International Journal of Engineering & Technology, 7 (1.1) (2018) 539-547

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Sequential particle filter with covariance features classified with artificial neural nets for continuous Indian sign language recognition

P. Praveen Kumar[1]*, P.V.G.D. Prasad Reddy[1], P. Srinivasa Rao[1]

[1]Department of Computer Science & Systems Engineering, College of Engineering Andhra University, Visakhapatnam, India.
**Corresponding author E-mail: pkpinjala.auce@gmail.com*

Abstract

Machine translation of sign language is a complex and challenging problem in computer vision research. In this work, we propose to handle issues such as hands tracking, feature representation and classification for efficient interpretation of sign language from isolated sign videos. Hands tracking is attempted in a sequential format with one hand after the other by nullifying the effects of head movement using serial particle filter. The estimated hand positions in the video sequence are used to extract the hand portions to create a feature covariance matrix. This matrix is a compact representation of the hand features representing a sign. Adaptability of the feature covariance matrix is explored in developing relationships with new signs without creating a new feature matrix for individual signs. The extracted features are then applied to a neural network classifier which is trained with error backpropagation algorithm. Multiple experiments were conducted on a 181 class signs with 50 sentence formations with 5 different signers. Experimental results show the proposed sequential hand tracking is closer to ground truth. The proposed covariance features resulted in a classification accuracy of 89.34% with the neural network classifier.

Keywords: *Sequential particle filter tracker; Indian sign language recognition; Covariance features; Artificial neural networks.*

1. Introduction

Sign language recognition is a computer vision based intact intricate language that engages signs shaped by hand moments in amalgamation with facial expressions and postures. It maps speech communication to human signs and gestures enabling hearing impaired people to communicate with normal people. Dynamic hand movements are involved in gestures and they form signs such as numbers, alphabets and sentences. Classification of gestures can be identified as both static and dynamic. Static gestures involve a time invariant finger orientation whereas dynamic gestures support a time varying hand orientations and head positions.

Basically, Sign language is used by the hearing-impaired people for their communication. Sign language recognition systems forms a bridge between normal people and hearing-impaired people. Sign Language Recognition System (SLR) translates sign gestures into voice or text commands and vice versa primarily to assist deaf person or hearing-impaired person to communicate with the normal person. A normal person communicates with spoken language where as a deaf person uses sign language for communication.

A sign language recognition system is based on the five attributes of a human signer. They are hand and head recognition, hand and head orientation, hand movement, shape of hand and location of hand and head (depends up on back ground). Among the five parameters there are two parameters which are most important and they are hand and head orientation and hand movement in a direction.

There are two types of signs namely stationary and non-stationary signs. Stationary signs are the signs in which the movement of hand in describing the sign is one time as shown in the form of video frames in Fig.1. (a). Whereas, non-stationary signs are the signs in which the movement of hand is more than one time as shown in Fig. 1. (b).

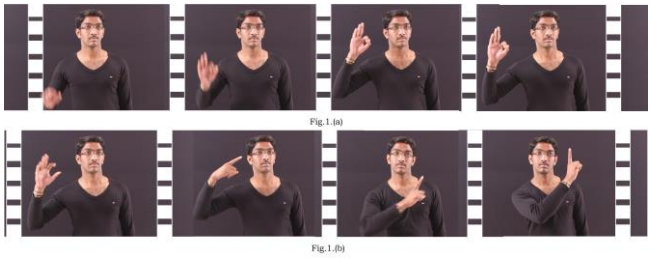

Fig. 1: (a) Stationary sign frames of sign "Hello", (b) Non- stationary sign frames for a sign "Good Morning".

Logically sign language understanding consists of linguistic analysis of hands tracking, hands shapes, hands orientations, sign verbalization, head movements and facial expressions. Sign language is in many ways different from spoken language such as facial and hand terminology, references in virtual signing space, and grammatical differences as explained.

In this work, we propose to identify hand movements in a sign sequentially using particle filter tracker and use covariance matrix to generate features of the tracked hands. A feature covariance matrix is built for both hands independently with the head removed. The feature covariance matrix is labelled for each sign class for training with artificial neural networks. This work pro-

jects the use of independent hands to represent sign. Performance of the proposed approach with other sign language recognition models using ANN classifiers is tested based on percentage recognition.

2. Literature Review

This part of the work discusses algorithms that are state of the art in addressing the problems in automatic sign language recognition [1]. However, the developed state of the art algorithms points to problems related to complexity of the algorithm, execution times and real-time operations. The objective of this chapter is to introduce the gaps formed in designing a smart sign language machine translator with the perspective of providing a suitable solution through this work. Sign language recognition has been a challenging area in multiple scales [2].

A sign language space can be obtained with different entities such as humans or objects stored in it around a 3Dimensional body cantered space of the signer. These entities are located with certain locations and later referenced it by pointing to space. To define a model for spatial information, containing entities is another challenge faced by researchers [3]. Cyberglove and a Flock of Birds 3-D motion tracker extract gesture features of 50 ASL words [4]. Finger joint angles define hand shapes while the motion tracker provides trajectory features. ANN model is trained and tested with the mixed set of shape and trajectory features. This work achieved a recognition rate of 90% [5]. Neelesh Sarawate et al. [6] developed American Sign Language (ASL) word recognition system supported neural networks and a probabilistic model is conferred. A Cyber Glove and Flock of Birds area unit won't to track bending fingers, hand position, and hand orientation. A probabilistic model supported the Markov chains and HMM is employed to method the outputs of the feature vectors. The system has accuracy of 95.4% over a vocabulary of 40 ASL words.

The challenging problems in 2D SLR are hand tracking, occlusions on hands and face, background lighting, changing signer backgrounds and camera sensor dynamics. Some of these challenges are extremely difficult to deal with in the 2D environment.

P.V.V.Kishore et al. [7] projected a skeleton of isolated video based Indian sign language (ISL) identification system with computational intelligence and image processing methodologies are integrated. Wavelet based video segmentation procedures are adopted for detection of hand shapes and head positions. Elliptical Fourier descriptors achieves unique feature vectors for each individual gesture and linear output membership function based Sugeno fuzzy inference system recognizes these gestures at a rate of 96% for a 80 word and 10 sentence trained system.

P.V.V.Kishore et al. [8] recommended an efficient theory of segmentation for SLR in which background in the sign videos is assumed to be non-static. Profuse attributes of images such as color, texture, and boundary are fused for the level sets energy function detraction. Colour frames are extracted and utilized as per the background requirement of the video frames. Spatial information is provided by edge mapping and boundary features are obtained through edge map of image. Gesture shapes are enumerated dynamically and are assumed to be adaptive in the entire process of segmentation of video frames.

Tzuu-Hseng et al. [9] endorsed a system for the realization of gestures using ABC based Markov Model. Filtered data from AHRS sensors undergo principle component analysis for redundant data elimination and feature vector acquisition. ABC-HMM serves the purpose of classifier with the number of hidden states obtained from k means algorithm. This work also provides a comparison of achieved recognition rate with various classifiers related to Markova models and is found to present encouraging recognition rates compared to other classifiers.

Li Chun wang et al. [10] embarked a model which could measure the closeness of video in Chinese sign language. This model considers sign semantics which defines the hand shape, location, orientation and vision component which is distance based on VLBP. The experimental results of this model proved effective assessment of measuring similarity. More research related material on 2D models and the corresponding research challenges can be found in [11]- [14]. The other challenge for researchers lies in converting the detected signs into meaningful sentences [15].

Kinect sensor captures 3D depth images which are sometimes combined with RGB color video data to form an RGB-D video images. 3D sign language is explored to an extent with these sensors in the recent times. Almeida et al. [16], RGB-D (red, green, blue and depth) images of signers to extract seven different features of Brazilian sign language. The extracted multiple features are classified with multi class Support vector machine classifier. This approach registered a recognition rate of 80% on a set of 10,000 sign frames.

Shao-ZiLi et al. [17], developed a feature learning approach based on sparse auto-encoder (SAE) and principle component analysis for sign language with RGB-D inputs. The features from RGB and D images are learned using a convolutional neural network and then the extracted learned features are fused using principle component analysis. Experimental results of ASL recorded a recognition rate in the range 75 to 99.2%.

The 3D data from Kinect sensor consists of hand trajectories [18], orientations and velocities [19] from a single depth image. Features such as 3D body joint locations [20] and Fingers earth movers distance (FEMD) [21] are used for sign classification.

This work is the extension of our previous model [22], which uses various shape features to represent a continuous sign video sequence with Adaboost classifier. However, multi feature fusion on transformed features has resulted in a large noise component around the signer which reduced recognition rate for large length sentences. This component is addressed in this work by using sequential hand tracking using particle filter [23], which tracks hands separately one after the other in case of two hand signs. Visual hand tracking has been attempted in the past using optical flow, kalman filter and extended kalman filter [24]. However, these models performed poorly due to the non-consideration of local hand information in multiple object tracking. Large movements are exceptionally well tracked in a multi-object environment, whereas for slower movements or objects moving with various speeds, the performance of these algorithms is not well established.

Hence in this work, we propose to track each hand separately as proposed in [26]. The estimated hand portions are extracted and a feature covariance matrix is constructed from the parameterized hand sub image frames in the entire video sequence. The features are flattened to create a unique feature vector representing a sign class. For each sign the length of the video sequence is curtailed to 4 seconds or 120 frames. Finally, each class vector is called the training set for the ANN. Different test subjects are used to train and test the proposed ANN model.

The rest of the paper is organized as: section 3 deals with methodology, results and discussion is in section 4 and section 5 concludes the work.

3. Continuous Sign Language Recognition

This section provides an elaborated discussion on the procedures of the proposed method for Indian sign language recognition. The block diagram of the proposed method is shown in fig.2.

3.1. Hand Identification

The focus of this work is to classify signs based on hand movements in a continuous sign video sequence. To isolate the hand movements from the other noisy movements attached to the human body such as head, we first remove the head portion of the signer. For this, Vola and Jones [27], face detection algorithm is applied on the raw sign sequence to extract face region of the signer. The extracted face is then removed from the actual sign video during hand identification for accurate hand tracking.

For hand segmentation, the face masked frames are consecutively subtracted using background subtraction algorithm [28]. Since the background in our video sequences is uniform, we propose a simple reference frame subtraction model. In this model, the first frame without hand movements is taken as a reference frame and the remaining frames are subtracted from the reference frame. The moving hands are obtained by setting a threshold on the subtracted frames. The threshold is generated using the maximum change in the subtracted frame.

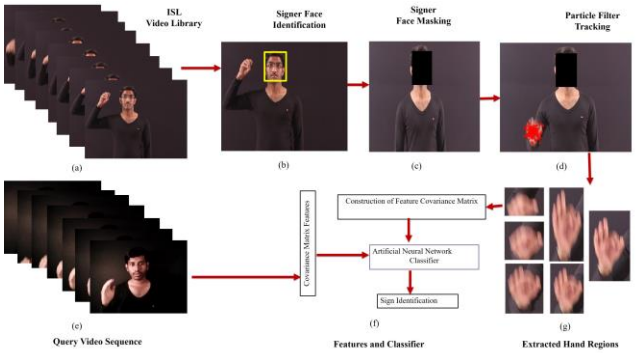

Fig. 2: Block diagram of the Proposed Model for Continuous sign language recognition.

The above process of face detection and hand extraction through background subtraction is illustrated in fig.3.

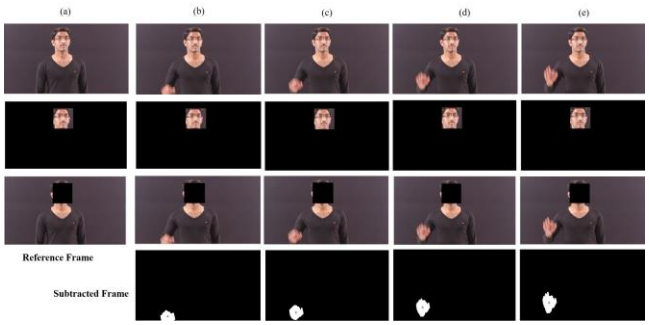

Fig. 3: (a) Reference background frame, (b) Frame 22, (c) Frame 24, (d) Frame 26, (f) Frame 28 and each column has identified extracted face, masked face and frame subtracted for hand identification.

3.2. Sequential particle filter

The primary objective of particle filter is tracking objects in a time series data provided the previous time step and the observable data at the pervious time step [26]. Particle filter can successfully track both hands or single hand in a sign language video sequence. However, to reduce ambiguity in tracking due to hand identification problem, the works in [26] propose to track individual hands independently in a serial fashion. In this work, we track left and right hands one after the other, whichever hand is detected or identified first.

Let the observation vector at any time instance or at any frame occurrence in a video sequence t is modelled as $x_t = \left[x_t, y_t, d_x, d_y, v_t^x, v_t^y\right]$, where x_t is a state vector. The variables (x_t, y_t) are the position vectors of the moving object with velocities in the directions (v_t^x, v_t^y). The variables (d_x, d_y) are hand sizes in terms of width and height of the hand in the frame defined along x and y dimensions. The dynamic particle filter model is formulated as

$$x_t = \begin{bmatrix} x_t \\ y_t \\ d_x \\ d_y \\ v_t^x \\ v_t^y \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ d_{t-1}^x \\ d_{t-1}^y \\ v_{t-1}^x \\ v_{t-1}^y \end{bmatrix} + \begin{bmatrix} n_x \\ n_y \\ n_{d^x} \\ n_{d^y} \\ n_{v^x} \\ n_{v^y} \end{bmatrix} \quad (1)$$

Where $n_x, n_y \Rightarrow N\left(0, \sigma_{xy}^2\right)$, $n_{d^x}, n_{d^y} \Rightarrow N\left(0, \sigma_{d^x d^y}^2\right)$ and $n_{v^x}, n_{v^y} \Rightarrow N\left(0, \sigma_{v^x v^y}^2\right)$ are zero mean noise terms with displacement variance σ_{xy}^2, hand dimension variance $\sigma_{d^x d^y}^2$ and velocity variance $\sigma_{v^x v^y}^2$ respectively. The hand dimension is generally ignored by most of the researchers due to the fixed focal distance of the camera on the signer. However, in sign language as shown in fig.3, the hand dimension changes constantly in majority of the frames. This poses a problem in detection of hand in the later stages. Moreover, the researchers in [26], normalized these hand variations for computing the covariance matrix. In this work, we use the same dimension of the hand extract and use it as a parameter for particle filter, where it was ignored by many previous works on sign language hand tracking.

After background subtraction, as shown in fig.3 last row, the hand intensities are used as an input to the particle filter observation matrix. From the hand intensity values in each frame, we compute the centre of the intensity hand region as shown in fig.3. The centres of the hand region on normalized pixel intensities z_t is computed as

$$z_t = \frac{\sum_{i \in R_{Hand}} I(x_i, y_i)}{A} \quad (2)$$

Where A is the area of the hand object $d_x \times d_y$. The hand centres are calculated as

$$x_t = \frac{\sum_{i \in R_{Hand}} x_i A_i}{\sum_i A_i} \quad (3)$$

$$y_t = \frac{\sum_{i \in R_{Hand}} y_i A_i}{\sum_i A_i} \quad (4)$$

The first frame of the signed video sequence is used as reference frame for all video sequences. This gives the initial state of the particles which are used to find the next state of the particles in all frames. In the subsequent stages, all particles are assigned in four different stages, namely, prediction, weight updation, estimation and resampling. In the prediction stage, the state x_t of the n^{th} particle is calculated based on the previous observed state x_{t-1}. The corresponding particle weight z_t is determined by the observation model. The final estimated state is computed as

$$x_t = \sum_{n=1}^{N} x_t^n z_t^n \tag{5}$$

The weights are re-sampled, so that equal weights can be re-assigned to the new particles. In this work this is a two-pass algorithm. In the first pass, right hand is tracked and in the second pass left hand is tracked. After the first hand is tracked and locations are estimated, the right-hand portion is masked and the other hand is tracked. Fig.4. shows the two-hand sequential particle filter on a sign video sequence.

Fig. 4: Sequential particle filter (a) Masked left hand for right hand tracking, (b) Tracked right hand, (c) Masked left hand for left hand tracking and (d) Tracked left hand.

3.3. Covariance Matrix features

In [26], feature covariance matrix is constructed using the hand cantered squared region for every frame t. In this work, the hand region is not normalized and the hand regions are used as extracted from the original frame. The dimensions are not squared in this our case. The computation of covariance matrix is achieved on features extracted from the hand regions. The first feature is the location of the hand in the video frames, which is estimated from the state vector of the particle filter (x_t, y_t). The location centred hand neighbourhood pixel intensities are represented as the 2nd feature, $I_h(x,y)$. These are the pixel values in the region over green colored bounding box on the hand in fig.4(b) & (d).

The other two features are computed by 1st order gradient vector and 2nd order gradient vector computed on the extracted hand regions $I_h(x,y)$ as

$$\nabla I_h(x,y) = \frac{\partial I_h}{\partial x} + \frac{\partial I_h}{\partial y} \tag{6}$$

$$\nabla^2 I_h(x,y) = \frac{\partial^2 I_h}{\partial x^2} + \frac{\partial^2 I_h}{\partial y^2} \tag{7}$$

The final feature matrix for a sign frame t is obtained as

$$F_v^t = \left\{ I_h(x,y), x_t, y_t, \nabla I_h(x,y), \nabla^2 I_h(x,y) \right\} \tag{8}$$

Each object in the feature matrix F_v^t is a different sized cell arrays. Computing covariance on this vector is not possible when all frame is considered at once. The idea of covariance matrix from the generated feature vector is to reduce the dimensionality of the feature matrix for a T frame video sequence. The covariance matrix in our algorithm is computed on a frame's feature vector to create a sparse coded 5×1 vector feature covariance matrix. The 5×1 feature covariance matrix for a frame t is computed as

$$C_f^t = \sum_{(x,y)\in I_h} \left(F_v^t(x,y) - m_v^t\right)\left(F_v^t(x,y) - m_v^t\right)^T \tag{9}$$

where m_v^t is the mean of the features. Instead of averaging on all covariance matrices as in [26], we rearrange the covariance values in the form of a vector representing the entire sign video sequence. The final covariance feature vector per video sign class is

$$C_S^F = \begin{bmatrix} C_f^1 \\ C_f^2 \\ . \\ . \\ C_f^T \end{bmatrix}_{1\times 5T} \tag{10}$$

The 5T, where T is number per video frames per sign class, represents 5 covariance values per frame multiplied by total number of frames. The covariance values are then normalised with the maximum value in the vector for feature value normalization. Similar features are generated for each sign class in the database, which are then used as training samples for the artificial neural network. Uniqueness of these features for four sign class as a box plot is shown in fig.5.

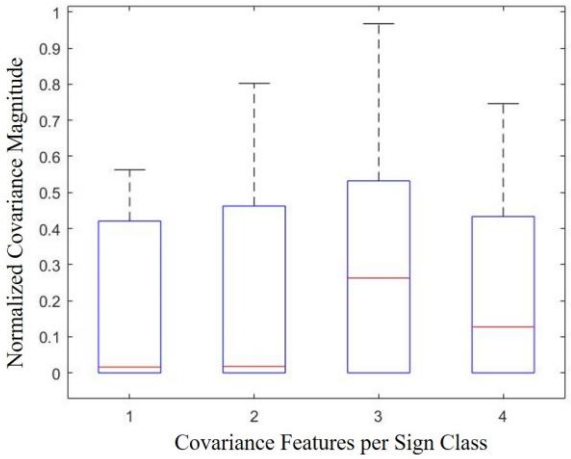

Fig. 5: Uniqueness of Covariance Features per sign

3.4. Artificial Neural Network for Sign Classification

These unique sets of features are used to train the Artificial Neural Network that learns by updating its weights from error calculated by difference between actual outputs and predicted outputs. The output of the neural network will drive a text based class label to corresponding sign. The output of ANN is a probability score that points to the maximum possibility class in the 181 class labels. ANN is trained using error back propagation algorithm with gradient descent learning model.

Multiple types of ANN's are tested in this work ranging from small training samples to large training samples with single and multiple hidden layers. The multiple hidden layer ANN is called deep ANN. Cross subject training and testing is experimented. One of the few systems that can handle our large data matrices is Feed-Forward Artificial Neural Network (ANN). The dimensionally reduced feature matrix is given as input for training the feed-forward neural network shown in Fig.6.

Generally, a Feed-Forward neural network is a combination of three layers of neurons: input layer, hidden layer and output layer. The neurons in these layers are activated by using a nonlinear sigmoid activation function [29]. Let $x_{i,j}(itr)$, where $1 \le i \le N$ and $1 \le j \le M$, be the input to the neural network derived from feature matrix. Where M and N denote the number of columns and rows of feature matrix. itr is the number of iterations called Epochs in neural network terminology. The neural network outputs are denoted by $y_{i,j}(itr)$ where $1 \le i \le N$ and $1 \le j \le M$.

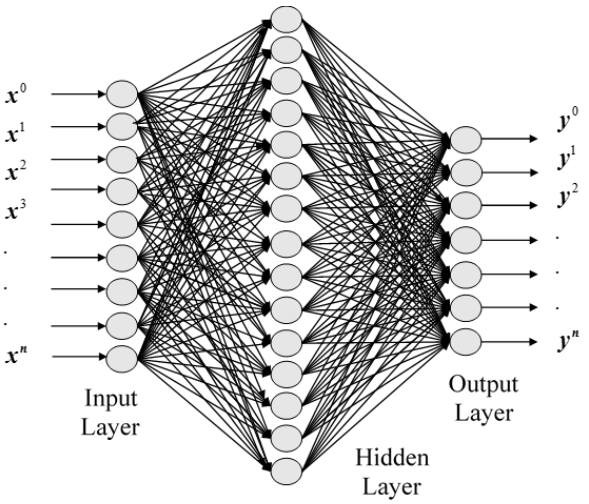

Fig. 6: ANN used for training and testing the Sign classifier.

The back-propagation algorithm follows the following steps in determining the output as discussed in [29]. The output of j^{th} unit in the first hidden layer is calculated from the values in the hidden layer as

$$y_j(itr) = f\left(\sum_{i=1}^{M} x_i(itr) w_{ij}(itr) + b_i\right) \tag{11}$$

where M is number of neurons in the hidden layer, w_{ij} is the weight vector connecting i^{th} unit in between hidden layer and input layer and j^{th} unit in between hidden layer and output layer. b_i is the bias value of j^{th} unit in hidden layer. $f(\square)$ is the activation function which we choose as sigmoid function define as

$$f(\square) = \frac{1}{1 + e^{-x_{ij}(itr)}} \tag{12}$$

The output layer output is calculated as

$$y_k(itr) = f\left(\sum_{j=1}^{M} x_{jk}(itr) w_{jk}(itr) + b_k\right) \tag{13}$$

where M is the number of neurons in the j^{th} layer for k^{th} neuron.

The covariance feature for each sign class becomes the input to the network and the targeted outputs are class labels represented as a binary matrix uniquely representing each class. Initial weight matrix is randomly chosen for each test. Therefor multiple trains of the network were needed to get the correct estimate. The target matrix is binary having the same size as the input matrix. Learning rate and momentum factor were 0.2 and 0.9 respectively. Error tolerance for all training and testing phase is set at 0.01.
The next section presents the results of the experimentation on 50 sign sentences from a 181-class word database of Indian sign language.

4. Results and Discussion

This section presents the experiments of the proposed method and database on which the method is being tested. We present the results of the proposed model against some of the sign language recognition models using ANN classifier with recognition rate as the performance measure. The percentage recognition is computed as

$$R_{\%} = \frac{True\_Positive}{True\_Positive + False\_Positive} \times 100 \tag{14}$$

4.1. The Database

We consider the Indian sign language database used in [14]. It has 181 words created with 5 test subjects under laboratory conditions. The background is simple and the video noise is almost eliminated during recording. The motion blur caused by the camera is kept minimum. The words are carefully crafted for regular usage, such as, good, morning, evening, how, you, woman, mother, men, shop, sport etc. From these words we have organized them in to 50 sentences for 10 to 12 words each or a combination of sentences. The videos are recoded at 30 frames per second with a resolution of 360×640 dimensions. Four camera angles in full color mode are used for capturing the signs for multi view representation. However, we use only the front view data for this experimentation.

4.2. Tracker Performance

Sequential particle filter tracker (SPFT) used in this work is estimating individual hand positional values based on the input video sequence. To ascertain the performance of the sequential particle tracker, we compared our tracking results with actual positional values computed using the ground truth model. The SPFT outputs are compared with optical flow based hand tracking [14], skin color and differential blob based in [30] and kalman filter based tracker in [31]. The visual results are compared against the ground truth on a set of frames of a sign language video sequence in fig.7.

4.3. ANN Classifier with same train and test vectors – Single Subject

The first training batch is a fifty 12-sign sentences from a train subject having 4 instances of the same sentence, which is named as ANN_1. The normalized feature covariance matrix has 3250 vectors in the feature matrix. The input layer has 3250 neurons, 2178 hidden neurons and 12 class labels as output neurons. The network is shown in fig.8.

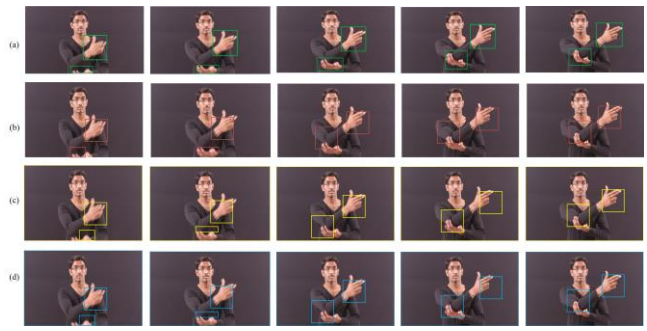

Fig. 7: (a) Proposed SPFT, (b) optical flow in [14], (c) Kalman filter in [30] and (d) Skin blob differentiator in [31].

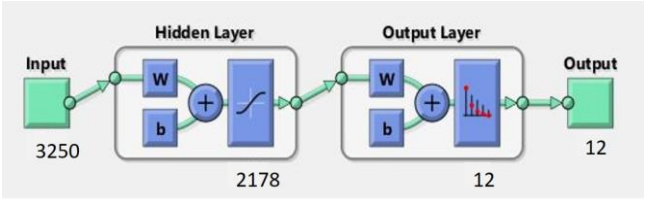

Fig. 8: ANN_1 Classifier.

The network is trained and tested with the same subject on 50 sentences of each 12-word length. For example, “Hi Good Morning, Going to Watch Movie, Without Helping Mother and Father” is a 12- word sentence used for testing. Fourteen such sentences were crafted out of 181 – word dataset. The confusion matrix as shown in fig.9. for the same train and test data on the proposed ANN resulted in a recognition rate of 98.1%. For all the 50 sentences, the average recognition for same training and test data has resulted in an average recognition rate of 96.44%. The average training epochs are 18569 for the 181-word training data.

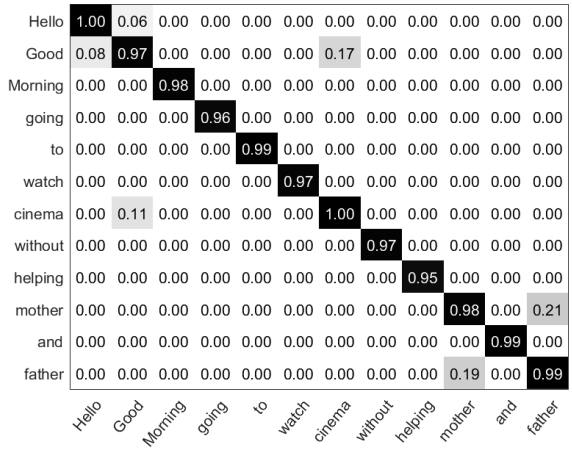

Fig. 9: Confusion matrix for same test and train data for a 12-word sign sentence.

4.4. ANN Classifier with different train and test vectors – 2 Subjects

In this experiment, we use the same trained ANN from the previous experiment and test the ANN_1 with a different test subject. The confusion matrix for this experiment is shown in fig.10. The recognition for a 12 – word sentence is shown in fig.10, which resulted in a recognition rate of 82.3%. Across all 50 sentences, the recognition rate dipped by 6% and recorded as 76.5%. This is due to overfitting of the data to the model during the training phase of the network.

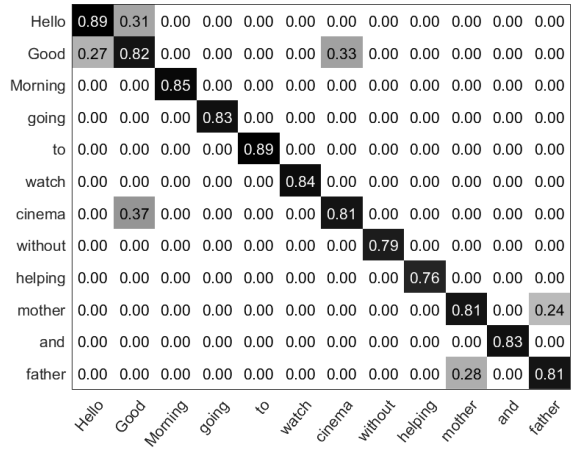

Fig. 10: Confusion matrix for different train and test subject.

The recognition rate decrease is attributed to linearity in network processed output data in the hidden layers. To increase the non-linearity in the output vectors of hidden layer, more versatile data samples per sign are required for training. In the next experiment, we re-train the ANN_1 with a feature covariance matrix generated from another subject.

4.5. ANN Classifier with different two sample train and two sample test vectors – 4 Subjects

This experiment shows the performance of the classifier for multiple training samples for the 181 – word classes. The results for the same 12 – word sentence for this experiment is shown in fig.11.

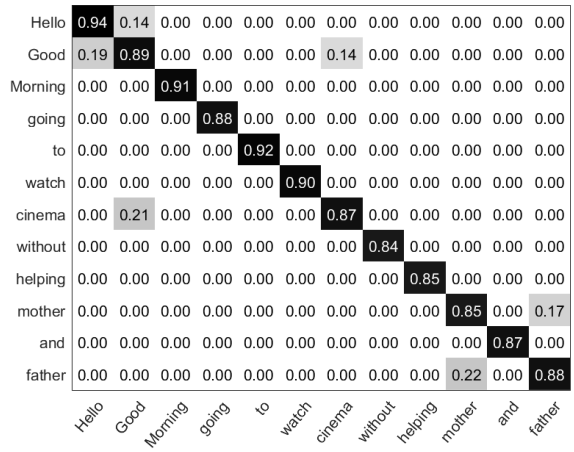

Fig. 11: Confusion matrix for two sample training and two different subject testing.

The average recognition for the 12 – word sentence is around 88.2%. For the 50 sentences, it was recorded as 86.89%. However, the number of epochs has nearly doubled to 36569, as the number of samples doubled in this experiment. All other parameters of the network are kept constant from the previous experiments.

4.6. ANN Classifier with different 3 sample train and 2 sample test vectors – 5 Subjects

Further, we re-trained the network with 3 samples per sign and tested with 2 unseen samples. Fig.12. gives the confusion matrix for this experiment. The average recognition has reached 90.25% for 50-word sentences. The increase in recognition rate is due to the large training data shown to the network for learning versatile features from the normalized covariance features of multiple subjects.

To validate the proposed model, we used two attributes in the form of features and the classifier. For feature vector validation we use multiple feature extraction models reported in literature for training with ANN classifier. In the second validation, multiple classifiers are used on the proposed feature matrix to check the performance of the ANN classifier against popular state-of-the-art classifier models. All classifiers are tested in cross subject mode, where the classifier is trained with one subject data and tested with another subject data.

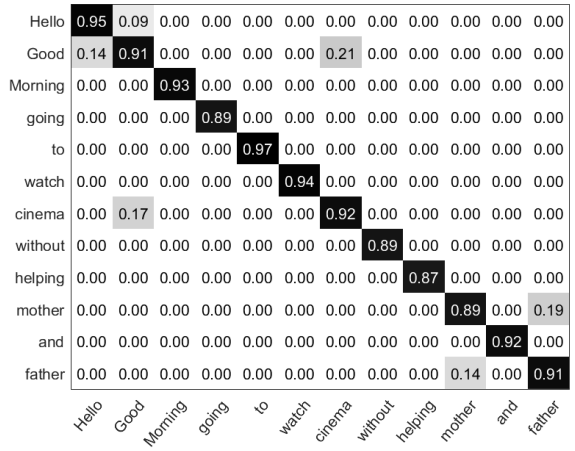

Fig. 12: Confusion matrix for 3 sample training and 2 sample testing confusion matrix for 12 – word sentence.

4.7. Feature vector validation

Table-1 reports the multiple types of classifiers used to represent 2D sign language data trained with ANN classifiers. The validation is checked based on the recognition rate computed from true positive rate (TPR) and false positive rate (FPR). TPR is the actual number of samples classified as a class as trained, whereas FPR defines the number of samples classified as a differently trained sample when a sample is shown to the network. All the checks are made using same subject and cross subject validation.

The region of convergence (ROC) curves was plotted in fig.13 for all the 50 sign sentences used in our work for the following features classified with ANN. The parameters of the ANN were modified according the input vectors used during training and testing. ROC curves point to the extent the feature vector is useful in maintaining inter class variations in the closely related signs.

Table 1: Validating the proposed features against the previous works with ANN.

Features+ANN	**Same Subject**	**Cross Subject**
Performance Measures	**R%**	**R%**
DCT+ANN[7]	88.02	75.23
Wavelet+ANN[34]	90.25	78.96
LBP+ANN[22]	92.36	81.03
SIFT+ANN [33]	84.26	72.32
HOG+ANN [35]	92.96	80.25
Fourier Descriptors + ANN[12]	86.23	81.58
Optical flow+ANN[14]	87.36	74.23
DTW+Fourier Descriptors+ANN[32]	94.63	82.96
Proposed Covariance Features+ANN	**96.44**	**90.25**

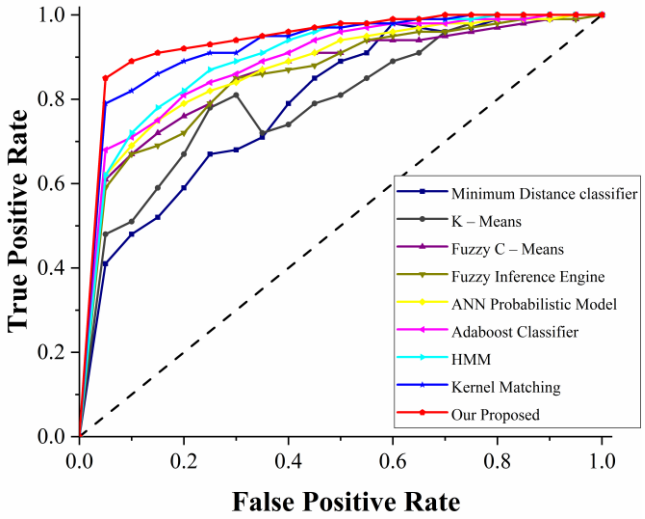

Fig. 13: Region of Convergence Curves for Feature Validation.

4.8. Classifier validation

In this final part, different classifiers are validated for the computed feature matrix. We used eight classifiers from literature on our feature vector for training and testing. The comparisons are tabulated in table - 2.

Table 2: Validating the proposed ANN Classifier against the state – of – the – art.

Classifiers Used for SLR	**Same Subject**	**Cross Subject**
Performance Measures	**R%**	**R%**
Fuzzy Inference Engine [14]	91.02	85.23
Minimum Distance classifier [13]	92.25	78.96
Adaboost Classifier [22]	92.16	88.03
HMM [36]	94.26	89.32
K – Means [9]	89.96	75.25
ANN Probabilistic Model [6]	90.23	87.58
Fuzzy C – Means [37]	92.36	84.23
Kernel Matching [38]	95.63	85.96
Our Proposed Covariance Features + ANN	**96.44**	**90.25**

The table clearly points to the superiority of the proposed model against the other popular classifiers in literature. The ROC curves on 50 sign sentences is plotted in fig.14 for the classifiers in table-2. The ROC curves show the ability of the classifiers to classify a

test sign accurately in its targeted class label. The curves should align sharply towards the true positive axis for better accuracy. This means that the signs are classified as labelled originally by the classifier.

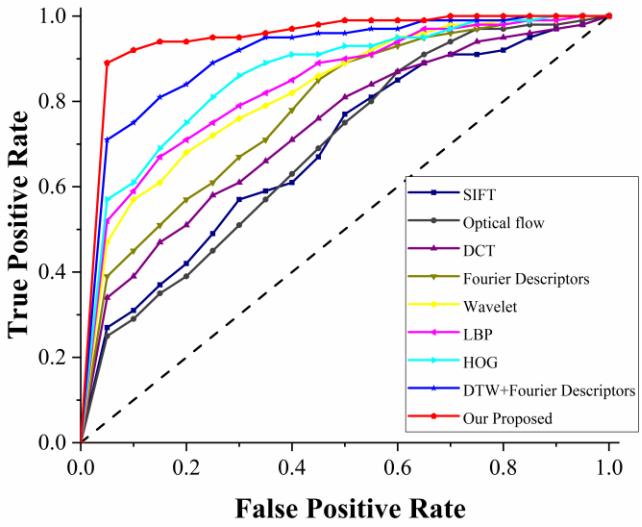

Fig. 14: Region of Convergence Curves for classifier performance.

5. Conclusion

The proposed work gives an insight into an artificial neural network based sign language recognizer with normalized covariance features as input training vector. Sequential hand tracking of both hands is accomplished using sequential particle filter. Each hand is tracked separately using the particle filter by masking head and other hand during tracking phase. The tracked position vectors are used to extract the hand regions in each frame. From the extract hand sub images in each frame, five features are computed on them to create a bulk feature matrix. Covariance is computed with normalization to create a compact size feature vector per frame. For a sign video, the covariance matrices of all frames are concatenated serially to construct a feature vector. The feature vector for all signs is used to provide training to an artificial neural network. The classifier network is tested exclusively on 50 sentences of 12 – words each of Indian sign language. The proposed classifier model recorded a recognition rate of 90.25%, which is better than most classifiers in literature.

References

[1] Ong, Sylvie CW, and Surendra Ranganath. "Automatic sign language analysis: A survey and the future beyond lexical meaning." IEEE Transactions on Pattern Analysis & Machine Intelligence 6 (2005): 873-891.

[2] Liang, Rung-Huei, and Ming Ouhyoung. "A real-time continuous gesture recognition system for sign language." In Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on, pp. 558-567. IEEE, 1998.

[3] Mitra, Sushmita, and Tinku Acharya. "Gesture recognition: A survey." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37, no. 3 (2007): 311-324.

[4] Oz, Cemil, and Ming C. Leu. "Recognition of finger spelling of American sign language with artificial neural network using position/orientation sensors and data glove." In International Symposium on Neural Networks, pp. 157-164. Springer, Berlin, Heidelberg, 2005.

[5] Oz, Cemil, and Ming C. Leu. "American Sign Language word recognition with a sensory glove using artificial neural networks." Engineering Applications of Artificial Intelligence 24, no. 7 (2011): 1204-1213.

[6] Neelesh SARAWATE, Ming Chan LEU, CemilOZ "A real-time American Sign Language word recognition system based on neural networks and a probabilistic model" in Turkish Journal of Electrical Engineering & Computer Sciences, vol.23, pp 2107 2123, 2015.

[7] P.V.V.Kishore, S.R.C.Kishore, M.V.D.Prasad "Conglomeration of hand shapes and texture information for recognizing gestures of Indian sign language using feed forward neural networks" International Journal of engineering and Technology, Vol. 5, No. 5, pp.3742-3756, 2013.

[8] P.V.V.Kishore ,M.V.D.Prasad"Optical Flow Hand Tracking and Active Contour Hand Shape Features for Continuous Sign Language Recognition with Artificial Neural Networks" International Journal of Software Engineering and Its Applications,v. 9, no. 12, pp. 231-250, 2015.

[9] Tzuu-Hseng S. Li, Min-Chi Kao, and Ping-Huan Kuo "Recognition System for Home-Service-Related Sign Language Using Entropy-Based K-Means Algorithm and ABC-Based HMM" in IEEE transactions on systems, man, and Cybernetics: systems, vol.46, no.1, pp. 150-162.

[10] Li-Chun Wang, Ru Wang, De-Hui Kong, Bao-Cai Yin, "Similarity Assessment Model for Chinese Sign Language Videos" IEEE Transactions On Multimedia, vol. 16, no. 3, pp. 751-761, 2014.

[11] P. V. V. Kishore, A. S. C. S. Sastry and A. Kartheek, "Visual-verbal machine interpreter for sign language recognition under versatile video backgrounds," 2014 First International Conference on Networks & Soft Computing (ICNSC2014), Guntur, 2014, pp. 135-140.

[12] Kishore, P. V. V., M. V. D. Prasad, Ch Raghava Prasad, and R. Rahul. "4-Camera model for sign language recognition using elliptical fourier descriptors and ANN." In Signal Processing And Communication Engineering Systems (SPACES), 2015 International Conference on, pp. 34-38. IEEE, 2015.

[13] Rao, G. Ananth, and P. V. V. Kishore. "Sign language recognition system simulated for video captured with smart phone front camera." International Journal of Electrical and Computer Engineering 6, no. 5 (2016): 2176.

[14] P. V. V. Kishore, M. V. D. Prasad, D. A. Kumar and A. S. C. S. Sastry, "Optical Flow Hand Tracking and Active Contour Hand Shape Features for Continuous Sign Language Recognition with Artificial Neural Networks," 2016 IEEE 6th International Conference on Advanced Computing (IACC), Bhimavaram, 2016, pp. 346-351.

[15] D. A. Kumar, P. V. V. Kishore, A. S. C. S. Sastry and P. R. G. Swamy, "Selfie continuous sign language recognition using neural network," 2016 IEEE Annual India Conference (INDICON), Bangalore, 2016, pp. 1-6.

[16] Almeida, Sílvia Grasiella Moreira, Frederico Gadelha Guimarães, and Jaime Arturo Ramírez. "Feature extraction in brazilian sign language recognition based on phonological structure and using rgb-d sensors." Expert Systems with Applications 41, no. 16 (2014): 7259-7271.

[17] Li, Shao-Zi, Bin Yu, Wei Wu, Song-Zhi Su, and Rong-Rong Ji. "Feature learning based on SAE–PCA network for human gesture recognition in RGBD images." Neurocomputing 151 (2015): 565-573.

[18] Chai, Xiujuan, Guang Li, Xilin Chen, Ming Zhou, Guobin Wu, and Hanjing Li. "Visualcomm: A tool to support communication between deaf and hearing persons with the kinect." In Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility, p. 76. ACM, 2013.

[19] Geng, Lubo, Xin Ma, Haibo Wang, Jason Gu, and Yibin Li. "Chinese sign language recognition with 3D hand motion trajectories and depth images." In Intelligent Control and Automation (WCICA), 2014 11th World Congress on, pp. 1457-1461. IEEE, 2014.

[20] Nai, Weizhi, Yue Liu, David Rempel, and Yongtian Wang. "Fast hand posture classification using depth features extracted from random line segments." Pattern Recognition 65 (2017): 1-10.

[21] Zhang, Zhengyou, and Alexey Vladimirovich Kurakin. "Dynamic hand gesture recognition using depth data." U.S. Patent 9,536,135, issued January 3, 2017.

[22] P. Praveen Kumar, P. V. G. D. Prasad Reddy and P. Srinivasa Rao, "Sign language recognition with multi feature fusion and Adaboost classifier", ARPN Journal of Engineering and Applied Sciences, vol.13, no.4, Feb(2018).

[23] Shan, Caifeng, Tieniu Tan, and Yucheng Wei. "Real-time hand tracking using a mean shift embedded particle filter." Pattern recognition 40, no. 7 (2007): 1958-1970.
[24] Mitra, Sushmita, and Tinku Acharya. "Gesture recognition: A survey." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37, no. 3 (2007): 311-324.
[25] Rautaray, Siddharth S., and Anupam Agrawal. "Vision based hand gesture recognition for human computer interaction: a survey." Artificial Intelligence Review 43, no. 1 (2015): 1-54.
[26] Lim, Kian Ming, Alan WC Tan, and Shing Chiang Tan. "A feature covariance matrix with serial particle filter for isolated sign language recognition." Expert Systems with Applications 54 (2016): 208-218.
[27] Viola, Paul, and Michael Jones. "Rapid object detection using a boosted cascade of simple features." In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, vol. 1, pp. I-I. IEEE, 2001.
[28] Piccardi, Massimo. "Background subtraction techniques: a review." In Systems, man and cybernetics, 2004 IEEE international conference on, vol. 4, pp. 3099-3104. IEEE, 2004.
[29] Schalkoff, Robert J. Artificial neural networks. Vol. 1. New York: McGraw-Hill, 1997.
[30] Awad, George, Junwei Han, and Alistair Sutherland. "A unified system for segmentation and tracking of face and hands in sign language recognition." In Pattern Recognition, 2006. ICPR 2006. 18th International Conference on, vol. 1, pp. 239-242. IEEE, 2006.
[31] Soontranon, N., Supavadee Aramvith, and Thanarat H. Chalidabhongse. "Improved face and hand tracking for sign language recognition." In Information Technology: Coding and Computing, 2005. ITCC 2005. International Conference on, vol. 2, pp. 141-146. IEEE, 2005.
[32] Shukla, Pushkar, Abhisha Garg, Kshitij Sharma, and Ankush Mittal. "A DTW and Fourier Descriptor based approach for Indian Sign Language recognition." In Image Information Processing (ICIIP), 2015 Third International Conference on, pp. 113-118. IEEE, 2015.
[33] Kaur, Gurwinder, and Gourav Bathla. "Hand Gesture Recognition based on Invariant Features and Artifical Neural Network." Indian Journal of Science and Technology 9, no. 43 (2016).
[34] Fu, Xingang, Jiang Lu, Ting Zhang, Chadwell Bonair, and Marvin L. Coats. "Wavelet Enhanced Image Preprocessing and Neural Networks for Hand Gesture Recognition." In Smart City/SocialCom/SustainCom (SmartCity), 2015 IEEE International Conference on, pp. 838-843. IEEE, 2015.
[35] Kim, Taehwan, Weiran Wang, Hao Tang, and Karen Livescu. "Signer-independent fingerspelling recognition with deep neural network adaptation." In Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on, pp. 6160-6164. IEEE, 2016.
[36] Starner, Thad, and Alex Pentland. "Real-time american sign language recognition from video using hidden markov models." In Motion-Based Recognition, pp. 227-243. Springer, Dordrecht, 1997.
[37] Li, Xingyan. "Gesture recognition based on fuzzy C-Means clustering algorithm." Department Of Computer Science The University Of Tennessee Knoxville (2003).
[38] P. V. V. Kishore, D. A. Kumar, A. S. C. S. Sastry and E. K. Kumar, "Motionlets Matching with Adaptive Kernels for 3D Indian Sign Language Recognition," in IEEE Sensors Journal, vol. PP, no. 99, pp. 1-11. doi: 10.1109/JSEN.2018.2810449.