Literature Survey on Gesture Classification Techniques

Sreenath S P
Department of Computer Science and Engineering
Amrita School of Engineering
Coimbatore, India.

Padmavathi S
Department of Computer Science and Engineering
Amrita School of Engineering
Coimbatore, India.

Abstract— **Gesture Recognition is been a prominent method in making human-computer interaction system. One of the main application of any gesture recognizer is sign language recognition. Particularly in this field a lots of advancements has been bought such as from identifying a static or isolated action to identifying continuous or dynamic gestures. The process of recognition of dynamic gestures involves various steps such as segmentation of ROI, tracking of key point, feature extraction and classification of gesture. In the process of gesture recognition, classification which is the final step involves computerized processing of the data which has been acquired from the actions or gestures performed and determine whether the data corresponds to a particular gesture. For improving the accuracy of recognition, various pattern recognition or machine learning algorithms as HMM, Artificial Neural Networks, and fast DTW. The main purpose of this paper is to analyze these methods and compare them, enabling the reader to find an optimal solution for their problem.**

Keywords— Feature vectors; tracking; hidden markov models(HMM); prior probabilities; transition probabilities; emission probabilities; evaluation; decoding; DTW grid; DTW wrap path; Artificial neural network; training phase; system learning period.

INTRODUCTION

The final step of any sign language recognition system is to find sign language action performed. The main input for this module is an array of feature vectors that are being extracted from each corresponding frame, of the performed sign video. For a given sign video there will be a pattern or sequence by which the features change with respect to time. The purpose is to find the action performed in the video using the features vectors of the frames of the video, by checking those features with the features of the trained videos. The choice of training videos is planned to be done in such a way that the system is robust for users to detect gestures even, if they are not performed with 100% accuracy. Since a video can be considered as a sequence of frames and in turn each frame can be taken a single static image, it is possible to do tracking and obtain feature vector for each frame. This process is done for each of the videos that are to be trained and the feature vectors are stored in database. Therefore this classification of gesture can be viewed as a pattern recognition problem [1], where based on the pattern or sequence of feature vectors a particular gesture is classified. Since the process is time dependent the following approaches will be more suitable [2] for classification,

1. Hidden Markov Model (HMM).
2. Dynamic Time Warping (DTW).
3. Artificial Neural Networks.

HIDDEN MARKOV MODEL (HMM)

A Markov model is a statistic time related modeling which real world events are mapped to time domain [8]. A typical Markov model can be the one in which the state of an event is directly visible to the observer, and hence the only parameters needed are the state transition probabilities. Similarly a Hidden Markov model, the state is not directly visible, but output, is dependent on the visible state. So Hidden Markov model (HMM) is a statistics based Markov model in which modeling of the system being done in such a way assuming that the states of Markov process are hidden states unobserved. For each state there a probability distribution for all the output tokens possible. Hence a sequence of tokens generated by an HMM gives information relating to the possible state sequences.

A HMM Model is specified by:

- The set of states S = $\{s_1, s_2 \ldots s_N\}$, and
- a set of parameters = $\{\pi, A, B\}$:

The prior probabilities $\pi_i = P(q_1 = s_i)$ are the probabilities of si being the first state of a state sequence. Collected in a vector π. (The prior probabilities were assumed equi-probable in the last example, $\pi_i = 1/N$.)

The transition probabilities are the probabilities to go from state i to state j: $a_{i,j} = P(q_{n+1} = s_j \mid q_n = s_i)$. They are collected in the matrix A.

The emission probabilities characterize the likelihood of a certain observation x, if the model is in states s_i. for discrete observations, $x_n \in \{v_1, v_2, \ldots v_k\}$: $b_{i,k} = P(x_n = v_k \mid q_n = s_i)$, the probabilities to observe v_k if the current state is $q_n = s_i$. The numbers $b_{i,k}$ can be collected in a matrix B.

Here in our case we can relate that each gesture can be considered as a likelihood output of a certain observation x. for each gesture there is a sequence by which the feature vector changes ,those state transition probabilities are kept in matrix A. Having collected these set of parameters, training is done by building FSMs based on the probabilities of transition of features of an image frames as the time moves on. Now the task is to find the HMM which gives a maximum match with the given video i.e. sequence of

transitions which most matches with the sequence of feature vector transitions in the given video. Steps involved [9] are,

A. Evaluation
Given: model, Observed Sequence.
Wanted: likelihood model produced by the observation sequence.
Compute the likelihood that a given model *M* produced a given observation sequence *O.* that is
P (p_1=O_1,….,p_T=O_T|M)
Likelihood can be found using dynamic programming either forward or backward approach.

B. Decoding
Given: model, Observed Sequence.
Wanted: the most likely hidden sequence.

Compute the most likely sequence of hidden states for a given model M and a given observation sequence O.
H=argmax P (q_1=H_1,….,q_T=H_T|M,O)
The most likely hidden path can be computed efficiently using the Viterbi-algorithm. It is a traces the most likely hidden states while reproducing the output sequence.

C. Learning
Given: Observed Sequence.
Wanted: Most likely model that produced the observed sequence.

Given an observation sequence O and the corresponding hidden sequence H, compute the most likely model M that produces those sequences
M=argmaxP[(p_1=O_1,q_1=H_1),….,(p_T=O_T,q_T=H_T)|M)]
Solved using "instance counting", Count the hidden state transitions and output state emissions. Use the relative frequencies as estimate for transition probabilities of M

DYNAMIC TIME WRAPING(DTW)

A time domain series is often said to be a collection of observations that are made in respect to time sequentially . For the sake of analyzing the time series a dynamic programming algorithm called DTW being used, dynamic time warping (DTW) is an algorithm for measuring similarity between two temporal sequences which may vary in time or speed [6]. A gesture video can be represented as a time series with feature vectors changing at as the time goes on. The main issues covered in this technique are,
It is possible that the gestures are performed at different speeds.
It is possible that the speed of the gestures could vary in different ways at different points.
DTW Grid:
We can arrange the two sequences of observations on the sides of a grid with the unknown sequence on the bottom and the stored template up the left hand side . Both sequences start on the bottom left of the grid. Inside each cell we can place a distance measure comparing the corresponding elements of the two sequences.

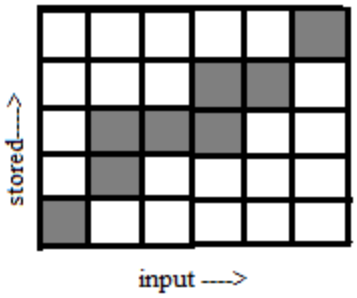

Fig 1.Warping window.

To find the best match between these two sequences we can find a path through the grid which minimizes the total distance between them. This is done for all the stored gestures and the gesture which has the lowest total distance of the grid path is considered to be the sign performed. To make the classification system to be more robust the following constraints are imposed. These major optimizations to the DTW algorithm arise from observations on the nature of good paths through the grid
By applying these observations [7] we can restrict the moves that can be made from any point in the path and so restrict the number of paths that need to be considered.

- **Monotonic condition**: the wrap path must not decrease (i.e.) i , j indexes increase or remain same but can't decrease.
- **Continuity condition**: the wrap path must be continuous throughout the grid w/o any intervals maintaining the continuity of the gesture action.
- **Boundary condition**: the path must start at bottom left ensuring the gesture starts correct and ends at the top right ensuring the completion of gesture action.
- **Adjustment window condition**: A adjustment window is framed so that the warping path can't wander too much, there by checking correctness of the gesture.

Generally the following steps are involved,

1. Finding an Optimal *D* path defines the "distance" between two given sequences. Cost involved in finding each node,
   $d(i,j)=[r(i)-t(j)]$
   Overall cost path,
   $D=\sum_k d[i(k),j(k)]$
2. Find an optimal path passing through the point (i,j) . It calculated using [5] a dynamic programming formula.
   $D(i,j)=Dist(i,j)+min[\ D(i-1,j),\ D(i,j-1),\ D(i-1,j-1)\ ]$

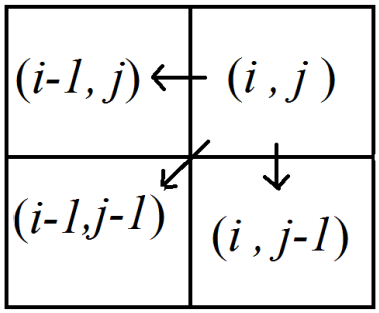

Fig 2.Wraping path calculation.

ARTIFICIAL NEURAL NETWORKS (ANN)

The typical Neural Network is basically an adaptive system which can teach itself to perform a function using the data sets. The term adaptive, refers that the parameters of system generally subjective and it may change during [4] the period of operation, this is usually called training phase [3]. As the training phase ends, the parameters of Neural Network are defined and the system is deployed to resolve the problem, which constitutes our testing phase. It is basically a system based on the operation of biological neural networks i.e. how a biological neural system works. Usually in this, a set of input-output pairs, this is often provided by means of external supervisors. Likewise using the difference between the desired response and the system output, the error value is computed. The error information obtained is been fed back to the system and the system parameters are adjusted in an organized fashion, this is usually called "System Learning Period".

The implementation can be done taking images or videos of the signs made by the signer by making use of a web camera [4]. In the case of video, its split into number of frames or images. Then the images are being processed and the characteristics that are essential for the process of recognition of sign are extracted and it's fed as inputs for an artificial neural network, which will recognize the sign.

ACKNOWLEDGMENT

We take this opportunity to express our sincere thanks to our university and our department for providing this opportunity to work on this. We would also like to thank the researchers whom we have stated in our references for works which helped us.

REFERENCES

[1] AcharyaTinku, MitraSushmita, "Gesture Recognition: A Survey" in *Ieee Transactions On Systems, Man, And Cybernetics—Part C: Applications And Reviews, Vol. 37, No. 3, May 2007.*

[2] Paranjape Ketki Vijay, Naphade Nilakshi Suhas, Chafekar Suparna Chandrashekhar & Deshpande Ketaki Dhananjay , "*Recent Developments in Sign Language Recognition : A Review*" in ISSN (Print): 2278-5140, Volume-1, Issue – 2, 2012.

[3] K. R. Linstrom and A.J. Boye. "A neural network prediction model for a psychiatric application" in *International Conference on Computational Intelligence and Multimedia Applications, pp. 36-40, 2005.*

[4] LungociuCorneliu, "Real Time Sign Language Recognition Using Artificial Neural Networks." in *Studia Univ. Babes_Bolyai, Informatica, Volume Lvi, Number 4, 2011.*

[5] Chunsheng Fang, "*From Dynamic Time Warping (DTW) to Hidden Markov Model (HMM)*" in Final project report for ECE742 Stochastic Decision, University of Cincinnati, 2009/3/19.

[6] http://en.wikipedia.org/wiki/Dynamic_time_warping

[7] Pavel Senin, "*Dynamic Time Warping Algorithm Review*", in Information and Computer Science Department, University of Hawaii at Manoa, Honolulu, USA, senin@hawaii.edu, December 2008.

[8] http://en.wikipedia.org/wiki/Hidden_Markov_model

[9] Lawrence R. Rabiner, "A Tutorial on Hidden Markov Models" in Readings in speech recognition (1990).