

An Extended Max-margin Non-negative Matrix Factorization for Face Recognition

Mai Lam

College of Information Technology, Da Nang University
mlam@cit.udn.vn

Abstract. Non-negative matrix factorization (NMF) is a dimension-reduction technique based on a low-rank approximation of the feature space. Unfortunately, most existing NMF based methods are not ready for encoding higher-order data information and ignore the local geometric structure contained in the data set. Additionally, the previous classification approaches which the classification and matrix factorization steps are separated independently. The first one performs data transformation and the second one classifies the transformed data using classification methods as support vector machine (SVM). In this paper, therefore, we joint SVM and constrained NMF into one by uniting maximum margin classification constraints into the constrained NMF optimization. Experimental results on the benchmark image datasets demonstrate the effectiveness of the proposed method.

Keywords: face recognition, graph regularization, nonnegative matrix factorization, support vector machine, spatial constraints

1 Introduction

The face plays an essential role in carrying identity of individuals. Face recognition has been broadly studied by several authors over the last thirty years [1][2][9]. As a consequence great progress has been achieved toward developing computer vision algorithms that can recognize individuals based on their facial images in a similar way that human beings do, and leading this technology to reliable personal identification systems.

Face recognition approaches on still images can be broadly grouped into geometric and template matching techniques. One way to represent the high dimensional face image data is to use Non-negative Matrix Factorization (NMF) algorithms. NMF aims to find intuitive basis such that training examples can be faithfully reconstructed using linear combination of basis images which are restricted to non-negative values.

However, due to NMF pays no attention to class labels of the face images, therefore the representative characteristics of the data may not be optimal for classification task. And most existing NMF based methods are not ready for encoding higher-order data information and ignore the local geometric structure contained in the data set.

In this paper, we introduce a novel algorithm that solves two main challenging problems in traditional NMF methods. Our main contribution in this work is to introduce a robust algorithm which is called as Max-margin Non-negative Matrix Factorization via Spatial and Graph Regularization (MNMF_SGR). We also apply the algorithm for a lower dimensional representation of face images that can be used for both classification and recognition tasks.

2 Related Works

2.1 Non-negative Matrix Factorization

NMF is a part-based subspace learning algorithm, and it gained its popularity after the work of Lee and Seung published [3]. Given a non-negative data matrix $\mathbf{X} \in R_+^{m \times n}$ where every entry in $\mathbf{X} \geq 0$ and a positive integer $k \ll \min(m, n)$, find two non-negative matrices \mathbf{U} and \mathbf{V} , where $\mathbf{U} = [u_1, u_2, \dots, u_k] \in R^{m \times k}$ is the basis vectors and $\mathbf{V} = [v_1, v_2, \dots, v_k] \in R^{n \times k}$ is coefficient vectors, that minimizes the following objective function:

$$\min_{\mathbf{U}, \mathbf{V}} f(\mathbf{U}, \mathbf{V}) = \frac{1}{2} \|\mathbf{X} - \mathbf{UV}\|_F^2, \text{ s. t. } \mathbf{U} \geq 0, \mathbf{V} \geq 0 \quad (1)$$

where $\|\cdot\|_F$ is Frobenius norm of a matrix, and the product \mathbf{UV} is the non-negative matrix factorization approximation of \mathbf{X} of rank at most k .

2.2 NMF Variants

Spatial Non-negative Matrix Factorization (SpaNMF)

The structured sparse NMF has been recently proposed in order to learn structured basis images. In 2012, Zheng et al.[4] produced a structured sparsity learning as automatically as possible. It is a pixel dispersion penalty, which effectively describes the spatial dispersion of pixels in an image without using any manually predefined structured patterns as constraints.

The Pixel Dispersion Penalty.

Let $e_{y,x} (\in R^d)$ be the indicator vector such that

$$e_{y,x}(j) = \begin{cases} 1 & j = (x-1) \times b + y \\ 0 & \text{otherwise,} \end{cases} \quad (2)$$

where $[x, y]$ is a coordinate vector in an image and b is the height of an image. Note that $\mathbf{w}_i^{2D}(y, x) = \mathbf{w}_i^T e_{y,x}$. When w_i is non-negative. The equation becomes

$$\mathbf{D}(\mathbf{w}_i) = \mathbf{w}_i^T \left\{ \sum_{x=1}^a \sum_{y=1}^b \sum_{x'=1}^a \sum_{y'=1}^b l([y, x], [y', x']) \times e_{y,x} e_{y',x'}^T \right\} \mathbf{w}_i \quad (3)$$

and \mathbf{E}_1 is Pixel Dispersion Penalty.

$$\mathbf{E}_1 = \sum_{x=1}^a \sum_{y=1}^b \sum_{x'=1}^a \sum_{y'=1}^b l([y, x], [y', x']) \times e_{y,x} e_{y',x'}^T \quad (4)$$

In order to extract non-negative structured local patterns of the data, we are now incorporating the pixel dispersion penalty to develop a new penalized NMF-based matrix factorization as follows:

$$\min_{\mathbf{U} \in R^{d \times L}, \mathbf{V}_i \in R^L} \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \mathbf{U} \mathbf{v}_i\|_F^2 + \frac{\lambda}{L} \text{trace}(\mathbf{U}^T \mathbf{E}_1 \mathbf{U}), s. t. \mathbf{U} \geq 0, 0 \leq \mathbf{v}_i \leq c_0 \quad (5)$$

where $\lambda \geq 0, L \ll d$ and c_0 is a simple positive constant bound parameter.

Graph Non-negative Matrix Factorization (GNMF)

Recent studies in spectral graph theory and manifold learning theory [5][6] have demonstrated that the local geometric structure can be effectively modeled through a nearest neighbor graph on a scatter of data points. Given a data matrix $\mathbf{X} = [x_{ij}] \in R^{M \times N}$, GNMF aims to find two non-negative matrix $\mathbf{U} = [u_{ik}] \in R^{M \times K}$ and $\mathbf{V} = [v_{jk}] \in R^{N \times K}$. We can also use the distance measure as follow:

$$\min_{\mathbf{U}, \mathbf{V} \geq 0} \frac{1}{N} \sum_{i=1}^N \|\mathbf{x}_i - \mathbf{U} \mathbf{V}^T\|_F^2 + \lambda \text{trace}(\mathbf{V}^T \mathbf{L} \mathbf{V}), s. t. \mathbf{U}, \mathbf{V} \geq 0, \lambda \geq 0 \quad (6)$$

where the regularization parameter $\lambda \geq 0$ controls the smoothness of the new representation.

Max-margin Non-negative Matrix Factorization (MNMF)[7]

Let $\{x_i, y_i\}_{i=1}^L$ denote a set of data vectors and their corresponding labels, where $x_i \in R^m, y_i \in \{-1, 1\}$. This aim is to determine a bases matrix \mathbf{U} that can be used to extract features that are optimal under a max-margin classification criterion. This is accomplished by imposing constraints on the feature vectors derived from \mathbf{U} . The optimization problem is given by

$$\min_{\mathbf{U}, \mathbf{V}, w, b, \varepsilon_i} \lambda \|\mathbf{X} - \mathbf{U} \mathbf{V}\|_F^2 + \frac{1}{2} w^T w + C \sum_{i=1}^L \varepsilon_i \quad (7)$$

$$s. t. y_i (w^T \mathbf{U}^T \mathbf{x}_i + b) \geq 1 - \varepsilon_i, \varepsilon_i \geq 0, 1 \leq i \leq L, \mathbf{V} \geq 0$$

where $\varepsilon = (\varepsilon_1, \dots, \varepsilon_i \dots \varepsilon_L)$ is the lack variable vector, λ and C are scalars.

3 Proposed Method

In this section, we introduce our unified objective function, and then we build the multiplicative update solution by the optimized gradient method. Finally, we describe how the system use this algorithm to perform the tasks we expect it to do.

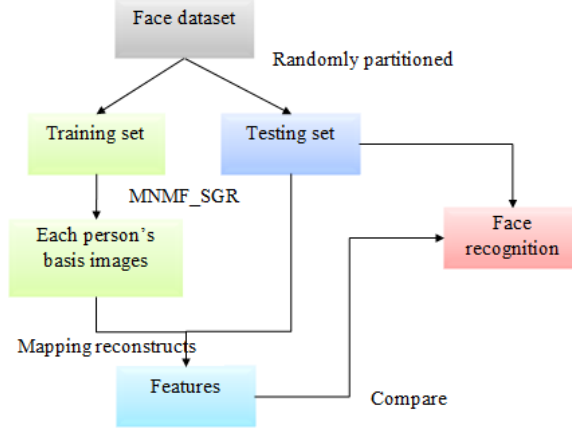


Fig. 1. The flow chart of MNMF_SGR based image reconstruction for face recognition

3.1 Objective Function

The unified objective function is constructed by jointing the data reconstruction objective function in (5) by (6) and (7):

$$\min_{\mathbf{U}, \mathbf{V}, \mathbf{w}, \mathbf{b}, \boldsymbol{\varepsilon}} \|\mathbf{X} - \mathbf{UV}\|_F^2 + \lambda_1 \left(\frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^n \varepsilon_i \right) + \lambda_2 \text{Tr}(\mathbf{VLV}^T) + \lambda_3 \text{Tr}(\mathbf{UEU}^T) \quad (8)$$

$$s. t. \mathbf{U} \geq 0, \mathbf{V} \geq 0, \varepsilon_i \geq 0, y_i(w^T v_i + b) \geq 1 - \varepsilon_i, i = 1..n$$

All variables are divided into three terms: the coefficient matrix (\mathbf{V}), the basis matrix (\mathbf{U}) and variables about max-margin projection (\mathbf{w} , \mathbf{b} , $\boldsymbol{\varepsilon}$). Where \mathbf{L} is called graph Laplacian, \mathbf{E} is called the dispersion kernel matrix.

3.2 Update the Projection Vector and Slack Variables

When the coefficient matrix and the basis matrix are fixed, MMNMF_MR optimization problem changes into the standard binary soft-margin SVM classification

$$\min_{\mathbf{w}, \mathbf{b}, \boldsymbol{\varepsilon}} \lambda_1 \left(\frac{1}{2} \|\mathbf{w}\|_2^2 + C \sum_{i=1}^n \varepsilon_i \right), s. t. \varepsilon_i \geq 0, y_i(w^T v_i + b) \geq 1 - \varepsilon_i, i = 1..n \quad (9)$$

The hyper-plane parameters \mathbf{w} , \mathbf{b} and slack variable vector $\boldsymbol{\varepsilon}$ are obtained using an off-the-shelf SVM classifier.

3.3 Update the Coefficient Matrix

When other variables are fixed, the optimization of the coefficient matrix is transformed to quadratic programming:

$$\min_{\mathbf{V}} \|\mathbf{X} - \mathbf{UV}\|_F^2 + \lambda_2 \text{Tr}(\mathbf{VLV}^T), s. t. \mathbf{V} \geq 0, y_i(w^T v_i + b) \geq 1 - \varepsilon_i, i = 1..n \quad (10)$$

The Lagrangian of above objective function is

$$L(V, \alpha, \beta) = Tr(\mathbf{X} - \mathbf{UV})(\mathbf{X} - \mathbf{UV})^T + \lambda_2 Tr(\mathbf{VLV}^T) - \alpha^T \mathbf{V} - \sum_{i=1}^n \beta_i [y_i(w^T v_i + b) - 1 + \varepsilon_i]$$

where α, β are Lagrangian multipliers, specifically α is Lagrangian multipliers vector. Under the Karush-Kuhn-Tucker (KKT) conditions, we get

$$\begin{aligned} 2\mathbf{U}^T \mathbf{UV} - 2\mathbf{U}^T \mathbf{X} + 2\lambda_2 \mathbf{VL}^T - \alpha - \beta y w &= 0 \\ \mathbf{1}^T V &= 0 \\ y(w^T V - b) - 1 + \varepsilon &= 0 \end{aligned} \quad (11)$$

Transform (11) equation into a matrix form

$$\begin{pmatrix} 2\mathbf{U}^T \mathbf{U} + 2\lambda_2 \mathbf{L}^T & -\mathbf{1}^T & -yw \\ \mathbf{1}^T & 0 & 0 \\ yw^T & 0 & 0 \end{pmatrix} \times \begin{pmatrix} \mathbf{V} \\ \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} 2\mathbf{U}^T \mathbf{X} \\ 0 \\ yb + 1 - \varepsilon \end{pmatrix} \quad (12)$$

where $\mathbf{1}$ is a unit vector whose size is the same as \mathbf{v} , $\mathbf{0}$ is the zero vector. We can derive \mathbf{v} by solving (3.7) equation.

3.4 Update the Basis Matrix

When other variables are fixed, the model is transformed to a non-negative matrix factorization:

$$O_3 = \min_U \|\mathbf{X} - \mathbf{UV}\|_F^2 + \lambda_3 Tr(\mathbf{UEU}^T), s.t. \mathbf{U} \geq 0 \quad (13)$$

Because of the non-negative constraints, we use gradient descent methods to solve this problem. The gradient of equation (13) is

$$\nabla = 2\mathbf{UVV}^T - 2\mathbf{X}^T \mathbf{V}^T + 2\lambda_3 \mathbf{EU}$$

3.5 Classification

During testing, the input test vector x_{test} is projected onto the basis matrix U to obtain the feature vector, $f_{test} = \mathbf{U}^T \mathbf{x}_{test}$. The feature vector is used by the max-margin classifier which predicts the class $\mathbf{y}_{test} = \text{sign}(\mathbf{U}^T f_{test} + b)$ where w, b, \mathbf{U} are computed during training.

4 Experimental Results

In this section, proposed method compared against several popular subspace learning algorithms, specifically the unsupervised methods (NMF, Spatial NMF and Graph

NMF). We also compared with the supervised algorithm (Max-margin NMF [7] and Max-margin nonnegative matrix factorization via factor analysis [8]).

4.1 Database

- YaleA Database [10]: It contains 165 gray-scale images in GIF format of 15 individuals. There are 11 images per subject including center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink.
- ORL Face Database [11]: There are 10 images of each 40 different subjects. Some of the images were taken at different times (e.g., open and closed eyes, smiling and not smiling, with glasses and without glasses).

4.2 Preprocessing

The image region where the face can be found is then cropped and only this region is used in the face recognition process. The face is cropped from the whole image manually depending on the position of the left and right eyes as well as the mouth. After cropping the face from the original image, the new images are normalized to a standard size of 16×16 pixels.

4.3 Experimental Settings

We first evaluated the classification accuracy was carried out on face recognition for the well-known face data sets, namely YaleA and ORL datasets. After that, we reported the classification under partial occlusion on both data sets.

We repeated the following procedure for 10 times. Each time we randomly selected two-thirds of number of image per individual and labeled them. All the other images were unlabeled and used as the testing set. All algorithms were initialized with 20 random U and V matrices, each of them were trained for 20 iterations and the one with the minimum objective function value was further trained for 1000 iterations.

The dimensionality reduction process with NMF, SpaNMF, GNMF and Semi-NMF algorithms, the trained coefficient matrix is ready to be used for classifying a testing face image. Then we use SVM algorithm for the classifiers in the face recognition.

With MNMF, MNMF_FA and MNMF_SGR, after training process we compute the feature vector from the input test vector which is projected onto the basis matrix. After that, this feature vector is used in predicting class of face recognition.

4.4 Experimental Results

Classification Result for Face Recognition

In this subsection, we evaluate the discriminating power of algorithm MNMF_SGR compared with the non-negative matrix factorization algorithms. Figure

2 shows the results of classification of NMF, SpaNMF, GNMF, MNMF, MNMF_FA and MNMF_SGR on the YaleA and ORL database.

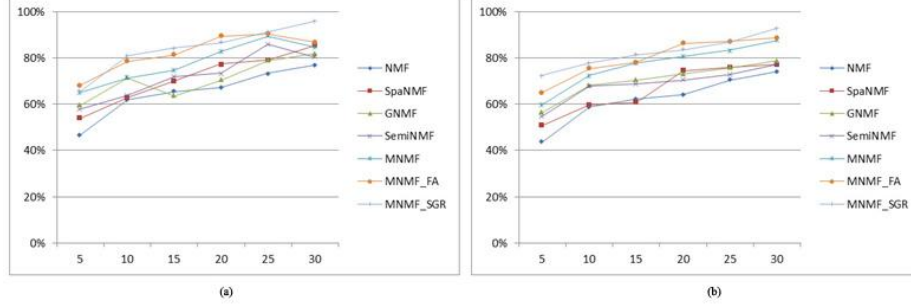


Fig. 2. Face recognition average accuracies (%) of different algorithms on (a) YaleA and (b) ORL datasets

In this experiment, we fixed $\lambda_1 = 1$, $\lambda_2 = 100$ and different values of λ_3 on the YaleA and ORL databases. In both cases, values between 10^{-2} and 10^{-1} of λ_3 performs better than other values on accuracy rate of classification. We set k values (from 5 to 30). The NMF variants achieved better classification accuracy than a basic NMF at each k values and comparing the second best algorithm, MNMF_SGR achieves 2.36% and 1.51% improvement in accuracy on two datasets, respectively.

Classification Results for Partially Occluded Images

In this section, we present the classification results for the values {5%, 10%} of partial occlusion on the ORL and Yale-A datasets. We randomize the place of a partial occlusion on all test images. All algorithms were trained and tested in the same way in section face recognition.



Fig. 3. Sample of ORL test images with 5% partially occluded

Figure 4 shows face recognition average accuracies under partial occlusion condition. We divided that the dataset into two parts, occluded partial on training set and without occluded partial on testing set and set default k at 30. For 5% partial occlusion, the MNMF_SGR has the highest accuracy at 82.21% then, followed by MNMF, and the other four NMF variants. However, for 10% occlusion the order has been little change. The best one is GNMF method, then MNMF and followed by MNMF_SGR. GNMF outperforms MNMF_SGR by 4.06%.

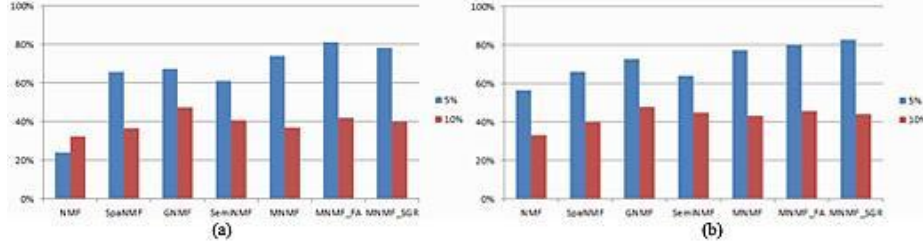


Fig. 4. Face recognition average accuracies (%) with 5% and 10% partial occlusion of different algorithms in three case. (a) YaleA data set (b) ORL dataset.

5 Conclusion

In this paper, our approach had been introduced in the context of face recognition. The results show that our proposed algorithm provides better facial representations and achieves higher recognition rates than standard non-negative matrix factorization and its variants. For future work more studying the convergence rate for proposed method and increasing the efficiency, they should be all in consideration.

Reference

1. W. Zhao and R. Chellappa, "Face recognition: A literature survey," *ACM Comput. Surv.*, vol. 35, no. 4, pp. 399–458, 2003.
2. B. Fasel, J. Luetttin, "Automatic facial expression analysis: a survey", *Pattern Recognition*, 36 (1) (2003), pp. 259-275.
3. D. D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization", *Advances in Neural Information Processing System*, 2000.
4. W. S. Zheng, J. Lai, S. Liao, R. He, "Extracting non-negative basis images using pixel dispersion penalty", *Pattern Recognition*, pp. 2912-2926, 2012.
5. D. Cai, X. He, J. Han and T. Huang, "Graph regularized nonnegative matrix factorization for data representation", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33(8), pp. 1548–1560, 2011.
6. H. Kim and H. Park, "Sparse non-negative matrix factorization via alternating non-negativity constrained least squares for microarray data analysis". *Bioinformatics*, vol. 23, no. 12, pp. 1495-1502, 2007.
7. B. G. Kumar, I. Kotsia and I. Patras, "Max-margin nonnegative matrix factorization", *Image Vision Computing*, pp. 279–291, 2012.
8. D. Liu and X. Tan, "Max-margin non-negative matrix factorization with flexible spatial constraints based on factor analysis", *Frontiers of Computer Science*, vol.10 (2), pp. 302-316, 2016.
9. L. Mai, "Joint Support Vector Machine with Constrained Nonnegative Matrix Factorization and Its Applications", *Master Thesis, National Central University*, July 2017.
10. "The Database of Faces," AT&T Laboratories Cambridge, [Online]. Available: <http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>, 2002.
11. Georgiades, A. "Yale face database." Center for computational Vision and Control at Yale University, <http://cvc.cs.yale.edu/cvc/projects/yalefaces/yalefaces.html> (2002).