Recent Advances in Discriminant Non-negative Matrix Factorization

Symeon Nikitidis, Anastasios Tefas and Ioannis Pitas
Department of Informatics, Aristotle University of Thessaloniki
Thessaloniki, Greece, 54124
Email: {nikitidis,tefas,pitas}@aiia.csd.auth.gr

***Abstract*—Non-negative Matrix Factorization (NMF) is among the most popular subspace methods widely used in a variety of pattern recognition applications. Recently, a discriminant NMF method that incorporates Linear Discriminant Analysis criteria and achieves an efficient decomposition of the provided data to its salient parts has been proposed. An extension of this work specialized for classification, optimized using projected gradients in order to ensure converge to a stationary limit point, resulted in a more efficient method of the latter approach. Assuming multimodality of the underlying data samples distribution and incorporating clustering discriminant inspired constraints into the NMF decomposition cost function, resulted in the Subclass Discriminant NMF algorithm which found to outperform both approaches under real life settings. In this work we review all these methods in the context of various pattern recognition problems using facial images.**

I. Introduction

It is common knowledge that the spatial facial image dimensionality is much higher than that exploited by many facial image analysis applications. This fact necessitates to seek for efficient dimensionality reduction methods for appropriate facial feature extraction, which will alleviate computational complexity and boost the performance of the succeeding facial features processing algorithms. Such a popular category of methods, is the subspace image representation algorithms which aim to discover the latent facial features by projecting the facial image to a linear (or nonlinear) low dimensional subspace, where a certain criterion is optimized.

Non-negative Matrix Factorization (NMF) [1], is a popular subspace learning algorithm widely used in image processing. It is an unsupervised data matrix decomposition method that requires both the data matrix being decomposed and the yielding factors to contain non-negative elements. The non-negativity constraint imposed in the NMF decomposition implies that the original data are reconstructed using only additive and no subtractive combinations of the yielding basic elements. This limitation distinguishes NMF from many other traditional dimensionality reduction methods, such as Principal Component Analysis (PCA), Independent Component Analysis (ICA) or Singular Value Decomposition (SVD).

Focusing on facial image analysis, numerous specialized NMF decomposition variants have been proposed for face recognition [2], [3], facial identity verification [4] and facial expression recognition [5], [6]. In such applications the entire facial image forms a feature vector and NMF aims to find its projections that optimize a given criterion. The resulting projections are then used in order to map unknown test facial images from the original high dimensional image space into a lower dimensional subspace, where the criterion under consideration is optimized.

In this paper we briefly review NMF algorithm and its discriminant counterpart that incorporates Linear Discriminant Analysis criteria in order to achieve a more efficient decomposition of the provided data to their salient parts. Moreover, we propose the Subclass Discriminant NMF algorithm which is able to enhance class separability in the reduced dimensional projection subspace when data samples distribution is multimodal and demonstrate its optimization in two different frameworks.

II. NMF Basics

In the following, without losing generality, we shall assume that the decomposed data are facial images. Obviously, the techniques that will be described can be applied to any kind of non-negative data. NMF approximates a facial image by a linear combination of elements, the so called basis images, that correspond to facial parts. Given a non-negative data matrix $\mathbf{X} \in R_+^{F\times L}$ whose columns are vectorized F-dimensional facial images, NMF attempts to perform the following factorization:

$$\mathbf{X} \approx \mathbf{Z}\mathbf{H} \tag{1}$$

where $\mathbf{Z} \in R_+^{F\times M}$ is a matrix containing the basis images, while matrix $\mathbf{H} \in R_+^{M\times L}$ contains the linear combination coefficients required to reconstruct each original facial image. To measure the cost of the decomposition in (1), the most common approximation error measures for NMF factorization methods are the Kullback-Leibler (KL) divergence metric and the matrix Frobenius norm. The KL divergence between two vectors $\mathbf{x} = [x_1 \dots x_F]^T$ and $\mathbf{q} = [q_1 \dots q_F]^T$ is defined as:

$$KL(\mathbf{x}||\mathbf{q}) \triangleq \sum_{i=1}^{F} \left(x_i \ln \frac{x_i}{q_i} + q_i - x_i \right). \tag{2}$$

Thus, the cost of the decomposition can be measured as the sum of all KL divergences between all original images and their respective reconstructed versions:

$$\mathcal{O}_{KL}(\mathbf{X}||\mathbf{ZH}) = \sum_{j=1}^{L} KL(\mathbf{x}_j||\mathbf{Zh}_j) = \tag{3}$$

$$= \sum_{j=1}^{L}\sum_{i=1}^{F}\left(x_{i,j}\ln(\frac{x_{i,j}}{\sum_k z_{i,k}h_{k,j}}) + \sum_k z_{i,k}h_{k,j} - x_{i,j}\right).$$

Frobenius norm measures the Euclidean distance between two matrices $\mathbf{A}$ and $\mathbf{B}$ as:

$$||\mathbf{A}-\mathbf{B}||_F = \sqrt{\sum_{i,j}\left(A_{i,j}-B_{i,j}\right)^2}. \tag{4}$$

Consequently, the decomposition cost is evaluated as:

$$\mathcal{O}_F(\mathbf{X}||\mathbf{ZH}) \quad = \quad ||\mathbf{X}-\mathbf{ZH}||_F^2 = \sum_{j=1}^{L}\sum_{i=1}^{F}(x_{i,j}-[\mathbf{ZH}]_{i,j})^2 \tag{5}$$

where $||.||_F$ is the Frobenius norm. NMF algorithm factorizes the data matrix $\mathbf{X}$ into $\mathbf{ZH}$, by solving the following optimization problem:

$$\min_{\mathbf{Z},\mathbf{H}} \mathcal{O}(\mathbf{X}||\mathbf{ZH}) \tag{6}$$

$$\text{subject to:} \qquad z_{i,k} \geq 0 \quad , h_{k,j} \geq 0, \quad \forall i,j,k.$$

Considering the KL-divergence based NMF, it has been shown in [7] that using an appropriately designed auxiliary function, the following multiplicative update rules update $h_{k,j}$ and $z_{i,k}$, yielding the desired factors, while guarantee a non increasing behavior of the cost function in (3):

$$h_{k,j}^{(t)} \quad = \quad h_{k,j}^{(t-1)}\frac{\sum_i z_{i,k}^{(t-1)}\frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)}h_{l,j}^{(t-1)}}}{\sum_i z_{i,k}^{(t-1)}}, \tag{7}$$

$$z_{i,k}^{(t)} \quad = \quad z_{i,k}^{(t-1)}\frac{\sum_j h_{k,j}^{(t)}\frac{x_{i,j}}{\sum_l z_{i,l}^{(t-1)}h_{l,j}^{(t)}}}{\sum_j h_{k,j}^{(t)}}. \tag{8}$$

Following a similar optimization strategy, the desired factors for the NMF algorithm based on the Frobenius norm, are derived by:

$$h_{k,j}^{(t)} \quad = \quad h_{k,j}^{(t-1)}\frac{[\mathbf{Z}^{(t-1)^T}\mathbf{X}]_{k,j}}{[\mathbf{Z}^{(t-1)T}\mathbf{Z}^{(t-1)}\mathbf{H}^{(t-1)}]_{k,j}}, \tag{9}$$

$$z_{i,k}^{(t)} \quad = \quad z_{i,k}^{(t-1)}\frac{[\mathbf{XH}^{(t)^T}]_{i,k}}{[\mathbf{Z}^{(t-1)}\mathbf{H}^{(t)}\mathbf{H}^{(t)^T}]_{i,k}}. \tag{10}$$

III. Discriminant NMF Variants

Next we will describe supervised NMF learning variants that incorporate discriminant constraints in order to provide a more efficient decomposition of the decomposed data to their discriminant parts.

A. Discriminant NMF

Discriminant Non-negative Matrix Factorization (DNMF) [4], [8] algorithm is an attempt to introduce discriminant constraints in the NMF decomposition cost function. DNMF aims to find projections that enhance class separability in the reduced dimensional projection subspace and basis images that correspond to discriminant salient facial parts such as eyes, nose, mouth, eyebrows etc.

In order to incorporate discriminant constraints into the NMF decomposition, the well known Fisher discriminant criterion [9] is exploited, given by:

$$J(\mathbf{\Psi}) = \frac{\text{tr}[\mathbf{\Psi}^T\mathbf{\Sigma}_b\mathbf{\Psi}]}{\text{tr}[\mathbf{\Psi}^T\mathbf{\Sigma}_w\mathbf{\Psi}]} \tag{11}$$

where tr[.] is the matrix trace operator. Fisher criterion attempts to find a transformation matrix $\mathbf{\Psi}$, that maximizes the ratio defined by the traces of the between-class and within-class scatter matrices $\acute{\mathbf{\Sigma}}_b = \mathbf{\Psi}^T\mathbf{\Sigma}_b\mathbf{\Psi}$ and $\acute{\mathbf{\Sigma}}_w = \mathbf{\Psi}^T\mathbf{\Sigma}_w\mathbf{\Psi}$, respectively, both evaluated over the projected data. DNMF cost function incorporates a discriminant factor, requiring the dispersion of the projected samples that belong to the same class around their corresponding mean to be as small as possible, while at the same time the scatter of the mean vectors of all classes around their global mean to be as large as possible. Consequently, the DNMF algorithm that considers the KL-divergence metric to measure the decomposition error, minimizes the following cost function:

$$\mathcal{O}_{DNMF}(\mathbf{X}||\mathbf{ZH}) = \mathcal{O}_{KL}(\mathbf{X}||\mathbf{ZH}) + \alpha\text{tr}[\acute{\mathbf{\Sigma}}_w] - \beta\text{tr}[\acute{\mathbf{\Sigma}}_b] \tag{12}$$

where α and β are positive constants. Using a similar optimization methodology as that followed by the NMF algorithm, the multiplicative update rule shown in (13) evaluate the weight coefficients $h_{k,j}$ that belong to the r-th class. Parameter T is defined as:

$$T = (2\alpha + 2\beta)\left(\frac{1}{N_r}\sum_{\lambda,\lambda\neq l} h_{k,\lambda}^{(t-1)}\right) - 2\beta\mu_k^{(r)} - 1 \tag{14}$$

where N_r denotes the number of samples of the r-th class and μ_k the k-th element of the mean vector $\boldsymbol{\mu}^{(r)}$ evaluated over the projected samples of the r-th class. On the other hand, performing optimization with respect to $\mathbf{Z}$, leads to the update formulae in (8) used by the original NMF algorithm, since the incorporated discriminant factor is independent from the basis images matrix $\mathbf{Z}$.

B. Subclass Discriminant NMF

Unfortunately, the considered by DNMF discriminant factor possesses certain shortcomings that arise from the LDA optimality assumptions. That is, it assumes that the sample vectors of each class are generated from underlying multivariate Gaussian distributions having a common covariance matrix but with different class means. Moreover, since it regards that each class is represented by a single compact data cluster, the problem of nonlinearly separable classes cannot be solved.

$$h_{k,l}^{(t)} = \frac{T + \sqrt{T^2 + 4\left(2\alpha - [2\alpha + 2\beta]\,\frac{1}{N_r}\right) h_{k,l}^{(t-1)} \sum_i z_{i,k}^{(t-1)} \frac{x_{i,l}}{\sum_n z_{i,n}^{(t-1)} h_{n,l}^{(t-1)}}}}{2\left(2\alpha - [2\alpha + 2\beta]\,\frac{1}{N_r}\right)} \tag{13}$$

However, this problem can be tackled if we consider that each class is partitioned into a set of disjoint subclasses and perform a discriminant analysis aiming at subclass separation between those belonging to different classes. Typically, in real world applications, data usually do have a subclass structure. In order to overcome these deficiencies we have recently proposed the Subclass Discriminant NMF (SDNMF) algorithm [10].

To overcome the aforementioned limitations, SDNMF relaxes the assumption that each class is expected to consist of a single compact data cluster and regards that data inside each class form various subclasses, where each one is approximated by a Gaussian distribution. Consequently, it approximates the underlying distribution of each class by a mixture of Gaussians and exploits discriminant criteria inspired by the Clustering based Discriminant Analysis (CDA) introduced in [11]. To formulate the SDNMF problem we modify the NMF algorithm by embedding appropriate discriminant constraints and adjust the cost function that drives the optimization process. This extension provides discriminant projections that are expected to enhance class separability in the reduced dimensional space when data samples distribution is multimodal.

To facilitate CDA in the n-class facial image data matrix, let us denote the number of subclasses composing the r-th class by C_r, the total number of formed subclasses by C, where $C = \sum_i^n C_i$, and the number of facial images belonging to the θ-th subclass of the r-th class by $N_{(r)(\theta)}$. Let us also define the projected ρ-th facial image that belongs to the θ-th subclass of the r-th class by the M-dimensional feature vector $\boldsymbol{\eta}_\rho^{(r)(\theta)} = [\eta_{\rho,1}^{(r)(\theta)} \ldots \eta_{\rho,M}^{(r)(\theta)}]^T$ resulting by applying the transformation $\boldsymbol{\eta}_\rho^{(r)(\theta)} = \mathbf{Z}^\dagger \mathbf{x}_\rho^{(r)(\theta)}$ and the mean vector for the θ-th subclass of the r-th class by $\boldsymbol{\mu}^{(r)(\theta)} = [\mu_1^{(r)(\theta)} \ldots \mu_M^{(r)(\theta)}]^T$ which is evaluated over the $N_{(r)(\theta)}$ projected facial images. Using the above notations we can define the within subclass scatter matrix $\mathbf{S}_w$ as:

$$\mathbf{S}_w = \sum_{r=1}^{n} \sum_{\theta=1}^{C_r} \sum_{\rho=1}^{N_{(r)(\theta)}} \left(\boldsymbol{\eta}_\rho^{(r)(\theta)} - \boldsymbol{\mu}^{(r)(\theta)}\right)\left(\boldsymbol{\eta}_\rho^{(r)(\theta)} - \boldsymbol{\mu}^{(r)(\theta)}\right)^T \tag{15}$$

and the between subclass scatter matrix $\mathbf{S}_b$ as:

$$\mathbf{S}_b = \sum_{i=1}^{n} \sum_{r,r\neq i}^{n} \sum_{j=1}^{C_i} \sum_{\theta=1}^{C_r} \left(\boldsymbol{\mu}^{(i)(j)} - \boldsymbol{\mu}^{(r)(\theta)}\right)\left(\boldsymbol{\mu}^{(i)(j)} - \boldsymbol{\mu}^{(r)(\theta)}\right)^T . \tag{16}$$

Adding appropriate penalty terms in the NMF decomposition the new cost function for the SDNMF problem is formulated as follows:

$$\mathcal{O}_{SDNMF}(\mathbf{X}||\mathbf{ZH}) = \mathcal{O}_{KL}(\mathbf{X}||\mathbf{ZH}) + \frac{\alpha}{2}\text{tr}[\mathbf{S}_w] - \frac{\beta}{2}\text{tr}[\mathbf{S}_b] \tag{17}$$

where α and β are positive constants, while $\frac{1}{2}$ is used to simplify subsequent mathematical derivations. Consequently, the new minimization problem is formulated as:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{H}} \mathcal{O}_{SDNMF}(\mathbf{X}||\mathbf{ZH}) \\ \text{s.t.:} \quad & z_{i,k} \geq 0 \quad , h_{k,j} \geq 0,\ \forall i,j,k, \end{aligned} \tag{18}$$

which requires the minimization of (17) subject to the non-negativity constraints applied on the elements of both the weights matrix $\mathbf{H}$ and the basis images matrix $\mathbf{Z}$.

To solve the SDNMF constrained optimization problem we introduce Lagrangian multipliers $\boldsymbol{\phi} = [\phi_{i,k}] \in R^{F\times M}$ and $\boldsymbol{\psi} = [\psi_{j,k}] \in R^{M\times L}$ each one associated with constraints $z_{i,k} \geq 0$ and $h_{k,j} \geq 0$, respectively. Thus, the Lagrangian function $\mathcal{L}$ is formulated as:

$$\mathcal{L} = \mathcal{O}_{KL}(\mathbf{X}||\mathbf{ZH}) + \frac{\alpha}{2}\text{tr}[\mathbf{S}_w] - \frac{\beta}{2}\text{tr}[\mathbf{S}_b] + \text{tr}[\boldsymbol{\phi}\mathbf{Z}^T] + \text{tr}[\boldsymbol{\psi}\mathbf{H}^T]. \tag{19}$$

Consequently, the optimization problem in (18) is equivalent to the minimization of the Lagrangian $\arg\min_{\mathbf{Z},\mathbf{H}} \mathcal{L}$. To minimize $\mathcal{L}$, we first obtain its partial derivatives with respect to $z_{i,j}$ and $h_{i,j}$ and set them equal to zero:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial h_{i,j}} &= -\sum_k \frac{x_{k,j} z_{k,i}}{\sum_l z_{k,l} h_{l,j}} + \sum_l z_{l,i} + \psi_{i,j} + \frac{\alpha}{2}\frac{\partial \text{tr}[\mathbf{S}_w]}{\partial h_{i,j}} \\ &- \frac{\beta}{2}\frac{\partial \text{tr}[\mathbf{S}_b]}{\partial h_{i,j}} = 0 \end{aligned} \tag{20}$$

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial z_{i,j}} &= -\sum_l \frac{x_{i,l} h_{j,l}}{\sum_k z_{i,k} h_{k,l}} + \sum_l h_{j,l} + \phi_{i,j} + \frac{\alpha}{2}\frac{\partial \text{tr}[\mathbf{S}_w]}{\partial z_{i,j}} \\ &- \frac{\beta}{2}\frac{\partial \text{tr}[\mathbf{S}_b]}{\partial z_{i,j}} = 0. \end{aligned} \tag{21}$$

According to KKT conditions [12] it is valid that $\phi_{i,j} z_{i,j} = 0$ and also $\psi_{i,j} h_{i,j} = 0$. Consequently, we obtain the following equalities:

$$\begin{aligned} \left(\frac{\partial \mathcal{L}}{\partial h_{i,j}}\right) h_{i,j} = 0 \Leftrightarrow & -\sum_k \frac{x_{k,j} z_{k,i}}{\sum_l z_{k,l} h_{l,j}} h_{i,j} + \sum_l z_{l,i} h_{i,j} \\ & + \alpha\left(h_{i,j} - \mu_i^{(r)(\theta)}\right) h_{i,j} - \frac{\beta}{N_{(r)(\theta)}} \mu_i^{(r)(\theta)} \left(C - C_r\right) h_{i,j} \\ & + \beta N_{(r)(\theta)} \sum_{m,m\neq r}^{n} \sum_{g=1}^{C_m} \mu_i^{(m)(g)} h_{i,j} = 0 \end{aligned} \tag{22}$$

$$\left(\frac{\partial \mathcal{L}}{\partial z_{i,j}}\right) z_{i,j} = 0 \Leftrightarrow -\sum_l \frac{x_{i,l} h_{j,l}}{\sum_k z_{i,k} h_{k,l}} z_{i,j} + \sum_l h_{j,l} z_{i,j} = 0. \tag{23}$$

Solving the quadratic function for $h_{i,j}$, resulting from equation (22), leads to the multiplicative update rule shown in (24). On the other hand, the update rule in (8) is directly derived by solving (23) for $z_{i,j}$. In (24) $h_{i,j}$ denotes the i-th feature element, in the projection subspace, of the j-th image belonging

$$h_{i,j}^{(t)} = \frac{A + \sqrt{A^2 + 4\left(\alpha - \left[\alpha + \frac{\beta}{N_{(r)(\theta)}}\left(C - C_r\right)\right]\frac{1}{N_{(r)(\theta)}}\right) h_{i,j}^{(t-1)} \sum_k z_{k,i}^{(t-1)} \frac{x_{k,j}}{\sum_n z_{k,n}^{(t-1)} h_{n,j}^{(t-1)}}}}{2\left(\alpha - \left[\alpha + \frac{\beta}{N_{(r)(\theta)}}\left(C - C_r\right)\right]\frac{1}{N_{(r)(\theta)}}\right)}, \tag{24}$$

to the θ-th subclass of the r-th facial class and A is defined as:

$$\begin{aligned} A &= \left(\alpha + \frac{\beta}{N_{(r)(\theta)}}(C - C_r)\right)\frac{1}{N_{(r)(\theta)}} \sum_{\lambda,\lambda\neq j} h_{i,\lambda}^{(t-1)} \\ &- \frac{\beta}{N_{(r)(\theta)}} \sum_{m,m\neq r}^{n} \sum_{g=1}^{C_m} \mu_i^{(m)(g)} - 1. \end{aligned} \tag{25}$$

As can be seen the developed multiplicative update rules for the SDNMF algorithm consider not only sample class labels but also their subclass origin.

C. Projected Gradient DNMF

Recent studies [13], [14] regarding the optimization properties of the derived multiplicative update rules have revealed that they only guarantee a non increasing behavior of the considered cost function and do not ensure that optimization converges to a limit point that is also stationary. In NMF-based optimization problems, stationarity is an important property, since all relevant objective functions are non-convex and there is no guarantee that every limit point in a sequence of iterations corresponds to a local minimum.

In order to exploit the well established optimization properties of [13], [14] that ensure stationarity of the reached limit point, Projected Gradient DNMF (PGDNMF) has been introduced in [6]. PGDNMF algorithm considers the following cost function:

$$\mathcal{J}(\mathbf{Z},\mathbf{H}) = \mathcal{O}_F(\mathbf{X}||\mathbf{ZH}) + \alpha \text{tr}[\tilde{\mathbf{\Sigma}}_w] - \beta \text{tr}[\tilde{\mathbf{\Sigma}}_b] \tag{26}$$

where the within class scatter matrix $\tilde{\mathbf{\Sigma}}_w$ and the between class scatter matrix $\tilde{\mathbf{\Sigma}}_b$ are evaluated using vectors $\tilde{\mathbf{x}}_j = \mathbf{Z}^T\mathbf{x}_j$ which are the actual features used for classification. Since, the cost function in (26) is convex either for $\mathbf{Z}$ or $\mathbf{H}$ but non-convex for both variables, we formulate two subproblems, by keeping one variable fixed and performing optimization for the other:

$$\min_{\mathbf{Z}} \mathcal{J}_1(\mathbf{Z}) \quad \text{subject to:} \quad z_{i,k} \geq 0, \quad \forall i,k \tag{27}$$

$$\min_{\mathbf{H}} \mathcal{J}_2(\mathbf{H}) \quad \text{subject to:} \quad h_{k,j} \geq 0, \quad \forall k,j. \tag{28}$$

1) Optimization of $\mathbf{Z}$ *solving the subproblem (27):* The performed optimization is an iterative steepest descent process that at a given iteration round t the following update rule is applied:

$$\mathbf{Z}^{(t)} = P[\mathbf{Z}^{(t-1)} - \alpha_t \nabla \mathcal{J}_1(\mathbf{Z}^{(t-1)})], \tag{29}$$

where the operator $P[.] = \max[.,0]$ guarantees that no negative values can be assigned to the updated elements of matrix $\mathbf{Z}$ and α_t is the learning step parameter for the t-th iteration.

By iterating the update rule in (29), a sequence of minimizers $\{\mathbf{Z}^{(t)}\}_{t=1}^{\infty}$ of $\mathcal{J}_1(\mathbf{Z})$ is generated and according to Bertsekas [15], it is guaranteed that a stationary point is found among its limit points. Thus, in order to verify whether the currently reached limit point is stationary or not, a stationarity check step [16] is performed, which examines whether the following condition is satisfied:

$$||\nabla^P \mathcal{J}_1(\mathbf{Z}^{(t)})||_F \leq e_{\mathbf{Z}} ||\nabla^P \mathcal{J}_1(\mathbf{Z}^{(1)})||_F, \tag{30}$$

where $\nabla^P \mathcal{J}_1(\mathbf{Z}^{(t)})$ is the projected gradient of $\mathcal{J}_1(\mathbf{Z}^{(t)})$, with respect to $\mathbf{Z}$, with its (i,k)-th element defined as:

$$[\nabla^P \mathcal{J}_1(\mathbf{Z}^{(t)})]_{i,k} = \begin{cases} [\nabla \mathcal{J}_1(\mathbf{Z}^{(t)})]_{i,k} & \text{,if } z_{i,k} > 0 \\ \min\left(0, [\nabla \mathcal{J}_1(\mathbf{Z}^{(t)})]_{i,k}\right) & \text{,if } z_{i,k} = 0 \end{cases} \tag{31}$$

and $e_{\mathbf{Z}}$ is a predefined stopping tolerance satisfying: $0 < e_{\mathbf{Z}} < 1$. A similar strategy is followed for the optimization of $\mathbf{H}$ solving the subproblem in (28). The iterative projected gradient optimization framework generates a sequence of minimizers $\{\mathbf{Z}^{(t)}, \mathbf{H}^{(t)}\}_{t=1}^{\infty}$ until the reached limit point is a stationary point of (18).

The minimization of both subproblems in (27) and (28) involves the calculation of the first and second order gradients of $\mathcal{J}_1(\mathbf{Z})$ and $\mathcal{J}_2(\mathbf{H})$ which are evaluated as follows:

$$\nabla \mathcal{J}_1(\mathbf{Z}) = \mathbf{ZHH}^T - \mathbf{XH}^T + \alpha \nabla \text{tr}[\tilde{\mathbf{\Sigma}}_w] - \beta \nabla \text{tr}[\tilde{\mathbf{\Sigma}}_b] \tag{32}$$

$$\nabla^2 \mathcal{J}_1(\mathbf{Z}) = \mathbf{HH}^T \otimes \mathbf{I}_M + \alpha \nabla^2 \text{tr}[\tilde{\mathbf{\Sigma}}_w] - \beta \nabla^2 \text{tr}[\tilde{\mathbf{\Sigma}}_b] \tag{33}$$

$$\nabla \mathcal{J}_2(\mathbf{H}) = \mathbf{Z}^T\mathbf{ZH} - \mathbf{Z}^T\mathbf{X} \tag{34}$$

$$\nabla^2 \mathcal{J}_2(\mathbf{H}) = \mathbf{Z}^T\mathbf{Z} \tag{35}$$

where $\otimes$ denotes the Kronecker product operation and $\mathbf{I}_M$ is a $M \times M$ identity matrix.

IV. Experimental Results

We compare the performance of the presented SDNMF method, considering its multiplicative optimization updates, with those of the DNMF and the conventional NMF algorithms for face recognition on the Extended Yale B database [17] and for facial expression recognition on the Cohn-Kanade [18] dataset.

A. Facial Expression Recognition in the Cohn-Kanade (CK) dataset

The CK AU-Coded facial expression database is among the most popular databases for benchmarking methods that perform automatic facial expression recognition. In order to

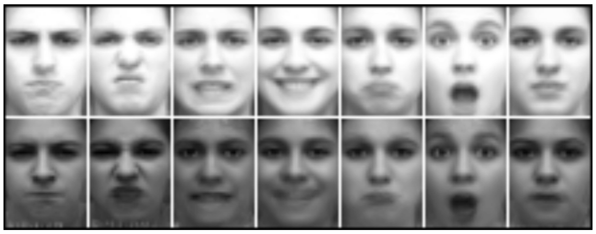

Fig. 1. Mean images for each expression considering that each facial expression class is partitioned into three subclasses. Mean images are derived from the two more distant subclasses inside every class. The diverse illumination conditions during facial expression capture in the CK database are evident.

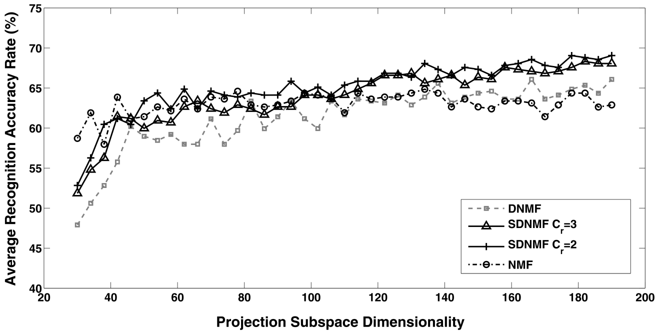

Fig. 2. Average facial expression recognition accuracy rates versus the dimensionality of the projection subspace in CK database.

form our data collection we only acquired a single video frame from each sequence, depicting a subject performing a facial expression at its highest intensity level. Consequently, face detection was performed and the resulting facial Regions Of Interest (ROIs) were manually aligned with respect to the eyes position and anisotropically scaled to a fixed size of 30×40 pixels. In total 407 expressive images were acquired which were used to compose either the training or the test set. To measure the facial expression recognition accuracy, we randomly partitioned the selected samples into 5-folds and a cross validation performed by feeding the projected discriminant facial expression representations to a linear SVM classifier. Consequently, the reported facial expression recognition accuracy rate is the mean value of the percentages of the correctly classified facial expressions in all 5-folds.

It is important to note that CK database depicts subjects of different racial background under severe illumination variations. Consequently, the data sample vectors do not correspond to one compact cluster per class, a fact that we expect to be successfully handled by the proposed SDNMF algorithm. To verify this, we have considered that each of the seven recognized facial expression classes namely: anger, fear, disgust, happiness, sadness, surprise and neutral is partitioned into three subclasses and the mean expressive image for every subclass of each class is computed. Figure 1 shows the mean image for each facial expression considering the two more distant subclasses inside every class. Clearly the illumination variations are captured during clustering.

Since the available samples for each expression class are relatively few (around 50) we have considered only class partitioning into two and three distinct subclasses. Figure 2, shows the measured average facial expression recognition accuracy rates versus the projection subspace dimensionality. The highest measured recognition accuracy rate attained by the proposed method is 69.05%, while for the NMF algorithm is 64.85%. Therefore, an increase by more than 4% has been achieved by incorporating the CDA inspired discriminant constraints in the NMF cost function. As can be seen, in Figure 2, SDNMF constantly outperforms both NMF and DNMF methods, when considering projections in a subspace of dimensionality greater than 100.

B. *Face Recognition on the Extended Yale B database*

Extended Yale B database consists of $2,414$ frontal face images of 38 individuals, captured under various laboratory controlled lighting conditions. For our experiments we have randomly selected for each subject half of the images for training, while the rest were used for testing. Searching for the optimal projection subspace, we have trained NMF, DNMF and SDNMF algorithms considering subspaces of dimensionality varying from 120 to 500. Moreover, since on average there are available 64 images for each subject, thus approximately 32 samples for each class for training, we have considered for the SDNMF algorithm that each class is composed by either two or three disjoint subclasses.

Figure 3 shows the attained face recognition accuracy rates of each examined method versus the projection subspace dimensionality. NMF achieved a highest recognition rate of 85.9% while, SDNMF considering 2 subclasses partitioning of each class, attained the best performance among the examined methods reaching a recognition rate of 92.7%. The maximum recognition rates for DNMF and SDNMF with $C_r = 3$ are 89.5% and 90.1%, respectively.

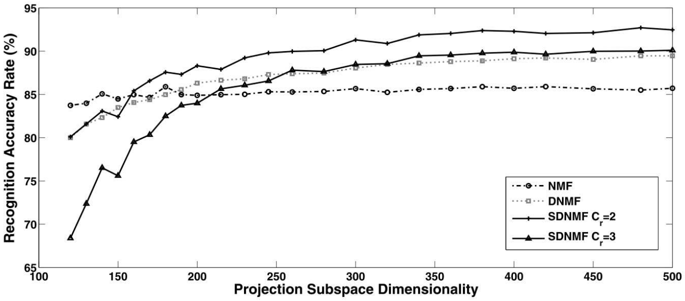

Fig. 3. Face recognition accuracy rates versus the dimensionality of the projection subspace in the Extended Yale B database.

V. Conclusion

In this paper we briefly reviewed NMF, DNMF and PGDNMF algorithms and presented SDNMF method which addresses the general problem of finding discriminant projections that enhance class separability by incorporating CDA inspired criteria in the NMF decomposition. To solve the SDNMF minimization problem, we developed multiplicative update rules using an iterative Lagrangian solution. We compared the performance of SDNMF algorithm with NMF and DNMF on two popular datasets for facial expression and face recognition. Experimental results verified the effectiveness of the proposed method on both tasks.

Acknowledgment

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 248434 (MOBISERV)

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] T. Zhang, B. Fang, Y. Tang, G. He, and J. Wen, "Topology preserving non-negative matrix factorization for face recognition," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 574–584, April 2008.

[3] Y. Wang, Y. Jia, C. Hu, and M. Turk, "Non-negative matrix factorization framework for face recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 19, no. 4, pp. 495–511, 2005.

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

[5] I. Buciu and I. Pitas, "A new sparse image representation algorithm applied to facial expression recognition," in *MLSP*, Sao Luis, Brazil, Sept. 29 - Oct. 1 2004, pp. 539–548.

[6] I. Kotsia, S. Zafeiriou, and I. Pitas, "A novel discriminant non-negative matrix factorization algorithm with applications to facial image characterization problems," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 588–595, September 2007.

[7] D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," in *Advances in Neural Information Processing Systems (NIPS)*, 2000, pp. 556–562.

[8] I. Buciu and I. Pitas, "NMF, LNMF, and DNMF modeling of neural receptive fields involved in human facial expression perception," *Journal of Visual Communication and Image Representation*, vol. 17, no. 5, pp. 958–969, 2006.

[9] R. Fisher, "The statistical utilization of multiple measurements," *Annals of Eugenics*, vol. 8, pp. 376–386, 1938.

[10] S. Nikitidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Facial expression recognition using clustering discriminant non-negative matrix factorization," in *IEEE International Conference in Image Processing (ICIP2011)*, Brussels, Belgium, September 11 - 14 2011.

[11] X. Chen and T. Huang, "Facial expression recognition: a clustering-based approach," *Pattern Recognition Letters*, vol. 24, no. 9-10, pp. 1295–1302, 2003.

[12] R. Fletcher, *Practical methods of optimization; (2nd ed.)*. New York, NY, USA: Wiley-Interscience, 1987.

[13] C. Lin, "On the convergence of multiplicative update algorithms for non-negative matrix factorization," *IEEE Transactions on Neural Networks*, vol. 18, no. 6, pp. 1589–1596, November 2007.

[14] C.-J. Lin, "Projected gradient methods for nonnegative matrix factorization," *Neural Computation*, vol. 19, no. 10, pp. 2756–2779, 2007.

[15] D. Bertsekas, "On the Goldstein-Levitin-Polyak gradient projection method," *IEEE Transactions on Automatic Control*, vol. 21, no. 2, pp. 174–184, April 1976.

[16] C. Lin and J. Moré, "Newton's method for large bound-constrained optimization problems," *SIAM Journal on Optimization*, vol. 9, no. 4, pp. 1100–1127, 1999.

[17] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, June 2001.

[18] T. Kanade, J. Cohn, and Y. Tian, "Comprehensive database for facial expression analysis," March 2000, pp. 46–53.