Exploring Manifold Structure of Face Images via Multiple Graphs

Masheal Alghamdi
Computer, Electrical and Mathematical Sciences and Engineering Division
King Abdullah University of Science and Technology (KAUST)
Thuwal 23955-6900, Saudi Arabia

ABSTRACT

Geometric structure in the data provides important information for face image recognition and classification tasks. Graph regularized non-negative matrix factorization (GrNMF) performs well in this task. However, it is sensitive to the parameters selection. Wang et al. proposed multiple graph regularized non-negative matrix factorization (MultiGrNMF) to solve the parameter selection problem by testing it on medical images. In this paper, we introduce the MultiGrNMF algorithm in the context of still face Image classification, and conduct a comparative study of NMF, GrNMF, and MultiGrNMF using two well-known face databases. Experimental results show that MultiGrNMF outperforms NMF and GrNMF for most cases.

Keywords: Face recognition, manifold regularization, non-negative matrix factorization (NMF), graph Laplacian.

1. INTRODUCTION

Face image recognition is a challenging problem in the field of computer vision. It plays an important role in many face image verification and identification applications. Verification systems are based on one-to-one matching which compares a face image against a template image. On the other hand, face identification systems are based on one-to-many matching in which a face image is compared against all the template images in the database. Jafri et al. defined the face recognition task as follows: Given an input face image and a database of known individuals faces images, how can we verify or determine the identity of the person in the input image [1]?

There are three types of methods for face image recognition from a still image [1,2]. The first type is holistic approaches such as PCA, LDA, ICA, Fisher discriminate analysis (FDA), and non-negative matrix factorization (NMF) [3,4]. The second category is the feature-based approaches in which the input image is processed to identify and extract distinct facial feature such as eyes, nose, mouth, eyebrow, and other marks. They then calculate geometric relationship among these points which lead to a compact image representation obtained as a vector of geometric features [1,2]. Examples of these methods include Elastic bunch graph matching and dynamic link matching. The third class is the hybrid approaches in which both local features and the whole face is used as inputs [1,2]. Examples of such approaches include Markove random field method, graph-regularized NMF (GrNMF) [5], adaptive GrNMF via feature selection [6,7] and multiple graph regularized NMF (MultiGrNMF) [8].

NMF has been widely used as a popular part-based data representation method. However it fails in discovering the manifold structure of a data set, which plays an important role in face image classification and clustering[5]. To overcome this problem, Cai et al. established a GrNMF by constructing an affinity graph and searching for a matrix factorization that respects the graph structure[5]. However, the performance of GrNMF depends mainly on the selection of the graph model and its corresponding parameters. Usually cross validation or discrete grid search are used in the GrNMF which lead to efficiency and over fitting problems[8]. One solution to these issues was proposed by Wang et al [8], MultiGrNMF, in which an approximation of the intrinsic manifold was obtained by a linear combination of several graphs using different models and parameters.

The rest of this paper is organized as follows. We will briefly review NMF, GrNMF and MultiGrNMF in section 2. Then in section 3 the experimental results on two face image data sets are presented. Finally, we give our concluding remarks in section 4.
2. METHODS

2.1 NMF

NMF represents data using non-negative constraints, which leads to part-based representation since only additive combination of the original data are allowed [9]. We assume that the data samples are organized as a non-negative data matrix $X = [x_1, \ldots, x_n]$, where the $n$-th column of $X$ is the feature vector of the $n$-th data point. NMF seeks to find two lower dimensional matrices $W$ and $H$, such that their product provides a good approximation of $X$. In reality each feature vector is approximated by linear combination of columns of $H$ and weighted by the component of $W$. As a result, $H$ can be regarded as a set of biases vectors and $W$ as a set of coding vectors. A standard NMF determines factorization matrices $H$ and $W$ by minimizing the loss function defined by the Euclidean distance or the divergence between $X$ and $H^*W$. The most commonly used NMF loss function is based on $L_2$ distance between two matrices [10]:

$$O^{NMF}(H, W) = \|X - HW\|^2$$

$$= Tr(X^T X) - 2Tr(X^T HW) + Tr(W^T H^T HW)$$

2.2 GrNMF

NMF knowledge obtained in the Euclidean space fails to discover the intrinsic geometrical and discriminative structure of the data space [5]. GrNMF, on the other hand, encodes the geometrical information by constructing an affinity graph and searching for a matrix factorization that respects the graph structure [5]. The local geometric structure is modeled by the $P$-nearest neighbor graph on the scattering of data points, where each data point is represented by a vertex. For each vertex, $x_j$, GrNMF finds its $P$ nearest neighbors and puts edges between $x_j$ and its neighbors [5]. A number of choices for defining a weight matrix $A$ on the graph are available. Three most commonly used methods are as follows:

- 0-1 weightings is used to regularize NMF and sparse coding in [5], where $A_{nm} = 1$ if and only if $x_n$ and $x_m$ are connected, otherwise $A_{nm} = 0$.

- Heat kernel weighting [5] is defined as follows: $A_{nm} = e^{-\|x_n - x_m\|^2/\sigma^2}$ if and only if $x_n$ and $x_m$ are connected, otherwise $A_{nm} = 0$.

- Histogram intersection kernel weighting [8] is usually used to construct the similarity graph of scale invariant feature transform features as follow $A_{nm} = \sum_{d=1}^{P} \min(x_{dn}, x_{dm})$, if and only if $x_n$ and $x_m$ are connected, otherwise $A_{nm} = 0$.

The objective function for GrNMF is thus defined as:

$$O^{GrNMF}(H, W) = \|X - HW\|^2 + \alpha \frac{1}{2} \sum_{n,m=1}^{N} \|W_n - W_m\|^2 A_{nm}$$

$$= \|X - HW\|^2 + \alpha Tr(WUW^T) - Tr(WAW^T)$$

$$= \|X - HW\|^2 + \alpha Tr(WLW^T)$$

Where $U$ is a diagonal matrix, the entries of which are column sums of $A$, $U_{nn} = \sum_{n=1}^{N} A_{nn}$, $L = U - A$ is the graph Laplacian, and the regularization parameter $\alpha \geq 0$ is a tradeoff parameter to balance the two terms[8].
As argued by [5], graph construction is critical for GrNMF and it depends mainly on the selection of the graph model and its corresponding parameters. Usually cross validation or discrete grid search are used in the GrNMF which lead to efficiency and over fitting problems [8].

2.3 MultiGrNMF

Wang et al. [8] addressed the GrNMF parameter issue by MultiGrNMF, which combines automatic intrinsic manifold approximation and NMF. Intrinsic manifold is obtained by a linear combination of several graphs using different models and parameters. To simplify the idea, by using different graph weight matrix model definitions with different parameters, they could compute several corresponding graph Laplacians. The number of nearest neighbors should also be noted as a parameter for nearest neighbor graph construction. As such, this parameter is not controlled by the user, but is selected automatically by the algorithm introduced as follows. Suppose that we have already computed a set of candidate graph Laplacians \( \{ L_1, ..., L_k \} \). This assumption constrains the search space of possible graph Laplacians as linear combinations of the candidate Laplacians.

\[
L_{MultiGrNMF} = \sum_{k=1}^{K} \tau_k L_k
\]

subject to,

\[
\sum_{k=1}^{K} \tau_k = 1, \quad \tau_k \geq 0
\]

where \( \tau_k \) is the mixture weight of the \( k \)-th graph Laplacian [8]. By substituting (3) and (1) in (2) we can obtain the loose function of MultiGrNMF

\[
O_{MultiGrNMF}(H, W, \tau)
= \|X - HW\|^2 + \alpha \sum_{k=1}^{K} \tau_k \text{Tr}(WL_k W^T) + \beta \|\tau\|^2
= \text{Tr}(X^T X) - 2\text{Tr}(X^T HW) + \text{Tr}(W^T H^T HW) + \alpha \sum_{k=1}^{K} \tau_k \text{Tr}(WL_k W^T) + \beta \|\tau\|^2.
\]

The \( L_2 \) norm regularization term is used to avoid parameter \( \tau \) over fitting to a single graph, \( \alpha \) and \( \beta \) are trade off parameters that balance the three terms. Inputs to MultiGrNMF algorithm are the data matrix \( X \), \( K \) Laplacian graphs, the initial guess of \( \tau \), \( W \), \( H \), and values of \( \alpha \) and \( \beta \) parameters. Refer to [8] for more details about MultiGrNMF algorithm.

3. EXPERIMENT

In this empirical evaluation, we compared the performance of MultiGrNMF with the original NMF [3], GrNMF [5] and GrNMF* (which uses the best graph selected by MultiGrNMF) for face image classification using the nearest neighbor classifier. Our experiments were performed on two image databases: the ORL database [11] and the Yale database [12].

3.1 Experiment Setup for ORL Data Set

We compared the accuracy obtained by applying MultiGrNMF, GrNMF*, GrNMF, and NMF to a widely used ORL face image database that has 40 individuals [11]. There are ten different face images resized into 32x32 for each of the individuals. Thus each image is represented by a 1024-dimentional vector in the image space, and the image set is organized as a data matrix \( X \in \mathbb{R}^{40 \times 1024} \).

We used the following graph types to construct multiple graphs for MultiGrNMF: 0-1 weighted graph and heat kernel weighted graph. By varying neighborhood size parameters between 1 and 9 for all graph types and the bandwidth parameter \( \sigma \) for heat kernel weighted graph between 0.1 and 15, we obtained a total of 195 graphs for this experiment.
Moreover, a 15-fold cross-validation was employed to test the performance of all algorithms and the mean accuracy is reported.

For all the experiments $\alpha=1$, $\tau_\alpha$ was initialized by setting all of its elements to $I/K$ as to base graphs with equal weights; and $\beta$ was initialized to the first positive integer number that avoided over fitting to a single graph.

### 3.2 Experiment Setup for Yale Data Set

We compared the accuracy obtained by applying MultiGrNMF, GrNMF*, GrNMF, and NMF to the Yale Face Database [12] that contains 165 grayscale images of 15 individuals. There are 11 images per subject (all resized into 32x32), one per different facial expression or configuration. Thus each image is represented by a 1024-dimensional vector in image space, and the image set is organized as a data matrix $X \in \mathbb{R}^{1204 \times 165}$.

We used the following graph types to construct multiple graphs for MultiGrNMF: 0-1 weighted graph and heat kernel weighted graph. By varying neighborhood size parameters between 1 and 10 for all graph types and the bandwidth parameter $\sigma$ for heat kernel weighted graph between 0.1 and 15, we obtained a total of 193 graphs for this experiment. Moreover, a 15-fold cross-validation is employed to test the performance of all algorithms and the mean accuracy is reported.

For all the experiments $\alpha=1$, $\tau_\alpha$ was initialized by setting all of its elements $I/K$ as to base graphs with equal weights; and $\beta$ was initialized to the first positive integer number that avoided over fitting to a single graph.

### 3.3 Results

For GrNMF, we learned coding vectors $W$ and biases vectors $H$ for each graph. The reported results in this paper are the mean accuracy of all graphs.

We first tested the classification performance of MultiGrNMF, GrNMF*, GrNMF, and NMF against the number of basis vectors $R$. Results are shown in Figure 1, classification accuracies of MultiGrNMF, GrNMF, and NMF in general increased when more basis vectors were used. However, there are some decreases for some $R$ values. This figure shows that MultiGrNMF coding outperforms the other three algorithms for most $R$ values.

![Figure 1](http://proceedings.spiedigitallibrary.org/)

Figure 1. Classification accuracies of MultiGrNMF, GrNMF*, GrNMF, and NMF versus parameter $R$ (a) on ORL database and (b) on Yale database.

In Figure 2 we report three samples of affinity matrix used by GrNMF and the learned one by MultiGrNMF. Clearly the learned one are better because it avoids a lot of noise in single graphs. In Figure 3 the $K$ graph weights learned by MultiGrNMF are shown for both data sets. Moreover in Figure 4 the 15-fold cross validation for nearest neighbor classifier results are shown for both data sets.
Figure 2. Affinity matrix, $A$, learned by MultiGrNMF. First row is for the ORL database and the second row is for the Yale database. Columns (a,b,c) represent samples for affinity matrices used as inputs for MultiGrNMF and (d) represents the learned affinity matrix by MultiGrNMF.

Figure 3. Learned graph weight vector $\tau$ using MultiGrNMF for (a) ORL database. (b) Yale database.

Figure 4. Accuracy for the 15-fold cross-validation of nearest neighbor classifier of MultiGrNMF, GrNMF*, GrNMF, and NMF for (a) ORL database with $R=40$ (b) Yale database $R=35$.

The coding vectors for both GrNMF and MultiGrNMF for Yale data set are shown in Figure 5. In Figure 6 the basis vectors for GrNMF, GrNMF*, and MultiGrNMF are shown for Yale data set. The obtained basis are sparser for MultiGrNMF which indicates better data representation. For ORL data set MultiGrNMF basis vectors are as sparse as most GrNMF basis vectors and little denser than some of GrNMF resulted basis vectors. For ORL data set there is no big difference between basis vector for GrNMF* and MultiGrNMF.
Figure 5. Coding vectors $W$ learned from Yale database (a) $W$ learned by GrNMF. (b) $W$ learned by MultiGrNMF.

Figure 6. Learned basis vectors (column vectors of $H$). The first row is for ORL database and the second row is for Yale database. Column (a) Basis vectors learned by GrNMF, (b) Basis vectors learned by GrNMF*, and (c) Basis vectors learned by MultiGrNMF.

4. CONCLUSION

This paper introduces the MultiGrNMF in the context of face image recognition. The performance of MultiGrNMF was tested on two face image data bases. Obtained results support the idea that using multiple graphs performs better than using single graph for most cases.

REFERENCES


