# Face Recognition Using A DCT-HMM Approach

Vinayadatt V. Kohir* and U. B. Desai
Department of Electrical Engineering
Indian Institute of Technology
Mumbai, INDIA. 400 076
email: {vvkohir,ubdesai}@ee.iitb.ernet.in

## Abstract

*A transform domain approach coupled with Hidden Markov Model (HMM) for face recognition is presented. JPEG kind of strategy is employed to transform input sub-image for training HMMs. DCT transformed vectors of face images are used to train ergodic HMM and later for recognition. ORL face database of 40 subjects with 10 images per subject is used to evaluate the performance of the proposed method. 5 images per subject are used for training and the rest 5 for recognition. This method has an accuracy of 99.5%. The results, to the best of knowledge of the authors, give the best recognition percentage as compared to any other method reported so far on ORL face database.*

## 1 Introduction

The present work on face recognition is carried out from the statistical analysis stand point. Inherent advantages of DCT (Discrete Cosine transform) and HMM (Hidden Markov Model) are exploited to get better performance. The technique is motivated by the work of Samaria and Young [8, 9].

Our face recognition system has two steps: First step is to build (or train) HMMs with the training face images. For, every subject to be recognized (i) DCT coefficients are computed for an $x \times y$ overlapping sub-image of all the training face images (see Fig.1 and Fig.2). (ii) Using DCT coefficients an HMM is trained to model the face image. In the second step (see Fig.3), i.e. recognition step, the unknown face image DCT coefficients are computed as in the training step. These coefficients are then used to compute the likelihood function; namely, $P[O, Q/\lambda_i]$. The recognized face corresponds to that $i$ which yields a maximum for $P[O, Q_i^*/\lambda_i]$, where $Q_i^* = \arg\max_Q P[O, Q/\lambda_i]$. We obtain recognition performance of 99.5% with ORL (Olivetti Research Laboratory) face database The training step is performed only once, and can be updated by training a new HMM for a new subject to be recognized.

In the following Subsection related previous work is described. Section 2 gives a brief description on HMM and DCT. Section 3 presents the proposed approach. Experimental results are presented in Section 4.

### 1.1 Previous Work

Substantial effort has gone into understanding the face recognition problem and developing algorithms for face recognition. A detailed survey is provided in [1, 10]. Most of the algorithms exploit either template matching, Eigenfaces [11], artificial neural networks [3] or probabilistic models like HMM [8, 9], n-tuple classifier [5] or a combination of these paradigms [4].

Eigenface based face recognition scheme of Turk and Pentland [11] treats an image as a 2D random signal, and its the covariance matrix is decomposed to extract principal components (Eigenfaces). The scheme assumes that face images and images other than face, are disjoint. Authors report an accuracy in the range of 64% to 96%. The test set is of 2500 face images (MIT face database) of 16 individuals digitized under different lighting conditions, head orientations and resolutions.

In an another approach Samaria and Young [8] adopt, a top down HMM model for a face. Their scheme uses grey level pixel values of the face image for training and recognition. With a test set of 50 face images of 24 individuals, success rate reported is 84%. All the training and test images were of the same size with no background. In the modified approach [9] authors use pseudo 2D HMM, and the recognition rate for 40 subjects (5 images per subject for training and another 5 each for recognition) recognition accuracy is reported to be 95.5%. The ORL face database is used for the experimentation.

In yet another method [5] n-tuple classifier based on random memory access concept, is used to obtain recognition accuracy in the range of 74% to 97% depending on the number images per subject used to train the classifier. Here, each image is sampled into a sparse set of n-tuples, where each

*On deputation from P.D.A. College of Engg. Gulbarga, Karnataka n-tuple defines a set n locations in the image space. Recognition is achieved by computing the distance between the stored n-tuples and the n-tuple generated by the image to be recognized. Here too, the ORL face database is used.

# 2 Background

Briefly, basic principles of HMM and DCT are hi-lighted in this section.

## 2.1 Hidden Markov Model (HMM)

Though HMMs are 1D data modelers, in the recent past, they have been used in vision: texture segmentation [12], face finding [7], object recognition [2], face recognition [8]. For an exposure to HMMs the reader may refer to [6].

Every HMM is associated with non-observable (hidden) states, and an observable sequence generated by the individual hidden states. HMM, $\lambda$ is characterized by three parameters $(A, B, \Pi)$. Let $O = (o_1, o_2, \ldots, o_T)$; where each $o_t$ is a $D$-element vector, be the observed sequence at $T$ different observation instances and corresponding state sequence be $Q = (q_1, q_2, \ldots, q_T)$, where $q_t \in \{1, 2, \ldots, N\}$ , N being the number of states in the model.

The HMM parameters $\lambda = (A, B, \Pi)$ are defined as follows :

$A$: is the transition probability matrix. The elements of $A$ are:

$$a_{ij} = P[q_{t+1} = j / q_t = i] \tag{1}$$

$B$: is the emission probability matrix determining the output observation given that the HMM is in a particular state. Every element of this matrix:

$$b_j(o_t): \quad 1 \leq j \leq N, \text{ and } 1 \leq t \leq T \tag{2}$$

is the posterior density of observation $o_t$ at time $t$ given that HMM is in state $q_t = s_j$.

$\Pi$: is the initial state distribution matrix with $i$-th entry

$$\pi_i = P[q_1 = i] \tag{3}$$

being the probability of being in state $i$ at the start of the observation.

## 2.2 DCT

The 2D, $N \times N$ DCT is defined as

$$C(u,v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) \cdot \cos(\frac{(2x+1)u\pi}{2N}) \cos(\frac{(2y+1)v\pi}{2N}) \tag{4}$$

and inverse DCT is

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u)\alpha(v) C(u,v) \cdot \cos(\frac{(2x+1)u\pi}{2N}) \cos(\frac{(2y+1)v\pi}{2N}) \tag{5}$$

where,

$$\alpha(w) = \begin{cases} \frac{1}{\sqrt{N}} & : \; for\ w = 0 \\ \frac{2}{\sqrt{N}} & : \; for\ w = 1, 2, \ldots, N-1 \end{cases} \tag{6}$$

The DCT transforms spatial information to decoupled frequency information in the form of DCT coefficients. Also it exhibits excellent energy compaction.

# 3 DCT-HMM Approach

Conventional HMMs are 1D data modelers. To facilitate them to be useful for 2D image applications, either they are to be redefined in 2D context, or 2D data must be converted to 1D, without losing significant information.

In the present context, the second approach is used, i.e. 2D face image data is converted to 1D sequence, and then used for HMM training and testing. This crucial step of converting 2D image information to 1D vector is performed via data generation step discussed in the following section.

## 3.1 Data Generation

Data generation consists of two steps. In the first step, a sub-image of the face image is picked. In the next step, the DCT coefficients of the sub-image are computed. These coefficients are then used to form an observation vector. This procedure is repeated on a sequence of sub-images.

The sequence of sub-images is generated by sliding a square window over the image in raster scan fashion from left to right with predefined overlap. Fig. 1 shows the sampling of a hypothetical image. At every position of the window over the image, the pixel values of the image are captured. These pixels represent a sub-image of the original image. 2D DCT is computed for each of the sub-images. Only few DCT coefficients are retained by scanning the DCT matrix in zig-zag fashion (Fig.2) which is analogous to that of JPEG / MPEG image coding. The zig-zag scanned DCT coefficients are arranged in a vector form as per the scan (Fig.2) i.e. 1st DCT coefficient scanned will be the first element of the vector, and the next element will be the next scanned DCT coefficient, and so on. This vector of DCT coefficients becomes the 1D observation vector for HMM training or testing.

For every position of scan window we have a corresponding observation vector. The dimension of the observation vector is determined by the number of DCT coefficients retained. This number is initially chosen based on some heuristics, and it can be frozen after experimentation. Each image gives an observation sequence, whose length is determined by the size of the sliding window, the amount of overlap between adjacent windows and the size of the image.

The 2D face image is now represented by a corresponding 1D DCT domain observation sequence, which is suitable for HMM training or testing.

### 3.2 HMM for Face Recognition

To do face recognition, for every subject to be recognized, a separate HMM is trained i.e. to recognize M subjects we have M distinct HMMs at our disposal.

An ergodic HMM is employed for the present work. The training of each of the HMM is carried out using the DCT based training sequence.

**HMM Training:** Following steps give a procedure of HMM training.

*Step* 1: Cluster all $\Omega$ training sequences, generated from $\Omega$ number of face images of one particular subject, i.e. $\{O^{\omega}\}$, $1 \leq \omega \leq \Omega$, each of length $T$, in $N$ clusters using some clustering algorithm, say k-means clustering algorithm. Each cluster will represent a state of the training vector. Let them be numbered from 1 to $N$.

*Step* 2: Assign cluster number of the nearest cluster to each of the training vector. i.e. $t^{th}$ training vector will be assigned a number $i$ if its distance, say Euclidean distance, is smaller than than its distance to any other cluster $j,\ j \neq i$.

*Step* 3: Calculate mean $\{\mu_i\}$ and covariance matrix $\{\Sigma_i\}$ for each state (cluster).

$$\mu_i = \frac{1}{N_i} \sum_{o_t \in i} o_t \ \text{ for } 1 \leq i \leq N \tag{7}$$

$$\Sigma_i = \frac{1}{N_i} \sum_{o_t \in i} (o_t - \mu_i)^t (o_t - \mu_i) \ \text{ for } 1 \leq i \leq N \tag{8}$$

where $N_i$ is the number of vectors assigned to state $i$.

*Step* 4: Calculate $A$ and $\Pi$ matrices using event counting.

$$\pi_i = \frac{\text{No. of occurrences of } \ o_1 \in i}{\text{No. of training sequences} = \Omega} \quad \text{for } 1 \leq i \leq N \tag{9}$$

$$a_{ij} = \frac{\text{No. of occurrences of } o_t \in i \text{ and } o_{t+1} \in j}{\text{No. of occurrences of } o_t \in i} \tag{10}$$
$$\text{for } 1 \leq i, j \leq N \text{ and for } 1 \leq t \leq T-1$$

*Step* 5: Calculate the $B$ matrix of probability density for each of the training vector for each state. Here we assume that $b_j(o_t)$ is Gaussian. For $1 \leq j \leq N$

$$b_j(o_t) = \frac{1}{(2\pi)^{\frac{D}{2}} |R|^{\frac{1}{2}}} exp[-\frac{(o_t - \mu_i)^t \Sigma_j^{-1} (o_t - \mu_i)}{2}] \tag{11}$$

where, $o_t$ is of size $D \times 1$.

*Step* 6: Now use the Viterbi algorithm to find the optimal state sequence $Q^*$ for each training sequence. Here, the state reassignment is done. A vector is assigned state $i$ if $q_t^* = i$.

*Step* 7: The reassignment is effective only for those training vectors whose earlier state assignment is different from the Viterbi state. If there is any effective state reassignment, then repeat Steps 3 to 6; else STOP and the HMM is trained for the given training sequences.

The details of Viterbi algorithm can be found in [6].

**Face Recognition:** Once, all the HMMs are trained, then we can go ahead with recognition. For the face image to be recognized, the data generation step is followed as described in Section 3.1. The trained HMMs are used to compute the likelihood function as follows: Let $O$ be the DCT based observation sequence generated from the face image to be recognized,

1: Using the Viterbi algorithm, we first compute

$$Q_i^* = \arg\max_{Q} P[O, Q/\lambda_i] \tag{12}$$

2: The recognized face corresponds to that $i$ for which the likelihood function $P[O, Q_i^*/\lambda_i]$ is maximum.

## 4 Experimental Evaluation

ORL face database is used in the experiments. There are 400 images of 40 subjects - 10 poses per subject. All the face images have are of size $92 \times 112$ with 256 grey levels. The face data has are male or female subjects, with or without glasses, with or without beard, with or without some facial expression. Pentium 200Mhz. machine in multiuser environment was used to evaluate our strategy.

**Training:** The training set (see Fig. 4) chosen has 5 different poses of each subject implying the number of training sequences per HMM to be 5. For every subject a HMM is trained as follows:

1: Sample the image with a square $16 \times 16$ window. Take DCT of the sub-image enclosed by this window. Scan the DCT matrix in a zig-zag fashion and select only few significant DCT coefficients, say 15. Arrange them in a vector form. Complete scanning of the face image from top-left corner down upto right-bottom of the image with 75% overlap. This gives an observation sequence.

2: Repeat Step 1 for all the training images. This step gives the set of observation sequences for HMM training.

3: Use the training algorithm described earlier to train the 5 state HMM.

**Recognition** The recognition test is performed on the remaining 200 images which are not the part of training. Sample test images are in Fig. 5. Each image is recognized separately following the steps outlined below.

1: For the input image, generate the DCT based observation sequence. Note that window size, amount of overlap and number of DCT coefficients is exactly same as in the training phase.

3: Use the Viterbi algorithm to decode the state sequence and find the state-optimized likelihood function for all the stored HMMs, namely, $\forall\ i,\ P[O, Q_i^*/\lambda_i]$, where $Q_i^* = \arg\max_Q P[O, Q/\lambda_i]$, and $O$ is the dct-based observation sequence corresponding to the image to be recognized.

4: Select that label of the HMM for which the likelihood function is maximum.

By varying the size of sampling window, percentage overlap and number of DCT coefficients, experiments are carried out with the same set of images to test the efficacy the proposed scheme.

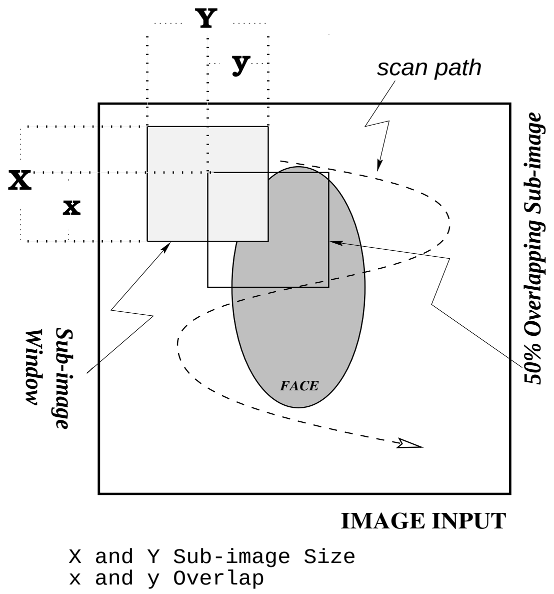
The highest recognition score achieved is 99.5% i.e. only one image (shown in Fig. 5 with a 'X' on it) out of 200 images is misclassified with a sampling window of $16 \times 16$ with an overlap of 75% using 10 significant DCT coefficients. Nevertheless, the recognition varies from 74.5% to 99.5%, depending on the size of sub-image, amount of overlap and the number of DCT coefficients. Table 1 gives the detailed experimental evaluation. Results tabulated are for sub-image of size $8 \times 8$ and $16 \times 16$ with 50% and 75% overlap with 10, 15 and 21 DCT coefficients.

Comparative results of some of the recent schemes applied on ORL face database are given in Table 2. Note that, the proposed scheme gives much better recognition percentage than any of the methods proposed. The results indicated are borrowed from the papers published by respective authors, which are indicated by the reference numbers in square braces. Our scheme is significantly slow, nevertheless there is price for better recognition, namely in terms of time. Though, it does not make much sense to compare timings obtained on different platforms, nevertheless the timings indicated in Table 2 give some idea about comparative speed of recognition.

## 5 Conclusions

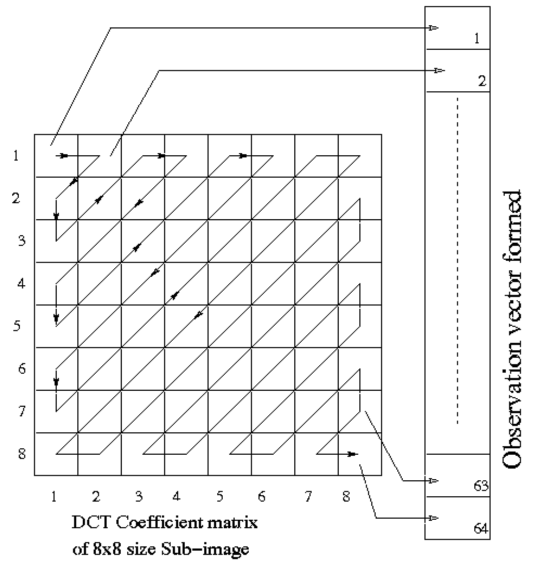
For the face data set considered, the results of the proposed DCT-HMM based scheme show substantial improvement over the results obtained from some other methods.

We are currently investigating the robustness of proposed scheme with respect to noise and possible scale variations.




-
-

**Figure 1. Sampling scheme to generate Sub-images.**




-
-

**Figure 2. Scheme of converting 2D DCT coeffecients to 1D vector.**

| Experimental Results | | Subimage size | | | |
|---|---|---|---|---|---|
| | | $8 \times 8$ | | $16 \times 16$ | |
| | | 50% Overlap | 75% Overlap | 50% Overlap | 75% Overlap |
| 10 DCT coeffs. | Train time | 30.10 s | 29.85 s | 05.35 s | 23.00 s |
| | Test time | 04.70 s | 08.71 s | 03.23 s | 06.84 s |
| | Recognition | 82% | 87.5% | 88.5% | 99.5% |
| 15 DCT coeffs. | Train time | 39.23 s | 30.78 s | 05.65 s | 33.38 s |
| | Test time | 06.90 s | 02.05 s | 04.37 s | 07.01 s |
| | Recognition | 83.5% | 89% | 86.5% | 99% |
| 21 DCT coeffs. | Train time | 60.00 s | 65.43 s | 15.43 s | 67.38 s |
| | Test time | 10.26 s | 02.25 s | 12.09 s | 10.58 s |
| | Recognition | 80% | 89.5% | 74.5% | 97% |

-
-

**Table 1. Results of experimentation for the proposed DCT-HMM based face recognition scheme.**

| Method | % Recognition | Training time | Recognition time |
|---|---|---|---|
| SOM+CN [3] | 96 | 4hours | 0.5 sec. |
| PDBNN [4] | 96 | 20min | $\leq$0.1 sec. |
| n-tuple [5] | 86 | 0.9 sec. | 0.025 sec. |
| cont. n-tuple [5] | 96 | 0.9 sec. | 0.025 sec. |
| Top-Down HMM [8] | 84 | n/a | n/a |
| Pseudo-2D HMM [8] | 95 | n/a | 240 sec. |
| Eigenface | 88 | 9.4 sec. | 0.62 sec. |
| DCT-HMM (75% overlap, 10 coeffs.) | 99.5 | 23.5 sec. | 3.5 sec. |

-
-

**Table 2. Comparative recognition results of some of the other methods as reported by the respective authors on ORL face database.**

## References


[1] R. Chellappa, C.L. Wilson and S.A. Sirohy, ‘Human and Machine Recognition of Faces: A Survey’ ,*IEEE Proceedings*, Vol. 83, No. 3, pp. 704-740, May 1995.

[2] Joachim Hornegger, Heinrich Niemann, D.Pulus and G. Schlottke, ‘Object Recognition using hidden Markov Models’, *Proc. of Pattern Recognition in Practice*, pp. 37-44, 1994.

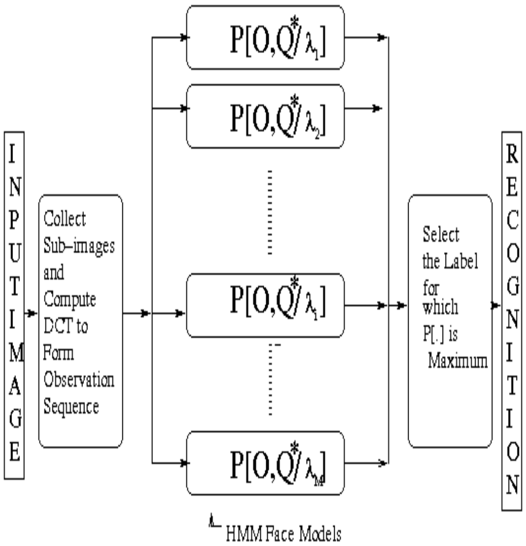
[3] S. Lawarance, C.L. Giles, A.C. Tsoi and A.D. Back, ‘Face Recognition : A Convolutional Neural-Network Approach’, *IEEE Trans. on Neural Networks*, Vol. 8, No. 1, pp. 98-113, Jan. 1997.

[4] S.H. Lin, S.Y. Kung and L.J.Lin, ‘Face Recognition/Detection by Probabilistic Decision Based Neural Network’, *IEEE Trans. on Neural Networks*, Vol. 8, No. 1, pp. 114-132, Jan. 1997.

[5] S.M. Lucas, ‘Face Recognition with the Continuous n-tuple Classifier’, *BMVC97*, 1997.

[6] L. R. Rabiner, ‘A tutorial on Hidden Markov Models and Selected Applications in Speech Recognition’, *IEEE Proceedings*, Vol. 77, No. 2, pp. 257-285, 1989.

[7] A.N. Rajagopalan, K. Sunil Kumar, Jayashree Karlekar, Manivasakan, Milind Patil, U.B. Desai, P.G. Poonacha, S. Choudhuri, ‘Finding Faces in Photographs’, *ICCV98*, pp. 640-645, Jan. 1998.




-
-

Figure 3. Face Recognition Scheme.

-
-

Figure 4. Sample ORL training face images. Each row represents the face images of the subject used for HMM training.

-
-

Figure 5. Sample ORL test face images. The face image with 'X' on it, is the only misrecognised face image out of 200 test face images for the recognition rate of 99.5%.


[8] F.S. Samaria and S. Young, 'HMM-based Architecture for Face Identification', *Image and Vision Computing*, Vol. 12, No. 8, pp. 537-543, Oct. 1994.

[9] F.S. Samaria, 'Face Recognition Using Hidden Markov Models', *Ph.D. Thesis*, University of Cambridge, UK, 1994.

[10] Theme Section - Face and Gesture Recognition, *IEEE Pattern Recognition and Machine Intelligence*, Vol. 19, No. 5, July 1997.

[11] M. Turk and A. Pentland, 'Eigenfaces for Recognition', *Journal of Neuroscience*, Vol. 3, No. 1, MIT 1991.

[12] Yang He and Amalan Kundu, 'Unsupervised Texture Segmentation Using Multichannel Decomposition and Hidden Markov Models', *IEEE Image Processing*, Vol. 4, No. 5, pp.603-619, May 1995.