Appl. Comput. Harmon. Anal. 38 (2015) 510–523

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

www.elsevier.com/locate/acha

Letter to the Editor

Convergence analysis for iterative data-driven tight frame construction scheme

Chenglong Bao, Hui Ji *, Zuowei Shen

Department of Mathematics, National University of Singapore, Singapore 117543, Singapore

A R T I C L E   I N F O

Article history:
Received 30 January 2014
Received in revised form 24 June 2014
Accepted 27 June 2014
Available online 2 July 2014
Communicated by Charles K. Chui

Keywords:
Tight frame
Sparse approximation
Non-convex optimization
Convergence analysis

A B S T R A C T

Sparse modeling/approximation of images plays an important role in image restoration. Instead of using a fixed system to sparsely model any input image, a more promising approach is using a system that is adaptive to the input image. A non-convex variational model is proposed in [1] for constructing a tight frame that is optimized for the input image, and an alternating scheme is used to solve the resulting non-convex optimization problem. Although it showed good empirical performance in image denoising, the proposed alternating iteration lacks convergence analysis. This paper aims at providing the convergence analysis of the method proposed in [1]. We first established the sub-sequence convergence property of the iteration scheme proposed in [1], i.e., there exists at least one convergent sub-sequence and any convergent sub-sequence converges to a stationary point of the minimization problem. Moreover, we showed that the original method can be modified to have sequence convergence, i.e., the modified algorithm generates a sequence that converges to a stationary point of the minimization problem.

1. Introduction

It is now well established that sparse modeling is a very powerful tool for many image recovery tasks, which models an image as the linear combination of only a small number of elements of some system. Such a system can be either a basis or an over-complete system. When using the sparsity prior of images to regularize image recovery, the performance largely depends on how effective images of interest can be sparsely approximated under the given system. Therefore, a fundamental question in sparsity-based image regularization is how to define a system such that the target image has an optimal sparse approximation. Earlier work on sparse modeling focuses on the design of orthonormal bases, such as *discrete cosine transform* [2], *wavelets* [3,4]. Owing to their better performance in practice, over-complete systems have been more recognized in sparsity-based image recovery methods. In particular, as a redundant extension of orthonormal

* Corresponding author.
E-mail addresses: baochenglong@nus.edu.sg (C. Bao), matjh@nus.edu.sg (H. Ji), matzuows@nus.edu.sg (Z. Shen).

http://dx.doi.org/10.1016/j.acha.2014.06.007

bases, tight frames are now wide-spread in many applications as they have the same efficient and simple decomposition and reconstruction schemes as orthonormal bases. Many types of tight frames have been proposed for sparse image modeling including *shift-invariant wavelets* [5], *framelets* [6,7], *curvelets* [8] and many others. These tight frames are optimized for the signals with certain functional properties, which do not always hold true for natural images. As a consequence, a more effective approach to sparsely approximate images of interest is to construct tight frames that are adaptive to the inputs.

In recent years, the concept of data-driven systems has been exploited to construct adaptive systems for sparsity-based modeling (see e.g. [1,9–11]). The basic idea is to construct the system that is adaptive to the input so as to obtain a better sparse approximation than the predefined ones. Most sparsity-based dictionary learning methods [9–11] treat the input image as the collection of small image patches, and then construct an over-complete dictionary for sparsely approximating these image patches. Despite the impressive performance in various image restoration tasks, the minimization problems proposed by these methods are very challenging to solve. As a result, the numerical methods proposed in past for these models not only lack rigorous analysis on their convergence and stability, but also are very computationally demanding.

Recently, Cai et al. [1] proposed a variational model to learn a tight frame system that is adaptive to the input image in terms of sparse approximation. Differently from the existing over-complete dictionary learning methods, the adaptive systems constructed in [1] are tight frames that have *perfect reconstruction property*, a property ensuring that any input can be perfectly reconstructed by its canonical coefficients in a simple manner. The tight frame property of the system constructed in [1] not only is attractive to many image processing tasks, but also leads to very efficient construction scheme. Indeed, by considering a special class of tight frames, the construction scheme proposed in [1] only requires solving an ℓ_0 norm related non-convex minimization problem:

$$\min_{D\in\mathbb{R}^{m\times m},C\in\mathbb{R}^{m\times n}} \left\|C - D^\top Y\right\|_F^2 + \lambda_0^2\|C\|_0, \quad \text{s.t.} \quad D^\top D = m^{-1}I_m, \tag{1}$$

where D contains framelet filters and C contains the canonical frame coefficients. An alternating iteration is proposed in [1] for solving (1), which is very fast as both sub-problems in each iteration have closed-form solutions. It is shown that, with comparable performance in image denoising, the proposed adaptive tight frame construction runs much faster than other generic dictionary learning methods (e.g. the K-SVD method [10]). However, Cai et al. [1] did not provide any convergence analysis of the proposed method.

As a sequel to [1], this paper provides the convergence analysis of the alternating iterative method proposed in [1] for solving (1). In this paper, we showed that the algorithm provided by [1] has sub-sequence convergence property. In other words, we showed that there exists at least one convergent sub-sequence of the sequence generated by the algorithm [1] and any convergent sub-sequence converges to a stationary point of (1). Moreover, we empirically observed that the sequence generated by the algorithm proposed in [1] itself is not convergent. Motivated by the theoretical interest, we modified the algorithm proposed in [1] by adding a proximal term in the iteration scheme, and then showed that the modified algorithm has sequence convergence. In other words, the sequence generated by the modified method convergences to a stationary point of (1).

2. Brief review on data-driven tight frame construction and related works

In this section, we gave a brief review on tight frames, data-driven tight frames proposed in [1] and some most related works. Interested readers are referred to [12,13] for more details.

2.1. Tight frames and data-driven tight frames

For a Hilbert space $\mathcal{H}$, a sequence $\{x_n\} \subset \mathcal{H}$ is a *tight frame* for $\mathcal{H}$ if

$$\|x\|^2 = \sum_n \big|\langle x, x_n\rangle\big|^2, \quad \text{for any } x \in \mathcal{H},$$

or equivalently, $x = \sum_n \langle x, x_n\rangle x_n, x \in \mathcal{H}$. The sequence $\{\langle x, x_n\rangle\}$ is called the canonical frame coefficient sequence. A tight frame $\{x_n\}$ is an orthonormal basis for $\mathcal{H}$ if and only if $\|x_n\| = 1$ for all x_n. A tight frame has two associated operators: the *analysis operator* $\boldsymbol{W}$ defined by

$$\boldsymbol{W} : x \in \mathcal{H} \longrightarrow \big\{\langle x, x_n\rangle\big\} \in \ell_2(\mathbb{N})$$

and its adjoint operator $\boldsymbol{W}^\top$ (often called the *synthesis operator*):

$$\boldsymbol{W}^\top : \{a_n\} \in \ell_2(\mathbb{N}) \longrightarrow \sum_n a_n x_n \in \mathcal{H}.$$

Then, the sequence $\{x_n\} \subset \mathcal{H}$ is a tight frame if and only if $\boldsymbol{W}^\top \boldsymbol{W} = I$, where I denotes the identity operator of $\mathcal{H}$. The tight frames considered in [1] are single-level un-decimal discrete wavelet systems generated by all integer shifts of a set of filters $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_m\}$. For any filter $\boldsymbol{a} \in \ell_2(\mathbb{Z})$, let $\mathcal{S}_{\boldsymbol{a}} : \ell_2(\mathbb{Z}) \to \ell_2(\mathbb{Z})$ denote its associated convolution operator defined by

$$\big[\mathcal{S}_{\boldsymbol{a}}(\boldsymbol{v})\big](n) := [\boldsymbol{a} \star \boldsymbol{v}](n) = \sum_{k \in \mathbb{Z}^2} \boldsymbol{a}(n-k)\boldsymbol{v}(k), \quad \forall \boldsymbol{v} \in \ell_2(\mathbb{Z}). \tag{2}$$

Then, for a given set of framelet filters, we define its associated analysis operator $\boldsymbol{W}$ by

$$\boldsymbol{W} = \big[\mathcal{S}^\top_{\boldsymbol{a}_1(-\cdot)}, \mathcal{S}^\top_{\boldsymbol{a}_2(-\cdot)}, \cdots, \mathcal{S}^\top_{\boldsymbol{a}_m(-\cdot)}\big]^\top. \tag{3}$$

The rows of $\boldsymbol{W}$ form a tight frame for $\ell_2(\mathbb{Z})$ if and only if $\boldsymbol{W}^\top \boldsymbol{W} = I$, and the corresponding synthesis operator is the transpose of $\boldsymbol{W}$, denoted by $\boldsymbol{W}^\top$.

The data-driven tight frame construction proposed in [1] constructs the set of framelet filters $\{\boldsymbol{a}_j\}_{j=1}^m$ via solving the following problem:

$$\min_{\boldsymbol{v}, \{\boldsymbol{a}_i\}_{i=1}^m} \big\|\boldsymbol{v} - \boldsymbol{W}(\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_m)\boldsymbol{g}\big\|_F^2 + \lambda_0^2 \|\boldsymbol{v}\|_0, \quad \text{s.t.} \quad \boldsymbol{W}^\top \boldsymbol{W} = I, \tag{4}$$

where $\boldsymbol{g}$ denotes the input signal, $\{\boldsymbol{a}_j\}_{j=1}^m$ denotes the set of framelet filters of the adaptive tight frame, and $\boldsymbol{v}$ denotes the canonical coefficient vector of $\boldsymbol{g}$. Here and throughout this paper, $\|\boldsymbol{v}\|_0$ stands for the number of non-zero elements of $\boldsymbol{v}$ and $\|\cdot\|_F$ denotes the Frobenius norm.

2.2. Data-driven tight frame construction scheme [1]

For general framelet filters, the minimization problem (4) is very challenging to solve. Therefore, a special class of framelet filters are considered in [1], which is composed of m^2 2D real-valued framelet filters $\{\boldsymbol{a}_j\}_{j=1}^{m^2} \subset \mathbb{R}^{m \times m}$. Let D denote the associated filter matrix defined by

$$A = [\vec{\boldsymbol{a}}_1, \vec{\boldsymbol{a}}_2, \ldots, \vec{\boldsymbol{a}}_{m^2}],$$

where $\vec{\boldsymbol{a}}_j$ denotes the vector form of $\boldsymbol{a}_j$ by concatenating all columns of a_j to a column vector. It is shown in [1, Proposition 3] that the rows of $\boldsymbol{W}$ defined by $\{\boldsymbol{a}_j\}_{j=1}^{m^2} \subset \mathbb{R}^{m\times m}$ form a tight frame for $\ell^2(\mathbb{Z})$, provided that $A^\top A = \frac{1}{m} I_{m^2}$. Thus, the minimization problem (4) for general tight frame construction is simplified to the following one:

$$\min_{\boldsymbol{v},\{\boldsymbol{a}_i\}_{i=1}^{m^2}} \left\|\boldsymbol{v} - \boldsymbol{W}(A)\boldsymbol{g}\right\|_F^2 + \lambda_0^2\|\boldsymbol{v}\|_0, \quad \text{s.t.} \quad A^\top A = \frac{1}{m} I_{m^2}. \tag{5}$$

The problem (5) can be re-formulated in terms of image patches as follows. Let $\{\vec{\boldsymbol{g}}_\ell\}_{\ell=1}^L \subset \mathbb{R}^{m^2}$ denote the set of all image patches of size $m \times m$ densely sampled from the image $\boldsymbol{g}$. For each patch vector $\vec{g}_\ell$, let $\vec{\boldsymbol{v}}_n = A^\top \vec{\boldsymbol{g}}_\ell \in \mathbb{R}^{m^2}$ denote the vector generated by the inner product between $\vec{\boldsymbol{g}}_n$ and all m^2 framelet filters $\{\vec{\boldsymbol{a}}_j\}_{j=1}^{m^2}$. Define three matrices as follows,

$$\begin{cases} Y := \dfrac{1}{\sqrt{m}}[\vec{\boldsymbol{g}}_1, \vec{\boldsymbol{g}}_2, \ldots, \vec{\boldsymbol{g}}_L] \in \mathbb{R}^{m^2\times L}; \\ D := \sqrt{m}A = \sqrt{m}[\vec{\boldsymbol{a}}_1, \vec{\boldsymbol{a}}_2, \ldots, \vec{\boldsymbol{a}}_{m^2}] \in \mathbb{R}^{m^2\times m^2}; \\ C := [\vec{\boldsymbol{v}}_1, \vec{\boldsymbol{v}}_2, \ldots, \vec{\boldsymbol{v}}_{m^2}] \in \mathbb{R}^{m^2\times L}. \end{cases} \tag{6}$$

Then, it is shown in [1] that the minimization (5) is equivalent to

$$\min_{D\in\mathbb{R}^{m^2\times m^2}, C\in\mathbb{R}^{m^2\times L}} \left\|C - D^\top Y\right\|_F^2 + \lambda^2\|C\|_0, \quad \text{s.t.} \quad D^\top D = I_{m^2\times m^2}, \tag{7}$$

where λ denotes some predefined regularization parameter.

The minimization model (7) is solved in [1] via an alternating scheme between D and C. More specifically, given the current estimate (D_k, C_k), the next iteration updates it via the following scheme:

$$\begin{cases} D_{k+1} \in \underset{D\in\mathbb{R}^{m^2\times m^2}}{\arg\min} \left\|C_k - D^\top Y\right\|_F^2, \quad \text{s.t.} \quad D^\top D = I; \\ C_{k+1} \in \underset{C\in\mathbb{R}^{m^2\times L}}{\arg\min} \left\|C - D_{k+1}^\top Y\right\|_F^2 + \lambda^2\|C\|_0. \end{cases} \tag{8}$$

Define the *hard thresholding operator* $T_\lambda : \mathbb{R}^{m^2\times L} \to \mathbb{R}^{m^2\times L}$ by

$$\left[T_\lambda(Y)\right]_{i,j} = \begin{cases} Y_{i,j}, & \text{if } |Y_{i,j}| > \lambda; \\ \{0, \lambda\}, & \text{if } |Y_{i,j}| = \lambda; \\ 0, & \text{if } |Y_{i,j}| < \lambda. \end{cases} \tag{9}$$

It is shown in [1] that both sub-problems in (8) have closed-form solutions given by

$$D_{k+1} := U_k V_k^\top; \qquad C_{k+1} \in T_\lambda\left(D_{k+1}^\top Y\right), \tag{10}$$

where U_k and V_k are given by the singular value decomposition (SVD) of $YC_k^\top$ such that $YC_k^\top = U_k \boldsymbol{\Sigma}_k V_k^\top$. See Algorithm 1 for the summary of the alternating iteration scheme [1].

2.3. Related works

The minimization (7) is an ℓ_0 norm relating non-convex problem with quadratic constraints. Algorithm 1 proposed in [1] for solving (7) alternatingly updates the filter matrix D by the SVD and updates the coefficient matrix C by hard thresholding the coefficients from the last estimate. Such an iterative hard

Algorithm 1 Alternating iteration scheme [1] for solving (7).

1: **INPUT:** Input image $\boldsymbol{g}$;
2: **OUTPUT:** Adaptive filter set D;
3: **Main Procedure:**
  i Set initial filter matrix D_0 and coefficient matrix C_0.
  ii Construct the patch matrix Y as (6).
  iii For $k = 0, 1, \cdots,$
    1. compute the SVD of $YC_k^\top = U_k \boldsymbol{\Sigma}_k V_k^\top$;
    2. $D_{k+1} := U_k V_k^\top$ and $C_{k+1} \in T_\lambda(D_{k+1}^\top Y)$.

thresholding on wavelet frame coefficients approach has been used in solving various linear inverse problems in image recovery, see e.g. the wavelet frame based image super-resolution methods [14,15].

As a sparsity prompting functional, the ℓ_0 norm is also used in other sparse approximation based dictionary learning methods. The popular K-SVD method [10] proposed the following minimization model for learning an over-complete dictionary $D = \{D_1, D_2, \ldots, D_m\} \subset \mathbb{R}^n$ with $m > n$:

$$\min_{D \in \mathbb{R}^{n\times m}, C \in \mathbb{R}^{m\times p}} \frac{1}{2}\|Y - DC\|_F^2 + \lambda\|C\|_0, \quad \text{s.t.} \quad \|D_i\|_2 = 1,\ i = 1, 2, \ldots, n. \tag{11}$$

An alternating iteration scheme between D and C is used in the K-SVD method for solving (11). Differently from the model (7) proposed in [1], the ℓ_0 norm related minimization problem for updating the code C is a challenging one. The greedy algorithm, such as orthogonal matching pursuit, is used in [10] for estimating the code. Therefore, the computational cost of the K-SVD method is much higher than of Algorithm 1.

Both the K-SVD method and Algorithm 1 perform noticeably better in image denoising than other wavelet frame based methods. The advantage of Algorithm 1 over the K-SVD method lies in its computational efficiency. Despite their impressive performances in practice, both methods lack the convergence analysis. Indeed, it is empirically observed that the sequences generated by both methods are not convergent. In this paper, we first provided the convergence analysis for Algorithm 1 by showing that the sequence generated by Algorithm 1 has sub-sequence convergence. Then we proposed a modified version of Algorithm 1 for solving (7) and established the sequence convergence of the new algorithm.

3. Sub-sequence convergence property of Algorithm 1

In this section, we will show that the sequence generated by Algorithm 1 has sub-sequence convergence property, i.e., there exists at lease one convergent subsequence and every convergent subsequence converges to a stationary point of (7). Before establishing the main result, we first introduce the definition of the stationary point of non-convex and non-smooth functions.

Definition 3.1. Let $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$ be a proper lower semi-continuous function.

1. The domain of f is defined by $\operatorname{dom} f := \{x \in \mathbb{R}^n : f(x) < +\infty\}$.
2. For each $x \in \operatorname{dom} f$, x is called the coordinate-wise minimum of f if it satisfies
$$f\big(x + (0, \cdots, d_k, \cdots, 0)\big) \le f(x), \quad \forall d_k,\ 1 \le k \le n,$$
where $x = (x_1, x_2, \cdots, x_n)$.
3. The Fréchet subdifferential $\partial_F f$ is defined by
$$\partial_F f(x) = \left\{ z : \liminf_{y \to x} \frac{f(y) - f(x) - \langle z, x - y\rangle}{\|x - y\|} \ge 0 \right\} \tag{12}$$
for any $x \in \operatorname{dom} f$ and $\partial_F f(x) = \emptyset$ if $x \notin \operatorname{dom} f$.
4. For each $x \in \operatorname{dom} f$, x is called the *stationary point* of f if it satisfies $0 \in \partial_F f(x)$.

Remark. There are several definitions for stationary points of proper lower semi-continuous functions. In [16], the stationary point x is defined as

$$\liminf_{\lambda \downarrow 0} \frac{f(x+\lambda y) - f(x)}{\lambda} \geq 0, \quad \forall y \in R^n.$$

In [17], the stationary point x of f is defined by $0 \in \partial f(x)$, where ∂f is the limiting subdifferential given by

$$\partial f(x) = \big\{ z : \exists x_n \to x, f(x_n) \to f(x), z_n \in \partial_F f(x_n) \to z \big\}.$$

The definition of stationary points used in this paper is different from the definitions used in [16] and [17]. Indeed, ours is stronger than the other two definitions.

To simplify notations, define $\mathcal{X} = \{D \in \mathbb{R}^{m^2 \times m^2} : D^\top D = I_{m^2}\}$ and define $\Omega_C = \mathbb{R}^{m^2 \times N}$, $\Omega_D = \mathbb{R}^{m^2 \times m^2}$, $\Omega_z = (\Omega_C, \Omega_D)$. Define

$$f(C) = \lambda^2 \|C\|_0, \qquad Q(C,D) = \big\|D^\top Y - C\big\|_F^2, \qquad g(D) = I_{\mathcal{X}}(D), \tag{13}$$

where $I_{\mathcal{X}}(D) = 0$, if $D \in \mathcal{X}$ and $+\infty$ otherwise. Then, the minimization (7) can be re-written as

$$\min_{C \in \Omega_C, D \in \Omega_D} L(C,D) := f(C) + Q(C,D) + g(D). \tag{14}$$

Before proving the sub-sequence convergence property of Algorithm 1, we first establish some facts and results related to (14). First, the function g is a lower semi-continuous function, as $\mathcal{X}$ is a compact set. Second, it can be seen that for any $Z = (C,D)$, the function $Q(Z)$ satisfies the following properties:

$$\begin{cases} Q(C,D) = Q(C_1,D) + \big\langle \nabla_C Q(C_1,D), C - C_1 \big\rangle + o\big(\|C - C_1\|_F\big), & \forall C_1 \in \Omega_C; \\ Q(C,D) = Q(C,D_1) + \big\langle \nabla_D Q(C,D_1), D - D_1 \big\rangle + o\big(\|D - D_1\|_F\big), & \forall D_1 \in \Omega_D; \\ Q(C,D) = Q(C_1,D_1) + \big\langle \nabla Q(C_1,D_1), Z - Z_1 \big\rangle + o\big(\|Z - Z_1\|_F\big), & \forall Z_1 \in \Omega_Z, \end{cases} \tag{15}$$

where $o(\|x\|_F)$ is defined by $\lim_{\|x\|_F \to 0} \frac{o(\|x\|_F)}{\|x\|_F} = 0$.

Lemma 3.2. *The sequence $Z_k := (C_k, D_k)$ generated by Algorithm 1 is a bounded sequence. For any convergent sub-sequence $Z_{k'}$ with limit point $Z^* = (C^*, D^*)$, we have*

$$\lim_{k' \to +\infty} f(C_{k'}) = f\big(C^*\big), \quad \textit{and} \quad \lim_{k' \to +\infty} L(Z_{k'}) = L\big(Z^*\big).$$

Proof. By the definition of (10), we have

$$L(Z_k) \leq L(C_k, D_{k-1}) \leq L(C_{k-1}, D_{k-1}) \leq \cdots \leq L(Z_0),$$

which implies

$$\|C_k\|_F - \big\|D_k^\top Y\big\|_F \leq \big\|D_k^\top Y - C^k\big\|_F \leq \sqrt{L(Z_0)}, \quad k = 1, 2, \ldots. \tag{16}$$

Together with (16) and the fact that $D_k \in \mathcal{X}$, we have that Z_k is bounded. Next, by the definition of (10), we also have

$$Q(C_{k'}, D_{k'}) + f(C_{k'}) \leq Q(C, D_{k'}) + f(C), \quad \forall C \in \Omega_C. \tag{17}$$

By substituting C by C^* and taking $k' \to \infty$ in (17), we have $\liminf_{k'\to+\infty} f(C_{k'}) \leq f(C^*)$. Together with the fact that $f(C) = \lambda^2\|C\|_0$ is lower semi-continuous and $C_{k'} \to C^*$ as $k' \to +\infty$, we have

$$\liminf_{k'\to+\infty} f(C_{k'}) = f\big(C^*\big).$$

Since $D_{k'} \in \mathcal{X}$ for all k' and $\mathcal{X}$ is a compact subset, $D^* \in \mathcal{X}$ and $g(D^*) = g(D_{k'}) = 0$ for all k'. It can be seen that $Q(C_{k'}, D_{k'}) \to Q(C^*, D^*)$ as $k' \to +\infty$, as Q is a continuous function. In addition, $L(Z_k)$ is decreasing by (16) and $L \geq 0$, which implies that $L(Z_k)$ is a convergent sequence. Consequently, we have

$$\lim_{k'\to+\infty} f(C_{k'}) = f\big(C^*\big),$$

since $f(C) = L(Z) - Q(Z) - g(D)$. Moreover, we have

$$\begin{aligned}\lim_{k'\to+\infty} L(Z_{k'}) &= \lim_{k'\to+\infty} f(C_{k'}) + \lim_{k'\to+\infty} Q(C_{k'}, D_{k'}) + \lim_{k'\to+\infty} g(D_{k'})\\ &= f\big(C^*\big) + Q\big(C^*, D^*\big) + g\big(D^*\big).\end{aligned}$$

Thus, $\lim_{k'\to+\infty} L(Z_{k'}) = L(Z^*)$. □

Lemma 3.3. *Let $Z_k := (C_k, D_k)$ denote the sequence generated by Algorithm 1 and let Ω_* denote the set that contains all limit points of Z_k. Then Ω_* is not empty and*

$$L\big(C^*, D^*\big) = \inf_k L(C_k, D_k), \quad \forall\big(C^*, D^*\big) \in \Omega_*.$$

Proof. By Lemma 3.2, Z_k is a bounded sequence. Thus, the set Ω_* is a non-empty set. Moreover, the set Ω_* is also a compact set as $\Omega_* = \bigcap_{j\in\mathbb{N}} \overline{\bigcup_{k\geq j}\{Z_k\}}$. Notice that $L(Z_k)$ is a decreasing sequence and $L(Z) \geq 0$. Then, there exists some constant ρ such that $\inf_k L(Z_k) = \rho$. Take any $Z^* \in \Omega_*$ and assume $Z_{k'} \to Z^*$ as $k' \to +\infty$. By Lemma 3.2, we have that $\lim_{k'\to+\infty} L(Z_{k'}) = L(Z^*) = \rho$. □

At last, we show that the sequence generated by Algorithm 1 has sub-sequence convergence property.

Proof. By Lemma 3.3, the sequence $Z_k := (C_k, D_k)$ generated by Algorithm 1 has at least one limit point. For any limit point $Z^1 = (C^1, D^1)$ of the sequence Z_k, let $\{Z_{k'}\}$ be the sub-sequence of Z_k that converges to Z^1. Without loss of generality, assume the sub-sequence $\{Z_{k'+1}\}$ converges to $Z^2 = (C^2, D^2)$. By the definition of the second step in (10), we have

$$Q(C_{k'}, D_{k'}) + f(C_{k'}) \leq Q(C, D_{k'}) + f(C), \quad \forall C \in \Omega_C. \tag{18}$$

Taking $k' \to +\infty$ in (18), by Lemma 3.2, we have

$$g\big(D^1\big) + Q\big(C^1, D^1\big) + f\big(C^1\big) \leq g\big(D^1\big) + Q\big(C, D^1\big) + f(C), \quad \forall C \in \Omega_C, \tag{19}$$

which implies

$$L\big(C^1, D^1\big) \leq L\big(C^1 + C, D^1\big), \quad \forall C \in \Omega_C. \tag{20}$$

As $Z_{k'+1}$ is defined from $Z_{k'}$ by (10), we have

$$\begin{cases} Q(C_{k'}, D_{k'+1}) + g(D_{k'+1}) \leq Q(C_{k'}, D) + g(D), & \forall D \in \Omega_D; \\ Q(C_{k'+1}, D_{k'+1}) + f(C_{k'+1}) \leq Q(C, D_{k'+1}) + f(C), & \forall C \in \Omega_C. \end{cases}$$

The summation of the first inequality and the second inequality with $C = C_{k'}$ gives

$$g(D_{k'+1}) + Q(C_{k'+1}, D_{k'+1}) + f(C_{k'+1}) \leq g(D) + Q(C_{k'}, D) + f(C_{k'}). \tag{21}$$

Taking $k' \to +\infty$ in (21). By Lemma 3.2 and Lemma 3.3, we have

$$L\big(C^1, D^1\big) = L\big(C^2, D^2\big) \leq L\big(C^1, D^1 + D\big). \tag{22}$$

Thus, the combination of (20) and (22) shows that the point (C^1, D^1) is a coordinate-wise minimum point of (14). Therefore, for any $\delta_Z = (\delta_C, \delta_D)$, we have

$$\begin{aligned}
&\liminf_{\|\delta_Z\|\to 0} \frac{L(Z^1 + \delta_Z) - L(Z^1)}{\|\delta_Z\|} \\
&\quad = \liminf_{\|\delta_Z\|\to 0} \frac{Q(Z^1 + \delta_Z) - Q(Z^1) + f(C^1 + \delta_C) - f(C^1) + g(D^1 + \delta_D) - g(D^1)}{\|\delta_Z\|} \\
&\quad \geq \liminf_{\|\delta_Z\|\to 0} \frac{\langle \nabla Q(Z^1), \delta_Z\rangle + f(C^1 + \delta_C) - f(C^1) + g(D^1 + \delta_D) - g(D^1)}{\|\delta_Z\|} \\
&\quad = \liminf_{\|\delta_Z\|\to 0} \bigg(\frac{Q(C^1 + \delta_C, D^1) - Q(C^1, D^1) - o(\|\delta_C\|) + f(C^1 + \delta_C) - f(C^1)}{\|\delta_Z\|} \\
&\qquad + \frac{Q(C^1, D^1 + \delta_D) - Q(C^1, D^1) - o(\|\delta_D\|) + g(D^1 + \delta_D) - g(D^1)}{\|\delta_Z\|} \bigg) \\
&\quad \geq \liminf_{\|\delta_Z\|\to 0} \frac{-o(\|\delta_C\|) - o(\|\delta_D\|)}{\|\delta_Z\|} = 0,
\end{aligned}$$

where the first inequality is from (15) and the second inequality is from the fact that $Z^1 := (C^1, D^1)$ is the coordinate-wise minimum point of (14). By Definition (3.1), the point Z^1 is a stationary point of (14). $\square$

4. A modified algorithm for (7) with sequence convergence

In the previous section, we showed that the sequence generated by Algorithm 1 has sub-sequence convergence property. The next question is whether the sequence itself is convergent or not. The experiments show that it is not the case; see Fig. 1(a) for the increments of the sequence C_k. The lack of sequence convergence is not crucial to the applications in image recovery, as the result we are seeking for is not the frame coefficient vector but the image synthesized from the coefficients. See Fig. 1(b) for an illustration. However, the divergence of the coefficient sequence could cause severe stability issue when the coefficient set is the one needed, e.g. in the case of sparse coding based recognition tasks. Motivated by both theoretical interest and the needs from applications, we proposed a modified version of Algorithm 1 with sequence convergence property, i.e., the sequence generated by the new algorithm converges to a stationary point of (14).

The modification on Algorithm 1 for gaining sequence convergence is done by adding a proximal term in each iteration, a technique which has been used in other alternating iterative methods to ensure the convergence. For example, the proximal method proposed in [17] for solving a class of non-convex and

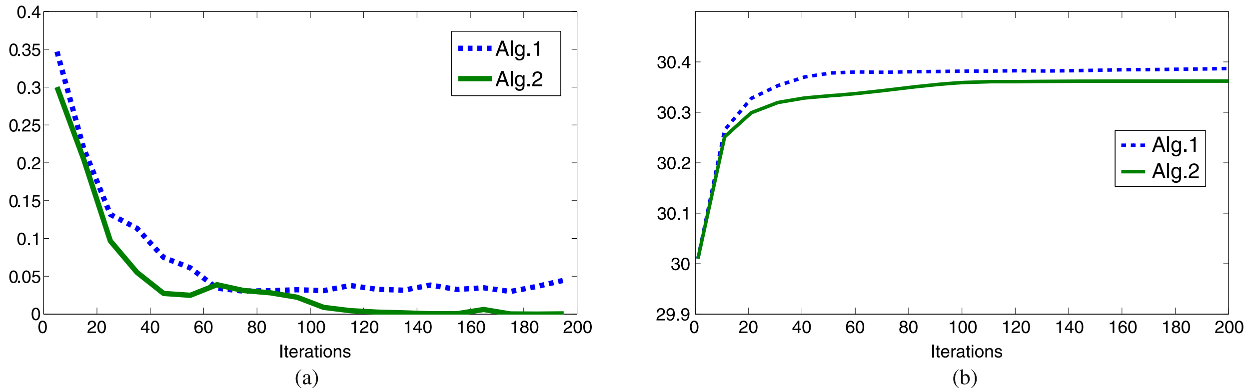

Fig. 1. Convergence behavior of Algorithm 1 and Algorithm 2. (a) The ℓ_2 norm of the increments of the framelet coefficient vector at each iteration; and (b) the PSNR values of the intermediate results at each iteration when denoising the image "boat" with noise level $\sigma = 20$.

non-smooth functions. The modified version of Algorithm 1 updates the estimates of C and D via solving the following problems:

$$\begin{cases} D_{k+1} \in \arg\min_{D} L(C_k, D) + \lambda_k \|D - D_k\|_F^2; \\ C_{k+1} \in \arg\min_{C} L(C, D_{k+1}) + \mu_k \|C - C_k\|_F^2, \end{cases} \tag{23}$$

where $\lambda_k, \mu_k \in (a, b)$ and $a, b > 0$. It can be seen that the new iteration (23) adds two additional proximal terms, $\lambda_k \|D - D_k\|_F^2$ and $\mu_k \|C - C_k\|_F^2$, to the original iteration (8). Same as (8), both minimization problems in (23) also have closed-form solutions.

Proposition 4.1. *The solution of (23) is given by*

$$\begin{cases} D_{k+1} = U_k V_k^\top, \\ C_{k+1} \in T_{\lambda/\sqrt{\mu_k+1}}\left(\dfrac{D_{k+1}^\top Y + \mu_k C_k}{1 + \mu_k}\right), \end{cases} \tag{24}$$

where U_k, V_k are given by the SVD of $Y C_k^\top + \lambda_k D_k = U_k \boldsymbol{\Sigma}_k V_k^\top$.

Proof. The proof is exactly the same as that of (10) provided in [1]. □

See Algorithm 2 for the summary of the modified algorithm for solving (14).

4.1. Convergence analysis of Algorithm 2

In this section, we first establish the sub-convergence property of Algorithm 2. Then we establish the sequence convergence of the algorithm by showing that the sequence is a Cauchy sequence and converges to a stationary point of (14). The main proof is built on the results presented in [17] about the convergence analysis of proximal methods for solving a class of non-smooth and non-convex problems.

Theorem 4.2. *Let $Z_k := (C_k, D_k)$ denote the sequence generated by Algorithm 2. Then, Z_k has at least one convergent subsequence and every convergent subsequence of Z_k converges to a stationary point of (14).*

Proof. By the definition of (23), we have

Algorithm 2 Proximal alternating iteration scheme for solving (7).

1: **INPUT:** Input image $\boldsymbol{g}$;
2: **OUTPUT:** Adaptive filter set D;
3: **Main Procedure:**
  i Set initial filter matrix D_0 and coefficient matrix C_0.
  ii Construct the patch matrix Y as (6).
  iii For $k = 0, 1, \cdots,$
    1. compute the SVD of $YC_k^\top + \lambda_k D_k = U_k \boldsymbol{\Sigma}_k V_k^\top$;
    2. $D_{k+1} = U_k V_k^\top$ and $C_{k+1} \in T_{\lambda/\sqrt{\mu_k+1}}(\frac{D_{k+1}^\top Y + \mu_k C_k}{1+\mu_k})$.

$$\begin{cases} L(C_k, D_{k+1}) + \lambda_k \|D_{k+1} - D_k\|_F^2 \le L(C_k, D_k), \\ L(C_{k+1}, D_{k+1}) + \mu_k \|C_{k+1} - C_k\|_F^2 \le L(C_k, D_{k+1}). \end{cases}$$

Sum up both inequalities and by the fact that $a \le \mu_k, \lambda_k \le b$, we have

$$L(Z_k) - L(Z_{k+1}) \ge a\|Z_k - Z_{k+1}\|_F^2 \ge 0. \tag{25}$$

By the same argument for (16), we have that Z_k is bounded and has at least one limit point. By (25), we obtain

$$L(Z_0) - L(Z_{k+1}) \ge \sum_{j=0}^{k} a\|Z_j - Z_{j+1}\|_F^2. \tag{26}$$

Let $k \to +\infty$ in (26). Together with the facts that $L(Z_k) \ge 0$ and $L(Z_k)$ is a decreasing sequence, we have

$$\sum_{k=1}^{+\infty} \|Z_k - Z_{k+1}\|_F^2 < +\infty,$$

which implies that

$$\lim_{k\to+\infty} \|Z_k - Z_{k+1}\|_F = 0. \tag{27}$$

Let $Z^1 := (C^1, D^1)$ denote any limit point of Z_k, i.e., there exists a sub-sequence $Z_{k'}$ that converges to Z^1. Next, we prove that the sub-sequence $Z_{k'+1}$ also converges to Z^1. For any $\epsilon > 0$, there exists N_0 such that $\|Z_{k'} - Z^1\|_F < \epsilon/2$ and $\|Z_{k'} - Z_{k'+1}\|_F < \epsilon/2$ for all $k' > N_0$. The first inequality is from the fact that $Z_{k'}$ converges to Z^1 and the second one is from (27). Thus, for all $k' > N_0$,

$$\left\|Z_{k'+1} - Z^1\right\|_F \le \left\|Z_{k'} - Z_{k'+1}\right\|_F + \left\|Z_{k'} - Z^1\right\|_F < \epsilon. \tag{28}$$

Consequently, we have $Z_{k'+1} \to Z^1$ as $k' \to +\infty$.

By the definition of (23), we have that, for any $C \in \Omega_C$,

$$L(C_{k'+1}, D_{k'+1}) + a\|C_{k'+1} - C_{k'}\|_F^2 \le L(C, D_{k'+1}) + b\|C - C_{k'}\|_F^2.$$

Similar to the derivation of (17), by setting $C = C^1$ and taking $k' \to +\infty$ in the inequality above, we have $\liminf_{k'\to+\infty} f(C_{k'+1}) \le f(C^1)$. As f is a lower semi-continuous function, we have

$$\liminf_{k'\to+\infty} f(C_{k'+1}) = f\left(C^1\right).$$

By the same arguments in the proof of Lemma 3.2, we have $\lim_{k'\to+\infty} f(C_{k'+1}) = f(C^1)$. Again, by using the same arguments for $\lim_{k'\to+\infty} f(C_{k'+1})$, we also have $\lim_{k'\to+\infty} f(C_{k'}) = f(C^1)$. Notice that $D_k \in \mathcal{X}$, $k = 1, 2, \ldots$, and $\mathcal{X}$ is a compact set. Thus, $g(D_{k'}) = g(D_{k'+1}) = g(D^1) = 0$ and Q is continuous, which leads to

$$\lim_{k'\to+\infty} L(Z_{k'}) = \lim_{k'\to+\infty} L(Z_{k'+1}) = L\bigl(C^1, D^1\bigr). \tag{29}$$

By the definition of C_k in (23), we have

$$L(C_{k'+1}, D_{k'+1}) + a\|C_{k'+1} - C_{k'}\|_F^2 \le L(C, D_{k'+1}) + b\|C - C_{k'}\|_F^2, \quad \forall C \in \Omega_C.$$

Taking $k' \to +\infty$ in the inequality above, together with (29) and (27), we have

$$L\bigl(C^1, D^1\bigr) \le L\bigl(C^1 + C, D^1\bigr) + b\|C\|_F^2, \quad \forall C \in \Omega_C. \tag{30}$$

Again, by the definition of (23), we have

$$\begin{cases} L(C_{k'}, D_{k'+1}) + \lambda_{k'}\|D_{k'+1} - D_{k'}\|_F^2 \le L(C_{k'}, D) + \lambda_{k'}\|D - D_{k'}\|_F^2; \\ L(C_{k'+1}, D_{k'+1}) + \mu_{k'}\|C_{k'+1} - C_{k'}\|_F^2 \le L(C, D_{k'+1}) + \mu_{k'}\|C - C_{k'}\|_F^2. \end{cases} \tag{31}$$

Recall that $\lambda_{k'}, \mu_{k'} \in (a, b)$. Then,

$$L(Z_{k'+1}) + a\|Z_{k'+1} - Z_{k'}\|_F^2 \le L(C_{k'}, D) + b\|D - D_{k'}\|_F^2, \quad \forall D \in \Omega_D.$$

Taking $k' \to +\infty$ in the above, together with (29) and (27), we have

$$L\bigl(C^1, D^1\bigr) \le L\bigl(C^1, D^1 + D\bigr) + b\|D\|_F^2, \quad \forall D \in \Omega_D.$$

Consequently, for any $\boldsymbol{d} = (\delta_C, \delta_D) \in (\Omega_C, \Omega_D)$, we have

$$\begin{aligned}
&\liminf_{\|\boldsymbol{d}\|\to 0} \frac{L(Z^1 + \boldsymbol{d}) - L(Z^1)}{\|\boldsymbol{d}\|} \\
&\quad = \liminf_{\|\boldsymbol{d}\|\to 0} \frac{Q(Z^1 + \boldsymbol{d}) - Q(Z^1) + f(C^1 + \delta_C) - f(C^1) + g(D^1 + \delta_D) - g(D^1)}{\|\boldsymbol{d}\|} \\
&\quad \ge \liminf_{\|\boldsymbol{d}\|\to 0} \frac{\langle \nabla Q(Z^1), \boldsymbol{d}\rangle + f(C^1 + \delta_C) - f(C^1) + g(D^1 + \delta_D) - g(D^1)}{\|\boldsymbol{d}\|} \\
&\quad = \liminf_{\|\boldsymbol{d}\|\to 0} \biggl(\frac{Q(C^1 + \delta_C, D^1) - Q(C^1, D^1) - o(\|\delta_C\|) + f(C^1 + \delta_C) - f(C^1)}{\|\boldsymbol{d}\|} \\
&\qquad + \frac{Q(C^1, D^1 + \delta_D) - Q(C^1, D^1) - o(\|\delta_D\|) + g(D^1 + \delta_D) - g(D^1)}{\|\boldsymbol{d}\|} \biggr) \\
&\quad \ge \liminf_{\|\boldsymbol{d}\|\to 0} \frac{-o(\|\delta_C\|) - o(\|\delta_D\|) - b(\|\delta_C\|_F^2 + \|\delta_D\|_F^2)}{\|\boldsymbol{d}\|} = 0.
\end{aligned}$$

By Definition 3.1, we have that Z^1 is a stationary point of (14). $\square$

Next, we will establish the convergence of the sequence $Z_k = (C_k, D_k)$ generated by (23) by showing that it satisfies the so-called *finite length property*, i.e.,

$$\sum_{k=1}^{+\infty} \|Z_{k+1} - Z_k\|_F < +\infty.$$

Clearly, a sequence with finite length property is a Cauchy sequence. Together with Theorem 4.2, we have the sequence Z_k converging to a stationary point of (14). The proof is based on the convergence analysis developed in a series of papers [18,17,19], which studied the convergence of the iteration scheme (23) for solving (14) with respect to a class of objective functions.

Theorem 4.3. *(See [17, Theorem 9].) The sequence $Z_k = (C_k, D_k)$ generated by the iteration (23) has* finite length property *if the following conditions hold:*

1. *$L(C, D)$ is a K–L function;*
2. *$Z_k, k = 1, 2, \dots$ is a bounded sequence and there exists some positive constants a, b such that $\lambda_k, \mu_k \in (a, b)$, $k = 1, 2, \dots$;*
3. *$\nabla Q(C, D)$ has Lipschitz constant on any bounded set.*

In Theorem 4.3, there are three conditions to ensure that the sequence satisfies the finite length property. The first condition requires that the objective function L satisfies the so-called *Kurdyka–Lojasiewicz (K–L)* property in its effective domain; see [19, Definition 3] for more details on K–L property. Given a function, it is often not easy to check whether it satisfies the K–L property. Nevertheless, it is shown in [18, Remark 5] and [18, Theorem 11] that any so-called *semi-algebraic* function satisfies the K–L property.

Definition 4.4. (See [19].) A subset S of $\mathbb{R}^n$ is called a semi-algebraic set if there exists a finite number of real polynomial functions g_{ij}, h_{ij} such that

$$S = \bigcup_j \bigcap_i \{\boldsymbol{u} \in \mathbb{R}^n : g_{ij}(\boldsymbol{u}) = 0, h_{ij}(\boldsymbol{u}) < 0\}.$$

A function $f(\boldsymbol{u})$ is called a semi-algebraic function if its graph $\{(\boldsymbol{u}, t) \in \mathbb{R}^n \times \mathbb{R}, t = f(\boldsymbol{u})\}$ is a semi-algebraic set.

Theorem 4.5. *Let $Z_k = (C_k, D_k)$ denote the sequence generated by (23). Then, the sequence Z_k has the finite length property and thus is a Cauchy sequence.*

Proof. The proof is done by showing that Theorem 4.3 is applicable to the objective function (14) and the sequence Z_k generated by (23). Thus, we only need to verify all three conditions in Theorem 4.3.

The first condition in Theorem 4.3 is verified by showing that all three terms in the objective function L given by (14) are semi-algebraic functions. The second term $Q(C, D) = \frac{1}{2}\|D^\top Y - C\|_F^2$ is clearly a semi-algebraic function as it is a real polynomial. Next, it can be seen that the set $\mathcal{X} = \{D \in \mathbb{R}^{m^2 \times m^2} : D^\top D = I\} = \bigcap_{j=1}^m \bigcap_{k=1}^m \{D : \sum_{i=1}^m \boldsymbol{d}_{ki}\boldsymbol{d}_{ji} = \delta_{j,k}\}$ is a semi-algebraic set. Thus, the last term $g(D) = I_{\mathcal{X}}(D)$ is also a semi-algebraic function, as it is shown in [20] that indicator functions of semi-algebraic sets are semi-algebraic functions. Regarding the first term $f(C) = \lambda^2\|C\|_0$, the graph of $F = \|C\|_0$ is $S = \bigcup_{k=0}^{m^2 L} L_k \triangleq \{(C, k) : \|C\|_0 = k\}$. For each $k = 0, \cdots, m^2 L$, let $\mathcal{S}_k = \{J : J \subseteq \{1, \cdots, m^2 L\}, |J| = k\}$, then $L_k = \bigcup_{J \in \mathcal{S}_k} \{(C, k) : C_{J^c} = 0, C_J \neq 0\}$. It can be seen that the set $\{(C, k) : C_{J^c} = 0, C_J \neq 0\}$ is a semi-algebraic set in $\mathbb{R}^{m^2 \times L} \times \mathbb{R}$. Thus, $F(C) = \|C\|_0$ is a semi-algebraic function, as the finite union of the semi-algebraic set is still semi-algebraic.

Regarding the second condition in Theorem 4.3, the boundedness of the sequence $Z_k = (C_k, D_k)$ is ensured by Theorem 4.2. Moreover, by the definition of (23), there exist two positive constants $a, b > 0$ such that $\lambda_k, \mu_k \in (a, b)$ for $k = 1, 2, \dots$.

Barbara Boat Couple Fingerprint Hill Lena

Fig. 2. Six test images.

For the last condition in Theorem 4.3, notice that the function $Q(C, D) = \frac{1}{2}\|C - D^\top Y\|_F^2$ is a smooth function. Thus, for any bounded set $\mathcal{M}$, there exists a constant $M > 0$ such that

$$\big\|\nabla Q(C_1, D_1) - \nabla Q(C_2, D_2)\big\| \le M\big\|(C_1, D_1) - (C_2, D_2)\big\|$$

for any $(C_1, D_1) \in \mathcal{M}$ and $(C_2, D_2) \in \mathcal{M}$. $\square$

In summary, we have the following result regarding the convergence of Algorithm 2.

Corollary 4.6. *The sequence $Z_k := (C_k, D_k)$ generated by Algorithm 2 converges to a stationary point of (14).*

5. Experiments on image denoising

There are two main parts in this paper: one is the convergence analysis of the method proposed in [1] and the other is the modifications of the original algorithm for gaining stronger convergence property. The later is more of theoretical interest and for potential benefit to other applications. Thus, the experimental evaluation done in this paper for image denoising is not as comprehensive as in [1]. The data-driven tight frame based image denoising is done as follows. Let $f = g + \epsilon(\sigma)$ denote some noisy observation of g, where $\epsilon(\sigma)$ is the additive i.i.d. Gaussian noise with zero mean and standard deviation σ. Taking f as the input and using 8×8 DCT as the initial guess, the filters of data-drive tight frame $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_{64}\}$ are constructed using Algorithm 1 (or Algorithm 2). Then the denoised result, denoted by $\widetilde{g}$, is obtained via hard thresholding:

$$\widetilde{g} = \boldsymbol{W}^\top\big(T_{\tilde{\lambda}}(\boldsymbol{W}f)\big),$$

where $\boldsymbol{W}$ denotes the analysis operator determined by $\{\boldsymbol{a}_j\}_{j=1}^{64}$ and $\tilde{\lambda}$ is thresholding parameter determined by noise level. Throughout all experiments, the parameter $\tilde{\lambda}$ is fixed at $\tilde{\lambda} = 2.7\sigma$ for both Algorithm 1 and Algorithm 2. The other settings for Algorithm 1 are the same as in [1]. For Algorithm 2, we set the maximum number of iterations to 70 and set $\lambda_k = 0.047$, $\mu_k = 0.024$ for all k.

We start with the demonstration of convergence behavior of Algorithm 1 proposed in [1] and Algorithm 2 proposed in this paper. See Fig. 1(a) for the comparison of the ℓ_2 norm of the increments of the frame coefficient vectors C^k generate by two algorithms. It can be seen that the coefficient sequence generated by Algorithm 1 does not converge while the one generated by Algorithm 1 converges. However, the lack of sequence convergence of Algorithm 1 does not impact its performance of image denoising, as shown in Fig. 1. The PSNR values of the denoised results from both algorithms are summarized in Table 1 with respect to different images (see Fig. 2) and different noise levels. It can be seen that the performances of both algorithms in image denoising are very close in terms of PSNR value.

Table 1
PSNR values of the denoised results.

Image	Barbara					Boat				
σ	10	20	30	40	50	10	20	30	40	50
Algorithm 1; 8	34.36	30.60	28.42	26.88	25.67	33.62	30.38	28.39	27.06	25.99
Algorithm 1; 16	34.63	31.07	29.07	27.60	26.48	33.59	30.41	28.45	27.18	26.08
Algorithm 2; 8	34.34	30.58	28.34	26.89	25.74	33.61	30.29	28.39	26.94	25.87
Algorithm 2; 16	34.63	31.14	29.02	27.58	26.41	33.58	30.39	28.48	27.16	26.13
Image	Fingerprint					Hill				
σ	10	20	30	40	50	10	20	30	40	50
Algorithm 1; 8	32.23	28.32	26.18	24.67	23.52	33.28	30.22	28.56	27.36	26.48
Algorithm 1; 16	32.25	28.40	26.34	24.95	23.88	33.28	30.30	28.61	27.52	26.63
Algorithm 2; 8	32.20	28.27	26.13	24.66	23.46	33.26	30.20	28.45	27.25	26.38
Algorithm 2; 16	32.24	28.38	26.33	24.93	23.87	33.22	30.23	28.64	27.50	26.65
Image	Couple					Lena				
Algorithm 1; 8	33.63	30.09	28.16	26.72	25.68	35.52	32.25	30.22	28.80	27.60
Algorithm 1; 16	33.55	30.19	28.27	26.95	25.87	35.65	32.56	30.58	29.16	28.14
Algorithm 2; 8	33.49	30.05	28.02	26.64	25.61	35.47	32.29	30.25	28.77	27.57
Algorithm 2; 16	33.52	30.10	28.25	26.93	25.89	35.64	32.53	30.51	29.16	28.06

Acknowledgments

The authors would like to thank the associated editor and the reviewer for their helpful comments and suggestions. This work was partially supported by Singapore MOE AcRF Research Grant MOE2011-T2-1-116 and R-146-000-165-112.

References

[1] J.-F. Cai, H. Ji, Z. Shen, G.-B. Ye, Data-driven tight frame construction and image denoising, Appl. Comput. Harmon. Anal. 1 (37) (2014) 89–105.
[2] K. Rao, P. Yip, Discrete Cosine Transform: Algorithms, Advantages and Applications, Academic Press, 1990.
[3] I. Daubechies, Ten Lectures on Wavelets, vol. 61, SIAM, Philadelphia, 1992.
[4] S. Mallat, A Wavelet Tour of Signal Processing: The Sparse Way, 3rd edition, Academic Press, 2008.
[5] R. Coifman, D. Donoho, Translation-invariant de-noising, in: Wavelet and Statistics, in: Springer Lecture Notes in Statistics, vol. 103, Springer-Verlag, 1994, pp. 125–150.
[6] A. Ron, Z. Shen, Affine systems in $L_2(R^d)$: the analysis of the analysis operator, J. Funct. Anal. 148 (1997) 408–447.
[7] I. Daubechies, B. Han, A. Ron, Z. Shen, Framelets: MRA-based constructions of wavelet frames, Appl. Comput. Harmon. Anal. 14 (2003) 1–46.
[8] E. Candes, D.L. Donoho, New tight frames of curvelets and optimal representations of objects with piecewise-C^2 singularities, Comm. Pure Appl. Math. 57 (2002) 219–266.
[9] M.S. Lewicki, T.J. Sejnowski, Learning overcomplete representations, Neural Comput. 12 (2) (2000) 337–365.
[10] M. Aharon, M.E. Bruckstein, K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (11) (2006) 4311–4322.
[11] J. Mairal, M. Elad, G. Sapiro, Sparse representation for color image restoration, IEEE Trans. Image Process. 17 (1) (2008) 53–69.
[12] Z. Shen, Wavelet frames and image restorations, in: Proceedings of the International Congress of Mathematicians, vol. 4, 2010, pp. 2834–2863.
[13] J.-F. Cai, B. Dong, S. Osher, Z. Shen, Image restoration: total variation, wavelet frames, and beyond, J. Amer. Math. Soc. 25 (4) (2012) 1033–1089.
[14] R.H. Chan, T.F. Chan, L. Shen, Z. Shen, Wavelet algorithms for high-resolution image reconstruction, SIAM J. Sci. Comput. 24 (2003) 1408–1432.
[15] R.H. Chan, S. Reimenschneider, L.S.Z. Shen, Tight frame: an efficient way for high-resolution image reconstruction, Appl. Comput. Harmon. Anal. 17 (2004) 91–115.
[16] P. Tseng, Convergence of a block coordinate descent method for nondifferentiable minimization, J. Optim. Theory Appl. 109 (3) (2001) 475–494.
[17] H. Attouch, J. Bolte, P. Redont, A. Soubeyran, Proximal alternating minimization and projection methods for nonconvex problems: an approach based on the Kurdyka–Łojasiewicz inequality, Math. Oper. Res. 35 (2) (2010) 438–457.
[18] J. Bolte, A. Daniilidis, A. Lewis, M. Shiota, Clarke subgradients of stratifiable functions, SIAM J. Optim. 18 (2) (2007) 556–572.
[19] J. Bolte, S. Sabach, M. Teboulle, Proximal alternating linearized minimization for nonconvex and nonsmooth problems, Math. Program. (2013) 1–36.
[20] H. Attouch, J. Bolte, On the convergence of the proximal algorithm for nonsmooth functions involving analytic features, Math. Program. 116 (2009) 5–16.