USING SUBCLASSES IN DISCRIMINANT NON-NEGATIVE SUBSPACE LEARNING FOR FACIAL EXPRESSION RECOGNITION

Symeon Nikitidis, Anastasios Tefas and Ioannis Pitas

Aristotle University of Thessaloniki, Department of Informatics, Greece
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 image processing problems. To achieve an efficient decomposition of the provided data to its discriminant parts, thus enhancing classification performance, we regard that data inside each class form clusters and use criteria inspired by Clustering based Discriminant Analysis. The proposed method combines these discriminant criteria as constraints in the NMF decomposition cost function in order to address the problem of finding discriminant projections that enhance class separability in the reduced dimensional projection space. The developed algorithm has been applied to the facial expression recognition problem and experimental results verified that it successfully identified discriminant facial parts, thus enhancing recognition performance.

1. INTRODUCTION

NMF [12], is an unsupervised matrix decomposition algorithm 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 algorithms, such as, Principal Component Analysis (PCA) [10], Independent Component Analysis (ICA) [6] or Singular Value Decomposition (SVD) [9].

Recently, numerous specialized NMF-based algorithms have been proposed applied in various problems in diverse fields. These algorithms are developed based on modifying the NMF decomposition cost function by incorporating additional penalty terms in order to fulfill specific requirements, arising in each application domain. A supervised NMF learning method that aims to extract discriminant facial parts is the Discriminant NMF (DNMF) algorithm proposed in [15]. DNMF incorporates Fisher's criterion in the NMF factorization and achieves a more efficient decomposition of the provided data to its discriminant parts, thus enhancing separability between classes compared with conventional NMF. However, the incorporation of Linear Discriminant Analysis (LDA) [8] inside DNMF poses two certain deficiencies. Firstly, LDA assumes that the sample vectors of each class are generated from underlying multivariate Normal distributions of common covariance matrix but with different means. Secondly, since LDA assumes that each class is represented by a single cluster, the problem of nonlinearly separable classes can not be solved. However, this problem can be tackled if we consider that each class is partitioned into a set of disjoint clusters and perform a discriminant analysis aiming at clusters separation. Unfortunately, in real world applications, data usually do not correspond to compact sets. This is also a common case in the facial expression recognition problem, since there is no unique way that people express certain emotions and moreover, there are confounding factors such as pose, texture and lighting variations, that significantly degrade the performance of NMF-based methods [2].

To remedy the aforementioned limitations, we relax the assumption that each class is expected to consist of a single compact cluster and regard that data inside each class form various clusters, where each one is approximated by a Gaussian distribution. Consequently, we approximate the underlying distribution of each class as a mixture of Gaussians and imply criteria inspired by the Clustering based Discriminant Analysis (CDA) introduced in [5]. Moreover, we extend the NMF algorithm modifying its decomposition by embedding appropriate discriminant constraints and reformulate the cost function that drives the optimization process. With this extension we expect the resulting discriminant projections, from one hand, to pose robustness in illumination changes and variations in expression and on the other hand, to enhance class separability in the reduced dimensional space. To solve the resulting optimization problem, we develop multiplicative update rules that consider not only samples class origin but also clusters formation inside each class.

The rest of the paper is organized as follows. A brief review of the NMF algorithm is given in Section 2. Section 3, introduces the proposed method which incorporates subclass discriminant constraints in the NMF decomposition framework and also, draws the proposed multiplicative update rules. Section 4, describes the conducted experiments that verify the efficiency of our algorithm on the facial expression recognition problem. Finally, concluding remarks are drawn in Section 5.

2. BRIEF REVIEW OF NMF

In this section we briefly present the NMF decomposition concept. In the following, without loosing generality, we will 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.

The basic idea of NMF is to approximate a facial image by a linear combination of basic elements the so called basis images, that correspond to facial parts. The non negativity constraints imply that the combinations of the multiple basis images are practically additions of ideally non-overlapping facial parts that attempt to reconstruct accurately the complete facial image. Let I be a facial image database comprised of L images belonging to n different classes and $\mathbf{X} \in R_+^{F \times L}$ is the data matrix whose columns are F-dimensional feature vectors obtained by scanning row-wise each facial image in the database. Thus $x_{i,j}$ is the i-th element of the j-th column vector $\mathbf{x}_j$. NMF considers factorizations of the form:

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

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 coefficients of the linear combinations of the basis images required to reconstruct each original facial image in the database. Thus the j-th facial image, represented by vector $\mathbf{x}_j$, can be approximated after the NMF decomposition by the factorization $\mathbf{x}_j \approx \mathbf{Zh}_j$, where $\mathbf{h}_j$ denotes the j-th weight column of matrix $\mathbf{H}$. Undoubtedly, useful factorizations for real world applications appear when the linear subspace transformation projects data from the original F-dimensional space to a M-dimensional subspace with $M \ll F$.

To measure the cost of the decomposition in (1), one popular approach is to use the Kullback-Leibler (KL) divergence metric which is a special case of Bregman distances [1]. However, using this metric to measure the decomposition error in (1) poses some certain deficiencies. More precisely, the decomposition cost is not well defined at any point of the bounded region, since the natural logarithm function involved in the KL divergence evaluation is undefined for zero arguments. This fact introduces the limitation to require both elements $x_{i,j}$ and $[\mathbf{ZH}]_{i,j}$ to be strictly positive and consequently, no zero values are allowed.

To overcome this deficiency we use the square of the Frobenius norm in order to measure the NMF decomposition error. The Frobenius norm measures the Euclidean distance between two matrices $\mathbf{A}$ and $\mathbf{B}$ as:

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

Thus the cost of the decomposition in (1) can be measured as the sum of the square Euclidean distances between all images in the database and their respective reconstructed versions, obtained from the factorization. Consequently, the cost function $O(\mathbf{X}||\mathbf{ZH})$ that defines the approximation error of factorizing $\mathbf{X}$ into $\mathbf{ZH}$ is evaluated as:

$$\begin{aligned} O(\mathbf{X}||\mathbf{ZH}) &= ||\mathbf{X}-\mathbf{ZH}||_F^2 = \sum_{j=1}^{L}\sum_{i=1}^{F}\left(x_{i,j}-[\mathbf{ZH}]_{i,j}\right)^2 \\ &= \sum_{j=1}^{L}\sum_{i=1}^{F}\left(x_{i,j}-\sum_{k=1}^{M} z_{i,k}h_{k,j}\right)^2 \end{aligned} \tag{3}$$

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

$$\min_{\mathbf{Z},\mathbf{H}} O(\mathbf{X}||\mathbf{ZH}) \tag{4}$$
$$\text{subject to:} \quad z_{i,k} \geq 0 \quad, h_{k,j} \geq 0, \quad \forall i,j,k.$$

Using an appropriately designed auxiliary function, it has been shown in [13] that 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 $O(\mathbf{X}||\mathbf{ZH})$ defined in (3). The update rule for the t-th iteration for $h_{k,j}^{(t)}$ is given by:

$$h_{k,j}^{(t)} = h_{k,j}^{(t-1)} \frac{[\mathbf{Z}^{(t-1)^T}\mathbf{X}]_{k,j}}{[\mathbf{Z}^{(t-1)^T}\mathbf{Z}^{(t-1)}\mathbf{H}^{(t-1)}]_{k,j}}, \tag{5}$$

while for $z_{i,k}^{(t)}$ the update rule is given by:

$$z_{i,k}^{(t)} = z_{i,k}^{(t-1)} \frac{[\mathbf{X}\mathbf{H}^{(t)^T}]_{i,k}}{[\mathbf{Z}^{(t-1)}\mathbf{H}^{(t)}\mathbf{H}^{(t)^T}]_{i,k}}. \tag{6}$$

3. PROPOSED METHOD

In this section we present the imposed clustering based discriminant criteria and demonstrate how these are incorporated in the NMF decomposition cost function creating the proposed Subclass Discriminant NMF (SDNMF) optimization problem. Next, we derive the proposed multiplicative update rules that solve SDNMF.

3.1 Clustering based discriminant analysis

Similarly to LDA, CDA seeks to determine a transformation matrix such that when applied on the initial input data the resulting transformed samples form classes in the projection subspace that are better separated. To do so, CDA assumes that data inside classes do not correspond to compact sets, but each class is partitioned into one or more clusters and attempts to discriminate classes while at the same time minimizes the scatter within every cluster.

In detail, CDA exploits the Fisher-Rao's criterion modified such as the between and within cluster scatter matrices are evaluated considering except of samples class labels their respective cluster origins. To formulate the CDA criteria in the n-class facial image database I, let us denote the number of clusters composing the r-th class by C_r, the total number of formed clusters in the database by C, where $C = \sum_i^n C_i$, and the number of facial images belonging to the θ-th cluster of the r-th class by $N_{(r)(\theta)}$. Let us also define the mean vector for the θ-th cluster of the r-th class by $\mathbf{m}^{(r)(\theta)} = [m_1^{(r)(\theta)} \dots m_F^{(r)(\theta)}]^T$ which is evaluated over the $N_{(r)(\theta)}$ facial images, while vector $\mathbf{x}_\rho^{(r)(\theta)} = [x_{\rho,1}^{(r)(\theta)} \dots x_{\rho,F}^{(r)(\theta)}]^T$ corresponds to the feature vector of the ρ-th facial image of the θ-th cluster of the r-th class. Using the above notations we can define the within cluster scatter matrix $\mathbf{S}_w$ as:

$$\mathbf{S}_w = \sum_{r=1}^{n}\sum_{\theta=1}^{C_r}\sum_{\rho=1}^{N_{(r)(\theta)}}\left(\mathbf{x}_\rho^{(r)(\theta)}-\mathbf{m}^{(r)(\theta)}\right)\left(\mathbf{x}_\rho^{(r)(\theta)}-\mathbf{m}^{(r)(\theta)}\right)^T \tag{7}$$

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

$$\mathbf{S}_b = \sum_{i=1}^{n}\sum_{r,r\neq i}^{n}\sum_{j=1}^{C_i}\sum_{\theta=1}^{C_r}\left(\mathbf{m}^{(i)(j)}-\mathbf{m}^{(r)(\theta)}\right)\left(\mathbf{m}^{(i)(j)}-\mathbf{m}^{(r)(\theta)}\right)^T. \tag{8}$$

Since NMF projects the initial data to a lower dimensional subspace using the pseudo-inverse $\mathbf{Z}^\dagger = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$ we desire to perform this projection in a discriminant manner and enhance class separability in the projection subspace.

To do so, we apply CDA inspired criteria in order to determine the optimum projection. Thus, we desire the projected facial images to the lower dimensional subspace to maximize the CDA criterion which we formulate evaluating the within and between cluster scatter matrices in the projection subspace. More precisely, the within cluster scatter matrix $\mathbf{\Sigma}_w$ when operates on the projected samples in the lower dimensional subspace is transformed with respect to its previous form as: $\mathbf{\Sigma}_w = \left(\mathbf{Z}^\dagger\right)^T \mathbf{S}_w \mathbf{Z}^\dagger$, while the between cluster scatter matrix as: $\mathbf{\Sigma}_b = \left(\mathbf{Z}^\dagger\right)^T \mathbf{S}_b \mathbf{Z}^\dagger$. Let us define the projected ρ-th facial image by the M-dimensional feature vector $\mathbf{h}_\rho^{(r)(\theta)} = [h_{\rho,1}^{(r)(\theta)} \dots h_{\rho,M}^{(r)(\theta)}]^T$ resulting by applying the transformation $\mathbf{h}_\rho^{(r)(\theta)} = \mathbf{Z}^\dagger \mathbf{x}_\rho^{(r)(\theta)}$. Using the above notations we can evaluate the within cluster scatter matrix $\mathbf{\Sigma}_w$ in the projection subspace as:

$$\mathbf{\Sigma}_w = \sum_{r=1}^{n} \sum_{\theta=1}^{C_r} \sum_{\rho=1}^{N_{(r)(\theta)}} \left(\mathbf{h}_\rho^{(r)(\theta)} - \widetilde{\mathbf{m}}^{(r)(\theta)}\right) \left(\mathbf{h}_\rho^{(r)(\theta)} - \widetilde{\mathbf{m}}^{(r)(\theta)}\right)^T \tag{9}$$

and the between cluster scatter matrix $\mathbf{\Sigma}_b$ as:

$$\mathbf{\Sigma}_b = \sum_{i=1}^{n} \sum_{r,r\neq i}^{n} \sum_{j=1}^{C_i} \sum_{\theta=1}^{C_r} \left(\widetilde{\mathbf{m}}^{(i)(j)} - \widetilde{\mathbf{m}}^{(r)(\theta)}\right) \left(\widetilde{\mathbf{m}}^{(i)(j)} - \widetilde{\mathbf{m}}^{(r)(\theta)}\right)^T \tag{10}$$

where the M-dimensional mean vectors $\widetilde{\mathbf{m}}$ are evaluated over the projected samples and $\widetilde{\mathbf{m}}^{(i)(j)}$ denotes the mean vector evaluated over the projected samples composing the j-th cluster of the i-th class.

Matrix $\mathbf{\Sigma}_w$ represents the scatter of the projected sample vector coefficients around their cluster mean. It is rational to desire after the projection, the dispersion of those samples that belong to the same cluster of a class to be as small as possible, since this would denote a high concentration of these samples around their cluster mean and consequently more compact clusters formation. In order to measure the samples dispersion inside clusters we compute the trace of the within cluster scatter matrix $\mathbf{\Sigma}_w$. Furthermore, matrix $\mathbf{\Sigma}_b$ defines the scatter of the mean vectors between all clusters that belong to different classes. To separate clusters belonging to different classes we desire to maximize the difference between the means of every cluster of a certain class to every cluster of each other class. Therefore, the trace of $\mathbf{\Sigma}_b$ is desired to be as large as possible.

3.2 Subclass Discriminant Non-negative Matrix Factorization Algorithm

In order to incorporate clustering based discriminant constraints derived from CDA in the NMF decomposition, we reformulate the NMF cost function adding appropriate penalty terms. Since we desire in the projection subspace the trace of matrix $\mathbf{\Sigma}_w$ to be as small as possible and at the same time, the trace of $\mathbf{\Sigma}_b$ to be as large as possible, the cost function of the SDNMF algorithm is formulated as:

$$O_{SDNMF}(\mathbf{X}||\mathbf{ZH}) = \frac{1}{2}||\mathbf{X} - \mathbf{ZH}||_F^2 + \frac{\alpha}{2}\mathrm{Tr}[\mathbf{\Sigma}_w] - \frac{\beta}{2}\mathrm{Tr}[\mathbf{\Sigma}_b] \tag{11}$$

where α and β are positive constants, Tr[.] denotes the trace operator, while $\frac{1}{2}$ is used to simplify subsequent derivations. Alternatively, the SDNMF cost function can be written using matrices trace form as follows:

$$\begin{aligned} O_{SDNMF}(\mathbf{X}||\mathbf{ZH}) &= \frac{1}{2}\mathrm{Tr}\left[(\mathbf{X}-\mathbf{ZH})(\mathbf{X}-\mathbf{ZH})^T\right] + \\ +\frac{\alpha}{2}\mathrm{Tr}[\mathbf{\Sigma}_w] - \frac{\beta}{2}\mathrm{Tr}[\mathbf{\Sigma}_b] &= \frac{1}{2}\mathrm{Tr}[\mathbf{XX}^T] - \mathrm{Tr}[\mathbf{ZHX}^T] + \\ +\frac{1}{2}\mathrm{Tr}[\mathbf{ZHH}^T\mathbf{Z}^T] &+ \frac{\alpha}{2}\mathrm{Tr}[\mathbf{\Sigma}_w] - \frac{\beta}{2}\mathrm{Tr}[\mathbf{\Sigma}_b] \end{aligned} \tag{12}$$

where we have applied the matrix properties $\mathrm{Tr}[\mathbf{AB}] = \mathrm{Tr}[\mathbf{BA}]$, $\mathrm{Tr}[\mathbf{A}] = \mathrm{Tr}[\mathbf{A}^T]$ and $||\mathbf{A}||_F^2 = \mathrm{Tr}[\mathbf{AA}^T]$

Consequently, the new minimization problem is formulated as:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{H}} O_{SDNMF}(\mathbf{X}||\mathbf{ZH}) \\ &\text{subject to:} \quad z_{i,k} \geq 0 \quad , h_{k,j} \geq 0, \quad \forall i,j,k. \end{aligned} \tag{13}$$

which requires the minimization of (11) subject to the non-negativity constraints applied on the elements of both the weights matrix $\mathbf{H}$ and the basis images matrix $\mathbf{Z}$.

In order to solve the optimization problem in (13), we follow a similar approach as that in [13]. It should be noted that as in every NMF-based optimization problem the objective function in (11) is convex either in $\mathbf{Z}$ or in $\mathbf{H}$, but non-convex in both variables. Therefore, we do not expect the optimization process of the SDNMF algorithm to reach the global minimum. Instead, the proposed iterative optimization algorithm can be used to find a local minimum. To do so, the proposed process successively optimizes either variable $\mathbf{Z}$ or $\mathbf{H}$ keeping the other fixed.

3.3 Update Rules Derivation for the Optimization of the SDNMF Problem

In order to solve the constrained optimization problem in (13) we introduce Lagrange multipliers $\mathbf{u} \in R_+^{F\times M} = [u_{i,k}]$ and $\mathbf{v} \in R_+^{M\times L} = [v_{j,k}]$ each one associated with each non-negativity constraint for $z_{i,k} \geq 0$ and $h_{k,j} \geq 0$, respectively. Consequently, we formulate the Lagrangian function L as follows:

$$\begin{aligned} L &= \frac{1}{2}\mathrm{Tr}[\mathbf{XX}^T] - \mathrm{Tr}[\mathbf{ZHX}^T] + \frac{1}{2}\mathrm{Tr}[\mathbf{ZHH}^T\mathbf{Z}^T] + \\ &+ \frac{\alpha}{2}\mathrm{Tr}[\mathbf{\Sigma}_w] - \frac{\beta}{2}\mathrm{Tr}[\mathbf{\Sigma}_b] + \sum_{i,k} u_{i,k} z_{i,k} + \sum_{j,k} v_{j,k} h_{j,k} = \\ &= \frac{1}{2}\mathrm{Tr}[\mathbf{XX}^T] - \mathrm{Tr}[\mathbf{ZHX}^T] + \frac{1}{2}\mathrm{Tr}[\mathbf{ZHH}^T\mathbf{Z}^T] + \\ &+ \frac{\alpha}{2}\mathrm{Tr}[\mathbf{\Sigma}_w] - \frac{\beta}{2}\mathrm{Tr}[\mathbf{\Sigma}_b] + \mathrm{Tr}[\mathbf{uZ}^T] + \mathrm{Tr}[\mathbf{vH}^T]. \end{aligned} \tag{14}$$

The optimization problem in equation (13) is equivalent to the minimization of the Lagrangian function $\arg\min_{\mathbf{Z},\mathbf{H}} L$. To minimize L, we first obtain its partial derivatives with respect to $z_{i,k}$ and $h_{k,j}$ and set them equal to zero:

$$\begin{aligned} &[\mathbf{Z}^T\mathbf{ZH}]_{k,j} - [\mathbf{Z}^T\mathbf{X}]_{k,j} + \frac{\alpha}{2}\frac{\partial \mathrm{Tr}[\mathbf{\Sigma}_w]}{\partial h_{k,j}} - \frac{\beta}{2}\frac{\partial \mathrm{Tr}[\mathbf{\Sigma}_b]}{\partial h_{k,j}} + v_{k,j} = 0 \\ &\left[\mathbf{ZHH}^T\right]_{i,k} - [\mathbf{XH}^T]_{i,k} + u_{i,k} = 0. \end{aligned} \tag{15}$$

According to KKT conditions [7] it is valid that $u_{i,k}z_{i,k} = 0$ and also $v_{k,j}h_{k,j} = 0$. Consequently, we obtain the following equalities:

$$\left(\frac{\partial L}{\partial h_{k,j}}\right) h_{k,j} = 0 \Leftrightarrow [\mathbf{Z}^T\mathbf{Z}\mathbf{H}]_{k,j}h_{k,j} - [\mathbf{Z}^T\mathbf{X}]_{k,j}h_{k,j} \quad + \\ +\alpha\left(h_{k,j} - \widetilde{m}_k^{(r)(\theta)}\right) h_{k,j} - \frac{\beta}{N_{(r)(\theta)}}\widetilde{m}_k^{(r)(\theta)}(C - C_r)\, h_{k,j} \quad + \\ + \frac{\beta}{N_{(r)(\theta)}}\sum_{q,q\neq r}^{n}\sum_{g=1}^{C_q}\widetilde{m}_k^{(q)(g)}h_{k,j} = 0 \quad (16)$$

$$\left(\frac{\partial L}{\partial z_{i,k}}\right) z_{i,k} = 0 \Leftrightarrow [\mathbf{Z}\mathbf{H}\mathbf{H}^T]_{i,k}z_{i,k} - [\mathbf{X}\mathbf{H}^T]_{i,k}z_{i,k} = 0. \quad (17)$$

Solving equation (16) for $h_{k,j}$ we derive the proposed multiplicative update rule:

$$h_{k,j}^{(t)} = h_{k,j}^{(t-1)} \frac{[\mathbf{Z}^{(t-1)^T}\mathbf{X}]_{k,j} + \frac{\beta}{N_{(r)(\theta)}}\widetilde{m}_k^{(r)(\theta)}(C - C_r)}{[\mathbf{Z}^{(t-1)^T}\mathbf{Z}^{(t-1)}\mathbf{H}^{(t-1)}]_{k,j} + A} \quad (18)$$

where A is defined as:

$$A = \alpha\left(h_{k,j}^{(t-1)} - \widetilde{m}_k^{(r)(\theta)}\right) + \frac{\beta}{N_{(r)(\theta)}}\sum_{q,q\neq r}^{n}\sum_{g=1}^{C_q}\widetilde{m}_k^{(q)(g)}.$$

On the other hand, solving equation (17) for $z_{i,k}$ we derive the update rule for the basis images matrix $\mathbf{Z}$ as:

$$z_{i,k}^{(t)} = z_{i,k}^{(t-1)} \frac{[\mathbf{X}\mathbf{H}^{(t)^T}]_{i,k}}{[\mathbf{Z}^{(t-1)}\mathbf{H}^{(t)}\mathbf{H}^{(t)^T}]_{i,k}}. \quad (19)$$

After we obtain the optimum factors, SDNMF necessitates to use the pseudo-inverse $\mathbf{Z}^\dagger = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$ of the basis images matrix $\mathbf{Z}$, in order to extract the discriminant features and compute the projection to the lower dimensional feature space for an unknown test sample $\mathbf{x}_j$ as: $\acute{\mathbf{x}}_j = \mathbf{Z}^\dagger\mathbf{x}_j$. However, as it has been shown in [3], $\mathbf{Z}^T$ can be also used as an appropriate alternative for this purpose, since the calculation of $\mathbf{Z}^\dagger$ is not only a computationally intensive task but also may suffer from numerical instability.

We can successively update $\mathbf{Z}$ and $\mathbf{H}$ either until the objective function does not achieve any significant improvement or when a predefined maximum number of iterations is reached. Since the added discriminant factors in the SDNMF cost function are totaly independent from the basis images matrix $\mathbf{Z}$, keeping variable $\mathbf{H}$ fixed and optimizing for $\mathbf{Z}$ results to the same optimization problem as thus optimized by the original NMF algorithm in [13] and consequently, leads to exactly the same update formulae. Thus, we can recall the convergence proof of conventional NMF to show that (11) is non-increasing under the update rules in (19). The interested reader is referred to [13] for more details. The proposed multiplicative update rule in (18) is also guaranteed to cause a non increasing behavior of the objective function. The interested reader is referred to [4, 14] for a detailed proof regarding convergence to a local minimum of other similarly derived updates. Moreover, when setting parameters $\alpha = \beta = 0$ it is obvious that the SDNMF algorithm degenerates to the original NMF method and the update rule in (18) reduce to that of equation (5).

4. EXPERIMENTAL RESULTS

In this section we evaluate the performance of the proposed SDNMF method compared with the DNMF and the conventional NMF algorithm on the Cohn-Kanade [11] facial expression database. Figure 1 shows example images, from the examined dataset, depicting the six basic facial expressions arranged in the following order: anger, fear, disgust, happiness, sadness, surprise and the neutral emotional state.

Figure 1: Sample images depicting the different facial expressions from the Cohn-Kanade database.

In order to form the training and test sets, face detection was performed and the resulting Regions Of Interest (ROIs) were manually aligned with respect to the eyes position. Each extracted facial image was anisotropically scaled, so as to have fixed size of 30×40 pixels (where 30 and 40 are the columns and rows of the image, respectively) and was converted to grayscale. Consequently, each facial image was scanned row-wise so as to form a feature vector $\mathbf{x} = [f_1 \dots f_{1200}]^T$, ($f_i$ being the luminance of the i-th pixel) which is used to compose the training and test sets.

Regarding the training and test sets formation, 5-fold cross validation has been performed using the available data samples. Apparently, the training set has been used to learn the basis images for the low dimensional projection space, while the test set has been used to report the facial expression recognition accuracy rates in the respective learned projection space. Training and testing have been performed by feeding the projected discriminant facial expression representations to a linear SVM classifier. Consequently, recognition accuracy is measured as the percentage of samples in the test set which were correctly classified. The reported average classification accuracy rate is the mean value of the percentages of the correctly classified facial expressions in each fold. Parameters α and β value was defined experimentally. We have found that the optimal values in terms of measured classification accuracy rates and convergence speed where achieved when α and β were set in the interval $(0,\ 1]$. Moreover, in all circumstances these parameters should be carefully defined such as to ensure convexity of the optimization subproblem.

Figure 2 shows the average expression recognition accuracy rates versus the projection subspace dimensionality. The highest measured recognition rates achieved by each examined method, as well as, the respective subspace dimensionality are summarized in Table 1. As it can be seen SDNMF outperforms both NMF and DNMF methods.

5. CONCLUSION

We proposed a novel method that addresses the general problem of finding discriminant projections that enhance class separability in the reduced dimensional space by incorporating CDA in the NMF decomposition. To solve the SDNMF problem, we develop a multiplicative update rule that considers not only samples class origin but also clusters formation

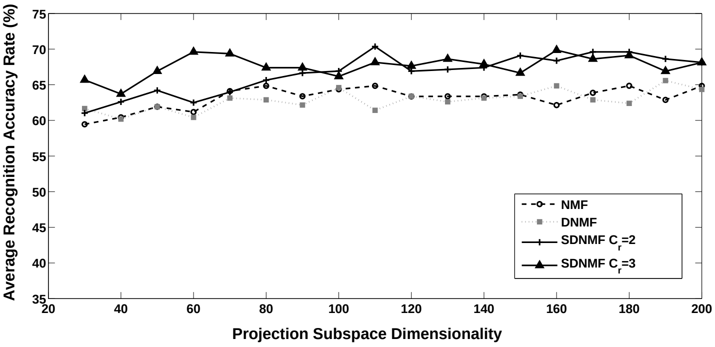

Figure 2: Average facial expression recognition accuracy rate versus the dimensionality of the projection subspace in the Cohn-Kanade database.

Table 1: Best average expression recognition accuracy rates in Cohn-Kanade database

Method	Accuracy Rate	Subspace Dimensionality
SDNMF $C_r = 2$	**70.36%**	110
SDNMF $C_r = 3$	69.86%	160
DNMF	65.59%	190
NMF	64.86%	180

inside each class. We compared the performance of SDNMF algorithm with NMF and DNMF and the experimental results verified the superiority of the proposed method in the facial expression recognition task.

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] L. Bregman. The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. *U.S.S.R. Computational Mathematics and Mathematical Physics*, 1:200–217, 1967.

[2] I. Buciu and I. Nafornita. Non-negative matrix factorization methods for face recognition under extreme lighting variations. In *IEEE International Symposium on Signals, Circuits and Systems (ISSCS)*, pages 1–4, 2009.

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

[4] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized non-negative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, to appear.

[5] X. Chen and T. Huang. Facial expression recognition: a clustering-based approach. *Pattern Recognition Letters*, 24(9-10):1295–1302, 2003.

[6] P. Comon. Independent component analysis, a new concept? *Signal processing*, 36(3):287–314, 1994.

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

[8] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press, second edition, 1990.

[9] G. Golub and C. Van Loan. *Matrix computations*. Johns Hopkins, 3rd edition edition, 1996.

[10] I. Jolliffe. *Principal component analysis*. New York: Springer-Verlag, 1986.

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

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

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

[14] Z. Yang and E. Oja. Linear and nonlinear projective nonnegative matrix factorization. *IEEE Transactions on Neural Networks*, 21(5):734–749, May 2010.

[15] 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*, 17(3):683–695, May 2006.