A Performance Evaluation of HMM and DTW for Gesture Recognition

Josep Maria Carmona and Joan Climent

Barcelona Tech (UPC), Spain

Abstract. It is unclear whether Hidden Markov Models (HMMs) or Dynamic Time Warping (DTW) techniques are more appropriate for gesture recognition. In this paper, we compare both methods using different criteria, with the objective of determining the one with better performance. For this purpose we have created a set of recorded gestures. The dataset used includes many samples of ten different gestures, with their corresponding ground truth obtained with a kinect. The dataset is made public for benchmarking purposes.

The results show that DTW gives higher performance than HMMs, and strongly support the use of DTW.

Keywords: Hidden Markov Models, Dynamic Time Warping, Gesture Recognition, Kinect.

1 Introduction

Visual recognition of hand gestures provides an attractive alternative to cumbersome interface devices for human-computer interaction. This has motivated a very active research concerned with computer vision-based analysis and interpretation of hand gestures. Computer vision and pattern recognition techniques, involving feature extraction, object detection, clustering, and classification, have been successfully used for many gesture recognition systems [4][7]. Preliminary works on vision-based gesture interpretation were focused on the recognition of static hand gestures or postures. However, hand gestures are dynamic actions and the motion of the hands conveys much more information than their posture does. While static gesture (pose) recognition can typically be accomplished by template matching and pattern recognition techniques, the dynamic gesture recognition problem involves the use of techniques such as Dynamic Time Warping (DTW) [6] or Hidden Markov Models (HMM) [8] [3].

There are some similarities between gesture and speech recognition [6] so that HMM or DTW, generally used for speech recognition, are also used for gesture recognition. In speech recognition applications, a hard task has been to recognize spoken words independent of their duration and variation in pronunciation. HMMs have shown to solve these tasks successfully. A HMM is associated with each different unit of language, while in gesture recognition each gesture can also be associated with a different HMM.

Although DTW and HMM have been applied in a large amount of works concerning gesture recognition, no previous work has done an exhaustive comparative study between both techniques. The objective of this paper is to compare the results of dynamic gesture recognition obtained using these two methods, according to recognition rates, sensitivity to the amount of training samples, optimal parameters, and computing times.

L. Alvarez et al. (Eds.): CIARP 2012, LNCS 7441, pp. 236–243, 2012.

The basics of HMM, DTW, and a description of the features commonly extracted from images are given in next section. Section 3 outlines the details of the experiments. The results are shown in section 4, and section 5 concludes the paper.

2 Preliminaries

2.1 Feature Extraction

The selection of the right features to be extracted from image sequences plays an important role in gesture recognition. Usually, the selected features are location, orientation, or velocity.

To determine the location, the coordinates are extracted directly from sequence frames, and they can be referenced to different coordinate origins. In some works, the hand location points are referenced to the distance from head. Others, like [1] use as origin the centroid of the hand trajectory, or the starting point of the hand gesture path. Holt et al. [2] use the position *(x,y,z)* relative to the head, of the left and right hand to recognize gesture from the standard vocabulary of Sign language of the Netherlands.

A second feature widely used is the orientation, which represents the direction of the hand at every point of the gesture path. It is computed as the displacement vector of every point and is represented by the orientation relative to the centroid of the gesture path, the orientation between two consecutives points or the orientation between the starting point and the current gesture point.

The third feature is the velocity, which plays an important role in recognition phase, particularly in some critical situations. It is computed as the distance between two successive points divided by the time measured in number of frames.

Ming-Hsuan et al. [9] use the position, velocity, and angle, to introduce the feature vector (x,y,v,θ) into a classifier whose outputs are the classes corresponding to each gesture.

2.2 Hidden Markov Models

A HMM λ (Fig. 1) consists of N states and a transition matrix $A=\{a_{ij}\}$, where a_{ij} is the probability of the transition from S_i state to S_j. Each state has assigned an output probability distribution function b_{im}, which gives the probability of the state S_i generating observation Om under the condition that the system is in S_i.

A HMM is a triple $\lambda=(A, B, \prod)$ as follows:

- A set of N states $S= \{s_1,s_2,\ldots s_N\}$.
- An initial probability for each state $\prod_i$; $i=1,2,\ldots,N$ such that $\prod_i = P(Si)$ in the initial step.
- A NxN transition matrix $A=\{a_{ij}\}$ where a_{ij} is the probability of the transition from S_i state to S_j; $1 \leq i, j \leq N$.
- A set of T possible emission $O = \{o_1,o_2,\ldots,o_T\}$.
- A set of M discrete symbols $V = \{v_1, v_2,\ldots, v_M\}$.
- An NxM observation matrix $B=b_{im}$ where b_{im} gives the probability of emitting symbol v_m from state S_i .

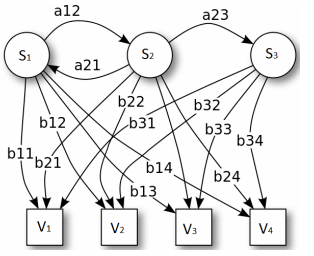


Fig. 1. HMM

In the training process, the parameters *(A, B, Π)* are modified to find the model that best describes the spatio-temporal dynamics of the desired gesture. Training is usually achieved by optimizing the maximum likelihood measure *log(P(observation|model))* over a set of training examples for the particular gesture. Such optimization involves the use of computationally expensive expectation-maximization procedures, like the Baum-Welch algorithm [5].

In the recognition stage, a gesture trajectory is tested over the set of trained HMMs in order to decide to which one it belongs. A probability of the gesture being produced by each HMM is evaluated using the Viterbi algorithm [5].

There are three main different topologies in HMMs; Fully connected (Ergodic model), where any state can be reached from any other state, Left-Right model (LR), such that any state can be iterate over itself or to next states, and Left-Right Banded (LBR) model that any state can only iterate over itself or to next state.

2.3 Dynamic Time Warping

The dynamic time warping algorithm computes an optimal matching path between two signals. The DTW algorithm calculates also the distance between the two signals computing the cumulative distance between each possible pair of points of both signals in terms of their associated feature values. The algorithm computes the local distance between the elements of two sequences *(a_1, a_2, …, a_n)* and *(b_1, b_2, …, b_m)* with lengths *n* and *m* respectively. The result is a distance matrix having *n* rows and *m* columns of terms:

$$d_{ij}=|a_i-b_j|, \quad i=1,n \quad j=1,m \tag{1}$$

From local distances, the minimal distance matrix between two sequences is calculated using a dynamic programming algorithm following the next optimization objective:

$$t_{ij} = d_{ij} + \min\left(t_{i-1,j-1}, t_{i-1,j}, t_{i,j-1}\right) \tag{2}$$

Being t_{ij} the minimum distance between (a_1, a_2, …, a_i) and (b_1, b_2, …, b_j).

A (*n*,*m*)-warping path is a sequence (a_{11}, …, a_{nm}) satisfying the following three conditions.

- Boundary condition: The path starts in left-down corner t_{11} and ends in right-up corner t_{nm}.

- Monotonicity condition: the path will not turn back on itself, that means that both *i* and *j* indexes either stay the same or increase, but never decrease.
- Step size condition: The path advances gradually. The indices *i* and *j* increase, at most, a single unit on each step.

3 The Experiment

We have compared DTW and HMMs recognition responses for a set of different gestures. For this purpose, we have created a dataset composed by 75 samples of ten different gestures corresponding to the numbers from 0 to 9. Therefore, our dataset is composed by 750 different samples. The gestures were made by three different persons, and different distances to the camera in order to ensure that the samples had different length and morphology.

The sequences have been obtained using a Microsoft's Kinect device. From the skeleton obtained using OpenNI libraries we get the coordinates of the position of the hands (Fig 2). Once we have these coordinates in each frame, the construction of the feature vector generated by hand movement, becomes an easy task. Kinect response is very robust to indoor illumination changes; therefore the coordinates obtained are not much dependent on the illumination conditions.

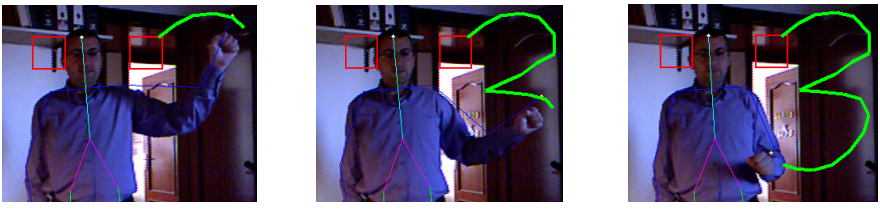


Fig. 2. Gesture path for number 3 captured from Kinect

Elmezain et al. analyze in [1] the performance of the three described main features: location, orientation, and velocity of the hand centroid. They prove that orientation is the most discriminant feature among the three, velocity has a lower discrimination power than orientation, and location feature has the lowest discriminative rate. Therefore, in our work, we have used as feature the orientations of the hand path between two consecutives frames.

We assign a codeword to each orientation. Our feature vector consists of a sequence of codewords corresponding to the directions, quantified in 18 bins. Thus, we have 75 feature vectors for each gesture, where each vector is composed by a sequence of codes from 1 to 18. The length of each of these sequences will vary according to gesture path length. Due to the angular nature of these features (our codebook is cyclic), we have adapted feature comparisons to circular arithmetic in order to avoid the zero-crossing problem.

We use exactly the same features for both experiments using DTW and HMM techniques, in order to avoid the influence of the chosen features.

In [1], they also study the optimal topology and number of states of HMMs. They compare Ergodic, LR and LRB topologies with a different number of states ranging from 3 to 10, and conclude that LRB topology is always better than LR and Ergodic topologies. Therefore, we use a LRB topology in our experiments.

DTW is a technique that does not recognize gestures directly, just gives distances between feature vectors. Thus, once we have got these distances, we use a K-NN classifier to determine which class is the most likely for a captured gesture. K-NN is a simple and effective classification procedure: decision surfaces are non-linear, it has a high capacity without training, and the quality of predictions assimptotically improves with the amount of data.

First, we have determined the optimum parameters for the HMM and the DTW algorithms. This is the first experiment. For HMMs we have studied the recognition rate using different number of states, ranging from 1 to 10. For DTW the objective is to determine the optimum number of neighbours, k, for the K-NN classifier.

Next, the second experiment determines the influence of the amount of training samples on the recognition rate.

In a third experiment we have studied the maximum, minimum and average recognition rates obtained using HMM and DTW. For each gesture class in the dataset, we have partitioned its sample group into two subgroups, performing the training using the first (training) subgroup, and cross-validating the analysis on the other (test) subgroup. In order to avoid the dependence on the samples chosen as training subgroup, we have used a bootstrap technique in all the evaluation experiments, repeatedly picking random subgroups of samples chosen from the dataset. The average performance using bootstraping is reported in the results section.

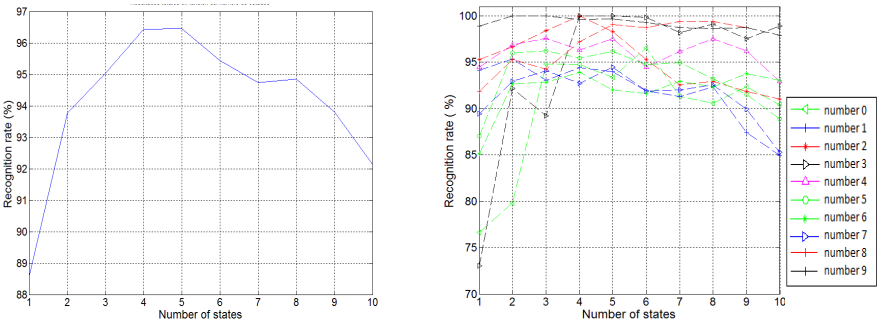
Finally, we have measured the training and recognition computing times for both methods.

4 Results

First of all we have to determine the optimal parameters for tuning both algorithms. For this purpose, we have first studied the classification results using HMMs, with LRB topology, for a different amount of states. We have tried from 1 state up to 10 states to evaluate which HMM achieved the highest recognition rate.

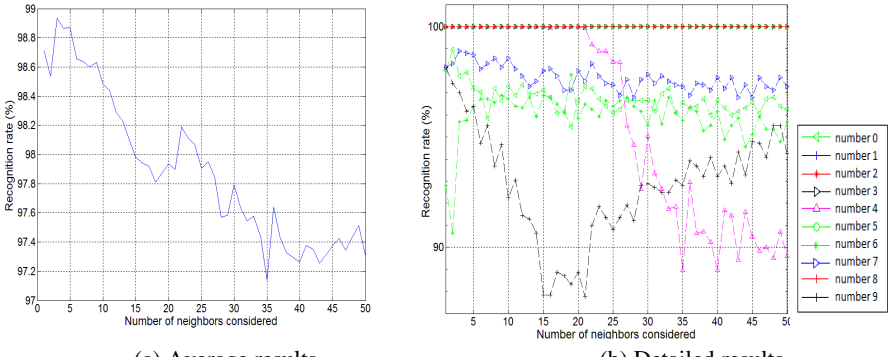
For each different gesture class, we have used 50 random subgroups consisting of 25 test samples and 50 training samples. Therefore, we have tested 1250 samples for each class. Fig 3(a) shows the average recognition rate, and fig. 3(b) shows the individual recognition rates obtained for each different gesture class. The best average recognition rate is 96% obtained with 5-state HMMs.

DTW method has been studied observing the recognition rate according the *K-NN* classifier. Only a single parameter k is needed, and it can be easily tuned by cross-validation. Again, we have used 50 random subgroups with 25 test samples and 50 training samples for each gesture class. Fig 4 shows the results. We can see that the best results are obtained with k=3, reaching 98.9% average recognition rate.



a) Average results (b) Detailed results

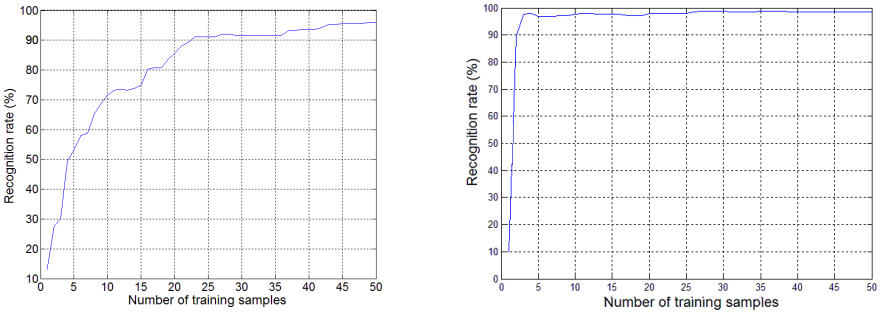
Fig. 3. HMM recognition rate depending on the number of states



(a) Average results (b) Detailed results

Fig. 4. DTW recognition rate depending on the number of neighbours

The next experiment is to determine the needed amount of training samples. We have tested the recognition rate of both algorithms using 25 test samples, and an increasing number of training samples from 1 to 50. Fig 5 shows the average recognition rates. We can see that recognition rates on DTW easily reach 90% with just 3 training samples, while HMMs need around 50 samples to achieve the same rates.



(a) HMM (b) DTW

Fig. 5. Recognition rate according number of training samples

Once we have the optimal parameters obtained from previous experiments, we can test HMM and DTW responses to determine their recognition rates. For each gesture class, we have randomly selected 100 groups of samples from original dataset. Each group consists of 25 test samples and 50 training samples, that is, we have tested 2500 samples of each gesture for both recognition techniques. We have used 5-state HMMs, and 3-Nearest neighbour classifier to classify gestures from DTW distances, since these were the optimal values obtained in the previous experiments. We have measured the recognition rate for each group of samples and computed the maximum, minimum and average recognition rate. Table 1 shows the recognition rates.

Table 1. Recognition rates

	Min	Max	Avg.
HMM	92,8%	99,2%	96,46%
DTW	97,2%	100%	98,84%

Finally, we have measured the computing times required for training and recognition. The average time needed to train a HMM with 50 samples for each gesture class is 17,3ms. DTW has no training stage. The average time taken for a HMM to classify a gesture feature vector is 3,6 ms. Using DTW, we need 0,2 ms. to compare two gesture feature vectors. We need 3 model samples with DTW in order to achieve an equal error rate with a HMM trained with 50 samples, therefore, comparing a test sample with 3 model samples for each gesture class, we obtain an average recognition time around 6 ms. All calculations have been performed with an Intel® Core™ 3.10GHz CPU.

5 Conclusions

We have constructed a set of experiments to test which one of the related recognition methods gives better results. We are not aware of such evaluation in previous literature. The most popular method used to identify gestures is by far HMM, but surprisingly, DTW gets better scores in all our experiments. Tuning both algorithms with the obtained optimal parameters, we obtain a 98.8% average recognition rate for DTW, whereas for HMM is only 96.46%.

We have also studied the sensitivity of the recognition rate to the number of training samples, and the conclusion is that HMMs need many more training samples than DTW to obtain similar recognition rates. This is an important fact to decide which method should be chosen according to the amount of samples available in user's dataset.

Moreover, we have constructed a dataset specifically designed to test both methods. It consists of different sequences of gesture images obtained with a kinect, together with their corresponding ground truth of hand positions. The dataset is freely downloadable from `http://urus.upc.edu/datasets/gestures/`.

Recognition time is lower on HMM than DTW, 3,6ms in front of 6 ms., but we must bear in mind that recognition times using DTW are directly proportional to the number of comparisons, and we have just proved that a small training database is

enough to obtain excellent rates using DTW. Even so, recognition time on DTW is more than acceptable for most applications.

The results obtained strongly encourage the use of DTW instead of HMMs.

Acknowledgments. This research was partially supported by Consolider Ingenio 2010, project (CSD2007-00018) and CICYT project DPI2010-17112.

References

1. Appenrodt, J., Elmezain, M., Al-Hamadi, A., Michaelis, B.: A hidden markov model-based isolated and meaningful hand gesture recognition. International Journal of Electrical, Computer, and Systems Engineering 3, 156–163 (2009)
2. ten Holt, G.A., Reinders, M.J.T., Hendriks, E.A.: Multi-Dimensional Dynamic Time Warping for Gesture Recognition. In: Thirteenth Annual Conference of the Advanced School for Computing and Imaging (2007)
3. Lee, H., Kim, J.: An HMM-based threshold model approach for gesture recognition. IEEE Trans. Pattern Analysis and Machine Intelligence 21(10), 961–973 (1999)
4. Pavlovic, V.I., Sharma, R., Huang, T.S.: Visual interpretation of hand gestures for human computer interaction. IEEE Trans. Pattern Anal. Mach. Intell. 19(7), 677–695 (1997)
5. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proc. IEEE 77, 257–286 (1989)
6. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Trans. Acoustics, Speech, and Signal Processing 26(1), 43–49 (1978)
7. Wexelblat, A.: An approach to natural gesture in virtual environments. ACM Transactions on Computer-Human Interaction 2(3), 179–200 (1995)
8. Wilson, A.D., Bobick, A.F.: Parametric hidden Markov models for gesture recognition. IEEE Trans. Pattern Analysis and Machine Intelligence 21(9), 884–900 (1999)
9. Yang, M.-H., Ahuja, N.: Recognizing Hand Gestures Using Motion Trajectories. In: IEEE Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 466–472 (1999)