Neurocomputing 173 (2016) 270–277

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Orthogonal Projective Sparse Coding for image representation

Wei Zhao [a], Zheng Liu [a], Ziyu Guan [b,*], Binbin Lin [c], Deng Cai [c]

[a] National Laboratory of Radar Signal Processing, Xidian University, Xi'an, Shaanxi 710071, China
[b] College of Information and Technology, Northwest University of China, Xi'an, Shaanxi 710127, China
[c] The State Key Lab of CAD & CG, Zhejiang University, No.388 Yu Hang Tang Road, Hangzhou 310058, China

ARTICLE INFO

Article history:
Received 15 July 2014
Received in revised form
30 September 2014
Accepted 29 October 2014
Available online 1 September 2015

Keywords:
Manifold
Sparse coding
Orthogonal projection

ABSTRACT

We consider the problem of image representation for visual analysis. When representing images as vectors, the feature space is of very high dimensionality, which makes it difficult for applying statistical techniques for visual analysis. One then hope to apply matrix factorization techniques, such as Singular Vector Decomposition (SVD) to learn the low dimensional hidden concept space. Among various matrix factorization techniques, sparse coding receives considerable interests in recent years because its sparse representation leads to an elegant interpretation. However, most of the existing sparse coding algorithms are computational expensive since they compute the basis vectors and the representations iteratively. In this paper, we propose a novel method, called *Orthogonal Projective Sparse Coding* (OPSC), for efficient and effective image representation and analysis. Integrating the techniques from manifold learning and sparse coding, OPSC provides a sparse representation which can capture the intrinsic geometric structure of the image space. Extensive experimental results on real world applications demonstrate the effectiveness and efficiency of the proposed approach.

1. Introduction

Image representation is the fundamental problem in image processing. Researchers have long sought effective and efficient representations of images. For a given image database, we may have thousands of distinct features. However, the degree of the freedom of each image could be far less. Instead of the original feature space, one might hope to find a hidden semantic "concept" space to represent the images. The dimensionality of this "concept" space will be much smaller than the feature space. To achieve this goal, matrix factorization based approaches have attracted considerable attention in the last decades [1,6,9,15].

Given an image data matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$, each column of $\mathbf{X}$ corresponding to an m-dimensional image vector, the matrix factorization methods find two matrices $\mathbf{U} \in \mathbb{R}^{m \times k}$ and $\mathbf{A} \in \mathbb{R}^{k \times n}$ whose product can well approximate $\mathbf{X}$. Each column vector of $\mathbf{U}$ can be regarded as a basis vector corresponding to a certain semantic concept and each column vector of $\mathbf{A}$ is the representation of an image in this k-dimensional concept space.

One of the most well known matrix factorization methods is Singular Value Decomposition (SVD) [13], which serves as the basis of Principle Component Analysis (PCA) [11] and Latent Semantic Analysis (LSA) [9]. SVD is optimal in the sense of reconstruction error and thus optimal for data representation when Euclidean structure is concerned. Another popular matrix factorization method is Non-negative Matrix Factorization (NMF) [15], which requires the factorization matrices (both $\mathbf{U}$ and $\mathbf{A}$) are non-negative. The non-negative constraints allow only additive combinations among different basis vectors and it is believed that NMF can learn a *parts-based* representation [15]. Inspired by biological visual systems, people has been arguing that sparse features of data points are useful for learning [10,22]. Sparse Coding (SC) [17,23] is recently a popular matrix factorization method which requires the representation matrix $\mathbf{A}$ to be sparse. The sparseness of $\mathbf{A}$ indicates that each sample will only relate to several concepts (with non-zero coefficients to the corresponding basis vectors).

All the above three matrix factorization methods only consider the Euclidean structure of the image space. Recent studies have shown that human generated image data are probably sampled from a submanifold of the ambient Euclidean space [2,24,25]. In fact, the image data cannot possibly "fill up" the high dimensional Euclidean space uniformly. Therefore, the intrinsic manifold structure needs to be considered while performing the matrix factorization. Cai et al. [6] extend the traditional NMF to Graph regularized NMF (GNMF). By incorporating a geometric regularizer [3] on the representation matrix $\mathbf{A}$, GNMF is able to exploring the intrinsic manifold structure of the data. The similar idea has also been applied on sparse coding which leads to GraphSC [29]. However, it is not clear for these approaches

* Corresponding author.
E-mail addresses: ywzhao@mail.xidian.edu.cn (W. Zhao), lz@xidian.edu.cn (Z. Liu), ziyuguan@nwu.edu.cn (Z. Guan), binbinlinzju@gmail.com (B. Lin), dengcai@cad.zju.edu.cn (D. Cai).

http://dx.doi.org/10.1016/j.neucom.2014.10.106
0925-2312/

that how to *efficiently* find the low dimensional representation of a new sample (out-of-sample extension).

The optimization problems in both NMF (GNMF) and SC (GraphSC) are non-convex. The typical algorithms [6,16,17,29] compute the basis matrix $\mathbf{U}$ and the representations $\mathbf{A}$ iteratively. Thus, these algorithms are computational expensive. Motivated from recent progress on orthogonal projective analysis [18], in this paper we propose a novel matrix factorization method, called *Orthogonal Projective Sparse Coding* (OPSC), for efficient image representation. OPSC is a two-step approach including basis learning and sparse representation learning. In the first step, OPSC learns the basis by exploiting the intrinsic geometric structure of the data. By requiring the gradient field as orthogonal as possible to the tangent spaces of the data, OPSC encodes the semantic structure in the basis vectors. In the second step, OPSC uses the LASSO [14] to learn a sparse representation with respect to the learned basis for each image.

The rest of the paper is organized as follows: in Section 2, we provide a brief review of matrix factorization. Our Orthogonal Projective Sparse Coding (OPSC) method is introduced in Section 3. The experimental results are presented in Section 4. Finally, we provide the concluding remarks in Section 5.

2. Background

Given a data set with high dimensionality, matrix factorization is a common approach to "compress" the data by finding a set of *basis* vectors and the *representation* with respect to the basis for each data point. Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{m\times n}$ be the data matrix, matrix factorization can be mathematically defined as finding two matrices $\mathbf{U} \in \mathbb{R}^{m\times k}$ and $\mathbf{A} \in \mathbb{R}^{k\times n}$ whose product can best approximate $\mathbf{X}$:

$$\mathbf{X} \approx \mathbf{UA}.$$

Each column of $\mathbf{U}$ can be regarded as a basis vector which captures the higher-level features in the data and each column of $\mathbf{A}$ is the k-dimensional representation of the original inputs with respect to the new basis. From this sense, matrix factorization can also be regarded as a dimensionality reduction method since it reduces the dimension from m to k.

A common way to measure the approximation is by Frobenius norm of a matrix $\|\cdot\|$. Thus, the matrix factorization can be defined as the optimization problem as follows:

$$\min_{\mathbf{U},\mathbf{A}} \ \| \mathbf{X} - \mathbf{UA} \|^2 \tag{1}$$

Various matrix factorization algorithms add different constraints on the above optimization problem based on different goals.

Singular Value Decomposition (SVD) is one of the most popular matrix factorization algorithms [13], which requires $\mathbf{U}^T\mathbf{U} = \mathbf{I}$. Suppose the rank of $\mathbf{X}$ is r, the SVD decomposition of $\mathbf{X}$ is as follows:

$$\mathbf{X} = \mathbf{U}\Sigma\mathbf{V}^T, \tag{2}$$

where $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_r)$ and $\sigma_1 \ge \sigma_2 \ge \cdots \ge \sigma_r > 0$ are the singular values of $\mathbf{X}$, the columns of $\mathbf{U} \in \mathbb{R}^{m\times r}$ are called left singular vectors and the columns of $\mathbf{V} \in \mathbb{R}^{n\times r}$ are called right singular vectors. It can be proven that the first k columns of $\mathbf{U}$ and the first k rows of $\Sigma\mathbf{V}^T$ are the optimal solution of the optimization problem (1) [13].

Another popular matrix factorization algorithm is Non-negative Matrix Factorization (NMF) [15], which focuses on the analysis of data matrices whose elements are non-negative. NMF adds the non-negative constraint on both $\mathbf{U}$ and $\mathbf{A}$ in the optimization problem (1). The non-negative constraints allow only additive combinations among different basis vectors. Thus, it is believed that NMF can learn a *parts-based* representation [15].

The representation matrix $\mathbf{A}$ learned in the above two methods is usually dense. Since each basis vector (column vector of $\mathbf{U}$) can be regarded as a concept, the denseness of $\mathbf{A}$ indicates that each image is a combination of *all* the concepts. This is contrary to our common knowledge since most of the images only include several semantic concepts. Sparse Coding (SC) [10,23] is a recently popular matrix factorization method trying to solve this issue. SC adds the sparse constraint on $\mathbf{A}$, more specifically, on each column of $\mathbf{A}$. In this way, SC can learn a sparse representation. SC has several advantages for data representation. First, it yields sparse representations such that each data point is represented as a linear combination of *a small number* of basis vectors. Thus, the data points can be interpreted in a more elegant way. Second, sparse representations naturally make for an indexing scheme that would allow quick retrieval. Third, the sparse representation can be over-complete, which offers a wide range of generating elements. Potentially, the wide range allows more flexibility in signal representation and more effectiveness at tasks like signal extraction and data compression. Finally, there is considerable evidence that biological vision adopts sparse representations in early visual areas [22]. The sparse coding approach is fundamentally different from those sparse subspace learning methods, *e.g.*, Sparse PCA [31], Sparse LDA [21] and Sparse LPP [4,5]. Instead of learning a sparse $\mathbf{A}$, the sparse subspace learning methods [4,5,21,31] learn a sparse $\mathbf{U}$. The low dimensional representation matrix $\mathbf{A}$ learned by these sparse subspace learning methods is still dense.

All the above three matrix factorization methods only consider the Euclidean structure of the image space. Recent studies have shown that human generated image data are probably sampled from a submanifold of the ambient Euclidean space [2,24,25]. In fact, the image data cannot possibly "fill up" the high dimensional Euclidean space uniformly. Therefore, the intrinsic manifold structure needs to be considered while performing the matrix factorization. A natural assumption here could be that if two data points $\mathbf{x}_i$, $\mathbf{x}_j$ are *close* in the *intrinsic* geometry of the data distribution, then $\mathbf{a}_i$ and $\mathbf{a}_j$, the representations of this two points with respect to the new basis, are also close to each other. This assumption is usually referred to as *manifold assumption* [3,7], which plays an essential role in the development of various kinds of algorithms including dimensionality reduction algorithms [2] and semi-supervised learning algorithms [3,30]. Recent studies in spectral graph theory [8] and manifold learning theory [2] have demonstrated that the local geometric structure can be effectively modeled through a nearest neighbor graph on a scatter of data points. Consider a graph with n vertices where each vertex corresponds to an image in the data set. The edge weight matrix $\mathbf{W}$ is usually defined as follows:

$$\mathbf{W}_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_i \in N_p(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_p(\mathbf{x}_s) \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

where $N_p(\mathbf{x}_i)$ denotes the set of p nearest neighbors of $\mathbf{x}_i$. Define a diagonal matrix $\mathbf{D}$ whose entries are column (or row, since $\mathbf{W}$ is symmetric) sums of $\mathbf{W}$, $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$, we can compute the graph Laplacian $\mathbf{L} = \mathbf{D} - \mathbf{W}$ [8].

The *manifold assumption* can then be mathematically formulated as minimizing the following geometric regularizer [3,6,29]:

$$\mathcal{R} = \mathrm{Tr}(\mathbf{A}^T\mathbf{L}\mathbf{A}), \tag{4}$$

where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. Adding this regularizer in Eq. (1), we get the objective function of manifold regularized matrix factorization [6,29]:

$$\min_{\mathbf{U},\mathbf{A}} \ \| \mathbf{X} - \mathbf{UA} \|^2 + \lambda\, \mathrm{Tr}(\mathbf{A}^T\mathbf{L}\mathbf{A}). \tag{5}$$