Rule Based Inference Engine for Continuous Indian Sign Language Recognition

Ms.N.Devi,
Assistant Professor,
Department of Information Technology,
Sri Venkateswara College of Engineering,
Sriperumbudur, India

Dr.K.S.Easwarakumar
Professor
Department of Computer Science and Engineering,
Anna University, Chennai, India

Abstract: Sign languages are naturally developed and fully fledged languages used by Deaf and Hard-of-Hearing for everyday communication among themselves. Information is conveyed visually using a combination of manual and non-manual means of expression. In this paper, a novel tiered architecture for recognizing Indian Sign language from a continuous video sequence is proposed. Our Architecture takes a vision based approach for tracking facial cues, upper body posture, and hand shapes. Facial cues are analyzed using active appearance model whereas upper body posture are estimated by solving sequential Bayesian filtering problem. Hand shape features are represented using freeman chain codes and classified using dynamic Bayesian network. In order to continuously recognize the sign language phrases and to guarantee soft real time execution, we use rule based inference engine which correlate canonical hand shapes to higher level events.

Keywords - Sign Language Recognition, Dynamic Bayesian Network, RETE

I. INTRODUCTION

Sign languages are natural languages that develop in communities of deaf people around the world and vary from region to region. Some signs can be distinguished by manual parameters alone, while others remain ambiguous unless additional non-manual information is made available. The manual parameters are hand shape, hand posture, hand location, and hand motion. The non-manual parameters include head and body posture, facial expression, gaze, and mouth movements. The latter encode, e.g., adjectives and adverbials, contribute to grammar or provide specialization of general items. Signs are performed either one-handed or two-handed. For one-handed signs the action of only one hand is required, where a person generally uses the same hand, known as the dominant hand. The grammar of sign language is fundamentally different from spoken language. The structure of a sentence in spoken language is linear, one word followed by another, whereas in sign language, a simultaneous structure exists with a parallel temporal and spatial configuration. The configuration of a sign language sentence carries rich information about time, location, person, or predicate.

There are mainly two different approaches in sign language recognition - Glove based approach and vision based approach. The first category requires signers to wear a sensor glove or a colored glove. The wearing of the glove simplifies the task of segmentation during processing. The drawback of this approach is that the signer has to wear the sensor hardware along with the glove during the operation of the system. Vision based approach [6] uses image processing algorithms to detect and track hand signs as well as facial expressions of the signer. This approach is easier to the signer since there is no need to wear any extra hardware. However, there are accuracy problems related to image processing algorithms and these problems are yet to be modified.

There are again two different approaches in vision based sign language recognition: 3D model based and appearance based [1]. 3D model based methods make use of 3D information of key elements of the body parts. Using this information, several important parameters, like palm position, joint angles etc., can be obtained. This approach uses volumetric or skeletal models, or a combination of the two. A second approach in sign language gesture detection using appearance-based models uses image sequences as the gesture templates. Either the images themselves, or certain features derived from these images can be used as the parameters.

In this paper we are proposing a new sign language recognition system that attends to both body and hands, and interprets sign gestures continuously from an unsegmented and unbounded input stream, for transforming signs of Indian sign language in to application commands using hand and head gestures of humans.

The main contributions of this paper are as follows:
We propose a vision based approach for tracking body, face and hand. Body postures are reconstructed using a generative model based approach with particle filter. Facial expression is modeled by an active appearance model (AAM), a statistical model which combines shape and texture information about human faces. Hand shapes are classified into one of several canonical hand shapes using freeman chain codes and dynamic Bayesian classifier.

We proposed tiered architecture for correlating the canonical hand shapes to create higher level events which can able to guarantee real time recognition and map the Sign language recognizer with any application.

II. SIGN LANGUAGE IN INDIA

In literature, it was found that count of hearing impaired people in India, is more compared to other countries. Not all of them use ISL but, more than one million deaf adults and around half million deaf children use ISL as a mode of communication. Deaf people, who live in villages usually, do not have access to sign language. However, in all large towns and cities across the Indian subcontinent, deaf people use sign language which is not standard sign language. Extensive work and awareness program are being done for implementation of ISL in education systems [3]. Most of the schools use their own native sign language as a teaching and learning aid, therefore, for awareness to use of standard ISL as a teaching aid is being done by different ISL cells and NGOs to help Indian deaf and dumb community to bridge the communication gap between them. There are some common wrong beliefs about sign language which is reported in ISL literature [3]:

- “Sign language is same all over the world”
- “Sign language is not a complete language. It is just a sort of pantomime or gesturing, and it has no grammar”
- “Sign language is dependent on spoken language. It is a representation of the spoken language of the hands”
- “Sign language is the language of the hands only”
- “Sign language has been invented by other people to help deaf people”
- “Signed Hindi or signed English is better than Indian sign language”

So overcoming these wrong beliefs, there is a need of developing ISL interpretation system to aid Indian hearing impaired people with the help of HCI and making them literate and self dependent. Major research work is going on awareness and multilingual Indian sign language dictionary tool [4], so there is a need for Indian sign language interpretation tool. Following may be the major advantages of ISL interpretation:

- Use and awareness of computer interface through ISL interpretation.
- Education and training will be easier through ISL interpretation/visualization for Indian deaf and dumb people.
- Serving the mankind by use of technology.
- Social aspect like humanity can increase in individual mind by involving physically impaired people in our day to day life.
- Blind people can also use the same system by extending it for voice interface.

III. PROPOSED SYSTEM

Figure 1 shows a schematic of our proposed system. The system utilizes a single video camera for data acquisition to ensure user-friendliness. Since sign languages make use of hand, upper body and facial means of expression, all the channels are employed for recognition. The main components are video preprocessing, Facial expression analyser, upper body posture estimator, hand shape classifier, rule based inference engine for continuous sign recognition.

A. Video Preprocessing

Preprocessing stage consists of basically three operations namely background subtraction, Silhouette based cropping, and adaptive skin detection.

1 Background Subtraction

As we are focusing on sign language recognition using hand, body and facial cues, a background segmentation
The codebook background subtraction algorithm works by sampling values over long times, without making parametric assumptions. For each pixel, it builds a codebook consisting of one or more codewords. Samples at each pixel are clustered into the set of codewords based on a color distortion metric together with brightness bounds. Mixed backgrounds can be modeled by multiple codewords. The background is encoded on a pixel-by-pixel basis. Background detection involves testing the difference of the current image from the background model with respect to color and brightness differences. If an incoming pixel meets two conditions, it is classified as background:

1. The color distortion to some codeword is less than the detection threshold, and
2. Its brightness lies within the brightness range of that codeword. Otherwise, it is classified as foreground.

### 2 Silhouette Based Cropping

In order to extract translation and scale invariant silhouettes, the following two steps are accomplished. First all frames are cropped with respect to the bounding box within which the performer lies. This allows horizontal and vertical translation invariances. Scale invariance are obtained by resizing the cropped frames to a fixed frame resolution.

### 3 Adaptive Skin Segmentation

We use adaptive detection of human skin in images based on a normalized lookup table proposed by[8]. This is done by collecting measures of skin and non-skin pixel color samples and arranging them in a normalized histogram. This histogram provides a probability indicating how likely each pixel is skin or non-skin, such that a probability map is created for the entire image. Color histograms were used to construct these histograms, each pixel forms a vector [RGB] which is translated to the lookup table as:

\[ H([RGB]) = H(R + [G \times 256] + [B \times 256 \times 256]) \]  

To find a probability of a pixel being in each group

\[ P([RGB]) = \frac{H([RGB]) \times |H|}{n=1 H(n)} \]  

For every test image, we use a threshold t and the LUT to classify the image as follows:

\[ \text{if } P(\text{pixel} | \text{skin}) P(\text{pixel} | \neg \text{skin}) \geq t \text{ then P is labeled as skin} \]  

Where \( P(\text{pixel} | \text{skin}) \) is the probability of a pixel containing skin (informed by the histogram of the skin group) and \( P(\text{pixel} | \neg \text{skin}) \) is the probability of a pixel being in the non-skin group. B.

After the skin pixels are detected, they are grouped together in skin blob according to their connectivity. In order to diminish the false positive from the skin detection, blobs that have an area below certain thresholds are discarded.

### B. Facial Expression Analyzer

The interpretation of facial expression is based on so-called Action Units which represent the muscular activity in a face. In order to classify these units, areas of interest, such as the eyes, eyebrows, and mouth (in particular the lips) as well as their spatial relation to each other, have to be extracted from the images. For this purpose, the face is modeled by an active appearance model (AAM), a statistical model which combines shape and texture information about human faces. Based on an eigenvalue approach the amount of data needed is reduced, hereby enabling real-time processing.
C. Upper Body Posture Estimation

The goal here is to reconstruct upper body posture for the given input image sequence. We formulate this as a sequential Bayesian filtering problem[9]. Given a sequence of image \( I(t) = \{I(1), I(2), ..., I(t)\} \) with prior state density \( p(x(t)) \), Predict a Posterior state density \( p(x(t)|I(t)) \) where \( x(t) = (x_{1,t}, x_{2,t}, ..., x_{k,t}) \) is a \( k \)-dimensional vector representing the estimated body posture.

Constitute 8x8 convolution kernel corresponding to elbow and shoulder silhouettes as these two feature also convey an important part of sign language utterance meaning apart from static hand gestures. Those body parts are then detected as convolution maxima.

The coordinates of the right elbow on the frame \( t \) will be written as \( E(t) (x_e(t),y_e(t)) \). The aim is to select as many valid coordinates as possible so that for any couple of valid coordinates\( (E(t_1),E(t_2)) \) measured at times \( t_1 \) and \( t_2 \) the average speed of the elbow has its coordinates smaller than a predefined threshold \( \text{thres} \).

\[
\forall t_1,t_2 \left( \frac{|x_e(t_2) - x_e(t_1)|}{t_2 - t_1} < \text{thres}, \frac{|y_e(t_2) - y_e(t_1)|}{t_2 - t_1} < \text{thres} \right)
\]

(4)

Now we reduce the sequestial bayesian problem to include only the hand and elbow features as

\[
P_1 = p(x(t)|Hl(t), Hr(t), Er(t), El(t))
\]

(5) Where \( Hl(t), Hr(t), Er(t), El(t) \) represent left hand, right hand right elbow and left elbow coordinates. \( P_1 \) will give measure of relative position of hand and shoulders. Next we have to compute the overall displacement of both hands by comparing hand coordinates with the relative position of hand and shoulders. This can be written as

\[
\arg\min \left( \sum_{t=1}^{T} [\text{dist}(t-1,t) + \alpha \ln(B(t))] \right)
\]

(6)

Where \( \text{dist}(t-1,t) \) is the sum of the hand displacement between frame \( t-1 \) and \( t \), \( B(t) = P_1.P(x(t)|Hl(t), Hr(t)) \) and \( \alpha \) a constant.

D. Hand Shape Classification

The goal of hand shape classification is to classify hand shapes made contemporaneously with gestures into one of several canonical hand shapes. This is done through three major processing steps namely: Setting search region, Feature extraction and classification.

1 Search Region

The wrist centre coordinates are computed using forward kinematics problem defined by Denavit-Hatnernberg convention[10]. A smaller search region was created around each of the estimated wrist position slightly larger than the average size of the actual hand image. A search for a hand shape is carried out using sliding window. If a hand is found at time \( t-1 \), for time \( t \) we centre the search region at the geometric mean of the estimated wrist position and hand position inorder to improve the accuracy.

2 Feature Extraction

Among the variety of possible features the most important about a hand shape will be the trajectory of hand in space over time which is encoded by by freeman chain code [11]. Chain codes are generated by following a boundary of an object in clockwise direction and assigning direction to the segments connecting every pair of pixels.

First we pick a starting pixel location anywhere on the object boundary. Our aim is to find the next pixel in the boundary. There must be an adjoining boundary pixel at one of the 8 location surrounding the current boundary. By looking at each of the 8 adjoining pixel, we find atleast one that is also a boundary pixel. Depending on which one it is we assign a numeric code of between 0 and 7 as shown in figure 3.

![Figure - 3: Direction Codes For 8-Directional Chain Code](image)

The process of locating the next boundary pixel and assigning a code is repeated until we came back to our first location or boundary pixel. The result is a list of chain code showing the direction taken in moving from each boundary pixel to the next. With the separate chain coding for each hand ambiguities can arise between gestures. To remove the ambiguities incurred by representing the motion using only the chain code we introduce two more features: the relative position of the
two hand (figure 4) and the position of each hand relative to the face (figure 5).

**Figure 4 Relative Position of Two Hands**

**Figure 5 Position of Hands Relative To Face**

### 3 Dynamic Bayesian Classifier

To classify the hand shapes, we propose a modified Dynamic Bayesian Network (DBN) which has 3 hidden variables and 5 observable variables. The two hidden variables X1 and X2 model the motion of the left and right hand respectively and each is associated with two observation of the feature of the corresponding hand motion and the position relative to face. The third hidden variable X3 has been introduced to resolve the ambiguity between similar gestures. It models the spatial relation between hands.

**Figure - 6: DBN with 3 Hidden States**

We use the DBN as shown in the figure 6 where X1 and X2 are marginally independent when the X3 is not known. That means each hand can make any motion irrespective of the other hand’s motion. We formulate the observable variables as Ohl, Ohr, Ohlf, Ohrf and Oh corresponding to chain code for left hand and right hand, spatial relation of face with left and right hand and spatial relation between two hands.

The inference engine over a DBN is simply computing the marginal probability $P(X_i|O_{1:t})$ of hidden variables $X_i$ given an observation sequence $O_{1:t} = \{O_1, O_2, O_3, O_4, O_5\}$. According to the naïve Bayesian rule of conditional probability, the desired joint probability can be factorized into a product of local conditional independencies. The full joint probability for the DBN can be computed by multiplying 3 factored probabilities as follows:

$$P(X, O) = P(O|X)P(X) = \pi A B$$  \hspace{1cm} (7)

$$\pi = P(X_1)P(X_2)P(X_3|X_1, X_2)$$

$$A = \prod_{t} P(21_{t}|x_{1_{t}} - 1)P(22_{t}|x_{1_{t}} - 1)P(36_{t}|x_{1_{t}} - 1)x_{1_{t}} x_{2_{t}}$$

$$B = \prod_{t} P(0Oh(t), 0Ohf(t)|x_{1_{t}}), P(Ohr(t), Ohrf(t)|x_{1_{t}}), P(Oh(t)|x_{1_{t}})$$

Next step is the inference where a probability distribution over the set of gesture classes is assigned to the feature vector representing a gesture using junction tree algorithm [12] which construct a data structure called a junction tree and produce the output according to the actual and predicted classes for each gesture with respective ID numbers.

### E. Rule Based Inference Engine for Continuous Sign Recognition

To provide a high quality user experience, an inference engine used for gesture recognition has to correlate events in a timely manner. In addition the system also needs to guarantee predictable response time. To achieve this requirement, a tiered architecture for event processing is proposed. Rules of tier n can consume only events that were generated by lower level tiers 1 to N-1. It enables the developers to easily modularize and compose their rules. For instance, wave gesture can be composed of two lower level gestures: flip right and flip left which themselves were recognized in the lower level recognizer viz hand shape classifier.

The architecture incorporates a RETE algorithm based production system to detect user-interaction pattern. This system process incoming lower level events from a set of event source based on a given set of rules which describe relevant patterns that need to be recognized in these event streams. The S-expression in Table 1 gives an example of a high-level motion gesture rule (receive sign) that can be processed by the inference engine. The expression defines the rule detectShape, which describes how opening hands and closing hands while moving towards body can combine into a shape. Line by line, the rule binds two events of type open and close hands to the variables ?hol, ?hor, ?hcl, ?hcr and binds an event of type move towards body and extend hand to the variable ?hA and ?hB. Then, it uses the user-defined functions end Meets Start and chronologically to verify that the shapes follow each other both spatially and temporally. As a consequent to the recognition of the Z-shape, the rule specifies that the callback...
reportZShapeCenteredOn should be called, passing the average x and y slots of the shape’s points.

Table 1: Rule For Receive Sign

<table>
<thead>
<tr>
<th>Rule For Receive Sign</th>
</tr>
</thead>
<tbody>
<tr>
<td>( deffunction detectShape</td>
</tr>
<tr>
<td>?hol &lt;- (open left hand)</td>
</tr>
<tr>
<td>?hor &lt;- (open right hand)</td>
</tr>
<tr>
<td>?hcl &lt;- (close left hand)</td>
</tr>
<tr>
<td>?hcr &lt;- (close right hand)</td>
</tr>
<tr>
<td>hA &lt;- (move towards body)</td>
</tr>
<tr>
<td>hB &lt;- (extend hand)</td>
</tr>
<tr>
<td>( test ( endMeetsStart ?hol ?hor ?hB ?))</td>
</tr>
<tr>
<td>( test ( chronologically ?hcl ?hcr ?hA ?))</td>
</tr>
<tr>
<td>=&gt; ( reportShapeCenteredOn</td>
</tr>
<tr>
<td>(avg ?hA. startX ?hA. endX</td>
</tr>
<tr>
<td>?hB. startX ?hB. endX</td>
</tr>
<tr>
<td>(avg ?hA. y ?hB. y)))</td>
</tr>
</tbody>
</table>

These rules are then inserted into the production system at start up to construct the RETE network. Each node in a RETE graph represents the set of variable bindings that match a rule assertion. The RETE procedure works by moving assertions through this graph, saving incremental match info as it goes. A path through the graph to a leaf node represents the bindings that match the antecedents in a rule. The RETE graph is constructed as shown in figure 7 using the procedure as follows:

- For each antecedent, create an alpha node, aka a match node.
- Join a first antecedent and second antecedent to create a beta node, aka a merge node.
- Join each subsequent antecedent with the previous merge node to create a new merge node.
- For each consequent, create a terminal node, aka an execution node, that carry out the consequent.

The production system then computes the lexical address of slots within facts and the corresponding partial matches as given below:

- Add the assertion variable bindings to the appropriate matche and merge node.
- If execute node is reached, carry out consequents. If consequents adds an assertion, make sure assertion is new.

The Production system then creates a set of actors and links them up to each other to constitute the RETE network. Finally the application layer uses a publish-subscribe model to register for high level events processed by the inference engine. Depending on the application it is useful to support different subscription model. Topic based subscription are used to filter the type of event and are the most common one.

CONCLUSIONS

Architecture for recognizing Indian sign language using hand gestures along with facial, upper body cues is presented in this paper. The system uses dynamic Bayesian network with freeman chain codes for feature extraction and rule based inference engine for continuous gesture recognition to provide soft real time guarantees. The system is able to recognize 10 sign gestures ("put", "push", "receive", "stop", "how many", "what time", "pull", "reach", "adjust", "agree") from Indian sign language dictionary. As the architecture implements completely by using digital image processing technique so the user does not have to wear any special hardware device to get features of hand shape. Developing such system translating sign language to text/voice format can be proved very useful for physically impaired people of India.

References


