Dynamic Training using Multistage Clustering for Face Recognition

Marios Kyperountas[a,,1], Anastasios Tefas[a,b], and Ioannis Pitas[a]*

[a] Aristotle University of Thessaloniki
Department of Informatics
Artificial Intelligence and Information Analysis Laboratory
Box 451, 54006 Thessaloniki, Greece
Email: {mkyper@aiia.csd.auth.gr , tefas@aiia.csd.auth.gr, pitas@aiia.csd.auth.gr }

[b] Technological Educational Institute of Kavala
Department of Information Management
Kavala 65404, Greece

* Corresponding author:

[1] Present address:
40 S. Patterson Avenue #207
Santa Barbara, CA 93111
Email: mkyper@aiia.csd.auth.gr

Abstract

A novel face recognition algorithm that uses dynamic training in a multistage clustering scheme is presented and evaluated. This algorithm uses discriminant analysis to project the face classes and a clustering algorithm to partition the projected face data, thus forming a set of discriminant clusters. Then, an iterative process creates subsets, whose cardinality is defined by an entropy-based measure, that contain the most useful clusters. The best match to the test face is found when only a single face class is retained. This method was tested on the ORL, XM2VTS and FERET face databases, whereas the UMIST database was used in order to train the proposed algorithm. Experimental results indicate that the proposed framework provides a promising solution to the face recognition problem.

Index Terms: Face recognition, dynamic training, multilevel clustering, discriminant analysis.

1. Introduction

Face recognition (FR) is an active research field that has received great attention in the past several years. A face recognition system usually attempts to determine the identity of the test face by computing and ranking all similarity scores between the test face and all human faces stored in the system database that constitute the training set. However, the performance of many state-of-the-art FR methods deteriorates rapidly when large, in terms of the number of faces, databases are considered [1, 2]. Specifically, the facial feature representation obtained by methods that use linear criteria, which normally require images to follow a convex distribution, is not capable of generalizing all the introduced variations due e.g. to large differences in viewpoint, illumination and facial expression, when large data sets are used. When nonlinear face representation methods are employed, problems such as over-fitting, computational complexity and difficulties in optimizing the involved parameters often appear [1]. Moreover, the performance of face recognition methods deteriorates when there is lack of a sufficiently large number of training samples for each face in the database as, in this case, the intra-person variations cannot be modelled properly. More specifically, linear methods, such as linear discriminant analysis (LDA), often suffer from the small sample size (SSS) problem, where the dimensionality of the samples is larger than the number of available training samples [3].

Recently, various methods have been proposed in order to restrict the maladies that are imposed by the two aforementioned types of problems on the recognition performance. The ‘divide and conquer’ principle, by which a database is decomposed into smaller sets in order to piecewise learn the complex distribution by a mixture of local linear models, has been widely used. In [1], a separability criterion is employed to partition a training set from a large database into a set of smaller maximal separability clusters (MSCs) by utilizing an LDA-like technique. Based on these MSCs, a hierarchical classification framework that consists of two levels of nearest neighbour classifiers is employed and the match is found. The work in [4] concentrates on the hierarchical partitioning of the feature spaces using hierarchical discriminant analysis (HDA). A space-tessellation tree is generated using the most expressive features (MEF), by employing Principal Component Analysis (PCA), and the most discriminating features (MDF), by employing LDA, at each tree level. This is done to avoid the limitations linked to global features, by deriving a recursively better-fitted set of features for each of the recursively subdivided sets of training samples. In general, hierarchical trees have been extensively used for pattern recognition purposes.

LDA is an important statistical tool that has been shown to be effective in face recognition or verification problems [5, 6]. Traditionally, in order to improve LDA-based methods and provide solutions for the SSS problem, LDA is applied in a lower-dimensional PCA subspace, so as to discard the null space (i.e., the subspace defined by the eigenvectors that correspond to zero eigenvalues) of the within-class scatter matrix of the training data set [5]. However, it has been shown [7] that significant discriminant information is contained in the discarded space and alternative solutions have been sought. Specifically, in [8] a direct-LDA (DLDA) algorithm is presented that discards the null space of the between-class scatter matrix, which is claimed to contain no useful information, rather than discard the null space of the within-class scatter matrix. More recently, in an attempt to address the SSS problem, the regularized LDA method (RLDA) was presented in [9], which employs a regularized Fisher's separability criterion. The purpose of regularization is to reduce the high variance related to the eigenvalue estimates of the within-class scatter matrix, at the expense of potentially increased classification bias.

The use of static training structures, where the input data is not involved in determining the system parameters, has been abundant when designing pattern classification systems. However, it has been demonstrated that the classification performance can be improved by employing dynamic training structures. In this spirit, the Dynamic face recognition Committee Machine (DCM) was presented in [10], consisting of five state-of-the-art pattern classification algorithms. The proposed dynamic structure requires for the input to be directly involved in the combining mechanism that employs an integrating unit to adjust the weight of each expert according to the input. A gating network is used to identify the situation that the input image is taken and assign particular weights to each expert. Experimental results indicate that using this dynamic structure gives higher recognition rates rather than using a static one where the weights for each expert are fixed. In [11], the authors derive an owner-specific LDA-subspace in order to create a personalized face verification (2-class classification) system, where the owner identity is the true identity. The training set is partitioned into a number of clusters and the cluster that contains face data that are most similar to the owner face is identified. The system assigns the owner training images to this particular cluster and this new data set is used to determine an LDA subspace that is used to compute the verification thresholds and matching score, when a test face claims the identify of the owner. The authors show that verification performance is enhanced when owner-specific LDA-subspaces are utilized, rather than using the LDA space created by processing the entire training set.

This paper presents a novel framework that uses Dynamic Training in a Multistage Clustering process that employs discriminant analysis. For notation compactness, this algorithm shall be referred to as DTMC throughout the rest of this paper. This methodology is not restricted to face recognition, but is able to deal with any problem that fits into the same formalism. At this point, it is imperative that two terms that are frequently used in this paper are defined: ‘class’ refers to a set of face images from the same person, whereas ‘cluster’ refers to a set of classes.

Initially, facial feature extraction is carried out by making use of the multilevel 2-D wavelet decomposition (MWD2) algorithm [12, 13], which provides dimensionality reduction and its use has been shown to be appropriate for classification purposes [6, 14, 15]. Then, the training and test face feature vectors are projected onto a MDF-space that is created by employing the RLDA method of [9]. Subsequently, the *k*-means algorithm is used to partition the training data into a set of discriminant clusters. The distance of the test face from the cluster centroids is used to collect a subset of clusters that are closest to the test face. The cardinality of this subset is set through an entropy-based measure that is calculated by making use of the discrete probability histogram. Then, a new MDF-space is created from this cluster subset with its dimensions set so as to reduce classification problems that stem from possible large variations in the set of images of each face class. The training data projected to this new space are again clustered and a new subset that is closer to the test face is selected. This process is repeated in as many iterations as necessary, until a single cluster is selected that contains just one face class. The identity of this face class is set as the best match to the identity of the test face.

The proposed method is computationally efficient, compared to ‘divide and conquer’ techniques such as the one in [1] where multiple classification results are produced by applying an individual discriminant analysis process and a nearest-neighbour classifier to each cluster. Our method uses a single discriminant analysis operation at each clustering level, with the number of clustering levels being generally much smaller than the number of clusters since only a small subset of the training data is retained at each level. A heavy computational cost also accompanies algorithms that construct hierarchical trees or space tessellation, as is the case with using the HDA algorithm in [4]. The purpose of this type of algorithms is to provide a manageable discriminant solution for each and every face class by recursively subdividing the complete set of training samples into smaller classification problems. On the other hand, at each clustering step our algorithm only has to provide a discriminant solution for the face classes that are closer to the test face; the training data that correspond to the remaining face classes are discarded.

The structure of the DTMC algorithm is flexible to the adding of new training faces. Specifically, when a new training face is added to the database the only change needed in the DTMC process is to increase the dimension of the first MDF-space by one. The characteristics of the test face will determine which set of clusters, which may or may not contain the new face class, will be retained for the clustering level that follows. On the contrary, the hierarchical tree structure requires a complete re-learning of the full training space since the new MDF space at the first tree level may lead to an entirely different decomposition result.

The MDF-spaces that the hierarchical tree or space tessellation structures utilize are generated in the learning phase and are not biased by the characteristics of the test face. On the contrary, the MDF-spaces created at each clustering level of the DTMC algorithm are indeed biased with respect to the characteristics of the test face. Based on the conclusions of [10] and [11] that have been summarized above, more accurate classification results are to be expected by DTMC since it employs a dynamic classification structure that utilizes a series of test-face-specific subspaces.

The outline of this paper is as follows: Section 2 describes the feature extraction method that utilizes the MWD2 algorithm, reviews the RLDA method that is used to extract the MDF-spaces before each clustering process and presents the *k*-means algorithm that is used to partition the training data as well as the entropy-based measure that is used to define the number of clusters that are retained. Section 3 describes the complete DTMC face recognition methodology that is proposed in this paper. Experimental results are reported in Section 4, where the DTMC methodology is tested using the well-established UMIST [16], ORL [17], and XM2VTS [18] databases in order to assess its recognition capabilities on standard data sets. Moreover, the performance of DTMC is compared to a number of FR algorithms that have been recently proposed by the research community.

2. Feature Selection and the DTMC Building Blocks

This section briefly describes how the MWD2 algorithm is utilized to extract features from the face images at a selected decomposition level. In addition, the RLDA and *k*-means algorithms that DTMC uses are briefly reviewed. Finally, the entropy-based measure that is used at each clustering level to select a subset of the training data is presented.

2.1 Feature Selection using MWD2

A proper wavelet transform can result in robust image representations with regard to illumination changes and be capable of capturing substantial facial features, while keeping computational complexity low [14]. The structure of the MWD2 algorithm that is employed in the feature extraction step of our algorithm in order to produce a multi-resolution image representation is described in detail in [12, 13]. An analysis filter bank that usually consists of a low-pass filter, Lo_D, and a high-pass filter, Hi_D, is utilized to decompose the signal into its low frequency component and its high frequency components at three different orientations [19].

The maximum decomposition level J_d of a signal is related to the signal's highest resolution level J by $J_d = J - j$, where j is the current resolution level of the signal. In this paper, the criterion that is used to define J_d requires that at least one coefficient of the convolved output is calculated properly, bearing in mind that the convolved output is down-sampled by a factor of 2 at each scale. Thus, the following should be satisfied: $2^{J_d}\left(\max\left(N_{Hi\_D}, N_{Lo\_D}\right)-1\right) < \min\left(N_v, N_h\right)$, where N_v and N_h are the vertical and horizontal dimensions of the 2-D signal f, and N_{Hi_D} and N_{Lo_D} are the lengths of the filter kernels Hi_D and Lo_D, respectively. J_d is calculated by

$$J_d = floor\left(\frac{\log_2\left(\frac{\min\left(N_v, N_h\right)}{\max\left(N_{Hi\_D}, N_{Lo\_D}\right)-1}\right)}{\log_2(2)}\right). \quad (1)$$

Earlier studies showed that the low resolution components of a wavelet decomposition are the most informative for face classification purposes [6]. In [20] it was concluded that facial expressions and small occlusions affect the image intensity manifold locally, which, under frequency-based representation, shows that only the high-frequency spectrum is affected. Similarly, in [21] it was shown that the effect of different facial expressions can be attenuated by removing the high-frequency components. As a result, the wavelet coefficients that correspond to the lowest-frequency band at scale J_d (or equivalently at resolution level $J = 0$), $f \prec A_0$, are selected as the set of features that the DTMC algorithm will process. The spline biorthogonal wavelet 'bior3.5' [13] is used to define the coefficients of the analysis filter bank (FIR) filters, Lo_D and Hi_D.

2.2 Finding MDF-spaces using RLDA

The RLDA method uses the following regularized Fisher's discriminant criterion, which is particularly robust against the SSS problem compared to the original one [9]:

$$\mathbf{W}_{\text{o}} = \arg\max_{\mathbf{W}} \frac{\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}}{\left|R\left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_b\mathbf{W}\right) + \left(\mathbf{W}^{\mathrm{T}}\mathbf{S}_w\mathbf{W}\right)\right|}, \tag{2}$$

where $\mathbf{S}_b$ is the between-class scatter matrix, $\mathbf{S}_w$ is the within-class scatter matrix and $0 \leq R \leq 1$ is a parameter that controls the strength of regularization. The RLDA algorithm is described in detail in [9]. The purpose of regularization is to reduce the high variance related to the eigenvalue estimates of $\mathbf{S}_w$, at the expense of potentially increased bias of the estimation of $\mathbf{W}$. The determination of the optimal value for R is computationally demanding, as it is based on exhaustive search [9]. In this work, an approximation of this optimal value is found, at each clustering level, by using data from the UMIST database.

2.3 The k-means Clustering Method

Given a set of N data vectors, realized by $\mathbf{y}_n$, $n = 1, \ldots N$, in the *d*-dimensional space, *k*-means is used to determine a set of K vectors in $\Re^d$, called cluster centroids, so as to minimize the sum of vector-to-centroid distances, summed over all K clusters [22, 23]. The objective function of *k*-means that is used in this paper employs the squared Euclidean distance and is presented in [22]. After the cluster centroids are found, a single vector $\mathbf{x}$ can be assigned to the cluster with the minimum vector-to-cluster-centroid distance, among the K distances that are calculated. The Euclidean distance measure is used to calculate these distances:

$$D_i\left(\mathbf{x}, \boldsymbol{\mu}_i\right) = \left\|\mathbf{x} - \boldsymbol{\mu}_i\right\|, \quad i = 1, \ldots, K\ . \tag{3}$$

2.4 Reducing the Cardinality of the Training Set using an Entropy-based Measure

Let us consider a set of K clusters, or partitions, in the data space $\mathcal{T}$. The surrounding *Voronoi region* of the *i*-th cluster is denoted as V_i. Theoretically, the a-priori probability for each cluster to be the best matching one to any sample vector $\mathbf{x}$ of the feature space is calculated as such, if the probability density function $p(\mathbf{x})$ is known:

$$P_i = P\left(\mathbf{x} \in V_i\right) = \int_{V_i} p(\mathbf{x}) d\mathbf{x}\ . \tag{4}$$

For discrete data, the discrete probability histogram can replace the continuous probability density function as follows [24]:

$$P_i = P\left(\mathbf{x} \in V_i\right) = \frac{\#\left\{j \mid \mathbf{x}_j \in V_i\right\}}{N}, \tag{5}$$

where $\#\{\cdot\}$ represents the cardinality of a set and N the size of the training data set whose members are $\mathbf{x}_j, \quad j = 0,1,\ldots,N-1$.

Let us consider a set of K partitions in the training data space $\mathcal{T}$ and their distribution $P = (P_1, P_2, \ldots, P_K)$. The entropy, a commonly used measure that indicates the randomness of the distribution of a variable, can be defined as [24]:

$$H = H(P) = -\sum_{i=1}^{K} P_i \log_2 P_i \tag{6}$$

An 'ideal' data partitioning separates the data such that overlap between partitions is minimal, which is equivalent to minimizing the expected entropy of the partitions over all observed data.

In this work, the entropy-based measure is calculated in a new data space $\mathcal{T}' \subset \mathcal{T}$, which consists of a subset that retains K' of the total K clusters that are generated by making use of the *k*-means algorithm. Let us assume that the K' clusters contain Y' face classes. A necessary assumption that is used to calculate the entropy is that a true match to the test face class $\mathcal{X}$ exists within the $\mathcal{T}'$ space. Let the probability for the i-th face class $\mathcal{Y}_i'$, that is now contained in $\mathcal{T}'$, to represent a true match for $\mathcal{X}$ be $P_i = p(\mathcal{Y}_i' | \mathcal{X})$. Since the prior probabilities $p(\mathcal{Y}_i' | \mathcal{X})$ are unknown, they can be defined using the discrete probability histogram, as in (5), as:

$$P_i = p(\mathcal{Y}_i' | \mathcal{X}) = \frac{N_{\mathcal{Y}_i'}}{N_{\mathcal{Y}'}}, \tag{7}$$

where $N_{\mathcal{Y}'}$ is the total number of face images contained in $\mathcal{T}'$ and $N_{\mathcal{Y}_i'}$ is the number of times that class i is represented in $\mathcal{T}'$, e.g. $N_{\mathcal{Y}_i'}$ different images of the person that is associated with class i are contained in $\mathcal{T}'$. Practically, in order to reduce computations, entropy can be approximated by substituting (7) into (6), as will be shown in the following section. The approximated entropy values are used to guarantee that at each step of the DTMC algorithm an easier, in terms of the ability to achieve better separation among the classes, classification problem is defined. Simply, a threshold T_H is applied on the entropy value H to limit the number of different classes that $\mathcal{T}'$ will contain. Essentially, this is done by limiting the number of clusters K' that comprise $\mathcal{T}'$.

3. The DTMC Face Recognition Methodology

The DTMC algorithm is a multilevel process that, at each level, attempts to solve a redefined classification problem that is formulated by making use of dynamic training. Let us assume that an image $\mathbf{X}$ of a test face is to be assigned to one of the Y distinct classes Υ_i, $i = 1 \ldots Y$, that lie in the training set space $\mathcal{T}$. In addition, let us assume that each i-th class in $\mathcal{T}$ is represented by N_{Υ_i} images and the total number of training images is N_{Υ}. The face images that comprise the training set $\mathcal{T}$ can be denoted by $\mathbf{Y}_n$, $n = 1, \ldots, N_{\Upsilon}$.

3.1 DTMC Algorithm: Step 1

Initially, facial features are extracted from the test and training data by applying the MWD2 algorithm and collecting the wavelet coefficients that correspond to the lowest frequencies at decomposition level J_d, where J_d is calculated using (1). Essentially, the approximation wavelet coefficients $\mathbf{X} \prec A_0$ and $\mathbf{Y}_n \prec A_0$, $n = 1, \ldots, N_{\Upsilon}$, that are generated are then converted to 1-D vectors, by means of row concatenation, thus forming $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{y}}_n$, $n = 1, \ldots, N_{\Upsilon}$, respectively. The training feature vectors are grouped in a matrix $\tilde{\mathbf{Y}}$, such that each of its columns holds a single feature vector.

3.2 DTMC Algorithm: Step 2

RLDA is applied on $\tilde{\mathbf{Y}}$ and the discriminant matrix $\mathbf{W}$ is found by utilizing the criterion in (2). All possible dimensions of the discriminant space are retained, thus, $\mathbf{W}$ consists of $Y - 1$ columns. The training and test feature vectors are then projected to the MDF-space by

$$\tilde{\mathbf{y}}'_n = \mathbf{W}^{\mathrm{T}} \tilde{\mathbf{y}}_n, \quad n = 1, \ldots, N_{\Upsilon} \tag{8}$$

and

$$\tilde{\mathbf{x}}' = \mathbf{W}^{\mathrm{T}} \tilde{\mathbf{x}}. \tag{9}$$

Each training feature vector $\tilde{\mathbf{y}}'_n$ is stored in a column of $\tilde{\mathbf{Y}}'$.

3.3 DTMC Algorithm: Step 3

The *k*-means algorithm is then employed in an effort to partition the training data $\tilde{\mathbf{y}}'_n$, $n = 1, \ldots N_{\Upsilon}$, into the Y distinct face classes. The square-Euclidean-distance-based objective function of [22] is employed and Y centroid vectors $\tilde{\boldsymbol{\mu}}'_i$, $i = 1, \ldots, Y$, are found. The distance between each training feature vector and the Y

centroids is found using (3) and the training feature vector is assigned to the cluster associated with the minimum distance:

$$if \qquad D_i^n\left(\tilde{\mathbf{y}}_n^{'}, \tilde{\boldsymbol{\mu}}_i\right) = \min\left\{D_i^n\right\} \qquad then \qquad \tilde{\mathbf{y}}_n^{'} \in C_i \ . \tag{10}$$

Ideally, a single face class should reside in each cluster, and this cluster should contain all images of that particular face. However, this is guaranteed only if the separation among the Y classes is sufficiently large. The Y distances, between the test feature vector $\tilde{\mathbf{x}}^{'}$ and the cluster centroids, are found by using (3) and are sorted in ascending order in the vector:

$$D_{\tilde{\mathbf{x}}^{'}} = \text{sort}\left(D_i\left(\tilde{\mathbf{x}}^{'}, \tilde{\boldsymbol{\mu}}_i\right)\right). \tag{11}$$

3.4 DTMC Algorithm: Step 4

At this point we would like to redefine the original classification problem to a simpler one, by discarding part of the training data and applying discriminant analysis on the new subset. The scatter plot shown in Fig. 1 illustrates how a classification problem can become easier. Let us assume that a test sample to be classified is closer to class 0,1 and 2, and furthest from class 3, in terms of its distance from the class centroids. The $\text{DL}_{0,1,2,3}$ solid line that is shown represents the discriminant line generated by RLDA in order to separate the data of all 4 classes by projecting (using orthogonal projections as the dotted lines in Fig. 1 indicate) the data onto this line. Alternatively, $\text{DL}_{0,1,2}$ is the discriminant line that was generated by RLDA in order to separate the data of class 0,1 and 2 only. Assuming that the match for the test sample can be found in class 0,1 or 2, it is then clear that $\text{DL}_{0,1,2}$ provides a better separation for these three classes than $\text{DL}_{0,1,2,3}$ and provides greater expectation that the test sample will be classified correctly.

In order to make use of the concept of breaking down the classification problem into a pipeline of easier classification problems, one must first guarantee a high probability value for $p\left(\tilde{\mathbf{x}}^{'}\right)$, which we use to represent the expectation that the true match to the test data will reside in the portion of the training data space that is retained. If this match does not exist, then $p\left(\tilde{\mathbf{x}}^{'}\right) = 0$. Let us assume that $K^{'} < K$ clusters are to be retained. The probability that a match for the class of $\tilde{\mathbf{x}}^{'}$ can be found in the $i\text{-th}$ cluster that is retained, $p\left(\tilde{\mathbf{x}}^{'} \mid i\right)$, is inversely proportional to the distance between $\tilde{\mathbf{x}}^{'}$ and the centroid of this cluster. For example, if $\tilde{\mathbf{x}}^{'}$ coincides with the centroid of the $i\text{-th}$ cluster this distance is zero and $\tilde{\mathbf{x}}^{'}$ is more likely to belong to this cluster rather than to any other. As a result, and as (4) indicates, the largest possible value for $p\left(\tilde{\mathbf{x}}'\right)$ is attained, if the K' clusters that are retained are associated with the smallest values of $D_{\tilde{\mathbf{x}}'}$, and, thus, with the K' largest values for $p\left(\tilde{\mathbf{x}}' \mid i\right)$. This set of clusters comprises the new training space $\mathcal{T}'$:

$$if \quad D_i\left(\tilde{\mathbf{x}}', \tilde{\boldsymbol{\mu}}_i\right) \le D_{\tilde{\mathbf{x}}'}\left(K'\right) \quad then \quad C_i \in \mathcal{T}' . \tag{12}$$

The training feature vector data in these K' clusters are collected by making use of (10). Let us assume that the Y' classes Υ_i' are contained in the subset that is selected and that each i-th class in $\mathcal{T}'$ is represented by $N_{\Upsilon_i'}$ images. It is noted that Υ_i', instead of Υ_i, is used since now a face class may be represented by a smaller number of images, than the initial number that corresponded to all K clusters. The reason for this is because in certain cases the face images of a person may be partitioned into more than one cluster and the subset of K' clusters may not contain all the clusters that contain images of this particular face class. Now, the total number of training feature vectors is $N_{\Upsilon'}$ and these vectors are stored as columns in $\tilde{\mathbf{Y}}_s' \in \mathcal{T}'$. The value of K' is limited by the threshold T_H applied on the entropy value, which, in order to guarantee a low computational cost is approximated by substituting (7) into (6), so that the following is satisfied:

$$-\sum_{i=1}^{K'} \frac{N_{\Upsilon_i'}}{N_{\Upsilon'}} \log_2\left(\frac{N_{\Upsilon_i'}}{N_{\Upsilon'}}\right) \le T_H . \tag{13}$$

3.5 Dealing with Large Inter-Class Variations

A new MDF-space needs to be defined in order to attempt to find a match for the test feature vector $\tilde{\mathbf{x}}'$ with one of the Y' face classes that reside in the new training space $\mathcal{T}' \subset \mathcal{T}$. If each cluster would only contain all $N_{\Upsilon_i'}$ training feature vectors of a single face class Υ_i', which is the ideal case for the clustering process, the dimensions of this new MDF-space should be set to $Y' - 1$. Let us consider, however, the case where the $N_{\Upsilon_i'}$ training feature vectors of class Υ_i' are distributed into more than one cluster. Essentially, this means that a subset of the $N_{\Upsilon_i'}$ vectors was found to be more similar to vectors of different face classes, rather than to the remaining vectors of its own class. In this case, the new MDF-space should have additional (discriminant) dimensions so as to also be able to discriminate this subset of vectors from the vectors that correspond to different identities.

In other words, if feature vectors that belong to the i-th class are distributed into $K_i^{'}$ clusters, the discriminant process will attempt to discriminate among the data of this class using $K_i^{'}-1$ dimensions, in addition to discriminating among the $Y^{'}$ different face classes using $Y^{'}-1$ dimensions. Thus, the MDF-space is defined so as to best discriminate $K_d^{'}$ classes from one another, where $K_d^{'}$ is defined as:

$$K_d^{'} = \sum_{i=1}^{Y^{'}} \left(K_i^{'}-1\right) + Y^{'} . \tag{14}$$

This is done to enhance the classification ability of DTMC, since it enables the algorithm to formulate a clustering process that considers possible large variations in the set of images that each face class is represented by. If these variations are larger than identity-related variations, then these images are clustered into disjoint clusters. An example to this would be when a subset of images that correspond to the i-th training person present this person having a beard, or wearing glasses, whereas the rest of this person's images present it without having a beard and without wearing glasses. As a result, the feature vectors that correspond to the images showing this subject while having a beard, or wearing glasses, could be clustered with feature vectors of a different subject that has a beard, or wears glasses. By using (14), when DTMC attempts to find the match of a test face that corresponds to the identity of this i-th training person, it takes into consideration the fact that the test face may have a beard, or not, or wear glasses, or not. As a result, the match with the subset of the training images of class i, whose appearance is most similar to the test face, is considered, thus the best match can be found.

3.6. Iterative Processing

From this point onwards, steps 2 through 4 of DTMC are repeated in as many iterations as are necessary, until a single cluster is selected that contains a single class. For clarity, it is stated that $K_d^{'}-1$ indicates the length of the discriminant vector that is obtained by the RLDA process that will follow, whereas $K_d^{'}$ is the number of clusters that the training data will be clustered into by applying the *k*-means algorithm. For each iteration, the value of the entropy-related threshold T_H that is used to select a subset of the training data, is determined heuristically as is explained in the following section. A flow-chart of the DTMC algorithm is shown in Fig. 2

4. EXPERIMENTAL EVALUATION OF DTMC

In this section, the efficiency of the proposed methodology is evaluated on standard facial image data sets. The classification ability of DTMC is investigated by using data from the ORL, XM2VTS and FERET databases, whereas the UMIST database was used to set the values of the threshold T_H and the regularization parameter R at each clustering level, i.e., at each iteration of the DTMC algorithm. Essentially, as in most face recognition applications, the classification experiments that are carried out fall under the SSS problem, since few training samples per subject are available. The performance of DTMC is presented for various degrees of severity of the SSS problem. This is done by providing recognition rates for experiments where each face class $\mathcal{Y}_i$ is represented by the smallest to the largest possible number of training samples, N_{T}. Since DTMC employs discriminant analysis, the smallest possible sample number is 2. The largest possible training sample number for each face class $\mathcal{Y}_i$ is determined by the number of available images in this class, $N_{\mathcal{Y}_i}$, and by considering that at least one of these samples needs to be excluded, in order to be able to test the recognition performance for that particular class. Thus, the range for the number of training samples N_{T} is $\lfloor 2,\ldots,N_{\mathcal{Y}_i}-1 \rfloor$. The remaining images that do not comprise the training set are used to test the performance of DTMC, thus, they constitute the test set. The training and test sets are created by a random selection on each set of the $N_{\mathcal{Y}_i}$ images of each face class. To give statistical significance to our experiments, this random selection is repeated N_{R} times and N_{R} recognition rates are accumulated and then averaged in order to calculate the average recognition rate R_{rec}:

$$R_{\mathrm{rec}} = \frac{1}{N_{\mathrm{R}}} \sum_{i=1}^{N_{\mathrm{R}}} \frac{n^{i}_{\mathrm{correct}}}{n_{\mathrm{total}}}, \tag{15}$$

where n_{correct} is the number of correct matches of test faces to their corresponding face class in the training set and n_{total} is the number of all matching tests that are carried out.

4.1 Estimation of threshold T_H and regularization parameter R using the UMIST database

The UMIST database consists of $K = 20$ different face classes, each of which is represented by at least $N_{\mathcal{Y}_i} = 19$ images, $i = 1,\ldots,20$. The faces are shown at various angles, from left profile to right profile. Consequently, 17 recognition rates were derived for training sets that contained $N_{\mathrm{T}} = 2,\ldots,18$ images from each of the 20 face classes. Each corresponding rate was the average out of $N_{\mathrm{R}} = 20$ repetitions. An approximation to the optimal values of T_H and R at each clustering level was found by means of exhaustive processing in which the overall recognition rate was to be maximized. That is, the goal was to find the maximum possible average recognition rate of the experiments with the 17 different quantitative representations of the training set. For reference, the recognition rates R_{rec} that were achieved having this criterion been satisfied are shown in Table 1. However, they are not meant to be appropriate for comparison with the results of other methods, since the test set of the UMIST database was used to determine the values of the DTMC parameters. For the first RLDA step, the best value was found to be $R = 0$, which makes RLDA equivalent to the DLDA method of [8], whereas for the remaining RLDA steps that followed the best value was found to be $R = 0.05$. The best value for thresholding the entropy at the first and second clustering levels was found to be $T_H = 4$, and $T_H = 1.45$, respectively. At subsequent clustering levels, this value was found to be $T_H = 1.0$. Thus, a single cluster is selected; the face classes residing in that cluster are partitioned into a new set of clusters, one for each class, and from that partition a single cluster is again selected until only one face class remains in the selected cluster. The average number of clusters that were retained at the first and second clustering levels is $K^{'} = 15.35$ and $K^{'} = 2.14$, respectively.

4.2 Evaluation of performance with respect to available number of training samples per subject (N_{T}), using the ORL and XM2VTS databases

Now that all parameters for the DTMC methodology have been defined, the algorithm is evaluated on the ORL and XM2VTS databases. The ORL database consists of $K = 40$ different face classes, each of which is represented by $N_{\Upsilon_i} = 10$ images. The XM2VTS database consists of $K = 200$ different face classes, each of them represented by $N_{\Upsilon_i} = 8$ images. Fig. 3 and 4 show the boxplots [25] that provide statistical information about the recognition rates that are achieved throughout the $N_{\text{R}} = 20$ independent runs, on the ORL and XM2VTS databases, respectively. The boxes have lines at the lower quartile, median, and upper quartile values. The whiskers are lines extending from each end of the boxes to show the range of the rest of the data, specified at 1.5 times the inter-quartile range. Outliers are data with values beyond the ends of the whiskers and are indicated using ‘+’. The mean recognition rates R_{rec} that correspond to Fig. 3 and 4 are reported in Table 1. For the ORL database experiments, the average number of clusters that were retained at the first and second clustering levels is $K^{'} = 13.94$ and $K^{'} = 1.84$, respectively. The corresponding results for the XM2VTS database experiments are

$K^{'} = 9.12$ and $K^{'} = 1.51$. It must be noted that for the face recognition experiments that were carried out, usually 3 to 5 clustering levels, or iterations, are required for finding the identity of a test face.

The face recognition performance of the DTMC algorithm is now compared to the performance of a number of face recognition algorithms that have been recently presented in the literature. In order to derive meaningful conclusions when comparing the performance of various algorithms, the testing and evaluation methodologies, as well as the facial image databases that are used, should be identical. Moreover, we compare our algorithm against methods whose data processing procedures are presented in an elaborate fashion, in the literature. In [26], an identical experimental process, to the one that was used to evaluate the performance of DTMC, was carried out using random selection of the training and test set from the ORL database. Experimental rates are provided for a nearest neighbour-based (NN-b) [26], a PCA-based (PCA-b) [27], an LDA-based (LDA-b) [5] and the Markov random field-based (MRF) method that is proposed, for 1 to 9 training images per person. The comparison of DTMC with just 1 training image per person is not possible. The NN-b and PCA-b methods outperform DTMC when 2 training images are used, whereas LDA-b and MRF show a similar performance. When the number of training images is in the range of 3 to 9, DTMC shows the best face recognition performance, by outperforming the top-performer of the four methods in [26] by 3.87%, 2.91%, 2.39%, 2.38%, 1.7%, 1.4%, and 0.85% respectively. The relevant face recognition rates are reported in Table 2.

A common experiment that is used in order to evaluate the performance of a face recognition algorithm using the ORL database is the random selection of five images from each subject for training, whereas the remaining five are used for testing; this experimental process has been used in [28, 29, 30, 34]. The relevant face recognition rates for this particular experimental setup are reported in Table 3. In [28] face recognition rates for the combination of Gabor and PCA method (GPCA) [29], the Gabor-Fisher classifier (GFC) [29], the combination of Gabor and the DLDA method (GDLDA) of [8], and the Gabor Generalized Foley-Sammon Transform method (GGFST) that is proposed are provided. The DTMC algorithm outperforms these algorithms by 6.73%, 1.53%, 1.53%, and 0.53% respectively. For the same experimental setup, the authors in [30] provide performance results for the Convolutional Neural Network method (CNN) of [31], the Nearest Feature Line method (NFL) of [32], the Multiresolution PCA method (M-PCA) of [33] and the RBF Neural Network method (RBFNN) that they propose. The DTMC outperforms CNN by 0.86% and NFL by 0.16% whereas M-PCA and RBFNN outperform DTMC by 0.57% and 1.05% respectively. In [34], the same testing procedure is followed to test the nearest neighbour classifier (NN) [35], the nearest feature plane method (NFP) [36] and the two classifiers that are proposed, the nearest neighbour line (NNL) and the nearest neighbour plane (NNP). DTMC outperforms these algorithms by 2.38%, 1.23%, 1.85%, and 1.28%, respectively.

In addition, the leave-one-out strategy is employed in [34] to evaluate the performance of the algorithms that are proposed. The leave-one-out strategy is also employed in [37] to evaluate the performance of the Fisherfaces (FF), Independent Component Analysis (ICA), Eigenfaces (EF) and Kernel Eigenfaces (KEF) algorithms in [38], as well as of the 2-Dimensional PCA method (2DPCA) that is proposed, using ORL data. The performance of DTMC using this strategy is found in 20 independent runs. DTMC outperforms all these methods with a recognition rate of 98.62%. The relevant face recognition rates calculated under the leave-one-out strategy are reported in Table 4.

Furthermore, a second type of experiment was performed in [37] where the first five images of each subject in the ORL database comprise the training set, whereas the remaining five constitute the test set. The same experiment has been applied for DTMC and the recognition rate was found to be 98.3 %. As a result, DTMC again shows the best performance, as the face recognition rates that are reported in Table 5 show.

In [39], face recognition rates are presented for both the ORL and the XM2VTS databases. Specifically, 4 images per person make up the training set and the remaining 6 form the test set, when ORL data is used. Results are presented for the kernel direct discriminant analysis (KDDA) method in [40] as well as the new KDDA (nKDDA) method that is proposed. As Table 6 illustrates, in which results corresponding to the identical experimental setup can be found, the DTMC outperforms these methods with a recognition rate of 94.73%. For the experiments done on the XM2VTS database, 4 images per person comprise the training set and the remaining 4 form the test set. Again, recognition rates reported in Table 6, which correspond to the identical experimental setup, illustrate that the DTMC method outperforms the best rate reported for KDDA by 8.94% and, in addition, outperforms all methods with a recognition rate of 96.54%.

The most common face recognition experimental setup that is reported in the literature when XM2VTS data are used requires 3 images per person to form the training set and a single image per person to form the test set. Then, image permutations are done so that each of the 4 images becomes the test image, thus, cross-validation is used for testing as is it shown in [41, 42, 43]. The FR rates that are calculated by cross-validation are reported in Table 7. The performance of the DTMC method for 100 independent runs of this experimental process reaches 97.55%. In [41], the performance of seven algorithms is reported, among which the best is the method that is proposed and combines a Bayesian probabilistic model with Gabor filter responses (GBPM), with a recognition rate of 97.1%. In [42] the best rate that is reported for the proposed wavelet sub-band representation and kernel associative memory algorithm (WKAM), using the same experimental setup, is 83.39%. In [43], recognition rates of 99%, that outperform the corresponding rates of DTMC, are achieved by the adaptive clustering Bayesian SVM (ACBSVM) and the adaptive clustering unified subspace SVM (ACSSVM) algorithms that are presented.

4.3 Evaluation of performance with respect to available number of training samples per subject (N_{T}) and number of subjects (Y), using the FERET database

The performance of the DTMC algorithm has also been evaluated using the FERET database which avails larger number of face classes, Y. The 'closed universe' model, where the identity that corresponds to each test image is included in the training set, is used, as with our previous experiments. The closed-universe model is recommended in [44] for evaluating a face recognition algorithm on the FERET database, since it allows one to ask how good an algorithm is at identifying the test image. It is noted that we could not implement the FERET protocol for which results are reported in [44], where only one image per person is used in the training set and one in the test set. This is because the discriminant analysis step that is employed in DTMC requires multiple (at least 2) training images per subject. Instead, we implement an alternative testing procedure that is suggested in [44] where the number of different face classes, Y, in the training set is varied in order to evaluate the performance of the face recognition algorithm with respect to the size of the training set. In addition, the experimental process once again observes the recognition performance as the number of training samples, N_{T}, varies.

The 1199 different face classes that are available in the FERET database are represented by different number of images. The images that correspond to the face classes that are represented by 3 or more images, since at least 2 training images and 1 test image are required, are permuted so that each image becomes the test image. In the training sets that are formed each face class is represented by N_{T} images. For the experiments corresponding to $N_{\mathrm{T}} = 2, 3, 4, 5, 6, 7, 8, 9, 10$ we use the largest possible number of available face classes. These numbers are $Y = 480, 255, 130, 119, 103, 90, 66, 48, 27$, respectively. The performance measure in [44] is the probability of identification (or percentage of correct matches) which corresponds to the calculation of R_{rec} in (15), therefore, the same evaluation measure is used. For the experimental results for the FERET data, the average number of clusters that were retained at the first and second clustering levels is $K^{'} = 10.53$ and $K^{'} = 1.67$, respectively.

Once again, for almost all the experiments 3 to 5 clustering levels were sufficient for the identity of the test face to be found.

The performance results for the DTMC algorithm using the FERET data are presented in Table 8. This table illustrates that as N_{T} becomes larger, the performance of DTMC becomes less sensitive to variations in the number of face classes Y. For instance, for the recognition results that correspond to $Y = 27$ and $Y = 90$ for $N_{\mathrm{T}} = 2,...,7$, it is clear that smaller deviations between these two sets of results are found for larger values of N_{T}. In addition, when only 2 samples per face class are available in the training set the recognition ability of the DTMC algorithm becomes poor when Y is large. This fact has also been demonstrated in the evaluation results that processed the ORL and XM2VTS data. This malady is justified by the fact that the SSS problem is very severe since the lack of sufficient training samples causes improper estimation of a linear separation hyperplane between the classes, thus discriminant analysis cannot me modeled properly [45].

In order to make salient comparisons with other relevant methods, we chose to implement, to the best of our understanding, the related state-of-the-art Hierarchical Discriminant Analysis (HDA) algorithm in [4]. The number of nodes that are expanded at each level is 10, like the authors in [4] propose. The same pre-processing was done on the images and features were generated using the MWD2 algorithm for both HDA and DTMC. Since the FERET test provides not only results corresponding to different number of available images per subject, but also to different number of face classes, we chose to evaluate this algorithm using FERET data. The recognition results for HDA are also shown in Table 8. Once again, we see that when $N_{\mathrm{T}} = 2$ recognition results are poor. This was expected since LDA is used and once again suffers from the SSS problem. In fact, results suggest that RLDA does a better job than using the traditional combination of MEF and MDF spaces under the SSS problem, therefore the results agree with the conclusions drawn in [9]. To verify this we run DTMC by replacing the RLDA step by first generating MEF and then MDF spaces, as in [4], and lower recognition rates were observed; at an average, the recognition rate dropped by 8.11%. From the results in Table 8 it is clear that the performance of HDA is more sensitive to the variations of the number of face classes Y, than DTMC is. This is because the training of HDA is carried out without any biasing to the features of the particular test face. On the other hand, DTMC selects a subset of the training faces that are closer to the test face. As a result, DTMC handles large number of face classes much more efficiently than HDA does.

When the number of training samples N_{T} gets larger, e.g. equal or larger than 5, both DTMC and HDA provide good results for small values of Y. Therefore, it is expected that deriving the MEF and then the MDF spaces accounts for a similar performance to using the RLDA step. In order to verify this, the DTMC algorithm was run by replacing the RLDA step with the traditional MEF and MDF discriminant processing and indeed the performance of the algorithm deteriorated only mildly. More specifically, for $N_{\mathrm{T}} = 2,...10$, the average drop in the recognition performance, $(R_{\mathrm{rec}}\%)$, respectively, is 8.11, 4.40, 2.55, 1.86, 1.34, 0.97, 0.23, 0.00 and 0.00.

From the experimental results above it is concluded that the fact that DTMC iteratively selects subsets of the facial classes that are closer to the test face is responsible for the algorithm being able to maintain high recognition performance when the number of face classes Y increases. On the other hand, the RLDA discriminant process that DTMC employs, is responsible for providing much larger recognition rates when the SSS problem is severe (e.g. when $N_{\mathrm{T}} = 2, 3$), rather than using the traditional discriminant approaches.

The experimental comparisons that are presented, illustrate that the DTMC outperforms most recently proposed face recognition methods and competes well with the rest of them in various databases and under various performance protocols. In addition, the process is quite fast due to the dimensionality reduction that MWD2 offers, and due to the reduction of the number of training images and, thus, to the number of comparison tests that are carried out at each clustering level.

5. CONCLUSION

A novel face recognition methodology is proposed and its performance is evaluated. The DTMC algorithm uses dynamic training in a multistage clustering scheme in order to classify a test face by solving a set of simpler classification problems. This process iterates until one final cluster is selected that consists of a single face class, whose identity is set to be the best match to the identity of the test face. Certain parameters of DTMC are defined using the UMIST face database. This method was tested on the ORL, XM2VTS and FERET face databases and the experimental results show that the proposed framework outperforms most other face recognition methods.

6. ACKNOWLEDGEMENTS

This work is funded by the by the network of excellence BioSecure IST-2002-507634 (Biometrics for Secure

Authentication, http://www.biosecure.info), under Information Society Technologies (IST) priority of the 6th Framework Programme of the European Community.

REFERENCES

[1] J. Lu and K.N. Plataniotis, "Boosting face recognition on a large-scale database", *in Proc. IEEE Int. Conf. on Image Processing*, Rochester, New York, USA, September 22-25, 2002.

[2] G.D. Guo, H.J. Zhang, and S.Z. Li, "Pairwise face recognition", *in Proc. 8th IEEE Int. Conf. on Computer Vision*, vol. 2, pp. 282-287, Vancouver, Canada, July 2001.

[3] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos, "Face recognition using LDA based algorithms", *IEEE Trans. on Neural Networks*, vol. 14, no. 1, pp. 195-200, January 2003.

[4] D.L. Swets and J. Weng, "Hierarchical discriminant analysis for image retrieval", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 21, no. 5, pp. 386-401, May 1999.

[5] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, July 1997.

[6] K. Etemad and R. Chellappa, "Discriminant analysis for recognition of human face images", *Journal of the Optical Society of America A: Optics Image Science and Vision*, vol. 14, no. 8, pp. 1724-33, Aug. 1997.

[7] L-F Chen, M. H-Y Liao, J-C Lin, M-T Ko, and G-J Yu, "A new LDA-based face recognition system which can solve the small sample size problem", *Pattern Recognition*, vol. 33, no. 10, pp. 1713-26, 2000.

[8] H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data with application to face recognition", *Pattern Recognition*, vol. 34, no.12, pp. 2067-2070, 2001.

[9] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos, "Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition", *Pattern Recognition Letters*, vol. 26, no. 2, pp. 181-191, 2005.

[10] H.-M. Tang, M.R. Lyu, and I. King, "Face recognition committee machines: dynamic vs. static structures", *in Proc. 12th Int. Conf. on Image Analysis and Processing*, pp. 121-126, Mantova, Italy, Sept. 17-19, 2003.

[11] H.-C. Liu, C.-H. Su, Y.-H. Chiang, and Y.-P. Hung, "Personalized face verification system using owner-specific cluster-dependent LDA-subspace", *in Proc. 17th Int. Conf. on Pattern Recognition*, vol. 4, pp. 344-347, Aug. 23-26, 2004.

[12] S. G. Mallat, "A theory for multi-resolution signal decomposition, the wavelet representation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674-693, July, 1989.

[13] I. Daubechies, "Ten lectures on wavelets", CBMS-*NSF conference series in applied mathematics*, SIAM Ed., 1992.

[14] B. Zhang, H. Zhang, and S. Ge, "Face recognition by applying wavelet subband representation and Kernel Associative Memories", *IEEE Trans. on Neural Networks*, vol. 15, no.1, pp. 166-177, Jan. 2004.

[15] M. Bicego, U. Castellani, and V. Murino, "Using Hidden Markov Models and wavelets for face recognition", *Proc. 12th Int. Conf. on Image Analysis and Processing*, pp. 52-56, Mantova, Italy, Sept. 17-19, 2003.

[16] D. B. Graham and N. M. Allinson, "Characterizing virtual eigensignatures for general purpose face recognition", In H. Wechsler, P. J. Phillips, V. Bruce, F. Fogelman-Soulie, and T. S. Huang, Ed., *Face Recognition: From Theory to Applications, NATO ASI Series F, Computer and Systems Sciences*, vol. 163 , pp. 446-456, 1998.

[17] AT&T Laboratories Cambridge, The Database of Faces, http://www.uk.research.att.com/facedatabase.html.

[18] J. Luettin, and G. Maitre, "Evaluation protocol for the extended M2VTS database (XM2VTSDB)", *in IDIAP Communication 98-05*, IDIAP, Martigny, Switzerland, 1998.

[19] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley, MA: Wellesley-Cambridge Press, 1996.

[20] C. Nastar and N. Ayach, “Frequency-based nonrigid motion analysis”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 11, pp. 1067-1079, Nov. 1996.

[21] J. H. Lai, P. C. Yuen, and G. C. Feng, “Face recognition using holistic Fourier invariant features”, *Pattern Recognition*, vol. 34, no. 1, pp. 95-109, 2001.

[22] F. Camastra and A. Verri, “A novel kernel method for clustering” , *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 27, no. 5, pp. 801-805, May 2005.

[23] G.A.F. Seber, *Multivariate Observations*, Wiley, New York, 1984.

[24] M. Koskela, J. Laaksonen, and E. Oja, “Entropy-based measures for clustering and SOM topology preservation applied to content-based image indexing and retrieval”, *in Proc. 17th Int. Conf. on Pattern Recognition, vol.* 2, pp. 1005-1009, 2004.

[25] R. McGill, J. W. Tukey, and W. A. Larsen, “Variations of boxplots”, *The American Statistician*, vol. 32, pp.12-16, 1978.

[26] R. Huang, V. Pavlovic, and D.N. Metaxas, “A hybrid face recognition method using Markov random fields”, *in Proc. 17th Int. Conf. Pattern Recognition*, vol. 3, pp. 157-160**,** Aug. 23-26, 2004.

[27] M. Turk and A. Pentland, “Eigenfaces for recognition”, *Journal of Cognitive Neuroscience,* vol. 3, no. 1, pp. 71-86, 1991.

[28] G. Dai and Y. Qian, “Face recognition with the robust feature extracted by the generalized Foley-Sammon transform”, *in Proc. Int. Symposium on Circuits and Systems,* vol. 2 , pp. 109-12, May 23-26, 2004.

[29] C. J. Liu and H. Wechsler, “Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition”, *IEEE Trans. on Image Processing*, vol. 11, no. 4, pp. 467-476, 2002.

[30] M. J. Er, S. Wu, J. Lu, and H. L. Toh, “Face recognition with radial basis function (RBF) neural networks”, *IEEE Trans. on Neural Networks*, vol. 13 , no. 3 , pp. 697-710, May 2002.

[31] S. Lawrence, 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.

[32] S. Z. Liand and J.Lu, “Face recognition using the nearest feature line method”, *IEEE Trans. Neural Networks*, vol. 10, no. 2, pp. 439-443, Mar. 1999.

[33] V. Brennan and J. Principe, “Face classification using a multiresolution principal component analysis*”, in Proc. IEEE Signal Processing Society Workshop on Neural Networks*, pp. 506-515, 1998.

[34] W. Zheng, C. Zou, and L. Zhao, “Face recognition using two novel nearest neighbor classifiers”, *in Proc. of IEEE Int. Conf. on Acoustics Speech and Signal Processing*, vol. 5 , pp. 725-728, May 17-21, 2004.

[35] T. M. Cover and P.E. Hart, “Nearest neighbour pattern classification”, *IEEE Trans. on Information Theory*, vol. 13, pp. 21-27, Jan, 1967.

[36] J. T. Chien and C. C. Wu, “Discriminant waveletfaces and nearest feature classifiers for face recognition”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 12, pp. 1644-49, 2002.

[37] J. Yang, D. Zhang, A.F. Frangi, and J.-Y. Yang, “Two-dimensional PCA: a new approach to appearance-based face representation and recognition”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 26, no. 1, pp. 131-137, Jan 2004.

[38] M. H. Yang, “Kernel eigenfaces vs. kernel fisherfaces: face recognition using kernel methods”, *in Proc. IEEE 5th Int. Conf. on Automatic Face and Gesture Recognition*, pp. 215-220, May 2002.

[39] W.U. Xiao-Jun, J. Kittler, Y. Jing-Yu, K. Messer, and W. Shi-Tong, “A new kernel direct discriminant analysis (KDDA) algorithm for face recognition”, *in Conf. British Machine Vision*, Kingston University, London, Sept. 7-9th, 2004.

[40] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos, “Face recognition using kernel direct discriminant analysis algorithms” , *IEEE Trans. on Neural Networks*, vol.14, no.1, pp. 117-126, 2003.

[41] X. Wang and X. Tang, “Bayesian face recognition using Gabor features”, *in Proc. of ACM SIGMM 2003 Multimedia Biometrics Methods and Applications Workshop*, Berkeley, CA, USA, Nov. 2003.

[42] B.-L. Zhang, H. Zhang, and S. Sam Ge, “Face recognition by applying wavelet subband representation and kernel associative memory”, *IEEE Trans. on Neural Networks*, vol. 15, no. 1, pp. 166-177, Jan. 2004.

[43] Z. Li, X. Tang, “Bayesian face recognition using support vector machine and face clustering”, *in Proc. Computer Vision and Pattern Recognition*, vol. 2 , pp. 374-380, 27 June-2 July, 2004.

[44] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, “The FERET evaluation methodology for face-recognition algorithms”, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 22, no. 10, pp. 1090-1104, Oct. 2000.

[45] M. Kyperountas, A. Tefas, and I. Pitas, “Weighted Piecewise LDA for Solving the Small Sample Size Problem in Face Verification”, *IEEE Trans. on Neural Networks*, vol. 18, no. 2, pp. 506-519, March 2007.

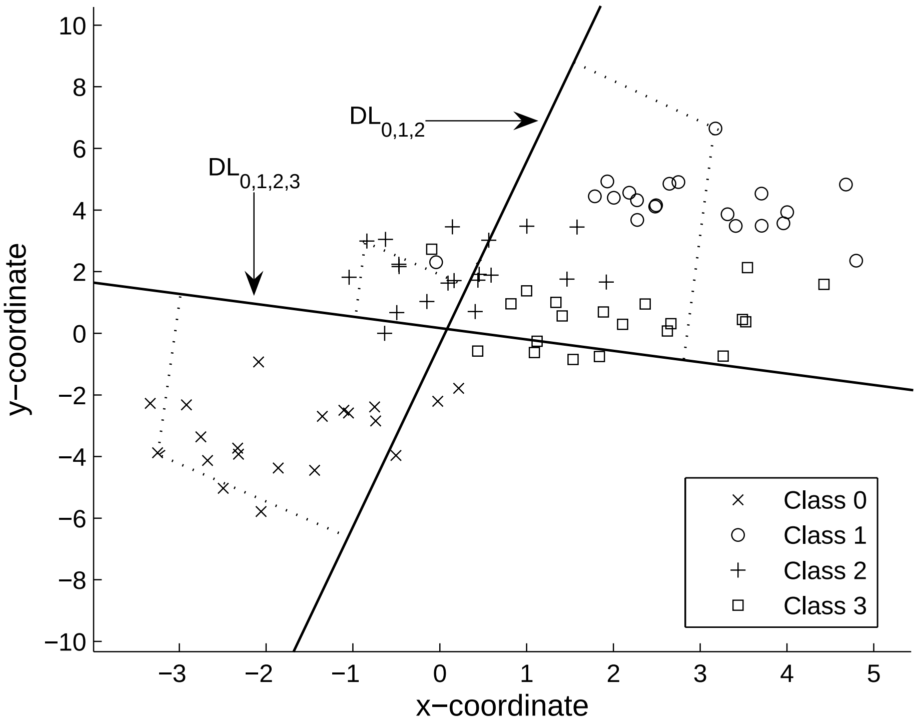

Fig. 1: Solving an easier classification problem by retaining a subset of the classes.

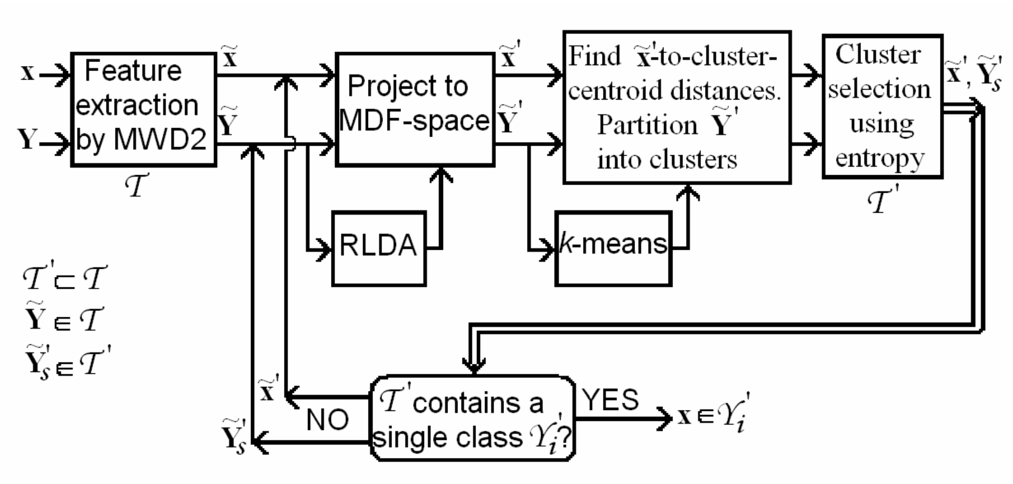
$\mathbf{x}$
$\mathbf{Y}$
Feature extraction by MWD2
$\mathcal{T}$
$\tilde{\mathbf{x}}$
$\tilde{\mathbf{Y}}$
Project to MDF-space
RLDA
$\tilde{\mathbf{x}}'$
$\tilde{\mathbf{Y}}'$
Find $\tilde{\mathbf{x}}'$-to-cluster-centroid distances. Partition $\tilde{\mathbf{Y}}'$ into clusters
k-means
Cluster selection using entropy
$\mathcal{T}'$
$\tilde{\mathbf{x}}', \tilde{\mathbf{Y}}_s'$
$\mathcal{T}'\subset\mathcal{T}$
$\tilde{\mathbf{Y}}\in\mathcal{T}$
$\tilde{\mathbf{Y}}_s'\in\mathcal{T}'$
$\mathcal{T}'$ contains a single class $\mathcal{Y}_i'$?
YES
$\mathbf{x}\in\mathcal{Y}_i'$
NO
$\tilde{\mathbf{x}}'$
$\tilde{\mathbf{Y}}_s'$

Fig. 2: Flow-chart of the DTMC algorithm.

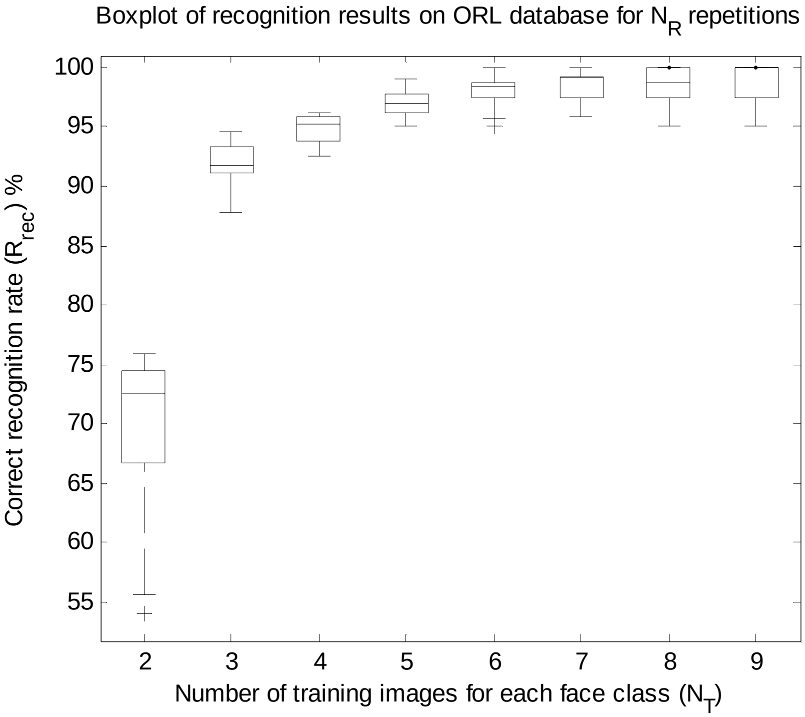

Fig. 3: Recognition rate versus N_{T} of experiments on ORL.

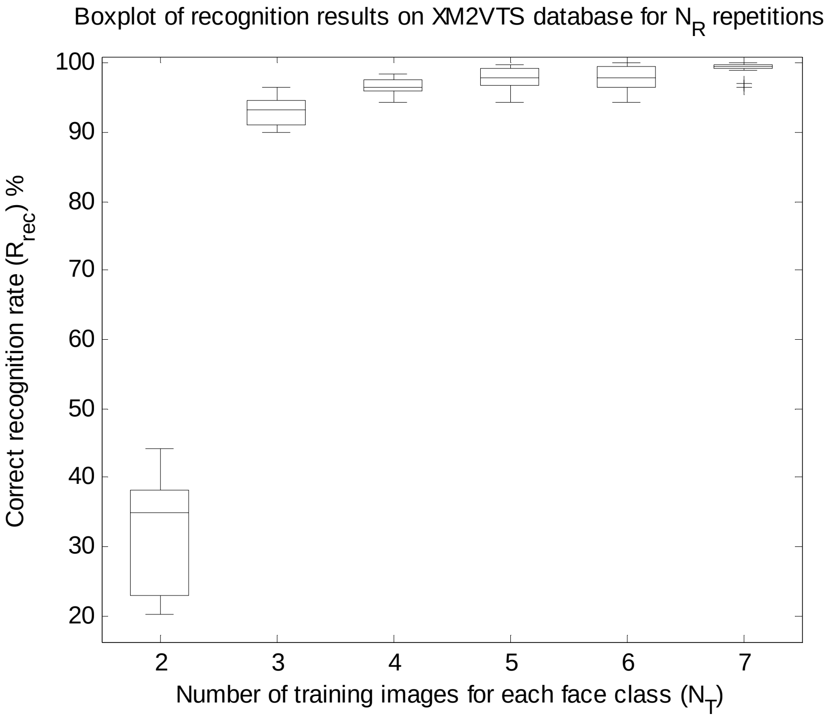

Fig. 4: Recognition rate versus N_{T} of experiments on XM2VTS.

Table 1: Mean recognition rates (R_{rec}) versus number of training samples per subject (N_{T}).

UMIST				ORL		XM2VTS	
N_{T}	R_{rec} (%)	N_{T}	R_{rec} (%)	N_{T}	R_{rec} (%)	N_{T}	R_{rec} (%)
2	59.26	11	97.03	2	69.44	2	31.89
3	82.67	12	97.04	3	91.96	3	93.03
4	90.20	13	97.38	4	94.73	4	96.54
5	92.23	14	97.95	5	97.03	5	97.78
6	92.46	15	98.13	6	98.06	6	97.98
7	94.94	16	98.42	7	98.50	7	**99.05**
8	95.86	17	98.63	8	98.50		
9	95.85	18	**100.00**	9	**98.75**		
10	96.03						

Table 2: Recognition rates of various methods versus the number of training samples per subject, using ORL data.

	R_{rec} (%)				
N_{T}	NN-b [26]	PCA-b [27]	LDA-b [5]	MRF [26]	DTMC
2	**81.08**	71.19	68.84	68.38	69.44
3	88.09	79.66	81.74	79.21	**91.96**
4	91.82	84.92	86.74	82.63	**94.73**
5	94.64	88.31	88.87	86.95	**97.03**
6	95.68	90.84	90.84	90.53	**98.06**
7	96.80	92.58	91.62	92.17	**98.50**
8	97.10	94.05	92.85	94.88	**98.50**
9	97.90	95.20	93.75	96.75	**98.75**

Table 3: Recognition rates of various methods for 5 training samples per subject, using ORL data.

Method	R_{rec} (%) for $N_{\text{T}} = 5$
GPCA [29]	90.30
GFC [29]	95.50
GDLDA [8]	95.50
GGFST [28]	96.50
CNN [31]	96.17
NFL [32]	96.87
M-PCA [33]	97.60
RBFNN [30]	**98.08**
NN [35]	94.65
NFP [36]	95.80
NNL [34]	95.18
NNP [34]	95.75
DTMC	97.03

Table 4: Recognition rates of various methods evaluated under the leave-one-out strategy, using ORL data.

Method	R_{rec} (%) using the leave-one-out strategy
NN [35]	98.25
NFP [36]	98.25
NNL [34]	98.50
NNP [34]	98.50
FF [38]	98.50
ICA [38]	93.80
EF [38]	97.50
KEF [38]	98.00
2DPCA [37]	98.30
DTMC	**98.62**

Table 5: Recognition rates of various methods with the training set being comprised of the first 5 images of a subject, using ORL data.

Method	R_{rec} (%) for $N_{\text{T}} = 5$ by selecting the first 5 images of each subject
FF [38]	94.50
ICA [38]	85.00
KEF [38]	94.00
2DPCA [37]	96.00
DTMC	**98.30**

Table 6: Recognition rates of various methods for 4 training samples per subject, using ORL and XM2VTS data.

	R_{rec} (%) for $N_{\text{T}} = 4$	
Method	ORL	XM2VTS
KDDA [40]	91.30	87.60
nKDDA [39]	91.30	92.50
DTMC	**94.73**	**96.54**

Table 7: Recognition rates of various methods calculated under the cross-validation strategy applied to 4 samples $(N_{\text{T}} = 3)$, using XM2VTS data.

Method	R_{rec} (%) for cross-validation using 4 samples, out of which 3 are used for training
GBPM [41]	97.10
WKAM [42]	83.39
ACBSVM [43]	**99.00**
ACSSVM [43]	**99.00**
DTMC	97.55

Table 8: Recognition rates for the DTMC and HDA [4] algorithms for various number of face classes, Y , and number of training samples, $\left(N_{\mathrm{T}}\right)$, using FERET data.

Mean recognition rates $\left(R_{\mathrm{rec}}\%\right)$ for various number of training samples per subject $\left(N_{\mathrm{T}}\right)$ and various number of face classes (Y) using FERET data.										
N_{T}	Method	$Y=27$	$Y=48$	$Y=66$	$Y=90$	$Y=103$	$Y=119$	$Y=130$	$Y=255$	$Y=480$
2	DTMC	88.65	82.34	77.34	73.85	71.65	69.14	67.72	62.57	57.34
	HDA	79.24	75.53	71.42	66.83	65.14	64.30	62.52	56.21	48.61
3	DTMC	96.46	95.30	94.79	94.06	93.82	93.64	93.34	92.54	-
	HDA	89.24	88.73	86.86	83.35	82.42	82.13	81.59	77.25	-
4	DTMC	98.48	98.18	97.53	97.40	97.24	96.95	96.17	-	-
	HDA	93.67	92.54	91.22	89.31	88.45	87.53	85.70	-	-
5	DTMC	99.24	99.17	98.83	98.65	98.48	98.61	-	-	-
	HDA	95.96	95.11	94.42	92.37	91.85	90.32	-	-	-
6	DTMC	99.49	99.34	98.96	98.85	98.67	-	-	-	-
	HDA	97.63	96.40	94.95	93.54	92.76	-	-	-	-
7	DTMC	100	100	99.35	99.38	-	-	-	-	-
	HDA	99.22	98.83	98.26	97.61	-	-	-	-	-
8	DTMC	100	100	100	-	-	-	-	-	-
	HDA	100	99.56	99.03	-	-	-	-	-	-
9	DTMC	100	100	-	-	-	-	-	-	-
	HDA	100	99.71	-	-	-	-	-	-	-
10	DTMC	100	-	-	-	-	-	-	-	-
	HDA	100	-	-	-	-	-	-	-	-