ON HAND GESTURES RECOGNITION USING HIDDEN MARKOV MODELS

Vasilică TĂTARU *, Radu-Laurenţiu VIERIU* and Liviu GORAŞ*,#
* *"Gheorghe. Asachi" Technical University, # Institute of Computer Science, Romanian Academy*
Iaşi, 700506, Bd. Carol I, No. 11, Phone: (+40) 232 270041, Fax: (+40) 232 217720,
vtataru@etti.tuiasi.ro, rvieriu@etti.tuiasi.ro, lgoras@etti.tuiasi.ro

Abstract: In this paper several results concerning static hand gesture recognition using an algorithm based on left-right Hidden Markov Models (HMM) are presented. The features used as observables in the training as well as in the recognition phases are based either on the 2D Discrete Cosine Transform (DCT) or on the Principal Component Analysis (PCA). The left-right topology of the HMM together with the Baum-Welch algorithm for training and Viterbi algorithm for testing led to the best results. Simulation results show that the system has a recognition rate of 97.5% for DCT and 95% for PCA.

Keywords: *Human-Computer Interaction, Hand Gestures Recognition, Hidden Markov Models.*

I. INTRODUCTION

The tremendous development of informatics/robotic systems led to spectacular progress in miniaturization, signal processing capability and speed. Besides, the need of simple communication between the user and the informatics/robotic system grew constantly – a matter belonging to the domain of Human Computer Interaction (HCI). Its role is to ensure an as simple, natural and as efficient as possible communication link. Gestures represent one of the most natural and intuitive means of interaction and communication. However, gesture recognition is comparable in difficulty with other recognition task like face recognition.

The first models used in gesture recognition were: template-matching [1], dictionary lookup [2], statistical matching [3], linguistic matching [4] and neural networks [5]. During the last decade the methods with potential applications in gestural interfaces for HCI are based on the following models: neural networks [6], fuzzy systems [7] and HMM [8] which are fundamental for most gesture recognition algorithms used at present. However the classical HMM model used in gesture recognition systems requires a large number of parameters for training in order to obtain satisfactory results. In [8] Rajko et al. introduce an alternative HMM which reduces the number of parameters required to provide the transition probabilities. Nevertheless the HMMs are capable of modeling the spatio-temporal variability, where the same gesture may differ in shape and duration. Due to the fact that gestures are spatio-temporal models, they can be static or dynamic. In this work we present a method of automatic feature extraction from a set of training data and the use of these features in HMM learning with a continuous distribution of the observable sequences [9] with the aim of test image identification.

II. HIDDEN MARKOV MODELS

In the recent years mathematical models developed rapidly and found applications in various fields. A particular class of models are HMM's, a powerful statistical modeling tool which characterizes the data emitted in a discrete-time process by a system with unknown states. A HMM represents a model of a stochastic process in which a set of observations is generated in discrete time by a sequence of states connected by transitions. Depending on the type of observable sequences the HMMs can be *discrete* or *continuous*. In [9] Rabiner offers a vast and comprehensive tutorial in this domain.

A continuous HMM is characterized by the following elements:

- the number of states in the model or the dimension N of the state space. If $S = \{S_1, S_2, ..., S_N\}$ is the state set, then at any time t the model will be in the state given by $q_t \in S,\ 1 \le t \le T$, where T is the length of the observable sequence.
- the probability distribution of the transition between states A

$$A = \{a_{ij}\}$$
$$a_{ij} = P[q_t = S_j \,|\, q_{t-1} = S_i]\ 1 \le i, j \le N \quad (1)$$

where $0 \le a_{ij} \le 1$ and $\sum_{j=1}^{N} a_{ij} = 1,\ 1 \le i \le N$

- the output distribution probability associated to each transition B

Manuscript received September 13, 2010; Revised November 3, 2010

$$B = \{b_j(O)\}$$
$$b_j(O) = \sum_{m=1}^{M} c_{jm} N(O; \mu_{jm}; \Sigma_{jm}), \quad 1 \le j \le N \tag{2}$$

where $N(O; \mu_{jm}; \Sigma_{jm})$ represents the Gaussian density of probability, with mean μ_{jm} and covariance matrix Σ_{jm}.

- the distribution of initial state $\Pi = \{\pi_i\}$

$$\pi_i = P[q_1 = S_i], \quad 1 \le i \le N \tag{3}$$

These elements are also called model parameters. A HMM model is symbolically described by the relation:

$$\lambda = (A, B, \Pi) \tag{4}$$

III. DESIGN OF HAND GESTURE MODELS

The success of using HMM's for instance in speech recognition applications [9] was posible due to its one-dimensional nature since the analysis using 1D HMM requires one-dimensional observable vectors. When going to images which are two-dimensional a conversion into a 1D sequences is necessary. A spatial 1D sequences which proved to be more adequate [10] can be generated by sampling the image using an exploration window. This window converts the image into an 1D data sequence, where each element of the sequence is a vector with a given number of samples. The number of vectors may vary depending on the size of the exploration window. In this paper we use the image conversion into 1D spatial sequence and thus we can use 1D HMMs in hand gesture recognition.

The technique of generating the spatial sequence is shown in Fig.1. For every image from the training set of each gesture, a sequence of vectors $O = o_1 o_2 ... o_T$ is obtained. We point out that the exploration windows moving down along the image overlap by a given number of pixels. Thus the image is divided into overlapping blocks of size $L \times W$, where L represents the height of the exploration window and W is the image width. The total number of such blocks extracted from an image can be calculated using the formula:

$$T = \frac{H - L}{L - P} + 1 \tag{5}$$

where H is the image height and P is the overlapping between two consecutive blocks.

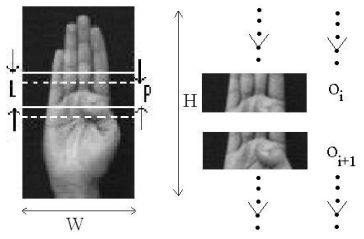

Figure 1. Conversion of a 2D image into a 1D spatial sequence

The means of generating the observable vectors implies the using of a left-right HMM. In Fig. 2 the general case for such a model is presented:

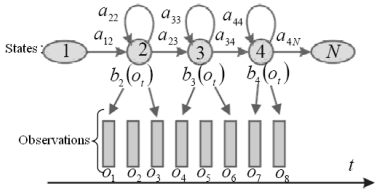

Figure 2. 1D hidden Markov model

In this paper we investigate the performances of several 1D HMM models in hand gesture recognition from grayscale images containing gestures, using as method of generating the observable vectors the two-dimensional Discrete Cosine Transform (2D DCT) and the Principal Component Analysis (PCA) algorithm.

In the case of 2D-DCT the observable vectors consist of the DCT coefficients resulted by applying the transform within each block extracted from the image. Due to the compression and de-correlation properties, this transform is suitable in using its coefficients as elements of the observable vectors. The number and especially the order of coefficients play an important role in generating adequate models. Unlike the method of Nefian [10] who uses the 2D-DCT coefficients resulted from a rectangular window extracted from each image block, in this paper we make use of the 2D-DCT coefficients whose values have small variations for images of the same gesture. This fact led to an increase of the recognition rate. For the PCA method, the observable vectors are made up of the Karhunen-Loeve transform (KLT) coefficients. For this transform as well, the properties of compression and correlation make it suitable for observable vectors extraction.

IV. TRAINING THE HAND MODEL

As is the case of neural networks, each HMM can be trained using the Baum-Welch algorithm, applied to a set of images with hand gestures from the database, one set for each gesture. After extracting the blocks from each image of the training set corresponding to a gesture, we apply the method of generating the observable vectors (2D-DCT or KLT). For each training set we obtain a set of observable sequences, each set being subsequently used in training a HMM.

The first step in HMM training consists in initializing parameters $\lambda = (A, B, \Pi)$ as follows: the images from the training set are *uniformly* segmented from top to bottom into N states. The observable vectors associated to each state are used for an initial estimation of the observable distribution probabilities:

$$b_{jm}(o_t) = N(o_t; \mu_{jm}; \Sigma_{jm}) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_{jm}|}} \exp\left(-\frac{1}{2}(o_t - \mu_{jm})' \Sigma_{jm}^{-1} (o_t - \mu_{jm})\right) \tag{6}$$
$$1 \le j \le N, \; 1 \le m \le M$$

The initial values for state transition probabilities and the initial state distribution are given by the left-right model structure.

$$A=\begin{pmatrix} a_{ii} & 1-a_{ii} & 0 & 0 & ... & 0 & 0 \\ 0 & a_{ii} & 1-a_{ii} & 0 & ... & 0 & 0 \\ 0 & 0 & a_{ii} & 1-a_{ii} & ... & 0 & 0 \\ ... & ... & ... & ... & ... & ... & ... \\ 0 & 0 & 0 & 0 & ... & a_{ii} & 1-a_{ii} \end{pmatrix}_{N\times N} \quad (7)$$

where $a_{ii}=1-\frac{1}{d}$, $d=\frac{T}{N}$, T is the number of observable vectors given by (1), and N is the number of states in the model.

$$\Pi=(1 \quad 0 \quad ... \quad 0)_{1\times N} \quad (8)$$

After the model parameters were initialized, the uniform segmentation of the images from the training set is replaced by Viterbi segmentation and the model parameters are recalculated. This step is an iterative one and it ends when the probabilities of the Viterbi segmentation for two successive iterations is lower than a given threshold. The final parameters of the HMM model are obtained using the recursive Baum-Welch algorithm.

Our recognition system comprises 8 HMMs, and each training set contains 20 images of size 112x64 ($H\times W$), and each image contains hand postures from different persons. We stress that the images contain uniform background. In Fig.3 several images from the training sets are shown.

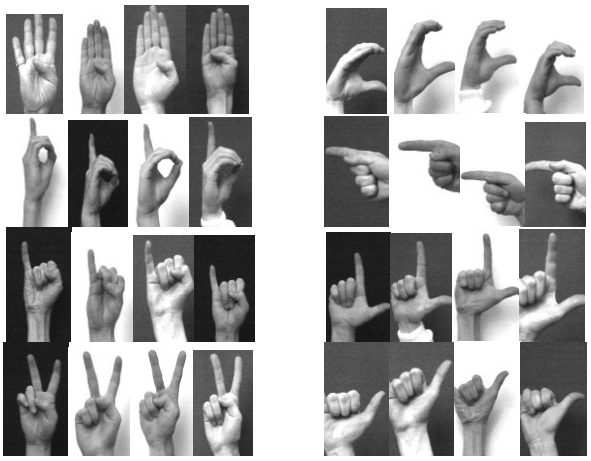

Figure 3. Examples of images use in model training

V. EXPERIMENTAL RESULTS

In this section we present the performances of a recognition system comprising 8 HMM models trained for the recognition of 8 types of hand gestures. The system is tested on a set of test images containing 40 gesture postures, with 5 images for each gesture. The images used in testing were not used during the training, and their background is similar to that of the training images. The algorithm was written in Matlab and the utilized database was downloaded from the site http://www.idiap.ch/resource/gestures/.
The recognition rate is calculated as follows:

$$\text{Rate recognition}=\frac{\text{No. of correctly classified hand gestures}}{\text{Total no. of hand gestures}}\times 100\% \quad (9)$$

Since the system performances depend significantly on the training parameters i.e., the block width *L* and the overlap *P* between two consecutive blocks, the number of model states and the size of the observable vectors (number of 2D-DCT or KLT coefficients) we performed a search for their most appropriate values in order to get the best recognition rate.

Fig. 4 displays the models performances for the *L* and *P* variation, while the other the parameters have arbitrarily fixed values: number of states N = 7; number of coefficients= 40.

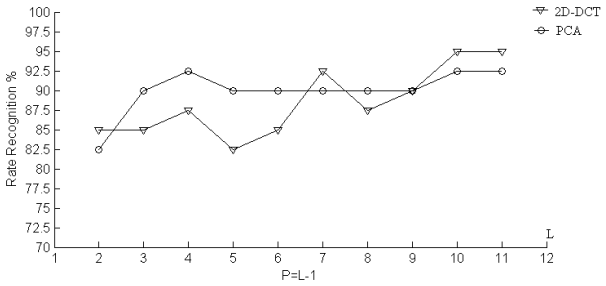

Figure 4. Performance of models for the variation of L and P

From the above results we notice that for both methods of generating the observable vectors the best performances are reached for L = 10 and P = 9. Next we set the values L = 10, P = 9, N = 7 and we vary the number of coefficients. The obtained results are shown in Fig.5.

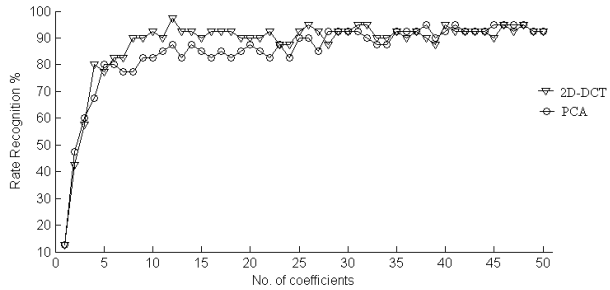

Figure 5. Performance of models for the variation of the number of coefficients

This time we remark that the two methods lead to different system performance; thus for the 2D-DCT method the optimal number of coefficients is nr_cf=12, while for PCA is nr_cf= 38.

Finally we study how the system performances change while varying the number of states. The results are presented in Fig.6.

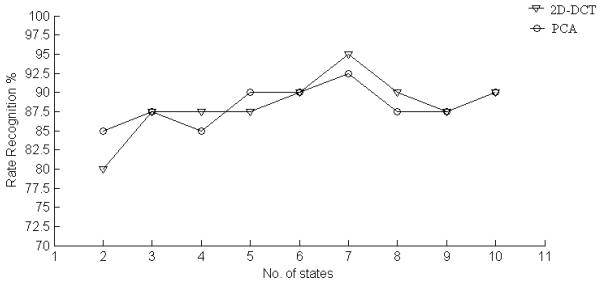

Figure 6. Performances of models at the variation of the number of model states

We notice in this case the same evolution for both methods, the best performances being obtained for a number of states N = 7.

As a final conclusion we mention that the Baum-Welch – Viterbi algorithm sets an optimal number of states, number of coefficients and sizes of observable sequences (L and P), in order to obtain the best performance of the recognition system. Corresponding to our database these parameters are: L = 10, P = 9, nr_cf = 12 for 2D-DCT, nr_cf = 38 for PCA and N = 7.

VI. CONCLUSIONS

In this paper we presented the performances of a hand gesture recognition system with 8 HMM models, using two image compression methods for generating the observable vectors. The simulations showed that the two methods yield almost similar results, with an advantage of the 2D-DCT method which uses less coefficients and the recognition rate is higher; nevertheless, both methods – 2D-DCT and PCA are suitable for obtaining good performance. For high recognition rates, besides the database which has an important role, the following training parameters must also be taken into account: the width *L* of the blocks and the overlapping *P* between two consecutive blocks, the number of model states and the size of the observable vectors. (number of 2D-DCT or KLT coefficients). From the obtained results we draw the following conclusions:

- the optimal number of states is 7;
- the exploration band from which the observable vector is extracted must satisfy the condition L=W/10, and the overlapping should be at least L-1;
- the optimal number of 2D-DCT coefficients is 12, respectively 40 KTL coefficients;

These conclusions are of course valid strictly for our database. If other database is used, these parameters will change within certain limits.

As further work we intend to study the performances of these models when the test images present illumination variations or are affected by various types of noise.

ACKNOWLEDGEMENT

The authors gratefully acknowledge financial support offered by Tessera Romania (VT), Eurodoc ID 59410 (RLV) and Higher Education Scientific Research National Council, PN2 – ID_310 (LG).

REFERENCES

[1] J. S. Lipscomb, *A trainable gestures recognizer*, Pattern Recognition, vol. 24, No.9, pp. 895 – 907, 1991.

[2] W. M Newman, R. F. Sproull, *Principles of interactive computer graphics*, McGraw-Hill, 1979.

[3] D. H. Rubine, *The automatic recognition of gestures*, Ph.D dissertation, Computer Science Department, Carnegie Mellon University, December 1991.

[4] K. S. Fu, *Syntactic recognition in character recognition*, Volume 112 of Mathematics in Science and Engineering, Academic Press, 1974.

[5] S. S. Fels, G. E. Hinton, *Glove – talk: a neural network interface between a data-glove and speech synthesizer*, IEEE Trans. Neural Networks, vol. 4, No. 1, pp. 2-8, 1993.

[6] X. Deyou, *A network approach for hand gesture recognition in virtual reality driving fraing system of SPG*, In International Conferences on Pattern Recognition, pp. 519-522, 2006.

[7] E. Holden, R. Owens, G. Roy, *Hand movement classification using adaptive fuzzy expert system*, Journal of Expert Systems Research and Application, vol. 9, pp. 465-480, 1996.

[8] S. Rajko, G. Qian, T. Ingalls, J. James, *Real time gestures recognition with minimal training requirements and on-line learning*, Computer Vision and Pattern Recognition 2007, IEEE Conferences on pages 1-8, 2007.

[9] L. R. Rabiner, *A tutorial of hidden Markov models and selected applications in speech recognition*, Proceeding of the IEEE, pp. 275-286, January 1989.

[10] Ara V. Nefian, *A hidden Markov model – based approach for face detection and recognition*, A Thesis, August 1999.

[11] I. Ciocoiu, V. Grigoras, *Tehnici moderne de procesarea semnalelor*, Editura Cermi, 2005.