Neural Networks 98 (2018) 212–222

Contents lists available at ScienceDirect

Neural Networks

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

Manifold optimization-based analysis dictionary learning with an $\ell_{1/2}$-norm regularizer

Zhenni Li [a], Shuxue Ding [b], Yujie Li [c], Zuyuan Yang [a], Shengli Xie [a], Wuhui Chen [d,*]

[a] School of Automation, Guangdong Key Laboratory of IoT Information Technology, Guangdong University of Technology, Guangzhou, 510006, China
[b] School of Computer Science and Engineering, The University of Aizu Aizu-Wakamatsu City, Fukushima, 965-8580, Japan
[c] Artificial Intelligence Center, AIST Tsukuba, Ibaraki 305-8560, Japan
[d] School of Data and Computer Science, Sun Yat-sen University, China

ARTICLE INFO

Article history:
Received 25 June 2017
Received in revised form 13 November 2017
Accepted 21 November 2017
Available online 6 December 2017

Keywords:
Sparse model
Analysis dictionary learning
$\ell_{1/2}$ norm regularizer
Orthonormality constraint
Manifold optimization

ABSTRACT

Recently there has been increasing attention towards analysis dictionary learning. In analysis dictionary learning, it is an open problem to obtain the strong sparsity-promoting solutions efficiently while simultaneously avoiding the trivial solutions of the dictionary. In this paper, to obtain the strong sparsity-promoting solutions, we employ the $\ell_{1/2}$ norm as a regularizer. The very recent study on $\ell_{1/2}$ norm regularization theory in compressive sensing shows that its solutions can give sparser results than using the ℓ_1 norm. We transform a complex nonconvex optimization into a number of one-dimensional minimization problems. Then the closed-form solutions can be obtained efficiently. To avoid trivial solutions, we apply manifold optimization to update the dictionary directly on the manifold satisfying the orthonormality constraint, so that the dictionary can avoid the trivial solutions well while simultaneously capturing the intrinsic properties of the dictionary. The experiments with synthetic and real-world data verify that the proposed algorithm for analysis dictionary learning can not only obtain strong sparsity-promoting solutions efficiently, but also learn more accurate dictionary in terms of dictionary recovery and image processing than the state-of-the-art algorithms.

1. Introduction

Sparsity-inspired models have demonstrated their value in signal processing, machine learning and optimization for networks (Liang, Zhang, Zeng, & Yu, 2014; Yang, Zhang, Yan, Xiang, & Xie, 2016; Zaki, Venkitaraman, Chatterjee, & Rasmussen, 2017; Zhang & Desrosiers, 2017). The best-known model is the sparse synthesis model (Elad, 2010), in which a natural signal $\mathbf{y} \in \mathbb{R}^m$ is assumed to be created as a linear combination of a few atoms (columns) from the overcomplete dictionary $\mathbf{W} \in \mathbb{R}^{m\times r}$ $(m < r)$. The model can be described by the relation $\mathbf{y} \approx \mathbf{W}\mathbf{h}$, where $\mathbf{h} \in \mathbb{R}^r$ is the signal's sparse representation with s nonzero elements and s is defined as the sparsity. The synthesis model is dependent on the number s of nonzero elements. Only a small number of columns in $\mathbf{W}$ are selected to represent each signal $\mathbf{y}$; therefore, every atom has the great significance. The wrong choice for the large weights of atoms leads to further deviation from the desired description. In contrast to the synthesis model, a sparse analysis model (Nam, Davies, Elad, & Gribonval, 2010) has been proposed. This model employs the overcomplete analysis dictionary $\mathbf{\Omega} \in \mathbb{R}^{r\times m}$ $(r > m)$ to "analyze" the signal $\mathbf{y}$ as $\mathbf{\Omega}\mathbf{y} \approx \mathbf{x}$, where $\mathbf{x} \in \mathbb{R}^r$ is the analysis representation of $\mathbf{y} \in \mathbb{R}^m$ and contains l zero elements. l is defined as the cosparsity. The rows in $\mathbf{\Omega}$ represent dictionary atoms. The analysis model is dependent on the number l of zero elements. The signal $\mathbf{y}$ is analyzed using all the rows of $\mathbf{\Omega}$. Furthermore, all atoms contribute equally to describing the signal, leading to a stabilizing effect in the recovery process.

A fundamental tool for sparse models is dictionary learning; i.e., the construction of a dictionary that is suitable for a family of signals (Aharon, Elad, & Bruckstein, 2006; Bahrampour, Nasrabadi, Ray, & Jenkins, 2016; Li, Ding, & Li, 2015; Rubinstein, Peleg, & Elad, 2013). Analysis dictionary learning (ADL) has attracted increasing attention recently (Hawe, Kleinsteuber, & Diepold, 2013; Rubinstein et al., 2013; Seibert, Wormann, Gribonval, & Kleinsteuber, 2016; Wen, Ravishankar, & Bresler, 2015), but faces two challenging issues. The first is how to choose a suitable sparsity constraint to drive the strong sparsity-promoting solution efficiently. The other is how to avoid trivial solutions for the analysis dictionary, such as $\mathbf{\Omega} = \mathbf{0}$!, which is a common problem in ADL.

Most ADL methods aiming for sparsity (Dong et al., 2015; Eksioglu & Bayir, 2014; Ravishankar & Bresler, 2013; Rubinstein et al.,

* Corresponding author.
E-mail addresses: lizhenni@gdut.edu.cn (Z. Li), sding@u-aizu.ac.jp (S. Ding), yujie-li@aist.go.jp (Y. Li), yangzuyuan@aliyun.com (Z. Yang), shlxie@gdut.edu.cn (S. Xie), chenwuh@mail.sysu.edu.cn (W. Chen).

https://doi.org/10.1016/j.neunet.2017.11.015
0893-6080/

2013) employ the ℓ_0 norm as sparsity constraint, which can enforce a strong sparsity. Unfortunately, the exact determination of the sparse representation using the ℓ_0 norm has been proved to be NP-hard, and only approximate solutions can be found by greedy algorithms. Some ADL approaches (Li, Ding, Hayashi, & Li, 2017; Yaghoobi, Nam, Gribonval, & Davies, 2013; Zhang, Yu, & Wang, 2014) choose the relaxation ℓ_1 norm, which is convex and the closed-form solution can be obtained easily. However, solutions using the ℓ_1 norm are not sparse enough. Furthermore, the ℓ_1 norm often leads to overpenalization of large elements of a sparse vector (Gong, Zhang, Lu, Huang, & Ye, 2013; Rakotomamonjy, Flamary, & Gasso, 2016). Therefore, choosing a suitable sparsity constraint to derive the strong sparsity-promoting solutions efficiently is imperative for the ADL problem.

To avoid trivial solutions in ADL, an orthonormality constraint on the dictionary has been proposed (Yaghoobi et al., 2013). The dictionary learning work (Yaghoobi et al., 2013) with the orthonormality constraint consists of first updating in the embedding Euclidean space $\mathbb{R}^d$ – i.e., without considering the constraint – and then an additional step of projecting the result onto the manifold so that the orthonormality constraint is satisfied. Such a method obtains an approximate solution and may not capture the intrinsic structure of the inherent dictionary, so that the important intrinsic properties of the data may not be represented in such a dictionary. This can be illustrated by a simple example. It is possible that two points x and y on the manifold $\mathcal{S}$ have a large geodesic distance separating them, but under the embedding i: $\mathcal{S} \rightarrow \mathbb{R}^d$, $i(x)$ and $i(y)$ are separated by a small distance in $\mathbb{R}^d$. Therefore, sparse coding using the dictionary learned in $\mathbb{R}^d$ is likely to code $i(x)$ and $i(y)$ using the same set of atoms with similar coefficients. Clearly, this will be undesirable and unsatisfactory if the applications require tasks such as classification and clustering, for which one would prefer the sparse coding to reflect some degree of actual similarity (i.e., geodesic distance) between the two samples x and y. Therefore, learning an analysis dictionary that can avoid the trivial solution while capturing the intrinsic structure of data is an important problem.

In this paper, to obtain the strong sparsity-promoting solutions efficiently, we consider the nonconvex $\ell_{1/2}$ norm as the regularizer in the ADL problem. The $\ell_{1/2}$ norm not only yields sparser solution but also allows more accurate results in many sparse signal estimation and reconstruction problems than the ℓ_1 norm (Niu, Zhou, Tian, Qi, & Zhang, 2016; Xu, Zhang, Wang, & Chang, 2010). Although the $\ell_{1/2}$ norm regularizer results in a complex nonconvex optimization problem that is harder to solve efficiently than the convex optimization problem, we propose to solve the $\ell_{1/2}$ norm regularized ADL problem by optimizing a number of one-dimensional minimization problems. We can then obtain the closed-form solution in each iteration efficiently. To avoid the trivial solutions, we impose an orthonormality constraint on the dictionary. Unlike the conventional method which requires an additional projection step, we first apply manifold optimization to the ADL problem to compute the dictionary directly on the manifold constructed by the orthonormality constraint, instead of in the Euclidean space. The orthonormality constraint can then be preserved during iterations, avoiding degenerate solutions. The learned dictionary may capture the inherent structure of the signal.

Our contributions in this paper are as follows.

(1) We propose to use the nonconvex $\ell_{1/2}$ norm as a regularizer for strong sparsity. In particular, we can obtain a closed-form solution of the $\ell_{1/2}$ norm regularized ADL problem, leading to a fast analysis sparse-coding stage.

(2) We employ the optimization scheme on the manifold constructed by the orthonormality constraint. The dictionary is computed effectively and directly on the manifold; therefore, it can represent the important intrinsic properties of the data. Furthermore, the orthonormality constraint is solved simultaneously while the dictionary is updated, resulting in a simple analysis dictionary update stage.

(3) We present a novel and efficient algorithm for a manifold-optimization-based ADL with an $\ell_{1/2}$ norm regularizer (MADL-$\ell_{1/2}$), which can not only obtain the strong sparsity-promoting solutions in closed form, but also learn more accurate dictionary in terms of dictionary recovery and image processing than the state-of-the-art algorithms. We also provide a powerful adaptive strategy for regularization of parameter settings (based on the estimation of data sparsity information). In addition, we find that the mutual coherence can be reduced by the dictionary with the orthonormality constraint.

The remainder of this paper is organized as follows. Related work will be introduced in Section 2. The conventional ADL problem will be reviewed in Section 3. We will formulate our ADL problem first and then develop a novel and efficient ADL algorithm in Section 4. The adaptive regularization parameter setting will also be described in Section 4. The theoretical findings are verified by experiments in Section 5. Finally, conclusions will be drawn in Section 6.

2. Related work

Several ADL algorithms have been proposed in the literature, including the analysis K-singular value decomposition (AKSVD) algorithm (Rubinstein et al., 2013), the analysis operator learning (AOL) algorithm (Yaghoobi et al., 2013), the transform K-SVD (TKSVD) algorithm (Eksioglu & Bayir, 2014), the analysis simultaneous codeword-optimization (ASimCO) (Dong et al., 2015) algorithm, the learning overcomplete sparsifying transforms (LOST) (Ravishankar & Bresler, 2013), the orthogonality constrained ADL with iterative hard thresholding (OIHT-ADL) (Zhang et al., 2014), and the proximal analysis dictionary learning (PADL) (Li et al., 2017).

As a sparsity measure, most ADL algorithms use the ℓ_0 norm for strong sparsity, such as AKSVD, TKSVD, ASimCO, and LOST. AKSVD uses a backward greedy algorithm to detect the sparsity and has high complexity. TKSVD, ASimCO, and LOST employ another greedy algorithm to detect the sparsity, the hard thresholding operation, which yields approximate results in many sparse signal estimation and reconstruction problems. Some ADL algorithms use the relaxation ℓ_1 norm for sparsity and the optimization can be solved easily and efficiently using methods such as AOL, OIHT-ADL, and our PADL. However, the ℓ_1 norm leads to overpenalization of large elements of a sparse vector, and its solution is no sparser than that obtained with the ℓ_0 norm.

For the analysis dictionary update, AKSVD, TKSVD, LOST, ASimCO, and PADL update the dictionary without an orthonormality constraint, while AOL and OIHT-ADL update the analysis dictionary with an orthonormality constraint to avoid trivial solutions. AOL updates the analysis dictionary using the subgradient descent method in Euclidean space $\mathbb{R}^d$, without considering the orthonormality constraint. The updated dictionary is first projected onto the manifold so that the orthonormality constraint is satisfied and is then projected onto the space of unit row frames so that trivial solutions can be avoided. However, this method is an approximation and cannot capture the intrinsic properties of the inherent dictionary. OIHT-ADL employs the term $\|\mathbf{\Omega}^T\mathbf{\Omega} - \mathbf{I}\|_F^2$ as a penalty, leading to a nonconvex objective function. Local minimums of the objective function can be found in Euclidean space $\mathbb{R}^d$; however, the learned dictionary cannot capture the intrinsic properties of the data.

In contrast to the above ADL algorithms, we propose to apply the $\ell_{1/2}$ norm to the ADL problem to obtain the strong sparsity-promoting solutions efficiently. Preliminary results of using the

$\ell_{1/2}$ norm as the regularizer were presented in Li, Hayashi, Ding, Li, and Li (2016). In contrast to Li et al. (2016), we propose to learn the dictionary on the manifold directly, so that the dictionary captures the intrinsic properties of the data and observes the orthonormality constraint.

3. Preliminaries

In this section, we first introduce some notation. Then the conventional ADL formulation is introduced.

3.1. Notation

We briefly summarize notation in this paper. A boldface uppercase letter such as $\mathbf{X}$ denotes a matrix, and $\mathbf{x}_l$ denotes the lth column vector in $\mathbf{X}$. A lowercase element x_{jl} denotes the entry in the jth row and the lth column of $\mathbf{X}$. A lowercase element x_j denotes the jth entry in the vector $\mathbf{x}$. A boldface uppercase Greek letter $\mathbf{\Omega}$ denotes an analysis dictionary matrix and $\mathbf{w}^i$ is the ith row vector in $\mathbf{\Omega}$, i.e., the ith atom. The Frobenius norm of $\mathbf{X}$ is defined as $\|\mathbf{X}\|_F = (\sum_{j,l} \mid x_{jl}\mid^2)^{1/2}$ and ℓ_2 norm of $\mathbf{x}$ is defined as $\|\mathbf{x}\|_2 = (\sum_j \mid x_j\mid^2)^{1/2}$. The ℓ_1 norm of $\mathbf{X}$ and $\mathbf{x}$ are defined as $\|\mathbf{X}\|_1 = \sum_{j,l} \mid x_{jl} \mid$ and $\|\mathbf{x}\|_1 = \sum_j \mid x_j \mid$, respectively. The $\ell_{1/2}$ norm of $\mathbf{X}$ and $\mathbf{x}$ are defined as $\|\mathbf{X}\|_{1/2} = (\sum_{j,l} \mid x_{jl}\mid^{1/2})^2$ and $\|\mathbf{x}\|_{1/2} = (\sum_j \mid x_j\mid^{1/2})^2$, respectively. $[\cdot]_j$ denotes indexed vector entries. In this paper, all parameters take real values.

3.2. Analysis Dictionary Learning (ADL)

We now describe the ADL problem. Given a matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$ of observed signals, the aim of ADL in the analysis model is to find $\mathbf{\Omega}$ such that $\mathbf{\Omega Y}$ is as sparse as possible. This motivates finding the minimized sparsity of $\mathbf{\Omega Y}$:

$$\min_{\mathbf{\Omega}} G(\mathbf{\Omega Y}), \tag{1}$$

where $G(\cdot)$ is a sparsity-promoting function, generally a sparsity measure such as ℓ_0 norm or ℓ_1 norm. A noisy formulation of ADL in the analysis model can be given as follows:

$$\min_{\mathbf{\Omega},\mathbf{Y}} G(\mathbf{\Omega Y}), \quad s.t. \|\mathbf{Y}-\mathbf{S}\|_F \le \varepsilon, \tag{2}$$

where $\mathbf{S} = \mathbf{Y} + \mathbf{V}$, and $\mathbf{V}$ is a Gaussian noise matrix. If a Lagrangian multiplier λ is used for the noisy ADL, the problem can be expressed as follows:

$$\min_{\mathbf{\Omega},\mathbf{Y}} \frac{1}{2}\|\mathbf{Y}-\mathbf{S}\|_F^2 + \lambda G(\mathbf{\Omega Y}). \tag{3}$$

The AKSVD (Rubinstein et al., 2013) and the AOL (Yaghoobi et al., 2013) algorithms are based on the analysis model (2) or (3).

Recently, another analysis model for ADL, the so-called transform model, has been introduced (Ravishankar & Bresler, 2012), where $\mathbf{X} \in \mathbb{R}^{r\times n}$ is introduced as the analysis representation matrix, together with an approximation error $\|\mathbf{\Omega Y}-\mathbf{X}\|_F^2$. The ADL problem based on this model is given as follows:

$$\min_{\mathbf{\Omega},\mathbf{X}} \frac{1}{2}\|\mathbf{\Omega Y}-\mathbf{X}\|_F^2 + \lambda G(\mathbf{X}). \tag{4}$$

In general, ADL in the transform model consists of two stages: i.e., the analysis sparse-coding (finding the sparsest $\mathbf{X}$) and the analysis dictionary-update (learning a better $\mathbf{\Omega}$) stages. The transform model can be viewed as a generalization of the analysis model with sparse $\mathbf{\Omega Y}$ that lies in the range space of $\mathbf{\Omega}$. The transform model will permit a solution to $\mathbf{X}$ that does not necessarily lie in the range space of $\mathbf{\Omega}$. This generalization allows the transform model to include a wider class of signals within its ambit than either the analysis model (1) or the noisy analysis models (2) and (3). In contrast to synthesis dictionary learning (Aharon et al., 2006; Engan, Aase, & Husoy, 1999) and the analysis models (1), (2), and (3), the transform model (4) allows $\mathbf{\Omega Y}$ to be approximated by $\mathbf{X}$, which makes the learning easier and faster in practice. Here, the analysis sparse coding is cheap and can be obtained exactly by thresholding the product of $\mathbf{\Omega Y}$ and the sparsity measure on $\mathbf{X}$. Unlike synthesis dictionary learning, the ADL in the model of (4) avoids including a highly nonconvex function involving the product of two unknown matrices. ASimCO (Dong et al., 2015), TKSVD (Eksioglu & Bayir, 2014), LOST (Ravishankar & Bresler, 2013), OIHT-ADL (Zhang et al., 2014) and PADL (Li et al., 2017) are based on the transform model (4). These algorithms have confirmed that using the transform model can avoid the expensive sparse-coding step in ADL and therefore could provide comparable dictionary-learning performance at a much-reduced cost.

Unfortunately, unconstrained minimizations of (1), (2), (3), and (4) have trivial solutions. Indeed, if no constraints are imposed on $\mathbf{\Omega}$, it is easy to see that the trivial solution $\mathbf{\Omega} = \mathbf{0}$! is the global minimizer of (1), (2), (3), and (4). A possible way to exclude the trivial solutions is to impose additional constraints on $\mathbf{\Omega}$. The following constraints have been investigated in Dong et al. (2015), Hawe et al. (2013) and Ravishankar and Bresler (2013):

(i) Row-norm constraints: $\|\mathbf{w}^i\|_2 = c,\ \forall i \in [1, r]$;
(ii) Full-column rank constraints: $rk(\mathbf{\Omega}) = m$;
(iii) No linearly dependent rows in $\mathbf{\Omega}$: $\mathbf{w}^i \neq \pm\mathbf{w}^j,\ i \neq j$.

As indicated in Yaghoobi et al. (2013), these constraints when imposed individually do not give satisfactory results. Therefore, it was proposed in Yaghoobi et al. (2013) to use an orthonormality constraint, $\mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}$, combined with the row-norm constraint, $\|\mathbf{w}^i\|_2 = c,\ \forall i$. It has been shown that such an analysis dictionary can avoid degenerate solutions well and moreover has low mutual coherence between the dictionary atoms.

4. Proposed algorithm

In this section, we first formulate a novel ADL problem in which the $\ell_{1/2}$ norm is employed for strong sparsity, and orthonormality and row norm constraints are imposed on the dictionary. Then, we develop the algorithm based on the alternating update scheme (Li et al., 2017, 2015), which has two stages: the analysis sparse-coding and the analysis dictionary-update stages. In the analysis sparse-coding stage, we optimize the nonconvex $\ell_{1/2}$ norm regularized problem by solving a number of one-dimensional minimization problems. In the analysis dictionary-update stage, we propose manifold optimization to compute the dictionary on the manifold while satisfying the orthonormality constraint. The overall algorithm is then summarized. In addition, setting the adaptive regularization parameters is discussed.

4.1. Learning goal of ADL

We first describe our ADL problem based on the transform model (4), where $G(\mathbf{X})$ is the $\ell_{1/2}$ norm over the analysis representation $\mathbf{X} \in \mathbb{R}^{r\times n}$ as sparsity regularizer and the orthonormality constraint; i.e., $\mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}$ is imposed on the dictionary $\mathbf{\Omega} \in \mathbb{R}^{r\times m}$. The row norm constraint $\|\mathbf{w}^i\|_2 = c,\ \forall i \in [1, r]$ is also imposed, where $c = \sqrt{\frac{m}{r}}$. Our ADL problem can then be formulated as the minimization problem:

$$\begin{aligned} &\min_{\mathbf{\Omega},\mathbf{X}} \phi(\mathbf{\Omega}, \mathbf{X}) \\ =&\frac{1}{2}\|\mathbf{\Omega Y}-\mathbf{X}\|_F^2 + \lambda \|\mathbf{X}\|_{1/2}^{1/2} \\ s.t.\quad &\mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}, \quad \|\mathbf{w}^i\|_2 = c, \quad \forall i = 1, \ldots, r, \end{aligned} \tag{5}$$

where $\lambda > 0$ is a regularization parameter that controls the sparsity of $\mathbf{X}$. Our algorithm for solving (5) updates $\mathbf{X}$ in the analysis sparse-coding stage, and Ω in the analysis dictionary-update stage, alternatively.

4.2. Analysis sparse coding

The purpose of the analysis sparse stage is to obtain the analysis representations $\mathbf{X}$ of the observed signals $\mathbf{Y}$ based on a given dictionary Ω. Hence, we solve (5) with a fixed Ω:

$$\min_{\mathbf{X}} \phi(\mathbf{X}) = \frac{1}{2}\|\Omega\mathbf{Y} - \mathbf{X}\|_F^2 + \lambda \|\mathbf{X}\|_{1/2}^{1/2}. \tag{6}$$

Consider that (6) is separable according to the columns of $\mathbf{X}$; that is:

$$\sum_{l=1}^{n} \min_{\mathbf{x}_l} \phi(\mathbf{x}_l) = \frac{1}{2}\|\Omega\mathbf{y}_l - \mathbf{x}_l\|_F^2 + \lambda_l \|\mathbf{x}_l\|_{1/2}^{1/2}, \tag{7}$$

where $\mathbf{y}_l$ and $\mathbf{x}_l$ are the lth columns of $\mathbf{Y}$ and $\mathbf{X}$, respectively. Inspired by the proximal operator for ℓ_1 norm regularization (Li et al., 2017), we try to derive the closed-form solution of the problem (7) with respect to $\mathbf{x}_l$, which can be transformed as follows:

$$\min_{x_{jl}} \sum_j \frac{1}{2}(x_{jl} - [\Omega\mathbf{y}_l]_j)^2 + \lambda|x_{jl}|^{1/2} \quad \forall l \in [1, n], \tag{8}$$

where $[\cdot]_j$ is the jth entry of the vector $\Omega\mathbf{y}_l$. The summation is separable so that the problem (8) can be solved by minimizing a number of one-dimensional problems.

Considering $x_{jl} \neq 0$, we take the derivative of (8) to obtain the optimal solution. Because $\nabla(|x_{jl}|^{1/2}) = \frac{\text{sign}(x_{jl})}{2\sqrt{|x_{jl}|}}$, we have

$$x_{jl} - [\Omega\mathbf{y}_l]_j + \lambda\frac{\text{sign}(x_{jl})}{2\sqrt{|x_{jl}|}} = 0, \tag{9}$$

for any fixed j, the solution x_{jl}^* satisfies $x_{jl}^*[\Omega\mathbf{y}_l]_j > 0$.

When $x_{jl} > 0$, we denote $\sqrt{|x_{jl}|} = z$ and have $x_{jl} = z^2$, and then Eq. (9) can be written as follows:

$$z^3 - [\Omega\mathbf{y}_l]_j z + \frac{\lambda}{2} = 0, \tag{10}$$

Because (10) is a cubic equation, we can obtain the solutions by the Cardano formula (Xing, 2003). Denoting $r = \sqrt{|[\Omega\mathbf{y}_l]_j|/3}$, $p = -[\Omega\mathbf{y}_l]_j$ and $q = \lambda/2$. When $\frac{q^2}{4} + \frac{p^3}{27} < 0$—i.e., $[\Omega\mathbf{y}_l]_j > \frac{3}{4}(2\lambda)^{2/3}$—Eq. (10) has three roots. Denoting $\varphi_\lambda = \arccos(\frac{q}{2r^3})$, the three roots of (10) can be given as $z_1 = -2r\cos(\varphi/3)$, $z_2 = 2r\cos(\pi/3+\varphi/3)$, and $z_3 = 2r\cos(\pi/3 - \varphi/3)$. Because $x_{jl} > 0$, z_1 is not the solution of (10). We further check z_2 and z_3: $z_3 > z_2$, and thus z_3 is the unique solution of (10). Therefore, in this case, (10) has a unique solution that is given as $x_{jl} = \frac{2}{3}[\Omega\mathbf{y}_l]_j\big(1 + \cos\big(\frac{2\pi}{3} - \frac{2\varphi_\lambda([\Omega\mathbf{y}_l]_j)}{3}\big)\big)$ if $[\Omega\mathbf{y}_l]_j > \frac{3}{4}(2\lambda)^{2/3}$, where $\varphi_\lambda([\Omega\mathbf{y}_l]_j) = \arccos\big(\frac{\lambda}{4}\big(\frac{|[\Omega\mathbf{y}_l]_j|}{3}\big)^{-\frac{3}{2}}\big)$.

When $x_{jl} < 0$, we denote $\sqrt{|x_{jl}|} = z$ and have $x_{jl} = -z^2$, and then Eq. (9) can be written as follows:

$$z^3 + [\Omega\mathbf{y}_l]_j z + \frac{\lambda}{2} = 0. \tag{11}$$

Similarly, when $x_{jl} < 0$, we obtain the unique solution written as $x_{jl} = -\frac{2}{3}[\Omega\mathbf{y}_l]_j\big(1+\cos\big(\frac{2\pi}{3} - \frac{2\varphi_\lambda([\Omega\mathbf{y}_l]_j)}{3}\big)\big)$, when $[\Omega\mathbf{y}_l]_j < -\frac{3}{4}(2\lambda)^{2/3}$.

Based on the above two cases, we thus conclude that $x_{jl} = 0$ if $-\frac{3}{4}(2\lambda)^{2/3} \leq [\Omega\mathbf{y}_l]_j \leq \frac{3}{4}(2\lambda)^{2/3}$.

Therefore, we summarize all the solutions of one-dimensional problems with respect to x_{jl} and thus obtain the solution $\mathbf{x}_l^*$ of (7) in closed form:

$$x_{jl}^* = \begin{cases} f_{\lambda,1/2}([\Omega\mathbf{y}_l]_j), & |\,[\Omega\mathbf{y}_l]_j\,| > y_l^*; \\ 0 \quad , & \text{otherwise}, \end{cases} \tag{12}$$

where $f_{\lambda,1/2}([\Omega\mathbf{y}_l]_j) = \frac{2}{3}\text{sign}([\Omega\mathbf{y}_l]_j)\,|\,[\Omega\mathbf{y}_l]_j\,|\,\big(1 + \cos\big(\frac{2\pi}{3} - \frac{2\varphi_\lambda([\Omega\mathbf{y}_l]_j)}{3}\big)\big)$ and $\varphi_\lambda([\Omega\mathbf{y}_l]_j) = \arccos\big(\frac{\lambda}{4}\big(\frac{|[\Omega\mathbf{y}_l]_j|}{3}\big)^{-\frac{3}{2}}\big)$. $y_l^* = \frac{3}{4}(2\lambda)^{2/3}$ is called the thresholding value. An exact solution $\mathbf{X}$ of (6) can therefore be obtained in closed form via Eq. (12) for all $l = 1, \ldots, n$.

We find that the exact representation $\mathbf{X}$ in the transform model can be calculated directly by simply multiplying the signals $\mathbf{Y}$ by the dictionary Ω, while the corresponding problem for the synthesis model and for the analysis model (1) or (2) cannot. That is why we apply the transform model to ADL with the $\ell_{1/2}$-norm regularizer.

4.3. Analysis dictionary update

The analysis dictionary update stage aims to update Ω. We solve (5) with fixed $\mathbf{X}$:

$$\min_{\Omega\in\mathbb{R}^{r\times m}} \phi(\Omega) = \frac{1}{2}\|\Omega Y - \mathbf{X}\|_F^2 \tag{13}$$
$$s.t. \quad \Omega^T\Omega = \mathbf{I}, \quad \|\mathbf{w}^i\|_2 = c, \quad \forall i = 1, \ldots, r,$$

where $\phi(\Omega)$ is a differentiable function.

For the analysis dictionary with an orthonormality constraint, conventional ADL methods learn the analysis dictionary in the Euclidean space and then retrace the updated dictionary back to satisfy the orthonormality constraint. If the intrinsic geometric structure is required to be incorporated in the model, the dictionary in the Euclidean space becomes inappropriate and inadequate. To optimize the problem of the analysis dictionary with an orthonormality constraint efficiently, we conduct our optimization on the manifold constructed by the orthonormality constraint, so that the analysis dictionary can be effectively computed on the manifold, not only avoiding the trivial solutions but also keeping the intrinsic properties of the data.

We begin with a brief introduction to manifold optimization. Consider an optimization problem in which the variable $\mathbf{U}$ is restricted to the Riemannian manifold $\mathcal{S}$:

$$\min_{\mathbf{U}\in\mathcal{S}} f(\mathbf{U}), \tag{14}$$

where f is a smooth real-valued function, $\mathbf{U}$ is a real matrix, and $\mathcal{S}$ is some Riemannian submanifold of $\mathbb{R}$. The manifold is not a vector space and has no global system of coordinates; however, locally at the point $\mathbf{U}$, the manifold is homeomorphic to a Euclidean space referred to as the tangent space. The main idea of manifold optimization is to treat the objective as a function $f : \mathcal{S} \to \mathbb{R}$ defined on the manifold and perform descent on the manifold itself rather than in Euclidean space. The term "manifold optimization" is applied to algorithms that can preserve manifold constraints well during iterations.

The feasible set of the orthonormality constraint $\mathcal{S}_r^m := \{\Omega \in \mathbb{R}^{r\times m} : \Omega^T\Omega = \mathbf{I}\}$ is a Riemannian manifold, known as the orthonormality manifold. There are various optimization methods in the orthonormality manifold, most of which rely on generating points along geodesics of $\mathcal{S}_r^m$ or matrix reorthogonalization (Absil, Mahony, & Sepulchre, 2008). The former method must compute matrix exponentials or solve partial differential equations, and it is difficult to compute points on geodesics of $\mathcal{S}_r^m$. The latter requires matrix factorizations such as SVD, which are costly.

In this paper, a curvilinear search approach (Gong, Tao, Liu, Liu, & Yang, 2017), based on the Cayley transform (Wen & Yin, 2010)

with an appropriate step size, is applied to the analysis dictionary-update stage to compute the dictionary on the manifold $\mathcal{S}_r^m$:

$$\min_{\mathbf{\Omega}\in\mathcal{S}_r^m} \phi(\mathbf{\Omega}) = \frac{1}{2}\|\mathbf{\Omega}\mathbf{Y} - \mathbf{X}\|_F^2 \tag{15}$$
$$s.t. \quad \|\mathbf{w}^i\|_2 = c, \quad \forall i = 1, \ldots, r.$$

We wish to obtain a curve point on $\mathcal{S}_r^m$ by the Cayley transform as follows.

Given a feasible point $\mathbf{\Omega}$ and the gradient $\mathbf{G} := \frac{\partial\phi(\mathbf{\Omega})}{\partial\mathbf{\Omega}}$, a skew-symmetric matrix $\mathbf{A}$ is defined as follows:

$$\mathbf{A} := \mathbf{G}\mathbf{\Omega}^T - \mathbf{\Omega}\mathbf{G}^T. \tag{16}$$

A natural idea is to compute the next iteration as $\mathbf{\Omega} = \mathbf{\Omega} - \tau\nabla\phi = \mathbf{\Omega} - \tau\mathbf{A}\mathbf{\Omega}$, where τ is a step size. The obstacle is that the new point $\mathbf{\Omega}$ may not satisfy $\mathcal{S}_r^m$. To obtain the next iteration satisfying $\mathcal{S}_r^m$, the term $\nabla\phi$ is modified and the new curve point is determined by Wen and Yin (2010) as follows:

$$\mathbf{\Omega}(\tau) := \mathbf{\Omega} - \frac{\tau}{2}\mathbf{A}(\mathbf{\Omega} + \mathbf{\Omega}(\tau)). \tag{17}$$

Then, $\mathbf{\Omega}(\tau)$ can be expressed in closed form as follows:

$$\mathbf{\Omega}(\tau) := \mathbf{Q}\mathbf{\Omega} \quad \text{and} \quad \mathbf{Q} := (\mathbf{I} + \frac{\tau}{2}\mathbf{A})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{A}). \tag{18}$$

This is known as the Cayley transform (Wen & Yin, 2010), which serves as a generalization of the gradient descent updating rule that follows the manifold. $\mathbf{Q}$ leads to several properties, such as $\mathbf{\Omega}(\tau)$ being smooth in τ, and $\mathbf{\Omega}(0) = \mathbf{\Omega}$. More importantly, $\mathbf{\Omega}^T(\tau)\mathbf{\Omega}(\tau) = \mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}$. Because $\mathbf{A}$ is a skew-symmetric matrix,

$\frac{\tau}{2}\mathbf{A}$ is also a skew-symmetric matrix.

Moreover, $\mathbf{\Omega}^T(\mathbf{I} + \frac{\tau}{2}\mathbf{A})\mathbf{\Omega} = \|\mathbf{\Omega}\|_2^2$ holds for any nonzero $\mathbf{\Omega} \in \mathbb{R}^{r\times m}$. Therefore, $(\mathbf{I} + \frac{\tau}{2}\mathbf{A})$ is invertible. Given that $(\mathbf{I}+\mathbf{B})(\mathbf{I}-\mathbf{B}) = (\mathbf{I}-\mathbf{B})(\mathbf{I}+\mathbf{B})$ for any matrix $\mathbf{B}$, we can obtain

$$\begin{aligned}
&\mathbf{\Omega}^T(\tau)\mathbf{\Omega}(\tau)\\
=&\mathbf{\Omega}^T(\mathbf{I} - \frac{\tau}{2}\mathbf{A})^T(\mathbf{I} + \frac{\tau}{2}\mathbf{A})^{(-1)T}(\mathbf{I} + \frac{\tau}{2}\mathbf{A})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{A})\mathbf{\Omega}\\
=&\mathbf{\Omega}^T(\mathbf{I} - \frac{\tau}{2}\mathbf{A})^{-1}(\mathbf{I} - \frac{\tau}{2}\mathbf{A})(\mathbf{I} + \frac{\tau}{2}\mathbf{A})(\mathbf{I} + \frac{\tau}{2}\mathbf{A})^{-1}\mathbf{\Omega}\\
=&\mathbf{\Omega}^T\mathbf{\Omega} = \mathbf{I}.
\end{aligned} \tag{19}$$

Therefore, the curve $\mathbf{\Omega}(\tau)$ always satisfies the orthonormality constraint. Eq. (18) requires inverting $(\mathbf{I} + \frac{\tau}{2}\mathbf{A})$ to preserve the orthonormality constraint, which is cheaper than computing either SVD or geodesics. That is the advantage of using the Cayley transform.

The update scheme (18) is the steep-descent step on the manifold satisfying the orthonormality constraint. It is well known that the steepest-descent method with a fixed step size may not converge. However, by choosing the step size wisely, convergence can be guaranteed, and its speed can be accelerated without significantly increasing the cost at each iteration. Like line search along a straight line, curvilinear search can be applied to find a proper step size τ and guaranteeing that the iterations will converge to a stationary point. The well-known Barzilai–Borwein (BB) step size (Goldfarb, Wen, & Yin, 2009) is often employed to accelerate the gradient method without extra cost. Hence, the BB step size is applied to the curvilinear search method to reduce the number of steepest-descent iterations, so that the global convergence of (15) can be guaranteed with faster speed. We can describe the curvilinear line search with BB step size as follows.

A new point is generated iteratively in the form of $\mathbf{Z}_{k+1} := \mathbf{\Omega}_k(\tau_k)$, which satisfies (Wen & Yin, 2010)[p]

$$\phi(\mathbf{\Omega}_k(\tau_k)) \le C_k + \rho_1\tau_k\phi'(\mathbf{\Omega}_k(0)), \tag{20}$$

Algorithm 1: MADL-$\ell_{1/2}$

Input: Data matrix $\mathbf{Y} \in \mathbb{R}^{m\times n}$.

1: Initialize $\mathbf{\Omega}_0 \in \mathbb{R}^{r\times m}$ and $\mathbf{X}_0 \in \mathbb{R}^{r\times n}$,

2: for $k = 1, 2, \ldots$ do

3: Adjust regularization parameter λ_l^k, $l = 1, \ldots, n$,

4: Compute the analysis representations $\mathbf{X}_k$ based on equation (12),

5: Update $\mathbf{\Omega}_k$ based on equation (23),

6: $\mathbf{w}^i = \begin{cases} \sqrt{\frac{m}{r}}\frac{\mathbf{w}^i}{\|\mathbf{w}^i\|_2}, & \|\mathbf{w}^i\|_2 \neq 0; \\ \mathbf{v}, & \text{otherwise}, \end{cases}$,

7: end for

output: $\mathbf{\Omega}$.

where $\phi'(\mathbf{\Omega})$ denotes the derivation of $\phi(\mathbf{\Omega})$ with respect to τ. $C_{k+1} = (\eta P_k C_k + f(\mathbf{Z}_{k+1}))/P_{k+1}$ and $P_{k+1} = \eta P_k + 1$, $P_0 = 1$. According to Wen and Yin (2010), we simply set $\tau_k = \tau_{k,1}\delta^h$ or $\tau_k = \tau_{k,2}\delta^h$, where h is the smallest integer. $\tau_{k,1}$ and $\tau_{k,2}$ are defined as follows:

$$\tau_{k,1} = \frac{\mathrm{tr}\big((\mathbf{Z}_k - \mathbf{Z}_{k-1})^T(\mathbf{Z}_k - \mathbf{Z}_{k-1})\big)}{|\mathrm{tr}\Big((\mathbf{Z}_k - \mathbf{Z}_{k-1})^T\big(\nabla\phi(\mathbf{Z}_k) - \nabla\phi(\mathbf{Z}_{k-1})\big)\Big)|}, \tag{21}$$

$$\tau_{k,2} = \frac{|\mathrm{tr}\big((\mathbf{Z}_k - \mathbf{Z}_{k-1})^T\big(\nabla\phi(\mathbf{Z}_k) - \nabla\phi(\mathbf{Z}_{k-1})\big)\big)|}{\mathrm{tr}\Big(\big(\nabla\phi(\mathbf{Z}_k) - \nabla\phi(\mathbf{Z}_{k-1})\big)^T\big(\nabla\phi(\mathbf{Z}_k) - \nabla\phi(\mathbf{Z}_{k-1})\big)\Big)}. \tag{22}$$

Note that ρ_1, δ and $\eta \in (0, 1)$.

Therefore, instead of τ in (18), $\mathbf{\Omega}(\tau_k)$ at iteration k with the BB step size is defined by $\mathbf{\Omega}$ and the skew-symmetric matrix $\mathbf{A}$ as follows:

$$\mathbf{\Omega}_k(\tau_k) = (\mathbf{I} + \frac{\tau_k}{2}\mathbf{A})^{-1}(\mathbf{I} - \frac{\tau_k}{2}\mathbf{A})\mathbf{\Omega}. \tag{23}$$

After updating $\mathbf{\Omega}$ by (23), we normalize the rows of $\mathbf{\Omega}$ to prevent scaling ambiguity and replace any zero rows by the normalized random vector $\mathbf{v}$ (Yaghoobi et al., 2013):

$$\mathbf{w}^i = \begin{cases} \sqrt{\frac{m}{r}}\frac{\mathbf{w}^i}{\|\mathbf{w}^i\|_2}, & \|\mathbf{w}^i\|_2 \neq 0; \\ \mathbf{v}, & \text{otherwise}. \end{cases} \tag{24}$$

4.4. The overall algorithm

The proposed algorithm MADL-$\ell_{1/2}$ for ADL, with the $\ell_{1/2}$ norm regularizer and based on manifold-optimization, alternates between the analysis sparse-coding and the analysis dictionary-update stages for a number of iterations. The structure of MADL-$\ell_{1/2}$ is outlined in Algorithm 1.

4.5. Adaptive regularization parameter setting

Because our proposed algorithm includes a regularizer, we need to determine the regularization parameter λ_l^k systemically and concisely. This parameter strategy is not only helpful in directing the algorithm toward more promising solutions but also convenient when applying the algorithm to applications. However, selecting appropriate values for regularization parameters is a very hard problem. In general, there is no optimal rule. Nevertheless, when some prior information (e.g., sparsity) is available, it is possible to set the regularization parameter value more reasonably and intelligently. In our dictionary learning problem, the sparseness

information, i.e., the number of nonzero elements, is known as *a priori* or can be obtained easily as a rough estimate. The data sparsity information is integrated into MADL-$\ell_{1/2}$ to enable adjustment of the regularization parameter adaptively with each iteration of the algorithm.

The minimization of the objective function (7) with respect to the analysis vector $\mathbf{x}_l \in \mathbb{R}^r$ $(l = 1, \ldots, n)$ is assumed a s-sparse problem. We need to solve an $\ell_{1/2}$ norm regularizer problem that is restricted to the subregion $\Gamma_s = \{\mathbf{x}_l = (x_{1l}, x_{2l}, \ldots, x_{rl}) : supp(x) = s\}$ of $\mathbb{R}^r$, where x_{rl} is the rth entry of $\mathbf{x}_l$ and $supp(x)$ is the support set of x, i.e., $supp(x) = s : x_s \neq 0$. The solution $\mathbf{x}_l$ of (7) is given as (12), which has s nonzero elements. Each $\mathbf{x}_l$ $(l = 1, \ldots, n)$ may have its own regularization parameter value, and so we may use n independent parameter values for each iteration. This is denoted by the use of λ_l^k in the kth iteration. Then we integrate the sparsity information into Eq. (12) to adjust λ_l^k.

Assume that $|\ [\Omega\mathbf{y}_l]_1\ | \geq |\ [\Omega\mathbf{y}_l]_2\ | \geq \cdots \geq |\ [\Omega\mathbf{y}_l]_r\ |$. As $\mathbf{x}_l$ contains s nonzero elements, according to (12), we have

$$x_{il} \neq 0 \Leftrightarrow |\ [\Omega\mathbf{y}_l]_i\ | > \frac{3}{4}(2\lambda_l^k)^{2/3} \quad i \in 1, 2, \ldots, s,$$

$$x_{jl} = 0 \Leftrightarrow |[\Omega\mathbf{y}_l]_j| \leq \frac{3}{4}(2\lambda_l^k)^{2/3} \quad j \in s+1, s+2, \ldots, r. \tag{25}$$

Assume that $T_{s+1} = |\ [\Omega\mathbf{y}_l]_{s+1}\ |$. According to $|\ [\Omega\mathbf{y}_l]_{s+1}\ | = \frac{3}{4}(2\lambda_l^k)^{2/3}$, we have

$$\lambda_l^k = \sqrt{\frac{16}{27}} T_{s+1}^{\frac{3}{2}}. \tag{26}$$

In this way, each regularization parameter λ_l^k can be determined adaptively, which can evolve independently and can decrease monotonically with each iteration of the algorithm (Wang & Qian, 2015).

5. Simulations

In this section, we describe a series of experiments on synthetic data and real-world data. They are designed to evaluate the performance of the proposed algorithm MADL-$\ell_{1/2}$. The first experiment uses synthetic data generated from a known dictionary; i.e., the reference dictionary. We apply the ADL algorithms to the synthetic data to learn dictionaries that are then compared with the reference dictionary. In the second experiment, we conduct image classifications to verify the improvement of the learned dictionaries with the orthonormality constraint on the manifold. Finally, we investigate the effectiveness of the learned dictionary from MADL-$\ell_{1/2}$ in the practical task of image denoising.

For the experiments, all programs were coded in Matlab and were run within Matlab (R2016a) on a PC with a 3.0 GHz Intel Core i5 CPU and 16 GB of memory, under the Microsoft Windows 7 operating system.

5.1. Experiments on synthetic data

Evaluating the performance of dictionary learning algorithms is a rather difficult task unless the reference dictionary is known. This is why for this experiment, we focus on synthetic data built from a known dictionary. The ADL algorithms are applied to learn the dictionaries, which are then compared with the reference dictionary, so the great ability to recover the dictionary can demonstrate the effectiveness of the algorithm.

(1) Experimental setup: First, we generated the reference dictionary $\mathbf{\Omega} \in \mathbb{R}^{24\times16}$. Synthetic data for the sample set $\mathbf{Y} \in \mathbb{R}^{16\times n}$ comprising signals $\mathbf{y}_i$ $(i = 1 \ldots n)$ were obtained from the reference dictionary $\mathbf{\Omega}$. Each signal $\mathbf{y}_i$ generated had cosparsity l with respect to the reference dictionary $\mathbf{\Omega}$. This was realized by picking l rows randomly from $\mathbf{\Omega}$ and then generating a vector, again randomly, from the orthogonal complement space of these chosen rows. The initial dictionary for the algorithms was generated by normalizing a random matrix of size 24×16.

(2) Performance criteria: To compare the performances of the ADL algorithms, we measured the performances in three ways.

The first criterion was the recovery ratio for the reference dictionary rows. The learned dictionary $\hat{\mathbf{\Omega}}$ was compared with the reference dictionary $\mathbf{\Omega}$. If the maximum ℓ_2 distance between any recovered atom (row) $\hat{\mathbf{w}}^i$ and the closest atom $\mathbf{w}^j$ was less than 0.01, it was considered to be a successful recovery of one atom, where $\hat{\mathbf{w}}^i$ and $\mathbf{w}^j$ are the atoms of $\hat{\mathbf{\Omega}}$ and $\mathbf{\Omega}$, respectively.

We used the recovery cosparsity of the original signals $\mathbf{Y}$ with respect to the learned dictionary $\hat{\mathbf{\Omega}}$ as the second performance measure (Dong et al., 2015). Because the aim of ADL is to acquire an analysis dictionary for which the analysis representations of the signals are sparse, the recovery cosparsity of the original signals is also significant in the experiment. $\|\mathbf{b}\|_0^\epsilon$ was introduced to count the number of elements in $\mathbf{b} \in \mathbb{R}^n$. It was defined as $\|\mathbf{b}\|_0^\epsilon = card(\{i :|\ b_i\ | < \epsilon, i \in \forall n\})$, where b_i was the ith element of $\mathbf{b}$ and ϵ was set to 0.001. The cosparsity of the signal could be obtained by applying $\|\mathbf{b}\|_0^\epsilon$ to the product of the learned dictionary $\hat{\mathbf{\Omega}}$ and the original signal $\mathbf{Y}$. More zeros – i.e., higher cosparsity – indicate a sparser matrix.

In addition, we used Hoyer's measure (Hoyer, 2004) as the sparsity measure, because it is absolute and has values in [0, 1] for different sparsities monotonically; therefore it could be controlled explicitly. It is defined as follows:

$$\text{Sparsity}(\mathbf{x}) = \frac{\sqrt{n} - \|\mathbf{x}\|_1/\|\mathbf{x}\|_2}{\sqrt{n} - 1} \in [0, 1]. \tag{27}$$

A larger Hoyer's sparsity value means that the vector $\mathbf{x}$ is sparser.

(3) Convergence of the proposed algorithm: Different values of n, the number of signal samples, were used to demonstrate the convergence behavior of the proposed MADL-$\ell_{1/2}$ algorithm. n was selected as 768 and 3072. The cosparsity was set to $l = 12$. We executed the algorithm several times to identify a reasonable number of iterations that would ensure the convergence of the algorithm. The objective values, the recovery ratios of the learned dictionary, the recovery cosparsities, and Hoyer's sparsity are shown in Fig. 1(a), (b), (c), and (d), respectively. For each point in the plots, the results were averaged over 30 independent trials. The objective function decreases monotonically and converges to a stable value within a reasonable number of iterations, as shown in Fig. 1(a). The convergence rates of the recovery ratios of the learned dictionary, the recovery cosparsities, and Hoyer's sparsity versus iteration number are consistent with the convergence rates of the objective value shown in Fig. 1.

(4) Exact recovery of the analysis dictionary: We compared our proposed algorithm MADL-$\ell_{1/2}$ with five state-of-the-art algorithms: AKSVD (Rubinstein et al., 2013), TKSVD (Eksioglu & Bayir, 2014), ASimCO (Dong et al., 2015), LOST (Ravishankar & Bresler, 2013), and AOL (Yaghoobi et al., 2013). MADL-$\ell_{1/2}$ and AOL (Yaghoobi et al., 2013) imposed the orthonormality constraint on the dictionary while the other algorithms did not, so that the reference dictionaries $\mathbf{\Omega}$ for the ADL algorithms with the orthonormality constraint and without the orthonormality constraint were different. The reference dictionary $\mathbf{\Omega}$ for the ADL algorithms with the orthonormality constraint was generated by projecting a random $\mathbf{\Omega}_0 \in \mathbb{R}^{24\times16}$ repeatedly onto the uniform normalized set and tight frame set (Yaghoobi et al., 2013). The $\mathbf{\Omega}_0$ matrix has an independent and identically distributed (IID) Gaussian distribution with zero mean and unit variance, while the reference dictionary $\mathbf{\Omega}$ for the ADL algorithms without the orthonormality constraint was generated by normalizing a random matrix with an IID Gaussian distribution with zero mean and unit variance. The initialized dictionaries for all the algorithms were generated from a random

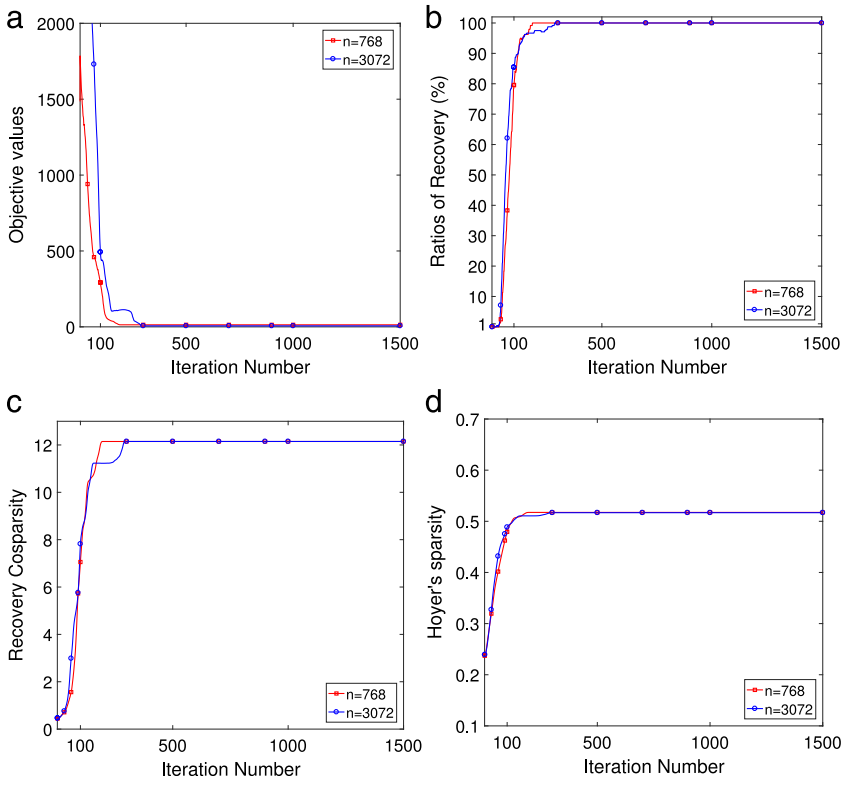

Fig. 1. (a) Objective values versus iteration number. (b) Average recovery ratios for the learned dictionaries versus iteration number. (c) Average recovery cosparsity versus iteration number. (d) Hoyer's sparsity versus iteration number.

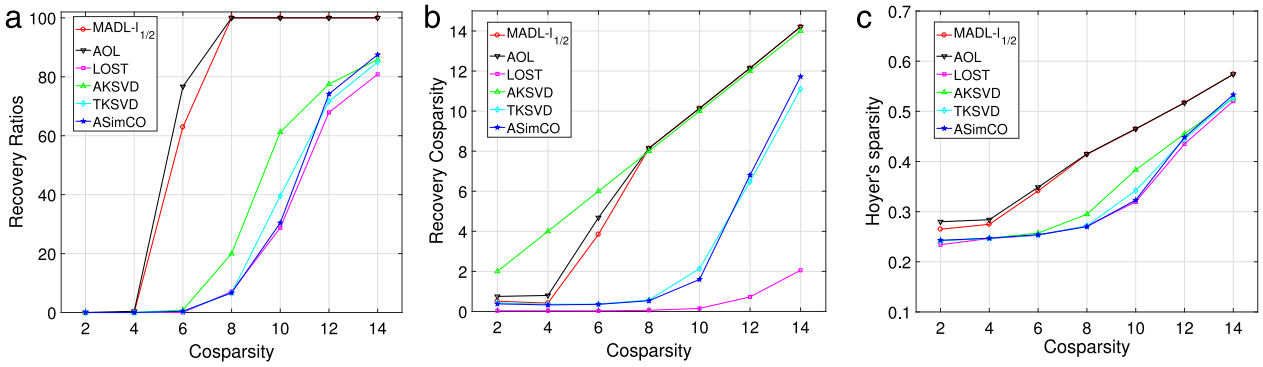

Fig. 2. $n = 768$: (a) Average recovery ratios for the learned dictionaries versus cosparsity for the various algorithms. (b) Average recovery cosparsity versus cosparsity for the various algorithms. (c) Average Hoyer's sparsity versus cosparsity for the various algorithms.

matrix. The algorithms were tested with different values for the cosparsity l, which was varied in the range $l = 2, \ldots, 14$. We also investigated the role of n in the dictionary recovery process via some simulations. We chose the numbers of samples as $n = 768$ and $n = 3072$. The regularization parameters of MADL-$\ell_{1/2}$ are given by Eq. (26). In AKSVD, TKSVD, ASimCO, and AOL, there are no parameters to be adjusted for these algorithms without considering the coherence between the dictionary rows. Hence, the settings for these algorithms were the same as those used in the experiments with synthetic data, as described in Dong et al. (2015), Eksioglu and Bayir (2014), Rubinstein et al. (2013) and Yaghoobi et al. (2013). In LOST (Ravishankar & Bresler, 2013), two penalty parameters, the parameter λ of the full column-rank penalty term and the parameter η of the correlation penalty term, required adjustments. In addition, the experiments in LOST were conducted with image patches that were different from the synthetic data. Thus, λ was set to 50 and η was set to 20, so that a better dictionary for the synthetic data could be learned. The step size for the inner gradient conjugate method of LOST was 10^{-4}. We executed the algorithms as many times as possible and determined the reasonable number of iterations that would ensure that the objective function of each algorithm converged to a stable value. In the experiments, we performed 500 iterations for AKSVD, 1000 iterations for TKSVD, 5000 iterations for ASimCO, LOST, and MADL-$\ell_{1/2}$, and 50,000 iterations for AOL, which were determined by the computational complexity and convergence of the algorithms.

The main performance indexes were the recovery ratios for the learned dictionaries, the recovery cosparsity, and Hoyer's sparsity versus the cosparsity l. The results were averaged over 30 independent trials for each point in the plots. Figs. 2 and 3 show the experimental results for different algorithms for $n = 768$ and $n = 3072$, respectively. Five observations can be made: (1) Better results, including the recovery ratios of the learned dictionaries, the recovery cosparsity, and Hoyer's sparsity, were obtained with the algorithms with larger cosparsities. (2) Better results were obtained with more signal samples. (3) The performances of our

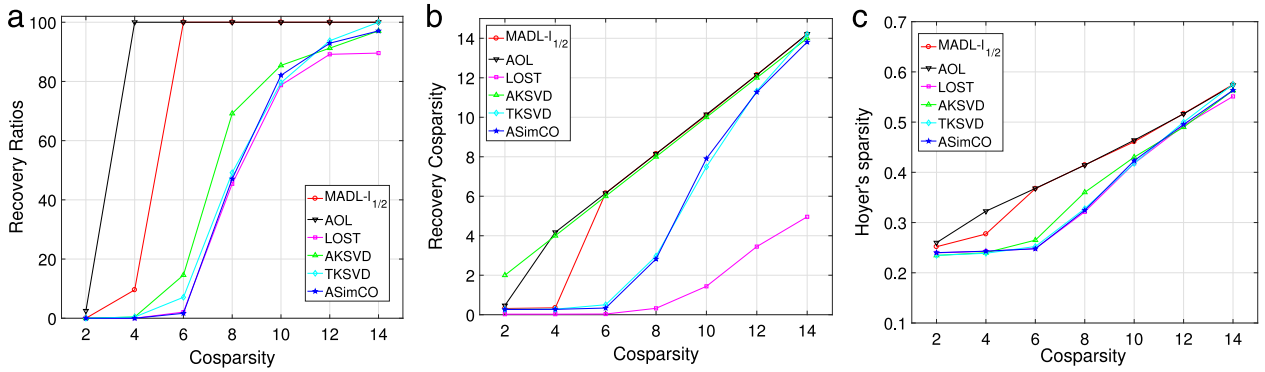

Fig. 3. $n = 3072$: (a) Average recovery ratios for the learned dictionaries versus cosparsity for the various algorithms. (b) Average recovery cosparsity versus cosparsity for the various algorithms. (c) Average Hoyer's sparsity versus cosparsity for the various algorithms.

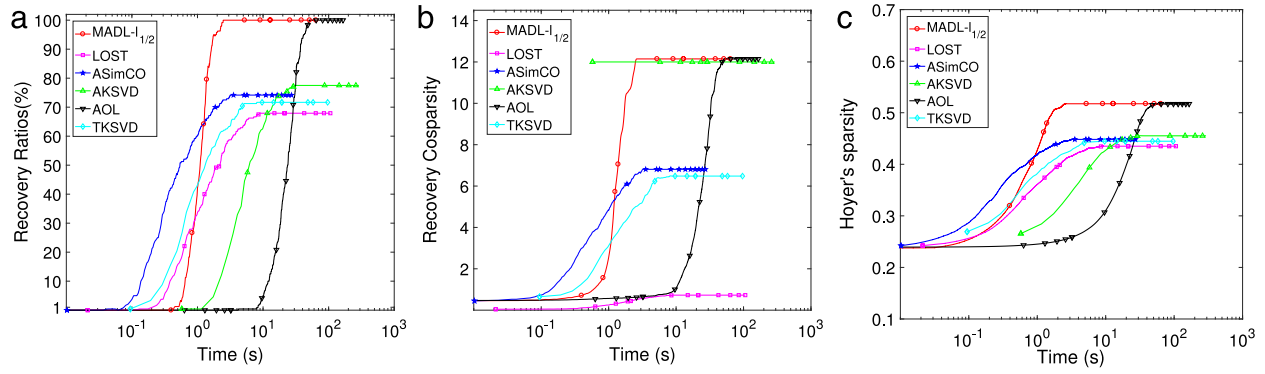

Fig. 4. $n = 768$ and $l = 12$: (a) Average recovery ratios for the learned dictionaries versus running time for the various algorithms. (b) Average recovery cosparsity versus running time for the various algorithms. (c) Average Hoyer's sparsity versus running time for the various algorithms.

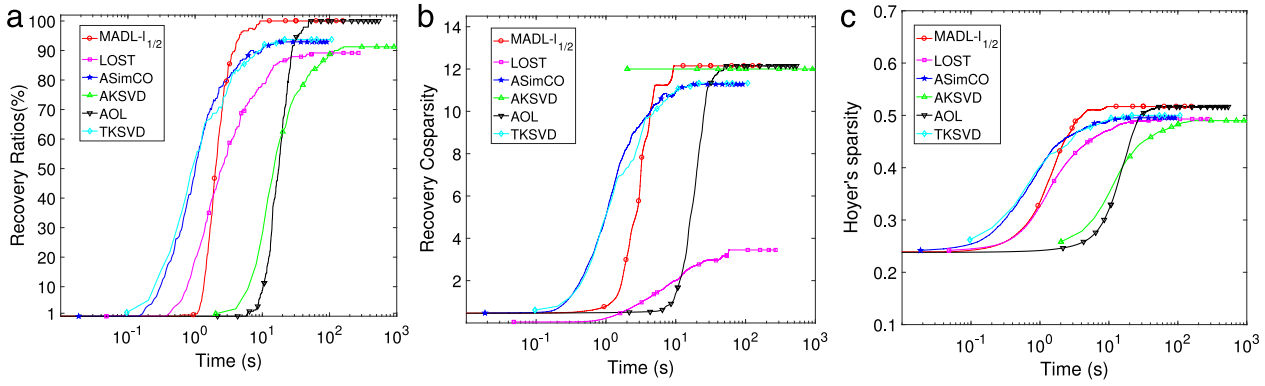

Fig. 5. $n = 3072$ and $l = 12$: (a) Average recovery ratios for the learned dictionaries versus running time for the various algorithms. (b) Average recovery cosparsity versus running time for the various algorithms. (c) Average Hoyer's sparsity versus running time for the various algorithms.

MADL-$\ell_{1/2}$ algorithm and AOL with the orthonormality constraint were significantly better than those of the other algorithms without the orthonormality constraint. Moreover, MADL-$\ell_{1/2}$ and AOL were still effective even for low cosparsities, while other algorithms degraded greatly. (4) Our MADL-$\ell_{1/2}$ algorithm significantly outperformed AKSVD, TKSVD, LOST, and ASimCO in terms of the recovery ratios, the recovery cosparsity, and Hoyer's sparsity. Note that AKSVD used the backward greedy algorithm, which maintained the recovery cosparsity at l throughout the iterations, as shown in Figs. 2(b) and 3 (b). That is why we employed Hoyer's sparsity as another sparsity measure. We can also see that Hoyer's sparsity for AKSVD was worse than that for MADL-$\ell_{1/2}$. (5) Our MADL-$\ell_{1/2}$ algorithm was slightly worse than AOL for low cosparsity. This might be because AOL is based on the analysis model in which $\|\mathbf{\Omega Y}\|_1$ is directly minimized, but its convergence is slower than that of MADL-$\ell_{1/2}$ based on the transform model, which can be verified by the next experimental results in Figs. 4 and 5.

It is clear that the algorithms converge differently while reaching stable values for the dictionary ratios, the recovery cosparsity, and Hoyer's sparsity. We retained the experimental settings from the above experiment and selected the case of $l = 12$ to visualize the convergence speeds for the algorithms. Note that the different algorithms require different numbers of iterations to converge, and the running times per iteration for the various algorithms were also different. Therefore, to compare the algorithms fairly, we show the performance of the algorithms versus running time per iteration.

Figs. 4 and 5 show the average recovery ratios for the learned dictionaries, the recovery cosparsities, and Hoyer's sparsity versus

Table 1
Classification error rates with different levels of white Gaussian noise.

Noise level	5 dB	10 dB	15 dB	20 dB	No noise
Fixed D	0.4502	0.4085	0.3411	0.1688	0.1526
D 1	0.3682	0.2693	0.1853	0.1286	0.1190
D 2	0.3412	0.2584	0.1767	0.1227	0.1158
D 3	0.3307	0.2373	0.1660	0.1153	0.1043

running time per iteration for $n = 768$ and $n = 3072$, respectively. It can be seen that the running times to reach stable values for MADL-$\ell_{1/2}$ were shorter than those for the state-of-the-art algorithms. Note that the recovery ratios of the learned dictionaries, the recovery cosparsity, and Hoyer's sparsity are similar for MADL-$\ell_{1/2}$ and AOL; however, the running time of convergence for MADL-$\ell_{1/2}$ is much shorter than that for AOL.

5.2. Image classification

We next used image classification as a further evaluation of the performance of the learned dictionaries on the manifold. Our MADL-$\ell_{1/2}$ is an ADL algorithm with the orthonormality constraint, and the solution of the dictionary is obtained directly on the manifold. AOL is also an ADL algorithm with the orthonormality constraint, but the dictionary is approximated by an unconstrained optimization followed by projection on the constrained space.

The experiment was conducted on the database of USPS handwritten digits (USPS, 0000). The database contains 8-bit grayscale images of the digits 0 through 9 with a size of 16 × 16, and there are 1100 examples of each digit. Following the conclusion of Huang and Aviyente (2006), 10-fold stratified cross-validation was adopted. First, we learned the dictionaries with the size of 300 × 256 by MADL-$\ell_{1/2}$ and AOL from the USPS data. We also learned the dictionary by AOL without the orthonormality constraint to evaluate the performance of the orthonormality constraint. In addition, we also used a fixed dictionary that was a combination of Haar wavelet basis and Gabor basis. Then the learned dictionaries and the fixed dictionary were applied to image classification. The classification was conducted with the decomposition coefficients as features and a support vector machine (SVM) as a classifier. In this implementation, we compared the results of four dictionaries: D 1 obtained by AOL without the orthonormality constraint, D 2 obtained by projection approximation in AOL with the orthonormality constraint, D 3 obtained by the manifold optimization in MADL-$\ell_{1/2}$, and the fixed dictionary (Fixed D). The results of classification accuracy versus the levels of white Gaussian noise are given in Fig. 6, while Table 1 summarizes the classification error rates obtained with different levels of noise. It can be seen that (1) the classification accuracies of the three learned dictionaries are significantly better than those of the fixed dictionary, (2) the classification accuracies of the learned dictionary with the orthonormality constraint (D 2) are better than those of the learned dictionary without the orthonormality constraint (D 1), and (3) the classification accuracies of the learned dictionary obtained by manifold optimization (D 3) are better than those of the learned dictionary obtained by projection approximation (D 2). The results indicate that our proposed MADL-$\ell_{1/2}$ based on manifold optimization can obtain more accurate solutions than those obtained by projections in AOL.

We also show the mutual coherence of the dictionary, which is defined as follows:

$$\mu(\mathbf{\Omega}) = \max_{i \neq j} \mid \langle \Omega_i, \Omega_j \rangle \mid . \tag{28}$$

The matrices of the mutual coherence of the dictionaries learned from the USPS data are shown in Fig. 7. Colors closer to yellow in Fig. 7 indicate higher values of the mutual coherence. We can see that the color for AOL without the orthonormality constraint is closer to yellow than that for MADL-$\ell_{1/2}$ and the original AOL. Hence, the mutual coherence of the dictionaries from AOL without the orthonormality constraint is higher than that from our MADL-$\ell_{1/2}$ and the original AOL, which indicates that the orthonormality constraint imposed in MADL-$\ell_{1/2}$ and the original AOL can reduce the mutual coherence.

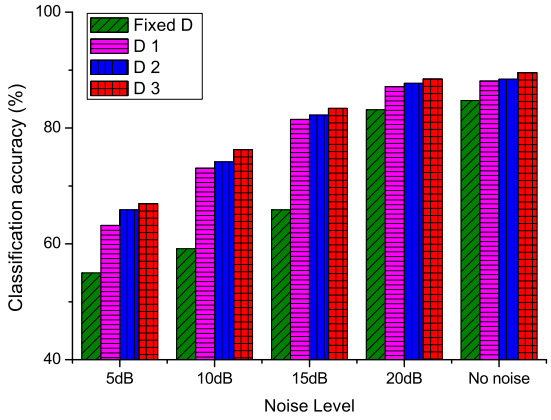

Fig. 6. Classification accuracy with different levels of noise.

5.3. Image denoising

We now describe the application of the proposed algorithm to image denoising by first learning an analysis dictionary and then using the learned dictionary to denoise the images.

(1) Image denoising framework: The image denoising framework involved two steps: ADL and image recovery. Small overlapping image patches of size 8 × 8 pixels were extracted from the noisy images and reshaped as column vectors that were concatenated as the training signal $\mathbf{Z} \in \mathbb{R}^{64\times n}$. The number of patches was n, and the patch overlap was set as seven. The algorithm was then applied to $\mathbf{Z}$ to learn the analysis dictionary $\mathbf{\Omega} \in \mathbb{R}^{128\times 64}$. For the image-recovery process, we used a method based on that in Yaghoobi et al. (2013), where the noiseless estimation $\mathbf{Y}$ could be obtained by solving the optimization problem $\min_{\mathbf{Y}} \frac{\alpha}{2}\|\mathbf{Z}-\mathbf{Y}\|_F^2 + \|\mathbf{\Omega Y}\|_1$. The final denoised image can be obtained by reshaping the columns of $\mathbf{Y}$ as image patches and averaging these overlapping patches. The core idea of this method is that image recovery can be formulated as a regularized optimization problem, where the learned analysis dictionary serves as a sparsity-promoting prior for the analysis representations of all patches. This method was selected for its high computational efficiency. Denoising was performed using a variety of settings for the regularization parameter α: {0.02, 0.025, 0.03, 0.035, 0.04, 0.045, 0.05, 0.1, 0.15, 0.2}. In general, α should be smaller when the noise level is higher.

In the image denoising experiment, we compared our MADL-$\ell_{1/2}$ algorithm with the algorithms used in the synthetic data experiments. The iteration numbers for AKSVD, AOL, and other algorithms were set to 200, 20,000, and 1000, respectively; these values were determined empirically from the computational time and convergence characteristics of the algorithms. For LOST, $\lambda = \eta = 4 \times 10^5$ and the step size was set to 10^{-9}. The cosparsity l was set to 40. The experimental results were averaged over 30 independent trials.

(2) Denoising performance evaluation: The noisy images were artificially contaminated by additive white Gaussian noise with standard deviation values of $\sigma = 8$ and 22.8, which were selected empirically to represent low and high levels of noise. We used the peak signal-to-noise ratio (PSNR) and the structural similarity

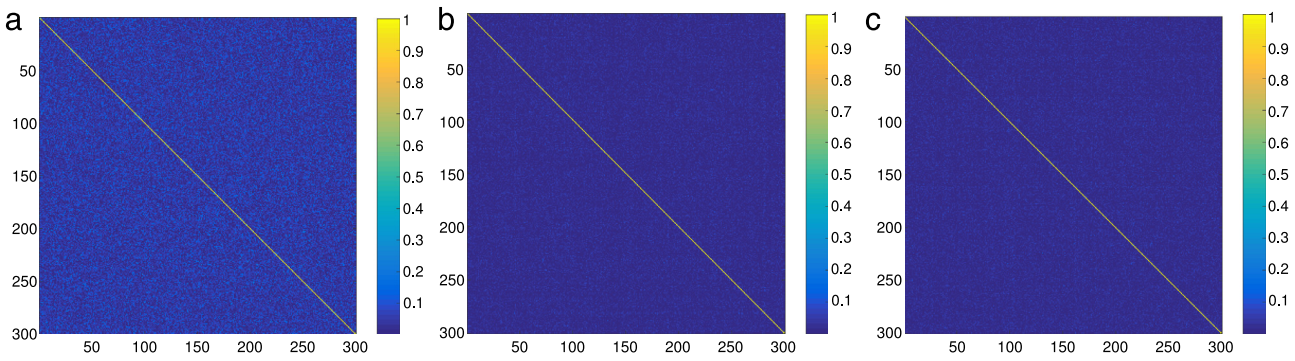

Fig. 7. The mutual coherence matrices of the learned dictionaries from USPS data using: (a) AOL without the orthonormality constraint, (b) AOL with the orthonormality constraint, and (c) MADL-$\ell_{1/2}$. The ith column and jth row element in each matrix represent the mutual coherence between the ith and jth atoms. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(SSIM) (Wang, Bovik, Sheikh, Sapiro, & Simoncelli, 2014) as quality measures to assess the denoised images. The values for SSIM ranged between 0 and 1, with the value being 1 if $\mathbf{Y} = \mathbf{Y}_0$, where $\mathbf{Y}$ was the denoised estimation and $\mathbf{Y}_0$ was the clean signal. The PSNR value of $\mathbf{Y}$ is defined as

$$\text{PSNR}(\mathbf{Y}) = 10 \cdot \log_{10} \frac{255^2}{\|\mathbf{Y} - \mathbf{Y}_0\|_F^2}. \tag{29}$$

(3) Natural image denoising: Standard test images including "Lena" (256 × 256), "Peppers" (256 × 256), and "House" (256 × 256) were used in this experiment. We generated a random dictionary $\mathbf{\Omega}_0 \in \mathbb{R}^{128\times 64}$ that was IID-Gaussian distributed with zero mean and unit variance in the initialization process. The number n of patches was selected as 20,000. The algorithms were then applied to $\mathbf{Z} \in \mathbb{R}^{64\times 20000}$ to learn the analysis dictionaries $\mathbf{\Omega} \in \mathbb{R}^{128\times 64}$.

We selected the best results obtained for various values of α. Among the experimental results, we found that, for $\alpha = 0.15$, the denoising performances of all algorithms were best when $\sigma = 8$. For $\alpha = 0.05$, they were best when $\sigma = 22.8$. The average PSNR and SSIM values obtained for the denoised images over 30 trials at different noise levels are summarized in Table 2. These results show that MADL-$\ell_{1/2}$ obtained better results than the state-of-the-art algorithms. In Fig. 8, we show the noisy images and denoised images by MADL-$\ell_{1/2}$.

Fig. 8. Noisy images (left); denoised images by MADL-$\ell_{1/2}$ (right).

6. Conclusion

In this paper, we have proposed a new and efficient algorithm for ADL, where the nonconvex $\ell_{1/2}$ norm regularizer is used for strong sparsity, and the manifold optimization is employed to update the dictionary. To solve the nonconvex $\ell_{1/2}$ norm regularized problem efficiently, we solve a number of one-dimensional minimization problems and then obtain the closed-form solutions directly in each iteration. We have also shown that adaptive adjustment of the regularization parameter in the proposed algorithm not only improves the algorithm performance but also suits real-data applications. To solve the orthonormality constraint on the dictionary efficiently, we compute the analysis dictionary on the manifold directly, leading to a more accurate solution than that obtained in Euclidean space while simultaneously avoiding the trivial solutions well.

The experiments on synthetic and real-world data have demonstrated the superior performance of the proposed algorithm: (1) the proposed algorithm can recover the analysis dictionary better and achieve greater Hoyer's sparsity than the state-of-the-arts algorithms can, while the proposed algorithm has shorter running times for convergence; (2) we have verified the improvement of the learned dictionaries on the manifold in image classification; and (3) the proposed algorithm performs well in image denoising. In future work, we plan to apply our proposed algorithm to dynamic texture classification to evaluate the performance of our method in this context.

Table 2
Peak signal-to-noise ratio (PSNR, dB) and structural similarity (SSIM) for the denoising results. The best result is highlighted in each cell.

σ/PSNR		Lena		Peppers		House	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
8/30.13	AKSVD	33.8132	0.8754	33.2523	0.8381	34.4011	0.8417
	TKSVD	32.9261	0.8581	33.1891	0.8400	33.5699	0.8020
	LOST	33.5470	0.8600	33.5618	0.8567	35.3819	0.9169
	ASimCO	34.0209	0.8918	34.1707	0.9247	34.4935	0.8369
	AOL	34.1017	0.9121	34.2107	0.9316	35.9412	0.9120
	MADL-$\ell_{1/2}$	**34.2298**	**0.9247**	**34.2735**	**0.9379**	**35.9975**	**0.9203**
22.8/20.75	AKSVD	27.5919	0.7170	27.1937	0.7101	29.6441	0.7600
	TKSVD	27.1051	0.6842	27.0001	0.6945	29.1065	0.7360
	LOST	27.5139	0.7158	27.7510	0.7200	28.6731	0.65140
	ASimCO	28.0247	0.7315	27.8791	0.7249	29.7128	0.7743
	AOL	28.0408	0.8045	28.1916	0.8124	30.3587	0.8192
	MADL-$\ell_{1/2}$	**28.1234**	**0.8163**	**28.2356**	**0.8322**	**30.5012**	**0.8204**

Acknowledgments

This work was supported in part by the Guangdong program for support of top-notch young professionals under grant 2014TQ01X144, the National Natural Science Foundation of China under grants 61722304, 61333013, the Pearl River S&T Nova Program of Guangzhou under grant 201610010196, and the Guangdong Natural Science Funds for Distinguished Young Scholar under grant 2014A030306037.

References

Absil, P.-A., Mahony, R., & Sepulchre, R. (2008). *Optimization algorithms on matrix manifolds*. Princeton, NJ: Princeton University Press.

Aharon, M., Elad, M., & Bruckstein, A. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, *54*(11), 4311–4322. http://dx.doi.org/10.1109/TSP.2006.881199.

Bahrampour, S., Nasrabadi, N. M., Ray, A., & Jenkins, W. K. (2016). Multimodal task-driven dictionary learning for image classification. *IEEE Transactions on Image Processing*, *25*(1), 24–38. http://dx.doi.org/10.1109/TIP.2015.2496275.

Dong, J., Wang, W., Dai, W., Plumbley, M., Han, Z., & Chambers, J. (2015). Analysis simco algorithms for sparse analysis model based dictionary learning. *IEEE Transactions on Signal Processing*, *64*(2), 417–431.

Eksioglu, E., & Bayir, O. (2014). K-SVD meets transform learning: Transform K-SVD. *IEEE Signal Processing Letters*, *21*(3), 347–351. http://dx.doi.org/10.1109/LSP.2014.2303076.

Elad, M. (2010). *Sparse and redundant representation*. Berlin, Germany: Springer.

Engan, K., Aase, S., & Husoy, J. (1999). Method of optimal directions for frame design. In *Proc. IEEE Int. conf. acoust., speech, signal process. (ICASSP)*, Vol. 5 (pp. 2443–2446).

Goldfarb, D., Wen, Z., & Yin, W. (2009). A curvilinear search method for p-harmonic flows on spheres. *SIAM Journal on Imaging Sciences*, *2*(1), 84–109.

Gong, C., Tao, D., Liu, W., Liu, L., & Yang, J. (2017). Label propagation via teaching-to-learn and learning-to-teach. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(6), 1452–1465.

Gong, P., Zhang, C., Lu, Z., Huang, J., & Ye, J. (2013). A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems, arXiv:1303.4434.

Hawe, S., Kleinsteuber, M., & Diepold, K. (2013). Analysis operator learning and its application to image reconstruction. *IEEE Transactions on Image Processing*, *22*(6), 2138–2150.

Hoyer, P. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, *5*, 1457–1469.

Huang, K., & Aviyente, S. (2006). Sparse representation for signal classification. In *Proc. conf. neur. inf. process. syst.* (pp. 609–616).

Li, Z., Ding, S., Hayashi, T., & Li, Y. (2017). Analysis dictionary learning using block coordinate descent framework with proximal operators. *Neurocomputing*, *239*, 165–180.

Li, Z., Ding, S., & Li, Y. (2015). A fast algorithm for learning overcomplete dictionary for sparse representation based on proximal operators. *Neural Computation*, *27*(9), 1951–1982.

Li, Z., Hayashi, T., Ding, S., Li, Y., & Li, X. (2016). Constrained analysis dictionary learning with the $\ell_{1/2}$-norm regularizer. In *IEEE 13th Int. conf. signal process. (ICSP)* (pp. 890–894). http://dx.doi.org/10.1109/ICSP.2016.7877958.

Liang, J., Zhang, M., Zeng, X., & Yu, G. (2014). Distributed dictionary learning for sparse representation in sensor networks. *IEEE Transactions on Image Processing*, *23*(6), 2528–2541. http://dx.doi.org/10.1109/TIP.2014.2316373.

Nam, S., Davies, M., Elad, M., & Gribonval, R. (2010). Cosparse analysis modeling-Uniqueness and algorithms. In *IEEE Int. conf. acoustics, speech and signal processing (ICASSP)* (pp. 5804–5807).

Niu, L., Zhou, R., Tian, Y., Qi, Z., & Zhang, P. (2016). Nonsmooth penalized clustering via ℓ_p regularized sparse regression. *IEEE Transactions on Cybernetics*, *PP*(99), 1–11. http://dx.doi.org/10.1109/TCYB.2016.2546965.

Rakotomamonjy, A., Flamary, R., & Gasso, G. (2016). DC proximal newton for non-convex optimization problems. *IEEE Transactions on Neural Networks and Learning Systems*, *27*(3), 636–647. http://dx.doi.org/10.1109/TNNLS.2015.2418224.

Ravishankar, S., & Bresler, Y. (2012). Learning sparsifying transforms for image processing. In *IEEE Int. conf. image processing (ICIP)* (pp. 681–684).

Ravishankar, S., & Bresler, Y. (2013). Learning overcomplete sparsifying transforms for signal processing. In *IEEE Int. conf. acoustics, speech and signal processing (ICASSP)* (pp. 3088–3092).

Rubinstein, R., Peleg, T., & Elad, M. (2013). Analysis K-SVD: A dictionary-learning algorithm for the analysis sparse model. *IEEE Transactions on Signal Processing*, *61*(3), 661–677.

Seibert, M., Wormann, J., Gribonval, R., & Kleinsteuber, M. (2016). Learning co-sparse analysis operators with separable structures. *IEEE Transactions on Signal Processing*, *64*(1), 120–130.

(0000). USPS handwritten digit database, available at: http://www.cs.toronto.edu/roweis/data.html.

Wang, Z., Bovik, A. C., Sheikh, H. R., Sapiro, G., & Simoncelli, E. P. (2014). Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, *13*(4), 600–612.

Wang, W., & Qian, Y. (2015). Adaptive $\ell_{1/2}$ sparsity-constrained nmf with half-thresholding algorithm for hyperspectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observation and Remote Sensing*, *8*(6), 2618–2631.

Wen, B., Ravishankar, S., & Bresler, Y. (2015). Structured overcomplete sparsifying transform learning with convergence guarantees and applications. *International Journal of Computer Vision*, *114*(2–3), 137–167.

Wen, Z., & Yin, W. (2010). A feasible method for optimization with orthogonality constraints, UCLA CAM (pp. 10–77).

Xing, F. C. (2003). Investigation on solutions of cubic equations with one unkonwn. *Natural Science Edition*, *12*(3), 207–218.

Xu, Z., Zhang, H., Wang, Y., & Chang, X. (2010). $L_{1/2}$ regularizer. *Science China*, *53*(6), 1159–1169.

Yaghoobi, M., Nam, S., Gribonval, R., & Davies, M. (2013). Constrained overcomplete analysis operator learning for cosparse signal modelling. *IEEE Transactions on Signal Processing*, *61*(9), 2341–2355. http://dx.doi.org/10.1109/TSP.2013.2250968.

Yang, Z., Zhang, Y., Yan, W., Xiang, Y., & Xie, S. (2016). A fast non-smooth nonnegative matrix factorization for learning sparse representation. *IEEE Access*, *4*(6), 5161–5168.

Zaki, A., Venkitaraman, A., Chatterjee, S., & Rasmussen, L. K. (2017). Greedy Sparse Learning over Network. *IEEE Transactions on Signal Information Processing Networks*. http://dx.doi.org/10.1109/TSIPN.2017.2710905.

Zhang, M., & Desrosiers, C. (2017). Image denoising based on sparse representation and gradient histogram. *IET Image Processing*, *11*(1), 54–63. http://dx.doi.org/10.1049/iet-ipr.2016.0098.

Zhang, Y., Yu, T., & Wang, W. (2014). An analysis dictionary learning algorithm under a noisy data model with orthogonality constraint. *Scientific World Journal*, *2014*(852978).