

# Efficient Blind Compressed Sensing Using Sparsifying Transforms with Convergence Guarantees and Application to MRI \*

Saiprasad Ravishankar and Yoram Bresler<sup>†</sup>

**Abstract.** Natural signals and images are well-known to be approximately sparse in transform domains such as Wavelets and DCT. This property has been heavily exploited in various applications in image processing and medical imaging. Compressed sensing exploits the sparsity of images or image patches in a transform domain or synthesis dictionary to reconstruct images from undersampled measurements. In this work, we focus on blind compressed sensing, where the underlying sparsifying transform is a priori unknown, and propose a framework to simultaneously reconstruct the underlying image as well as the sparsifying transform from highly undersampled measurements. The proposed block coordinate descent type algorithms involve highly efficient optimal updates. Importantly, we prove that although the proposed blind compressed sensing formulations are highly nonconvex, our algorithms are *globally convergent* (i.e., they converge from any initialization) to the set of critical points of the objectives defining the formulations. These critical points are guaranteed to be at least partial global and partial local minimizers. The exact point(s) of convergence may depend on initialization. We illustrate the usefulness of the proposed framework for magnetic resonance image reconstruction from highly undersampled k-space measurements. As compared to previous methods involving the synthesis dictionary model, our approach is much faster, while also providing promising reconstruction quality.

**Key words.** Sparsifying transforms, Inverse problems, Compressed sensing, Medical imaging, Magnetic resonance imaging, Sparse representation, Dictionary learning.

**AMS subject classifications.** 68T05, 65F50, 94A08

**1. Introduction.** Sparsity-based techniques have become extremely popular in various applications in image processing and imaging in recent years. These techniques typically exploit the sparsity of images or image patches in a transform domain or dictionary to compress [47], denoise, or restore/reconstruct images. In this work, we investigate the subject of blind compressed sensing, which aims to reconstruct images in the scenario when a good sparse model for the image is unknown a priori. In particular, we will work with the classical sparsifying transform model [70] that has been shown to be useful in various imaging applications. In the following, we briefly review the topics of sparse modeling, compressed sensing, and blind compressed sensing. We then list the contributions of this work.

**1.1. Sparse Models.** Sparsity-based techniques in imaging and image processing typically rely on the existence of a good sparse model for the data. Various models are known to provide sparse representations of data such as the *synthesis dictionary model* [24, 7], and the *sparsifying transform model* [70, 56].

The synthesis dictionary model suggests that a signal  $y \in \mathbb{C}^n$  can be sparsely represented

\*This work was supported in part by the National Science Foundation (NSF) under grants CCF-1018660 and CCF-1320953. A shorter version of this work appears elsewhere [72].

<sup>†</sup>S. Ravishankar and Y. Bresler are with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois, Urbana-Champaign, IL, 61801 USA (([ravisha3](mailto:ravisha3@illinois.edu), [ybresler](mailto:ybresler@illinois.edu))@illinois.edu). Questions, comments, or corrections to this document may be directed to that email address.

as a linear combination of a small number of atoms or columns from a synthesis dictionary  $D \in \mathbb{C}^{n \times K}$ , i.e.,  $y = Dx$  with  $x \in \mathbb{C}^K$  being sparse, i.e.,  $\|x\|_0 \ll K$ . The  $l_0$  quasi norm measures the sparsity of  $x$  by counting the number of non-zeros in  $x$ . Real-world signals usually satisfy  $y = Dx + e$ , where  $e$  is the approximation error in the signal domain [7]. The well-known *synthesis sparse coding* problem finds  $x$  that minimizes  $\|y - Dx\|_2^2$  subject to  $\|x\|_0 \leq s$ , where  $s$  is a given sparsity level. A disadvantage of the synthesis model is that sparse coding is typically NP-hard (Non-deterministic Polynomial-time hard) [50, 17], and the various approximate sparse coding algorithms [53, 46, 16, 12, 32, 22] tend to be computationally expensive for large scale problems.

The alternative sparsifying transform model suggests that a signal  $y$  is approximately sparsifiable using a transform  $W \in \mathbb{C}^{m \times n}$ . The assumption here is that  $Wy = x + \eta$ , where  $x \in \mathbb{C}^m$  is sparse in some sense, and  $\eta$  is a small residual error in the *transform domain* rather than in the signal domain. Natural signals and images are known to be approximately sparsifiable by analytical transforms such as the discrete cosine transform (DCT), Wavelets [45], finite differences, and Contourlets [18]. The advantage of the transform model over the synthesis dictionary model <sup>1</sup> is that sparse coding can be performed exactly and cheaply by zeroing out all but a certain number of nonzero transform coefficients of largest magnitude [70].

**1.2. Compressed Sensing.** In the context of imaging, the recent theory of Compressed Sensing (CS) [8, 19, 9] (see also [28, 5, 27, 79, 6, 29, 84, 4] for the earliest versions of CS for Fourier-sparse signals and for Fourier imaging) enables accurate recovery of images from significantly fewer measurements than the number of unknowns. In order to do so, it requires that the underlying image be sufficiently sparse in some transform domain or dictionary, and that the measurement acquisition procedure be incoherent, in an appropriate sense, with the transform. However, the image reconstruction procedure for compressed sensing is typically computationally expensive and non-linear.

The image reconstruction problem in compressed sensing is typically formulated as follows

$$(1.1) \quad \min_x \|\Psi x\|_0 \quad s.t. \quad Ax = y$$

Here,  $x \in \mathbb{C}^p$  is a vectorized representation of the image (obtained by stacking the image columns on top of each other) to be reconstructed, and  $y \in \mathbb{C}^m$  denotes the measurements. The operator  $A \in \mathbb{C}^{m \times p}$ , with  $m \ll p$  is known as the sensing matrix, or measurement matrix. The matrix  $\Psi \in \mathbb{C}^{t \times p}$  is a sparsifying transform (typically chosen as orthonormal). The aim of Problem (1.1) is to find the image satisfying the measurement equation  $Ax = y$ , that is the sparsest possible in the  $\Psi$ -transform domain. Since, in CS, the measurement equation  $Ax = y$  represents an underdetermined system of equations, an additional model (such as the sparsity model above) is needed to estimate the true underlying image.

When  $\Psi$  is orthonormal, Problem (1.1) can be rewritten as

$$(1.2) \quad \min_z \|z\|_0 \quad s.t. \quad A\Psi^H z = y$$

---

<sup>1</sup>The sparsifying transform model enjoys similar advantages over other models such as the noisy signal analysis dictionary model [70], which is not discussed here for reasons of space.

where we used the substitution  $\Psi x = z$ , and  $(\cdot)^H$  denotes the matrix Hermitian (conjugate transpose) operation. Similar to the synthesis sparse coding problem, Problem (1.2) too is NP-hard. Often the  $l_0$  quasi norm in (1.1) is replaced with its convex relaxation, the  $l_1$  norm [20], and the following convex problem is solved to reconstruct the image, when the CS measurements are noisy [38, 21].

$$(1.3) \quad \min_x \|Ax - y\|_2^2 + \lambda \|\Psi x\|_1$$

In Problem (1.3), the  $\ell_2$  penalty for the measurement fidelity term can also be replaced with alternative penalties such as a weighted  $\ell_2$  penalty, depending on the physics of the measurement process and the statistics of the measurement noise.

Recently, CS theory has been applied to imaging techniques such as magnetic resonance imaging (MRI) [38, 39, 10, 76, 34, 58, 62], computed tomography (CT) [11, 13, 35], and Positron emission tomography (PET) imaging [78, 43], demonstrating high quality reconstructions from a reduced set of measurements. Such compressive measurements are highly advantageous in these applications. For example, they help reduce the radiation dosage in CT, and reduce scan times and improve clinical throughput in MRI. Well-known inverse problems in image processing such as inpainting (where an image is reconstructed from a subset of measured pixels) can also be viewed as compressed sensing problems.

**1.3. Blind Compressed Sensing.** While conventional compressed sensing techniques utilize fixed analytical sparsifying transforms such as wavelets [45], finite differences, and contourlets [18], to reconstruct images, in this work, we instead focus on the idea of blind compressed sensing (BCS) [64, 65, 30, 36, 80], where the underlying sparse model is assumed unknown a priori. The goal of blind compressed sensing is to simultaneously reconstruct the underlying image(s) as well as the dictionary or transform from highly undersampled measurements. Thus, BCS enables the sparse model to be adaptive to the specific data under consideration. Recent research has shown that such data-driven adaptation of dictionaries or transforms is advantageous in many applications [23, 41, 1, 42, 85, 64, 69, 66, 73, 81]. While the adaptation of synthesis dictionaries [52, 26, 2, 83, 75, 40] has been extensively studied, recent work has shown advantages in terms of computation and application-specific performance, for the adaptation of transform models [70, 73, 81].

In a prior work on BCS [64], we successfully demonstrated the usefulness of dictionary-based blind compressed sensing for MRI, even in the case when the undersampled measurements corresponding to only a single image are provided. In the latter case, the overlapping patches of the underlying image are assumed to be sparse in a dictionary, and the (unknown) patch-based dictionary, that is typically much smaller in size than the image, is learnt directly from the compressive measurements.

BCS techniques have been demonstrated to provide much better image reconstruction quality compared to compressed sensing methods that utilize a fixed sparsifying transform or dictionary [64, 36, 80]. This is not surprising since BCS methods allow for data-specific adaptation, and data-specific dictionaries typically sparsify the underlying images much better than analytical ones.

#### 1.4. Contributions.

**1.4.1. Highlights.** The BCS framework assumes a particular class of sparse models for the underlying image(s) or image patches. While prior work on BCS primarily focused on the synthesis dictionary model, in this work, we instead focus on the sparsifying transform model. We propose novel problem formulations for BCS involving well-conditioned or orthonormal adaptive *square* sparsifying transforms. Our framework simultaneously adapts the sparsifying transform and reconstructs the underlying image(s) from highly undersampled measurements. We propose efficient block coordinate descent-type algorithms for transform-based BCS. Importantly, we establish that our iterative algorithms are *globally convergent* (i.e., they converge from any initialization) to the set of critical points of the proposed highly non-convex BCS cost functions. These critical points are guaranteed to be *at least* partial global and partial local minimizers. The exact point(s) of convergence may depend on initialization. Such convergence guarantees have not been established for prior *blind* compressed sensing methods.

Note that although we focus on compressed sensing in the discussions and experiments of this work, the formulations and algorithms proposed by us can also handle the case when the measurement or sensing matrix  $A$  is square (e.g., in signal denoising), or even tall (e.g., deconvolution).

**1.4.2. Magnetic Resonance Imaging Application.** MRI is a non-invasive and non-ionizing imaging technique that offers a variety of contrast mechanisms, and enables excellent visualization of anatomical structures and physiological functions. However, the data in MRI, which are samples in k-space of the spatial Fourier transform of the object, are acquired sequentially in time. Hence, a major drawback of MRI, that affects both clinical throughput and image quality especially in dynamic imaging applications, is that it is a relatively slow imaging technique. Although there have been advances in scanner hardware [57] and pulse sequences, the rate at which MR data are acquired is limited by MR physics and physiological constraints on RF energy deposition. Compressed sensing MRI (either blind, or with known sparse model) has become quite popular in recent years, and it alleviates some of the aforementioned problems by enabling accurate reconstruction of MR images from highly undersampled measurements.

In this work, we illustrate the usefulness of the proposed transform-based BCS schemes for magnetic resonance image reconstruction from highly undersampled k-space data. We show that our adaptive transform-based BCS provides better image reconstruction quality compared to prior methods that involve fixed image-based, or patch-based sparsifying transforms. Importantly, transform-based BCS is shown to be more than 10x faster than synthesis dictionary-based BCS [64] for reconstructing 2D MRI data. The speedup is expected to be much higher when considering 3D or 4D MR data. These advantages make the proposed scheme more amenable for adoption for clinical use in MRI.

**1.5. Organization.** The rest of this paper is organized as follows. Section 2 describes our transform learning-based blind compressed sensing formulations and their properties. In Section 3, we derive efficient block coordinate descent algorithms for solving the BCS Problems, and discuss the algorithms' computational costs. In Section 4, we present novel convergence guarantees for our algorithms. The proof of convergence is provided in the Appendix. Section

5 presents experimental results demonstrating the convergence behavior, performance, and computational efficiency of the proposed scheme for the MRI application. In Section 6, we conclude with proposals for future work.

## 2. Problem Formulations.

**2.1. Synthesis dictionary-based Blind Compressed Sensing.** The compressed sensing-based image reconstruction Problem (1.3) can be viewed as a particular instance of the following constrained regularized inverse problem, with  $\zeta(x) = \lambda \|\Psi x\|_1$ ,  $\nu = 1$ , and  $\mathcal{S} = \mathbb{C}^p$

$$(2.1) \quad \min_{x \in \mathcal{S}} \nu \|Ax - y\|_2^2 + \zeta(x)$$

However, CS image reconstructions employing fixed, non-adaptive sparsifying transforms typically suffer from many artifacts at high undersampling factors [64]. Blind compressed sensing allows for the sparse model to be directly adapted to the object(s) being imaged. For example, the overlapping patches of the underlying image may be assumed to be sparse in a certain adaptive model. In prior work [64], the following patch-based dictionary learning regularizer was used within Problem (2.1) along with  $\mathcal{S} = \mathbb{C}^p$

$$(2.2) \quad \zeta(x) = \min_{D, B} \sum_{j=1}^N \|P_j x - D b_j\|_2^2 \quad \text{s.t.} \quad \|d_k\|_2 = 1 \quad \forall k, \quad \|b_j\|_0 \leq s \quad \forall j.$$

The resulting synthesis dictionary based BCS formulation is as follows

$$(P0) \quad \min_{x, D, B} \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|P_j x - D b_j\|_2^2 \quad \text{s.t.} \quad \|d_k\|_2 = 1 \quad \forall k, \quad \|b_j\|_0 \leq s \quad \forall j.$$

Here,  $\nu > 0$  is a weight for the measurement fidelity term ( $\|Ax - y\|_2^2$ ), and  $P_j \in \mathbb{C}^{n \times p}$  represents the operator that extracts a  $\sqrt{n} \times \sqrt{n}$  2D patch as a vector  $P_j x \in \mathbb{C}^n$  from the image  $x$ . A total of  $N$  overlapping 2D patches are used. The synthesis model allows each patch  $P_j x$  to be approximated by a linear combination  $D b_j$  of a small number of columns from a dictionary  $D \in \mathbb{C}^{n \times K}$ , where  $b_j \in \mathbb{C}^K$  is sparse. The columns of the learnt dictionary (represented by  $d_k, 1 \leq k \leq K$ ) in (P0) are additionally constrained to be of unit norm in order to avoid the scaling ambiguity [33]. The dictionary, and the image patch, are assumed to be much smaller than the image ( $n, K \ll p$ ) in (P0). Problem (P0) thus enforces all the  $N$  (a typically large number) overlapping image patches to be sparse in some dictionary  $D$ , which can be considered as a strong yet flexible prior on the underlying image.

We use  $B \in \mathbb{C}^{n \times N}$  to denote the matrix that has the sparse codes of the patches  $b_j$  as its columns. Each sparse code is permitted a maximum sparsity level of  $s \ll n$  in (P0). Although a single sparsity level is used for all patches in (P0) for simplicity, in practice, different sparsity levels may be allowed for different patches (for example, by setting an appropriate error threshold in the sparse coding step of optimization algorithms [64]). For the case of MRI, the sensing matrix  $A$  in (P0) is  $F_u \in \mathbb{C}^{m \times p}$ , the undersampled Fourier encoding matrix [64]. The weight  $\nu$  in (P0) is set depending on the measurement noise standard deviation  $\sigma$  as

$\nu = \frac{\theta}{\sigma}$ , where  $\theta$  is a positive constant [64]. In practice, if the noise level is unknown, it may be estimated.

Problem (P0) is to learn a patch-based synthesis sparsifying dictionary ( $n, K \ll p$ ), and reconstruct the image simultaneously from highly undersampled measurements. As discussed before, we have previously shown significantly superior image reconstructions for MRI using (P0), as compared to non-adaptive compressed sensing schemes that solve Problem (1.3). However, the BCS Problem (P0) is both non-convex and NP-hard. Approximate iterative algorithms for (P0) (e.g., the DLMRI algorithm [64]) typically solve the synthesis sparse coding problem repeatedly, which makes them computationally expensive. Moreover, no convergence guarantees exist for the algorithms that solve (P0).

## 2.2. Sparsifying Transform-based Blind Compressed Sensing.

**2.2.1. Problem Formulations with Sparsity Constraints.** In order to overcome some of the aforementioned drawbacks of synthesis dictionary-based BCS, we propose using the sparsifying transform model in this work. Sparsifying transform learning has been shown to be effective and efficient in applications, while also enjoying good convergence guarantees [73]. Therefore, we use the following transform learning regularizer [70]

$$(2.3) \quad \zeta(x) = \min_{W, B} \sum_{j=1}^N \|WP_j x - b_j\|_2^2 + \lambda Q(W) \quad s.t. \quad \|B\|_0 \leq s$$

along with the constraint set  $\mathcal{S} = \{x \in \mathbb{C}^p : \|x\|_2 \leq C\}$  within Problem (2.1) to arrive at the following adaptive sparsifying transform-based BCS problem formulation

$$(P1) \quad \min_{x, W, B} \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|WP_j x - b_j\|_2^2 + \lambda Q(W) \quad s.t. \quad \|B\|_0 \leq s, \quad \|x\|_2 \leq C.$$

Here,  $W \in \mathbb{C}^{n \times n}$  denotes a square sparsifying transform for the patches of the underlying image. The penalty  $\|WP_j x - b_j\|_2^2$  denotes the sparsification error (transform domain residual) [70] for the  $j^{\text{th}}$  patch, with  $b_j$  denoting the transform sparse code. The sparsity term  $\|B\|_0 = \sum_{j=1}^N \|b_j\|_0$  counts the number of non-zeros in the sparse code matrix  $B$ . Notice that the sparsity constraint in (P1) is enforced on all the overlapping patches, taken together. This is a way of enabling variable sparsity levels for each specific patch. The constraint  $\|x\|_2 \leq C$  with  $C > 0$  in (P1), is to enforce any prior knowledge on the signal energy (or, range). For example, if the pixels of the underlying image take intensity values in the range  $0 - 255$ , then  $C = 255\sqrt{p}$  is an appropriate bound. The function  $Q(W) : \mathbb{C}^{n \times n} \mapsto \mathbb{R}$  in Problem (P1) denotes a regularizer for the transform, and the weight  $\lambda > 0$ . Notice that without an additional regularizer,  $W = 0$  is a trivial sparsifier for any patch, and therefore,  $W = 0, b_j = 0 \forall j, x = A^\dagger y$  (assuming this  $x$  satisfies  $\|x\|_2 \leq C$ ) with  $(\cdot)^\dagger$  denoting the pseudo-inverse, would trivially minimize the objective (without the regularizer  $Q(W)$ ) in Problem (P1).

Similar to prior work on transform learning [70, 67], we set  $Q(W) \triangleq -\log |\det W| + 0.5 \|W\|_F^2$  as the regularizer in the objective to prevent trivial solutions. The  $-\log |\det W|$  penalty eliminates degenerate solutions such as those with repeated rows. The  $\|W\|_F^2$  penalty helps remove a ‘scale ambiguity’ in the solution [70], which occurs when the optimal solution



satisfies an exactly sparse representation, i.e., the optimal  $(x, W, B)$  in (P1) is such that  $WP_jx = b_j \forall j$ , and  $\|B\|_0 \leq s$ . In this case, if the  $\|W\|_F^2$  penalty is absent in (P1), the optimal  $(W, B)$  can be scaled by  $\beta \in \mathbb{C}$ , with  $|\beta| \rightarrow \infty$ , which causes the objective to decrease unbounded.

The  $-\log |\det W|$  and  $0.5 \|W\|_F^2$  penalties together also additionally help control the condition number  $\kappa(W)$  and scaling of the learnt transform. If we were to minimize only the  $Q(W)$  regularizer in Problem (P1) with respect to  $W$ , then the minimum is achieved with a  $W$  that is unit conditioned, and with spectral norm (scaling) of 1 [70], i.e., a *unitary* or *orthonormal* transform  $W$ . Thus, similar to Corollary 2 in [70], it is easy to show that as  $\lambda \rightarrow \infty$  in Problem (P1), the optimal sparsifying transform(s) tends to a unitary one. In practice, transforms learnt via (P1) are typically close to unitary even for finite  $\lambda$ . Adaptive well-conditioned transforms (small  $\kappa(W) > 1$ ) have been previously shown to perform better than adaptive (strictly) orthonormal ones in some scenarios in image representation, or image denoising [70, 67].

In this work, we set  $\lambda = \lambda_0 N$  in (P1), where  $\lambda_0 > 0$  is a constant. This setting allows  $\lambda$  to scale with the size of the data (i.e., total number of patches). In practice, the weight  $\lambda_0$  needs to be set according to the expected range (in intensity values) of the underlying image, as well as depending on the desired condition number of the learnt transform. The weight  $\nu$  in (P1) is set similarly as in (P0).

When a unitary sparsifying transform is preferred, the  $Q(W)$  regularizer in (P1) (and in (2.3)) could instead be replaced by the constraint  $W^H W = I$ , where  $I$  denotes the identity matrix, yielding the following formulation

$$(P2) \quad \min_{x, W, B} \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|WP_jx - b_j\|_2^2 \quad \text{s.t.} \quad W^H W = I, \|B\|_0 \leq s, \|x\|_2 \leq C.$$

The unitary sparsifying transform case is special, in that Problem (P2) is also a unitary synthesis dictionary-based blind compressed sensing problem, with  $W^H$  denoting the synthesis dictionary. This follows from the identity  $\|WP_jx - b_j\|_2 = \|P_jx - W^H b_j\|_2$ , for unitary  $W$ .

### 2.2.2. Properties of Transform BCS Formulations - Identifiability and Uniqueness.

The following simple proposition considers an “error-free” scenario and establishes the global identifiability of the underlying image and sparse model in BCS via solving the proposed Problems (P1) or (P2).

**Proposition 2.1.** *Let  $x \in \mathbb{C}^p$  with  $\|x\|_2 \leq C$ , and let  $y = Ax$  with  $A \in \mathbb{C}^{m \times p}$ . Suppose that  $W \in \mathbb{C}^{n \times n}$  is a unitary transform that sparsifies the collection of patches of  $x$  as  $\sum_{j=1}^N \|WP_jx\|_0 \leq s$ . Further, let  $B$  denote the matrix that has  $WP_jx$  as its columns. Then,  $(x, W, B)$  is a global minimizer of both Problems (P1) and (P2), i.e., it is identifiable by solving these problems.*

**Proof.** : For the given  $(x, W, B)$ , the terms  $\sum_{j=1}^N \|WP_jx - b_j\|_2^2$  and  $\|Ax - y\|_2^2$  in (P1) and (P2) each attain their minimum possible value (lower bound) of zero. Since  $W$  is unitary, the penalty  $Q(W)$  in (P1) is also minimized by the given  $W$ . Notice that the constraints in both (P1) and (P2) are satisfied for the given  $(x, W, B)$ . Therefore, this triplet is feasible for both problems and achieves the minimum possible value of the objective in both cases. Thus, it is a global minimizer of both (P1) and (P2). ■

Thus, when “error-free” measurements are provided, and the patches of the underlying image are *exactly* sparse (as defined by the constraint in (P1)) in some unitary transform, Proposition 2.1 guarantees that the image as well as the model are jointly identifiable by solving (i.e., finding global minimizers in) (P1) (or, (P2)).

An interesting topic, which we do not fully pursue here pertains to the condition(s) under which the underlying image in Proposition 2.1 is the unique minimizer of the proposed BCS problems. The proposed problems do admit an equivalence class of solutions/minimizers with respect to the transform  $W$  and the set of sparse codes  $\{b_j\}_{j=1}^N$ <sup>2</sup>. Given a particular minimizer  $(x, W, B)$  of (P1) or (P2), we have that  $(x, \Theta W, \Theta B)$  is another equivalent minimizer for all *sparsity-preserving unitary matrices*  $\Theta$ , i.e.,  $\Theta$  such that  $\Theta^H \Theta = I$  and  $\sum_j \|\Theta b_j\|_0 \leq s$ . For example,  $\Theta$  can be a row permutation matrix, or a diagonal  $\pm 1$  sign matrix. Importantly however, the minimizer with respect to  $x$  in (P1) or (P2) is invariant to the modification of  $(W, B)$  by sparsity-preserving unitary matrices  $\Theta$ , i.e., the optimal  $x$  in (P1) or (P2) remains the same for all such choices of  $(\Theta W, \Theta B)$ .

We note that by imposing additional structure on  $W$  in our transform-based BCS formulations, one can derive conditions for the uniqueness of the minimizers. Assume a global minimizer  $(x, W, B)$  exists in (P2) satisfying the conditions (error-free scenario) in Proposition 2.1. Then, for example, when the unitary transform  $W$  in (P2) is further constrained to be doubly sparse, i.e.,  $W = S\Phi$ , with  $S$  a sparse matrix and  $\Phi$  a known matrix (e.g., DCT, or Wavelet  $\Phi$ ), then because  $W^H = \Phi^H S^H$  is an equivalent (doubly sparse) synthesis dictionary (corresponding to the transform  $W$ ), the uniqueness conditions (involving the spark condition) proposed in prior work on synthesis dictionary-based BCS [30] (Section V-A of [30]) can be extended to the transform-based setting here. A detailed analysis and description of such uniqueness results will be presented elsewhere.

**2.2.3. Problem Formulations with Sparsity Penalties.** While Problem (P1) involves a sparsity constraint, an alternative version of Problem (P1) is obtained by replacing the  $\ell_0$  sparsity constraint with an  $\ell_0$  penalty in the objective (and in (2.3)), in which case we have the following optimization problem

$$(P3) \quad \min_{x, W, B} \sum_{j=1}^N \|W P_j x - b_j\|_2^2 + \nu \|Ax - y\|_2^2 + \lambda Q(W) + \eta^2 \|B\|_0 \quad \text{s.t.} \quad \|x\|_2 \leq C.$$

where  $\eta^2$ , with  $\eta > 0$ , denotes the weight for the sparsity penalty.

A version of Problem (P3) (without the  $\ell_2$  constraint) has been used very recently in adaptive tomographic reconstruction [54, 55]. However, it is interesting to note that in the absence of the  $\|x\|_2 \leq C$  condition, the objective in (P3) is actually non-coercive. To see this, consider  $W = I$  and  $x_\beta = x_0 + \beta z$ , where  $x_0$  is a particular solution to  $y = Ax$ ,  $\beta \in \mathbb{R}$ , and  $z \in \mathcal{N}(A)$  with  $\mathcal{N}(A)$  denoting the null space of  $A$ . For this setting, as  $\beta \rightarrow \infty$  with the  $j^{\text{th}}$  sparse code in (P3) set to  $W P_j x_\beta$ , it is obvious that the objective in (P3) remains finite, thereby making it non-coercive. The energy constraint on  $x$  in (P3) restricts the set of feasible

---

<sup>2</sup>In the remainder of this work, when certain indexed variables are enclosed within braces, it means that we are considering the set of variables over the range of all the indices.



images to a compact set, and alleviates potential problems (such as unbounded iterates within a minimization algorithm) due to the non-coercive objective.

While a single weight  $\eta^2$  is used for the sparsity penalty  $\|B\|_0 = \sum_{j=1}^N \|b_j\|_0$  in (P3), one could also use different weights  $\eta_j^2$  for the sparsity penalties  $\|b_j\|_0$  corresponding to different patches, if such weights are known, or estimated.

Just as Problem (P3) is an alternative to Problem (P1), we can also obtain a corresponding alternative version (denoted as (P4)) of Problem (P2) by replacing the sparsity constraint with a penalty. Although, in the rest of this paper, we consider Problems (P1)-(P3), the proposed algorithms and convergence results in this work easily extend to the case of (P4).

**2.3. Extensions.** While the proposed sparsifying transform-based BCS problem formulations are for the (extreme) scenario when the CS measurements corresponding to a single image are provided, these formulations can be easily extended to other scenarios too. For example, when multiple images (or frames, or slices) have to be jointly reconstructed using a single adaptive (spatial) sparsifying transform, then the objectives in Problems (P1)-(P3) for this case are the summation of the corresponding objective functions for each image. In applications such as dynamic MRI (or for example, compressive video), the proposed formulations can be extended by considering adaptive spatiotemporal sparsifying transforms of 3D patches (cf. [80] that extends Problem (P0) in such a way to compressed sensing dynamic MRI). Similar extensions are also possible for higher-dimensional applications such as 4D imaging.

### 3. Algorithm and Properties.

**3.1. Algorithm.** Here, we propose block coordinate descent-type algorithms to solve the proposed transform-based BCS problem formulations (P1)-(P3). Our algorithms alternate between solving for the sparse codes  $\{b_j\}$  (*sparse coding step*), transform  $W$  (*transform update step*), and image  $x$  (*image update step*), with the other variables kept fixed. One could also alternate a few times between the sparse coding and transform update steps, before performing one image update step. In the following, we describe the three main steps in detail. We show that each of the steps has a simple solution that can be computed cheaply in practical applications such as MRI.

**3.1.1. Sparse Coding Step.** The sparse coding step of our algorithms for Problems (P1) and (P2) involves the following optimization problem

$$(3.1) \quad \min_B \sum_{j=1}^N \|WP_j x - b_j\|_2^2 \quad s.t. \quad \|B\|_0 \leq s.$$

Now, let  $Z \in \mathbb{C}^{n \times N}$  be the matrix with the transformed (vectorized) patches  $WP_j x$  as its columns. Then, using this notation, Problem (3.1) can be rewritten as follows, where  $\|\cdot\|_F$  denotes the standard Frobenius norm.

$$(3.2) \quad \min_B \|Z - B\|_F^2 \quad s.t. \quad \|B\|_0 \leq s.$$

The above problem is to project  $Z$  onto the non-convex set  $\{B \in \mathbb{C}^{n \times N} : \|B\|_0 \leq s\}$  of matrices that have sparsity  $\leq s$ , which we call the  $s$ - $\ell_0$  ball. The optimal projection  $\hat{B}$  is easily computed

by zeroing out all but the  $s$  coefficients of largest magnitude in  $Z$ . We denote this operation by  $\hat{B} = H_s(Z)$ , where  $H_s(\cdot)$  is the corresponding projection operator. In case, there is more than one choice for the  $s$  elements of largest magnitude in  $Z$ , then  $H_s(Z)$  is chosen as the projection for which the indices of these  $s$  elements are the lowest possible in lexicographical order.

In the case of Problem (P3), the sparse coding step involves the following unconstrained (and non-convex) optimization problem

$$(3.3) \quad \min_B \|Z - B\|_F^2 + \eta^2 \|B\|_0$$

The optimal solution  $\hat{B}$  in this case is obtained as  $\hat{B} = \hat{H}_\eta^1(Z)$ , with the hard-thresholding operator  $\hat{H}_\eta^1(\cdot)$  defined as follows, where the subscript  $ij$  indexes matrix entries ( $i$  for row and  $j$  for column).

$$(3.4) \quad \left( \hat{H}_\eta^1(Z) \right)_{ij} = \begin{cases} 0 & , \quad |Z_{ij}| < \eta \\ Z_{ij} & , \quad |Z_{ij}| \geq \eta \end{cases}$$

The optimal solution to Problem (3.3) is not unique when the condition  $|Z_{ij}| = \eta$  is satisfied for some  $i, j$  (cf. Page 3 of [73] for a similar scenario and an explanation). The definition in (3.4) chooses *one* of the multiple optimal solutions in this case.

**3.1.2. Transform Update Step.** Here, we solve for  $W$  in the proposed formulations, with the other variables kept fixed. In the case of Problems (P1) and (P3), this involves the following optimization problem

$$(3.5) \quad \min_W \sum_{j=1}^N \|W P_j x - b_j\|_2^2 + 0.5\lambda \|W\|_F^2 - \lambda \log |\det W|$$

Now, let  $X \in \mathbb{C}^{n \times N}$  be the matrix with the vectorized patches  $P_j x$  as its columns, and recall that  $B$  is the matrix of codes  $b_j$ . Then, Problem (3.5) becomes

$$(3.6) \quad \min_W \|WX - B\|_F^2 + 0.5\lambda \|W\|_F^2 - \lambda \log |\det W|$$

An analytical solution for this problem has been recently derived [67, 73], and is stated in the following proposition. It is expressed in terms of an appropriate singular value decomposition (SVD). We let  $M^{\frac{1}{2}}$  denote the positive definite square root of a positive definite matrix  $M$ .

**Proposition 3.1.** *Given  $X \in \mathbb{C}^{n \times N}$ ,  $B \in \mathbb{C}^{n \times N}$ , and  $\lambda > 0$ , factorize  $XX^H + 0.5\lambda I$  as  $LL^H$ , with  $L \in \mathbb{C}^{n \times n}$ . Further, let  $L^{-1}XB^H$  have a full SVD of  $V\Sigma R^H$ . Then, a global minimizer for the transform update step (3.6) is*

$$(3.7) \quad \hat{W} = 0.5R \left( \Sigma + (\Sigma^2 + 2\lambda I)^{\frac{1}{2}} \right) V^H L^{-1}$$

*The solution is unique if and only if  $XB^H$  is non-singular. Furthermore, the solution is invariant to the choice of factor  $L$ .*

*Proof.* : See the proof of Proposition 1 of [73], particularly the discussion following that proof. ■

The factor  $L$  in Proposition 3.1 can for example be the factor  $L$  in the Cholesky factorization  $XX^H + 0.5\lambda I = LL^H$ , or the Eigenvalue Decomposition (EVD) square root of  $XX^H + 0.5\lambda I$ . The closed-form solution (3.7) is nevertheless invariant to the particular choice of  $L$  [73]. Although in practice both the SVD and the square root of non-negative scalars are computed using iterative methods, we will assume in the convergence analysis in this work, that the solution (3.7) (as well as later ones that involve such computations) is computed exactly. In practice, standard numerical methods are guaranteed to quickly provide machine precision accuracy for the SVD or other (aforementioned) computations.

In the case of Problem (P2), the transform update step involves the following problem

$$(3.8) \quad \min_W \|WX - B\|_F^2 \quad \text{s.t.} \quad W^H W = I.$$

The solution to the above problem can be expressed as follows (see [67], or Proposition 2 of [73]).

**Proposition 3.2.** *Given  $X \in \mathbb{C}^{n \times N}$  and  $B \in \mathbb{C}^{n \times N}$ , let  $XB^H$  have a full SVD of  $U\Sigma V^H$ . Then, a global minimizer in (3.8) is*

$$(3.9) \quad \hat{W} = VU^H$$

*The solution is unique if and only if  $XB^H$  is non-singular.*

**3.1.3. Image Update Step: General Case.** In this step, we solve Problems (P1)-(P3) for the image  $x$ , with the other variables fixed. This involves the following optimization problem

$$(3.10) \quad \min_x \sum_{j=1}^N \|WP_j x - b_j\|_2^2 + \nu \|Ax - y\|_2^2 \quad \text{s.t.} \quad \|x\|_2 \leq C.$$

Problem (3.10) is a least squares problem with an  $\ell_2$  (alternatively, squared  $\ell_2$ ) constraint [31]. It can be solved exactly by using the Lagrange multiplier method [31]. The corresponding Lagrangian formulation is

$$(3.11) \quad \min_x \sum_{j=1}^N \|WP_j x - b_j\|_2^2 + \nu \|Ax - y\|_2^2 + \mu (\|x\|_2^2 - C)$$

where  $\mu \geq 0$  is the Lagrange multiplier. The solution to (3.11) satisfies the following Normal Equation

$$(3.12) \quad (G + \nu A^H A + \mu I) x = \sum_{j=1}^N P_j^T W^H b_j + \nu A^H y$$

where

$$(3.13) \quad G \triangleq \sum_{j=1}^N P_j^T W^H W P_j$$

and  $(\cdot)^T$  (matrix transpose) is used instead of  $(\cdot)^H$  above for real matrices. The solution to (3.12) is unique for any  $\mu \geq 0$  because matrix  $G$  is positive-definite. To see why, consider any  $z \in \mathbb{C}^p$ . Then, we have  $z^H G z = \sum_{j=1}^N \|W P_j z\|_2^2$ , which is strictly positive unless  $W P_j z = 0 \forall j$ . Since the  $W$  in our algorithm is ensured to be invertible, we have that  $W P_j z = 0 \forall j$  if and only if  $P_j z = 0 \forall j$ , which implies (assuming that the set of patches in our formulations covers all pixels in the image) that  $z = 0$ . This ensures  $G \succ 0$ . The *unique* solution to (3.12) can be found by direct methods (for small-sized problems), or by conjugate gradients (CG).

To solve the original Problem (3.10), the Lagrange multiplier  $\mu$  in (3.12) must also be chosen optimally. This is done by first computing the EVD of the  $p \times p$  matrix  $G + \nu A^H A$  as  $U \Sigma U^H$ . Since  $G \succ 0$ , we have that  $\Sigma \succ 0$ . Then, defining  $z \triangleq U^H \left( \sum_{j=1}^N P_j^T W^H b_j + \nu A^H y \right)$ , we have that (3.12) implies  $U^H x = (\Sigma + \mu I)^{-1} z$ . Therefore,

$$(3.14) \quad \|x\|_2^2 = \|U^H x\|_2^2 = \sum_{i=1}^p \frac{|z_i|^2}{(\Sigma_{ii} + \mu)^2} \triangleq \tilde{f}(\mu)$$

where  $z_i$  above denotes the  $i^{\text{th}}$  entry of vector  $z$ . If  $\tilde{f}(0) \leq C^2$ , then  $\hat{\mu} = 0$  is the optimal multiplier. Otherwise, the optimal  $\hat{\mu} > 0$ . In the latter case, since the function  $\tilde{f}(\mu)$  in (3.14) is monotone decreasing for  $\mu > 0$ , and  $\tilde{f}(0) > C^2$ , there is a unique multiplier  $\hat{\mu} > 0$  such that  $\tilde{f}(\hat{\mu}) - C^2 = 0$ . The optimal  $\hat{\mu}$  is found by using the classical Newton's method (or, alternatively [25]), which in our case is guaranteed to converge to the optimal solution at a quadratic rate. Once the optimal  $\hat{\mu}$  is found (to within machine precision), the *unique* solution to the image update Problem (3.10) is  $U (\Sigma + \hat{\mu} I)^{-1} z$ , coinciding with the solution to (3.12) with  $\mu = \hat{\mu}$ .

In practice, when a large value (or, loose estimate) of  $C$  is used (for example, in our experiments later in Section 5), the optimal solution to (3.10) is typically obtained with the setting  $\hat{\mu} = 0$  for the multiplier in (3.11). In this case, the unique minimizer of the objective in (3.10) (e.g., obtained with CG) directly satisfies the constraint. Therefore, the additional computations (e.g., EVD) to find the optimal  $\hat{\mu}$  can be avoided in this case. Other alternative ways to find the solution to (3.10) when the optimal  $\hat{\mu} \neq 0$ , without the EVD computation, include using the projected gradient method, or solving (3.11) repeatedly (by CG) for various  $\mu$  (tuned in steps) until the  $\|x\|_2 = C$  condition is satisfied.

When employing CG (or, the projected gradient method), the structure of the various matrices in (3.12) can be exploited to enable efficient computations. First, we show that under certain assumptions, the matrix  $G$  in (3.12) is a 2D circulant matrix, i.e., a *Block Circulant matrix with Circulant Blocks* (abbreviated as BCCB matrix), enabling efficient computation (via FFTs) of the product  $Gx$  (used in CG). Second, in certain applications, the matrix  $A^H A$  in (3.12) may have a structure (e.g., sparsity, Toeplitz structure, etc.) that enables efficient computation of  $A^H A x$  (used in CG). In such cases, the CG iterations can be performed efficiently.

We now show the matrix  $G$  in (3.12) is a BCCB matrix. To do this, we make some assumptions on how the overlapping 2D patches are selected from the image(s) in our formulations. First, we assume that periodically positioned, overlapping 2D image patches are used. Furthermore, the patches that overlap the image boundaries are assumed to ‘wrap around’

on the opposite side of the image [64]. Now, defining the patch *overlap stride*  $r$  to be the distance in pixels between corresponding pixel locations in adjacent image patches, it is clear that the setting  $r = 1$  results in a maximal set of overlapping patches. When  $r = 1$  (and assuming patch ‘wrap around’), the following proposition establishes that the matrix  $G$  is a Block Circulant matrix with Circulant Blocks. Let  $F \in \mathbb{C}^{p \times p}$  denote the full (2D) Fourier encoding matrix assumed normalized such that  $F^H F = I$ .

**Proposition 3.3.** *Let  $r = 1$ , and assume that all ‘wrap around’ image patches are included. Then, the matrix  $G = \sum_{j=1}^N P_j^T W^H W P_j$  in (3.12) is a BCCB matrix with eigenvalue decomposition  $F^H \Gamma F$ , with  $\Gamma \succ 0$ .*

*Proof.* : First, note that  $W^H W = \sum_{i=1}^n e_i e_i^T W^H W$ , where  $\{e_i\}_{i=1}^n$  are the columns of the  $n \times n$  identity matrix. Denote the  $i^{\text{th}}$  row of  $W^H W$  by  $h_i$ . Then, the matrix  $e_i e_i^T W^H W$  is all zero except for its  $i^{\text{th}}$  row, which is equal to  $h_i$ . Let  $G_i \triangleq \sum_{j=1}^N P_j^T (e_i e_i^T W^H W) P_j$ . Then,  $G = \sum_{i=1}^n G_i$ .

Consider a vectorized image  $z \in \mathbb{C}^p$ . Because the set of entries of  $G_i z$  is simply the set of inner products between  $h_i$  and all the (overlapping and wrap around) patches of the image corresponding to  $z$ , it follows that applying  $G_i$  on 2D circularly shifted versions of the image corresponding to  $z$  produces correspondingly shifted versions of the output  $G_i z$ . Hence,  $G_i$  is an operator corresponding to 2D circular convolution.

Now, it follows that each  $G_i$  is a BCCB matrix. Since  $G = \sum_{i=1}^n G_i$  is a sum of BCCB matrices, it is therefore a BCCB matrix as well. Now, from standard results regarding BCCB matrices, we know that  $G$  has an EVD of  $F^H \Gamma F$ , where  $\Gamma$  is a diagonal matrix of eigenvalues. It was previously established that  $G$  is positive-definite. Therefore,  $\Gamma \succ 0$  holds. ■

Proposition 3.3 states that matrix  $G$  in (3.12) is a BCCB matrix, that is diagonalizable by the Fourier basis. Therefore,  $G$  can be applied (via FFTs) on a vector (e.g., in CG) at  $O(p \log p)$  cost. This is much lower than the  $O(n^2 p)$  cost ( $n^2 \gg \log p$  typically) of applying  $G = \sum_{j=1}^N P_j^T W^H W P_j$  directly using patch-based operations. We now show how the diagonal matrix  $\Gamma$  in Proposition 3.3 can be computed efficiently. First, in Proposition 3.3, in the case that  $W$  is a unitary matrix (as in (P2)), the matrix  $G = \sum_{j=1}^N P_j^T P_j$  is simply the identity scaled by a factor  $n$  [64]. In this case,  $\Gamma = nI$ . Second, when  $W$  is non-unitary, let us assume that the Fourier matrix  $F$  is arranged so that its first column is the constant column (with entries  $= 1/\sqrt{p}$ ). Then, the diagonal of  $\Gamma$  in this case is obtained efficiently (via FFTs) as  $\sqrt{p} F a_1$ , where  $a_1$  is the first column of  $G$ . Now, the first column of  $G$  can itself be easily computed by applying this operator (using simple patch-based operations) on  $z \in \mathbb{C}^p$  that has a one in its first entry (corresponding to an image with a 1 in the top left corner) and zeros elsewhere. Note that since  $z$  is extremely sparse,  $a_1$  is computed at a negligible cost. The aforementioned computation for finding the diagonal of  $\Gamma$  is equivalent, of course, to finding the spectrum of the operator corresponding to  $G$ , as the DFT of its impulse response.

**3.1.4. Image Update Step: Case of MRI.** In certain scenarios, the optimal  $\hat{x}$  in (3.10) can be found very efficiently. Here, we consider the case of Fourier imaging modalities, or more specifically, MRI, where  $A = F_u$ , the undersampled Fourier encoding matrix. In order to obtain an efficient solution for the  $\hat{x}$  in (3.10), we assume that the k-space measurements in MRI are obtained by subsampling on a uniform Cartesian grid. We then show that the optimal multiplier  $\hat{\mu}$  and the corresponding optimal  $\hat{x}$  can be computed without any EVD

computations, or CG, for MRI.

Assume  $r = 1$ , and that all ‘wrap around’ image patches are included in the formulation. Then, empowered with the diagonalization result of Proposition 3.3, we can simplify equation (3.12) for MRI, by rewriting it as follows

$$(3.15) \quad (FGF^H + \nu FF_u^H F_u F^H + \mu I) Fx = F \sum_{j=1}^N P_j^T W^H b_j + \nu FF_u^H y$$

All  $p$ -dimensional vectors (vectorized images) in (3.15) are in Fourier or k-space. Vector  $FF_u^H y \in \mathbb{C}^p$  represents the zero-filled (or, zero padded) k-space measurements. The matrix  $FF_u^H F_u F^H$  is a diagonal matrix consisting of ones and zeros, with the ones at those diagonal entries that correspond to sampled locations in k-space. By Proposition 3.3, for  $r = 1$ , the matrix  $FGF^H = \Gamma$  is diagonal. Therefore, the matrix pre-multiplying  $Fx$  in (3.15) is diagonal and invertible. Denoting the diagonal of  $\Gamma$  by  $\gamma \in \mathbb{R}^p$  (all positive vector),  $S_0 \triangleq FF_u^H y$ , and  $S \triangleq F \sum_{j=1}^N P_j^T W^H b_j$ , we have that the solution to (3.15) for fixed  $\mu$  is

$$(3.16) \quad Fx_\mu(k_x, k_y) = \begin{cases} \frac{S(k_x, k_y)}{\gamma(k_x, k_y) + \mu} & , (k_x, k_y) \notin \Omega \\ \frac{S(k_x, k_y) + \nu S_0(k_x, k_y)}{\gamma(k_x, k_y) + \nu + \mu} & , (k_x, k_y) \in \Omega \end{cases}$$

where  $(k_x, k_y)$  indexes k-space locations, and  $\Omega$  represents the subset of k-space that has been sampled. Equation (3.16) provides a closed-form solution to the Lagrangian Problem (3.11) for CS MRI, with  $Fx_\mu(k_x, k_y)$  representing the optimal updated value (for a particular  $\mu$ ) in k-space at location  $(k_x, k_y)$ .

The function  $\tilde{f}(\mu)$  in (3.14) now has a simple form (no EVD needed) as follows

$$(3.17) \quad \tilde{f}(\mu) = \|Fx\|_2^2 = \sum_{(k_x, k_y) \notin \Omega} \frac{|S(k_x, k_y)|^2}{(\gamma(k_x, k_y) + \mu)^2} + \sum_{(k_x, k_y) \in \Omega} \frac{|S(k_x, k_y) + \nu S_0(k_x, k_y)|^2}{(\gamma(k_x, k_y) + \nu + \mu)^2}$$

We check if  $\tilde{f}(0) \leq C^2$  first, before applying Newton’s method to solve  $\tilde{f}(\hat{\mu}) = C^2$ . The optimal  $\hat{x}$  in (3.10) is obtained via a 2D inverse FFT of the updated  $Fx_{\hat{\mu}}$  in (3.16).

The pseudocodes of the overall Algorithms A1, A2, and A3 corresponding to the BCS Problems (P1), (P2), and (P3) respectively, are shown below. An appropriate choice for the initial  $(W^0, B^0, x^0)$  in Algorithms A1-A3 would depend on the specific application. For example,  $W^0$  could be the  $n \times n$  2D DCT matrix,  $x^0 = A^\dagger y$  (normalized so that it satisfies  $\|x^0\|_2 \leq C$ ), and  $B^0$  can be the minimizer of Problems (P1)-(P3) for these fixed  $W^0$  and  $x^0$ .

**3.2. Computational Cost.** Algorithms A1, A2, and A3 involve the steps of sparse coding, transform update, and image update. We now briefly discuss the computational costs of each of these steps.

First, in each outer iteration of our Algorithms A1 and A3, the computation of the matrix  $XX^H + 0.5\lambda I$ , where  $X$  has the image patches as its columns, can be done in  $O(n^2 N)$  operations, where typically  $n \ll N$ . The computation of the inverse square root  $L^{-1}$  requires only  $O(n^3)$  operations.



---

Transform-Based BCS Algorithms <sup>3</sup>A1, A2, and A3

---

**Inputs:**  $y$  - measurements obtained with sensing matrix  $A$ ,  $s$  - sparsity,  $\lambda$  - weight,  $\nu$  - weight,  $C$  - energy bound,  $\hat{M}$  - number of inner iterations,  $J$  - number of outer iterations.

**Outputs:**  $x$  - reconstructed image,  $W$  - adapted sparsifying transform,  $B$  - matrix with sparse codes of all overlapping patches as columns.

**Initial Estimates:**  $(W^0, B^0, x^0)$ .

**For  $t = 1$ : J Repeat**

1) Form the matrix  $X$  with  $P_j x^{t-1}$  as its columns. Compute  $L^{-1} = (XX^H + 0.5\lambda I)^{-1/2}$  for Algorithms A1 and A3. Set  $\tilde{B}^0 = B^{t-1}$ .

2) **For  $l = 1$ :  $\hat{M}$  Repeat**

(a) **Transform Update Step:**

i. Set  $V\Sigma R^H$  as the full SVD of  $L^{-1}X (\tilde{B}^{l-1})^H$  for Algorithms A1 and A3, or the full SVD of  $X (\tilde{B}^{l-1})^H$  for Algorithm A2.

ii.  $\tilde{W}^l = 0.5R \left( \Sigma + (\Sigma^2 + 2\lambda I)^{\frac{1}{2}} \right) V^H L^{-1}$  for Algorithms A1 and A3, or  $\tilde{W}^l = RV^H$  for Algorithm A2.

(b) **Sparse Coding Step:**  $\tilde{B}^l = H_s(\tilde{W}^l X)$  for Algorithms A1 and A2, or  $\tilde{B}^l = \hat{H}_\eta^1(\tilde{W}^l X)$  for Algorithm A3.

**End**

3) Set  $W^t = \tilde{W}^{\hat{M}}$  and  $B^t = \tilde{B}^{\hat{M}}$ . Set  $b_j^t$  as the  $j^{\text{th}}$  column of  $B^t \forall j$ .

4) **Image Update Step:**

(a) For generic CS scheme, solve (3.12) for  $x^t$  with  $\mu = 0$ , by linear CG. If  $\|x^t\|_2 > C$ ,

i. Compute  $U\Sigma U^H$  as EVD of  $\sum_{j=1}^N P_j^T (W^t)^H W^t P_j + \nu A^H A$ .

ii. Compute  $z = U^H \left( \sum_{j=1}^N P_j^T (W^t)^H b_j^t + \nu A^H y \right)$ .

iii. Use Newton's method to find  $\hat{\mu}$  such that  $\tilde{f}(\hat{\mu}) = C^2$  in (3.14).

iv.  $x^t = U (\Sigma + \hat{\mu} I)^{-1} z$ .

(b) For MRI, do the following

i. Compute the image  $c = \sum_{j=1}^N P_j^T (W^t)^H b_j^t$ .  $S \leftarrow FFT(c)$ .

ii. Compute  $a_1$  as the first column of  $\sum_{j=1}^N P_j^T (W^t)^H W^t P_j$ .

iii. Set  $\gamma \leftarrow \sqrt{p} \times FFT(a_1)$ .

iv. Compute  $\tilde{f}(0)$  as per (3.17).

v. If  $\tilde{f}(0) \leq C^2$ , set  $\hat{\mu} = 0$ . Else, use Newton's method to solve  $\tilde{f}(\hat{\mu}) = C^2$  for  $\hat{\mu}$ .

vi. Update  $S$  to be the right hand side of (3.16) with  $\mu = \hat{\mu}$ .  $x^t = IFFT(S)$ .

**End**

---

The cost of the sparse coding step in our algorithms is dominated by the computation of the

---

<sup>3</sup>The superscripts  $t$  and  $l$  denote the main iterates, and the iterates in the inner alternations between transform update and sparse coding, respectively. The  $FFT$  and  $IFFT$  denote the fast implementations of the normalized 2D DFT and 2D IDFT. For MRI,  $r = 1$ , and the encoding matrix  $F$  is assumed normalized, and arranged so that its first column is the constant DC column. In Step 4a, although we list the image update method involving EVD, an alternative scheme is one mentioned in the text in Section 3.1.3, involving only CG.

matrix  $Z = WX$  in (3.2) (for Algorithms A1, A2) or (3.3) (for Algorithm A3), and therefore scales as  $O(n^2N)$ . Notably, the projection onto the  $s$ - $\ell_0$  ball in (3.2) costs only  $O(nN \log N)$  operations, when employing sorting [70], with  $\log N \ll n$  typically. Alternatively, in the case of (3.3), the hard thresholding operation costs only  $O(nN)$  comparisons.

The cost of the transform update step of our algorithms is dominated by the computation of the matrix  $XB^H$ . Since  $B$  is sparse,  $XB^H$  is computed with  $\alpha n^2N$  multiply-add operations, where  $\alpha < 1$  is the fraction of non-zeros in  $B$ .

The cost of the image update step in our algorithms depends on the specific application (i.e., the specific structure of  $A^HA$ , etc.). Here, we discuss the cost of the image update step discussed in Section 3.1.4, for the specific case of MRI. We assume  $r = 1$ , and that the patches ‘wrap around’, which implies that  $N = p$  (i.e., number of patches equals number of image pixels). The computational cost here is dominated by the computation of the term  $\sum_{j=1}^N P_j^T W^H b_j$  in the normal equation (3.12), which takes  $O(n^2N)$  operations. On the other hand, the various FFT and IFFT operations cost only  $O(N \log N)$  operations, where  $\log N \ll n^2$  typically. The Newton’s method to compute the optimal multiplier  $\hat{\mu}$  is only used when  $\mu = 0$  is non-optimal. In the latter case, Newton’s method takes up  $O(N\tilde{J})$  operations, with  $\tilde{J}$  being the number of iterations (typically small, and independent of  $n$ ) of Newton’s method.

Based on the preceding arguments, it is easy to observe that the total cost per (outer) iteration of the Algorithms A1-A3 scales (for MRI) as  $O(n^2N\hat{M})$ . Now, the recent synthesis dictionary-based BCS method called DLMRI [64] learns a dictionary  $D \in \mathbb{C}^{n \times K}$  from CS MRI measurements by solving Problem (P0). For this scheme, the computational cost per outer iteration scales as  $O(NKns\hat{J})$  [70], where  $\hat{J}$  is the number of (inner) iterations of dictionary learning (using the K-SVD algorithm [2]), and the other notations are the same as in (P0). Assuming that  $K \propto n$ , and that the synthesis sparsity  $s \propto n$ , we have that the cost per iteration of DLMRI scales as  $O(n^3N\hat{J})$ . Thus, the per-iteration computational cost of the proposed BCS schemes is much lower (lower in order by factor  $n$  assuming  $\hat{M} \sim \hat{J}$ ) than that for synthesis dictionary-based BCS. This gap in computations is amplified for higher-dimensional imaging applications such as 3D or 4D imaging, where the size of the 3D or 4D patches is typically much bigger than in the case of 2D imaging.

As illustrated in our experiments in Section 5, the proposed BCS algorithms converge in few iterations in practice. Therefore, the per-iteration computational advantages over synthesis dictionary-based BCS also typically translate to a net computational advantage in practice.

**4. Convergence Results.** Here, we present convergence guarantees for Algorithms A1, A2, and A3, that solve Problems (P1), (P2), and (P3), respectively. These problems are highly non-convex. Notably they involve either an  $\ell_0$  penalty or constraint for sparsity, a non-convex transform regularizer or constraint, and the term  $\|WP_jx - b_j\|_2^2$  that is a non-convex function involving the product of unknown matrices and vectors. The proposed algorithms for Problems (P1)-(P3) are block coordinate descent-type algorithms. We previously established in Proposition 2.1, the (noiseless) identifiability of the underlying image by solving the proposed problems (specifically, (P1) or (P2)). We are now interested in understanding whether the proposed algorithms converge to a minimizer of the corresponding problems, or whether they possibly get stuck in non-stationary points. Due to the high degree of non-convexity in-

volved here, standard results on convergence of block coordinate descent methods (e.g., [77]) do not apply here.

Very recent works on the convergence of block coordinate descent-type algorithms (e.g., [82], or the Block Coordinate Variable Metric Forward-Backward algorithm [14]) prove convergence of the iterate sequence (for specific algorithm) to a critical point of the objective. However, these works make numerous assumptions, some of which can be easily shown to be violated for the proposed formulations (for example, the term  $\sum_{j=1}^N \|W P_j x - b_j\|_2^2$  in the objectives of our formulations, although differentiable, violates the L-Lipschitzian gradient property described in Assumption 2.1 of [14]).

In fact, in certain simple scenarios, one can easily derive non-convergent iterate sequences for the Algorithms A1-A3. Non-convergence mainly arises for the transform or sparse code sequences (rather than the image sequence) due to the fact that the optimal solutions in the sparse coding or transform update steps may be non-unique. For example, in the trivial case when  $y = 0$ , the image  $x = 0$  is the unique minimizer in the proposed BCS problems. Hence, if  $x^0 = 0$ , then the iterates in Algorithms A1-A3 easily satisfy  $x^t = 0 \forall t$ . Since the patches of  $x^t$  are all zero, then any unitary matrix would be an (non-unique) optimal sparsifier in the transform update step of the algorithms, no matter how many inner alternations are performed between sparse coding and transform update within each outer iteration of the algorithms (the sparse codes are always 0 in this case). Hence, the  $\{W^t\}$  sequence can be any sequence of unitary transforms in this case (a limit need not exist).

In this work, we provide convergence results for the proposed BCS approaches, where the only assumption is that the various steps in our algorithms are solved exactly. (Recall that machine precision is typically guaranteed in practice.) Unlike prior works on dictionary-based blind compressed sensing [64, 80], wherein the update steps such as the  $\ell_0$  “norm” based synthesis sparse coding are only solved approximately, our analysis here for the transform based BCS approaches makes use of the explicit solutions for the update steps in our algorithms, to prove convergence.

**4.1. Preliminaries.** We first list some definitions that will be used in our analysis.

**Definition 4.1.** For a function  $\phi : \mathbb{R}^q \mapsto (-\infty, +\infty]$ , its domain is defined as  $\text{dom}\phi = \{z \in \mathbb{R}^q : \phi(z) < +\infty\}$ . Function  $\phi$  is proper if  $\text{dom}\phi$  is nonempty.

Next, we define the notion of Fréchet sub-differential for a function as follows [74, 49]. The norm and inner product notations used below correspond to the euclidean  $\ell_2$  settings.

**Definition 4.2.** Let  $\phi : \mathbb{R}^q \mapsto (-\infty, +\infty]$  be a proper function and let  $z \in \text{dom}\phi$ . The Fréchet sub-differential of the function  $\phi$  at  $z$  is the following set:

$$(4.1) \quad \hat{\partial}\phi(z) \triangleq \left\{ h \in \mathbb{R}^q : \liminf_{b \rightarrow z, b \neq z} \frac{1}{\|b - z\|} (\phi(b) - \phi(z) - \langle b - z, h \rangle) \geq 0 \right\}$$

If  $z \notin \text{dom}\phi$ , then  $\hat{\partial}\phi(z) \triangleq \emptyset$ . The sub-differential of  $\phi$  at  $z$  is defined as

$$(4.2) \quad \partial\phi(z) \triangleq \left\{ \tilde{h} \in \mathbb{R}^q : \exists z_k \rightarrow z, \phi(z_k) \rightarrow \phi(z), h_k \in \hat{\partial}\phi(z_k) \rightarrow \tilde{h} \right\}.$$

A necessary condition for  $z \in \mathbb{R}^q$  to be a minimizer of the function  $\phi : \mathbb{R}^q \mapsto (-\infty, +\infty]$  is that  $z$  is a *critical point* of  $\phi$ , i.e.,  $0 \in \partial\phi(z)$ . If  $\phi$  is a convex function, this condition is also sufficient. Critical points can be thought of as “generalized stationary points” [74].

We say that a sequence  $\{a^t\}$  with  $a^t \in \mathbb{C}^q$  has an accumulation point  $a$ , if there is a subsequence that converges to  $a$ .

**4.2. Notations.** Problems (P1) and (P2) have the constraint  $\|B\|_0 \leq s$ , which can instead be added as a penalty in the respective objectives by using a barrier function  $\psi(B)$  (which takes the value  $+\infty$  when the sparsity constraint is violated, and is zero otherwise). Problem (P2) also has the constraint  $W^H W = I$ , which can be equivalently added as a penalty in the objective of (P2) by using the barrier function  $\varphi(W)$ , which takes the value  $+\infty$  when the unitary constraint is violated, and is zero otherwise. Finally, the constraint  $\|x\|_2 \leq C$  in our formulations is replaced by the barrier function penalty  $\chi(x)$ . With these modifications, all the proposed problem formulations can be written in an unconstrained form. The objectives of (P1), (P2), and (P3), are then respectively denoted as

$$(4.3) \quad g(W, B, x) = \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|W P_j x - b_j\|_2^2 + \lambda Q(W) + \psi(B) + \chi(x)$$

$$(4.4) \quad u(W, B, x) = \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|W P_j x - b_j\|_2^2 + \varphi(W) + \psi(B) + \chi(x)$$

$$(4.5) \quad v(W, B, x) = \nu \|Ax - y\|_2^2 + \sum_{j=1}^N \|W P_j x - b_j\|_2^2 + \lambda Q(W) + \eta^2 \|B\|_0 + \chi(x)$$

It is easy to see that (cf. [73] for a similar statement and justification) the unconstrained minimization problem involving the objective  $g(W, B, x)$  is exactly equivalent to the corresponding constrained formulation (P1), in the sense that the minimum objective values as well as the set of minimizers of the two formulations are identical. The same result also holds with respect to (P2) and  $u(W, B, x)$ , or (P3) and  $v(W, B, x)$ .

Since the functions  $g$ ,  $u$ , and  $v$  accept complex-valued (input) arguments, we will compute all derivatives or sub-differentials (Definition 4.2) of these functions with respect to the (real-valued) real and imaginary parts of the variables  $(W, B, x)$ . Note that the functions  $g$ ,  $u$ , and  $v$  are proper (we set the negative log-determinant penalty to be  $+\infty$  wherever  $\det W = 0$ ) and lower semi-continuous. For the Algorithms A1-A3, we denote the iterates (outputs) in each outer iteration  $t$  by the set  $(W^t, B^t, x^t)$ .

For a matrix  $H$ , we let  $\rho_j(H)$  denote the magnitude of the  $j^{\text{th}}$  largest element (magnitude-wise) of the matrix  $H$ . For some matrix  $E$ ,  $\|E\|_\infty \triangleq \max_{i,j} |E_{ij}|$ . Finally,  $\text{Re}(A)$  denotes the real part of some scalar or matrix  $A$ .

**4.3. Main Results.** The following theorem provides the convergence result for Algorithm A1 that solves Problem (P1). We assume that the initial estimates  $(W^0, B^0, x^0)$  satisfy all problem constraints.

**Theorem 4.3.** *Let  $\{W^t, B^t, x^t\}$  denote the iterate sequence generated by Algorithm A1 with measurements  $y \in \mathbb{C}^m$  and initial  $(W^0, B^0, x^0)$ . Then, the objective sequence  $\{g^t\}$  with  $g^t \triangleq g(W^t, B^t, x^t)$  is monotone decreasing, and converges to a finite value, say  $g^* = g^*(W^0, B^0, x^0)$ . Moreover, the iterate sequence  $\{W^t, B^t, x^t\}$  is bounded, and all its accumulation points are equivalent in the sense that they achieve the exact same value  $g^*$  of*

the objective. The sequence  $\{a^t\}$  with  $a^t \triangleq \|x^t - x^{t-1}\|_2$ , converges to zero. Finally, every accumulation point  $(W, B, x)$  of  $\{W^t, B^t, x^t\}$  is a critical point of the objective  $g$  satisfying the following partial global optimality conditions

$$(4.6) \quad x \in \arg \min_{\tilde{x}} g(W, B, \tilde{x})$$

$$(4.7) \quad W \in \arg \min_{\tilde{W}} g(\tilde{W}, B, x)$$

$$(4.8) \quad B \in \arg \min_{\tilde{B}} g(W, \tilde{B}, x)$$

Each accumulation point  $(W, B, x)$  also satisfies the following partial local optimality conditions

$$(4.9) \quad g(W + dW, B + \Delta B, x) \geq g(W, B, x) = g^*$$

$$(4.10) \quad g(W, B + \Delta B, x + \tilde{\Delta}x) \geq g(W, B, x) = g^*$$

The conditions each hold for all  $\tilde{\Delta}x \in \mathbb{C}^p$ , and all sufficiently small  $dW \in \mathbb{C}^{n \times n}$  satisfying  $\|dW\|_F \leq \epsilon'$  for some  $\epsilon' > 0$  that depends on the specific  $W$ , and all  $\Delta B \in \mathbb{C}^{n \times N}$  in the union of the following regions R1 and R2, where  $X \in \mathbb{C}^{n \times N}$  is the matrix with  $P_j x$ ,  $1 \leq j \leq N$ , as its columns.

R1. The half-space  $\text{Re}(\text{tr}\{(WX - B)\Delta B^H\}) \leq 0$ .

R2. The local region defined by  $\|\Delta B\|_\infty < \rho_s(WX)$ .

Furthermore, if  $\|WX\|_0 \leq s$ , then  $\Delta B$  can be arbitrary.

Theorem 4.3 establishes that for each initial point  $(W^0, B^0, x^0)$ , the iterate sequence in Algorithm A1 converges to an equivalence class of accumulation points. Specifically, every accumulation point corresponds to the same value  $g^* = g^*(W^0, B^0, x^0)$  of the objective. The exact value of  $g^*$  could vary with initialization. Importantly, the equivalent accumulation points are all critical points as well as *at least* partial minimizers of the objective  $g(W, B, x)$ , in the following sense. Every accumulation point  $(W, B, x)$  is a partial global minimizer of  $g(W, B, x)$  with respect to each of  $W$ ,  $B$ , and  $x$ , as well as a partial local minimizer of  $g(W, B, x)$  with respect to  $(W, B)$ , and  $(B, x)$ , respectively. Therefore, we have the following corollary to Theorem 4.3.

**Corollary 4.4.** For each  $(W^0, B^0, x^0)$ , the iterate sequence in Algorithm A1 converges to an equivalence class of critical points, that are also partial minimizers satisfying (4.6), (4.7), (4.8), (4.9), and (4.10).

Conditions (4.9) and (4.10) in Theorem 4.3 hold true not only for local (or small) perturbations of the sparse code matrix (accumulation point)  $B$ , but also for arbitrarily large perturbations of the sparse codes in a half space, as defined by region R1. Furthermore, the partial optimality condition (4.10) also holds for arbitrary perturbations  $\tilde{\Delta}x$  of  $x$ .

Theorem 4.3 also says that  $\|x^t - x^{t-1}\|_2 \rightarrow 0$ . This is a necessary but not sufficient condition for convergence of the entire sequence  $\{x^t\}$ .

The following corollary to Theorem 4.3 also holds, where ‘globally convergent’ refers to convergence from any initialization.

**Corollary 4.5.** *Algorithm A1 is globally convergent to a subset of the set of critical points of the non-convex objective  $g(W, B, x)$ . The subset includes all critical points  $(W, B, x)$ , that are at least partial global minimizers of  $g(W, B, x)$  with respect to each of  $W$ ,  $B$ , and  $x$ , as well as partial local minimizers of  $g(W, B, x)$  with respect to each of the pairs  $(W, B)$ , and  $(B, x)$ .*

Theorem 4.3 holds for Algorithm A1 irrespective of the number of inner alternations  $\hat{M}$ , between transform update and sparse coding, within each outer algorithm iteration. In practice, we have observed that a larger value of  $\hat{M}$  (particularly in initial algorithm iterations) may enable Algorithm A1 to be insensitive (for example, in terms of the quality of the image reconstructed) to the initial (even, badly chosen) values of  $W^0$  and  $B^0$ .

The convergence results for Algorithms A2 or A3 are quite similar to that for Algorithm A1. The following two Theorems briefly state the results for Algorithms A3 and A2, respectively.

**Theorem 4.6.** *Theorem 4.3 applies to Algorithm A3 and the corresponding objective  $v(W, B, x)$  as well, except that the set of perturbations  $\Delta B \in \mathbb{C}^{n \times N}$  in Theorem 4.3 is restricted to  $\|\Delta B\|_\infty < \eta/2$  for Algorithm A3.*

**Theorem 4.7.** *Theorem 4.3, except for the condition (4.9), applies to Algorithms A2 and the corresponding objective  $u(W, B, x)$  as well.*

Note that owing to Theorems 4.6 and 4.7, results similar to Corollaries 4.4 and 4.5 also apply for Algorithms A3 and A2, respectively. The proofs of the stated convergence theorems are provided in Appendix A.

In general, the subset of the set of critical points to which each algorithm converges may be larger than the set of global minimizers (Section 2.2.2) in the respective problems. The question of the conditions under which the proposed algorithms converge to the (perhaps smaller) set of global minimizers in the proposed problems is open, and its investigation is left for future work.

## 5. Numerical Experiments.

**5.1. Framework.** Here, we study the usefulness of the proposed sparsifying transform-based blind compressed sensing framework for the CS MRI application<sup>4</sup>. The MR data used in these experiments are  $512 \times 512$  complex-valued images shown (only the magnitude is displayed) in Fig. 1(a) and Fig. 2(a). The image in Fig. 1(a) was kindly provided by Prof. Michael Lustig, UC Berkeley, and is one image slice (with rich features) from a multislice data acquisition. The image in Fig. 2(a) is publicly available<sup>5</sup>. We simulate various undersampling patterns in k-space<sup>6</sup> including variable density 2D random sampling<sup>7</sup> [76, 64], and Cartesian sampling with (variable density) random phase encodes (1D random).

<sup>4</sup>We have also proposed another sparsifying transform-based BCS MRI method recently [71]. However, the latter approach involves many more parameters (e.g., error thresholds to determine patch-wise sparsity levels), which may be hard to tune in practice. In contrast, the methods proposed in this work involve only a few parameters that are relatively easy to set.

<sup>5</sup>It can be downloaded from <http://web.stanford.edu/class/ee369c/data/brain.mat>. A phase-shifted version of the image was used in the experiments in [72].

<sup>6</sup>We simulate the k-space of an image  $x$  using the command `fftshift(fft2(ifftshift(x)))` in Matlab. Fig. 2(b) shows the k-space (only magnitude is displayed) of the reference in Fig. 2(a).

<sup>7</sup>This sampling scheme is feasible when data corresponding to multiple image slices are jointly acquired, and the frequency encode direction is chosen perpendicular to the image plane.



We employ Problem (P1) and the corresponding Algorithm A1 to reconstruct images from undersampled measurements in the experiments here<sup>8</sup>. Our reconstruction method is referred to as Transform Learning MRI (TLMRI).

First, we illustrate the empirical convergence behavior of TLMRI. We also compare the reconstructions provided by the TLMRI method to those provided by the following schemes: 1) the Sparse MRI method of Lustig et al [38], that utilizes Wavelets and Total Variation as *fixed* sparsifying transforms; 2) the DLMRI method [64] that learns adaptive overcomplete sparsifying dictionaries; 3) the PANO method [63] that exploits the non-local similarities between image patches (similar to [15]), and employs a 3D transform to sparsify groups of similar patches; and 4) the PBDWS method [51]. The PBDWS method is a very recent *partially* adaptive sparsifying transform based compressed sensing reconstruction method that uses redundant Wavelets and trained patch-based geometric directions. It has been shown to perform better than the earlier PBDW method [62].

We simulated the performance of the Sparse MRI, PBDWS, and PANO methods using the software implementations available from the respective authors' websites [37, 61, 59]. We used the built-in parameter settings in those implementations, which performed well in our experiments. Specifically, for the PBDWS method, the shift invariant discrete Wavelet transform (SIDWT) based reconstructed image is used as the *guide* (initial) image [51, 61]. We employed the zero-filling reconstruction (produced within the PANO demo code [59]) as the initial guide image for the PANO method [63, 59].

The implementation of the DLMRI algorithm that solves Problem (P0) is also available online [68]. For DLMRI, image patches of size  $6 \times 6$  ( $n = 36$ ) are used, as suggested in [64]<sup>9</sup>, and a four fold overcomplete synthesis dictionary ( $K = 144$ ) is learnt using 25 iterations of the algorithm. A patch overlap stride of  $r = 1$  is used, and 14400 (found empirically<sup>10</sup>) randomly selected patches are used during the dictionary learning step (executed for 20 iterations) of the DLMRI algorithm. Mean-subtraction is not performed for the patches prior to the dictionary learning step of DLMRI. A maximum sparsity level (of  $s = 7$  per patch) is employed together with an error threshold (for sparse coding) during the dictionary learning step. The  $\ell_2$  error threshold per patch varies linearly from 0.48 to 0.15 over the DLMRI

---

<sup>8</sup> Problem (P3) has been recently shown to be useful for adaptive tomographic reconstruction [54, 55]. The corresponding Algorithm A3 has the advantage that the sparse coding step involves the cheap hard thresholding operation, rather than the more expensive projection onto the  $s$ - $\ell_0$  ball (used in Algorithms A1 and A2). We have observed that Algorithm A3 also works well for MRI. Problems (P1) and (P2) differ in that (P2) enforces unit conditioning of the learnt transform, whereas (P1) also allows for more general condition numbers. Algorithm A2 (for (P2)) involves slightly cheaper computations than Algorithm A1 (for (P1)). In our experiments for MRI, we observed that well-conditioned transforms learnt via (P1) performed (in terms of image reconstruction quality) slightly better than strictly unitary learnt transforms. Therefore, we show results for (P1) in this work. We did not observe any dramatic difference in performance between the proposed methods in our experiments here. A detailed investigation of scenarios and applications where one of (P1), (P2), or (P3), performs the best (in terms of reconstruction quality compared to the others) is left for future work on specific applications. In this work, we have emphasized more the properties of these formulations, and the novel convergence guarantees of the corresponding algorithms.

<sup>9</sup>The reconstruction quality improves slightly with a larger patch size, but with a substantial increase in runtime.

<sup>10</sup>Using a larger training size ( $> 14400$ ) during the dictionary learning step of the algorithm provides negligible improvement in final image reconstruction quality, while leading to increased runtimes.

iterations. These parameter settings (all other parameters are set as per the indications in the DLMRI-Lab toolbox [68]) were observed to work well for the DLMRI algorithm.

The parameters for TLMRI (with Algorithms A1) are set to  $n = 36$ ,  $r = 1$  (with patch wrap around),  $\nu = 3.81$ ,  $\hat{M} = 1$ ,  $\lambda_0 = 0.2$ , and  $C = 10^5$ . The sparsity level  $s = 0.055 \times nN$  (this corresponds to an average sparsity level per patch of  $0.055 \times n$ , or 5.5% sparsity)<sup>11</sup>, where  $N = 512^2$ , is used in our experiments except in Section 5.2, where  $s = 0.045 \times nN$  is used. The initial transform estimate  $W^0$  is the (simple) patch-based 2D DCT [70], and the initial image  $x^0$  is set to be the standard zero-filling Fourier reconstruction<sup>12</sup>. The initial sparse code settings are the solution to (3.1), for the given  $(W^0, x^0)$ . Our TLMRI implementation was coded in Matlab version R2013a. Note that this implementation has not been optimized for efficiency. A link to the Matlab implementation is provided at <http://www.ifp.illinois.edu/~yoram/>. All simulations in this work were executed in Matlab. All computations were performed with an Intel Core i5 CPU at 2.5GHz and 4GB memory, employing a 64-bit Windows 7 operating system.

Similar to prior work [64], we quantify the quality of MR image reconstruction using the peak-signal-to-noise ratio (PSNR), and high frequency error norm (HFEN) metrics. The PSNR (expressed in decibels (dB)) is computed as the ratio of the peak intensity value of some reference image to the root mean square reconstruction error (computed between image magnitudes) relative to the reference. In MRI, the reference image is typically the image reconstructed from fully sampled k-space data. The HFEN metric quantifies the quality of reconstruction of edges or finer features. A rotationally symmetric Laplacian of Gaussian (LoG) filter is used, whose kernel is of size  $15 \times 15$  pixels, and with a standard deviation of 1.5 pixels [64]. HFEN is computed as the  $\ell_2$  norm of the difference between the LoG filtered reconstructed and reference magnitude images.

**5.2. Convergence and Learning Behavior.** In this experiment, we consider the reference image in Fig. 1(a). We perform four fold undersampling of the k-space space of the (peak normalized<sup>13</sup>) reference. The (variable density [64]) sampling mask is shown in Fig. 1(b). When the TLMRI algorithm is executed using the undersampled data, the objective function converges monotonically and quickly over the iterations as shown in Fig. 1(e). The changes between successive iterates  $\|x^t - x^{t-1}\|_2$  (Fig. 1(g)) converge towards 0. Such convergence was established by Theorem 4.3, and is indicative (a necessary but not sufficient condition) of convergence of the entire sequence  $\{x^t\}$ . As far as the performance metrics are concerned, the PSNR metric (Fig. 1(f)) increases over the iterations, and the HFEN metric decreases,

<sup>11</sup>The sparsity level  $s$  is a regularization parameter in our framework that provides a trade-off between how much aliasing is removed over the algorithm iterations, and how much image information is kept or restored (i.e., not eliminated by the sparsity condition). We determined the sparsity level empirically in the experiments in this work.

<sup>12</sup>While we used the naive zero-filling Fourier reconstruction in our experiments here for simplicity, one could also use other better initializations for  $x$  such as the SIDWT based reconstructed image [51], or the reconstructions produced by recent methods (e.g., PBDWS, etc.). We have observed empirically that better initializations may lead to faster convergence of TLMRI, and TLMRI typically only improves the image quality compared to the initializations (assuming properly chosen sparsity levels).

<sup>13</sup>In practice, the data or k-space measurements can always be normalized prior to processing them for image reconstruction. Otherwise, the parameter settings for algorithms may need to be modified to account for data scaling.

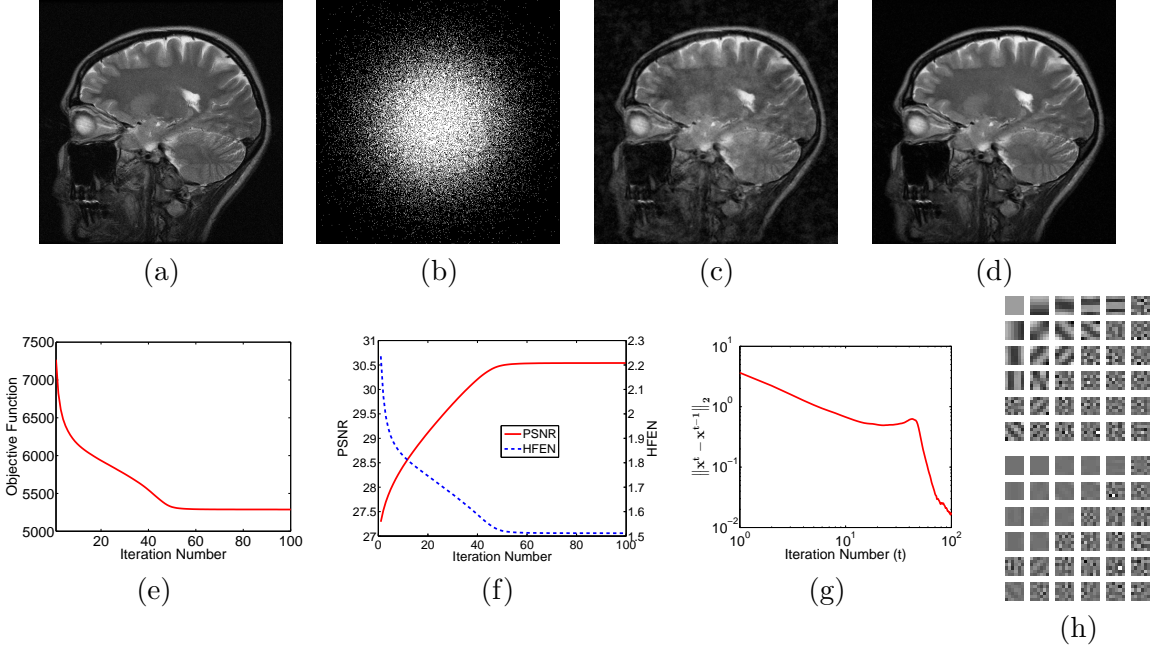


Figure 1: Convergence of TLMRI with 5x undersampling: (a) Reference image; (b) sampling mask in k-space; (c) initial zero-filling reconstruction (26.93 dB); (d) TLMRI reconstruction (30.54 dB); (e) objective function (since the regularizer  $Q(W) \geq n/2$  [70], we have subtracted out the constant offset of  $n\lambda/2$  from the objective values here); (f) PSNR and HFEN; (g) changes between successive iterates ( $\|x^t - x^{t-1}\|_2$ ), and (h) real (top) and imaginary (bottom) parts of the learnt  $W$ , with the matrix rows shown as patches.

indicating improving reconstruction quality over the algorithm iterations. These metrics also converge quickly.

The initial zero-filling reconstruction (Fig. 1(c)) shows aliasing artifacts that are typical in the undersampled measurement scenario, and has a PSNR of only 26.93 dB. On the other hand, the final TLMRI reconstruction (Fig. 1(d)) is much enhanced (by 3.6 dB), with a PSNR of 30.54 dB. Since Algorithm A1 is guaranteed to converge to the set of critical points of Problem (P1), the result in Fig. 1(d) suggests that, in practice, the set of critical points may in fact include images that are close to the true image. Note that our identifiability result (Proposition 2.1) in Section 2.2 ensured global optimality of the underlying image only in a noiseless (or error-free) scenario. The learnt transform  $W$  ( $\kappa(W) = 1.01$ ) for this example is shown in Fig. 1(h). This is a complex valued transform. Both the real and imaginary parts of  $W$  display texture or frequency like structures, that sparsify the patches of the MR image. Our algorithm is thus able to learn this structure and reconstruct the image using only the undersampled measurements.

**5.3. Comparison to Other Methods.** In the following experiments, we execute the TLMRI algorithm for 40 iterations (all other parameters are set to the values mentioned

Image	Sampling Scheme	Undersampling	Zero-filling	Sparse MRI	PBDWS	PANO	DLMRI	TLMRI
2	2D Random	4x	25.3	26.13	31.69	32.80	33.01	<b>33.12</b>
1	2D Random	5x	26.9	27.84	30.27	30.37	30.49	<b>30.56</b>
2	2D Random	7x	25.3	26.38	31.10	30.92	31.70	<b>31.94</b>
2	Cartesian	4x	28.9	29.73	31.67	32.24	32.67	<b>32.78</b>
2	Cartesian	7x	27.9	28.58	31.11	31.08	30.91	<b>31.24</b>

Table 1: PSNRs corresponding to the Zero-filling, Sparse MRI [38], PBDWS [51], PANO [63], DLMRI [64], and TLMRI reconstructions, for various sampling schemes and undersampling factors. The best PSNRs are marked in bold.

in Section 5.1). We also use a lower sparsity level ( $< 0.055 \times nN$ ) during the initial several algorithm iterations, which leads to faster convergence. We consider the complex-valued reference images in Fig. 1(a) and Fig. 2(a) that are labeled as Image 1 and Image 2 respectively, and simulate variable density 2D random or Cartesian undersampling [64] of the k-spaces of these images. Table 1 lists the reconstruction PSNRs corresponding to the zero-filling, Sparse MRI, PBDWS, PANO, DLMRI, and TLMRI<sup>14</sup> reconstructions for various cases.

The TLMRI algorithm is seen to provide the best PSNRs (analogous results were observed to typically hold with respect to the HFEN metric not shown in the table) for the various scenarios in Table 1. Significant improvements (up to 7 dB) are observed over the Sparse MRI method, that uses fixed sparsifying transforms. Moreover, TLMRI provides up to 1.4 dB improvement in PSNR over the recent (partially adaptive) PBDWS method, and up to 1 dB improvement over the recent non-local patch similarity-based PANO method. Finally, the TLMRI reconstruction quality is somewhat (up to 0.33 dB) better than DLMRI. This is despite the latter using a 4 fold overcomplete (i.e., larger or richer) dictionary.

Fig. 2 shows the TLMRI reconstruction (Fig. 2(d)) of Image 2 for the case of 2D random sampling (sampling mask shown in Fig. 2(c)) and seven fold undersampling. The reconstruction errors (i.e., the magnitude of the difference between the magnitudes of the reconstructed and reference images) for several schemes are shown in Figs. 2 (e)-(h). The error map for TLMRI clearly shows the smallest image distortions. Fig. 3 shows the TLMRI (Fig. 3(c)) and DLMRI (Fig. 3(e)) reconstructions and reconstruction error maps (Figs. 3 (d), (f)) for Image 2 with Cartesian sampling (sampling mask shown in Fig. 3(b)) and seven fold undersampling. TLMRI provides a better reconstruction of image edges and better aliasing removal than DLMRI in this case.

The average run times of the Sparse MRI, PBDWS, PANO, DLMRI, and TLMRI algorithms in Table 1 are 251 seconds, 797 seconds, 400 seconds, 3273 seconds, and 243 seconds, respectively. The PBDWS run time includes the time taken for computing the initial SIDWT based reconstruction or guide image [51] in the PBDWS software package [61]. The TLMRI

<sup>14</sup>We observed that in Table 1, if we sampled vertical (instead of horizontal) lines for the Cartesian sampling patterns (by transposing the Cartesian sampling masks used in Table 1), the reconstructed images for TLMRI had PSNRs (32.52 dB and 31.05 dB respectively, for the 4x and 7x undersampling cases) similar to the horizontal lines sampling case. However, the learnt sparsifying transforms for the two cases had many dissimilar rows. This is not surprising since prior work on transform learning [73] has empirically shown that even the patches of a single image may admit multiple equally good sparsifying transforms, that may not be related by only row permutations or sign changes.

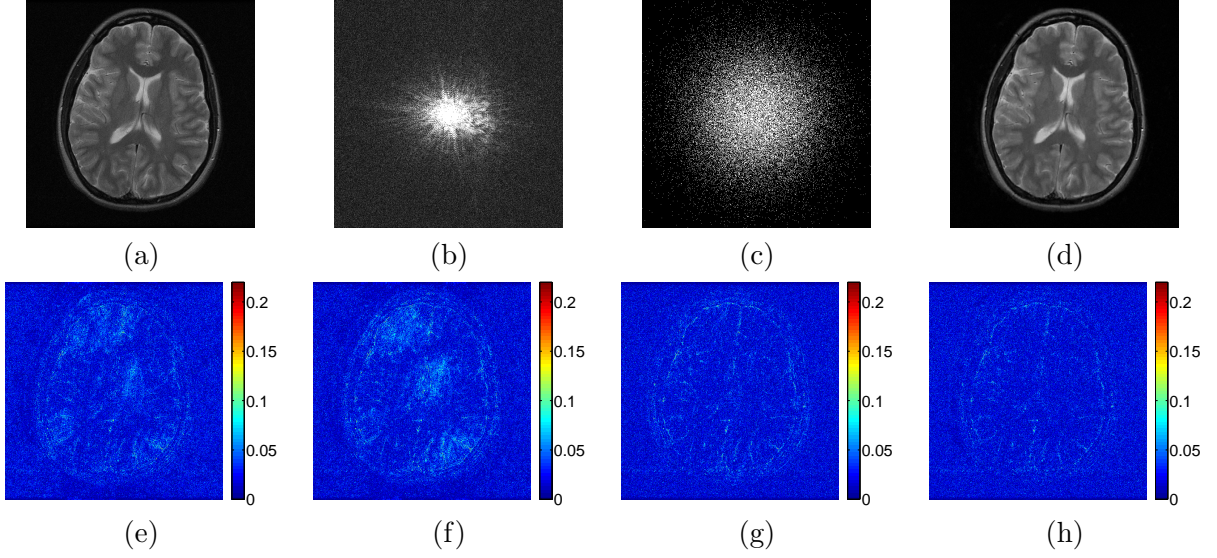


Figure 2: Results for 2D random sampling and 7x undersampling: (a) Reference image; (b) k-space of reference; (c) sampling mask in k-space; (d) TLMRI reconstruction (31.94 dB); (e) magnitude of PBDWS [51] reconstruction error; (f) magnitude of PANO [63] reconstruction error; (g) magnitude of DLMRI [64] reconstruction error; and (h) magnitude of TLMRI reconstruction error.

algorithm is thus the fastest one in Table 1, and provides a speedup of about 13.5x over the synthesis dictionary-based DLMRI, and a speedup of about 3.3x and 1.6x over the PBDWS and PANO<sup>15</sup> methods, respectively. Note that the speedups for TLMRI over PBDWS or PANO were obtained by comparing our unoptimized Matlab implementation of TLMRI against the MEX (or C) implementations of PBDWS and PANO.

While our results show some (preliminary) potential for the proposed sparsifying transform-based blind compressed sensing framework (for MRI), a much more detailed investigation will be presented elsewhere. Combining the proposed scheme with the patch-based directional Wavelets ideas [62, 51], or non-local patch similarity ideas [63, 48], or extending our framework to learning overcomplete sparsifying transforms (c.f., [81]) could potentially boost transform-based BCS performance further.

**6. Conclusions.** In this work, we presented a novel sparsifying transform-based framework for blind compressed sensing. Our formulations exploit the (adaptive) transform domain sparsity of overlapping image patches in 2D, or voxels in 3D. The proposed formulations are however highly nonconvex. Our block coordinate descent-type algorithms for solving the proposed problems involve efficient update steps. Importantly, our algorithms are guaranteed to converge to the critical points of the objectives defining the proposed formulations. These crit-

<sup>15</sup>Another faster version of the PANO method is also publicly available [60]. However, we found that although this version has an average run time of only 40 seconds in Table 1, it also provides worse reconstruction PSNRs than [59] in Table 1.



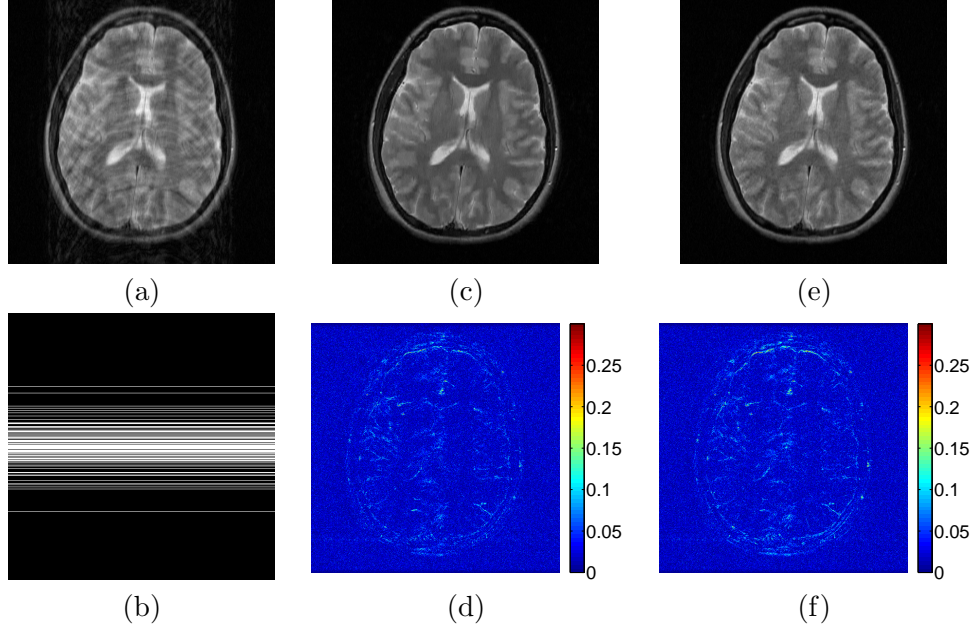


Figure 3: Cartesian sampling with 7 fold undersampling: (a) Initial zero-filling reconstruction (27.9 dB); (b) sampling mask in k-space; (c) TLMRI reconstruction (31.24 dB); (d) magnitude of TLMRI reconstruction error; (e) DLMRI reconstruction (30.91 dB); and (f) magnitude of DLMRI reconstruction error.

ical points are also guaranteed to be at least partial global and partial local minimizers. Our numerical examples showed the usefulness of the proposed scheme for magnetic resonance image reconstruction from highly undersampled k-space data. Our approach while being highly efficient also provides promising MR image reconstruction quality. The usefulness of the proposed blind compressed sensing methods in other inverse problems and imaging applications merits further study. A detailed investigation of the theoretical rate of convergence in our algorithms is also of potential interest.

#### Appendix. Convergence Proofs.

**A.1. Proof of Theorem 4.3.** In this proof, we let  $\tilde{H}_s(Z)$  denote the set of all optimal projections of  $Z \in \mathbb{C}^{n \times N}$  onto the  $s$ - $\ell_0$  ball  $\{B \in \mathbb{C}^{n \times N} : \|B\|_0 \leq s\}$ , i.e.,  $\tilde{H}_s(Z)$  is the set of minimizers for the following problem.

$$(A.1) \quad \tilde{H}_s(Z) = \arg \min_{B : \|B\|_0 \leq s} \|Z - B\|_F^2$$

Let  $\{W^t, B^t, x^t\}$  denote the iterate sequence generated by Algorithm A1 with measurements  $y \in \mathbb{C}^m$  and initial  $(W^0, B^0, x^0)$ . Assume that the initial  $(W^0, B^0, x^0)$  is such that  $g(W^0, B^0, x^0)$  is finite. Throughout this proof, we let  $X^t$  be the matrix with  $P_j x^t$  ( $1 \leq j \leq N$ ) as its columns. The various results in Theorem 4.3 are now proved in the following order.



- (i) The sequence of values of the objective in Algorithm A1 converges.
- (ii) The iterate sequence generated by Algorithm A1 has an accumulation point.
- (iii) All the accumulation points of the iterate sequence are equivalent in terms of their objective value.
- (iv) Every accumulation point of the iterates is a critical point of the objective  $g(W, B, x)$  satisfying (4.6), (4.7), and (4.8).
- (v) The difference between successive image iterates  $\|x^t - x^{t-1}\|_2$ , converges to zero.
- (vi) Every accumulation point of the iterates is a local minimizer of  $g(W, B, x)$  with respect to  $(B, x)$  or  $(W, B)$ .

The following two Lemmas establish the convergence of the objective, and the boundedness of the iterate sequence.

**Lemma A.1.** *Let  $\{W^t, B^t, x^t\}$  denote the iterate sequence generated by Algorithm A1 with input  $y \in \mathbb{C}^m$  and initial  $(W^0, B^0, x^0)$ . Then, the sequence of objective function values  $\{g(W^t, B^t, x^t)\}$  is monotone decreasing, and converges to a finite value  $g^* = g^*(W^0, B^0, x^0)$ .*

*Proof.* : Algorithm A1 first alternates between the transform update and sparse coding steps (Step 2 in algorithm pseudocode), with fixed image  $x$ . In the transform update step, we obtain a global minimizer (i.e., (3.7)) with respect to  $W$  for Problem (3.5). In the sparse coding step too, we obtain an exact solution for  $B$  in Problem (3.2) as  $\hat{B} = H_s(Z)$ . Therefore, the objective function can only decrease when we alternate between the transform update and sparse coding steps (similar to the case in [73]). Thus, we have  $g(W^{t+1}, B^{t+1}, x^t) \leq g(W^t, B^t, x^t)$ .

In the image update step of Algorithm A1 (Step 4 in algorithm pseudocode), we obtain an exact solution to the constrained least squares problem (3.10). Therefore, the objective in this step satisfies  $g(W^{t+1}, B^{t+1}, x^{t+1}) \leq g(W^{t+1}, B^{t+1}, x^t)$ . By the preceding arguments, we have  $g(W^{t+1}, B^{t+1}, x^{t+1}) \leq g(W^t, B^t, x^t)$ , for every  $t$ . Now, every term, except  $\lambda Q(W)$ , in the objective (4.3) is trivially non-negative. Furthermore, the  $Q(W)$  regularizer is bounded as  $Q(W) \geq \frac{n}{2}$  (cf. [70]). Therefore, the objective  $g(W, B, x) > 0$ . Since the sequence  $\{g(W^t, B^t, x^t)\}$  is monotone decreasing and lower bounded, it converges. ■

**Lemma A.2.** *The iterate sequence  $\{W^t, B^t, x^t\}$  generated by Algorithm A1 is bounded, and it has at least one accumulation point.*

*Proof.* : The existence of a convergent subsequence (and hence, an accumulation point) for a bounded sequence is a standard result. Therefore, we only prove the boundedness of the iterate sequence.

Since  $\|x^t\|_2 \leq C \forall t$  trivially, we have that the sequence  $\{x^t\}$  is bounded. We now show the boundedness of  $\{W^t\}$ . Let us denote the objective  $g(W^t, B^t, x^t)$  as  $g^t$ . It is obvious that the squared  $\ell_2$  norm terms and the barrier functions  $\psi(B^t)$  and  $\chi(x^t)$  in the objective  $g^t$  (4.3), are non-negative. Therefore, we have

$$(A.2) \quad \lambda Q(W^t) \leq g^t \leq g^0$$

where the second inequality follows from Lemma A.1. Now, the function  $Q(W^t) = \sum_{i=1}^n (0.5\alpha_i^2 - \log \alpha_i)$ , where  $\alpha_i$  ( $1 \leq i \leq n$ ) are the singular values of  $W^t$ , is a coercive function of the (non-negative) singular values, and therefore, it has bounded lower level sets

<sup>16</sup>. Combining this fact with (A.2), we can immediately conclude that  $\exists c_0 \in \mathbb{R}$  depending on  $g^0$  and  $\lambda$ , such that  $\|W^t\|_F \leq c_0 \forall t$ .

Finally, the boundedness of  $\{B^t\}$  follows from the following arguments. First, for Algorithm A1 (see pseudocode), we have that  $B^t = H_s(W^t X^{t-1})$ . Therefore, by the definition of  $H_s(\cdot)$ , we have

$$(A.3) \quad \|B^t\|_F = \|H_s(W^t X^{t-1})\|_F \leq \|W^t X^{t-1}\|_F \leq \|W^t\|_2 \|X^{t-1}\|_F$$

Since, by our previous arguments,  $W^t$  and  $X^{t-1}$  are both bounded by constants independent of  $t$ , we have that the sequence of sparse code matrices  $\{B^t\}$ , is also bounded. ■

We now establish some key optimality properties of the accumulation points of the iterate sequence in Algorithm A1.

**Lemma A.3.** *All the accumulation points of the iterate sequence generated by Algorithm A1 with a given initial  $(W^0, B^0, x^0)$  correspond to a common objective value  $g^*$ . Thus, they are equivalent in that sense.*

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . Before proving the lemma, we establish some simple properties of  $(W^*, B^*, x^*)$ . First, equation (A.2) implies that  $-\log |\det W^{q_t}| \leq (g^0/\lambda)$ , for every  $t$ . This further implies that  $|\det W^{q_t}| \geq e^{-g^0/\lambda} > 0 \forall t$ . Therefore, due to the continuity of the function  $|\det W|$ , the limit  $W^*$  of the subsequence is also non-singular, with  $|\det W^*| \geq e^{-g^0/\lambda}$ . Second,  $B^{q_t} = H_s(W^{q_t} X^{q_t-1})$ , where  $X^{q_t-1}$  is the matrix with  $P_j x^{q_t-1}$  ( $1 \leq j \leq N$ ) as its columns. Thus,  $B^{q_t}$  trivially satisfies  $\psi(B^{q_t}) = 0$  for every  $t$ . Now,  $\{B^{q_t}\}$  converges to  $B^*$ , which makes  $B^*$  the limit of a sequence of matrices, each of which has no more than  $s$  non-zeros. Thus, the limit  $B^*$  obviously cannot have more than  $s$  non-zeros. Therefore,  $\|B^*\|_0 \leq s$ , or equivalently  $\psi(B^*) = 0$ . Finally, since  $x^{q_t}$  satisfies the constraint  $\|x^{q_t}\|_2^2 \leq C^2$ , we have  $\chi(x^{q_t}) = 0 \forall t$ . Additionally, since  $x^{q_t} \rightarrow x^*$  as  $t \rightarrow \infty$ , we also have  $\|x^*\|_2^2 = \lim_{t \rightarrow \infty} \|x^{q_t}\|_2^2 \leq C^2$ . Therefore,  $\chi(x^*) = 0$ . Now, it is obvious from the above arguments that

$$(A.4) \quad \lim_{t \rightarrow \infty} \chi(x^{q_t}) = \chi(x^*), \quad \lim_{t \rightarrow \infty} \psi(B^{q_t}) = \psi(B^*)$$

Moreover, due to the continuity of  $Q(W)$  at non-singular matrices  $W$ , we have  $\lim_{t \rightarrow \infty} Q(W^{q_t}) = Q(W^*)$ . We now use these limits along with some simple properties of convergent sequences (e.g., the limit of the sum/product of convergent sequences is equal to the sum/product of their limits), to arrive at the following result.

$$(A.5) \quad \begin{aligned} & \lim_{t \rightarrow \infty} g(W^{q_t}, B^{q_t}, x^{q_t}) = \\ & \lim_{t \rightarrow \infty} \left\{ \sum_{j=1}^N \|W^{q_t} P_j x^{q_t} - b_j^{q_t}\|_2^2 + \nu \|A x^{q_t} - y\|_2^2 + \lambda Q(W^{q_t}) + \psi(B^{q_t}) \right\} \\ & + \lim_{t \rightarrow \infty} \chi(x^{q_t}) = \sum_{j=1}^N \|W^* P_j x^* - b_j^*\|_2^2 + \nu \|A x^* - y\|_2^2 + \lambda Q(W^*) + \psi(B^*) + \chi(x^*) \\ & = g(W^*, B^*, x^*) \end{aligned}$$

---

<sup>16</sup>The lower level sets of a function  $f : A \subset \mathbb{R}^n \mapsto \mathbb{R}$  (where  $A$  is unbounded) are bounded if  $\lim_{t \rightarrow \infty} f(z^t) = +\infty$  whenever  $\{z^t\} \subset A$  and  $\lim_{t \rightarrow \infty} \|z^t\| = \infty$ .

The above result together with the fact (from Lemma A.1) that  $\lim_{t \rightarrow \infty} g(W^{q_t}, B^{q_t}, x^{q_t}) = g^*$  implies that  $g(W^*, B^*, x^*) = g^*$ . ■

The following lemma establishes partial global optimality with respect to the image, of every accumulation point.

**Lemma A.4.** *Any accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 satisfies*

$$(A.6) \quad x^* \in \arg \min_x g(W^*, B^*, x)$$

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . Then, due to the optimality of  $x^{q_t}$  in the image update step of Algorithm A1, we have the following inequality for any  $t$  and any  $x \in \mathbb{C}^p$ .

$$(A.7) \quad \sum_{j=1}^N \|W^{q_t} P_j x^{q_t} - b_j^{q_t}\|_2^2 + \nu \|Ax^{q_t} - y\|_2^2 + \chi(x^{q_t}) \leq \sum_{j=1}^N \|W^{q_t} P_j x - b_j^{q_t}\|_2^2 + \nu \|Ax - y\|_2^2 + \chi(x)$$

Now, by (A.4), we have that  $\lim_{t \rightarrow \infty} \chi(x^{q_t}) = \chi(x^*)$ . Taking the limit  $t \rightarrow \infty$  term by term on both sides of (A.7) for some fixed  $x \in \mathbb{C}^p$  yields the following result

$$(A.8) \quad \sum_{j=1}^N \|W^* P_j x^* - b_j^*\|_2^2 + \nu \|Ax^* - y\|_2^2 + \chi(x^*) \leq \sum_{j=1}^N \|W^* P_j x - b_j^*\|_2^2 + \nu \|Ax - y\|_2^2 + \chi(x)$$

Since the choice of  $x$  in (A.8) is arbitrary, (A.8) holds for any  $x \in \mathbb{C}^p$ . Recall that  $\psi(B^*) = 0$  and  $Q(W^*)$  is finite based on the arguments in the proof of Lemma A.3. Therefore, (A.8) implies that  $g(W^*, B^*, x^*) \leq g(W^*, B^*, x) \forall x \in \mathbb{C}^p$ . This establishes the result (A.6) of the Lemma. ■

The following lemma will be used to establish that the change between successive image iterates  $\|x^t - x^{t-1}\|_2$ , converges to 0.

**Lemma A.5.** *Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . Then, the subsequence  $\{x^{q_t-1}\}$  also converges to  $x^*$ .*

*Proof.* : First, by Lemma A.3, we have that

$$(A.9) \quad g(W^*, B^*, x^*) = g^*$$

Next, consider a convergent subsequence  $\{x^{q_{n_t}-1}\}$  of (the bounded sequence)  $\{x^{q_t-1}\}$  that converges to say  $x^{**}$ . Now, applying the same arguments as in Equation (A.5) (in the proof of Lemma A.3), but with respect to the (convergent) subsequence  $\{x^{q_{n_t}-1}, W^{q_{n_t}}, B^{q_{n_t}}\}$ , we have that

$$(A.10) \quad \lim_{t \rightarrow \infty} g(W^{q_{n_t}}, B^{q_{n_t}}, x^{q_{n_t}-1}) = g(W^*, B^*, x^{**})$$

Now, the monotonic decrease of the objective in Algorithm A1 implies

$$(A.11) \quad g(W^{q_{n_t}}, B^{q_{n_t}}, x^{q_{n_t}}) \leq g(W^{q_{n_t}}, B^{q_{n_t}}, x^{q_{n_t}-1}) \leq g(W^{q_{n_t}-1}, B^{q_{n_t}-1}, x^{q_{n_t}-1})$$

Taking the limit  $t \rightarrow \infty$  all through (A.11) and using Lemma A.1 (for the extreme left/right limits), and Equation A.10 (for the middle limit), immediately yields that  $g(W^*, B^*, x^{**}) = g^*$ . This result together with (A.9) implies  $g(W^*, B^*, x^{**}) = g(W^*, B^*, x^*)$ .

Now, we know by Lemma A.4 that

$$(A.12) \quad x^* \in \arg \min_x g(W^*, B^*, x)$$

Furthermore, it was shown in Section 3.1.3 that if the set of patches in our formulation (P1) cover all pixels in the image (always true for the case of periodically positioned overlapping image patches), then the minimization of  $g(W^*, B^*, x)$  with respect to  $x$  has a unique solution. Therefore,  $x^*$  is the unique minimizer in (A.12). Combining this with the fact that  $g(W^*, B^*, x^{**}) = g(W^*, B^*, x^*)$  yields that  $x^{**} = x^*$ . Since we worked with an arbitrary convergent subsequence  $\{x^{q_{n_t}-1}\}$  (of  $\{x^{q_t-1}\}$ ) in the above proof, we have that  $x^*$  is the limit of any convergent subsequence of  $\{x^{q_t-1}\}$ . Finally, since every convergent subsequence of the bounded sequence  $\{x^{q_t-1}\}$  converges to  $x^*$ , it means that the (compact) set of accumulation points of  $\{x^{q_t-1}\}$  coincides with  $x^*$ . Since  $x^*$  is the unique accumulation point of a bounded sequence, we therefore have [44] that the sequence  $\{x^{q_t-1}\}$  itself converges to  $x^*$ , which completes the proof. ■

**Lemma A.6.** *The iterate sequence  $\{W^t, B^t, x^t\}$  in Algorithm A1 satisfies*

$$(A.13) \quad \lim_{t \rightarrow \infty} \|x^t - x^{t-1}\|_2 = 0$$

*Proof.* : Consider the sequence  $\{a^t\}$  with  $a^t \triangleq \|x^t - x^{t-1}\|_2$ . We will show below that every convergent subsequence of the bounded sequence  $\{a^t\}$  converges to 0, thereby implying that 0 is both the limit inferior and limit superior of  $\{a^t\}$ , which means that the sequence  $\{a^t\}$  itself converges to 0.

Now, consider a convergent subsequence  $\{a^{q_t}\}$  of  $\{a^t\}$ . Since the sequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  is bounded, there exists a convergent subsequence  $\{W^{q_{n_t}}, B^{q_{n_t}}, x^{q_{n_t}}\}$  converging to say  $(W^*, B^*, x^*)$ . By Lemma A.5, we then have that  $\{x^{q_{n_t}-1}\}$  also converges to  $x^*$ . Thus, the subsequence  $\{a^{q_{n_t}}\}$  with  $a^{q_{n_t}} \triangleq \|x^{q_{n_t}} - x^{q_{n_t}-1}\|_2$  converges to 0. Since,  $\{a^{q_{n_t}}\}$  itself is a subsequence of a convergent sequence, we must have that  $\{a^{q_t}\}$  converges to the same limit (i.e., 0). We have thus shown that zero is the limit of any convergent subsequence of  $\{a^t\}$ . ■

The next property is the partial global optimality of every accumulation point with respect to the sparse code. In order to establish this property, we need the following result.

**Lemma A.7.** *Consider a bounded matrix sequence  $\{Z^k\}$  with  $Z^k \in \mathbb{C}^{n \times N}$ , that converges to  $Z^*$ . Then, every accumulation point of  $\{H_s(Z^k)\}$  belongs to the set  $\tilde{H}_s(Z^*)$ .*

*Proof.* : The proof is very similar to that for Lemma 10 in [73]. ■

**Lemma A.8.** *Any accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 satisfies*

$$(A.14) \quad B^* \in \arg \min_B g(W^*, B, x^*)$$

Moreover, denoting by  $X^* \in \mathbb{C}^{n \times N}$  the matrix whose  $j^{\text{th}}$  column is  $P_j x^*$ , for  $1 \leq j \leq N$ , the above condition can be equivalently stated as

$$(A.15) \quad B^* \in \tilde{H}_s(W^* X^*)$$

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . Then, by Lemma A.5,  $\{x^{q_t-1}\}$  converges to  $x^*$ , and the following inequalities hold.

$$(A.16) \quad B^* = \lim_{t \rightarrow \infty} B^{q_t} = \lim_{t \rightarrow \infty} H_s(W^{q_t} X^{q_t-1}) \in \tilde{H}_s(W^* X^*)$$

Since  $W^{q_t} X^{q_t-1} \rightarrow W^* X^*$  as  $t \rightarrow \infty$ , the last containment relationship above follows by Lemma A.7. While we have proved (A.15), the result in (A.14) now immediately follows by applying the definition of  $\tilde{H}_s(\cdot)$  from (A.1). ■

The next result pertains to the partial global optimality with respect to  $W$ , of every accumulation point in Algorithm A1. We use the following lemma (simple extension of Lemma 1 in [73] to the complex field) to establish the result.

**Lemma A.9.** Consider a sequence  $\{M_k\}$  with  $M_k \in \mathbb{C}^{n \times n}$ , that converges to  $M$ . For each  $k$ , let  $V_k \Sigma_k R_k^H$  denote a full SVD of  $M_k$ . Then, every accumulation point  $(V, \Sigma, R)$  of the sequence  $\{V_k, \Sigma_k, R_k\}$  is such that  $V \Sigma R^H$  is a full SVD of  $M$ . In particular,  $\{\Sigma_k\}$  converges to  $\Sigma$ , the  $n \times n$  singular value matrix of  $M$ .

**Lemma A.10.** Any accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 satisfies

$$(A.17) \quad W^* \in \arg \min_W g(W, B^*, x^*)$$

*Proof.* : The proposed Algorithm A1 involves  $\hat{M}$  alternations between the transform update and sparse coding steps in every outer iteration. At a certain (outer) iteration  $t$ , let us denote the intermediate transform update outputs as  $\tilde{W}^{(l,t)}$ ,  $1 \leq l \leq \hat{M}$ . Then,  $W^t = \tilde{W}^{(\hat{M},t)}$ . We will only use the sequence  $\{\tilde{W}^{(1,t)}\}$  (i.e., the intermediate outputs for  $l = 1$ ) in the following proof.

Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence in Algorithm A1, that converges to the accumulation point  $(W^*, B^*, x^*)$ . Then,  $\tilde{W}^{(1,q_t+1)}$  is computed as follows, using the full SVD  $V^{q_t} \Sigma^{q_t} (R^{q_t})^H$ , of  $(L^{q_t})^{-1} X^{q_t} (B^{q_t})^H$ , where  $L^{q_t} = \left( X^{q_t} (X^{q_t})^H + 0.5\lambda I \right)^{1/2}$ .

$$(A.18) \quad \tilde{W}^{(1,q_t+1)} = 0.5 R^{q_t} \left( \Sigma^{q_t} + \left( (\Sigma^{q_t})^2 + 2\lambda I \right)^{\frac{1}{2}} \right) (V^{q_t})^H (L^{q_t})^{-1}$$

To prove the lemma, we will consider the limit  $t \rightarrow \infty$  in (A.18). In order to take this limit, we need the following results. First, due to the continuity of the matrix square root and matrix

inverse functions at positive definite matrices, we have that the following limit holds, where  $X^* \in \mathbb{C}^{n \times N}$  has  $P_j x^*$  ( $1 \leq j \leq N$ ) as its columns.

$$\lim_{t \rightarrow \infty} (L^{q_t})^{-1} = \lim_{t \rightarrow \infty} \left( X^{q_t} (X^{q_t})^H + 0.5\lambda I \right)^{-1/2} = \left( X^* (X^*)^H + 0.5\lambda I \right)^{-1/2}$$

Next, defining  $L^* \triangleq \left( X^* (X^*)^H + 0.5\lambda I \right)^{1/2}$ , we also have that

$$(A.19) \quad \lim_{t \rightarrow \infty} (L^{q_t})^{-1} X^{q_t} (B^{q_t})^H = (L^*)^{-1} X^* (B^*)^H$$

Applying Lemma A.9 to (A.19), we have that every accumulation point  $(V^*, \Sigma^*, R^*)$  of the sequence  $\{V^{q_t}, \Sigma^{q_t}, R^{q_t}\}$  is such that  $V^* \Sigma^* (R^*)^H$  is a full SVD of  $(L^*)^{-1} X^* (B^*)^H$ . Now, consider a convergent subsequence  $\{V^{q_{n_t}}, \Sigma^{q_{n_t}}, R^{q_{n_t}}\}$  of  $\{V^{q_t}, \Sigma^{q_t}, R^{q_t}\}$ , with limit  $(V^*, \Sigma^*, R^*)$ . Then, taking the limit  $t \rightarrow \infty$  in (A.18) along this subsequence, we have

$$W^{**} \triangleq \lim_{t \rightarrow \infty} \tilde{W}^{(1, q_{n_t} + 1)} = 0.5 R^* \left( \Sigma^* + \left( (\Sigma^*)^2 + 2\lambda I \right)^{\frac{1}{2}} \right) (V^*)^H (L^*)^{-1}$$

Combining this result with the aforementioned definitions of the square root  $L^*$  and the full SVD  $V^* \Sigma^* (R^*)^H$ , and applying Proposition 3.1, we get

$$(A.20) \quad W^{**} \in \arg \min_W g(W, B^*, x^*)$$

Finally, applying the same arguments as in the proof of Lemma A.3 to the subsequence  $\{B^{q_{n_t}}, x^{q_{n_t}}, \tilde{W}^{(1, q_{n_t} + 1)}\}$ , we easily get that  $g^* = g(W^{**}, B^*, x^*)$ . Since, by Lemma A.3, we also have that  $g^* = g(W^*, B^*, x^*)$ , we get  $g(W^*, B^*, x^*) = g(W^{**}, B^*, x^*)$ , which together with (A.20) immediately establishes the required result (A.17). ■

The following lemma establishes that every accumulation point of the iterate sequence in Algorithm A1 is a critical point of the objective  $g(W, B, x)$ . All derivatives or sub-differentials are computed with respect to the real and imaginary parts of the corresponding variables/vectors/matrices below.

**Lemma A.11.** *Every accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 is a critical point of the objective  $g(W, B, x)$  satisfying*

$$(A.21) \quad 0 \in \partial g(W^*, B^*, x^*)$$

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . By Lemmas A.10, A.4, and A.8, we have that

$$(A.22) \quad W^* \in \arg \min_W g(W, B^*, x^*)$$

$$(A.23) \quad x^* \in \arg \min_x g(W^*, B^*, x)$$

$$(A.24) \quad B^* \in \arg \min_B g(W^*, B, x^*)$$



The function  $g(W, B, x)$  is continuously differentiable with respect to  $W$  at non-singular points  $W$ . Since  $W^*$  is a (non-singular) partial global minimizer in (A.22), we have as a necessary condition that  $\nabla_W g(W^*, B^*, x^*) = 0$ . Next, recall the statement in Section 4.1, that a necessary condition for  $z \in \mathbb{R}^q$  to be a minimizer of some function  $\phi : \mathbb{R}^q \mapsto (-\infty, +\infty]$  is that  $z$  satisfies  $0 \in \partial\phi(z)$ . Now, since  $x^*$  and  $B^*$  are partial global minimizers in (A.23) and (A.24), respectively, we therefore have that  $0 \in \partial_{g_x}(W^*, B^*, x^*)$  and  $0 \in \partial_{g_B}(W^*, B^*, x^*)$ , respectively. It is also easy to derive these conditions directly using the definition (Definition 4.2) of the sub-differential.

Finally, (see Proposition 3 in [3]) the subdifferential  $\partial g$  at  $(W^*, B^*, x^*)$  satisfies

$$(A.25) \quad \partial g(W^*, B^*, x^*) = \nabla_W g(W^*, B^*, x^*) \times \partial_{g_B}(W^*, B^*, x^*) \times \partial_{g_x}(W^*, B^*, x^*)$$

Now, using the preceding results, we easily have that  $0 \in \partial g(W^*, B^*, x^*)$  above. Thus, every accumulation point in Algorithm A1 is a critical point of the objective. ■

The following two lemmas establish pairwise partial local optimality of the accumulation points in Algorithm A1. Here,  $X^* \in \mathbb{C}^{n \times N}$  is the matrix with  $P_j x^*$  as its columns.

**Lemma A.12.** *Every accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 is a partial minimizer of the objective  $g(W, B, x)$  with respect to  $(W, B)$ , in the sense of (4.9), for sufficiently small  $dW \in \mathbb{C}^{n \times n}$ , and all  $\Delta B \in \mathbb{C}^{n \times N}$  in the union of the regions R1 and R2 in Theorem 4.3. Furthermore, if  $\|W^* X^*\|_0 \leq s$ , then the  $\Delta B$  in (4.9) can be arbitrary.*

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . First, by Lemmas A.8 and A.10, we have

$$(A.26) \quad B^* \in \tilde{H}_s(W^* X^*)$$

$$(A.27) \quad 2W^* X^* (X^*)^H - 2B^* (X^*)^H + \lambda W^* - \lambda (W^*)^{-H} = 0$$

where (A.27) follows from the first order conditions for partial global optimality of  $W^*$  in (A.17). The accumulation point  $(W^*, B^*, x^*)$  also satisfies  $\psi(B^*) = 0$  and  $\chi(x^*) = 0$ .

Now, considering perturbations  $dW \in \mathbb{C}^{n \times n}$ , and  $\Delta B \in \mathbb{C}^{n \times N}$ , we have that

$$(A.28) \quad \begin{aligned} g(W^* + dW, B^* + \Delta B, x^*) &= \|W^* X^* - B^* + (dW)X^* - \Delta B\|_F^2 \\ &+ \nu \|Ax^* - y\|_2^2 + 0.5\lambda \|W^* + dW\|_F^2 - \lambda \log |\det(W^* + dW)| + \psi(B^* + \Delta B) \end{aligned}$$

In order to prove the condition (4.9) in Theorem 4.3, it suffices to consider *sparsity preserving* perturbations  $\Delta B$ , that is  $\Delta B \in \mathbb{C}^{n \times N}$  such that  $B^* + \Delta B$  has sparsity  $\leq s$ . Otherwise  $g(W^* + dW, B^* + \Delta B, x^*) = +\infty > g(W^*, B^*, x^*)$  trivially. Therefore, we only consider sparsity preserving  $\Delta B$  in the following, for which  $\psi(B^* + \Delta B) = 0$  in (A.28).

Since the image  $x^*$  is fixed here, we can utilize (A.26) and (A.27), and apply similar arguments as in the proof of Lemma 9 (equations (43)-(46)) in [73], to simplify the right hand side in (A.28). The only difference is that the matrix transpose  $(\cdot)^T$  operations in [73] are replaced with Hermitian  $(\cdot)^H$  operations here, and the operation  $\langle Q, R \rangle$  involving two matrices

(or, vectors)  $Q$  and  $R$  in [73] is redefined<sup>17</sup> here as

$$(A.29) \quad \langle Q, R \rangle \triangleq \text{Re} \{ \text{tr}(QR^H) \}$$

Upon such simplifications, we can conclude [73] that  $\exists \epsilon' > 0$  depending on  $W^*$  such that whenever  $\|dW\|_F < \epsilon'$ , we have

$$(A.30) \quad g(W^* + dW, B^* + \Delta B, x^*) \geq g(W^*, B^*, x^*) - 2 \langle W^* X^* - B^*, \Delta B \rangle$$

Consider first the case of  $\Delta B$  in region R1. Then, the term  $-\langle W^* X^* - B^*, \Delta B \rangle$  above is trivially non-negative for such  $\Delta B$  in Theorem 4.3, and therefore,  $g(W^* + dW, B^* + \Delta B, x^*) \geq g(W^*, B^*, x^*)$  for  $\Delta B \in \text{R1}$ .

Next, consider the case of  $\Delta B$  in region R2. Then, when  $\|W^* X^*\|_0 > s$ , it is easy to see that any such sparsity preserving  $\Delta B$  in region R2 in Theorem 4.3 will have its support (i.e., non-zero locations) contained in the support of  $B^* \in \tilde{H}_s(W^* X^*)$ . Therefore,  $\langle W^* X^* - B^*, \Delta B \rangle = 0$  in this case. On the other hand, if  $\|W^* X^*\|_0 \leq s$ , then by (A.26),  $W^* X^* - B^* = 0$ . Therefore, by these arguments,  $\langle W^* X^* - B^*, \Delta B \rangle = 0$  for any sparsity preserving  $\Delta B \in \text{R2}$ . This result together with (A.30) implies  $g(W^* + dW, B^* + \Delta B, x^*) \geq g(W^*, B^*, x^*)$  for any  $\Delta B \in \text{R2}$ .

Finally, if  $\|W^* X^*\|_0 \leq s$ , then since  $W^* X^* - B^* = 0$  in (A.30), therefore in this case,  $\Delta B$  in (4.9) can be arbitrary. ■

**Lemma A.13.** *Every accumulation point  $(W^*, B^*, x^*)$  of the iterate sequence generated by Algorithm A1 is a partial minimizer of the objective  $g(W, B, x)$  with respect to  $(B, x)$ , in the sense of (4.10), for all  $\tilde{\Delta}x \in \mathbb{C}^p$ , and all  $\Delta B \in \mathbb{C}^{n \times N}$  in the union of the regions R1 and R2 in Theorem 4.3. Furthermore, if  $\|W^* X^*\|_0 \leq s$ , then the  $\Delta B$  in (4.10) can be arbitrary.*

*Proof.* : Consider the subsequence  $\{W^{q_t}, B^{q_t}, x^{q_t}\}$  (indexed by  $q_t$ ) of the iterate sequence, that converges to the accumulation point  $(W^*, B^*, x^*)$ . It follows from Lemmas A.8 and A.4 that  $\psi(B^*) = \chi(x^*) = 0$ . Now, considering perturbations  $\Delta B \in \mathbb{C}^{n \times N}$  whose columns are denoted as  $\Delta b_j$  ( $1 \leq j \leq N$ ), and  $\tilde{\Delta}x \in \mathbb{C}^p$ , we have that

$$(A.31) \quad \begin{aligned} g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) &= \sum_{j=1}^N \left\| W^* P_j x^* - b_j^* + W^* P_j \tilde{\Delta}x - \Delta b_j \right\|_2^2 \\ &+ \lambda Q(W^*) + \nu \|Ax^* - y + A\tilde{\Delta}x\|_2^2 + \psi(B^* + \Delta B) + \chi(x^* + \tilde{\Delta}x) \end{aligned}$$

In order to prove the condition (4.10) in Theorem 4.3, it suffices to consider *sparsity preserving* perturbations  $\Delta B$ , that is  $\Delta B \in \mathbb{C}^{n \times N}$  such that  $B^* + \Delta B$  has sparsity  $\leq s$ . It also suffices to consider *energy preserving* perturbations  $\tilde{\Delta}x$ , which are such that  $\|x^* + \tilde{\Delta}x\|_2 \leq C$ . For any other  $\Delta B$  or  $\tilde{\Delta}x$ ,  $g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) = +\infty > g(W^*, B^*, x^*)$  trivially. Therefore, we only consider the energy/sparsity preserving perturbations in the following, for which  $\psi(B^* + \Delta B) = 0$  and  $\chi(x^* + \tilde{\Delta}x) = 0$  in (A.31). Now, upon expanding the squared  $\ell_2$  terms

<sup>17</sup>We include the  $\text{Re}(\cdot)$  operation in the definition here, which allows for simpler notations in the rest of the proof. However, the  $\langle Q, R \rangle$  defined in (A.29) is no longer the conventional inner product.

in (A.31), and dropping non-negative perturbation terms, we get

$$\begin{aligned}
 g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) &\geq g(W^*, B^*, x^*) - 2 \sum_{j=1}^N \langle W^* P_j x^* - b_j^*, \Delta b_j \rangle \\
 &\quad + 2 \sum_{j=1}^N \langle W^* P_j x^* - b_j^*, W^* P_j \tilde{\Delta}x \rangle + 2\nu \langle Ax^* - y, A\tilde{\Delta}x \rangle
 \end{aligned}
 \tag{A.32}$$

where the  $\langle \cdot, \cdot \rangle$  notation is as defined in (A.29). Now, using arguments identical to those used with (A.30), it is clear that the term  $-2 \sum_{j=1}^N \langle W^* P_j x^* - b_j^*, \Delta b_j \rangle \geq 0$  in (A.32) for all sparsity preserving  $\Delta B \in \text{R1} \cup \text{R2}$ . Furthermore, since by Lemma A.4,  $x^*$  is a global minimizer of  $g(W^*, B^*, x)$ , it must satisfy the Normal equation (3.12) for some (unique)  $\mu \geq 0$ . Using these arguments, (A.32) simplifies to

$$g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) \geq g(W^*, B^*, x^*) - 2\mu \langle x^*, \tilde{\Delta}x \rangle
 \tag{A.33}$$

Now, if (the optimal)  $\mu = 0$  above, then  $g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) \geq g(W^*, B^*, x^*)$  for arbitrary  $\tilde{\Delta}x \in \mathbb{C}^p$ , and  $\Delta B \in \text{R1} \cup \text{R2}$ . The alternative scenario is the case when (the optimal)  $\mu > 0$ , which can occur only if  $\|x^*\|_2 = C$  holds. Since we are considering energy preserving perturbations  $\tilde{\Delta}x$ , we have  $\|x^* + \tilde{\Delta}x\|_2^2 \leq C^2 = \|x^*\|_2^2$ , which implies  $-2 \langle x^*, \tilde{\Delta}x \rangle \geq \|\tilde{\Delta}x\|_2^2 \geq 0$ . Combining this result with (A.33), we again have (now for the  $\mu > 0$  case) that  $g(W^*, B^* + \Delta B, x^* + \tilde{\Delta}x) \geq g(W^*, B^*, x^*)$  for arbitrary  $\tilde{\Delta}x \in \mathbb{C}^p$ , and  $\Delta B \in \text{R1} \cup \text{R2}$ . ■

**A.2. Proofs of Theorems 4.6 and 4.7.** The proofs of Theorems 4.6 and 4.7 are very similar to that for Theorem 4.3. We only discuss some of the minor differences, as follows.

First, in the case of Theorem 4.6, the main difference in the proof is that the non-negative barrier function  $\psi(B)$  and the operator  $H_s(\cdot)$  (in the proof of Theorem 4.3) are replaced by the non-negative penalty  $\eta^2 \sum_{j=1}^N \|b_j\|_0$  and the operator  $\hat{H}_\eta^1(\cdot)$ , respectively. Moreover, the mapping  $\tilde{H}_s(\cdot)$  defined in (A.1) for the proof of Theorem 4.3, is replaced by the matrix-to-set mapping  $\hat{H}_\eta(\cdot)$  (for the proof of Theorem 4.6) defined as

$$\left( \hat{H}_\eta(Z) \right)_{ij} = \begin{cases} 0 & , \quad |Z_{ij}| < \eta \\ \{Z_{ij}, 0\} & , \quad |Z_{ij}| = \eta \\ Z_{ij} & , \quad |Z_{ij}| > \eta \end{cases}
 \tag{A.34}$$

By thus replacing the relevant functions and operators, the various steps in the proof of Theorem 4.3 can be easily extended to the case of Theorem 4.6. As such, Theorem 4.6 mainly differs from Theorem 4.3 in terms of the definition of the set of allowed (local) perturbations  $\Delta B$ . In particular, the proofs of partial local optimality of accumulation points in Theorem 4.6 are easily extended from the aforementioned proofs for Lemmas A.12 and A.13, by using the techniques and inequalities presented in Appendix F of [73].

Finally, the main difference between the proofs of Theorems 4.3 and 4.7 is that the non-negative  $\lambda Q(W)$  penalty in the former is replaced by the barrier function  $\varphi(W)$  (that enforces

the unitary property, and keeps  $W^t$  always bounded) in the latter. Otherwise, the proof techniques are very similar for the two cases.

## REFERENCES

- [1] M. AHARON AND M. ELAD, *Sparse and redundant modeling of image content using an image-signature-dictionary*, SIAM Journal on Imaging Sciences, 1 (2008), pp. 228–247.
- [2] M. AHARON, M. ELAD, AND A. BRUCKSTEIN, *K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation*, IEEE Transactions on signal processing, 54 (2006), pp. 4311–4322.
- [3] H. ATTOUCH, J. BOLTE, P. REDONT, AND A. SOUBEYRAN, *Proximal alternating minimization and projection methods for nonconvex problems: An approach based on the kurdyka-lojasiewicz inequality*, Math. Oper. Res., 35 (2010), pp. 438–457.
- [4] Y. BRESLER, *Spectrum-blind sampling and compressive sensing for continuous-index signals*, in 2008 Information Theory and Applications Workshop Conference, 2008, pp. 547–554.
- [5] Y. BRESLER AND P. FENG, *Spectrum-blind minimum-rate sampling and reconstruction of 2-D multiband signals*, in Proc. 3rd IEEE Int. Conf. on Image Processing, ICIP’96, sep 1996, pp. 701–704.
- [6] Y. BRESLER, M. GASTPAR, AND R. VENKATARAMANI, *Image compression on-the-fly by universal sampling in Fourier imaging systems*, in Proc. 1999 IEEE Information Theory Workshop on Detection, Estimation, Classification, and Imaging, feb 1999, p. 48.
- [7] A. M. BRUCKSTEIN, D. L. DONOHO, AND M. ELAD, *From sparse solutions of systems of equations to sparse modeling of signals and images*, SIAM Review, 51 (2009), pp. 34–81.
- [8] E. CANDÈS, J. ROMBERG, AND T. TAO, *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Information Theory, 52 (2006), pp. 489–509.
- [9] E. CANDÈS AND T. TAO, *Decoding by linear programming*, IEEE Trans. on Information Theory, 51 (2005), pp. 4203–4215.
- [10] R. CHARTRAND, *Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data*, in Proc. IEEE International Symposium on Biomedical Imaging (ISBI), 2009, pp. 262–265.
- [11] G. H. CHEN, J. TANG, AND S. LENG, *Prior image constrained compressed sensing (piccs): A method to accurately reconstruct dynamic CT images from highly undersampled projection data sets*, Med. Phys., 35 (2008), pp. 660–663.
- [12] S. S. CHEN, D. L. DONOHO, AND M. A. SAUNDERS, *Atomic decomposition by basis pursuit*, SIAM J. Sci. Comput., 20 (1998), pp. 33–61.
- [13] K. CHOI, J. WANG, L. ZHU, T.-S. SUH, S. BOYD, AND L. XING, *Compressed sensing based cone-beam computed tomography reconstruction with a first-order method*, Med. Phys., 37 (2010), pp. 5113–5125.
- [14] E. CHOUZENOUX, J.-C. PESQUET, AND A. REPETTI, *A block coordinate variable metric forward-backward algorithm*, (2014). Preprint: [http://www.optimization-online.org/DB\\_HTML/2013/12/4178.html](http://www.optimization-online.org/DB_HTML/2013/12/4178.html).
- [15] K. DABOV, A. FOI, V. KATKOVNIK, AND K. EGIAZARIAN, *Image denoising by sparse 3D transform-domain collaborative filtering*, IEEE Trans. on Image Processing, 16 (2007), pp. 2080–2095.
- [16] W. DAI AND O. MILENKOVIC, *Subspace pursuit for compressive sensing signal reconstruction*, IEEE Trans. Information Theory, 55 (2009), pp. 2230–2249.
- [17] G. DAVIS, S. MALLAT, AND M. AVELLANEDA, *Adaptive greedy approximations*, Journal of Constructive Approximation, 13 (1997), pp. 57–98.
- [18] M. N. DO AND M. VETTERLI, *The contourlet transform: an efficient directional multiresolution image representation*, IEEE Trans. Image Process., 14 (2005), pp. 2091–2106.
- [19] D. DONOHO, *Compressed sensing*, IEEE Trans. Information Theory, 52 (2006), pp. 1289–1306.
- [20] D. L. DONOHO, *For most large underdetermined systems of linear equations the minimal  $l_1$ -norm solution is also the sparsest solution*, Comm. Pure Appl. Math, 59 (2004), pp. 797–829.
- [21] D. L. DONOHO, M. ELAD, AND V. N. TEMLYAKOV, *Stable recovery of sparse overcomplete representations in the presence of noise*, IEEE Trans. Inform. Theory, 52 (2006), pp. 6–18.
- [22] B. EFRON, T. HASTIE, I. JOHNSTONE, AND R. TIBSHIRANI, *Least angle regression*, Annals of Statistics, 32 (2004), pp. 407–499.
- [23] M. ELAD AND M. AHARON, *Image denoising via sparse and redundant representations over learned dictionaries*, IEEE Trans. Image Process., 15 (2006), pp. 3736–3745.

- [24] M. ELAD, P. MILANFAR, AND R. RUBINSTEIN, *Analysis versus synthesis in signal priors*, Inverse Problems, 23 (2007), pp. 947–968.
- [25] L. ELDÉN, *Solving quadratically constrained least squares problems using a differential-geometric approach*, BIT Numerical Mathematics, 42 (2002), pp. 323–335.
- [26] K. ENGAN, S.O. AASE, AND J.H. HAKON-HUSOY, *Method of optimal directions for frame design*, in Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 1999, pp. 2443–2446.
- [27] P. FENG, *Universal Spectrum Blind Minimum Rate Sampling and Reconstruction of Multiband Signals*, PhD thesis, University of Illinois at Urbana-Champaign, mar 1997. Yoram Bresler, adviser.
- [28] P. FENG AND Y. BRESLER, *Spectrum-blind minimum-rate sampling and reconstruction of multiband signals*, in ICASSP, vol. 3, may 1996, pp. 1689–1692.
- [29] M. GASTPAR AND Y. BRESLER, *On the necessary density for spectrum-blind nonuniform sampling subject to quantization*, in ICASSP, vol. 1, jun 2000, pp. 348–351.
- [30] S. GLEICHMAN AND YONINA C. ELДАР, *Blind compressed sensing*, IEEE Transactions on Information Theory, 57 (2011), pp. 6958–6975.
- [31] G. H. GOLUB AND C. F. VAN LOAN, *Matrix Computations*, Johns Hopkins University Press, Baltimore, Maryland, 1996.
- [32] I. F. GORODNITSKY, J. GEORGE, AND B. D. RAO, *Neuromagnetic source imaging with FOCUSS: A recursive weighted minimum norm algorithm*, Electroencephalography and Clinical Neurophysiology, 95 (1995), pp. 231–251.
- [33] R. GRIBONVAL AND K. SCHNASS, *Dictionary identification–sparse matrix-factorization via  $l_1$  -minimization*, IEEE Trans. Inform. Theory, 56 (2010), pp. 3523–3539.
- [34] Y. KIM, M. S. NADAR, AND A. BILGIN, *Wavelet-based compressed sensing using gaussian scale mixtures*, in Proc. ISMRM, 2010, p. 4856.
- [35] X. LI AND S. LUO, *A compressed sensing-based iterative algorithm for ct reconstruction and its possible application to phase contrast imaging*, BioMedical Engineering OnLine, 10 (2011), p. 73.
- [36] S. G. LINGALA AND M. JACOB, *Blind compressive sensing dynamic mri*, IEEE Transactions on Medical Imaging, 32 (2013), pp. 1132–1145.
- [37] M. LUSTIG, *Michael Lustig home page*. <http://www.eecs.berkeley.edu/~mlustig/Software.html>, 2014. [Online; accessed October, 2014].
- [38] M. LUSTIG, D.L. DONOHO, AND J.M. PAULY, *Sparse MRI: The application of compressed sensing for rapid MR imaging*, Magnetic Resonance in Medicine, 58 (2007), pp. 1182–1195.
- [39] M. LUSTIG, J. M. SANTOS, D. L. DONOHO, AND J. M. PAULY, *k-t SPARSE: High frame rate dynamic MRI exploiting spatio-temporal sparsity*, in Proc. ISMRM, 2006, p. 2420.
- [40] J. MAIRAL, F. BACH, J. PONCE, AND G. SAPIRO, *Online learning for matrix factorization and sparse coding*, J. Mach. Learn. Res., 11 (2010), pp. 19–60.
- [41] J. MAIRAL, M. ELAD, AND G. SAPIRO, *Sparse representation for color image restoration*, IEEE Trans. on Image Processing, 17 (2008), pp. 53–69.
- [42] J. MAIRAL, G. SAPIRO, AND M. ELAD, *Learning multiscale sparse representations for image and video restoration*, SIAM Multiscale Modeling and Simulation, 7 (2008), pp. 214–241.
- [43] K. MALCZEWSKI, *Pet image reconstruction using compressed sensing*, in Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), 2013, Sept 2013, pp. 176–181.
- [44] S. C. MALIK, *Principles of Real Analysis*, New Age International, New Delhi, India, 1982.
- [45] S. MALLAT, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 1999.
- [46] S. G. MALLAT AND Z. ZHANG, *Matching pursuits with time-frequency dictionaries*, IEEE Transactions on Signal Processing, 41 (1993), pp. 3397–3415.
- [47] M. W. MARCELLIN, M. J. GORMISH, A. BILGIN, AND M. P. BOLIEK, *An overview of JPEG-2000*, in Proc. Data Compression Conf., 2000, pp. 523–541.
- [48] Y. MOHSIN, G. ONGIE, AND M. JACOB, *Iterative shrinkage algorithm for patch-smoothness regularized medical image recovery*, IEEE Transactions on Medical Imaging, (2015).
- [49] B. S. MORDUKHOVICH, *Variational Analysis and Generalized Differentiation. Vol. I: Basic theory*, Springer-Verlag, Heidelberg, Germany, 2006.
- [50] B. K. NATARAJAN, *Sparse approximate solutions to linear systems*, SIAM J. Comput., 24 (1995), pp. 227–234.
- [51] B. NING, X. QU, D. GUO, C. HU, AND Z. CHEN, *Magnetic resonance image reconstruction using*



- trained geometric directions in 2d redundant wavelets domain and non-convex optimization*, Magnetic Resonance Imaging, 31 (2013), pp. 1611–1622.
- [52] B. A. OLSHAUSEN AND D. J. FIELD, *Emergence of simple-cell receptive field properties by learning a sparse code for natural images*, Nature, 381 (1996), pp. 607–609.
  - [53] Y. PATI, R. REZAIIFAR, AND P. KRISHNAPRASAD, *Orthogonal matching pursuit : recursive function approximation with applications to wavelet decomposition*, in Asilomar Conf. on Signals, Systems and Comput., 1993, pp. 40–44 vol.1.
  - [54] L. PFISTER, *Tomographic reconstruction with adaptive sparsifying transforms*, master’s thesis, University of Illinois at Urbana-Champaign, Aug. 2013.
  - [55] L. PFISTER AND Y. BRESLER, *Model-based iterative tomographic reconstruction with adaptive sparsifying transforms*, in SPIE International Symposium on Electronic Imaging: Computational Imaging XII, vol. 9020, 2014, pp. 90200H–1–90200H–11.
  - [56] W. K. PRATT, J. KANE, AND H. C. ANDREWS, *Hadamard transform image coding*, Proc. IEEE, 57 (1969), pp. 58–68.
  - [57] K. P. PRUESSMANN, *Encoding and reconstruction in parallel MRI*, NMR in Biomedicine, 19 (2006), pp. 288–299.
  - [58] C. QIU, W. LU, AND N. VASWANI, *Real-time dynamic MR image reconstruction using kalman filtered compressed sensing*, in Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, 2009, pp. 393–396.
  - [59] X. QU, *PANO Code*. [http://www.quxiaobo.org/project/CS\\_MRI\\_PANO/Demo\\_PANO\\_SparseMRI.zip](http://www.quxiaobo.org/project/CS_MRI_PANO/Demo_PANO_SparseMRI.zip), 2014. [Online; accessed May, 2015].
  - [60] —, *PANO Code with multi-core cpu parallel computing*. [http://www.quxiaobo.org/project/CS\\_MRI\\_PANO/Demo\\_Parallel\\_PANO\\_SparseMRI.zip](http://www.quxiaobo.org/project/CS_MRI_PANO/Demo_Parallel_PANO_SparseMRI.zip), 2014. [Online; accessed April, 2015].
  - [61] —, *PBDWS Code*. [http://www.quxiaobo.org/project/CS\\_MRI\\_PBDWS/Demo\\_PBDWS\\_SparseMRI.zip](http://www.quxiaobo.org/project/CS_MRI_PBDWS/Demo_PBDWS_SparseMRI.zip), 2014. [Online; accessed September, 2014].
  - [62] X. QU, D. GUO, B. NING, Y. HOU, Y. LIN, S. CAI, AND Z. CHEN, *Undersampled MRI reconstruction with patch-based directional wavelets*, Magnetic Resonance Imaging, 30 (2012), pp. 964–977.
  - [63] X. QU, Y. HOU, F. LAM, D. GUO, J. ZHONG, AND Z. CHEN, *Magnetic resonance image reconstruction from undersampled measurements using a patch-based nonlocal operator*, Medical Image Analysis, 18 (2014), pp. 843–856.
  - [64] S. RAVISHANKAR AND Y. BRESLER, *MR image reconstruction from highly undersampled k-space data by dictionary learning*, IEEE Trans. Med. Imag., 30 (2011), pp. 1028–1041.
  - [65] —, *Multiscale dictionary learning for MRI*, in Proc. ISMRM, 2011, p. 2830.
  - [66] —, *Learning doubly sparse transforms for image representation*, in IEEE Int. Conf. Image Process., 2012, pp. 685–688.
  - [67] —, *Closed-form solutions within sparsifying transform learning*, in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2013, pp. 5378–5382.
  - [68] —, *DLMRI - Lab: Dictionary learning MRI software*. <http://www.ifp.illinois.edu/~yoram/DLMRI-Lab/DLMRI.html>, 2013. [Online; accessed October, 2014].
  - [69] —, *Learning doubly sparse transforms for images*, IEEE Trans. Image Process., 22 (2013), pp. 4598–4612.
  - [70] —, *Learning sparsifying transforms*, IEEE Trans. Signal Process., 61 (2013), pp. 1072–1086.
  - [71] —, *Sparsifying transform learning for compressed sensing MRI*, in Proc. IEEE Int. Symp. Biomed. Imag., 2013, pp. 17–20.
  - [72] —, *Blind compressed sensing using sparsifying transforms*, in International Conference on Sampling Theory and Applications (SampTA), May 2015, pp. 513–517.
  - [73] —,  *$\ell_0$  sparsifying transform learning with efficient optimal updates and convergence guarantees*, IEEE Trans. Signal Process., 63 (2015), pp. 2389–2404.
  - [74] R. T. ROCKAFELLAR AND ROGER J.-B. WETS, *Variational Analysis*, Springer-Verlag, Heidelberg, Germany, 1998.
  - [75] K. SKRETTEING AND K. ENGAN, *Recursive least squares dictionary learning algorithm*, IEEE Transactions on Signal Processing, 58 (2010), pp. 2121–2130.



- [76] J. TRZASKO AND A. MANDUCA, *Highly undersampled magnetic resonance image reconstruction via homotopic  $l_0$ -minimization*, IEEE Trans. Med. Imaging, 28 (2009), pp. 106–121.
- [77] P. TSENG, *Convergence of a block coordinate descent method for nondifferentiable minimization*, J. Optim. Theory Appl., 109 (2001), pp. 475–494.
- [78] S. VALIOLLAHZADEH, T. CHANG, J. W. CLARK, AND O. R. MAWLAWI, *Image recovery in pet scanners with partial detector rings using compressive sensing*, in IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC), Oct 2012, pp. 3036–3039.
- [79] R. VENKATARAMANI AND Y. BRESLER, *Further results on spectrum blind sampling of 2D signals*, in Proc. IEEE Int. Conf. Image Proc., ICIP, vol. 2, Oct. 1998, pp. 752–756.
- [80] Y. WANG, Y. ZHOU, AND L. YING, *Undersampled dynamic magnetic resonance imaging using patch-based spatiotemporal dictionaries*, in 2013 IEEE 10th International Symposium on Biomedical Imaging (ISBI), April 2013, pp. 294–297.
- [81] B. WEN, S. RAVISHANKAR, AND Y. BRESLER, *Structured overcomplete sparsifying transform learning with convergence guarantees and applications*, International Journal of Computer Vision, (2014), pp. 1–31.
- [82] Y. XU AND W. YIN, *A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion*, SIAM Journal on Imaging Sciences, 6 (2013), pp. 1758–1789.
- [83] M. YAGHOUBI, T. BLUMENSATH, AND M. DAVIES, *Dictionary learning for sparse approximations with the majorization method*, IEEE Transaction on Signal Processing, 57 (2009), pp. 2178–2191.
- [84] J. C. YE, Y. BRESLER, AND P. MOULIN, *A self-referencing level-set method for image reconstruction from sparse Fourier samples*, Int. J. Computer Vision, 50 (2002), pp. 253–270.
- [85] G. YU, G. SAPIRO, AND S. MALLAT, *Image modeling and enhancement via structured sparse model selection*, in Proc. IEEE International Conference on Image Processing (ICIP), 2010, pp. 1641–1644.