Data-Driven Learning of a Union of Sparsifying Transforms Model for Blind Compressed Sensing

Saiprasad Ravishankar, *Member, IEEE,* Yoram Bresler, *Fellow, IEEE*

***Abstract*—Compressed sensing is a powerful tool in applications such as magnetic resonance imaging (MRI). It enables accurate recovery of images from highly undersampled measurements by exploiting the sparsity of the images or image patches in a transform domain or dictionary. In this work, we focus on blind compressed sensing (BCS), where the underlying sparse signal model is a priori unknown, and propose a framework to simultaneously reconstruct the underlying image as well as the unknown model from highly undersampled measurements. Specifically, our model is that the patches of the underlying image(s) are approximately sparse in a transform domain. We also extend this model to a union of transforms model that better captures the diversity of features in natural images. The proposed block coordinate descent type algorithms for blind compressed sensing are highly efficient, and are guaranteed to converge to at least the partial global and partial local minimizers of the highly non-convex BCS problems. Our numerical experiments show that the proposed framework usually leads to better quality of image reconstructions in MRI compared to several recent image reconstruction methods. Importantly, the learning of a union of sparsifying transforms leads to better image reconstructions than a single adaptive transform.**

***Index Terms*—Sparsifying transforms, Inverse problems, Compressed sensing, Medical imaging, Magnetic resonance imaging, Sparse representations, Dictionary learning, Machine learning**

I. Introduction

The sparsity of signals and images in transform domains or dictionaries is a key property that has been exploited in several applications including compression [2], denoising, and in inverse problems in imaging. Sparsity in either a fixed or data-adaptive dictionary or transform is fundamental to the success of popular techniques such as compressed sensing that aim to reconstruct images from a few sensor measurements. In this work, we focus on methods for blind compressed sensing, where not only the image but also the dictionary or transform is estimated from the measurements. In the following, we briefly review compressed sensing and blind compressed sensing, before summarizing our contributions in this work.

DOI: 10.1109/TCI.2016.2567299.

This work was supported in part by the National Science Foundation (NSF) under grant CCF-1320953. A short version of this work appears elsewhere [1].

S. Ravishankar is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI, 48109 USA email: ravisha@umich.edu. Y. Bresler is with the Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois, Urbana-Champaign, IL, 61801 USA e-mail: ybresler@illinois.edu.

A. Compressed Sensing

Compressed sensing (CS) [3]–[5] (see also [6]–[13] for the earliest versions of CS for Fourier-sparse signals and for Fourier imaging) is a technique that enables accurate reconstructions of images from far fewer measurements than the number of unknowns. To do so, it assumes that the underlying image is sufficiently sparse in some transform domain or dictionary, and that the measurement acquisition procedure is incoherent, in an appropriate sense, with the transform. The image reconstruction problem in CS is often formulated (using a convex relaxation of the ℓ_0 counting "norm" for sparsity) as follows [14]

$$\min_{x} \ \|Ax - y\|_2^2 + \lambda \|\Psi x\|_1 \tag{1}$$

Here, $x \in \mathbb{C}^p$ is a vectorized version of the image to be reconstructed, $\Psi \in \mathbb{C}^{t \times p}$ is a sparsifying transform for the image (often chosen as orthonormal), $y \in \mathbb{C}^m$ denotes the imaging measurements, and $A \in \mathbb{C}^{m \times p}$, with $m \ll p$ is the sensing or measurement matrix for the application.

Compressed sensing has become an increasingly attractive tool for imaging in recent years. CS has been applied to several imaging modalities such as magnetic resonance imaging (MRI) [14]–[20], computed tomography (CT) [21]–[23], and Positron emission tomography (PET) imaging [24], [25], demonstrating high quality reconstructions from few measurements. Such compressive measurements may help reduce the radiation dosage in CT, or reduce the scan times in MRI.

In this work, we will develop methods that apply to compressed sensing and other general inverse problems. We illustrate our methods in the particular application of MRI. MRI is a non-invasive and non-ionizing imaging modality 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 or the spatial Fourier transform of the object, are acquired sequentially in time. Hence, a drawback of MRI that affects both clinical throughput and image quality is that it is a relatively slow imaging technique. Although there have been advances in scanner hardware [26] and pulse sequences, the rate at which MR data are acquired is limited by MR physics and physiological constraints on RF energy deposition. CS accelerates the data acquisition in MRI by collecting fewer k-space measurements than mandated by Nyquist sampling conditions. In particular, for MRI, the sensing matrix A in (1) is $F_u \in \mathbb{C}^{m \times p}$, the undersampled Fourier encoding matrix.

B. Blind Compressed Sensing

While compressed sensing techniques typically work with fixed sparsifying transforms such as Wavelets, finite differences (total variation) [14], [27], Contourlets [28], etc. to reconstruct images, there has been a growing interest in data-driven models in recent years. Some recent works considered learning dictionaries [29] or tight frames [30] from reference images, but in these methods, the model is kept fixed during the CS image reconstruction process, and not adapted to better sparsify and reconstruct the features/dynamics of the underlying (unknown) images. In this work, we instead focus on the subject of blind compressed sensing (BCS) [31]–[43]. In BCS, the sparse model for the underlying image(s) or image patches is assumed unknown a priori. The goal in BCS is then to reconstruct both the image(s) as well as the dictionary or transform from only the undersampled measurements. Thus, the BCS problem is harder than conventional compressed sensing. However, BCS allows the sparse model to be better adaptive to the current (unknown) image(s).

In an early work [44], Fowler proposed a method for recovering the principal eigenvectors of data (principal components) from random projections. This work shares similarities with BCS in its attempt to estimate a model for data from compressive measurements. However, while the prior work [44] learns an under-complete principal components model, BCS can enable the learning of much richer data models by exploiting sparsity criteria.

The sparse model in BCS can take a variety of forms. For example, the well-known synthesis dictionary model suggests that a real-world signal $z \in \mathbb{C}^n$ can be approximately represented as a linear combination of a small number of (or, a sparse set of) atoms or columns from a synthesis dictionary $D \in \mathbb{C}^{n \times m}$, i.e., $z = D\alpha + e$ with $\alpha \in \mathbb{C}^m$ sparse, or $\|\alpha\|_0 \ll n$, and e is the approximation or modeling error in the signal domain [45]. The alternative sparsifying transform model (which is a generalized analysis model [46]) suggests that the signal z is approximately sparsifiable using a transform $W \in \mathbb{C}^{m \times n}$, i.e., $Wz = \alpha + \eta$, where $\alpha \in \mathbb{C}^m$ is sparse in some sense, and η is a small residual error in the *transform domain* rather than in the signal domain. The advantage of the transform model over the synthesis dictionary model is that sparse coding (the process of finding α for a signal z, using a given D or W) can be performed cheaply by thresholding [46], whereas it is NP-hard (Non-deterministic Polynomial-time hard) in the latter case [47], [48]. In recent years, the data-driven adaptation of such sparse signal models has received increasing attention and has been shown to be advantageous in several applications [31], [49]–[54].

In prior work on BCS [31], we proposed synthesis dictionary-based blind compressed sensing for MRI. The overlapping patches of the underlying image were modeled as sparse in an unknown patch-based dictionary (of size much smaller than the image), and this dictionary was learnt jointly with the image from undersampled k-space measurements. BCS techniques can provide much better image reconstructions for MRI compared to conventional CS methods that use only a fixed sparsifying transform or dictionary [31], [32], [34], [35], [39], [40]. However, previous dictionary-based BCS methods, which typically solve non-convex or NP-hard problems by block coordinate descent type approaches, tend to be computationally expensive, and lack any convergence guarantees.

C. Contributions

In this work, we focus on the efficient sparsifying transform model [46], and study a particular transform-based blind compressed sensing framework that has not been explored in prior work [41], [43]. The proposed framework is to simultaneously reconstruct the underlying image(s) and learn the transform model from compressive measurements. First, we model the patches of the underlying image(s) as approximately sparse in a *single* (square) transform domain. We then further extend this model to a *union of transforms model* (also known as OCTOBOS model [54]) that is better suited to capture the diversity of features in natural images. The transforms in our formulations are constrained to be *unitary*. This results in computationally cheap transform update and image update steps in the proposed block coordinate descent type BCS algorithms. We also work with an ℓ_0 penalty (instead of constraint) for sparsity in our formulations, which enables a very efficient and exact sparse coding step involving thresholding in our block coordinate descent algorithms. The ℓ_0 penalty also plays a key role in enabling the generalization of the proposed formulation and algorithm for single transform BCS to the union of transforms case. We present convergence results for our algorithms that solve the single transform or union of transforms BCS problems. In both cases, the algorithms are guaranteed to converge to at least the partial global and partial local minimizers of the highly non-convex BCS problems. Our numerical experiments show that the proposed BCS framework usually leads to better quality of image reconstructions in MRI compared to several recent image reconstruction methods. Importantly, the learning of a union of sparsifying transforms leads to better image reconstructions than when learning a single transform. The data adaptive regularizers proposed in this work can be used in general inverse problem settings, and are not restricted to compressed sensing.

D. Relation to Recent Works

In prior work, we proposed the idea of learning square sparsifying transforms from training signals [46], [53]. A method for learning a union of transforms model from training data has also been proposed [54]. However, these works did not consider the problem of jointly estimating images and image models from compressive measurements (i.e., blind compressed sensing). The latter idea was considered in recent papers [41], [43][1], where methods for simultaneously reconstructing images and learning square sparsifying transforms for image patches were considered. In this work, we instead investigate a novel and efficient framework for blind compressed sensing involving the richer union of transforms

[1]The method in [41] lacks any convergence analysis and also involves many parameters (e.g., error thresholds to determine patch-wise sparsity levels) that may be hard to tune in practice.

model. We use as a building block a specific square transform-based blind compressed sensing formulation involving an ℓ_0 sparsity penalty and unitary transform constraint that is related to formulations ((P2) and (P3)) in prior work [43], but was not explicitly considered therein. We show promise for the proposed methods for MR image reconstruction, where they achieve improved or faster reconstructions compared to our recent TLMRI method [43], which uses a single adaptive transform.

The application of the methods proposed in this work for MRI was briefly considered in a very recent conference publication [1]. However, unlike the conference work, here, we also provide detailed theoretical convergence results for the proposed union of transforms-based blind compressed sensing method. An empirical study of the convergence and (blind) learning behavior of the proposed methods is also presented here, along with expanded experimental results and comparisons. Importantly, the theoretical and empirical convergence results presented in this work are for union of transforms-based blind compressed sensing rather than for (the simpler) transform learning (from training signals) [53], [54]. The theoretical results here generalize results from our prior work [43] to related as well as more complex scenarios.

E. Organization

The rest of this paper is organized as follows. Section II describes our transform learning-based blind compressed sensing formulations and their properties. Section III derives efficient block coordinate descent algorithms for the proposed problems, and discusses the algorithms' computational costs. Section IV discusses the theoretical convergence properties of the proposed algorithms. Section V presents experimental results demonstrating the practical convergence behavior and performance of the proposed schemes for the MRI application. Section VI presents our conclusions and proposals for future work.

II. Blind Compressed Sensing Problem Formulations

The image reconstruction Problem (1) for compressed sensing is a particular instance of the following constrained regularized inverse problem, with $\mathcal{S} = \mathbb{C}^p$

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

The regularizer $\zeta(x) = \lambda \|\Psi x\|_1$ encourages sparsity of the image in a fixed sparsifying transform Ψ. To overcome the limitations of such a non-adaptive CS formulation, or the limitations of the recent dictionary-based BCS methods, we explore sparsifying transform-based BCS formulations in this work. These are discussed in the following subsections.

A. Unitary BCS

Sparsifying transform learning has been demonstrated to be effective and efficient in several applications, while also enjoying good convergence properties [53]–[56]. Here, we propose to use the following transform learning regularizer [53]

$$\zeta(x) = \frac{1}{\nu} \min_{W,B} \sum_{j=1}^{N} \left\{ \|WP_j x - b_j\|_2^2 + \eta^2 \|b_j\|_0 \right\} \quad \text{s.t. } W^H W = I$$

along with the constraint set $\mathcal{S} = \{x \in \mathbb{C}^p : \|x\|_2 \leq C\}$ within Problem (2) to arrive at the following transform BCS formulation

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

Here, $P_j \in \mathbb{C}^{n \times p}$ represents the operator that extracts a patch[2] as a vector $P_j x \in \mathbb{C}^n$ from the image x, and $W \in \mathbb{C}^{n \times n}$ is a square sparsifying transform for the patches of the image. A total of N overlapping image patches are assumed, and $\nu > 0$, $\eta > 0$ are weights in (P1). The term $\|WP_j x - b_j\|_2^2$ in the cost denotes the sparsification error or transform domain residual [46] for the jth image patch, with b_j denoting the transform *sparse code* (i.e., the sparse approximation to the transformed patch). The penalty $\|b_j\|_0$ counts the number of non-zeros in b_j. We use $B \in \mathbb{C}^{n \times N}$ to denote the matrix that has the sparse codes b_j as its columns. The constraint $W^H W = I$, with I denoting the $n \times n$ identity matrix, restricts the set of feasible transforms to unitary matrices. The constraint $\|x\|_2 \leq C$ with $C > 0$ in (P1) enforces any prior knowledge on the signal energy (or, range).

In the absence of the $\|x\|_2 \leq C$ condition, the objective in (P1) is non-coercive. In particular, consider $W = I$ (a unitary matrix) and $x_\alpha = x_0 + \alpha z$, where x_0 is a solution to $y = Ax$, $\alpha \in \mathbb{R}$, and $z \in \mathcal{N}(A)$ with $\mathcal{N}(A)$ denoting the null space of A. Then, as $\alpha \to \infty$ with b_j set to $WP_j x_\alpha$, the objective in (P1) remains always finite (non-coercive). The constraint $\|x\|_2 \leq C$ alleviates possible problems (e.g., unbounded iterates in algorithms) due to such a non-coercive objective. It can also be alternatively replaced with constraints such as box constraints depending on the application and underlying image properties.

While a single weight η^2 is used for the sparsity penalties $\|b_j\|_0 \ \forall \ j$ in (P1), one could also use different weights η_j^2 for the penalties corresponding to different patches, if such weights are known, or estimated. When measurements from multiple images (or frames, or slices) are available, then by considering the summation of the corresponding objective functions for each image, Problem (P1) can be easily extended to enable joint reconstruction of the images using a single adaptive (spatial) transform. For applications such as dynamic MRI, one can also work with adaptive spatiotemporal sparsifying transforms of 3D patches in (P1).

We have studied some transform BCS methods in very recent works [41], [43]. However, the formulation (P1) investigated here was not explored in the prior work. Exploiting

[2]For 2D imaging, this would be a $d \times d$ patch, with $n = d^2$ pixels. For 3D or 4D imaging, the corresponding 3D or 4D patches would have sizes $d \times d \times d$ or $d \times d \times d \times d$, with $n = d^3$ or $n = d^4$, respectively.

both a unitary transform constraint (as opposed to a penalty that enables well-conditioning [43]) and a sparsity penalty (as opposed to a sparsity constraint [43]) in the transform BCS formulation leads to a very efficient block coordinate descent algorithm in this work. Moreover, Problem (P1) and the algorithm proposed to solve it can be readily extended to accomodate richer models as shown in the following discussions.

B. Union of Transforms BCS

Here, we extend the single transform model in Problem (P1) to a union of transforms model (similar to [54]). In this model, we consider a collection (union) of square transforms $\{W_k\}_{k=1}^K$ with $W_k \in \mathbb{C}^{n \times n} \, \forall \, k$, and each image patch is assumed to have a corresponding "best matching transform" (i.e., a transform that best sparsifies the particular patch) in this collection. A motivation for the proposed model is that natural images or image patches need not be sufficiently sparsifiable by a single transform. For example, image patches from different regions of an image usually contain different types of features, or textures. Thus, having a union of transforms would allow groups of patches with common features (or, textures) to be better sparsified by their own specific transform.

Such a union of square transforms can be interpreted as an overcomplete transform, also called OverComplete TransfOrm model with BlOck coSparsity constraint, or OCTOBOS. The equivalent overcomplete transform is obtained by stacking the square 'sub-transforms' as $W = \begin{bmatrix} W_1^T \mid W_2^T \mid \ \ldots \ \mid W_K^T \end{bmatrix}^T$. The matrix $W \in \mathbb{R}^{m \times n}$, with $m = Kn$, and thus, $m > n$ (overcomplete transform) for $K > 1$. Proposition 1 of [54] proves the equivalence between the following two (sparse coding) problems, where the first one involves the union of transforms, and the second one is based directly on an overcomplete (OCTOBOS) one.

$$\min_{1 \leq k \leq K} \min_{\alpha^k} \left\| W_k z - \alpha^k \right\|_2^2 \ \text{ s.t. } \ \left\| \alpha^k \right\|_0 \leq s \, \forall \, k \tag{3}$$

$$\min_{\alpha} \left\| W z - \alpha \right\|_2^2 \ \text{ s.t. } \ \left\| \alpha \right\|_{0,s} \geq 1 \tag{4}$$

Here, $z \in \mathbb{C}^n$ is a given signal, and $\alpha \in \mathbb{C}^m$ in (4) is obtained by stacking K blocks $\alpha^k \in \mathbb{C}^n$, $1 \leq k \leq K$. The operation $\left\| \alpha \right\|_{0,s} \triangleq \sum_{k=1}^K I(\left\| \alpha^k \right\|_0 \leq s)$ with $I(\cdot)$ denoting the indicator function, counts the number of blocks of α with at least $n - s$ zeros (co-sparse blocks), where s is a parameter. Proposition 1 of [54] showed that the minimum sparsification errors (objectives) in (3) and (4) are identical and that the sparse minimizer(s) in (3) (i.e., best/minimizing sparse code(s) over $1 \leq k \leq K$) are simply the block(s) with at least $n - s$ zeros of the minimizer(s) in (4).

We have investigated the learning of a union of transforms, or OCTOBOS learning, from training data in a recent work [54]. Here, we propose to use the following union of transforms learning regularizer

$$\zeta(x) = \frac{1}{\nu} \min_{\{W_k, b_j, C_k\}} \sum_{k=1}^K \sum_{j \in C_k} \left\{ \left\| W_k P_j x - b_j \right\|_2^2 + \eta^2 \left\| b_j \right\|_0 \right\}$$
$$\text{s.t. } \ W_k^H W_k = I \ \forall \, k, \ \{C_k\} \in G$$

along with the constraint set $\mathcal{S} = \{x \in \mathbb{C}^p : \left\| x \right\|_2 \leq C\}$ within Problem (2) to arrive at the following union of transforms BCS formulation:

$$\text{(P2)} \min_{x, B, \{W_k, C_k\}} \sum_{k=1}^K \sum_{j \in C_k} \left\{ \left\| W_k P_j x - b_j \right\|_2^2 + \eta^2 \left\| b_j \right\|_0 \right\}$$
$$+ \nu \left\| A x - y \right\|_2^2$$
$$\text{s.t. } \ W_k^H W_k = I \ \forall \, k, \ \{C_k\} \in G, \ \ \left\| x \right\|_2 \leq C.$$

Here and 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 the indices. The set $\{C_k\}_{k=1}^K$ in (P2) indicates a clustering of the image patches $\{P_j x\}_{j=1}^N$ into K disjoint sets. The cluster C_k contains the indices j corresponding to the patches $P_j x$ in the kth cluster. The patches in the kth cluster are considered (best) matched to the transform W_k. The set G in (P2) is the set of all possible partitions of the set of integers $[1 : N] \triangleq \{1, 2, ..., N\}$ into K disjoint subsets, i.e.,

$$G = \left\{ \{C_k\} : \bigcup_{k=1}^K C_k = [1 : N], \, C_j \bigcap C_k = \emptyset, \, \forall \, j \neq k \right\}$$

The term $\sum_{k=1}^K \sum_{j \in C_k} \left\| W_k P_j x - b_j \right\|_2^2$ in (P2) is the sparsification error of the patches of x in the (richer) union of transforms model. Problem (P2) is to jointly reconstruct the image x and learn the (unknown) union of transforms for the image patches, as well as cluster the patches, using only the compressive imaging measurements. The optimal objective function value (i.e., the minimum value) in Problem (P2) can only be lower than the corresponding optimal value in (P1). This is obvious because the single transform model in (P1) is a subset of the richer (or more general) union of transforms model in (P2).

The recent PANO method [57] for MR image reconstruction also involves a patch grouping methodology, but differs from the method proposed here in several important aspects: (i) the patch grouping criterion; (ii) the type and dimension of sparsifying transform; and (iii) the use of a reference reconstruction vs. joint clustering and reconstruction. In particular, in the PANO method, the patches of a reference reconstruction are grouped together according to their similarity measured in terms of the Euclidean ℓ_2 distance. A penalty based on the sparsity of such groups of similar (2D) patches in a *fixed* 3D transform (Haar wavelet) domain is used as a regularizer in the CS image reconstruction problem. Unlike the PANO method, Problem (P2) clusters together patches that are best sparsified by a common *adaptive* transform, i.e., the clustering measure is based on the sparsification error. Thus the clustered patches need not be similar in Euclidean distance and the adapted clusterings in (P2) can be quite general. Furthermore, in (P2), because the transform is adapted to the patches in the cluster, the transform depends on the clustering, and the clustering depends on the transform. Another difference is that, unlike the 3D (fixed) transform in PANO, for 2D patches, the adapted transform here is a 2D transform sparsifying each patch individually. Finally, unlike PANO, (P2) jointly clusters

patches and reconstructs x, and is not based on reference reconstructions.

III. Algorithms and Properties

A. Algorithms

Problems (P1) and (P2) involve highly nonconvex and non-differentiable (in fact, discontinuous) objectives, as well as nonconvex constraints. Because of the lack of analytical solutions, iterative approaches are commonly adopted for problems of this kind. Here, we adopt iterative block coordinate descent algorithms for (P1) and (P2) that lead to highly efficient solutions for the corresponding subproblems. Another advantage of block coordinate descent is that it does not require the choice of additional parameters such as step sizes. We first describe our algorithm for (P2). The algorithm for (P1) is just a special case (with $K = 1$) of the one for (P2).

In one step of our proposed block coordinate descent algorithm for (P2) called the *sparse coding and clustering step*, we solve for $\{C_k\}$ and B in (P2) with the other variables fixed. In another step called the *transform update step*, we solve for the transforms $\{W_k\}$ in (P2), while keeping all other variables fixed. In the third step called the *image update step*, we update only the image x, with the other variables fixed. We now describe these steps in detail.

1) Sparse Coding and Clustering Step: In this step, we solve the following optimization problem:

$$\text{(P3)} \quad \min_{\{C_k\},\{b_j\}} \sum_{k=1}^{K} \sum_{j \in C_k} \left\{ \|W_k P_j x - b_j\|_2^2 + \eta^2 \|b_j\|_0 \right\} \quad \text{s.t.} \quad \{C_k\} \in G.$$

By first performing the (inner) optimization with respect to the b_j's in (P3), it is easy to observe that Problem (P3) can be rewritten in the following equivalent form:

$$\sum_{j=1}^{N} \min_{1 \leq k \leq K} \left\{ \|W_k P_j x - H_\eta(W_k P_j x)\|_2^2 + \eta^2 \|H_\eta(W_k P_j x)\|_0 \right\} \tag{5}$$

where the minimization over k for each patch $P_j x$ ($1 \leq j \leq N$) determines the cluster C_k in (P3) to which that patch belongs. The hard-thresholding operator $H_\eta(\cdot)$ appears in (5) because of the aforementioned (inner) optimization with respect to the b_j's [53] in (P3), and $H_\eta(\cdot)$ is defined as follows, where $\alpha \in \mathbb{C}^n$ is any vector, and the subscript i indexes vector entries.

$$\left(H_\eta(\alpha)\right)_i = \begin{cases} 0 & , \ |\alpha_i| < \eta \\ \alpha_i & , \ |\alpha_i| \geq \eta \end{cases} \tag{6}$$

For each patch $P_j x$, the optimal cluster index $\hat{k}_j$ in (5) is then

$$\hat{k}_j = \arg\min_k \ \|W_k P_j x - H_\eta(W_k P_j x)\|_2^2 + \eta^2 \|H_\eta(W_k P_j x)\|_0 \tag{7}$$

The optimal sparse code $\hat{b}_j$ in (P3) is then $H_\eta(W_{\hat{k}_j} P_j x)$ [53]. There is no coupling between the sparse coding/clustering problems in (5) for the different image patches $\{P_j x\}_{j=1}^N$. Thus, they are clustered and sparse coded in parallel.

The optimal cluster membership or the optimal sparse code for any particular patch $P_j x$ in (P3) need not be unique. When there are multiple optimal cluster indices in (7), we pick the lowest such index. The optimal sparse code for the patch $P_j x$ is not unique when the condition $\left| \left(W_{\hat{k}_j} P_j x \right)_i \right| = \eta$ is satisfied for some i (cf. [53] for a similar scenario and an explanation). The definition in (6) chooses *one* of the multiple optimal solutions (corresponding to the transform $W_{\hat{k}_j}$) in this case.

Note that if instead of employing a sparsity penalty (i.e., penalizing $\sum_{j=1}^N \|b_j\|_0$), we were to constrain the term $\sum_{j=1}^N \|b_j\|_0$ (i.e., force it to have an upper bound of s [43]) in (P2), then the sparse coding and clustering step of such a modified BCS problem shown below suffers from the drawback that the sparsity constraint creates inter-patch coupling, which in turn leads to exponential scaling of the computation with the number of patches.

$$\min_{\{C_k\},B} \sum_{k=1}^{K} \sum_{j \in C_k} \|W_k P_j x - b_j\|_2^2 \quad \text{s.t.} \quad \sum_{j=1}^{N} \|b_j\|_0 \leq s, \ \{C_k\} \in G. \tag{8}$$

For a fixed clustering $\{C_k\}$, the optimal B above is readily obtained by zeroing out all but the s largest magnitude elements of the matrix $\left[W_{k_1} P_1 x \mid W_{k_2} P_2 x \mid ... \mid W_{k_N} P_N x\right]$, where k_j denotes the cluster index of patch $P_j x$. However, this requires examination of all the sparsified patches *jointly*. Now, consider the objective value attained in (8) for the clustering $\{C_k\}$ and its corresponding optimal B, to which we refer as the sparsification error for that clustering. The exact solution to Problem (8) requires computing this sparsification error for each possible clustering, and then picking the clustering that achieves the minimum error. Because there are K^N possible clusterings, the cost of computing the solution scales exponentially with the number of patches as $O(Nn^2K^N)$. Thus, Problem (8) is computationally intractable. [3] This is one of the reasons for pursuing formulations with sparsity penalties (rather than constraint) in this work. Furthermore, employing a sparsity penalty leads to a simpler sparse coding solution (with a given clustering) involving hard-thresholding, whereas using a sparsity constraint (as in (8)) for sparse coding necessitates projections onto the s-ℓ_0 ball [43] using a computationally more expensive sorting procedure.

2) Transform Update Step: In this step, we solve (P2) with respect to the cluster transforms $\{W_k\}$, with all other variables fixed. This results in the following optimization problem:

$$\min_{\{W_k\}} \sum_{k=1}^{K} \sum_{j \in C_k} \left\{ \|W_k P_j x - b_j\|_2^2 + \eta^2 \|b_j\|_0 \right\} \quad \text{s.t.} \quad W_k^H W_k = I \ \forall k. \tag{9}$$

[3] One way to modify Problem (8) is to set a constraint of the form $\|b_j\|_0 \leq s$ for each patch. In this case, the sparse coding and clustering problem has a cheap solution [54]. However, different regions of natural images typically carry different amounts of information, and therefore, a fixed sparsity level for each patch often does not work well in practice. In contrast, both Problems (8) and (P3) encourage variable sparsity levels for individual patches.

The above problem is in fact separable (since the objective is in summation form) into K independent constrained optimization problems, each involving a particular square transform W_k. The kth ($1 \leq k \leq K$) such optimization problem is as follows:

$$\text{(P4)} \quad \min_{W_k} \sum_{j \in C_k} \|W_k P_j x - b_j\|_2^2 \ \text{ s.t. } \ W_k^H W_k = I.$$

Denoting by X_{C_k}, the matrix that has the patches $P_j x$ for $j \in C_k$, as its columns, and denoting by B_{C_k}, the matrix whose columns are the corresponding sparse codes, Problem (P4) can be written in compact form as

$$\min_{W_k} \|W_k X_{C_k} - B_{C_k}\|_F^2 \ \text{ s.t. } \ W_k^H W_k = I. \tag{10}$$

Now, let $X_{C_k} B_{C_k}^H$ have a full singular value decomposition (SVD) of $U \Sigma V^H$. Then, a global minimizer [53], [58] in (10) is $\hat{W}_k = V U^H$. This solution is unique if and only if $X_{C_k} B_{C_k}^H$ is non-singular. To solve Problem (9), Problem (P4) is solved for each k, which can be done in parallel.

3) Image Update Step: In this step, we solve (P2) with respect to the unknown image x, keeping the other variables fixed. The corresponding optimization problem is as follows.

$$\text{(P5)} \quad \min_{x} \ \nu \|Ax - y\|_2^2 + \sum_{k=1}^{K} \sum_{j \in C_k} \|W_k P_j x - b_j\|_2^2 \quad \text{s.t.} \ \ \|x\|_2 \leq C.$$

Problem (P5) is a least squares problem with an ℓ_2 (or, alternatively squared ℓ_2) constraint [59]. It can be solved for example using the projected gradient method, or using the Lagrange multiplier method [59]. In the latter case, the corresponding Lagrangian formulation is simply a least squares problem. Therefore, the solution to (P5) satisfies the following Normal Equation

$$\left(\sum_{j=1}^{N} P_j^T P_j \ + \ \nu \, A^H A + \hat{\mu} I \right) x = \sum_{k=1}^{K} \sum_{j \in C_k} P_j^T W_k^H b_j \ + \ \nu \, A^H y \tag{11}$$

where $\hat{\mu} \geq 0$ is the optimally chosen Lagrange multiplier. The optimal $\hat{\mu}$ is the smallest non-negative real for which the solution[4] (i.e., the x) in (11) satisfies the norm constraint in (P5). Problem (P5) can be solved by solving the Lagrangian least squares problem (or, corresponding normal equation) repeatedly (by CG) for various multiplier values (tuned in steps) until the $\|x\|_2 \leq C$ condition is satisfied.

We now discuss the solution to (P5) for the specific case of single-coil MRI. (In the case of multi-coil or parallel MRI, when A is for example a SENSE type sensing matrix [40], the aforementioned iterative strategies can be used to solve (P5).) Recall that $A = F_u$ for (single-coil) MRI, and we assume that the k-space measurements are obtained by subsampling on a uniform Cartesian grid. Assuming that periodically positioned, overlapping image patches (patch *overlap stride* [31] denoted by r) are used in our formulations, and that the patches that overlap the image boundaries 'wrap around' on the opposite side of the image [31], we have that the matrix $\sum_{j=1}^{N} P_j^T P_j = \beta I$, with $\beta = \frac{n}{r^2}$. Then, equation (11) simplifies for MRI as

$$\left(\beta I + \nu \, F F_u^H F_u F^H + \hat{\mu} I\right) F x = F \sum_{k=1}^{K} \sum_{j \in C_k} P_j^T W_k^H b_j + \nu F F_u^H y \tag{12}$$

where $F \in \mathbb{C}^{p \times p}$ denotes the full Fourier encoding matrix assumed normalized ($F^H F = I$), and $F F_u^H F_u F^H$ is a diagonal matrix of ones and zeros, with the ones at those entries that correspond to sampled locations in k-space.

Denote $S \triangleq F \sum_{k=1}^{K} \sum_{j \in C_k} P_j^T W_k^H b_j$ and $S_0 \triangleq F F_u^H y$. S_0 represents the undersampled k-space measurements expanded to full (matrix) size, by inserting zeros at non-sampled locations. The solution to (11) for single-coil MRI, in Fourier space, is then

$$F x_{\hat{\mu}} \, (k_x, k_y) = \begin{cases} \frac{S(k_x,k_y)}{\beta + \hat{\mu}} & , \ (k_x, k_y) \notin \Omega \\ \frac{S(k_x,k_y) + \nu \, S_0(k_x,k_y)}{\beta + \nu + \hat{\mu}} & , \ (k_x, k_y) \in \Omega \end{cases} \tag{13}$$

where (k_x, k_y) indexes k-space locations, and Ω is the subset of k-space that is sampled. Note that the optimal Lagrange multiplier $\hat{\mu}$ is the smallest non-negative real such that

$$f(\hat{\mu}) \triangleq \|x_{\hat{\mu}}\|_2^2 = \sum_{(k_x,k_y) \notin \Omega} \frac{|S(k_x,k_y)|^2}{(\beta + \hat{\mu})^2} + \sum_{(k_x,k_y) \in \Omega} \frac{|S(k_x,k_y) + \nu \, S_0(k_x,k_y)|^2}{(\beta + \nu + \hat{\mu})^2} \leq C^2 \tag{14}$$

We check if the above condition is satisfied for $\hat{\mu} = 0$ first. If not, then we apply Newton's method to find the optimal $\hat{\mu}$ in $f(\hat{\mu}) = C^2$. The optimal $\hat{x}$ in (P5) for MRI is the 2D inverse FFT of the optimal $F x_{\hat{\mu}}$ in (13).

The unitary property of the transforms W_k leads to efficient solutions in the image update step for MRI. In particular, if the W_k's were not unitary, the matrix $\sum_{j=1}^{N} P_j^T P_j$ in (11) and later equations would be replaced with $\sum_{k=1}^{K} \sum_{j \in C_k} P_j^T W_k^H W_k P_j$. The latter matrix is neither diagonal nor readily diagonalizable. Hence, we cannot exploit the simple closed-form solution in (13) in this case and would have to employ slower iterative solution techniques for MRI.

The overall algorithm corresponding to the BCS Problem (P2) is shown in Fig. 1. The algorithm begins with an initial estimate $x^0, \left\{W_k^0, C_k^0\right\}, B^0$ (e.g., $x^0 = A^\dagger y$ (assuming $\left\|A^\dagger y\right\|_2 \leq C$), a random or k-means clustering initialization $\left\{C_k^0\right\}$, $W_k^0 = \text{2D DCT} \ \forall \ k$, and B^0 set to be the minimizer of (P2) for these $x^0, \left\{W_k^0, C_k^0\right\}$). Each outer iteration of the algorithm involves the sparse coding and clustering, transform update, and image update steps. (In general, one could alternate between some of these steps more frequently than between others.) Our algorithm for solving Problem (P1) is similar to that for Problem (P2), except that we work with a single cluster ($K = 1$) in the former case. In particular, the sparse coding and clustering step for (P2) is replaced by just a sparse coding step (in a single unitary transform) for (P1).

[4]The solution in (11) (for any $\hat{\mu} \geq 0$) is unique if the set of patches in our formulation covers all pixels in the image. This is because $\sum_{j=1}^{N} P_j^T P_j$ is a positive definite diagonal matrix in this case.

Union of Transforms-Based BCS Algorithm A2 for (P2) for MRI

Inputs: y - CS measurements, η - weight, ν - weight, C - bound on $\|x\|_2$, J - number of iterations.

Outputs: x - reconstructed image, $\{W_k\}$ - adapted union of transforms, $\{C_k\}$ - learnt clustering of patches, B - matrix with sparse codes of patches as columns.

Initial Estimates: $x^0, \left\{W_k^0, C_k^0\right\}, B^0$.

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

1) **Transform Update Step: For** $k = 1 : K$ **do**
   a) Form the matrices Q_k and R_k with $P_j x^{t-1}$ and b_j^{t-1}, for $j \in C_k^{t-1}$, as their columns, respectively.
   b) Set $U\Sigma V^H$ as the full SVD of $Q_k R_k^H$. $W_k^t = VU^H$.
2) **Sparse Coding and Clustering Step: For** $j = 1 : N$ **do**
   a) If $j = 1$, set $C_k^t = \emptyset \ \forall \ k$.
   b) Compute $\gamma_k = \left\|W_k^t P_j x^{t-1} - H_\eta\left(W_k^t P_j x^{t-1}\right)\right\|_2^2 + \eta^2 \left\|H_\eta\left(W_k^t P_j x^{t-1}\right)\right\|_0$, $1 \leq k \leq K$. Set $\hat{k} = \min\{k : \gamma_k = \min_k \gamma_k\}$. Set $C_{\hat{k}}^t \leftarrow C_{\hat{k}}^t \cup \{j\}$.
   c) $b_j^t = H_\eta\left(W_{\hat{k}}^t P_j x^{t-1}\right)$.
3) **Image Update Step:**
   a) Compute the image $c = \sum_{k=1}^K \sum_{j \in C_k^t} P_j^T \left(W_k^t\right)^H b_j^t$. $S \leftarrow \mathrm{FFT}(c)$.
   b) Compute $f(0)$ as per (14). If $f(0) \leq C^2$, set $\hat{\mu} = 0$. Else, use Newton's method to solve the equation $f(\hat{\mu}) = C^2$ for $\hat{\mu}$.
   c) Update S to be the right hand side of (13). $x^t = \mathrm{IFFT}(S)$.

End

Fig. 1. Algorithm for (P2). The superscript t denotes the iterates in the algorithm. The encoding matrix F in (single-coil) MRI is assumed normalized and the abbreviations FFT and IFFT denote the fast implementations of the normalized 2D DFT and 2D IDFT, respectively. The algorithm for (P1) is identical to the one above except that there is no clustering involved in the sparse coding and clustering step.

B. Computational Costs

Here, we briefly analyze the computational costs of our algorithms for Problems (P1) and (P2), called Algorithm A1 and A2, respectively.

For a fixed number of clusters (constant K), the computational cost per iteration of Algorithm A2 for (P2) for MRI scales as $O(n^2 N)$. Thus, the cost scales quadratically with the parameter n (number of pixels in a patch) and linearly with N (number of patches). The cost per iteration of Algorithm A1 for (P1) scales similarly with respect to these parameters. These costs are dominated by the computations for matrix-matrix or matrix-vector products in our algorithms. In contrast, overcomplete dictionary-based BCS methods such as DLMRI [31] that learn a dictionary $D \in \mathbb{C}^{n \times m}$ ($m \geq n$) from compressive measurements have a cost per outer iteration that scales as $O(nmsN\hat{J})$ [31], [43], where s is the synthesis sparsity level per patch, and $\hat{J}$ is the number of inner dictionary learning (K-SVD [60]) iterations in DLMRI. The DLMRI cost is dominated by synthesis sparse coding (an NP-hard problem). Assuming $m \propto n$ and $s \propto n$, the cost per iteration of DLMRI scales as $O(n^3 N\hat{J})$. Thus, the per-iteration cost of Algorithm A1 or A2 scales much better with patch size than that for prior synthesis dictionary-based BCS methods. This would be particularly advantageous in the context of higher-dimensional imaging applications such as 3D or 4D imaging, where the corresponding 3D or 4D patches are much bigger than the patches in 2D imaging. As illustrated in Section V, the proposed algorithms tend to converge quickly in practice. Therefore, the per-iteration computational advantages usually translate to a net computational advantage in practice.

Clearly the union of transforms based Algorithm A2 involves more computations/operations than the single transform based Algorithm A1. As the number of clusters K varies, the computational cost per iteration of Algorithm A2 for MRI scales as $O(Kn^2 N)$ (i.e., the cost scales linearly with the number of clusters). In particular, these computations (with respect to parameter K) are dominated by the clustering step, where the product between each W_k ($1 \leq k \leq K$) and every patch needs to be computed (to determine the optimal matching transforms or clusters). The computations in Algorithm A2 can be reduced by performing the clustering step less often (than the sparse coding, transform update, and image update steps) in the block coordinate descent algorithm.

IV. Convergence Properties

Since (P1) and (P2) are highly non-convex, standard results on convegence of block coordinate descent methods [61] do not apply. In fact, in certain scenarios, one can easily construct non-convergent iterate sequences for Algorithm A1 or A2 (cf. Section 4 of [43] for examples of such scenarios for related algorithms). Here, we present convergence results for Algorithms A1 and A2 assuming that the various steps (such as SVD computations) are performed exactly. Each outer iteration of our algorithms involves a transform update step, a sparse coding and clustering step (only sparse coding in the case of (P1)), and an image update step.

A. Notations

Problem (P1) is a constrained and non-convex minimization problem. By replacing each constraint with an equivalent barrier penalty (a function that takes the value $+\infty$ when the constraint is violated, and is zero otherwise), Problem (P1) can be written in an unconstrained form involving the following

objective function:

$$g(W,B,x) = \nu \left\|Ax-y\right\|_2^2 + \varphi(W) + \chi(x) + \sum_{j=1}^{N} \left\{ \left\|WP_jx - b_j\right\|_2^2 + \eta^2 \left\|b_j\right\|_0 \right\} \quad (15)$$

where $\varphi(W)$ and $\chi(x)$ are the barrier penalties corresponding to the unitary transform constraint and energy constraint on x, respectively. Problem (P2) can also be written in the following unconstrained form:

$$h(W,B,\Gamma,x) = \nu \left\|Ax-y\right\|_2^2 + \chi(x) + \sum_{k=1}^{K} \varphi(W_k) + \sum_{k=1}^{K} \sum_{j \in C_k} \left\{ \left\|W_kP_jx - b_j\right\|_2^2 + \eta^2 \left\|b_j\right\|_0 \right\} \quad (16)$$

where $\varphi(W_k)$ is the barrier penalty corresponding to the unitary constraint on W_k, $W \in \mathbb{C}^{Kn \times n}$ is obtained by stacking the W_k's on top of one another (or, equivalent OCTOBOS), and the row vector $\Gamma \in \mathbb{R}^{1 \times N}$ is such that its jth element $\Gamma_j \in \{1, .., K\}$ denotes the cluster index (label) corresponding to the patch P_jx. As discussed previously, the clusters $\{C_k\}$ partition $[1:N]$. Here, we refer to patch cluster memberships using the row vector variable Γ rather than using the C_k's.

We denote the iterates (outputs) in each iteration t of Algorithm A1 by the set (W^t, B^t, x^t). For Algorithm A2, the iterates are denoted by the set $(W^t, B^t, \Gamma^t, x^t)$, where W^t in this case denotes the matrix obtained by stacking the cluster-specific transforms W_k^t $(1 \leq k \leq K)$, and Γ^t is a row vector containing the patch cluster indices Γ_j^t $(1 \leq j \leq N)$ as its elements.

B. Main Results

For Algorithm A1 proposed for Problem (P1), the convergence results take the same form as those for similar schemes presented in a very recent work [43]. The result for (P1) is summarized in the following Theorem and corollaries, where for a matrix H, $\left\|H\right\|_\infty \triangleq \max_{i,j} |H_{ij}|$, and by 'globally convergent', we mean convergence from any initialization.

Theorem 1: For an initial (W^0, B^0, x^0), the objective sequence $\{g^t\}$ in Algorithm A1 with $g^t \triangleq g\left(W^t, B^t, x^t\right)$ is monotone decreasing, and converges to a finite value, say $g^* = g^*(W^0, B^0, x^0)$. Moreover, the bounded iterate sequence $\{W^t, B^t, x^t\}$ is such that all its accumulation points are equivalent and achieve the same value g^* of the objective. The sequence $\{a^t\}$ with $a^t \triangleq \left\|x^t - x^{t-1}\right\|_2$, converges to zero. Every accumulation point (W, B, x) of $\{W^t, B^t, x^t\}$ is a critical point [43], [62] of the objective g satisfying the following partial global optimality conditions

$$x \in \arg\min_{\tilde{x}} \; g\left(W, B, \tilde{x}\right) \quad (17)$$

$$W \in \arg\min_{\tilde{W}} \; g\left(\tilde{W}, B, x\right) \quad (18)$$

$$B \in \arg\min_{\tilde{B}} \; g\left(W, \tilde{B}, x\right) \quad (19)$$

Each (W, B, x) also satisfies the following partial local optimality condition that holds for all $\Delta x \in \mathbb{C}^p$, and all $\Delta B \in \mathbb{C}^{n \times N}$ satisfying $\left\|\Delta B\right\|_\infty < \eta/2$:

$$g(W, B + \Delta B, x + \Delta x) \geq g(W, B, x) = g^* \quad (20)$$

Corollary 1: 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 (17), (18), (19), and (20).

Corollary 2: Algorithm A1 is globally convergent to a subset of the set of critical points of the non-convex objective $g\left(W, B, x\right)$. 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, and partial local minimizers of $g(W, B, x)$ with respect to (B, x).

Theorem 1 establishes that for each initial (W^0, B^0, x^0), the iterate sequence in Algorithm A1 converges to an equivalence class of accumulation points (corresponding to the same objective value $g^* = g^*(W^0, B^0, x^0)$ – that could vary with initialization). The equivalent accumulation points are all critical points (generalized stationary points [62]) and at least partial minimizers of the objective g.

In the case of the Algorithm A2 proposed for (P2), because the cluster memberships are discrete rather than continuous variables, we do not have a critical points [43], [62] property as for Algorithm A1. Instead, we establish the following convergence results for Algorithm A2.

Theorem 2: For an initial $(W^0, B^0, \Gamma^0, x^0)$, the objective sequence $\{h^t\}$ in Algorithm A2 with $h^t \triangleq h\left(W^t, B^t, \Gamma^t, x^t\right)$ is monotone decreasing, and converges to a finite value, say $h^* = h^*(W^0, B^0, \Gamma^0, x^0)$. Moreover, the iterate sequence $\{W^t, B^t, \Gamma^t, x^t\}$ is bounded, and all its accumulation points are equivalent and achieve the same value h^* of the objective. The sequence $\{a^t\}$ with $a^t \triangleq \left\|x^t - x^{t-1}\right\|_2$, converges to zero. Every accumulation point (W, B, Γ, x) of $\{W^t, B^t, \Gamma^t, x^t\}$ satisfies the following partial global optimality conditions

$$x \in \arg\min_{\tilde{x}} \; h\left(W, B, \Gamma, \tilde{x}\right) \quad (21)$$

$$W \in \arg\min_{\tilde{W}} \; h\left(\tilde{W}, B, \Gamma, x\right) \quad (22)$$

$$(B, \Gamma) \in \arg\min_{\tilde{B}, \tilde{\Gamma}} \; h\left(W, \tilde{B}, \tilde{\Gamma}, x\right) \quad (23)$$

Each (W, B, Γ, x) also satisfies the following partial local optimality condition that holds for all $\Delta x \in \mathbb{C}^p$, and all $\Delta B \in \mathbb{C}^{n \times N}$ satisfying $\left\|\Delta B\right\|_\infty < \eta/2$:

$$h(W, B + \Delta B, \Gamma, x + \Delta x) \geq h(W, B, \Gamma, x) = h^* \quad (24)$$

Theorem 2 establishes that for each initial $(W^0, B^0, \Gamma^0, x^0)$, the iterate sequence in Algorithm A2 converges to an equivalence class of accumulation points. The equivalent accumulation points are at least partial minimizers of the objective h. In light of Theorem 2, results similar to Corollaries 1 and 2 apply for Algorithm A2, as follows.

Corollary 3: For each $(W^0, B^0, \Gamma^0, x^0)$, the iterate sequence in Algorithm A2 converges to an equivalence class of

accumulation points that are also partial minimizers satisfying (21), (22), (23), and (24).

Corollary 4: The iterate sequence in Algorithm A2 is globally convergent to the set of partial minimizers of the non-convex objective $h(W, B, \Gamma, x)$. The set includes all points (W, B, Γ, x), that are at least partial global minimizers of $h(W, B, \Gamma, x)$ with respect to each of W, (B, Γ), x, and partial local minimizers of $h(W, B, \Gamma, x)$ with respect to (B, x).

Notice that for both Algorithms A1 and A2, $\left\|x^t - x^{t-1}\right\|_2 \to 0$. This is a necessary but not sufficient condition for convergence of the entire sequence $\{x^t\}$. In general, the set of points to which Algorithm A2 or A1 converges may be larger than the set of global minimizers in Problem (P2) or (P1). We leave the investigation of the conditions under which the proposed algorithms converge to the set of global minimizers of the proposed problems to future work.

A brief proof of Theorem 2 is included in the supplementary material available online [63]. The proof draws on a few results from recent works [43], [53], but is for the more complex union of transforms-based blind compressed sensing scenario. Since Algorithm A1 for (P1) is simply a special case (with $K = 1$) of Algorithm A2 for (P2), we do not provide a separate proof for Theorem 1.

V. NUMERICAL EXPERIMENTS

A. Framework

We study the convergence behavior and effectiveness of the proposed BCS methods involving (P1) and (P2) for compressed sensing MRI (CS MRI). The MR data used in our experiments are shown[5] in Figure 2, and are labeled a-f. We simulate various undersampling patterns in k-space[6] including variable density 2D random sampling[7] [17], [31], and Cartesian sampling with variable density random phase encodes (1D random). We then use the proposed algorithms for (P1) and (P2) to reconstruct the images from undersampled measurements. Our algorithm for (P1) for MRI is called Unitary Transform learning MRI (UTMRI), and our method for (P2) for MRI is referred to as UNIon of Transforms lEarning MRI (UNITE-MRI).

We compare the reconstructions provided by our methods to those provided by the following schemes: 1) the Sparse MRI method [14] that utlilizes wavelets and total variation as *fixed* transforms; 2) the DLMRI method [31] that learns adaptive overcomplete synthesis dictionaries; 3) the PANO method [57] that exploits the non-local similarities between image patches (similar to [64]), and employs a 3D transform to sparsify groups of similar patches; and 4) the PBDWS method [65] that is a recent *partially* adaptive sparsifying transform based reconstruction method that uses redundant wavelets and trained patch-based geometric directions. We also include in our comparisons the TLMRI method that was proposed and used in the experiments in a very recent work [43]. The TLMRI method (Algorithm A1 in [43]) is for a variant of Problem (P1) involving a sparsity constraint (instead of penalties) and a transform regularizer $-\log|\det W| + 0.5\|W\|_F^2$ that controls the condition number of W.

We simulated the Sparse MRI, PBDWS, PANO, DLMRI, and TLMRI methods using the software implementations available from the respective authors' websites [66]–[70]. We used the built-in parameter settings in the first three implementations, which performed well in our experiments[8]. Specifically, for the PBDWS method, the shift invariant discrete Wavelet transform (SIDWT) based reconstructed image is used as the *guide* (initial) image [65], [67]. We employed the zero-filling reconstruction (produced within the PANO demo code [68]) as the initial guide image for the PANO method [57], [68].

The DLMRI implementation [69] used image patches of size 6×6 [31], and learned a four fold overcomplete dictionary $D \in \mathbb{R}^{36\times 144}$ using 25 iterations of the algorithm. The patch stride $r = 1$, and 14400 (found empirically) 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. (We adopted this strategy for DLMRI as it led to better performance in our experiments.) 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 ℓ_2 error threshold per patch varies linearly from 0.48 to 0.04 over the DLMRI iterations, except in the case of Figs. 2(a), 2(c), and 2(f) (noisier data), where it varies from 0.48 to 0.15 over the iterations. These parameter settings (all other settings are as per the indications in the DLMRI-Lab toolbox [69]) worked quite well for DLMRI.

For UTMRI and UNITE-MRI, image patches of size 6×6 were again used ($n = 36$ like for DLMRI), $r = 1$ (with patch wrap around), $\nu = 10^6/p$ (where p is the number of image pixels), $C = 10^5$, and $K = 16$. The image, transforms, and sparse coefficients in the algorithms are initialized[9] as indicated in Section III. The clusters in UNITE-MRI were initialized appropriately. Both algorithms ran for 120 iterations in the experiments in Sections V-C and V-D. The parameter η in our methods is set to 0.007, except in the case of Figs. 2(a), 2(c), and 2(f) (noisier data), where it is set to 0.05, and in the

[5]The images have pixel intensities (magnitudes) in the range [0, 1] (normalized). We use a gamma correction to (better) display some of the images and results in this work.

[6]We simulate the k-space of an image x using the command fftshift(fft2(ifftshift(x))) in Matlab.

[7]Although 2D random sampling is not practically realizable for 2D imaging, the sampling scheme is feasible when data corresponding to multiple image slices are jointly acquired, and the frequency encode (readout) direction is chosen perpendicular to the image plane. In this case, one could apply an inverse Fourier transform for such 3D data along the (fully sampled) readout direction, and then perform decoupled 2D reconstructions (slice by slice). The BCS methods would learn 2D models in this case (a different model for each 2D slice) that sparsify spatial features. Our experiments in this work with 2D random sampling are meant to simulate such 2D reconstructions.

[8]Upon tuning the parameters of these methods (from their default settings) for a subset of the data used in our experiments, we did not observe any marked performance improvements with tuning.

[9]While we use the naive zero-filling Fourier reconstruction to initialize x in our experiments here for simplicity, one could also use other better initializations for x such as the SIDWT based reconstructed image [65], or the reconstructions produced by recent methods (e.g., PBDWS, PANO, etc.). We have observed empirically that better initializations may lead to faster convergence of our algorithms, and our methods typically tend to improve the image quality compared to the initializations (assuming properly chosen parameters).

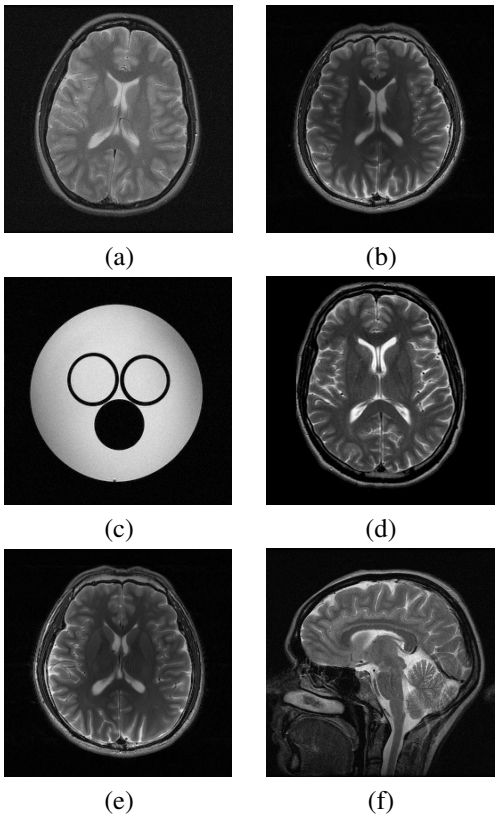

Fig. 2. Test data (only magnitudes are displayed here): (a) A 512×512 complex-valued brain image that is available for download at http://web.stanford.edu/class/ee369c/data/brain.mat; (b) 256×256 complex-valued T2 weighted brain image that is publicly available [68], and was acquired from a healthy volunteer at a 3 T Siemens Trio Tim MRI scanner using the T2-weighted turbo spin echo sequence (TR/TE = 6100/99 ms, 220×220 mm^2 field of view, 3 mm slice thickness); (c) water phantom data (complex-valued and size 256×256) that is publicly available [67], and was acquired at a 7 T Varian MRI system (Varian, Palo Alto, CA, USA) with the spin echo sequence (TR/TE =2000/100 ms, 80×80 mm^2 field of view, 2 mm slice thickness); (d) 512×512 real-valued (magnitude) MR image that was used in the simulations in a prior work [31]; (e) 256×256 complex-valued T2 weighted brain image that is publicly available [67], and was acquired from a healthy volunteer at a 3 T Siemens Trio Tim MRI scanner using the T2-weighted turbo spin echo sequence (TR/TE = 6100/99 ms, 220×220 mm^2 field of view, 3 mm slice thickness); (f) 512×512 complex-valued reference sagittal slice provided by Prof. Michael Lustig, UC Berkeley. The image in (b) has been rotated clockwise by 90° here from the orientation in [68] for display purposes. In the experiments, we use the same orientation as in [68].

experiment in Section V-B (where our algorithms' convergence behavior is illustrated through an example), where it is set to 0.07. We use even larger values of η during the initial several iterations of our algorithms in Sections V-C and V-D, leading to faster convergence and aliasing removal.

Finally, for the recent TLMRI algorithm [70], we employed similar parameter settings and initializations as for UTMRI, but initialized the sparse coefficients as indicated in Section 5.1 of our prior work [43]. Additionally, the parameters $\hat{M} = 1$ and $\lambda_0 = 0.2$ for TLMRI, and the sparsity parameter $s = 0.28 \times nN$ except in the case of Figs. 2(a), (c), and (f), where $s = 0.1 \times nN$. Even lower sparsity levels s are used during the initial several iterations of TLMRI in Sections V-C and V-D.

All simulations used Matlab. The computing platform used for the experiments was an Intel Core i5 CPU at 2.5 GHz and 4 GB memory, employing a 64-bit Windows 7 operating system.

Similar to prior work [31], we employ the peak-signal-to-noise ratio (PSNR), and high frequency error norm (HFEN) metrics to measure the quality of MR image reconstructions. The PSNR (expressed in decibels (dB)) is computed as the ratio of the peak intensity value of a reference image to the root mean square reconstruction error (computed between image magnitudes) relative to the reference. 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 15×15 pixels, and with a standard deviation of 1.5 pixels [31]. HFEN is computed as the ℓ_2 norm of the difference between the LoG filtered reconstructed and reference magnitude images.

B. Convergence Behavior

In this experiment, we consider the complex-valued reference image in Fig. 2(c), and perform 2.5 fold undersampling of the k-space of the reference. The variable density sampling mask is shown in Fig. 3(a). We study the behavior of the UTMRI and UNITE-MRI algorithms when used to reconstruct the image from the undersampled k-space measurements. UNITE-MRI is employed with 3 clusters for the image patches. The objective function (Fig. 3(e)) converged monotonically and quickly over the iterations for both UTMRI and UNITE-MRI. In particular, the objective for UNITE-MRI converged to a much lower value than for UTMRI. This is because UNITE-MRI learned a richer model and achieved a lower value than UTMRI for both the sparsification error and sparsity penalty terms in its cost (i.e., in (P2)). The sparsity fractions (i.e., the fraction of non-zeros in the sparse code matrix B) achieved by the UTMRI and UNITE-MRI algorithms over their iterations are shown in Fig. 3(f). UNITE-MRI has clearly achieved lower (i.e., better) sparsities for image patches. The changes between successive iterates $\left\|x^t - x^{t-1}\right\|_2$ (Fig. 3(g)) decrease to small values for both UTMRI and UNITE-MRI. Such behavior was established for these algorithms by Theorems 1 and 2, and is indicative (a necessary but not suffficient condition) of convergence of the entire sequence $\{x^t\}$.

The initial zero-filling reconstruction (Fig. 3(b)) has large aliasing artifacts along the phase encoding (vertical) direction, as expected for the undersampled measurements scenario. The initial PSNR is only 24.9 dB. In contrast, the UNITE-MRI reconstruction (Fig. 3(c)) is much improved and has a PSNR of 37.3 dB. Both the PSNR and HFEN metrics (Fig. 3(d)) improve significantly and converge quickly for UNITE-MRI. UTMRI exhibited similar behavior. However, the UTMRI reconstruction has a lower PSNR of 37.1 dB.

Why does UNITE-MRI provide an improvement over UTMRI in reconstructing the rather simple (mostly smooth) image in this experiment? To answer this question, we study the clustering results produced by the union-of-transforms based UNITE-MRI. Since we work with overlapping image patches, each pixel in the image belongs to several different overlapping patches. We cluster an image pixel into a particular class C_k if the majority of the patches to which it belongs

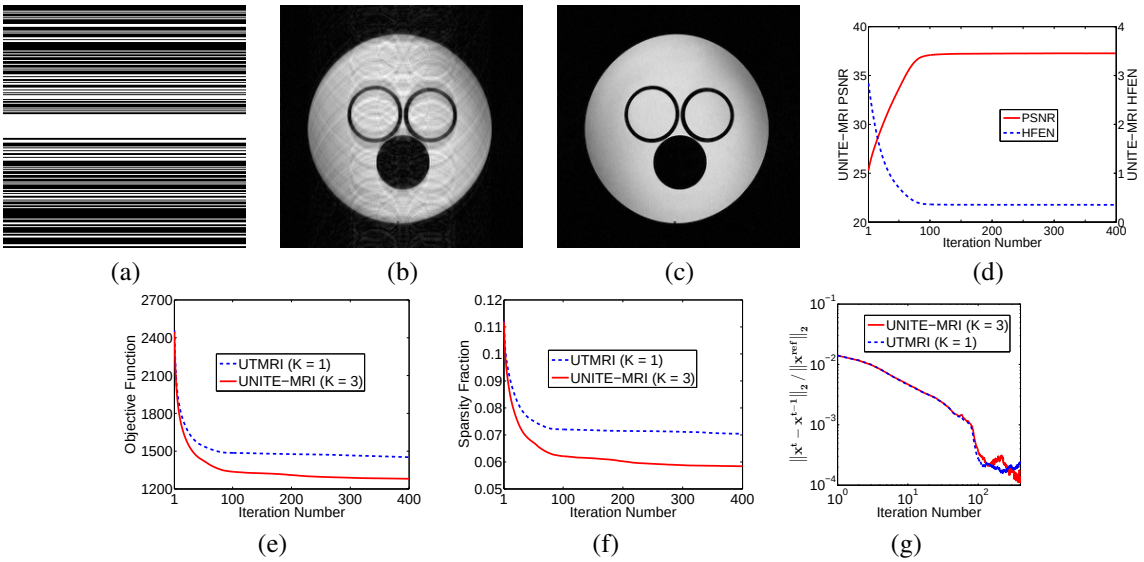

Fig. 3. Convergence behavior of UTMRI and UNITE-MRI ($K = 3$) with Cartesian sampling and 2.5x undersampling: (a) sampling mask in k-space; (b) magnitude of initial zero-filling reconstruction (24.9 dB); (c) UNITE-MRI reconstruction magnitude (37.3 dB); (d) PSNR and HFEN for UNITE-MRI; (e) objective function values for UNITE-MRI and UTMRI; (f) Sparsity fractions (i.e., fraction of non-zeros in the sparse code matrix B) for UNITE-MRI and UTMRI; and (g) changes between successive iterates ($\left\|x^t - x^{t-1}\right\|_2$) normalized by the norm of the reference image ($\left\|x^{\text{ref}}\right\|_2 = 122.2$) for UNITE-MRI and UTMRI.

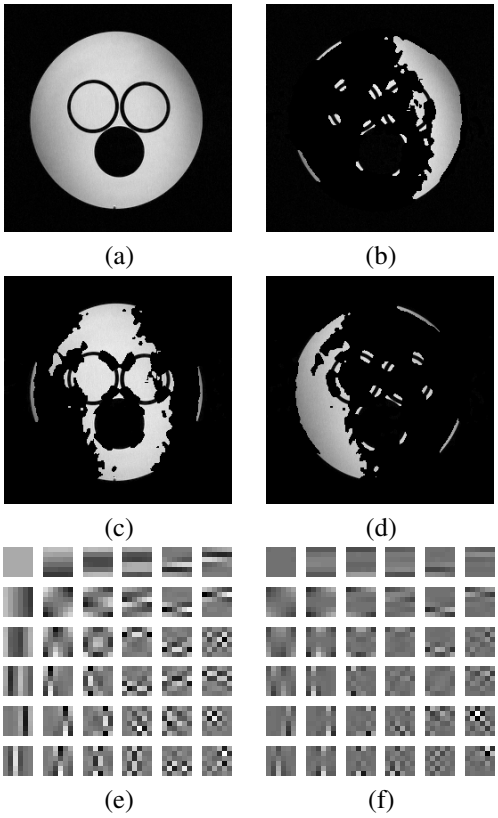

Fig. 4. UNITE-MRI ($K = 3$) clustering and learning: (a) UNITE-MRI reconstruction (magnitude shown); (b)-(d) image pixels (with reconstructed intensities) in (a) grouped into the first, second, and third clusters, respectively, overlaid on black backgrounds; (e) real and (f) imaginary parts of the learnt transform in the second cluster, with the matrix rows shown as patches. Each of the images (b)-(d) shows edges with specific orientations.

are clustered into that class by the UNITE-MRI algorithm. Figs. 4(b)-(d) show image pixels from the reconstructed image that are clustered into each of the 3 classes by UNITE-MRI. The pixels from each class (shown with the reconstructed intensities) are overlaid on a black background in these images. The results show that UNITE-MRI groups regions that share common orientations of edges. (Edges exist at a variety of orientations for the phantom image.) For example, the second cluster (Fig. 4(c)) captures near horizontal and vertical edges[10]. The learnt transform for this cluster is also shown. This is a complex-valued transform. The real (Fig. 4(e)) and imaginary (Fig. 4(f)) parts of the transform display frequency like structures, and in particular contain horizontal and vertical features that were learnt to sparsify the corresponding edges better.

In this example, the UNITE-MRI algorithm is able to group patches together according to the directionality of their edges, and it learns transforms in each cluster that are better suited to sparsify specific types of edges. Since UTMRI learns only a square transform, the learned transform is unable to sparsify the diverse features (edges) of the phantom image as well as the more adaptive and overcomplete UNITE-MRI transform. The reconstruction error maps shown later in Fig. 6 show UNITE-MRI recovering the image edges better than UTMRI.

C. Results and Comparisons

We now consider the images a-f in Fig. 2 and simulate the performance of the proposed UTMRI and UNITE-MRI algorithms at various undersampling factors, and with Cartesian sampling or 2D random sampling of k-space. Table I lists the reconstruction PSNRs corresponding to the zero-filling, Sparse MRI, DLMRI, PBDWS, PANO, TLMRI, UTMRI, and UNITE-MRI reconstructions for various cases.

The transform-based blind compressed sensing algorithms typically provide the best reconstruction PSNRs in Table I (analogous results were observed to usually hold with respect

[10]The smooth regions of the image appear in all the clusters. This is because they are fairly (equally) well sparsified by any of the learnt directional transforms.

Image	Sampling	UF	Zero-filling	Sparse MRI	DLMRI	PBDWS	PANO	TLMRI	UTMRI	UNITE-MRI
c	Cartesian	2.5x	24.9	29.9	36.6	35.8	34.8	37.2	37.2	**37.4**
d	2D random	10x	23.2	24.9	41.4	41.1	42.4	44.3	44.0	**44.6**
d	2D random	20x	21.6	22.9	34.1	36.7	37.8	38.8	38.4	**39.4**
a	Cartesian	6.9x	27.9	28.6	30.9	31.1	31.1	31.4	31.3	**31.5**
e	Cartesian	2.5x	28.1	31.7	37.5	42.5	40.0	40.7	40.8	**43.4**
b	Cartesian	2.5x	27.7	31.6	39.2	43.3	41.3	42.6	42.5	**44.3**
f	2D random	5x	26.3	27.4	30.5	30.4	30.4	30.6	30.6	**30.7**
c	2D random	6x	13.9	14.5	15.4	15.2	33.0	33.2	32.4	**33.6**
e	Cartesian	4x	28.5	30.6	32.4	**34.5**	33.7	33.6	33.5	**34.5**

TABLE I
PSNRS CORRESPONDING TO THE ZERO-FILLING, SPARSE MRI [14], DLMRI [31], PBDWS [65], PANO [57], TLMRI [43], UTMRI, AND UNITE-MRI RECONSTRUCTIONS FOR VARIOUS IMAGES AND UNDERSAMPLING FACTORS (UF), WITH CARTESIAN OR 2D RANDOM SAMPLING. THE BEST PSNRS ARE MARKED IN BOLD. THE IMAGE LABELS ARE AS PER FIG. 2.

to the HFEN metric not shown in the table). Specifically, the UNITE-MRI method provides an improvement of 3.2 dB in reconstruction PSNR on average in Table I over the recent partially adaptive PBDWS method, and an average improvement of 1.7 dB over the recent non-local patch similarity-based PANO method. It also provides significant improvements in reconstruction quality over the non-adaptive Sparse MRI method, and an average improvement of 4.6 dB over the synthesis dictionary-based DLMRI method. (Unlike the proposed transform-based schemes, the overcomplete dictionary-based DLMRI algorithm lacks convergence guarantees, and the NP-hard synthesis sparse coding in DLMRI lacks closed-form solutions.) The single transform-based TLMRI or UTMRI methods perform better than prior methods in most cases in Table I. TLMRI provides slightly better (about 0.2 dB better) PSNRs than UTMRI on the average. Importantly, the union of transforms based UNITE-MRI provides 1 dB better reconstruction PSNR on the average compared to the single transform based UTMRI. This indicates that the union of transforms (or OCTOBOS [54]) model is a better match for the characteristics of the MR images than a single transform model for all image patches.

Fig. 5 and Fig. 6 show the reconstructions (only magnitudes are displayed here and elsewhere) obtained with several methods for the image in Fig. 2(c), with Cartesian sampling and 2.5 fold undersampling of k-space. The Sparse MRI (Fig. 5(a)), PANO (Fig. 5(c)), PBDWS (Fig. 5(e)), and DLMRI (Fig. 6(a)) reconstructions show some residual artifacts that are mostly removed in the UTMRI (Fig. 6(c)) and UNITE-MRI (Fig. 6(e)) reconstructions. Figs. 5 and 6 also show the reconstruction error maps (i.e., the magnitude of the difference between the magnitudes of the reconstructed and reference images) for various methods. The error maps for the transform-based BCS methods (UTMRI, UNITE-MRI) clearly show much smaller image distortions than those for other methods. In particular, the error map for the UNITE-MRI method shows fewer artifacts along the image edges than that for the UTMRI scheme.

Fig. 7 shows another example of reconstructions obtained with the UTMRI (Fig. 7(c)) and UNITE-MRI (Fig. 7(e)) methods, along with the TLMRI reconstruction (Fig. 7(a)), for the image in Fig. 2(b) with Cartesian sampling and 2.5 fold undersampling of k-space. The reconstruction error maps for TLMRI (Fig. 7(b)), UTMRI (Fig. 7(d)), and UNITE-MRI (Fig. 7(f)) are also shown. The UNITE-MRI method with 16 clusters ($K = 16$) clearly provides a much better reconstruction of image features (i.e., fewer artifacts) than the single transform-based UTMRI and TLMRI schemes in this case.

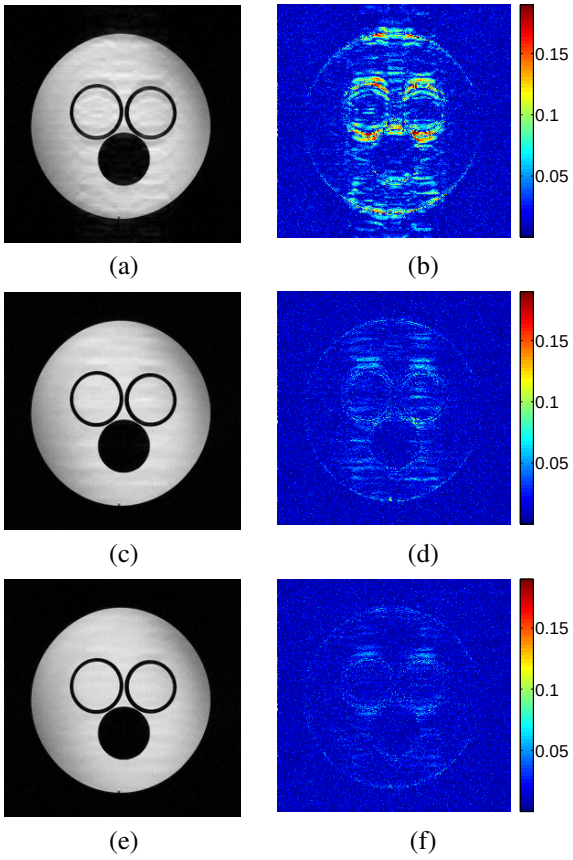

Fig. 5. Cartesian sampling with 2.5 fold undersampling. The sampling mask is shown in Fig. 3(a). Reconstructions (magnitudes): (a) Sparse MRI [14]; (c) PANO [57]; and (e) PBDWS [65]. Reconstruction error maps: (b) Sparse MRI; (d) PANO; and (f) PBDWS.

The average runtimes for the Sparse MRI, DLMRI, PBDWS, PANO[11], TLMRI, UTMRI, and UNITE-MRI methods in Table I are 166 seconds, 2581 seconds, 423 seconds, 223 seconds, 430 seconds, 291 seconds, and 1480 seconds, respectively. The PBDWS runtime includes the time taken for

[11]Another faster version of the PANO method (that uses multi-core CPU parallel computing) is also publicly available [71]. However, we found that although this version (employed with the built-in parameter settings) has an average runtime of only 25 seconds in Table I, it also provides 0.4 dB worse reconstruction PSNR on an average than the version [68] used in Table I.

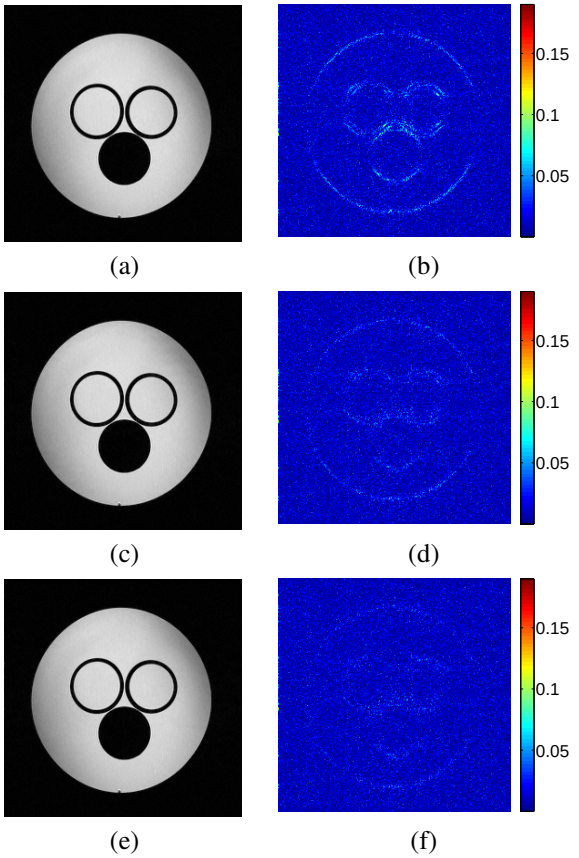

Fig. 6. Cartesian sampling with 2.5 fold undersampling. The sampling mask is shown in Fig. 3(a). Reconstructions (magnitudes): (a) DLMRI [31]; (c) UTMRI; and (e) UNITE-MRI. Reconstruction error maps: (b) DLMRI; (d) UTMRI; and (f) UNITE-MRI.

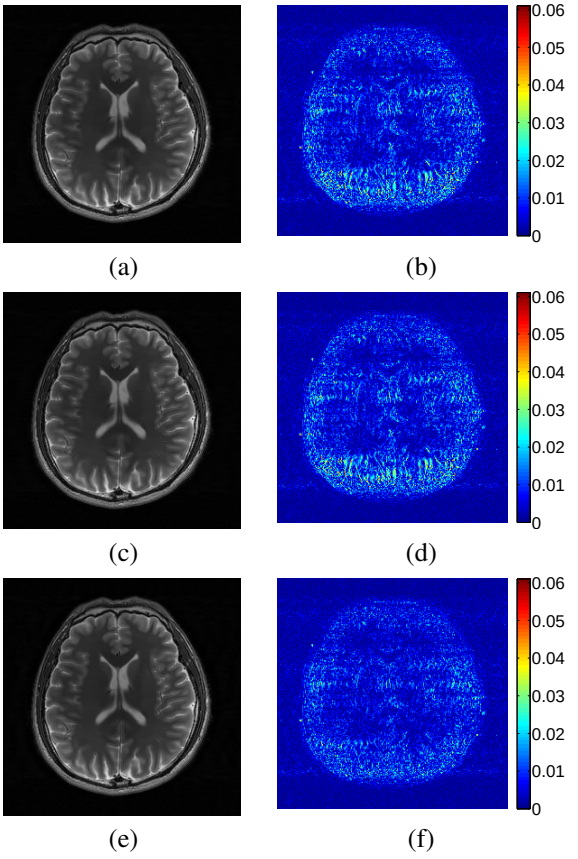

Fig. 7. Cartesian sampling with 2.5 fold undersampling. Sampling mask shown in Fig. 8(a). Reconstructions (magnitudes): (a) TLMRI (42.6 dB) [43]; (c) UTMRI (42.5 dB); and (e) UNITE-MRI with $K = 16$ (44.3 dB). Reconstruction error maps: (b) TLMRI; (d) UTMRI; and (f) UNITE-MRI. All images here have been rotated clockwise by 90° for display.

computing the initial SIDWT-based reconstruction or guide image [65] in the PBDWS software package [67]. Note that the runtimes for the proposed UTMRI and UNITE-MRI algorithms were obtained by employing our unoptimized Matlab implementations of these methods, whereas the implementations of PBDWS and PANO are based on MEX (or C) code. While UTMRI has small runtimes, the larger runtimes for UNITE-MRI can be substantially reduced (at the price of a small degradation in the reconstruction PSNRs) by performing the (computationally expensive) clustering step less often (compared to the transform update, sparse coding, and image update steps) in the block coordinate descent algorithm. For the same number of (120) algorithm iterations, UTMRI is faster than TLMRI because of the more efficient updates in UTMRI. We expect speed-ups for our algorithms with conversion of the code to C/C++, code optimization, and parallel computing.

D. The Number of Clusters in UNITE-MRI

Here, we investigate the peformance of the UNITE-MRI method as a function of the number of clusters K. We work with the same data and k-space sampling as in Fig. 7, and perform image reconstruction using the UNITE-MRI method at various values of K (all other algorithm parameters are the same as in Fig. 7). Fig. 8(g) shows the image reconstruction PSNRs for UNITE-MRI for various K values. The PSNR improves monotonically and significantly as K is increased above 1 (i.e., UTMRI). This is because UNITE-MRI learns richer and more specific or adaptive models that provide sparser representations for patches, and hence better iterative reconstructions. However, for very large K values, the PSNR saturates and begins to decrease. This is because it becomes impossible to reliably learn very rich or complex (non-trivial) models from limited compressive measurements and from limited number of patches. Fig. 8(g) shows the UNITE-MRI runtimes varying quite linearly with the number of clusters, although at a more gradual rate than $O(Kn^2N)$.[12]

Fig. 8(b) shows the UNITE-MRI reconstruction with $K = 10$ clusters. Figs. 8(c)-(f) show image pixels from the reconstructed image that are clustered into four specific classes (similar to Fig. 4). The pixels from each of these classes (shown with the reconstructed intensities) are overlaid on a black background in these images. The results show that UNITE-MRI groups together regions of the brain image with specific types of features or edges.

The data obtained from MR scanners (in Fig. 2) and used in our experiments typically contain physical noise. To explicitly evaluate the effect of measurement noise in UNITE-MRI, we add simulated i.i.d. complex Gaussian noise to the image in Fig. 2(b). This is equivalent to adding noise to the simulated

[12]The actual runtimes would also be quite dependant on the specific implementation.

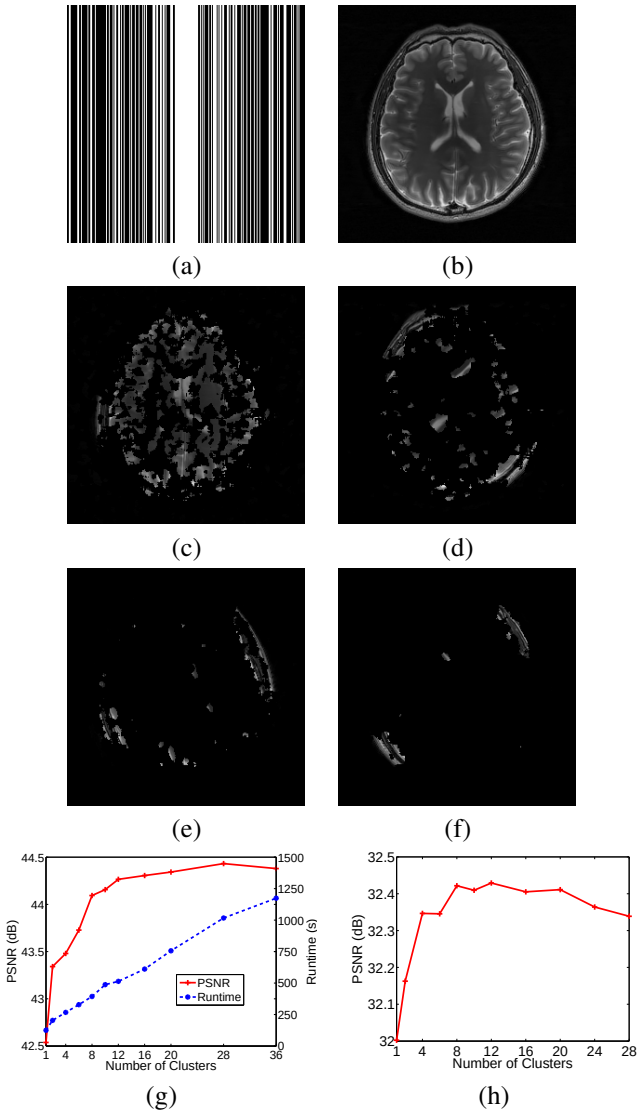

Fig. 8. Cartesian sampling with 2.5 fold undersampling: (a) k-space sampling mask; (b) UNITE-MRI reconstruction (using k-space samples of image in Fig. 2(b)) magnitude for $K = 10$ (44.2 dB); (c)-(f) image pixels (with reconstructed intensities) in (b) grouped into four specific clusters overlaid on black backgrounds; (g) reconstruction PSNR and runtime vs. number of clusters K, when k-space samples are obtained using data in Fig. 2(b); and (h) reconstruction PSNR (computed with respect to reference in Fig. 2(b)) vs. number of clusters K, when the measurements are obtained from a noisier version (PSNR = 26.7 dB) of the image in Fig. 2(b). The images (a)-(f) have all been rotated clockwise by 90° for display.

k-space data thus modeling a realistic higher noise acquisition. The corrupted image has a PSNR (computed with respect to Fig. 2(b)) of 26.7 dB. We now repeat the experiment of Fig. 8(g), but sample the k-space of the corrupted image. All algorithm parameters are the same as before, except that $\nu = 30/p$, and η is set as for Fig. 2(a). Fig. 8(h) shows the image reconstruction PSNRs computed with respect to Fig. 2(b), for UNITE-MRI for various K values. In this case, the PSNR saturates earlier than in Fig. 8(g), but UNITE-MRI still provides up to 0.43 dB better PSNRs than UTMRI ($K = 1$), and both methods achieve better PSNRs than the original (fully sampled) corrupted image (26.7 dB).

E. Extensions and Improvements

Our results show the promise of the proposed blind compressed sensing methods for MRI. The PSNRs for our schemes in our experiments can be further improved with better parameter selection strategies. There may be several directions to potentially improve the proposed methods. For example, combining the UNITE-MRI scheme (or, Algorithm A2) with the patch-based directional wavelets model [20], [65], or with nonlocal patch similarity ideas [57], [72] could potentially boost the BCS performance further. Incorporating additional information from related reference images (e.g., from a database) in the proposed framework could make our schemes more robust to noise and other artifacts. While we focused on learning a union of unitary transforms, because of the efficiency in computations that they provide, we plan to investigate unions of more general well-conditioned transforms [53] or unions of overcomplete transforms [73] in applications in future work.

In our experiments, we simulated single coil-based undersampled MRI acquisitions. To the extent that our results show reasonable signal to noise (and distortion) ratio at high acceleration, our approach would be equally applicable in practice, whether the acquisition is using a single coil or more. In usual parallel coil MRI (p-MRI) setups, there are also noise amplification issues at high acceleration factors (such as 10x or 20x), because of the increased condition number of the inverse problem. However, our data-driven and sparsity-driven approach may be able to provide accelerations on top of that provided by p-MRI. We have not explored this extension here, and leave its investigation to future work.

Finally, although image reconstruction is the primary goal in this work, the proposed BCS method involving a union of transforms model achieves joint image reconstruction and unsupervised patch clustering (resulting in image segmentation). There have been other methods for joint reconstruction and segmentation involving information theoretic criteria [74], multiscale approaches [75], or mixture models [76]–[78]. Incorporating additional prior information on the specific classes (e.g., tissue types) in our method (a semi-supervised strategy) could potentially lead to clinically meaningful segmentations. The exploration of such extensions is left for future work. The union of transforms methodology could also potentially capture textures and other features in patient or disease datasets.

VI. Conclusions

In this work, we presented a novel sparsifying transform-based framework for blind compressed sensing. The patches of the underlying image(s) were modeled as approximately sparse in an unknown (unitary) sparsifying transform, and this transform was learnt jointly with the image from only the compressive measurements. We also considered a union of transforms model that better captures the diverse features of natural images. The proposed blind compressed sensing algorithms involve highly efficient updates. We demonstrated the superior performance of the proposed schemes over several recent methods for MR image reconstruction from highly undersampled measurements. In particular, the union of transforms model outperformed the single transform model in

terms of the achieved quality of image reconstructions. The usefulness of the proposed BCS methods in other inverse problems and imaging applications merits further study.

ACKNOWLEDGMENT

The authors thank Prof. Jeffrey A. Fessler at the University of Michigan for his feedback and comments on this work.

REFERENCES

[1] S. Ravishankar and Y. Bresler, "Data-driven adaptation of a union of sparsifying transforms for blind compressed sensing MRI reconstruction," in *Proc. SPIE*, vol. 9597, 2015, pp. 959 713–959 713–10. [Online]. Available: http://dx.doi.org/10.1117/12.2188952

[2] 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.

[3] E. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[4] D. Donoho, "Compressed sensing," *IEEE Trans. Information Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[5] E. Candès and T. Tao, "Decoding by linear programming," *IEEE Trans. on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[6] P. Feng and Y. Bresler, "Spectrum-blind minimum-rate sampling and reconstruction of multiband signals," in *ICASSP*, vol. 3, may 1996, pp. 1689–1692.

[7] 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.

[8] P. Feng, "Universal spectrum blind minimum rate sampling and reconstruction of multiband signals," Ph.D. dissertation, University of Illinois at Urbana-Champaign, mar 1997, Yoram Bresler, adviser.

[9] 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.

[10] 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.

[11] 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.

[12] 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*, vol. 50, no. 3, pp. 253–270, dec 2002.

[13] Y. Bresler, "Spectrum-blind sampling and compressive sensing for continuous-index signals," in *2008 Information Theory and Applications Workshop Conference*, 2008, pp. 547–554.

[14] M. Lustig, D. Donoho, and J. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, 2007.

[15] 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.

[16] 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.

[17] J. Trzasko and A. Manduca, "Highly undersampled magnetic resonance image reconstruction via homotopic l_0-minimization," *IEEE Trans. Med. Imaging*, vol. 28, no. 1, pp. 106–121, 2009.

[18] Y. Kim, M. S. Nadar, and A. Bilgin, "Wavelet-based compressed sensing using gaussian scale mixtures," in *Proc. ISMRM*, 2010, p. 4856.

[19] 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.

[20] 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*, vol. 30, no. 7, pp. 964–977, 2012.

[21] 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.*, vol. 35, no. 2, pp. 660–663, 2008.

[22] 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.*, vol. 37, no. 9, pp. 5113–5125, 2010.

[23] 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*, vol. 10, no. 1, p. 73, 2011.

[24] 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.

[25] K. Malczewski, "Pet image reconstruction using compressed sensing," in *Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA), 2013*, Sept 2013, pp. 176–181.

[26] K. P. Pruessmann, "Encoding and reconstruction in parallel MRI," *NMR in Biomedicine*, vol. 19, no. 3, pp. 288–299, 2006.

[27] S. Ma, W. Yin, Y. Zhang, and A. Chakraborty, "An efficient algorithm for compressed MR imaging using total variation and wavelets," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2008*, 2008, pp. 1–8.

[28] X. Qu, D. Guo, Z. Chen, and C. Cai, "Compressed sensing MRI based on nonsubsampled contourlet transform," in *Proc. IEEE International Symposium on IT in Medicine and Education*, 2008, pp. 693–696.

[29] Y. Chen, X. Ye, and F. Huang, "A novel method and fast algorithm for mr image reconstruction with significantly under-sampled data," *Inverse Problems and Imaging*, vol. 4, no. 2, pp. 223–240, 2010.

[30] Q. Wang, J. Liu, N. Janardhanan, M. Zenge, E. Mueller, and M. S. Nadar, "Tight frame learning for cardiovascular mri," in *IEEE 10th International Symposium on Biomedical Imaging (ISBI)*, 2013, pp. 290–293.

[31] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Trans. Med. Imag.*, vol. 30, no. 5, pp. 1028–1041, 2011.

[32] ——, "Multiscale dictionary learning for MRI," in *Proc. ISMRM*, 2011, p. 2830.

[33] S. Gleichman and Y. C. Eldar, "Blind compressed sensing," *IEEE Transactions on Information Theory*, vol. 57, no. 10, pp. 6958–6975, 2011.

[34] S. G. Lingala and M. Jacob, "Blind compressive sensing dynamic mri," *IEEE Transactions on Medical Imaging*, vol. 32, no. 6, pp. 1132–1145, 2013.

[35] 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.

[36] J. Caballero, D. Rueckert, and J. V. Hajnal, "Dictionary learning and time sparsity in dynamic mri," in *Medical Image Computing and Computer-Assisted Intervention MICCAI 2012*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2012, vol. 7510, pp. 256–263.

[37] Y. Huang, J. Paisley, Q. Lin, X. Ding, X. Fu, and X. P. Zhang, "Bayesian nonparametric dictionary learning for compressed sensing mri," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5007–5019, 2014.

[38] S. P. Awate and E. V. R. DiBella, "Spatiotemporal dictionary learning for undersampled dynamic mri reconstruction via joint frame-based and dictionary-based sparsity," in *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, 2012, pp. 318–321.

[39] J. Caballero, A. N. Price, D. Rueckert, and J. V. Hajnal, "Dictionary learning and time sparsity for dynamic mr data reconstruction," *IEEE Transactions on Medical Imaging*, vol. 33, no. 4, pp. 979–994, 2014.

[40] S. Wang, X. Peng, P. Dong, L. Ying, D. D. Feng, and D. Liang, "Parallel imaging via sparse representation over a learned dictionary," in *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 2015, pp. 687–690.

[41] S. Ravishankar and Y. Bresler, "Sparsifying transform learning for compressed sensing MRI," in *Proc. IEEE Int. Symp. Biomed. Imag.*, 2013, pp. 17–20.

[42] 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. 90 200H–1–90 200H–11.

[43] S. Ravishankar and Y. Bresler, "Efficient blind compressed sensing using sparsifying transforms with convergence guarantees and application to magnetic resonance imaging," *SIAM Journal on Imaging Sciences*, vol. 8, no. 4, pp. 2519–2557, 2015.

[44] J. E. Fowler, "Compressive-projection principal component analysis," *IEEE Transactions on Image Processing*, vol. 18, no. 10, pp. 2230–2242, 2009.

[45] 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*, vol. 51, no. 1, pp. 34–81, 2009.

[46] S. Ravishankar and Y. Bresler, "Learning sparsifying transforms," *IEEE Trans. Signal Process.*, vol. 61, no. 5, pp. 1072–1086, 2013.

[47] B. K. Natarajan, "Sparse approximate solutions to linear systems," *SIAM J. Comput.*, vol. 24, no. 2, pp. 227–234, Apr. 1995.

[48] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Journal of Constructive Approximation*, vol. 13, no. 1, pp. 57–98, 1997.

[49] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, 2006.

[50] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. on Image Processing*, vol. 17, no. 1, pp. 53–69, 2008.

[51] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *SIAM Multiscale Modeling and Simulation*, vol. 7, no. 1, pp. 214–241, 2008.

[52] S. Ravishankar and Y. Bresler, "Learning doubly sparse transforms for images," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4598–4612, 2013.

[53] ——, "ℓ_0 sparsifying transform learning with efficient optimal updates and convergence guarantees," *IEEE Trans. Signal Process.*, vol. 63, no. 9, pp. 2389–2404, May 2015.

[54] B. Wen, S. Ravishankar, and Y. Bresler, "Structured overcomplete sparsifying transform learning with convergence guarantees and applications," *International Journal of Computer Vision*, vol. 114, no. 2-3, pp. 137–167, 2015.

[55] S. Ravishankar, B. Wen, and Y. Bresler, "Online sparsifying transform learning – part i: Algorithms," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 625–636, 2015.

[56] S. Ravishankar and Y. Bresler, "Online sparsifying transform learning – part ii: Convergence analysis," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 637–646, 2015.

[57] 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*, vol. 18, no. 6, pp. 843–856, Aug 2014.

[58] S. Ravishankar and Y. Bresler, "Closed-form solutions within sparsifying transform learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 5378–5382.

[59] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, Maryland: Johns Hopkins University Press, 1996.

[60] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[61] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *J. Optim. Theory Appl.*, vol. 109, no. 3, pp. 475–494, 2001.

[62] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*. Heidelberg, Germany: Springer-Verlag, 1998.

[63] S. Ravishankar and Y. Bresler, "Data-driven learning of a union of sparsifying transforms model for blind compressed sensing: Supplementary material," *IEEE Transactions on Computational Imaging*, 2016, http://ieeexplore.ieee.org/document/7468471/media.

[64] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3D transform-domain collaborative filtering," *IEEE Trans. on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.

[65] 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*, vol. 31, no. 9, pp. 1611–1622, 2013.

[66] M. Lustig, "Michael Lustig home page," http://www.eecs.berkeley.edu/~mlustig/Software.html, 2014, [Online; accessed October, 2014].

[67] X. Qu, "PBDWS Code," http://www.quxiaobo.org/project/CS_MRI_PBDWS/Demo_PBDWS_SparseMRI.zip, 2014, [Online; accessed September, 2014].

[68] ——, "PANO Code," http://www.quxiaobo.org/project/CS_MRI_PANO/Demo_PANO_SparseMRI.zip, 2014, [Online; accessed May, 2015].

[69] S. Ravishankar and Y. Bresler, "DLMRI - Lab: Dictionary learning MRI software," http://www.ifp.illinois.edu/~yoram/DLMRI-Lab/DLMRI.html, 2013, [Online; accessed October, 2014].

[70] ——, "Transform learning software," http://transformlearning.csl.illinois.edu/software/, 2016.

[71] X. Qu, "PANO Code with multi-core cpu parallel computing," http://www.quxiaobo.org/project/CS_MRI_PANO/Demo_Parallel_PANO_SparseMRI.zip, 2014, [Online; accessed April, 2015].

[72] Y. Mohsin, G. Ongie, and M. Jacob, "Iterative shrinkage algorithm for patch-smoothness regularized medical image recovery," *IEEE Transactions on Medical Imaging*, 2015.

[73] S. Ravishankar and Y. Bresler, "Learning overcomplete sparsifying transforms for signal processing," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013, pp. 3088–3092.

[74] I. B. Kerfoot, Y. Bresler, and A. S. Belmont, "Mean-field and information-theoretic algorithms for direct segmentation of tomographic images," in *Proc. SPIE*, vol. 1905, 1993, pp. 956–963.

[75] I. B. Kerfoot and Y. Bresler, "Theoretical analysis of a multiscale algorithm for the direct segmentation of tomographic images," in *IEEE International Conference Image Processing*, vol. 2, 1994, pp. 177–181 vol.2.

[76] I.-T. Hsiao, A. Rangarajan, and G. Gindi, "Joint-map reconstruction/segmentation for transmission tomography using mixture-models as priors," in *Conference Record. 1998 IEEE Nuclear Science Symposium*, vol. 3, 1998, pp. 1689–1693 vol.3.

[77] D. V. de Sompel and M. Brady, "Simultaneous reconstruction and segmentation algorithm for positron emission tomography and transmission tomography," in *IEEE International Symposium on Biomedical Imaging*, 2008, pp. 1035–1038.

[78] J. Caballero, W. Bai, A. N. Price, D. Rueckert, and J. V. Hajnal, "Application-driven mri: Joint reconstruction and segmentation from undersampled mri data," in *Medical Image Computing and Computer-Assisted Intervention*, 2014, pp. 106–113.