A Novel Discriminant Non-Negative Matrix Factorization Algorithm With Applications to Facial Image Characterization Problems

Irene Kotsia, Stefanos Zafeiriou, and Ioannis Pitas, *Fellow, IEEE*

***Abstract*—The methods introduced so far regarding discriminant non-negative matrix factorization (DNMF) do not guarantee convergence to a stationary limit point. In order to remedy this limitation, a novel DNMF method is presented that uses projected gradients. The proposed algorithm employs some extra modifications that make the method more suitable for classification tasks. The usefulness of the proposed technique to frontal face verification and facial expression recognition problems is demonstrated.**

***Index Terms*—Facial expression recognition, frontal face verification, linear discriminant analysis, non-negative matrix factorization (NMF), projected gradients.**

I. Introduction

OVER the past few years, the non-negative matrix factorization (NMF) algorithm and its alternatives have proven to be very useful for several problems, especially in facial image characterization and representation problems [1]–[9]. NMF, similar to the principal component analysis (PCA) algorithm [10], represents a facial image as a linear combination of basis images and does not allow negative elements in either the basis images or the representation coefficients used in the linear combination of the basis images. Thus, it represents a facial image only by the additions of weighted basis images. The non-negativity constraints correspond better to the intuitive notion of combining facial parts to create a complete facial image. The bases of PCA are the Eigenfaces, resembling distorted versions of the entire face, while the bases of NMF are localized features that correspond better to the intuitive notion of facial parts [1]. The original NMF algorithm does not incorporate any sparseness constraints in the decomposition, even though in many cases, it has been experimentally verified that it produces sparse bases (i.e., bases with components that are spatially distributed without any connectivity).

The belief that NMF produces local representations is mainly intuitive (i.e., addition of different non-negative bases using non-negative weights). Recently, some theoretical work has been done [11] in order to determine whether NMF provides a correct decomposition into parts and, at the same time, a set of requirements has been defined. This set of requirements is quite restrictive and cannot be satisfied by all kinds of image databases (e.g., facial image databases) [9]. Nevertheless, the sparsity of NMF in various facial image characterization problems has been verified by many researchers [1], [4], [9].

In order to enhance the sparsity of NMF, many methods have been proposed [3], [6], [8]. NMF has been further extended to supervised alternatives—the so-called DNMF or Fisher-NMF (FNMF) methods [5], [7], [9] by incorporating discriminant constraints in the decomposition (for simplicity reasons, we will refer to all of these methods [5], [7], [9] as DNMF variants). The intuitive motivation behind DNMF methods is to extract bases that correspond to discriminant facial regions for facial expression recognition [5], face recognition [7], and facial identity verification [9]. An important issue related to visual representation when DNMF is applied to facial identity verification or facial expression recognition problems is the fact that almost all features found by its basis images are represented by salient facial features, such as eyes, eyebrow, or mouth [5], [9], [12]. While discarding less important information (which is not the case for NMF, since it provides a not so localized representation) or emphasizing it less, DNMF approximately preserves the spatial topology of salient features (which are mostly absent in the case of other sparse NMF approaches such as local NMF (LNMF) [3]) by emphasizing them. The features retrieved by LNMF have rather random positions [3], [5], [9]. Although there is no external intervention, we believe that the preservation of these salient features in the learning process of DNMF is caused by the class information taken into account by the algorithm, since these features are of great importance for the classification framework (for facial identity verification and facial expression recognition) [5], [9], [12].

In order to calculate the update rules for the weights and basis images of DNMF, a similar procedure to the one followed in the NMF decomposition [2], [5], [7], [9] was used. More precisely, the cost optimization of the decomposition has been calculated using an auxiliary function [9]. Although this auxiliary function guarantees the nonincreasing behavior of the cost function, it does not ensure the convergence of the algorithm to a limit point that is also a stationary point of the optimization problem [13], [14] (i.e., the first derivative of the cost function at the limit point is equal to zero). Furthermore, in the DNMF methods [5], [7], [9], the discriminant analysis is employed on the representation coefficients and not on the actual features used in the classification procedure. The actual features used for classification in [5],

Manuscript received November 1, 2006; revised April 19, 2007. This work was supported by the "SIMILAR" European Network of Excellence on Multimodal Interfaces of the IST Programme of the European Union (www.similar.cc). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Josef Bigun.

The authors are with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Greece (e-mail: ekotsia@aiia.csd.auth.gr; dralbert@aiia.csd.auth.gr; pitas@aiia.csd.auth.gr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2007.902017

[7], and [9] are derived from the projection of the facial images to the bases matrix and only implicitly depend on the representation coefficients.

In this paper, a novel DNMF method that takes into consideration all of the previously mentioned issues is proposed. Discriminant analysis is employed on the classification features and not on the representation coefficients. The NMF-based optimization problems [2], [3], [5], [7], [9] are nonconvex. They may have several local minima and produce a sequence of iterations. A common misunderstanding is that the limit points of this sequence are local minima [14]. In optimization theory, most nonconvex optimization methods guarantee only the stationarity of the limit points. Such a property is very useful as a local minimum must be a stationary point. In order to ensure stationarity, projected gradients are used in order to solve the constrained optimization problem. Similar methods have been successfully applied to the original NMF [14]. The proposed technique has been applied to facial expression recognition and face verification where it is demonstrated that it outperforms other DNMF methods [5], [7], [9], while having well-established theoretical properties. The basis images that are produced by the proposed algorithm have, as well, the same property with those derived using the DNMF method [9] and are represented by salient facial features.

The rest of this paper is organized as follows. In Section II, the main concepts of the DNMF methods and the proposed approach are outlined. In Section III, the novel DNMF algorithm using projected gradients is presented. The experimental results are described in Section IV. Finally, conclusions are drawn in Section V.

II. Discriminant Non-Negative Matrix Factorization Algorithms

In this section, the NMF algorithm and the procedure is followed to formulate the DNMF approach [9] are briefly presented. For simplicity reasons, the formulation in [9] will be used, since the methods presented in [5] and [7] are very similar to the one proposed in [9]. The method proposed in [5] is a mix of DNMF and LNMF [3] algorithms and the only difference or the DNMF method presented in [9] with the one proposed in [7] is the definition of the between-class scatter matrix. From now on, y_i will denote the ith element of a vector $\mathbf{y}$, while $\mathbf{A}_{i,j}$ is the element stored in the i,j position of a matrix $\mathbf{A}$.

An image scanned row-wise is used to form a vector $\mathbf{x} = [x_1 \dots x_F]^T$ for the NMF algorithm. The basic idea behind NMF is to approximate the image $\mathbf{x}$ with a linear combination of the basis images in $\mathbf{Z} \in \Re_+^{F\times M}$, whose coefficients are the elements of $\mathbf{h} \in \Re_+^M$ such that $\mathbf{x} \approx \mathbf{Zh}$. Using the conventional least squares formulation, the approximation error $\mathbf{x} \approx \mathbf{Zh}$ is measured in terms of $L(\mathbf{x}||\mathbf{Zh}) \triangleq ||\mathbf{x} - \mathbf{Zh}||^2 = \sum_i (x_i - [\mathbf{Zh}]_i)^2$. Another way to measure the error of the approximation is using the Kullback–Leibler (KL) divergence $KL(\mathbf{x}||\mathbf{Zh}) \triangleq \sum_i (x_i \ln(x_i/[\mathbf{Zh}]_i) + [\mathbf{Zh}]_i - x_i)$ [2] which is the most common error measure for all DNMF methods [5], [7], [9]. A limitation of KL divergence is that it requires $\mathbf{x}_i$ and $[\mathbf{Zh}]_i$ to be strictly positive (i.e., neither negative nor zero values are allowed).

In order to apply the NMF algorithm, the matrix $\mathbf{X} \in \Re_+^{F\times T} = [x_{ij}]$ should be constructed, where x_{ij} is the ith element of the jth image vector. In other words, the jth column of $\mathbf{X}$ is the facial image $\mathbf{x}_j$. NMF aims at finding two matrices $\mathbf{Z} \in \Re_+^{F\times M} = [z_{i,k}]$ and $\mathbf{H} \in \Re_+^{M\times T} = [h_{k,j}]$ such that

$$\mathbf{X} \approx \mathbf{ZH}. \tag{1}$$

After the NMF decomposition, the facial image $\mathbf{x}_j$ can be written as $\mathbf{x}_j \approx \mathbf{Zh}_j$, where $\mathbf{h}_j$ is the jth column of $\mathbf{H}$. Thus, the columns of the matrix $\mathbf{Z}$ can be considered as basis images and the vector $\mathbf{h}_j$ as the corresponding weight vector. The vector $\mathbf{h}_i$ can be also considered as the projection of $\mathbf{x}_j$ in a lower dimensional space.

The defined cost for the decomposition (1) is the sum of all KL divergences for all images in the database

$$\begin{aligned} D(\mathbf{X}||\mathbf{ZH}) =& \sum_j KL(\mathbf{x}_j||\mathbf{Zh}_j) \\ =& \sum_{i,j}\left(x_{i,j}\ln\left(\frac{x_{i,j}}{\sum_k z_{i,k}h_{k,j}}\right) + \sum_k z_{i,k}h_{k,j} - x_{i,j}\right). \end{aligned} \tag{2}$$

The NMF factorization is the outcome of the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{H}} \quad D(\mathbf{X}||\mathbf{ZH}) \\ &\text{subject to} \quad z_{i,k} \geq 0, \quad h_{k,j} \geq 0, \quad \sum_i z_{i,j} = 1, \quad \forall j. \end{aligned} \tag{3}$$

In order to formulate the DNMF algorithm, let the matrix $\mathbf{X}$ that contains all of the facial images be organized as follows. The jth column of the database $\mathbf{X}$ is the ρth image of the rth image class. Thus, $j = \sum_{i=1}^{r-1} N_i + \rho$, where N_i is the cardinality of the image class i. The rth image class could consist of one person's facial images, for face recognition, and verification problems. For facial expression recognition, the rth class could consist of the images of one of the six basic facial expression classes (i.e., anger, disgust, fear, happiness, sadness, and surprise). The vector $\mathbf{h}_j$ that corresponds to the jth column of the matrix $\mathbf{H}$ is the coefficient vector for the ρth facial image of the rth class and will be denoted as $\boldsymbol{\eta}_\rho^{(r)} = \left[\eta_{\rho,1}^{(r)} \dots \eta_{\rho,M}^{(r)}\right]^T$. The mean vector of the vectors $\boldsymbol{\eta}_\rho^{(r)}$ for the class r is denoted as $\boldsymbol{\mu}^{(r)} = \left[\mu_1^{(r)} \dots \mu_M^{(r)}\right]^T$ and the mean of all classes as $\boldsymbol{\mu} = [\mu_1 \dots \mu_M]^T$. Then, the within-class scatter matrix for the coefficient vectors $\mathbf{h}_j$ is defined as

$$\mathbf{S}_w = \sum_{r=1}^{K}\sum_{\rho=1}^{N_r}\left(\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)}\right)\left(\boldsymbol{\eta}_\rho^{(r)} - \boldsymbol{\mu}^{(r)}\right)^T \tag{4}$$

whereas the between-class scatter matrix is defined as

$$\mathbf{S}_b = \sum_{r=1}^{K} N_r(\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})(\boldsymbol{\mu}^{(r)} - \boldsymbol{\mu})^T. \tag{5}$$

The matrix $\mathbf{S}_w$ defines the scatter of the sample vector coefficients around their class mean. The dispersion of samples that belong to the same class around their corresponding mean should be as small as possible. A convenient measure for the dispersion of the samples is the trace of $\mathbf{S}_w$. The matrix $\mathbf{S}_b$ denotes the between-class scatter matrix and defines the scatter of the mean vectors of all classes around the global mean $\boldsymbol{\mu}$. Each class must be as far as possible from the other classes. Therefore, the trace of $\mathbf{S}_b$ should be as large as possible.

To formulate the DNMF method [9], discriminant constraints have been incorporated in the NMF decomposition inspired by the minimization of Fisher's criterion [9]. The DNMF cost function is given by

$$D_d(\mathbf{X}||\mathbf{ZH}) = D(\mathbf{X}||\mathbf{ZH}) + \gamma \mathrm{tr}[\mathbf{S}_w] - \delta \mathrm{tr}[\mathbf{S}_b] \qquad (6)$$

where γ and δ are non-negative constants. The update rules that guarantee the nonincreasing behavior of (6) for the weights $h_{k,j}$ and the bases $z_{i,k}$, under the constraints of (2), can be found in [9]. Unfortunately, the update rules only guarantee nonincreasing behavior for (6) and do not ensure that the limit point will be stationary.

Two more issues arise regarding the DNMF algorithm. The first is that in DNMF methods [5], [7], [9], the discriminant constraints are not employed in the features used for classification but in the weights of the representation. Therefore, the vectors $\mathbf{h}_j$ are considered to be the projected vectors of the original facial vectors $\mathbf{x}_j$ in a lower dimensional feature space. Actually, the features used for classification using the DNMF bases matrix $\mathbf{Z}$ are derived from either the projection $\acute{\mathbf{x}} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{x}$ or the projection $\tilde{\mathbf{x}} = \mathbf{Z}^T\mathbf{x}$ [5], [9]. In all cases, the actual features used in the classification framework depend directly on $\mathbf{Z}^T\mathbf{X}$ and only implicitly on the coefficient matrix $\mathbf{H}$. Hence, it is reasonable to incorporate discriminant constraints for the feature vectors $\tilde{\mathbf{x}}$ and remove the discriminant constraints of the coefficient vectors $\mathbf{h}$.

Moreover, the cost in (2) is not well defined at any point of the bounded region, since the $\ln(u)$ function is not well defined for zero argument u. Thus, in order to measure the approximation of the decomposition, least squares will be used.

III. Projected Gradient Methods for Discriminant NMF

In the previous section, the use of a new cost function for discriminant non-negative matrix factorization has been motivated. Let $\mathbf{E} = \mathbf{X} - \mathbf{ZH}$ be the error signal of the decomposition. The modified optimization problem should minimize

$$D_p(\mathbf{X}||\mathbf{ZH}) = ||\mathbf{E}||_F^2 + \gamma \mathrm{tr}[\tilde{\mathbf{S}}_w] - \delta \mathrm{tr}[\tilde{\mathbf{S}}_b] \qquad (7)$$

under non-negativity constraints, where $||\cdot||_F$ is the Frobenius norm. The within-class scatter matrix $\tilde{\mathbf{S}}_w$ and the between-scatter scatter matrix $\tilde{\mathbf{S}}_b$ are defined using the vectors $\tilde{\mathbf{x}}_j = \mathbf{Z}^T\mathbf{x}_j$ and the definitions of the scatter matrices in (4) and (5).

The minimization of (7) subject to non-negative constraints yields the new discriminant non-negative decomposition. The new optimization problem is the minimization of (7) subject to non-negative constraints for the weights matrix $\mathbf{H}$ and the bases matrix $\mathbf{Z}$. This optimization problem will be solved using projected gradients in order to guarantee that the limit point will be stationary. In order to find the limit point, two functions are defined

$$f_{\mathbf{Z}}(\mathbf{H}) = D_p(\mathbf{X}||\mathbf{ZH}) \text{ and } f_{\mathbf{H}}(\mathbf{Z}) = D_p(\mathbf{X}||\mathbf{ZH}) \qquad (8)$$

by keeping $\mathbf{Z}$ and $\mathbf{H}$ fixed, respectively.

The projected gradient method used in this paper, successively optimizes two subproblems [14]

$$\min_{\mathbf{Z}} f_{\mathbf{H}}(\mathbf{Z}) \text{ subject to } \quad z_{i,k} \geq 0 \qquad (9)$$

and

$$\min_{\mathbf{H}} f_{\mathbf{Z}}(\mathbf{H}) \text{ subject to } \quad h_{k,j} \geq 0. \qquad (10)$$

The method requires the calculation of the first- and the second-order gradients of the two functions in (8)

$$\begin{aligned} \nabla f_{\mathbf{Z}}(\mathbf{H}) &= \mathbf{Z}^T(\mathbf{ZH} - \mathbf{X}) \\ \nabla^2 f_{\mathbf{Z}}(\mathbf{H}) &= \mathbf{Z}^T\mathbf{Z} \\ \nabla f_{\mathbf{H}}(\mathbf{Z}) &= (\mathbf{ZH} - \mathbf{X})\mathbf{H}^T + \gamma \nabla \mathrm{tr}[\tilde{\mathbf{S}}_w] - \delta \nabla \mathrm{tr}[\tilde{\mathbf{S}}_b] \\ \nabla^2 f_{\mathbf{H}}(\mathbf{Z}) &= \mathbf{HH}^T + \gamma \nabla^2 \mathrm{tr}[\tilde{\mathbf{S}}_w] - \delta \nabla^2 \mathrm{tr}[\tilde{\mathbf{S}}_b]. \end{aligned} \qquad (11)$$

The detailed calculations of $\nabla \mathrm{tr}[\tilde{\mathbf{S}}_w]$, $\nabla \mathrm{tr}[\tilde{\mathbf{S}}_b]$, $\nabla^2 \mathrm{tr}[\tilde{\mathbf{S}}_w]$, and $\nabla^2 \mathrm{tr}[\tilde{\mathbf{S}}_b]$ can be found in the Appendix. The projected gradient DNMF method is an iterative method that is comprised of two main phases. These two phases are iteratively repeated until the ending condition is met or the number of iterations exceeds a given number. In the first phase, an iterative procedure is followed for the optimization of (9) while, in the second phase, a similar procedure is followed for the optimization of (10). In the beginning, the bases matrix $\mathbf{Z}^{(1)}$ and the weight matrix $\mathbf{H}^{(1)}$ are initialized either randomly or by using a structured initialization [15], [16] in such a way that their entries are non-negative. The regularization parameters γ and δ that are used to balance the tradeoff between accuracy of the approximation and discriminant decomposition of the computed solution and their selection are typically problem dependent.

A. Solving the Subproblem (9)

Consider the subproblem of optimizing with respect to $\mathbf{Z}$, while keeping the matrix $\mathbf{H}$ constant. The optimization is an iterative procedure that is repeated until $\mathbf{Z}^{(t)}$ becomes a stationary point of (9). In every iteration, a proper step size a_t is required to update the matrix $\mathbf{Z}^{(t)}$. When a proper update is found, the stationarity condition is checked and, if met, the procedure stops.

1) Update the Matrix $\mathbf{Z}$*:* For a number of iterations $t = 1, 2, \ldots$, the following updates are performed [14]:

$$\mathbf{Z}^{(t+1)} = P\left[\mathbf{Z}^{(t)} - a_t \nabla f_{\mathbf{H}}(\mathbf{Z}^{(t)})\right] \qquad (12)$$

where $a_t = \beta^{g_t}$ and g_t are the first non-negative integers such that

$$f_{\mathbf{H}}(\mathbf{Z}^{(t+1)}) - f_{\mathbf{H}}(\mathbf{Z}^{(t)}) \leq \sigma \left\langle \nabla f_{\mathbf{H}}(\mathbf{Z}^{(t)}), \mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)} \right\rangle. \qquad (13)$$

The projection rule $P[\cdot] = \max[\cdot, 0]$ refers to the elements of the matrix and guarantees that the update will not contain any negative entries. The operator $\langle \cdot, \cdot \rangle$ is the inner product between matrices defined as

$$\langle \mathbf{A}, \mathbf{B} \rangle = \sum_i \sum_j a_{i,j} b_{i,j} \tag{14}$$

where $[\mathbf{A}]_{i,j} = a_{i,j}$ and $[\mathbf{B}]_{i,j} = b_{i,j}$. The condition (13) ensures the sufficient decrease of the $f_{\mathbf{H}}(\mathbf{Z})$ function values per iteration. Since the function $f_{\mathbf{H}}$ is quadratic in terms of $\mathbf{Z}$, the inequality (13) can be reformulated as

$$(1-\sigma)\left\langle \nabla f_{\mathbf{H}}(\mathbf{Z}^{(t)}), \mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)} \right\rangle + \frac{1}{2}\left\langle \mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}, \nabla^2 f_{\mathbf{H}}(\mathbf{Z}^{(t+1)}) \right\rangle \leq 0 \tag{15}$$

which is the actual condition checked.

The search of a proper value for a_t is the most time-consuming procedure, thus, as few iteration steps as possible are desired. Several procedures have been proposed for the selection and update of the a_t values [17], [18]. The Algorithm 4 in [14] has been used in our experiments and β, σ are chosen to be equal to 0.1 and 0.01 ($0 < \beta < 1$, $0 < \sigma < 1$), respectively. These values are typical values used in other projected gradient methods as [14]. The choice of σ has been thoroughly studied in [14], [17], [18]. During the experiments, it was observed that a smaller value of β reduces the step size more aggressively, but it may also result in a step size that is too small. The search for a_t is repeated until the point $\mathbf{Z}^{(t)}$ becomes a stationary point.

2) Check of Stationarity: In this step, the limit point is checked as to whether the first-order derivatives are close to zero (stationarity condition). A commonly used condition to check the stationarity of a point is the following [17]:

$$\left\| \nabla^P f_{\mathbf{H}}(\mathbf{Z}^{(t)}) \right\|_F \leq \epsilon_{\mathbf{Z}} \left\| \nabla f_{\mathbf{H}}(\mathbf{Z}^{(1)}) \right\|_F \tag{16}$$

where $\nabla^P f_{\mathbf{H}}(\mathbf{Z})$ is the projected gradient for the constraint optimization problem defined as

$$[\nabla^P f_{\mathbf{H}}(\mathbf{Z})]_{i,k} = \begin{cases} [\nabla f_{\mathbf{H}}(\mathbf{Z})]_{i,k}, & \text{if } z_{i,k} > 0 \\ \min(0, [\nabla f_{\mathbf{H}}(\mathbf{Z})]_{i,k}), & z_{i,k} = 0 \end{cases} \tag{17}$$

and $0 < \epsilon_{\mathbf{Z}} < 1$ is the predefined stopping tolerance. A very low $\epsilon_{\mathbf{Z}}$ (i.e., $\epsilon_{\mathbf{Z}} \approx 0$) leads to a termination after a large number of iterations. On the other hand, a tolerance close to one will result in a premature iteration termination.

B. Solving the Subproblem (10)

A similar procedure should be followed in order to find a stationary point for the subproblem (10) while keeping the matrix $\mathbf{Z}$ fixed and optimizing with respect to $\mathbf{H}$. A value for a_t is iteratively sought and the weight matrix is updated according to

$$\mathbf{H}^{(t+1)} = P\left[\mathbf{H}^{(t)} - a_t \nabla f_{\mathbf{Z}}(\mathbf{H}^{(t)})\right] \tag{18}$$

until the function $f_{\mathbf{Z}}(\mathbf{H})$ value is sufficient decreased and the following inequality holds $\langle a, b \rangle$:

$$(1-\sigma)\left\langle \nabla f_{\mathbf{Z}}(\mathbf{H}^{(t)}), \mathbf{H}^{(t+1)} - \mathbf{H}^{(t)} \right\rangle + \frac{1}{2}\left\langle \mathbf{H}^{(t+1)} - \mathbf{H}^{(t)}, \nabla^2 f_{\mathbf{Z}}(\mathbf{H}^{(t+1)}) \right\rangle \leq 0. \tag{19}$$

This procedure is repeated until the limit point $\mathbf{H}^{(t)}$ is stationary. The stationarity is checked using a similar criterion to (16), i.e.,

$$\left\| \nabla^P f_{\mathbf{Z}}(\mathbf{H}^{(t)}) \right\|_F \leq \epsilon_{\mathbf{H}} \left\| \nabla f_{\mathbf{Z}}(\mathbf{H}^{(1)}) \right\|_F \tag{20}$$

where $\epsilon_{\mathbf{H}}$ is the predefined stopping tolerance for this subproblem.

C. Convergence Rule

The procedure followed for the minimization of the two subproblems, in Sections III-A and Section III-B, is iteratively followed until the global convergence rule is met:

$$\|\nabla f(\mathbf{H}^{(t)})\|_F + \|\nabla f(\mathbf{Z}^{(t)})\|_F \leq \epsilon \left(\|\nabla f(\mathbf{H}^{(1)})\|_F + \|\nabla f(\mathbf{Z}^{(1)})\|_F \right) \tag{21}$$

which checks the stationarity of the solution pair $\mathbf{H}^{(t)}$, $\mathbf{Z}^{(t)}$.

IV. Experimental Results

The proposed DNMF method will be denoted as projected gradient DNMF (PGDNMF) from now on. It has been applied to the frontal verification and facial expression recognition problems.

A. Frontal Face Verification Experiments

The experiments were conducted in the XM2VTS database using the protocol described in [19]. The images were aligned semiautomatically according to the eyes position of each facial image using the eye coordinates. The facial images were downscaled to a resolution of 64 × 64 pixels. Histogram equalization was used for the normalization of the facial image luminance.

The XM2VTS database contains 295 subjects, four recording sessions, and two shots (repetitions) per recording session. It provides two experimental setups, namely Configuration I and Configuration II [19]. Each configuration is divided into three different sets: the training set, the evaluation set, and the test set. The training set is used to create client and impostor models for each person. The evaluation set is used to learn the verification decision thresholds. In case of multimodal systems, the evaluation set is also used to train the fusion manager [19]. For both configurations, the training set has 200 clients, 25 evaluation impostors, and 70 test impostors. The two configurations differ in the distribution of client training and client evaluation data. For additional details concerning the XM2VTS database, an interested reader can refer to [19].

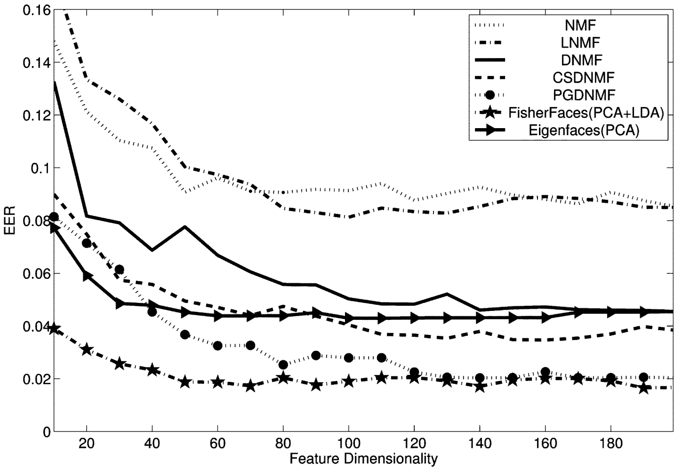

Fig. 1. EER for configuration I of XM2VTS versus dimensionality.

The experimental procedure followed in the experiments was the one also used in [9]. For comparison reasons, the same methodology using Configuration I of the XM2VTS database was used. The performance of the algorithms is quoted by the equal error rate (EER) which is the scalar figure of merit that is often used to judge the performance of a verification algorithm. An interested reader may refer to [9] and [19] for more details concerning the XM2VTS protocol and the experimental procedure followed. In Fig. 1, the verification results are shown for the various tested approaches: NMF [2]; LNMF [3]; DNMF [9]; class-specific DNMF [9]; PCA [20]; PCA plus LDA [21]; and the proposed PGDNMF. EER is plotted versus the dimensionality of the new lower dimension space. As can be seen, the proposed PGDNMF algorithm outperforms (giving a best $\mathrm{EER} \approx 2.0\%$) all of the other part-based approaches and PCA. The best performance of LDA has been 1.7%, which is very close to the best performance of PGDNMF.

Moreover, in order to further boost the verification performance, we have employed support vector machines (SVMs) during training and testing [22] in all of the tested approaches (producing NMF + SVMs, DNMF + SVMs, etc. approaches). For the training of SVMs, we have used additional samples from the evaluation dataset. The best EER achieved for the PGDNMF + SVMs has been measured at 0.8% while for LDA + SVMs at 0.7%. As can be seen, PGDNMF is as good as LDA in the XM2VTS database, while it outperforms all of the other tested approaches. In the XM2VTS database contest [23], an LDA classifier has been among the best in Configuration I.

B. Facial Expression Recognition Experiments

The database used for the facial expression recognition experiments was created using the Cohn–Kanade database [24]. This database is annotated with FAUs. These combinations of FAUs were translated into facial expressions according to [25] in order to define the corresponding ground truth for the facial expressions. All of the subjects were taken into consideration and their differences images, created by subtracting the neutral image intensity values from the corresponding values of the fully expressive facial expression image, were calculated. Each differences image was initially normalized, resulting in an image built only from positive values and afterwards scanned row-wise to form a vector $\mathbf{x} \in \Re_{+}^{F}$ of dimension $F = 40 \times 30$ (40 and 30 are the rows and columns of the image, respectively). The differences images are used instead of the original facial expressive images, due to the fact that in the differences images, the facial parts in motion are emphasized. In Fig. 2, an example of the differences images for each facial expression is depicted.

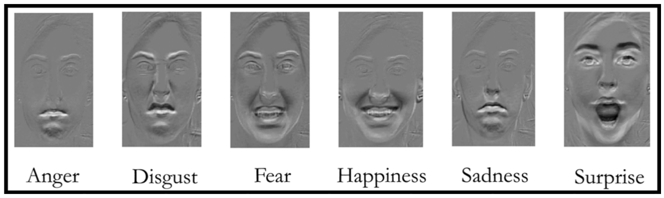

Fig. 2. Differences images for each facial expression for a poser from the Cohn–Kanade database.

In Fig. 3, five basis images extracted from the Cohn–Kanade database for PCA, NMF, LNMF, DNMF, and PGDNMF algorithms are shown. As can be seen, the bases extracted by the proposed algorithm are visually better related to facial parts that participate in expression development than those derived from the other representations.

In the experimental procedure, five sets containing 20% of the data for each of the six facial expression classes, chosen randomly, were created. One set containing 20% of the samples for each class is used as the test set, while the remaining sets form the training set. After the classification procedure is performed, the samples forming the test set are incorporated into the current training set while a new set of samples (20% of the samples for each class) is extracted to form the new test set. The remaining samples create the new training set. This procedure is repeated five times. The average classification accuracy is the mean value of the percentages of the correctly classified facial expressions.

The tested approaches have been the NMF, the LNFM, the DNMF, PCA, PCA plus LDA, and the proposed PGDNMF. Fig. 4 shows the performance of the tested approaches in facial expression recognition using 200 basis images in every approach, except from PCA plus LDA that gives a total of five features (six class problem). As can be seen, the proposed PGDNMF method outperforms all of the other tested part-based approaches in facial expression recognition. The best facial expression recognition accuracies achieved when using NMF, LNFM, DNMF, and PGDNMF were equal to 75.6%, 82.2%, 86.7%, and 88.4%, respectively. Therefore, an increase of the recognition accuracy by 1.7% (in comparison with the DNMF results) is introduced due to the use of the proposed PGDNMF.

In order to boost the performance of all the tested methods, we have incorporated multiclass SVMs [22]. The best performance of the proposed PGDNMF + SVMs has been about 93% followed by the DNMF + SVMs method that achieved a recognition rate of 90%. The LDA method in this problem has not achieved a recognition rate of more than 70%. Thus, the proposed method significantly outperforms LDA in facial expression recognition.

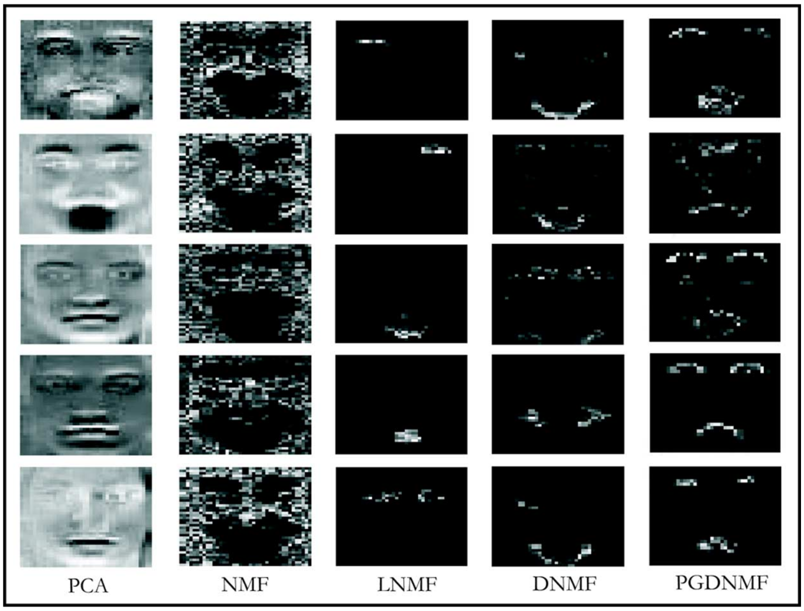

Fig. 3. Basis images extracted for PCA, NMF, LNMF, DNMF, and PGDNMF algorithms from the facial expression experiments in the Cohn–Kanade database.

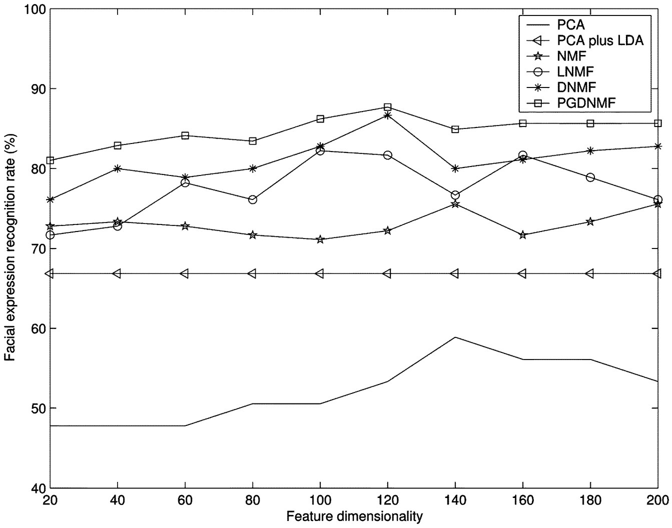

Fig. 4. Facial expression recognition rate versus dimensionality in the Cohn–Kanade database.

Moreover, in order to understand whether the proposed approach is statistically significantly better than the other tested approaches, the McNemar's test [26] has been used for the facial expression recognition experiments. The McNemar's test is a null hypothesis statistical test based on a Bernoulli model. If the resulting value is below a desired significance level (for example, 0.02), the null hypothesis is rejected and the performance difference between two algorithms is considered to be significantly better statistically. Using this test, it has been verified that the proposed PGDNMF + SVMs classifier outperforms the other tested classifiers (i.e., DNMF + SVMs, etc.) in the demonstrated experiments at a significant level of less than $p = 10^{-5}$.

V. Conclusion

A novel DNMF method has been proposed based on projected gradients. The incorporated discriminant constraints focus on the actual features used for classification and not on the weight vectors of the decomposition. We believe that this incorporation results in a classification performance increase. Moreover, we have applied projected gradients in order to ensure that the limit point is stationary. The proposed technique has been applied in supervised facial feature extraction for facial expression recognition and face verification, where it was shown that it outperforms several other subspace methods. We have observed that the basis images obtained from the proposed approach are approximately represented by salient facial features, such as eyes, eyebrows, mouth, etc., which are features that are very important for facial identity verification and facial expression analysis and recognition. Such features cannot be retrieved neither by other holistic representations, such as PCA and LDA, nor by sparse NMF approaches, such as LNMF. A possible drawback of the proposed method, which is actually a drawback of all NMF-based methods, is that it can be sensitive to the initialization of the basis images and the weights. Nevertheless, we have not observed significant variance in performance (i.e., in face verification experiments, the variance has been less than 0.1% in terms of EER and less than 0.3% in terms of the facial expression recognition rate after ten restarts with different initializations). Further research on the topic includes the theoretical investigation on how PGDNMF can be combined with biological-inspired models of vision. These models can be incorporated with the help of proper constraints inside the decomposition. Another interesting topic could be the investigation on how PGDNMF can be used to model receptive fields (e.g., neural receptive fields [12], [27], [28]). Also, future research includes the attempt to create online NMF and DNMF methods.

APPENDIX
CALCULATION OF $\nabla\mathrm{tr}[\tilde{\mathbf{S}}_w]$, $\nabla\mathrm{tr}[\tilde{\mathbf{S}}_b]$, $\nabla^2\mathrm{tr}[\tilde{\mathbf{S}}_w]$ AND $\nabla^2\mathrm{tr}[\tilde{\mathbf{S}}_b]$

Let $\tilde{\mathbf{m}}^{(r)}$ and $\tilde{\mathbf{m}}$ be the mean of the projected vectors $\tilde{\mathbf{x}}$ for the rth class and the total mean vector, respectively. The gradient $\left[\nabla\mathrm{tr}[\tilde{\mathbf{S}}_w]\right]_{i,k} = \partial\mathrm{tr}[\tilde{\mathbf{S}}_w]/\partial z_{i,k}$ is given by

$$\begin{aligned}\frac{\partial\mathrm{tr}[\tilde{\mathbf{S}}_w]}{\partial z_{i,k}} &= \frac{\partial\sum_k\sum_{r=1}^{K}\sum_{\tilde{\mathbf{x}}_j\in\mathcal{U}_r}\left(\tilde{x}_{k,j}-\tilde{m}_k^{(r)}\right)^2}{\partial z_{i,k}}\\ &=\sum_{r=1}^{K}\sum_{\tilde{\mathbf{x}}_j\in\mathcal{U}_r}\frac{\partial\left(\tilde{x}_{k,j}-\tilde{m}_i^{(r)}\right)^2}{\partial z_{i,k}}\\ &=2\sum_{r=1}^{K}\sum_{\tilde{\mathbf{x}}_j\in\mathcal{U}_r}\left(x_{i,j}-m_i^{(r)}\right)\left(\tilde{x}_{k,j}-\tilde{m}_k^{(r)}\right)\end{aligned} \tag{22}$$

since $\tilde{x}_{k,j} = [\tilde{\mathbf{x}}_j]_k = \mathbf{z}_k^T\mathbf{x}_j$ and $\partial\tilde{x}_{j,k}/\partial z_{i,k} = x_{i,j}$.

The $\left[\nabla\mathrm{tr}[\tilde{\mathbf{S}}_b]\right]_{i,k} = (\partial\mathrm{tr}[\tilde{\mathbf{S}}_b]/\partial z_{i,k})$ is given by

$$\begin{aligned}\frac{\partial\mathrm{tr}[\tilde{\mathbf{S}}_b]}{\partial z_{i,k}} &= \frac{\partial\sum_k\sum_{r=1}^{K}\left(\tilde{m}_{k,j}^{(r)}-\tilde{m}_k\right)^2}{\partial z_{i,k}}\\ &=\sum_{r=1}^{K}\frac{\partial\left(\tilde{m}_{k,j}^{(r)}-\tilde{m}_i\right)^2}{\partial z_{i,k}}\\ &=2\sum_{r=1}^{K}\left(m_{i,j}^{(r)}-m_i\right)\left(\tilde{m}_{k,j}^{(r)}-\tilde{m}_k\right).\end{aligned} \tag{23}$$

For the second partial derivative of $\mathrm{tr}[\tilde{\mathbf{S}}_w]$ and of $\mathrm{tr}[\tilde{\mathbf{S}}_b]$, $(\partial^2\mathrm{tr}[\tilde{\mathbf{S}}_w]/\partial z_{i,k}\partial z_{i,l}) = 0$ and $(\partial^2\mathrm{tr}[\tilde{\mathbf{S}}_b]/\partial z_{i,k}\partial z_{i,l}) = 0$ for $l \neq k$, while for $l = k$

$$\begin{aligned}\frac{\partial^2\mathrm{tr}[\tilde{\mathbf{S}}_w]}{\partial^2 z_{i,k}} &= 2\sum_{r=1}^{K}\sum_{\mathbf{x}_j\in\mathcal{U}_r}\left(x_{i,j}-m_i^{(r)}\right)^2 \text{ and}\\ \frac{\partial^2\mathrm{tr}[\tilde{\mathbf{S}}_b]}{\partial^2 z_{i,k}} &= 2\sum_{r=1}^{K}\left(m_{i,j}^{(r)}-m_i\right)^2\end{aligned} \tag{24}$$

where $\mathbf{m}^{(r)}$ and $\mathbf{m}$ are the mean vectors of the vectors $\mathbf{x}$ for the rth class and the total mean vector, respectively. Using the above calculations, the calculation of $\nabla\mathrm{tr}[\tilde{\mathbf{S}}_w]$, $\nabla\mathrm{tr}[\tilde{\mathbf{S}}_b]$, $\nabla^2\mathrm{tr}[\tilde{\mathbf{S}}_w]$ and $\nabla^2\mathrm{tr}[\tilde{\mathbf{S}}_b]$ is now straightforward.

REFERENCES

[1] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, pp. 788–791, 1999.

[2] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Proc. NIPS*, 2000, pp. 556–562.

[3] S. Z. Li, X. W. Hou, and H. J. Zhang, "Learning spatially localized, parts-based representation," in *Proc. CVPR*, Kauai, HI, Dec. 8–14, 2001, pp. 207–212.

[4] I. Buciu and I. Pitas, "Application of non-negative and local non negative matrix factorization to facial expression recognition," in *Proc. ICPR*, Cambridge, U.K., Aug. 23–26, 2004, pp. 288–291.

[5] I. Buciu and I. Pitas, "A new sparse image representation algorithm applied to facial expression recognition," in *Proc. MLSP*, Sao Luís, Brazil, Oct. 1, 2004.

[6] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learning Res.*, vol. 5, pp. 1457–1469, 2004.

[7] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Non-negative matrix factorization framework for face recognition," *Int. J. Pattern Recognition Artif. Intell.*, vol. 19, no. 4, pp. 1–17, 2005.

[8] A. Pascual-Montano, J. M. Carazo, K. Kochi, D. Lehmann, and R. D. Pascual-Marqui, "Nonsmooth nonnegative matrix factorization (nsNMF)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 403–415, Mar. 2006.

[9] S. Zafeiriou, A. Tefas, I. Buciu, and I. Pitas, "Exploiting discriminant information in nonnegative matrix factorization with application to frontal face verification," *IEEE Trans. Neural Netw.*, vol. 17, no. 3, pp. 683–695, May 2006.

[10] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 1, pp. 103–108, Jan. 1990.

[11] D. Donoho and V. Stodden, "When does non-negative matrix factorization give a correct decomposition into parts?," *Advances Neural Inf. Process. Syst.*, vol. 17, 2004.

[12] I. Buciu and I. Pitas, "NMF, LNMF, and DNMF modeling of neural receptive fields involved in human facial expression perception," *J. Vis. Commun. Image Representation*, vol. 17, no. 5, pp. 958–969, Oct. 2006.

[13] E. Gonzalez and Y. Zhang, "Accelarating the Lee-Seung algorithm for nonnegative matrix factorization," Rice Univ., Houston, TX, Tech. Rep. TR-05-02, 2005.

[14] C.-J. Lin, Projected Gradient Methods for Non-Negative Matrix Factorization, Dept. Comput. Sci., National Taiwan Univ., Tech. Rep. Inform. Support Service ISSTECH-95-013, 2005.

[15] S. Wild, J. Curry, and A. Dougherty, "Improving non-negative matrix factorizations through structured initialization," *Pattern Recognit.*, vol. 37, pp. 2217–2232, 2004.

[16] I. Buciu, N. Nikolaidis, and I. Pitas, "On the initialization of the DNMF algorithm," in *Proc. of IEEE Int. Symp. Circuits Syst.*, Kos, Greece, May 21–24, 2006.

[17] C.-J. Lin and J. J. More, "Newton's method for large-scale bound constrained problems," *SIAM J. Optim.*, vol. 9, pp. 1100–1127, 1999.

[18] P. H. Calamai and J. J. More, "Projected gradient methods for linearly constrained problems," *Math. Programming*, vol. 39, pp. 93–116, 1987.

[19] K. Messer, J. Matas, J. V. Kittler, J. Luettin, and G. Maitre, "XM2VTSDB: The extended M2VTS database," in *Proc. AVBPA*, Washington, DC, Mar. 22–23, 1999, pp. 72–77.

[20] M. Turk and A. P. Pentland, "Eigenfaces for recognition," *J. Cognitive Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[21] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[22] V. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[23] K. Messer, J. V. Kittler, M. Sadeghi, S. Marcel, C. Marcel, S. Bengio, F. Cardinaux, C. Sanderson, J. Czyz, L. Vandendorpe, S. Srisuk, M. Petrou, W. Kurutach, A. Kadyrov, R. Paredes, B. Kepenekci, F. B. Tek, G. B. Akar, F. Deravi, and N. Mavity, "Face verification competition on the XM2VTS database," in *Proc. AVBPA*, Guildford, U.K., Jun. 9–11, 2003, pp. 964–974.

[24] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," in *Proc. IEEE Int. Conf. Face and Gesture Recognition*, Mar. 2000, pp. 46–53.

[25] M. Pantic and L. J. M. Rothkrantz, "Expert system for automatic analysis of facial expressions," *Image Vis. Comput.*, vol. 18, no. 11, pp. 881–905, Aug. 2000.

[26] J. Devore and R. Peck, *Statistics: The Exploration and Analysis of Data*, 3rd ed. Pacific Grove, CA: Brooks/Cole, 1997.

[27] P. O. Hoyer, "Non-negative sparse coding," in *Proc. IEEE Workshop Neural Networks for Signal Processing*, Martigny, Switzerland, Sep. 4–6, 2002, pp. 557–565.

[28] P. O. Hoyer, "Modeling receptive fields with non-negative sparse coding," *Neurocomputing*, vol. 52–54, pp. 547–552, 2003.

Irene Kotsia was born in Kastoria, Greece, in 1980. She received the B.S. degree in informatics from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2002, where she is currently pursuing the Ph.D. degree in the Department of Informatics at the University of Thessaloniki.

Currently, she is a Researcher and Teaching Assistant. Her research interests are in the areas of facial expression recognition from static images and image sequences as well as in the area of graphics and animation.

Stefanos Zafeiriou was born in Thessaloniki, Greece, in 1981. He received the B.Sc. (Hons.) and Ph.D. degrees in informatics from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2003 and 2007, respectively.

Currently, he is a Researcher and Teaching Assistant in the Department of Informatics at the Aristotle University of Thessaloniki. His current research interests are in the areas of signal and image processing, computational intelligence, pattern recognition, and computer vision, as well as in the area of watermarking for copyright protection and authentication of digital media. He has coauthored many journal and conference publications.

Dr. Zafeiriou has received various scholarships and awards during his undergraduate and Ph.D. studies.

Ioannis Pitas (SM'94–F'07) received the D.Eng. and Ph.D. degrees in electrical engineering from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 1980 and 1985, respectively.

Currently, he is a Professor at the Department of Informatics, Aristotle University of Thessaloniki. From 1980 to 1993, he was Scientific Assistant, Lecturer, Assistant Professor, and Associate Professor in the Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki. He was a Visiting Research Associate at the University of Toronto, Toronto, ON, Canada; University of Erlangen, Nuernberg, Germany; Tampere University of Technology, Tampere, Finland; and Visiting Assistant Professor at the University of Toronto and as Visiting Professor at the University of British Columbia, Vancouver, BC, Canada. He was a Lecturer in short courses for continuing education. He has published many journal and conference papers and contributed to many books in his areas of interest. He is the co-author of the books "*Nonlinear Digital Filters: Principles and Applications*" (Kluwer, 1990), "*3-D Image Processing Algorithms*" (Wiley, 2000), *Nonlinear Model-Based Image/Video Processing and Analysis* (Wiley, 2001), and author of "*Digital Image Processing Algorithms and Applications*" (Wiley, 2000). He is the editor of the book "*Parallel Algorithms and Architectures for Digital Image Processing, Computer Vision and Neural Networks*" (Wiley, 1993).

Dr. Pitas has also been an invited speaker and/or member of the program committee of several scientific conferences and workshops. In the past, he was Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON IMAGE PROCESSING, *EURASIP Journal on Applied Signal Processing*, and co-editor of *Multidimensional Systems and Signal Processing*. He was General Chair of the 1995 IEEE Workshop on Nonlinear Signal and Image Processing (NSIP95), Technical Chair of the 1998 European Signal Processing Conference, and General Chair of IEEE ICIP 2001. His current interests are in the areas of digital image and video processing and analysis, multidimensional signal processing, watermarking, and computer vision.