SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

October 2015, Vol. 58 102302:1–102302:15
doi: 10.1007/s11432-014-5258-6

Learning block-structured incoherent dictionaries for sparse representation

ZHANG YongQin[1], XIAO JinSheng[2*], LI ShuHong[3], SHI CaiYun[4] & XIE GuoXi[4*]

[1] *School of Information Science and Technology, Northwest University, Xi'an* 710127*, China;*
[2] *School of Electronic Information, Wuhan University, Wuhan* 430079*, China;*
[3] *College of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou* 450002*, China;*
[4] *Shenzhen Key Laboratory for MRI, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen* 518055*, China*

Received June 1, 2014; accepted September 9, 2014; published online May 14, 2015

Abstract Dictionary learning is still a challenging problem in signal and image processing. In this paper, we propose an efficient block-structured incoherent dictionary learning algorithm for sparse representations of image signals. The constrained minimization of dictionary learning is achieved by iteratively alternating between sparse coding and dictionary update. Without relying on any prior knowledge of the group structure for the input data, we develop a two-stage clustering method that identifies the underlying block structure of the dictionary under certain restricted constraints. The two-stage clustering method mainly consists of affinity propagation and agglomerative hierarchical clustering. To meet the conditions of both the upper bound and the lower bound of the mutual coherence of dictionary atoms, we introduce a regularization term for the objective function to adjust the block coherence of the overcomplete dictionary. The experiments on synthetic data and real images demonstrate that the proposed dictionary learning algorithm has lower representation error, higher visual quality and better reconstructed results than most of the state-of-the-art methods.

Keywords dictionary learning, sparse representation, sparse coding, block sparsity, mutual coherence

Citation Zhang Y Q, Xiao J S, Li S H, et al. Learning block-structured incoherent dictionaries for sparse representation. Sci China Inf Sci, 2015, 58: 102302(15), doi: 10.1007/s11432-014-5258-6

1 Introduction

Sparse signal representations over redundant dictionaries have attracted great attention and interest in the fields of signal processing and computer vision [1,2]. The sparse coding for representing natural images resembles the receptive fields of simple cells in the mammalian visual cortex [3–5]. This provides the physiological basis for sparse representation and its application to image processing [6]. For sparse representation of signals, there are generally two methods: the synthesis model and the analysis model [7]. The analysis model being out of the scope of this paper, we will focus on the research of the synthesis model in this paper. In the past decade, the synthesis model has been a very popular approach for sparse representations [8,9]. Such a model assumes that a signal $\boldsymbol{X} \in \mathbb{R}^d$ can be composed of a linear combination of a few atoms chosen from a given dictionary $\boldsymbol{D}^{d\times K}$, satisfying $\|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_p \leqslant \varepsilon$, where ε

* Corresponding author (email: xiaojs@whu.edu.cn, gx.xie@siat.ac.cn)

is an arbitrarily small positive number. The vector $\boldsymbol{A} \in \mathbb{R}^K$ denotes the sparse coefficients of the signal $\boldsymbol{X}$. Indeed, this general problem has been proven to be NP-hard. So far, the development of many sub-optimal solutions for the above model has estimated the sparse representation $\boldsymbol{A}$ from a corrupted signal $\boldsymbol{X}$ (i.e., sparse coding) by inferring the dictionary $\boldsymbol{D}$ from signal examples (i.e., dictionary learning).

Sparse coding has been extensively studied in recent years. The literature on sparse recovery can be broadly categorized into two basic approaches. One of them is greedy algorithm that iteratively selects locally optimal basis vectors. This includes Matching Pursuit (MP) [10], Orthogonal Matching Pursuit (OMP) [11,12], Block Orthogonal Matching Pursuit (BOMP) [13,14], Stage-wise Orthogonal Matching Pursuit (StOMP) [15,16], CoSaMP [17], Least-Angle Regression (LARS) [18], Subspace Pursuit (SP) [19] and Gradient Pursuit(GP) [20]. The other approach is convex optimization methods, such as Basis Pursuit (BP) [21], Block Basis Pursuit (BBP) [22], FOCUSS [23], interior point methods [24], and shrinkage-based algorithms [25,26]. Currently, adaptive dictionary learning is an efficient technique for signal modeling.

The choice of dictionary of the synthesis model has been extensively studied in the recent literature. These literature are generally divided into two groups: the analytical approaches and the learning-based approaches. In the first group, natural signals and images have an essentially sparse representation in analytical transform domains. The dictionaries of this type are highly structured and efficient, which include discrete cosine transform (DCT) [27], Wavelets [28], Curvelets [29], Contourlets [30], Ridgelets [31], Bandelets [32] and Shearlets [33]. However, although such analytical dictionaries provide fast transforms, they have a fixed data representation and limited ability to adapt to different types of data. The second group uses machine learning techniques to optimize the sparsity of the representation for a large set of examples. These learning-based dictionary methods can be further divided into three categories: the probabilistic learning, the clustering-based learning and learning dictionaries with specific structures [34]. In the first category, Olshausen and Field [35] developed a maximum likelihood (ML) method consisting of two-step optimization structure: the sparse approximation and the dictionary learning. They also introduced two main assumptions: the independent sparse coefficients and the additive zero-mean Gaussian noise. However, this iterative algorithm may converge to a local minimum. Subsequently, the method of optimal directions (MOD) [36] uses the OMP algorithm [11,12] to find sparse representations and introduces a closed-form solution for the dictionary update. The MOD method is faster, but its convergence cannot be guaranteed. The maximum a posteriori (MAP) method [37] uses an additional constraint like the unit Frobenius norm for the dictionary and applies FOCUSS [23] for sparse approximation. The majorization method [38] adopts an iterative thresholding technique for the sparse approximation and the different constraints for the dictionary update. To avoid expensive computations, the online learning algorithm [39] is performed with a subset of the training data and can also be applied to dynamic systems. The task-driven method [40] provides a supervised general formulation for learning dictionaries adapted to a wide variety of tasks instead of dictionaries only adapted to data reconstruction. In the second category, the vector quantization (VQ) approach [41] for dictionary learning is achieved by K-means clustering in MP-based video coding. As an extension of the K-means method for dictionary learning, the K-SVD algorithm [42] adopts a singular value decomposition (SVD) [43–45] to update each column of the dictionary sequentially after the sparse coding with the OMP method [11,12]. To optimize the K-SVD method [42], the BK-SVD+SAC algorithm [46] uses the agglomerative clustering to exploit the block structure of the dictionary. Eksioglu [47] also proposed a general framework for the block structure identification to optimize the dictionary design. However, these algorithms are not guaranteed to converge in the finite time and tend to generate visual artifacts. In the third category, the parametric dictionaries can be built on the atom parameters and prior knowledge about the signal formation or the target task. Since they are usually structured, some desired constraints can be enforced for the dictionary learning, such as feature-aware regularization [48], minimal dictionary coherence [38,49], the orthogonality between subspaces [50,51], block-based structures [52,53], multi-scale characteristics of the atoms [54,55], shift and rotation invariance properties [56], global sparsity constraint [57] and nonlocal tensor decomposition [58]. These prior constraints are beneficial for designing a learning strategy and its approximate performance. However, each of these learning methods mentioned above has its own drawbacks, e.g., the first two categories lack consideration on the priors about the dictionary, whereas the third category is only concerned with some general constraints.

It is found that the K-SVD method [42] has been widely used in many applications. The group sparse coding [59,60] and block-sparse signals [14,61] have become the focus of ongoing research in the fields of sparse representations. Recently, the optimized K-SVD method [46] has appeared, which adopts the sparse agglomerative clustering (SAC) approach to identify the block structure of atoms in the dictionary. However, the major drawbacks of the SAC approach are its susceptibility to errors in the initial steps that propagate all the way to its final output. Although the literature [49] pointed out the constraints of both the upper bound [11] and the lower bound [62] of the mutual coherence of dictionary atoms, the coherence of the dictionary trained by the K-SVD method [42] or its variant [46,47] has received little attention in the reported literatures. In this paper, we focus on the dictionary learning of the synthesis model and propose a block-structured incoherent K-SVD (BSIK-SVD) algorithm that would identify and extract inherent structures of the dictionary and regulate intra-block coherence. To learn a universal adaptive over-complete dictionary for sparse representation of image signals, we need to explore the priors of image signals before designing an efficient sparse signal model. Different from the current methods, our proposed dictionary learning algorithm utilizes the prior knowledge of both block sparsity and mutual coherence of dictionary. The main contributions of our work can be summarized as follows: 1) the two-stage clustering technique, mainly consisting of affinity propagation and agglomerative hierarchical clustering method, is employed to extract the inherent block structures of the learned dictionary, and 2) the proposed BSIK-SVD framework incorporates a regularization constraint by adjusting the intra-block coherence of dictionary atoms. The purpose of this work is to find an efficient block-structured incoherent dictionary for sparse represents of signals. The advantages of this optimized dictionary include compact representation, the rapid convergence, and robust stability under the noise.

The rest of this paper is organized as follows. The previous work is reviewed in Subsection 2.1. Subsections 2.2–2.4 provide the descriptions of the proposed algorithm in detail. The experimental results and analysis are given in Section 3. The conclusions and future work are elaborated in Section 4.

2 Dictionary optimization

In this section, the previous work involved with the basic concepts is first described that justify the purpose of our work. We then formulate the problem of the block-structured incoherent dictionary design, and propose a dictionary learning algorithm. As a generalization of the K-SVD method, this proposed algorithm incorporates the block structure identification and the intra-block coherence adjustment for block sparse representations of the image signals.

2.1 Previous work

Many problems in signal and image processing can be cast as inverting the linear system:

$$\boldsymbol{X} = \boldsymbol{D}\boldsymbol{A} + \upsilon, \tag{1}$$

where $\boldsymbol{X} \in \mathbb{R}^{d\times L}$ is the vectors of noisy observations (or measurements), $\boldsymbol{D} \in \mathbb{R}^{d\times K}$ with $d < K$ is a bounded linear operator, $\boldsymbol{A} \in \mathbb{R}^{K\times L}$ is the data to recover, and υ is an additive noise with bounded variance.

For this observation model (1) of natural images, in order to recover $\boldsymbol{A}$ from the measurement $\boldsymbol{X}$, the problem of dictionary learning is formulated as:

$$\arg\min_{\boldsymbol{D},\boldsymbol{A}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_0 \leqslant \kappa, \quad i = 1,\ldots,L, \tag{2}$$

where $\boldsymbol{A} \in \mathbb{R}^{K\times L}$ contains all the sparse coefficients, $\boldsymbol{\alpha}_i$ is a column of the sparse matrix $\boldsymbol{A}$, $\boldsymbol{D} \in \mathbb{R}^{d\times K}$ with $d < K$ is the overcomplete dictionary, $\boldsymbol{X} \in \mathbb{R}^{d\times L}$ is a matrix containing L given training examples, and κ is the known sparsity level that is fixed with a predetermined number of nonzero entries. Note that each column of $\boldsymbol{D}$ is normalized so that its l_2 norm is 1.

Since no general analytic solution can be found for this challenging optimization problem (2), the numerical strategy is commonly employed in practice. The K-SVD method [42] uses a relaxation technique to provide an approximate solution to Problem (2) by taking the l_1 norm instead of the l_0 norm. However, the performance of the K-SVD method [42] strongly depends on the initial dictionary, and its convergence to the global optimum cannot be guaranteed. Subsequently, the block sparse model [14,22,63] is developed based on the assumption of the block structure from the dictionary. Furthermore, the optimized K-SVD method [46,47] uses the clustering approach to identify the block structure of atoms in the dictionary, and then adopts the BOMP [14] for the sparse coding. For the block-sparse model, the optimization problem for (1) can be reformulated in another form as follows:

$$\arg\min_{\boldsymbol{D},\boldsymbol{b},\boldsymbol{A}} \|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{A}\|_F^2, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}} \leqslant \kappa,\ i=1,\ldots,L,\ \ |b_j| \leqslant s,\ j\in[1,B], \tag{3}$$

where $\boldsymbol{b}$ is a block structure with maximal block size s, $b_j=\{i\in 1,\ldots,K\,|\,b[i]=j\}$ is the set of indices for the block j, and B is the number of the blocks. In most previous works, the block structure is assumed to be known by prior information, or directly obtained by a certain clustering method. However, the underlying block structure contained in the data is unknown in most cases. To solve this problem, we need to extract the underlying block structure automatically and efficiently from the data. In fact, since the two different atoms located in different sub-blocks have low coherence level, the maximum coherence between two different atoms only appears in the intra-block. Therefore, we focus on the minimization of the intra-block coherence of the dictionary.

2.2 Problem statement

For a given set of the signals $\boldsymbol{X}\in\mathbb{R}^{d\times L}$, our goal is to find a dictionary $\boldsymbol{D}\in\mathbb{R}^{d\times K}$ whose atoms are block-structured and incoherent, ensuring that the signals can be accurately reconstructed through a computationally efficient algorithm. Then we formulate the following optimization problem:

$$\arg\min_{\boldsymbol{D},\boldsymbol{b},\boldsymbol{A}} \left\{\|\boldsymbol{X}-\boldsymbol{D}\boldsymbol{A}\|_F^2+\lambda\Phi(\boldsymbol{D})\right\}, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}} \leqslant \kappa, \forall i,\ \ |b_j| \leqslant s,\ j\in[1,B], \tag{4}$$

where λ is a balance parameter and $\Phi(\boldsymbol{D})$ is the regularization term on the bounded coherence of the learned dictionary $\boldsymbol{D}$, which is defined as

$$\Phi(\boldsymbol{D})=\sum_{j=1}^{B}\left(\sum_{p,q\in b_j,p\neq q}\left\|\boldsymbol{\varphi}_p^{\mathrm{T}}\boldsymbol{\varphi}_q\right\|^2\right), \tag{5}$$

where B is the total number of sub-blocks, T is the transpose operator, $\boldsymbol{\varphi}_p$ and $\boldsymbol{\varphi}_q$ denote the different atoms in the sub-block b_j of the current dictionary, respectively.

Let $\boldsymbol{A}_j$ denote the jth block of sparse representation $\boldsymbol{A}$ that corresponds to the jth block of the dictionary $\boldsymbol{D}$. Figure 1 illustrates the proposed block-structured and incoherent dictionary learning problem. The proposed BSIK-SVD framework extends BK-SVD+SAC [46] and K-SVD [42], and incorporates an additional constraint on the mutual coherence suppression of the dictionary. Furthermore, an efficient cluster-based approach is proposed to distinguish the underlying block structure existed in the dictionary.

2.3 Method preview

In this section, we develop a dictionary learning framework that employs the block coordinate relaxation method for solving the optimization problem (4). The collection of K random vectors is used to initialize the dictionary whose atoms are normalized to unit l_2 norm. The proposed BSIK-SVD algorithm starts from an initial dictionary and iteratively alternates between a local sparse coding step and a dictionary update step. At each iteration t, the two successive steps are performed as follows:

(1) Sparse coding: calculate the sparse approximation coefficients $\boldsymbol{A}$ and extract inherent structures $\boldsymbol{b}$ while keeping the dictionary $\boldsymbol{D}^{(t-1)}$ fixed:

$$\left[\boldsymbol{b}^{(t)},\boldsymbol{A}^{(t)}\right]=\arg\min_{\boldsymbol{b},\boldsymbol{A}}\left\|\boldsymbol{X}-\boldsymbol{D}^{(t-1)}\boldsymbol{A}\right\|_F^2, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}} \leqslant \kappa, \forall i,\ \ |b_j| \leqslant s,\ j\in\boldsymbol{b}. \tag{6}$$

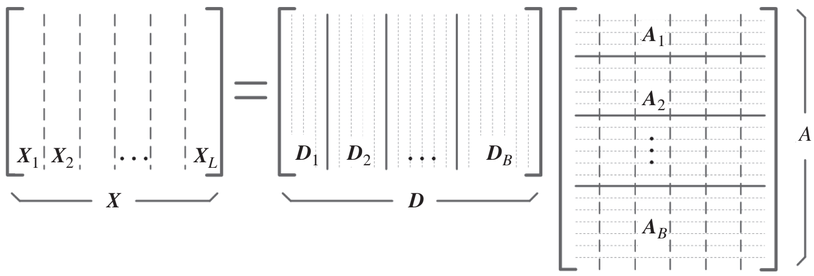

Figure 1 Illustration of the proposed block-structured and incoherent dictionary learning algorithm. A group of signals $\boldsymbol{X} = [\boldsymbol{X}_1, \dots, \boldsymbol{X}_L]$ can be represented by the block-sparse matrix $\boldsymbol{A} = \left[\boldsymbol{A}_1^{\mathrm{T}}, \dots, \boldsymbol{A}_B^{\mathrm{T}}\right]^{\mathrm{T}}$ with respect to the block-structured dictionary $\boldsymbol{D} = [\boldsymbol{D}_1, \dots, \boldsymbol{D}_B]$.

We propose an approximate solution to Problem (6), which adopts a two-stage clustering technique to identify the block structure. That is, the BOMP method [14] is first used to calculate sparse coefficients. Then the two-stage clustering technique is applied to these sparse coefficients to extract the underlying block structure that is repeatedly used for sparse coding.

(2) Dictionary update: adapt the dictionary $\boldsymbol{D}$ to the data $\boldsymbol{X}$ by solving Problem (4) for $\boldsymbol{D}$ and $\boldsymbol{A}$ and then suppress the mutual coherence between different atoms in the dictionary $\boldsymbol{D}$ while keeping the block structure $\boldsymbol{b}^{(t)}$ fixed:

$$\left[\boldsymbol{D}^{(t)}, \boldsymbol{A}^{(t)}\right] = \arg\min_{\boldsymbol{D},\boldsymbol{A}} \left\{ \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2 + \lambda\Phi\left(\boldsymbol{D}\right) \right\}, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}^{(t)}} \leqslant \kappa, \ \forall i. \tag{7}$$

In fact, the proposed BSIK-SVD method can be seen as a generalized case of K-SVD [42] or BK-SVD+SAC [46]. When the blocks of atoms in the dictionary $\boldsymbol{D}^{(t)}$ are sequentially updated, the corresponding non-zero blocks of coefficients in the sparse representation $\boldsymbol{A}^{(t)}$ are also updated simultaneously.

In the following sections, we give the detailed descriptions and the specific steps of the proposed BSIK-SVD algorithm. In brief, the overview of the procedures of our BSIK-SVD algorithm is summarized as follows (See Algorithm 1).

Algorithm 1 Block-structured incoherent dictionary learning algorithm

Input: A set of observed signals $\boldsymbol{X}$, block sparsity κ and maximal block size s.
Output: A learned dictionary $\boldsymbol{D}$, block structure $\boldsymbol{b}$ and the sparse representation $\boldsymbol{A}$.

- Initialization: set the initial dictionary $\boldsymbol{D}^{(0)}$ with unit l_2 norm columns.
- Repeat from $t = 1$ until convergence:
  1. Fix the dictionary $\boldsymbol{D}^{(t-1)}$, and compute the sparse representation $\boldsymbol{A}^{(t-1)}$;
  2. Extract the block structure $\boldsymbol{b}^{(t)}$ by applying the two-stage clustering technique to the sparse coefficients $\boldsymbol{A}^{(t-1)}$, and then update the sparse representation $\boldsymbol{A}^{(t)}$ again;
  3. Fix the block structure $\boldsymbol{b}^{(t)}$, update the dictionary $\boldsymbol{D}^{(t)}$ and the sparse representation $\boldsymbol{A}^{(t)}$ by applying block K-SVD;
  4. Regulate the intra-block coherence of $\boldsymbol{D}^{(t)}$;
  5. $t = t + 1$.

2.4 BSIK-SVD algorithm

For input signals $\boldsymbol{X}$, we propose the BSIK-SVD algorithm to infer the dictionary $\boldsymbol{D}$, the block structure $\boldsymbol{b}$ and the sparse representation $\boldsymbol{A}$ by solving Problem (4). To solve this problem, we employ the coordinate relaxation technique to minimize the objective function based on alternating $\boldsymbol{A}$ and $\boldsymbol{D}$. Assume that the block structure $\boldsymbol{b}$ is initialized as K blocks of size 1, i.e., $\boldsymbol{b}^{(0)} = [1, \dots, K]$ and the dictionary $\boldsymbol{D}$ is initialized as the Gaussian random matrix $\boldsymbol{D}^{(0)}$ of size $d \times K$ such that the l_2 norm of each column equals to 1. The signals $\boldsymbol{X}$ are also supposed as the combinations of κ blocks of atoms with a maximum size constraint s. For the given $\boldsymbol{D}^{(0)}$ and $\boldsymbol{b}^{(0)}$, the sparse representation $\boldsymbol{A}^{(0)}$ is initialized as the solution to

Problem (6) over $\boldsymbol{A}$, which is solved by using the OMP method with $\kappa \times s$ instead of κ non-zero entries. At each iteration t, the dictionary $\boldsymbol{D}^{(t-1)}$ is first fixed so that Problem (4) is reduced to Problem (6). Then the BOMP method is used to calculate the sparse representation $\boldsymbol{A}^{(t-1)}$ by solving Problem (6).

Next, the two-stage clustering technique identifies the intrinsic block structure $\boldsymbol{b}^{(t)}$ of the block sparse representation $\boldsymbol{A}^{(t-1)}$, and then the BOMP method is used to re-update $\boldsymbol{A}^{(t)}$. Apparently, we would solve Problem (6) to update $\boldsymbol{b}$ while keeping $\boldsymbol{A}^{(t-1)}$ fixed. Therefore, Problem (6) finds a block structure $\boldsymbol{b}^{(t)}$ with maximal block size s satisfying the block-sparsity constraint of $\boldsymbol{A}^{(t-1)}$. To update the block structure $\boldsymbol{b}^{(t)}$, we minimize the cost function of the block sparse representation $\boldsymbol{A}^{(t-1)}$ mentioned by Zenlnik-Manor et al. [46] as follows:

$$\boldsymbol{b}^{(t)} = \min_{\boldsymbol{b}} \sum_{i=1}^{L} \left\| \boldsymbol{A}_i^{(t-1)} \right\|_{0,\boldsymbol{b}}, \ \text{s.t.}\ |b_j| \leqslant s,\ \forall j \in [1, B], \tag{8}$$

where B is the current number of blocks. Note that the non-zero patterns of rows in $\boldsymbol{A}^{(t-1)}$ may coincide with the corresponding columns of the same blocks in $\boldsymbol{D}^{(t-1)}$. That is, according to the block-sparse structure, classifying the atoms of the dictionary $\boldsymbol{D}^{(t-1)}$ is equivalent to classifying the rows of the sparse representation $\boldsymbol{A}^{(t-1)}$. Assume that $\omega_j\left(\boldsymbol{A}^{(t-1)}, \boldsymbol{b}\right)$ denotes the set of columns which contains the non-zero rows in the sub-block b_j of $\boldsymbol{A}^{(t-1)}$. Therefore, Problem (8) can be reformulated as follows:

$$\boldsymbol{b}^{(t)} = \min_{\boldsymbol{b}} \sum_{j \in [1,B]} \left| \omega_j \left(\boldsymbol{A}^{(t-1)}, \boldsymbol{b} \right) \right|, \ \text{s.t.}\ |b_j| \leqslant s,\ \ j \in [1, B], \tag{9}$$

where $|\omega_j|$ denotes the number of non-zero values in ω_j.

To solve Problem (9), the BK-SVD+SAC method [46] uses the agglomerative clustering algorithm [64] extracting the block structure of the dictionary. However, the agglomerative clustering algorithm [64] has the disadvantage that the blocks may have been incorrectly grouped at an early stage owing to lack of relocation provision. In fact the maximal block size may be larger than the predetermined fixed number s because it is possibly less than the number of the most similar atoms. To overcome these drawbacks, while considering the nonlocal similarity property of natural images [50], we propose a two-stage clustering approach to automatically extract the underlying block structure for dictionary optimization in this paper. The two-stage clustering approach is based on affinity propagation and agglomerative hierarchical clustering. The first stage yields an initial estimation of the block structure by affinity propagation clustering, which uses the city block distance criterion to build the similarity matrix. The similar blocks are grouped into a single block by maximizing the sum of responsibility and availability messages. The second stage will further merge the pair of blocks with the shortest distance repeatedly until all blocks are in one cluster. Since the initial block structure is estimated by affinity propagation in the first stage, the agglomerative clustering accuracy will be comparatively improved in the second stage so that the final clustering result is much better than the traditional agglomerative clustering algorithm [64].

Specifically, in the first stage, the affinity propagation (AP) clustering [65] considers all the vectors $\omega_j, j \in \boldsymbol{b}^{(0)}$ as the potential exemplars. It is carried out to produce a final set of exemplars and clusters by iteratively exchanging messages between the vectors until the convergence is achieved. Particularly, let $\Omega\left(\omega_j, \omega_{c_j}\right)$ be the similarity between the vectors ω_j and ω_{c_j}, which denotes the suitability of the vector ω_{c_j} to serve as the exemplar for the vector ω_j. For affinity propagation, if $j \neq c_j$, then our choice for similarity measure is the negative city block distance:

$$\Omega\left(\omega_j, \omega_{c_j}\right) = -\sum_{i=1}^{L} \left| \omega_j - \omega_{c_j} \right|. \tag{10}$$

Incorporating the similarity metric between the different items, the objective function of the affinity propagation algorithm is given by

$$\boldsymbol{c}^* = \arg\min_{\boldsymbol{c}} \left\{ -\sum_{j=1}^{B} \Omega\left(\omega_j, \omega_{c_j}\right) + \sum_{i=1}^{B} \delta_i\left(\boldsymbol{c}\right) \right\} \tag{11}$$

with

$$\delta_i(\boldsymbol{c}) = \begin{cases} \infty, & c_i \neq i \text{ and } \exists j : c_j = i; \\ 0, & \text{else}, \end{cases} \tag{12}$$

where $\boldsymbol{c} = [c_1, \ldots, c_B]$ and $c_j \in \{1, \ldots, B\}$ is the exemplar label that assigns each vector ω_j to its exemplar ω_{c_j}. The function $\delta_i(\boldsymbol{c})$ is an exemplar-consistency constraint ensuring the possibility of only valid configurations. The preference of each vector ω_j is also called the self-similarity $\Omega(\omega_j, \omega_j)$, which influences the number of the identified exemplars. For simplicity, the preference ρ_c is set to a global value in our algorithm. To solve the optimization problem (11), affinity propagation [65] takes the input similarities between data points and utilizes the max-product belief propagation algorithm to generate clusters insensitive to initialization after convergence [66]. There are two kinds of messages: responsibility $r(\omega_i, \omega_j)$ and availability $a(\omega_i, \omega_j)$. After setting their initial value to zero, responsibility and availability messages are iteratively updated, respectively. To avoid numerical oscillations, the message updates with the damping factor β_c are attenuated to guarantee the convergence of the algorithm. When affinity propagation converges, a vector ω_j becomes the exemplar with its self-responsibility plus self-availability being positive, i.e., $r(\omega_j, \omega_j) + a(\omega_j, \omega_j) > 0$. Then the non-exemplars are assigned to their respective exemplars and the valid configuration of labels $\boldsymbol{c} = [c_1, \ldots, c_B]$ for all the vectors $\omega_j, j \in \boldsymbol{b}^{(0)}$ is computed as

$$c_i = \max_{\omega_j \in \boldsymbol{c}} \{r(\omega_i, \omega_j) + a(\omega_i, \omega_j)\}. \tag{13}$$

When affinity propagation is finished, the updates of both the block structure $\boldsymbol{b}^{(t)}$ and the sparse representations are carried out at the end of the first stage. Therefore, the initial estimation $\tilde{\boldsymbol{b}}^{(t)}$ of the block structure is constructed and the clustering result of $\boldsymbol{A}^{(t-1)}$ is denoted by $\tilde{\boldsymbol{A}}^{(t-1)}$ with its vectors $\tilde{\omega}_j$.

Subsequently, in the second stage the estimated block structure in the first stage is used as the starting values. Like the traditional agglomerative clustering algorithm [46,64], at each loop we find the closest pair $[i^*, j^*]$ of blocks satisfying the following formula:

$$[i^*, j^*] = \arg\max_{i \neq j} |\tilde{\omega}_i \cap \tilde{\omega}_j|, \text{ s.t. } \left|\tilde{b}_i\right| + \left|\tilde{b}_j\right| \leqslant s. \tag{14}$$

Then we aggregate the closest pair of blocks i^* and j^* and update the block structure $\boldsymbol{b}^{(t)}$. Specifically, the blocks i^* and j^* are combined by enforcing $\forall j' \in b_j : i \to b[j']$, $\{\tilde{\omega}_i \cup \tilde{\omega}_j\} \to \tilde{\omega}_i$ and $\phi \to \tilde{\omega}_j$. This loop procedure is repeated until no blocks can be merged without breaking the constraint on the block size. After the two-stage clustering applied to the sparse representation $\boldsymbol{A}^{(t-1)}$, the block structure $\boldsymbol{b}^{(t)}$ of the dictionary is obtained. Figure 2 explains the procedures for our two-stage clustering approach.

According to the block structure $\boldsymbol{b}^{(t)}$ provided by the two-stage clustering process, we employ the BOMP method with κ non-zero entries to solve Problem (6) and then update the sparse representation $\boldsymbol{A}^{(t)}$ again. After the block structure $\boldsymbol{b}^{(t)}$ and the sparse representation $\boldsymbol{A}^{(t)}$ are available, we look for the solution of Problem (7) instead of Problem (4). According to the definition of block coherence of the overcomplete dictionary [14,63,67], we can reformulate Problem (7) in another form:

$$\left[\boldsymbol{D}^{(t)}, \boldsymbol{A}^{(t)}\right] = \arg\min_{\boldsymbol{D},\boldsymbol{A}} \left\{ \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2 + \lambda \sum_{j=1}^{B} \sum_{p,q \in b_j, p \neq q} \left\|\boldsymbol{\varphi}_p^{\mathrm{T}} \boldsymbol{\varphi}_q\right\|^2 \right\}, \quad \text{s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}^{(t)}} \leqslant \kappa, \ \forall i, \tag{15}$$

where $\boldsymbol{\varphi}_p$ and $\boldsymbol{\varphi}_q$ denote the various atoms of different blocks in the current dictionary, respectively.

This Problem (15) can be solved in two steps: first by updating dictionary, then regulating the block coherence for the available dictionary. In the first step, by omitting the second term on the right-hand side of the equation defined in Problem (15), we can solve the reduced version of Problem (15) as follows:

$$\left[\boldsymbol{D}^{(t)}, \boldsymbol{A}^{(t)}\right] = \arg\min_{\boldsymbol{D},\boldsymbol{A}} \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F^2, \text{ s.t. } \|\boldsymbol{\alpha}_i\|_{0,\boldsymbol{b}^{(t)}} \leqslant \kappa, \ \forall i. \tag{16}$$

Like block K-SVD method [46], we sequentially update every block of atoms in the dictionary $\boldsymbol{D}^{(t)}$ and the corresponding nonzero values in $\boldsymbol{A}^{(t)}$ to minimize the representation error in Problem (16). For each

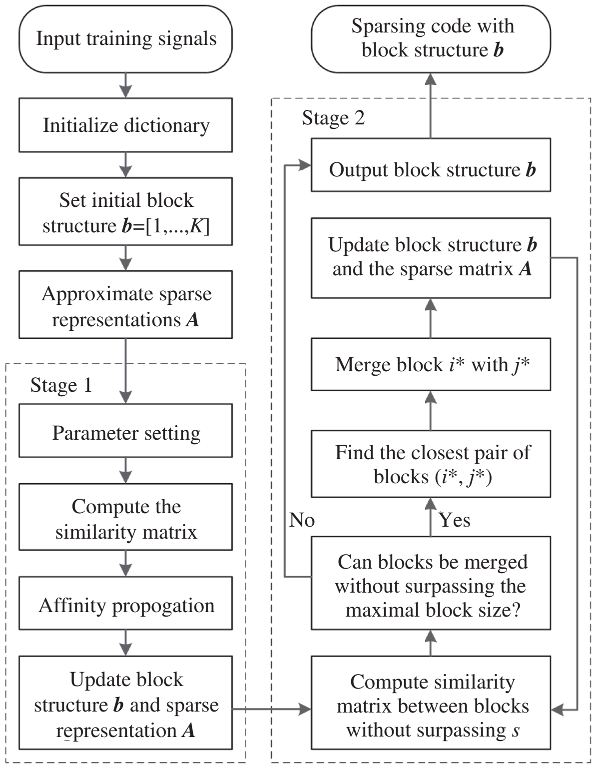

Figure 2 The procedures of our two-stage clustering approach.

block $j \in [1, B]$, the dictionary update step proceeds as follows. Let $\boldsymbol{R}_{\omega_j}$ denote the representative error for the signals $\boldsymbol{X}_{\omega_j}$ excluding the contribution by the jth block, i.e. $\boldsymbol{R}_{\omega_j} = \boldsymbol{X}_{\omega_j} - \sum_{i \neq j} \boldsymbol{D}_{b_i} \boldsymbol{A}_{\omega_j}^{b_i}$. We deduce that $\|\boldsymbol{R}_{\omega_j} - \boldsymbol{D}_{b_j} \boldsymbol{A}_{\omega_j}^{b_j}\|_F$ is the representative error for the signals with the indices ω_j. Apparently, the value of each atom in the dictionary depends not only on how much it is used to encode $\boldsymbol{X}$, but also on how much other atoms are being used to encode $\boldsymbol{X}$. In order to minimize this error, we take the matrix of maximum rank $|b_j|$ that best approximates of $\boldsymbol{R}_{\omega_j}$ as $\boldsymbol{D}_{b_j} \boldsymbol{A}_{\omega_j}^{b_j}$. According to the singular value decomposition (SVD) of a matrix, $\boldsymbol{R}_{\omega_j}$ can be written as the following formula:

$$\boldsymbol{R}_{\omega_j} = \boldsymbol{U} \boldsymbol{\Delta} \boldsymbol{V}^{\mathrm{T}}. \tag{17}$$

Therefore, the dictionary update process is carried out in the following form:

$$\boldsymbol{D}_{b_j} = \left[\boldsymbol{U}_1, \ldots, \boldsymbol{U}_{|b_j|}\right], \tag{18}$$

$$\boldsymbol{A}_{\omega_j}^{b_j} = \left[\boldsymbol{\Delta}_1^1 \boldsymbol{V}_1, \ldots, \boldsymbol{\Delta}_{|b_j|}^{|b_j|} \boldsymbol{V}_{|b_j|}\right]^{\mathrm{T}}, \tag{19}$$

where the first $|b_j|$ principal components of $\boldsymbol{R}_{\omega_j}$ are truncated to update the block of atoms $\boldsymbol{D}_{b_j}$ and the group sparse coefficients $\boldsymbol{A}_{\omega_j}^{b_j}$. Note that the $|b_j| - |\omega_j|$ excess atoms in block j can be discarded in case $|b_j| > |\omega_j|$. The simultaneous updates of atoms in the same block speeds up the convergence of the dictionary update step in the block K-SVD algorithm than that in the K-SVD method [42].

Finally, the second step of the solution to Problem (15) is to regulate the block coherence of the updated dictionary $\boldsymbol{D}^{(t)}$. Suppose that the optimal solution for each atom has a non-zero norm. We compute the gradient of the objective function with respect to $\boldsymbol{\varphi}_r$ and equate it to zero. Thus the closed-form solution of Problem (15) on $\boldsymbol{\varphi}_r$ is given as follows:

$$\boldsymbol{\varphi}_r = \left(\boldsymbol{I}_d \boldsymbol{\alpha}_r \boldsymbol{\alpha}_r^{\mathrm{T}} + \lambda \sum_{j \in b_j, j \neq r} \boldsymbol{\varphi}_j \boldsymbol{\varphi}_j^{\mathrm{T}}\right) \Bigg\backslash \left(\boldsymbol{X} \boldsymbol{\alpha}_r^{\mathrm{T}} - \sum_{k \neq r} \boldsymbol{\varphi}_k \boldsymbol{\alpha}_k \boldsymbol{\alpha}_r^{\mathrm{T}}\right), \tag{20}$$

where $\boldsymbol{I}_d$ is the identity matrix of size $d \times d$, $\boldsymbol{\alpha}_r$ is the rth row group of the sparse representation $\boldsymbol{A}^{(t)}$, $\boldsymbol{\varphi}_r$ is in the sub-block of atoms for b_j and $\boldsymbol{\alpha}_r \boldsymbol{\alpha}_r^{\mathrm{T}}$ indicates the weight of the atom $\boldsymbol{\varphi}_r$ used to encode $\boldsymbol{X}$. Furthermore, the zero atoms may be filled with random vectors for the overcomplete representation of signals.

3 Experimental results and analysis

In this work, we carried out the experiments on simulated data as well as real-world images. The proposed BSIK-SVD algorithm was compared with the state-of-the-art methods published recently [42, 46] for verifying its validity subjectively and objectively. For a fair comparison, the Gaussian random initialization is carried out for these different methods in the simulation and experiments.

3.1 Simulation evaluation

To test the performance of the proposed BSIK-SVD algorithm comprehensively, we have implemented the qualitative and quantitative evaluation on the simulated data. The capability of the proposed algorithm was tested and analyzed in retrieving the underlying block structure and optimizing the structured dictionary update. The simulation setting is given as follows. A random matrix of size $d \times K$ is generated as the dictionary $\boldsymbol{D} \in \mathbb{R}^{64\times 96}$ with independent and identically normally distributed entries. Each column of the dictionary is normalized so that its Euclidean norm equals to 1. The block structure for the dictionary $\boldsymbol{D} \in \mathbb{R}^{64\times 96}$ is chosen as $\boldsymbol{b} = [1, 1, 1, 2, 2, 2, \ldots, 32, 32, 32]$. That is, the dictionary $\boldsymbol{D}$ consists of 32 subspaces with 3 atoms in each one. $\boldsymbol{X}$ is a set of the $L = 2500$ test signals of dimension 64 with a 2-block sparse representation over $\boldsymbol{D}$. The active blocks in $\boldsymbol{A}$ are chosen randomly and the coefficients are uniformly redistributed random entries. According to the signal modeling defined in (1), the additive white Gaussian noise (AWGN) with variant noise levels is added to the data set $\boldsymbol{X}$ in order to generate the simulated data synthetically with the desired SNR values. We also assessed the performance of our BSIK-SVD by using the objective evaluation indexes, one of which is the normalized representation error (NRE) defined as follows:

$$\mathrm{NRE} = \|\boldsymbol{X} - \boldsymbol{D}\boldsymbol{A}\|_F / \|\boldsymbol{X}\|_F, \tag{21}$$

where $\|\cdot\|_F$ is the Frobenius norm.

In this simulation, for the given signal $\boldsymbol{X}$, we evaluated our proposed algorithm for recovering the overcomplete dictionary $\boldsymbol{D}$ with the underlying block structure $\boldsymbol{b}$. The NRE was computed as a function of the signal-to-noise ratio (SNR) of the signals $\boldsymbol{X}$ corrupted by the Gaussian noise according to (1). For a quantitative comparison, the oracle method assumes that the dictionary $\boldsymbol{D}$ and the sparse matrix $\boldsymbol{A}$ are known for the given signals $\boldsymbol{X}$. For the synthetic signals $\boldsymbol{X}$ with the variant SNR, the results of the proposed BSIK-SVD algorithm are compared with those of the oracle method, the K-SVD method [42] and the BK-SVD+SAC method [46] shown in Table 1. Note that for SNR $\leqslant$ 25 dB, K-SVD [42] reaches lower reconstruction error than BK-SVD+SAC [46] and our BSIK-SVD, which implies that the use of block-sparsifying dictionaries is unjustified. That is, the block structure is absent in the data when the SNR is very low, whereas the block structure appears when the SNR is high. Therefore, the dictionary learning methods based on the block structure incurs incorrect results at the low SNR values. Since there are unpredictable fluctuations of NRE in the iterative process for our proposed algorithm and the competing methods [42,46], we repeated this simulation 100 times in the noiseless setting and computed the average values as the final NREs for each of these methods, respectively. Figure 3 compares the averaged NRE results of these different algorithms evaluated as a function of the number of iterations in the noiseless setting. As can be seen from simulation results, the proposed BSIK-SVD algorithm has smaller representation errors in high SNR cases and faster convergence rate than the existing state-of-the-art methods.

Table 1 The compared NRE results of our proposed BSIK-SVD algorithm, K-SVD [42], BK-SVD+SAC [46] and the oracle for simulated data with the variant SNR values

SNR	5	10	15	20	25	30	35	40	45	50
Oracle	0.6224	0.4088	0.2443	0.1401	0.0793	0.0447	0.0252	0.0142	0.0080	0.0045
Ref. [42]	0.4736	0.3282	0.2104	0.1274	0.0734	0.0424	0.0264	0.0162	0.0128	0.0106
Ref. [46]	0.5691	0.3835	0.2298	0.1326	0.0756	0.0424	0.0243	0.0135	0.0077	0.0044
BSIK-SVD	0.5645	0.3814	0.2293	0.1330	0.0745	0.0422	0.0240	0.0133	0.0075	0.0042

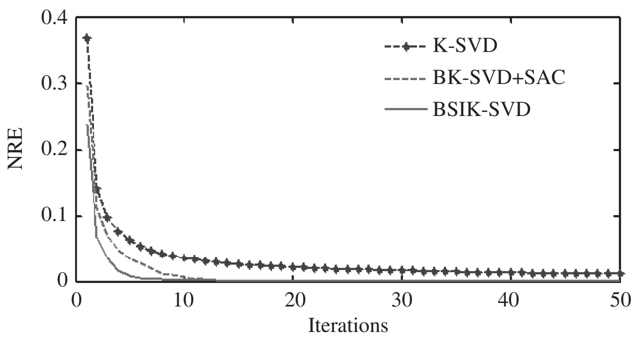

Figure 3 The compared average NRE results of the proposed BSIK-SVD algorithm, K-SVD [42] and BK-SVD+SAC [46] evaluated as a function of the number of iterations in the noiseless setting.

3.2 Experiments on real images

To further measure the sparse representation performance of the proposed BSIK-SVD algorithm, we comprehensively compared it with the state-of-the-art methods [42,46]. Besides the simulations on synthetic signals, we have also implemented the qualitative and quantitative evaluation on a set of test images from standard image databases1)2) and the literature [46]. The well-known full-reference quality metrics were considered for measuring the similarity between the restored image and the original image. We adopted the Structural SIMilarity (SSIM) [68] that was reportedly more reliable than PSNR [69] for evaluating the performance of these different algorithms. For sparse representation of signals, each patch of the test image can be represented as a linear combination of several atoms selected from the given overcomplete dictionary satisfying the limited error criterion.

In this experiment, the specific parameters of our BSIK-SVD algorithm are set as follows: $d = 64$, $K = 96$, $\kappa = 2$, $\lambda = 1$, $s = 3$ and $\beta_c = 0.99$. We first initialized the dictionary $\boldsymbol{D}$ as a random matrix of size $d \times K$ with normally distributed entries and its normalized columns. All the non-overlapping image patches of the same size 8×8 extracted from a test image were reshaped as the training signals $\boldsymbol{X}$. Then we adopted the proposed BSIK-SVD algorithm and other popular methods [42,46] to optimize the dictionary where the number of iteration is 50. Like K-SVD [42] and BK-SVD+SAC [42], the dictionary atoms of our BSIK-SVD algorithm are normalized in each iteration. Next, after extracting all the overlapping patches, we adopted a sensing matrix [70] to compress the 64 long vectors into 16 dimensions and then detected the corresponding sparse representation of each patch. Technically, both the proposed BSIK-SVD algorithm and the BK-SVD+SAC method were used to find a κ-block sparse solution, while the K-SVD method was used to find a $\kappa \times s$ sparse representation. Finally, we evaluated the results of these algorithms by the NRE and their convergence. For a set of noiseless test images, Table 2 shows the comparison of the NRE/SSIM results obtained by the prevalent methods [42,46] and our BSIK-SVD algorithm separately. To evaluate the computational complexity of dictionary learning, the runtime of our BSIK-SVD algorithm is compared with that of the baseline methods using Matlab version 8.2 on the platform of Intel(R) Core(TM) i7-3537U CPU @ 2.00 GHz, 8.00 GB RAM. Table 3 shows the comparison of the computation time of these methods. Since both the two-stage clustering approach and the incoherence regularization term are

1) CIPR Still Images, May 26, 2014. http: //www.cipr.rpi.edu/resource/stills/index.html.
2) CVG, Test images, May 26, 2014. http: //decsai.ugr.es/cvg/dbimagenes/.

Table 2 The NRE/SSIM results of K-SVD [42], BK-SVD+SAC [46] and the proposed BSIK-SVD algorithm separately applied to a test set of noiseless images*

Images	Ref. [42]	Ref. [46]	BSIK-SVD
Cutebaby 200×201	0.5048/0.7054	0.0744/0.8714	**0.0585/0.8990**
Einstein 256×256	0.2735/0.8100	0.1228/0.8087	**0.1082/0.8423**
House 256×256	0.1603/0.8939	0.0773/0.9118	**0.0737/0.9182**
Monarch 256×256	0.3292/0.8463	0.2629/0.8584	**0.2359/0.8640**
Barbara 512×512	0.1621/0.8834	0.1095/0.8948	**0.1073/0.9004**
Boat 512×512	0.1546/0.8530	0.1031/0.8571	**0.1016/0.8604**
Elaine 512×512	0.1047/0.8350	0.0586/0.8473	**0.0577/0.8511**
Hill 512×512	0.2082/**0.8516**	0.1290/0.8314	**0.1064**/0.8509
Lena 512×512	0.1326/0.9011	0.0791/0.9049	**0.0713/0.9107**
Milkdrop 512×512	0.1142/0.9212	0.0861/0.9073	**0.0552/0.9317**
Peppers 512×512	0.1190/0.8848	0.0860/0.8774	**0.0789/0.8880**
Plane 512×512	0.1734/0.8680	0.1295/0.9043	**0.0754/0.9238**

*Note that the best results are highlighted in bold.

Table 3 Runtime results of K-SVD [42], BK-SVD+SAC [46] and our BSIK-SVD algorithm (Unit: second)

Images	Ref. [42]	Ref. [46]	BSIK-SVD
House 256×256	20.31	21.92	40.97
Lena 512×512	53.32	50.86	87.75

incorporated in the design of our proposed dictionary learning model, the computational complexity of our proposed algorithm is increased. To further inspect the performance of dictionary learning, the detailed reconstructed results of the proposed BSIK-SVD algorithm, K-SVD [42] and BK-SVD+SAC [46] for the fragments of the *Cutebaby*, *Monarch*, *Hill* and *Plane* images are shown in Figures 4–7, respectively. As can be seen from Tables 2 and 3, although our proposed BSIK-SVD algorithm has a little longer computation time, it has less NRE and higher SSIM results than those from K-SVD [42] and BK-SVD+SAC [46]. The visual comparisons in Figures 4–7 demonstrate that the proposed BSIK-SVD algorithm have better recovery of real images than the baseline methods. It is observed that the results of our proposed algorithm have sharper edges and clearer structures than those of the state-of-the-art methods. The experimental reslults on sparse representation of synthetic and real images demonstrate that the proposed algorithm has the high reconstruction accuracy, the good robustness and visual quality.

4 Conclusion and future work

In this paper, we addressed the problem of block-structured incoherent dictionary learning for block sparse representations of image signals. Unlike the traditional method, e.g., BK-SVD+SAC [46], our proposed BSIK-SVD framework of dictionary learning can break the assumption of the maximal block size of similar atoms. In fact, our BSIK-SVD algorithm, which can be seen as an extension of K-SVD [42] or BK-SVD+SAC [46], consists of two successive steps at each iteration: sparse coding and dictionary update. In sparse coding step, we employed the two-stage clustering technique to identify the existence of intrinsic block structure in the dictionary for structured sparse coding. For dictionary update step, we solved the minimization problem that incorporated the desired constraints of the bounded block coherence between the dictionary atoms after the block-sparsifying dictionary update. From the simulation and experiments, we have shown that the procedures of the two-stage clustering and the block coherence adjustment greatly influence the reconstructed results. The experimental results of synthetic data and real images demonstrate that our BSIK-SVD algorithm has less representation error and fewer artifacts when the SNR is high enough, which leads to accurate sparse representations of signals. Our BSIK-SVD algorithm outperforms the state-of-the-art methods both visually and quantitatively, especially in the

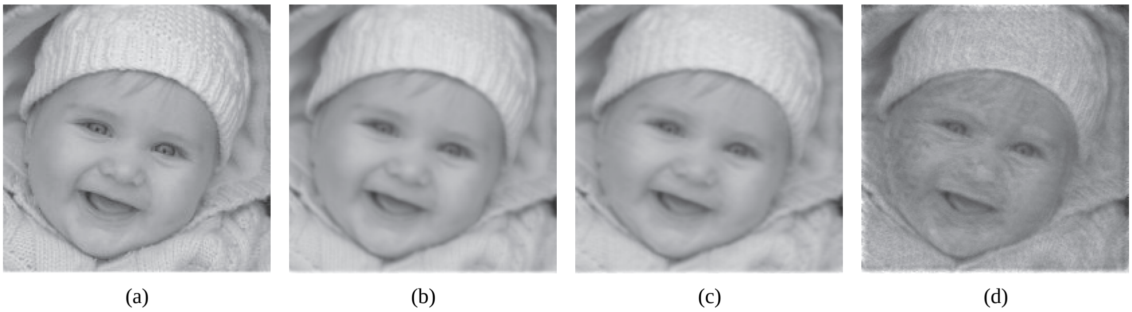

Figure 4 Visual comparisons of the results of the proposed algorithm and other state-of-the-art methods for the *Cutebaby* image. From left to right. (a) Baseline image; (b) the proposed algorithm (NRE = 0.0585, SSIM = 0.8990, PSNR = 33.43 dB); (c) BK-SVD+SAC [46] (NRE = 0.0744, SSIM = 0.8714, PSNR = 32.30 dB); and (d) K-SVD [42] (NRE = 0.5048, SSIM = 0.7054, PSNR = 18.54 dB).

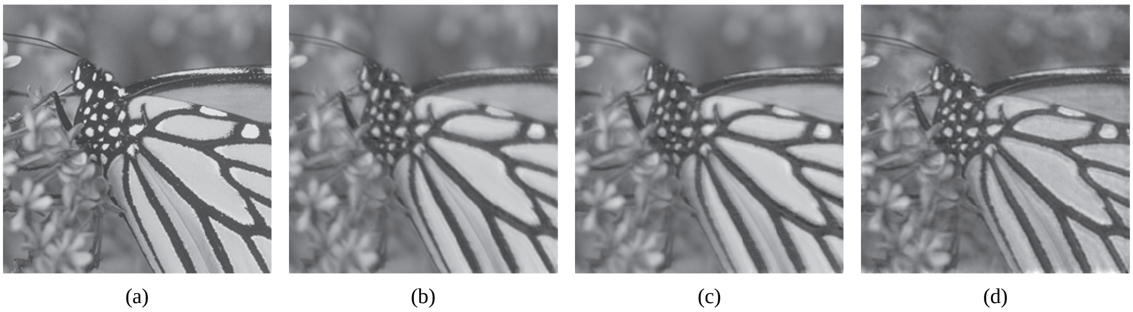

Figure 5 Visual comparisons of the results of the proposed algorithm and other state-of-the-art methods for the *Monarch* image. From left to right. (a) Baseline image; (b) the proposed algorithm (NRE = 0.2359, SSIM = 0.8640, PSNR = 24.44 dB); (c) BK-SVD+SAC [46] (NRE = 0.2629, SSIM = 0.8584, PSNR = 23.72 dB); and (d) K-SVD [42] (NRE = 0.3292, SSIM = 0.8463, PSNR = 23.02 dB).

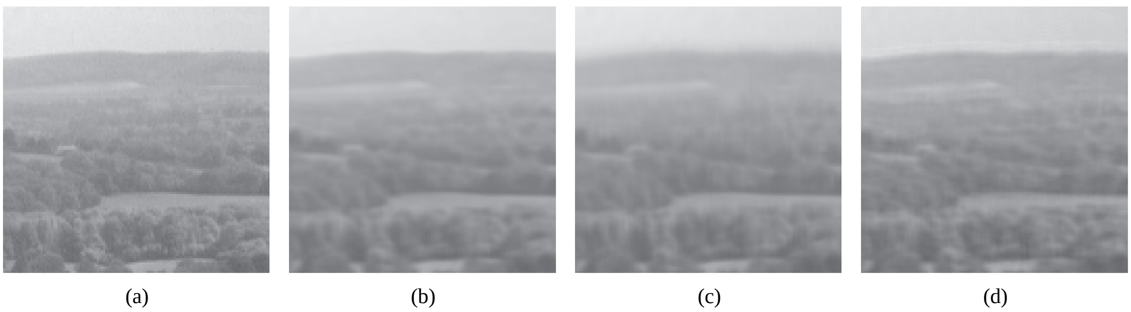

Figure 6 Visual comparisons of the results of the proposed algorithm and other state-of-the-art methods for the *Hill* image. From left to right. (a) Baseline image; (b) the proposed algorithm (NRE = 0.1064, SSIM = 0.8509, PSNR = 31.12 dB); (c) BK-SVD+SAC [46] (NRE = 0.1290, SSIM = 0.8314, PSNR = 30.06 dB); and (d) K-SVD [42] (NRE = 0.2082, SSIM = 0.8516, PSNR = 29.03 dB).

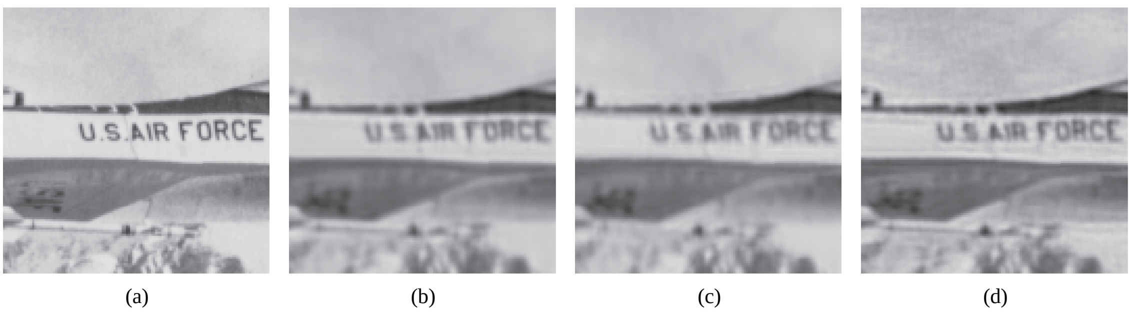

Figure 7 Visual comparisons of the results of the proposed algorithm and other state-of-the-art methods for the *Plane* image. From left to right. (a) Baseline image; (b) the proposed algorithm (NRE = 0.0585, SSIM = 0.8990, PSNR = 33.43 dB); (c) BK-SVD+SAC [46] (NRE = 0.0744, SSIM = 0.8714, PSNR = 32.30 dB); and (d) K-SVD [42] (NRE = 0.5048, SSIM = 0.7054, PSNR = 18.54 dB).

case of image regions containing the abundant edges and the fine structures. Finally, the issues of the convergence and the efficiency of dictionary learning algorithms are left for our future research.

Acknowledgements

This work was supported by National Natural Science Foundation of China (Grant Nos. 61201442, 61471272), and China Postdoctoral Science Foundation (Grant No. 2013M530481).

References

1 Mairal J, Elad M, Sapiro G. Sparse representation for color image restoration. IEEE Trans Image Process, 2008, 17: 53–69
2 Zhang Z Y. Community structure detection in social networks based on dictionary learning. Sci China Inf Sci, 2013, 56: 078103
3 Olshausen B A, Field D J. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature, 1996, 381: 607–609
4 Vinje W E, Gallant J L. Sparse coding and decorrelation in primary visual cortex during natural vision. Science, 2000, 287: 1273–1276
5 Nirenberg S, Carcieri S M, Jacobs A L, et al. Retinal ganglion cells act largely as independent encoders. Nature, 2001, 411: 698–701
6 Fadili M J, Starck J L, Bobin J, et al. Image decomposition and separation using sparse representations: an overview. Proc IEEE, 2010, 98: 983–994
7 Elad M, Milanfar P, Rubinstein R. Analysis versus synthesis in signal priors. Inverse Probl, 2007, 23: 947–968
8 Bruckstein A M, Donoho D L, Elad M. From sparse solutions of systems of equations to sparse modeling of signals and images. SIAM Rev, 2009, 51: 34–81
9 Elad M. Sparse and redundant representation modeling-What next? IEEE Signal Process Lett, 2012, 19: 922–928
10 Mallat S, Zhang Z. Matching pursuits with time-frequency dictionaries. IEEE Trans Signal Process, 1993, 41: 3397–3415
11 Tropp J A. Greed is good: algorithmic results for sparse approximation. IEEE Trans Inf Theory, 2004, 50: 2231–2242
12 Cai T T, Wang L. Orthogonal matching pursuit for sparse signal recovery with noise. IEEE Trans Inf Theory, 2011, 57: 4680–4688
13 Rubinstein R, Zibulevsky M, Elad M. Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit. Technical Report, CS, Technion, 2008
14 Eldar Y C, Kuppinger P, Bolcskei H. Block-sparse signals: uncertainty relations and efficient recovery. IEEE Trans Signal Process, 2010, 58: 3042–3054
15 Donoho D L, Tsaig Y, Drori I, et al. Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit. IEEE Trans Inf Theory, 2012, 58: 1094–1121
16 Zhang W H, Zhou T, Huang B X. Outlier deletion based improvement on the StOMP algorithm for sparse solution of large-scale underdetermined problems. Sci China Inf Sci, 2014, 57: 092203
17 Needell D, Tropp J A. Cosamp: iterative signal recovery from incomplete and inaccurate samples. Appl Comput Harmonic Anal, 2009, 26: 301–321
18 Efron B, Hastie T, Johnstone I, et al. Least angle regression. Ann Statist, 2004, 32: 407–499
19 Dai W, Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction. IEEE Trans Inf Theory, 2009, 55: 2230–2249
20 Blumensath T, Davies M E. Gradient pursuits. IEEE Trans Signal Process, 2008, 56: 2370–2382
21 Chen S S, Donoho D L, Saunders M A. Atomic decomposition by basis pursuit. SIAM Rev, 2001, 43: 129–159
22 Eldar Y C, Mishali M. Robust recovery of signals from a structured union of subspaces. IEEE Trans Inf Theory, 2009, 55: 5302–5316
23 Gorodnitsky I F, Rao B D. Sparse signal reconstruction from limited data using FOCUSS: a re-weighted minimum norm algorithm. IEEE Trans Signal Process, 1997, 45: 600–616
24 Wright M H. The interior-point revolution in optimization: history, recent developments, and lasting consequences. Bull Amer Math Soc, 2004, 42: 39–56
25 Elad M. Why simple shrinkage is still relevant for redundant representations. IEEE Trans Inf Theory, 2006, 52: 5559–5569
26 Bioucas-Dias J M, Figueiredo M A T. A new TwIST: two-step iterative shrinkage/thresholding algorithms for image restoration. IEEE Trans Image Process, 2007, 16: 2992–3004

27 Ahmed N, Natarajan T, Rao K R. Discrete cosine transform. IEEE Trans Comput, 1974, C-23: 90–93
28 Mallat S. A Wavelet Tour of Signal Processing, 3rd Edition: the Sparse Way. New York: Academic Press, 2008

29 Ma J, Plonka G. The curvelet transform. IEEE Signal Process Mag, 2010, 27: 118–133

30 Do M N, Vetterli M. The contourlet transform: an efficient directional multiresolution image representation. IEEE Trans Image Process, 2005, 14: 2091–2106

31 Do M N, Vetterli M. The finite ridgelet transform for image representation. IEEE Trans Image Process, 2003, 12: 16–28

32 Le Pennec E, Mallat S. Sparse geometric image representations with bandelets. IEEE Trans Image Process, 2005, 14: 423–438

33 Easley G, Labate D, Lim W Q. Sparse directional image representations using the discrete shearlet transform. Appl Comput Harmonic Anal, 2008, 25: 25–46

34 Tosic I, Frossard P. Dictionary Learning: what is the right representation for my signal? IEEE Signal Process Mag, 2011, 28: 27–38

35 Olshausen B A, Field D J. Sparse coding with an overcomplete basis set: a strategy employed by V1? Vis Res, 1997, 37: 3311–3325

36 Engan K, Aase S O, Hakon Husoy J. Method of optimal directions for frame design. In: Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, Phoenix, 1999. 2443–2446

37 Kreutz-Delgado K, Murray J, Rao B, et al. Dictionary learning algorithms for sparse representation. Neural Comput, 2003, 15: 349–396

38 Yaghoobi M, Blumensath T, Davies M E. Dictionary learning for sparse approximations with the majorization method. IEEE Trans Signal Process, 2009, 57: 2178–2191

39 Mairal J, Bach F, Ponce J, et al. Online learning for matrix factorization and sparse coding. J Mach Learn Res, 2010, 11: 19–60

40 Mairal J, Bach F, Ponce J. Task-driven dictionary learning. IEEE Trans Patt Anal Mach Intell, 2012, 34: 791–804

41 Schmid-Saugeon P, Zakhor A. Dictionary design for matching pursuit and application to motion-compensated video coding. IEEE Trans Circ Syst Video Technol, 2004, 14: 880–886

42 Aharon M, Elad M, Bruckstein A. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans Signal Process, 2006, 54: 4311–4322

43 Stewart G W. On the early history of the singular value decomposition. SIAM Rev, 1993, 35: 551–566

44 Zhang Y Q, Ding Y, Xiao J S, et al. Visibility enhancement using an image filtering approach. EURASIP J Appl Signal Process, 2012, 220: 1–6

45 Zhang Y Q, Ding Y, Liu J Y, et al. Guided image filtering using signal subspace projection. IET Image Process, 2013, 7: 270–279

46 Zelnik-Manor L, Rosenblum K, Eldar Y C. Dictionary optimization for block-sparse representations. IEEE Trans Signal Process, 2012, 60: 2386–2395

47 Eksioglu E M. A clustering based rramework for dictionary block structure identification. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Prague, 2011. 4044–4047

48 Oiwa H, Matsushima S, Nakagawa H. Feature-aware regularization for sparse online learning. Sci China Inf Sci, 2014, 57: 052104

49 Barchiesi D, Plumbley M D. Learning incoherent dictionaries for sparse approximation using iterative projections and rotations. IEEE Trans Signal Process, 2013, 61: 2055–2065

50 Zhang Y Q, Liu J Y, Li M D, et al. Joint image denoising using adaptive principal component analysis and self-similarity. Inf Sci, 2014, 259: 128–141

51 Gribonval R, Nielsen M. Sparse representations in unions of bases. IEEE Trans Inf Theory, 2003, 49: 3320–3325

52 Engan K, Skretting K, Husoy J H. Family of iterative LS-based dictionary learning algorithms, ILS-DLA, for sparse signal representation. Digit Signal Process, 2007, 17: 32–49

53 Chi Y T, Ali M, Rajwade A, et al. Block and group regularized sparse modeling for dictionary learning. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Portland, 2013. 1–8

54 Mairal J, Sapiro G, Elad M. Learning multiscale sparse representations for image and video restoration. Multiscale Model Simul, 2008, 7: 214–241

55 Ophir B, Lustig M, Elad M. Multi-scale dictionary learning using wavelets. IEEE J-STSP, 2011, 5: 1014–1024

56 Barthelemy Q, Larue A, Mayoue A, et al. Shift boolean and 2D rotation invariant sparse coding for multivariate signals. IEEE Trans Signal Process, 2012, 60: 1597–1611

57 Meng D Y, Zhao Q, Leung Y, et al. Learning dictionary from signals under global sparsity constraint. Neurocomputing, 2013, 119: 308–318

58 Peng Y, Meng D Y, Xu Z B, et al. Decomposable nonlocal tensor dictionary learning for multispectral image denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Columbus, 2014. 2949–2956

59 Bengio S, Pereira F, Singer Y, et al. Group sparse coding. In: Proceedings of Advances in Neural Information Processing Systems, Vancouver, 2009. 82–89

60 Gao S, Chia L T, Tsang I W. Multi-layer group sparse coding—for concurrent image classification and annotation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Providence, 2011. 2809–2816

61 Elhamifar E, Vidal R. Block-sparse recovery via convex optimization. IEEE Trans Signal Process, 2012, 60: 4094–4107

62 Strohmer T, Heath Jr R W. Grassmannian frames with applications to coding and communication. Appl Comput Harmonic Anal, 2003, 14: 257–275
63 Eldar Y C, Bolcskei H. Block-sparsity: coherence and efficient recovery. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, Taipei, 2009. 2885–2888
64 Murtagh F, Contreras P. Algorithms for hierarchical clustering: an overview. Data Min Knowl Discov, 2012, 2: 86–97
65 Frey B J, Dueck D. Clustering by passing messages between data points. Science, 2007, 315: 972–976
66 Wang C D, Lai J H, Suen C Y, et al. Multi-exemplar affinity propagation. IEEE Trans Patt Anal Mach Intell, 2013, 35: 2223–2237
67 Sigg C D, Dikk T, Buhmann J M. Learning dictionaries with bounded self-coherence. IEEE Signal Process Lett, 2012, 19: 861–864
68 Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process, 2004, 13: 600–612
69 Huynh-Thu Q, Ghanbari M. Scope of validity of PSNR in image/video quality assessment. Electron Lett, 2008, 44: 800–801
70 Duarte-Carvajalino J M, Sapiro G. Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization. IEEE Trans Image Process, 2009, 18: 1395–1408