose an approach which addresses local-global ambiguity identification, inter-class variability enhancement for each hand gesture. Hand region is segmented and detected by YCbCr skin color model reference. The shape, texture and finger features of each hand are extracted using Principle Curvature Based Region (PCBR) detector, Wavelet Packet Decomposition (WPD-2) and complexity defects algorithms respectively for hand posture recognition process. To classify each hand posture, multi class non linear support vector machines (SVM) is used, for which a recognition rate of 91.3% is achieved. Dynamic gestures are classified using Dynamic Time Warping (DTW) with the trajectory feature vector with 86.3% recognition rate. The performance of the proposed approach is analyzed with well known classifiers like SVM, KNN & DTW. Experimental results are compared with the conventional and existing algorithms to prove the better efficiency of the proposed approach.**

Keywords-Dynamic Time Warping (DTW); Indian Sign Language (ISL); Principle Curvature Based Region (PCBR) Detector; Support Vector Machines (SVM); Wavelet Packet Decomposition (WPD-2)

I. INTRODUCTION

India is diversified in culture, language and religion. Since there is a large diversity among Indian languages, literature survey reports the non-existence of standard form of Indian Sign Language (ISL) gestures. ISL alphabets are derived from British Sign Language (BSL) and French Sign Language (FSL). Due to these issues, standard database for ISL alphabet/gestures have not been developed so far. Few research works has been carried out in ISL recognition and interpretation using image processing/vision techniques. But those are only initial work tried with simple image processing techniques and are not dealt with real time data. A novel approach for recognizing Indian Sign Language (ISL) gestures for Humanoid Robot Interaction (HRI) is introduced which imparts an elegant way of interaction between humanoid robot HOAP-2 and human being [1]. Orientation of edges in an image is being considered as a feature vector. Different direction histogram in the form of 18 and 36 bins are obtained from the entire ISL video. The classification technique deals with the Euclidean distance metric. Later a system to translate the input speech to ISL which is displayed with the help of a 3D virtual human avatar is proposed later [3]. Input to the system is the clerk's speech which is in English. The speech recognition module recognizes speech and makes a text output. This text is subsequently passed to a parser module which tokenizes the string and tag the part-of-speech using a sample file. The output from the parser is given to an eliminator module which performs a reduction task by eliminating unwanted elements and further the root form of verbs are found using the stemmer module. The structural divergence of English and ISL is handled by a phrase reordering module using ISL dictionary and rules [3]. This module generates ISL-gloss strings which can be played back through the 3D virtual human. A 3D animation module creates the animation from the motion captured data. In this approach a whole lot of 3D model data is used which makes the system clumsy and bulk. Attempt to automatic translation of static as well as dynamic gestures of ISL with image processing features such as skin tone detection spatial filter velocimetry and temporal tracing is developed [4] [5]. The power spectrum representation of each gesture is given as motion prints. Edge detection, clipping and boundary tracing are used as features for recognition process. These methods work well for static signs of ISL. They do not deal with the dynamic, global and local movements of ISL gestures. For example, ISL signs of letters A-B, M-N, U-V look similar in appearance. Sometimes it becomes hard for human being to correctly recognize the sign. When it comes to computer, the inter-class variability parameter should be considered.

So we propose an approach which will prominently distinguish the classes (alphabets), by considering local-global configuration of each hand gesture through finger movement and count. Along with this, shape and texture information is used to accurately predict the alphabets signed by the signer.

This paper is arranged in the following fashion: section II describes proposed method in detail. Experimental results are discussed in detail in Section III. Section IV summarizes and concludes the paper.

30

II. PROPOSED METHOD

The proposed method consists of following steps: Segmentation and detection of hand from each video frame using skin color model, which is followed by hand feature extraction using Principal Curvature based Region Detector [7] and 2-D Wavelet Packet Decomposition [6]. At the same time, to extract the local movements in each hand gesture finger count is taken into account. The extracted features are converted into appropriate feature vectors. Multi class non linear support vector machines (SVM) is used for the classification of each hand gestures (Alphabets).

A. Segmentation and hand detection from video

First, pre-processing and normalization are done on the video object frames. Skin color segmentation is performed in YCbCr color space since it reduces the effect of uneven illumination in an image. YCbCr is an encoded nonlinear RGB signal with simple transformation; explicit separation of luminance and chrominance components makes this color space attractive for skin color modeling. RGB color frames *I(m,n,p)* (where m, n and p are number of rows , number of columns and number of color planes) are converted into YCbCr images using

$$Y = 0.299R + 0.587G + 0.114B \qquad (1)$$
$$C_r = R - Y$$
$$C_b = B - Y$$

A parametric method called Single Gaussian Model [3] for skin color uses mean and covariance of chrominant color with a bivariate Gaussian distribution with c as color vector representing random measured values of chrominance *(x,y)* of a pixel with coordinates *(i,j)* in an image. W_s is the class describing the skin. μ_s is the mean vector and the covariance matrix for skin chrominance.

$$p\left[\frac{c}{W_s}\right] = (2\pi)^{-1}|\Sigma_s|^{-1/2}exp^{(c-\mu_s)^T \Sigma_s^{-1}(c-\mu_s)} \qquad (2)$$

$$c = [x(i,j)y(i,j)]^T$$

$$\mu_s = \frac{1}{n}\Sigma_{j=1}^n C_j\text{t} \qquad (3)$$

Mahalanobis distance can be used to measure the distance between c color vectors to mean vector *μs*.

$$\lambda_s(c) = (c - \mu_s)^T \Sigma_s^{-1}(c - \mu_s) \qquad (4)$$

The input image frame color vector is compared with the stored skin color model data and the pixels representing minimum distance is considered as skin. Those skin pixels are converted back to RGB space. By considering the skin region threshold value hand is detected in each frame. Now the detected hand image is given by *D(i,j)*.

B. Feature Extraction for ISL Alphabet Recognition

Good features are indeed necessary for better classification. Features represent a particular object in a well defined manner.

Sometimes one or more features are required to define an object. Since hand has maximum flexibility, even a small change in the finger configuration of a particular gesture needs consideration. Exactly the same gesture cannot be repeated by the user every time. There may be some factors which will change with time.

Two well known feature extraction methods are applied on hand image to obtain the shape and texture information. They are Principal Curvature Based Region detector (PCBR) [7] and 2-D Wavelet Packet Decomposition (WPD) [6].

1) Principle Curvature Based Region Detector- PCBR

Basically PCBR captures semi-local structural cues such as edges and curvilinear shapes. These structural cues tend to be more robust to intensity, color and pose variations. They provide more stable interest operator, which in turn improves object recognition accuracy. It identifies stable watershed regions within the multi-scale principal curvature image. So this detector can be used to extract features from hand images which are varied in view point, scale and illumination. Detected hand portion from each frame is represented by *I(x,y)*.

Curvilinear structures are lines (either curved or straight) such as roads in aerial or satellite images or blood vessels in medical scans. These curvilinear structures can be detected over a range of viewpoints, scales, and illumination changes. The PCBR detector employs the first steps of Steger's curvilinear detector algorithm [7]. It forms an image of the maximum or minimum Eigen value of the Hessian matrix at each pixel. This is called the principal curvature image [7], as it measures the principal curvature of the image intensity surface. This process generates a single response for both lines and edges, producing a clearer structural sketch of an hand image than is usually provided by the gradient magnitude image.

2) Principal Curvature Image

The local shape characteristics of the surface at a particular point can be described by the Hessian matrix,

$$H(X, D\sigma) = \begin{bmatrix} L_{xx}(X,D\sigma) & L_{xy}(X,D\sigma) \\ L_{xy}(X,D\sigma) & L_{yy}(X,D\sigma) \end{bmatrix} \qquad (5)$$

where Ixx, Ixy and Iyy are the second-order partial derivatives of the image evaluated at the point x and σD is the Gaussian scale of the partial derivatives. Note that both the Hessian matrix and the related second moment matrix have been applied in several other interest operators (e.g., the Harris, Harris-affine, and Hessian-affine detectors) to find image positions where the local image geometry is changing in more than one direction. Likewise, Lowe's maximal difference-of-Gaussian (DoG) detector also uses components of the Hessian matrix (or at least approximates the sum of the diagonal elements) to find points of interest. However, PCBR detector is quite different from these other methods and is complementary to them. Rather than finding extremal "points", this detector applies the watershed algorithm to ridges, valleys, and cliffs of the image principal-curvature

surface to find "regions". As with extreme points, the ridges, valleys, and cliffs can be detected over a range of viewpoints, scales, and appearance changes. The principal curvature image is given by either

$$P(X) = \max(\lambda 1(x), 0) \text{ or}$$

$$P(X) = \min(\lambda 2(x), 0) \quad (6)$$

where $\lambda 1(x)$ and $\lambda 2(x)$ are the maximum and minimum eigenvalues, respectively, of H at x. First provides a high response only for dark lines on a light background (or on the dark side of edges) while the later is used to detect light lines against a darker background. Like SIFT and other detectors, principal curvature images are calculated in scale space.

3) Procedure to calculate PCBR features

First double the size of the original image to produce our initial image, I_{11}, and then produce increasingly Gaussian smoothed images, I_{1j}, with scales of $\sigma = k_j - 1$ where $k = 2^{1/3}$ and $j = 2..6$. This set of images spans the first octave consisting of six images, I_{11} to I_{16}. Image I_{14} is down sampled to half its size to produce image I_{21}, which becomes the first image in the second octave. Apply the same smoothing process to build the second octave, and continue to create a total of $n = \log 2(\min(w, h)) - 3$ octaves, where w and h are the width and height of the doubled image, respectively. Finally, calculate a principal curvature image, P_{ij}, for each smoothed image by computing the maximum eigenvalue of the Hessian matrix at each pixel. For computational efficiency, each smoothed image and its corresponding Hessian image is computed from the previous smoothed image using an incremental Gaussian scale. Given the principal curvature scale space images, calculate the maximum curvature over each set of three consecutive principal curvature images to form the following set of four images in each of the n octaves:

$$MP_{12} \; MP_{13} \; MP_{14} \; MP_{15}$$

$$MP_{22} \; MP_{23} \; MP_{24} \; MP_{25} \quad (7)$$

$$...$$

$$MP_{n2} \; MP_{n3} \; MP_{n4} \; MP_{n5}$$

where $MP_{ij} = \max(P_{ij}-1, P_{ij}, P_{ij}+1)$. MP is created by maximizing the principal curvature at each pixel over three consecutive principal curvature images. From these maximum principal curvature images, the stable regions are found via watershed algorithm.

4) Stable Regions across Scale

Computing the maximum principal curvature image is only one way to achieve stable region detections.
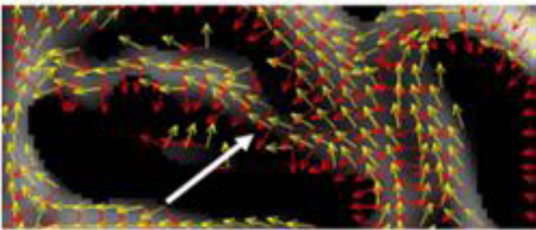

Figure 1. Illustration of how the eigenvector low helps overcome weak principal curvature responses.

To achieve this, compute the overlap error of the detected regions across each triplet of consecutive scales in every octave. The overlap error is calculated. Overlapping regions that are detected at different scales normally exhibit some variation. This variation is valuable for object recognition because it provides multiple descriptions of the same pattern. An object category normally exhibits large within-class variation in the same area. Since detectors have difficulty locating the interest area accurately, rather than attempt to detect the "correct" region and extract a single descriptor vector, it is better to extract multiple descriptors for several overlapping regions, provided that these descriptors are handled properly by the classifier.

5) Textures- Wavelet Packet Decomposition (WPD-2)

The most significant information of a texture often appears in the high frequency channels. Images are decomposed through the Wavelet Packet Decomposition [6] using the Haar basis function up to level two. Gray level co-occurrence matrix is constructed for the coefficient sub bands of the wavelet transform. The Haralick texture features are extracted from the co-occurrence matrix.

Input hand images are decomposed through the Wavelet Packet Decomposition using the Haar (Daubechies 1) basis function to get the four sub band images namely Approximation (A) and three detail coefficients - Horizintal (H), Vertical (V) and the Diagonal (D) as shown in Figure 2. The Haar wavelet transformation is chosen because the resulting wavelet bands are strongly correlated with the orientation elements in the GLCM computation.
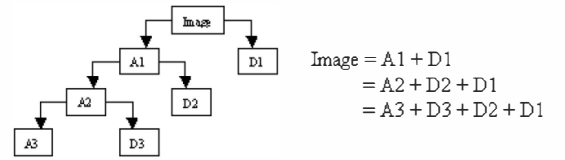

Figure 2. Wavelet Decomposition Tree

The second reason is that the total pixel entries for Haar wavelet transform are always minimum. Through experimentation the Haar basis function up to the level two is found to be best, yielding distinct features and hence Haar basis function up to level two (refer Figure 3) is used in this method. This result in a total of 20 sub bands, four sub bands at the first level and sixteen sub bands (four for each sub band) in the next level.
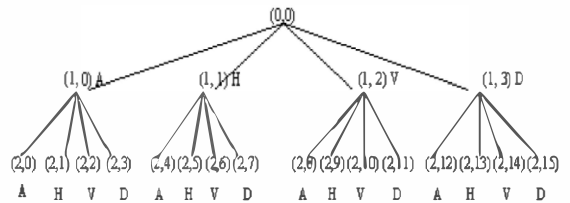

Figure 3. Wavelet Packet Tree upto level -2

It is not necessary to consider all the twenty sub bands for feature extraction. This is because, the four types of sub bands – approximation, horizontal, vertical and diagonal obtained from the wavelet transforms retain specific type of

32

information by filtering out other information. Therefore, when the horizontal, vertical and diagonal sub bands are decomposed further into four bands, all the four sub bands may not be necessarily used as some information in the second level is lost. So, it is necessary to consider only the relevant sub bands at the second level. For example, in the sub band (1, 1) of level one which gives only horizontal detail coefficients, it is sufficient to consider only the approximation and horizontal detail coefficients of its second level. The vertical and diagonal sub bands of (1, 1) are not considered as they exhibit less or poor information. Thus, the sub bands that exhibit the similar type of coefficients from the two levels are selected. In the proposed method, the sub bands of the two levels are combined to form four groups as given below.

Group 1: Approximation sub bands: (1, 0), (2, 0) = (A, AA)

Group 2: Horizontal sub bands: (1, 1), (2, 1), (2, 4), (2, 5) = (H, AH, HA, HH)

Group 3: Vertical sub bands: (1, 2), (2, 2), (2, 8), (2, 10) = (V, AV, VA, VV)

Group 4: Diagonal sub bands: (1, 3), (2, 3), (2, 12), (2, 15) = (D, AD, DA, DD)

Thus, only fourteen sub bands - two approximate sub band, four horizontal sub band, four vertical sub bands and four diagonal sub bands are selected out of the twenty sub bands.

Gray Level Co-occurrence Matrix (GLCM) [6] has been proven to be a very powerful tool for texture image segmentation. Gray-level co-occurrence matrices (GLCMs) are used to represent the pair wise joint statistics of the pixels of an image and have been used for many years as a means of characterizing texture. GLCM is a two dimensional measure of texture, which show how often each gray occurs at a pixel located at a fixed geometric position relative to each other pixel, as a function of its gray level. GLCMs are in general very expensive to compute due to the requirement that the size of each matrix is NxN, where N is the number of gray levels in the image. So, it is necessary to reduce the number of discrete gray levels of the input image in order to obtain co-occurrence matrix of smaller size. So, if the gray levels are divided into fewer ranges, the size of the matrix would be reduced, thus leading to less noisy entries in the matrix.

The gray levels of the quantized sub bands are divided into fewer ranges to obtain a new transformed sub band, which results in reduced size of the co-occurrence matrix. Then, from the new transformed sub bands, GLCMs are constructed using the values d=2, where d represents the linear distance in pixels. The value of θ is fixed based on the type of the sub band. For the approximate sub bands i.e., group1 ((1,0), (2, 0)), GLCMs are constructed with the value θ = {0∘, 45∘, 90∘, 135∘}. The value of θ is taken as 0∘ for horizontal sub bands (group2), 90∘ for vertical sub bands (group3) and, 45∘ and 135∘ for diagonal sub bands (group4). Thus, totally, twenty four GLCM (eight GLCM for group1, four GLCM for group2, four GLCM for group3 and eight GLCM for group4) are constructed. Haralick has proposed the textural features that can be extracted from the cooccurrence matrix. In this paper, Haralick texture

features such as inertia, total energy, entropy, contrast, local homogeneity, cluster shade, cluster prominence, and information measure of correlation (refer Fig) are extracted from the gray level co-occurrence matrices obtained from the coefficients of the sub bands. These features are known as the wavelet packet co-occurrence features.

The eight Haralick texture features are extracted from the twenty four GLCMs resulting in a total of 192 features. In order to reduce the dimension of the features, the mean values of the eight features are computed individually from the GLCMs constructed for each of the fourteen sub bands of the four groups. Hence, eight features of any sub band is obtained by taking the average of the each features computed from each GLCMs of that sub band. For example, construction of four GLCMs for the sub band (1, 0), eight features are computed from these four GLCMs and the average of each feature is computed from the four GLCMs of the sub band (1, 0). Thus, eight features of the sub band (1, 0) and eight features of sub band (2, 0) of group-1, results in 16 texture features. Similarly, eight features from each sub band of group-2 results in 32 texture features; eight features from each sub band of group-3 results in 32 texture features and eight features from each sub band of group-4 results in 16 texture features. Totally, 112 texture features are computed from the GLCMs constructed for the fourteen sub bands of the wavelet packet transforms and hence, these 112 features are called the wavelet packet co-occurrence features.

Thus, the dimensionality of the features has been reduced from 192 to 112 features. As the features are extracted from the two levels of wavelet packet transforms and then the average of the feature values are computed, they could be considered as optimal features.

Inertia	$C_1 = \sum_{i,j=0}^{n} (i-j)^2 C(i,j)$
To al nergy	$C_2 = \sum_{i,j=0}^{n} C^2(i,j)$
Entropy	$C_3 = \sum_{i,j=0}^{n} C(i,j) \log C(i,j)$
Local homogeneity	$C_4 = \sum_{i,j=0}^{n} \frac{1}{1+(i-j)^2} C(i,j)$
Max. Probability	$C_5 = \max_{i,j} C(i,j)$
Cluster shad	$C_6 = \sum_{i,j=0}^{n} (i-M_x+j-M_y)^3 C(i,j)$
Cluster prominence	$C_7 = \sum_{i,j=0}^{n} (i-M_x+j-M_y)^4 C(i,j)$
Information measure of correlation	$C_8 = \frac{(C_3 - H_{xy})}{\max(H_x, H_y)}$

$$\text{Where } M_x = \sum_{i,j=0}^{n} i C(i,j) \text{ and } M_y = \sum_{i,j=0}^{n} j C(i,j)$$

$$S_x(i) = \sum_{j=0}^{n} C(i,j) \text{ and } S_y(i) = \sum_{i=0}^{n} C(i,j)$$

$$H_{xy} = \sum_{i,j=0}^{n} C(i,j) \log(S_x(i) S_y(j))$$

$$H_x = \sum_{i=0}^{n} S_x(i) \log S_x(i) \text{ and}$$

$$H_y = \sum_{j=0}^{n} S_y(j) \log S_y(j)$$

Figure 4. Wavelet Packet Co-Occurrence Features extracted from Co-occurrence Matrix C(i,j).

8)   Finger Detection using Convexity Defects
    In this paper, two hands are considered as a single shape to recognize a particular sign in ISL, the finger count will add up to the accuracy of the classifier. Addition to shape and texture features, finger features are also taken into consideration for

the purpose of ISL Alphabet recognition. From the detected hand in each frame, contour is formed for the hand region, convex hull property is checked and convexity defects are calculated.
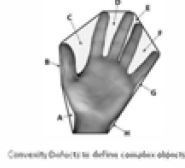

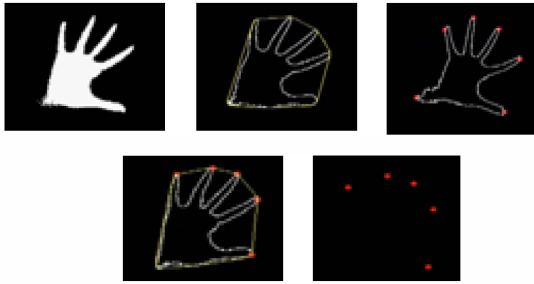Figure5. Convexity Defects for complex objects

Figure6. Finger Detection and Counting

9) Classification using Multi Class SVM

Finally, the generated vector is fed into the multi-class SVM training classifier model that was built in the training stage to classify and recognize the hand gestures. The multi-class SVM classifier implies the concept of one-to-all differentiation to classify each class from all other classes in the database.

Now a set of feature vector given by the PCBR, WPD-2 & finger count for each hand signs (alphabets of ISL) are stored in a database. Group/class ={ a,b,c,d,e,....z} and sample image feature vector is f={x1,....xn}, Number of sample images in the database is represented as Y={y1........ym}.

To train the classifier to clearly distinguish all these images as a particular class (whether A/B/C....Z), the feature vectors are loaded into the multi class SVM module. Here the feature vectors of each image sample in the database are compared and distinguished from all other classes of the group. This concept is referred to as One-to-All matching. In this way the ISL alphabets are recognized.

III. EXPERIMENTAL RESULTS

Experimentation is carried out in a Intel i3 processor machine with 2.27GHz, 2GB RAM, MATLAB R2009a and a 1.2 MP USB camera.

A. Dataset for ISL

There are no resources to download ISL alphabet image dataset. So after a hard effort in getting the dataset from various resources, the Indian Sign Language database has been made in our CMERI lab with the own lighting and environmental setup. There are 23x10=230 training static images and 3x 20= 60 dynamic alphabet videos. Original resolution of the images is 640x480. Those are cropped and normalized into 120 x120 size. The video samples are 320x260 in size and are taken in a lab lighting condition.
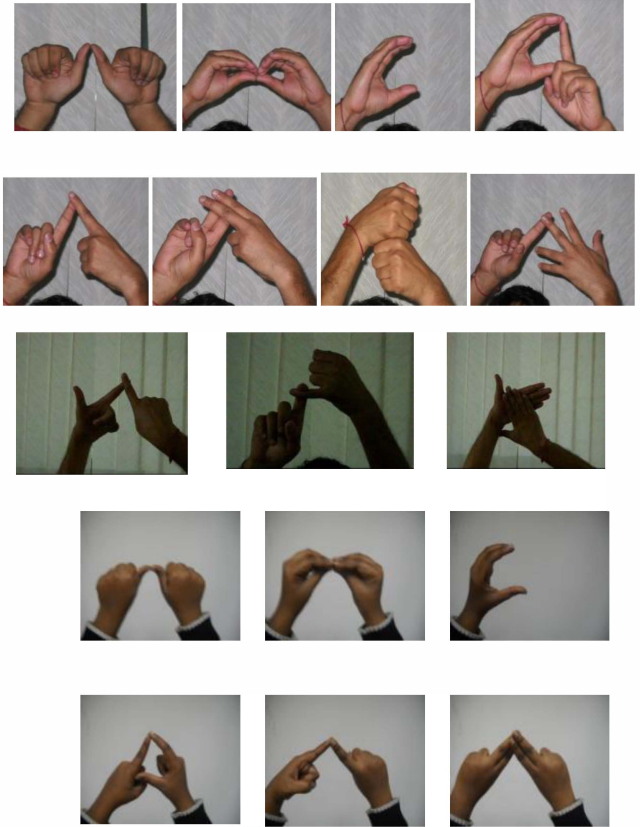

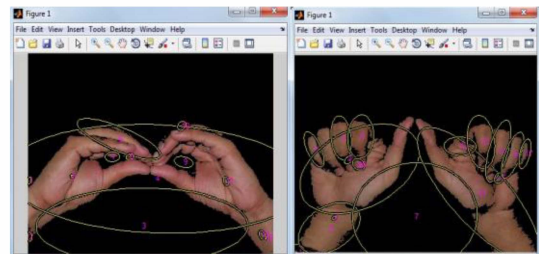Figure 7. ISL Training images made in our lab

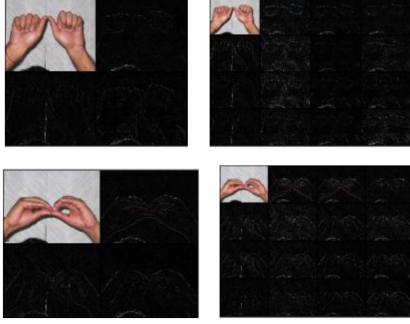Figure 8. PCBR features of ISL alphabet "B" & "A"

Figure 9. Wavelet Decomposed Details Level 1& 2 of ISL Alphabet "A" & "B"

Method	Avg. Processing Time (sec)	Classification Rate using DTW (%) (Dynamic Gestures)	Classifiers & Recognition Rate for posture recognition (%)		
			KNN	SVM	SVM Multi class
PCBR & WPD-2D	67	78.3	83.5	85.1	79.9
PCBR & WPD-2D Feature Vector 48D	48	77	88.7	89.2	82.3
PCBR+ WPD-2D + Finger	55	**77.2**	89.1	91.7	**86.3**

IV. CONCLUSION

This paper proposed an approach to address the inter-class ambiguity issue in ISL alphabet recognition. With the help of local-global finger movement information and shape-texture features, accurate recognition of each ISL signs have been achieved for both static & dynamic gestures. However, due to the less stable nature of PCBR features, the accuracy slips down slightly in case of dynamic gestures of ISL. Our future prospective work concentrates on the analysis of dynamic nature of gestures under different conditions.

ACKNOWLEDGMENT

B. ISL Alphabet Recognition

Using structure and shape features along with the finger count information for ISL Recognition, the following procedure is carried out to conduct the experiment. The sample database images and videos are fed into the respective classifiers namely KNN, SVM and Multi-class SVM for the training phase. After the training phase, the classifier will be learned and familiar with the signs (gestures). Now testing phase is conducted with the new input image or video which is containing the hand sign or gestures to detect and recognize. For training the classifiers,
No. of Training samples:
- Static:23x40=920 [a,b,c,d,e,f,g,i,k,l,m,n,o,p,q,r,s,t,u,v,w,x,z]
- Dynamic: 3x 22 =66 [h,j,y]

No. of Test Samples:
- 22 videos (length: 46secs)
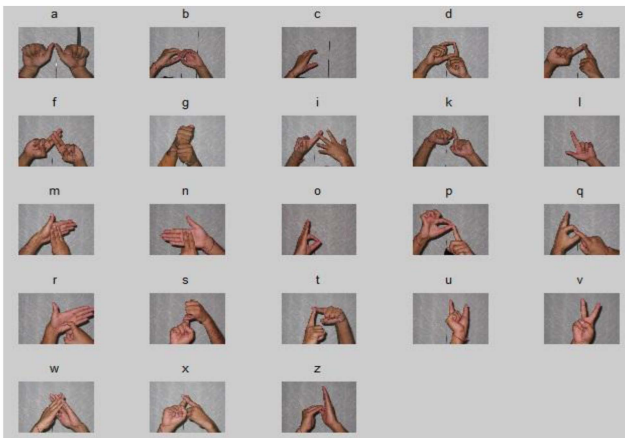- Resolution: 320x240, 640x480, Colour: RGB ,Frame Rate for processing: 4

Figure 10. Training images of ISL

REFERENCES

[1] Nandy,S. Mondal, J.S.Prasad, P.Chakraborty & G.C.Nandi, "Recognizing & interpreting Indian Sign Language gesture for Human Robot Interaction", IEEE International Conference on Computer and Communication Technology (ICCCT), Page(s): 712-717, 2010.

[2] P.Subha Rajam, G. Balakrishnan, "Indian sign language recognition system to aid deaf-dumb people", IEEE International Conference on Computing Communication and Networking Technologies (ICCCNT), Page(s): 1-9, Trichy, India, 2010.

[3] K. Anuja, S. Suryapriya, S.M. Idicula, "Design And Development Of A Frame Based Mt System For English-To-Isl", Ieee World Congress On Nature & Biologically Inspired Computing Nabic, Page(S): 1382 - 1387, Cochin, India, 2009.

[4] B. Lekhashri, A. Arun Pratap, "Use of motion-print in sign language recognition" IEEE National Conference on Innovations in Emerging Technology (NCOIET), Page(s): 99-102, Trichy, India, 2011.

[5] R. Pravin Futane, Rajiv V. Dharaskar, "Hasta Mudra": An interpretation of Indian sign hand gestures" 2011 IEEE 3rd International Conference on Electronics Computer Technology (ICECT), Vol:2, Page(s): 377-380, Kanyakumari, India, 2011.

[6] M.C. Padma & P.A. vijaya, "Global Approach For Script Identification Using Wavelet Packet Based Features", International Journal of Signal Processing, Image Processing And Pattern Recognition, Vol.3, 2010.

[7] H. Deng, W. Zhang, E. Mortensen, T. Dietterich & L. Shapiro, "Principal Curvature- Based Region Detector for Object Recognition", In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition CVPR'07, Pages(s): 1-8, 2007.