FRIST - Flipping and Rotation Invariant Sparsifying Transform Learning and Applications of Inverse Problems

Bihan Wen[1], **Saiprasad Ravishankar** [2], **and Yoram Bresler**[1]

[1] Department of Electrical and Computer Engineering and the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Champaign, IL 61801, USA.
[2] Department of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, MI 48109, USA.

E-mail: bwen3@illinois.edu, ravisha@umich.edu, and ybresler@illinois.edu

January 2016

Abstract. Features based on sparse representation, especially using synthesis dictionary model, have been heavily exploited in signal processing and computer vision. However, synthesis dictionary learning involves NP-hard sparse coding and expensive learning steps. Recently, sparsifying transform learning received interest for its cheap computation and closed-form solution. In this work, we develop a methodology for learning of Flipping and Rotation Invariant Sparsifying Transform, dubbed FRIST, to better represent natural images that contain textures with various geometrical directions. The proposed alternating learning algorithm involves simple closed-form solutions. We provide the convergence guarantee, and demonstrate empirical convergence behavior of the proposed FRIST learning algorithm. Preliminary experiments show the usefulness of adaptive sparse representation by FRIST for image sparse representation, segmentation, denoising, robust inpainting, and MRI reconstruction with promising performances.

Keywords: Sparsifying transform learning, Dictionary learning, Convergence guarantee, Overcomplete representation, Machine learning, Clustering, Image representation, Inverse problem, Image denoising, Inpainting, Magnetic resonance imaging.

1. Introduction

Sparse representation of natural signals in certain transform domain or dictionary has been widely exploited. Various sparse models, such as the synthesis model [1, 2] and the transform model [3], have been studied. The popular *synthesis model* suggests that a signal $y \in \mathbb{R}^n$ can be sparsely represented as $y = Dx + \eta$, where $D \in \mathbb{R}^{n \times m}$ is synthesis dictionary, the $x \in \mathbb{R}^m$ is sparse code, and η is small approximation error in the signal domain. Synthesis dictionary learning methods [4, 5] typically involve

a synthesis sparse coding step which is, however, NP-hard [6], such that approximate solution techniques [7, 8, 9] are widely used. Various dictionary learning algorithms [4, 10, 11, 13], especially the well-known K-SVD method [5], have been proposed and are popular in numerous applications such as denoising, inpainting, deblurring, and demosaicing [12, 14, 15]. However, they are typically computationally expensive when used for large-scale problems. Moreover, heuristic methods such as K-SVD can get easily caught in local minima, or saddle points [16].

The alternative *transform model* suggests that the signal y is approximately sparsifiable using a transform $W \in \mathbb{R}^{m \times n}$, i.e., $Wy = x + e$, with $x \in \mathbb{R}^m$ sparse, and e a small approximation error in the transform domain (rather than in the signal domain). It is well-known that natural images are sparsifiable by analytical transforms such as discrete cosine transform (DCT), or wavelet transform [17]. Furthermore, recent works proposed learning square sparsifying transforms (SST) [18], which turn out to be advantageous in various applications such as image denoising, magnetic resonance imaging (MRI), and computed tomography (CT) [18, 19, 20, 21]. Alternating minimization algorithms for learning SST have been proposed with cheap and closed-form solutions [22].

Since SST learning is restricted to one adaptive square transform for all data, the diverse patches of natural images may not be sufficiently sparsified in the SST model. A recent work focuses on learning a union of (unstructured) sparsifying transforms [23, 24], dubbed OCTOBOS, to sparsify images with different contents and diverse features. However, the unstructured OCTOBOS suffers from overfitting in various applications. While previous works exploited transformation symmetries in synthesis model sparse coding [25], and applied rotational operators with analytical transforms [26], the usefulness of rotational invariance property in learning adaptive sparse model has not been explored. Hence, in this work, we propose the Flipping and Rotation Invariant Sparsifying Transform (FRIST) learning scheme, and show that it can provide better sparse representation by capturing the "optimal" orientations of patches in natural images.

2. FRIST Model and Its Learning Formulation

FRIST Model. The learning of the sparsifying transform model [18] has been proposed recently. Here, we propose a *FRIST model* that first applies a flipping (corresponds to a mirror image of the patch) and rotation (FR) operator $\Phi \in \mathbb{R}^{n \times n}$ to a signal $y \in \mathbb{R}^n$, and suggests that $\Phi\, y$ is approximately sparsifiable by some sparsifying transform $W \in \mathbb{R}^{m \times n}$ as $W\Phi y = x + e$, with $x \in \mathbb{R}^m$ sparse, and e small. A finite set of flipping and rotation operators $\{\Phi_k\}_{k=1}^K$ is considered, and the sparse coding problem for FRIST model is as follows,

$$\text{(P1)} \quad \min_{1 \le k \le K} \min_{z^k} \left\| W\, \Phi_k\, y - z^k \right\|_2^2 \quad s.t. \quad \left\| z^k \right\|_0 \le s \ \ \forall\ k$$

Here, z^k denotes the sparse code of $\Phi_k\, y$ in the transform W domain, with maximum sparsity s. Equivalently, the optimal $\hat{z}^{\hat{k}}$ is called the sparse code in the FRIST domain. We further decompose the FR matrix as $\Phi_k \triangleq G_q\, F$, where F can be either an identity matrix, or a left-to-right flipping permutation matrix. Though there are various methods of formulating the rotation operator G with arbitrary angles [27, 28]), rotating image patches by an angle θ that is not a multiple of 90° requires interpolation, and may result in misalignment with the pixel grid. Here, we adopt the matrix $G_q \triangleq G(\theta_q)$ which permutes the patch pixels along a set of geometrical directions $\{\theta_q\}$ without interpolation. The details of constructing such $\{G_q\}$ and its implementations have been proposed in existing works [29, 30, 26]. With such implementation, the total number of generated rotations via $\{G_q\}$ is $\tilde{Q}$, which is finite and grows linearly with the data dimension n. The possible number of FR operators is $\tilde{K} = 2\tilde{Q}$.

In practice, one can select a subset $\{\Phi_k\}_{k=1}^{K}$, containing a constant number of FR candidates K ($K \leq \tilde{K}$), from which the optimal $\hat{\Phi} = \Phi_{\hat{k}}$ is chosen. For each Φ_k, the optimal sparse code $\hat{z}^k$ can be solved exactly as $\hat{z}^k = H_s(W\Phi_k y)$, where $H_s(\cdot)$ is the projector onto the s-ℓ_0 ball [31], i.e., $H_s(b)$ zeros out all but the s elements of largest magnitude in $b \in \mathbb{R}^m$. The optimal FR operator $\Phi_{\hat{k}}$ is selected to provide the smallest sparsification (modeling) error $\|W\,\Phi_{\hat{k}}\, y - H_s(W\Phi_{\hat{k}} y)\|_2^2$.

The FRIST model can be interpreted as a structured union-of-transforms model, or structured OCTOBOS model [23]. In parcular, we can construct structured sub-transform W_k as $W_k = W\Phi_k$, in which case the collection $\{W_k\}_{k=1}^{K}$ represents a union-of-transforms model. Thus, each signal y is best sparsified by one particular W_k from the union of transforms $\{W_k\}$. The transforms in the collection all share a common transform W. We name the shared common transform W as the **parent transform** in FRIST, and each generated W_k is refered as a **child transform**. Problem (P1) is similar to the OCTOBOS sparse coding problem [23], where each W_k corresponds to a block of OCTOBOS. Similar to the clustering procedure in OCTOBOS, Problem (P1) matches a signal y to a particular child transform W_k with its directional FR operator Φ_k. Thus, FRIST is potentially capable of automatically clustering a collection of signals (i.e., image patches) according to their geometric orientations. When the parent transform W is unitary, FRIST is also equivalent to an overcomplete synthesis dictionary model with block sparsity [32], with W_k^T denoting the kth block of the equivalent overcomplete dictionary. Compared to an overcomplete dictionary or OCTOBOS, FRIST is much more constrained, with fewer free parameters. This property turns out to be useful in inverse problems such as denoising and inpainting, and represents the overfitting of the model in the presence of limited or highly corrupted data or measurements.

FRIST Learning Formulation. Generally, the parent transform W can be overcomplete [33, 23, 21]. In this work, we restrict ourselves to learning FRIST with square parent transform W (i.e., $m = n$), which leads to highly efficient learning algorithm with closed-form solution. Note that the FRIST model is still overcomplete, even with a square parent W. Given the training data $Y \in \mathbb{R}^{n\times N}$, we formulate the

FRIST learning problem as follows

$$\text{(P2)} \min_{W,\{X_i\},\{C_k\}} \sum_{k=1}^{K} \left\{ \sum_{i \in C_k} \|W\Phi_k Y_i - X_i\|_2^2 \right\} + \lambda Q(W)$$
$$s.t. \;\; \|X_i\|_0 \leq s \;\; \forall \; i, \;\; \{C_k\} \in \Gamma$$

where $\{X_i\}$ represent the FRIST-domain sparse codes of the corresponding columns $\{Y_i\}$ of Y. The l_0 "norm" counts the number of non-zeros in X_i, and the set $\{C_k\}$ indicates a clustering of the signals $\{Y_i\}$ such that each signal is associated exactly with one FR operator Φ_k. The set Γ is the set of all possible partitions of the set of integers $\{1, 2, ..., N\}$, which enforces all of the C_k's to be disjoint [23].

Problem (P2) is to minimize the FRIST learning objective that includes the modeling error $\sum_{k=1}^{K} \left\{ \sum_{i \in C_k} \|W\Phi_k Y_i - X_i\|_2^2 \right\}$ for Y, as well as the regularizer $Q(W) = -\log |\det W| + \|W\|_F^2$ to prevent trivial solutions [18]. Here, the log determinant penalty $-\log |\det W|$ enforces full rank on W, and the $\|W\|_F^2$ penalty helps remove 'scale ambiguity' in the solution. Regularizer $Q(W)$ fully controls the condition number and scaling of the learned parent transform [18]. The regularizer weight λ is chosen as $\lambda = \lambda_0 \|Y\|_F^2$, in order to scale with the first term in (P2). Previous works [18] showed that the condition number and spectral norm of the optimal parent transform $\hat{W}$ approaches to 1 and $1/\sqrt{2}$ respectively, as $\lambda_0 \to \infty$ in (P2).

3. FRIST Learning Algorithm and Convergence Analysis

3.1. Learning Algorithm

We propose an efficient algorithm for solving (P2) which alternates between a *sparse coding and clustering* step, and a *transform update* step.

Sparse Coding and Clustering. Given the training matrix Y, and fixed parent transform W, we solve the following Problem (P3) for the sparse codes and clusters,

$$\text{(P3)} \quad \min_{\{C_k\},\{X_i\}} \sum_{k=1}^{K} \sum_{i \in C_k} \|W\Phi_k Y_i - X_i\|_2^2 \quad s.t. \;\; \|X_i\|_0 \leq s \;\; \forall \; i, \;\; \{C_k\} \in \Gamma$$

The modeling error $\|W\Phi_k Y_i - X_i\|_2^2$ serves as the clustering measure corresponding to signal Y_i, where the best sparse code with FR permutation Φ_k ‡ is $X_i = H_s(W\Phi_k Y_i)$. Problem (P3) is to find the "optimal" FR permutation $\Phi_{\hat{k}_i}$ for each data vector that minimizes such measure by clustering. Clustering of each signal Y_i can thus be decoupled into the following optimization problem,

$$\min_{1 \leq k \leq K} \|W\Phi_k Y_i - H_s(W\Phi_k Y_i)\|_2^2 \quad \forall \; i \tag{1}$$

‡ The FR operator $\Phi_k = G_q F$, where both G_q and F are permutation matrices. Therefore the composite operator Φ_k is a permutation matrix.

where the minimization over k for each Y_i determines the optimal $\Phi_{\hat{k}_i}$, or the cluster $C_{\hat{k}_i}$ to which Y_i belongs. The corresponding optimal sparse code for Y_i in (P3) is thus $\hat{X}_i = H_s(W\Phi_{\hat{k}_i}Y_i)$. Given the sparse code §, one can also easily recover a least square estimate of each signal as $\hat{Y}_i = \Phi_{\hat{k}_i}^T W^{-1}\hat{X}_i$. Since the Φ_k's are permutation matrices, applying and computing Φ_k^T (which is also a permutation matrix) is cheap.

Transform Update Step. We solve for W in (P2) with fixed $\{C_k\}$ and $\{X_i\}$, which leads to the following problem:

$$\text{(P4)} \quad \min_{W} \left\| W\tilde{Y} - X \right\|_F^2 + \lambda Q(W)$$

where $\tilde{Y} = \left[\Phi_{\hat{k}_1}Y_1 \mid \Phi_{\hat{k}_2}Y_2 \mid \ \ldots \ \mid \Phi_{\hat{k}_N}Y_N\right]$ contains signals after applying their optimal FR operations, and the columns of X are the corresponding sparse codes X_i's. Problem (P4) has a closed-form solution, which is similar to the transform update step in SST [22]. We first decompose the positive-definite matrix $\tilde{Y}\tilde{Y}^T + \lambda I_n = UU^T$ (e.g., using Cholesky decomposition). Then, denoting the full singular value decomposition (SVD) of the matrix $U^{-1}\tilde{Y}X^T = S\Sigma V^T$, where $S, \Sigma, V \in \mathbb{R}^{n\times n}$, an optimal transform $\hat{W}$ in (P4) is

$$\hat{W} = 0.5V\left(\Sigma + (\Sigma^2 + 2\lambda I_n)^{\frac{1}{2}}\right)S^T U^{-1} \tag{2}$$

where $(\cdot)^{\frac{1}{2}}$ above denotes the positive definite square root, and I_n is the $n \times n$ identity.

Initialization Insensitivity and Cluster Elimination. Unlike the previously proposed OCTOBOS learning algorithm [23], which requires initialization of the clusters using heuristic methods such as K-means, The FRIST learning algorithm only needs initialization of the parent transform. In Section 5.1, numerical results demonstrate the fast convergence of the proposed FRIST learning algorithm, which is insensitive to parent transform initialization. In practice, we apply a heuristic cluster elimination strategy in the FRIST learning algorithm, to select the desired K operators. In the first iteration, all of the available FR operators Φ_k's (i.e., all available child transforms W_k's) are considered for sparse coding and clustering. After each clustering step, the learning algorithm eliminates half of the operators with smallest cluster sizes, until the number of selected reduces to K.

Computational Cost Analysis. The *sparse coding and clustering* step computes the optimal sparse codes and clusters, with $O(Kn^2N)$ cost. In the transform update step, we compute the closed-form solution for the square parent transform. The cost of the closed-form solution scales as $O(n^2N)$, assuming $N \gg n$, which is cheaper than the sparse coding step. Thus, the overall computational cost per iteration of FRIST learning using the proposed alternating algorithm scales as $O(Kn^2N)$, which is typically lower than the cost per iteration of overcomplete KSVD learning algorithm. We observe that FRIST learning algorithm normally requires less number of iterations

§ The sparse code includes the value of $\hat{X}_i$, as well as the membership index $\hat{k}$ which adds just $\log_2 K$ bits to the code storage.

Table 1: Computational cost comparison among SST ($W \in \mathbb{R}^{n\times n}$), OCTOBOS ($K$ clusters, each $W_k \in \mathbb{R}^{n\times n}$), FRIST and KSVD ($D \in \mathbb{R}^{n\times m}$) learning. N is the amount of training data.

	SST.	OCTOBOS	FRIST	KSVD
Cost	$O(n^2N)$	$O(Kn^2N)$	$O(Kn^2N)$	$O(mn^2N)$

to converge, compared to K-SVD method. The computational costs per-iteration of SST, OCTOBOS, FRIST, and K-SVD learning are summarized in Table 1.

3.2. Convergence Analysis

We analyze the convergence behavior of the proposed FRSIT learning algorithm that solves (P2), assuming that every steps in the algorithms (such as SVD) are computed exactly.

Notations. The Problem (P2) is formulated with sparsity constraint, which is equivalent to the unconstrained formulation with sparsity barrier penalty $\phi(X)$ (which equals to $+\infty$ when the constraint is violated, and zero otherwise). Thus, the objective function of Problem (P2) can be rewritten as

$$f\left(W,X,\Lambda\right) = \sum_{k=1}^{K}\sum_{i\in C_k} \|W\Phi_k Y_i - X_i\|_2^2 + \phi(X) + \lambda Q(W_k) \tag{3}$$

where $\Lambda \in \mathbb{R}^{1\times N}$ is the row vector whose i$^{\text{th}}$ element $\Lambda_i \in \{1,..,K\}$, which denotes the cluster label k, corresponding to the signal $Y_i \in C_k$. We use $\{W^t, X^t, \Lambda^t\}$ to denote the output in each iteration t, generated by the proposed FRIST learning algorithm. We define the infinity norm of matrix as $\|A\| \triangleq \max_{i,j}|A_{i,j}|$, the operator $\psi_s(\cdot)$ to return the s$^{\text{th}}$ largest magnitude of a vector.

Main Results. As FRIST can be interpreted as structured OCTOBOS, the convergence results of the FRIST learning algorithm take the similar form as those obtained for the OCTOBOS learning algorithms [23] in recent works. The convergence result for the FRIST learning algorithm, solving (P2), is summarized in the following theorem and corollaries.

Theorem 1 *For each initialization* (W^0, X^0, Λ^0)*, the following conclusions hold.*

(i) The objective $\{f^t\}$ *in the FRIST learning algorithm is monotone decreasing, and converges to a finite value,* $f^* = f^*(W^0, X^0, \Lambda^0)$*.*

(ii) The iterate sequence $\{W^t, X^t, \Lambda^t\}$ *is bounded, with all of its accumulation points equivalent, i.e., achieving the same value* f^**.*

(iii) The iterate sequence has an accumulation point, and all of the accumulation points of the iterate sequence have a common objective value.

(iv) Every accumulation point $\{W, X, \Lambda\}$ *of the iterate sequence satisfies the following partial global optimality conditions*

$$(X, \Lambda) \in \arg\min_{\tilde{X},\tilde{\Lambda}} \; f\left(W, \tilde{X}, \tilde{\Lambda}\right) \tag{4}$$

$$W \in \arg\min_{\tilde{W}} \; g\left(\tilde{W}, X, \Lambda\right) \tag{5}$$

(v) Each accumulation point $\{W, X, \Lambda\}$ *satisfies the local optimality condition*

$$g\left(W + dW, X + \Delta X, \Gamma\right) \geq g\left(W, X, \Gamma\right) \tag{6}$$

which holds for all $dW \in \mathbb{R}^{n\times n}$ *satisfying* $\|dW\|_F \leq \epsilon$ *for some* $\epsilon > 0$*, and all* $\Delta X \in \mathbb{R}^{n\times N}$ *satisfying* $\|\Delta X\|_\infty < \min_k \min_{i\in C_k} \{\psi_s(W\Phi_k Y_i) : \|W\Phi_k Y_i\|_0 > s\}$

Corollary 1 *For a particular initial* (W^0, X^0, Λ^0)*, the iterate sequence in FRIST learning algorithm converges to an equivalence class of accumulation points, which are also partial minimizers satisfying* (4)*,* (5)*, and* (6)*.*

Corollary 2 *The iterate sequence* $\{W^t, X^t, \Lambda^t\}$ *in FRIST learning algorithm is globally convergent to the set of partial minimizers of the non-convex objective* $f\left(W, X, \Lambda\right)$*.*

Due to the space limit, we only provide the outline of proofs. The conclusion (i) in Theorem 1 is obvious, as the proposed alternating algorithm solve problem in each step exactly. The proof of conclusions (ii) and (iii) follow the same arguments of Lemma 3 and Lemma 5 in [23]. In the conclusion (iv), the condition (4) can be proved using the arguments in Lemma 7 from [23], while the condition (5) can be proved with the arguments in Lemma 6 from [22]. The last conclusion in Theorem 1 can be shown using arguments from Lemma 9 in [22].

The Theorem 1, and the Corollary 1 and 2 establish that with any initialization (W^0, X^0, Λ^0), the iterate sequence $\{W^t, X^t, \Lambda^t\}$, generated by the FRIST learning algorithm, converges to an equivalent class of fixed points, or an equivalence class of partial minimizers of the objective.

4. Applications

Natural, or biomedical images typically contain a variety of directional features and edges, thus the FRIST learning is particularly appealing for applications in image processing and inverse problems. In this section, we consider three such applications, namely image denoising, image inpainting, and blind compressed sensing (BCS) based magnetic resonance imaging (MRI).

4.1. Image Denoising

Image denoising is one of the most fundamental inverse problems in image processing. The goal is to reconstruct a 2D image, which is vectorized as $x \in \mathbb{R}^P$, from its measurement $y = x + h$, corrupted by noise vector h. Various denoising algorithms have been proposed recently, with state-of-the-art performance [34, 35]. Similar to previous dictionary and transform learning based image denoising methods [12, 23], we propose the following patch-based image denoising formulation using FRIST learning,

$$\text{(P5)} \quad \min_{W,\{x_i,\alpha_i,C_k\}} \sum_{k=1}^{K} \sum_{i \in C_k} \left\{ \|W\Phi_k x_i - \alpha_i\|_2^2 + \tau \|R_i\, y - x_i\|_2^2 \right\} + \lambda\, Q(W)$$
$$s.t. \;\; \|\alpha_i\|_0 \leq s_i \;\; \forall\, i, \;\; \{C_k\} \in \Gamma$$

where $R_i \in \mathbb{R}^{n \times P}$ denotes the patch extraction operator, i.e., $R_i y \in \mathbb{R}^n$ represents the ith overlapping patch of the image y as a vector.We assume N overlapping patches in total. The data fidelity term $\tau \|R_i\, y - x_i\|_2^2$ is imposed, with a weight τ that is set inversely proportional to the given noise level σ [12, 22]. The vector $\alpha_i \in \mathbb{R}^n$ represents the sparse code of x_i in the FRIST domain, with an a priori unknown sparsity level s_i.

We propose a simple iterative denoising algorithm based on (P5). Each iteration involves the following steps: (i) sparse coding and clustering, (ii) sparsity level update, and (iii) transform update. Once the iterations complete, we have a denoised image reconstruction step. We initialize the $\{x_i\}$ in (P5) using the noisy image patches $\{R_i y\}$. Step (i) is the same as it was described in Section 3.1. We then update the sparsity levels s_i for all i, similar to that in the SST learning-based denoising algorithm [31]: With fixed W and clusters $\{C_k\}$, we solve for x_i in (P5) in the least squares sense,

$$x_i = \Phi_k^T \begin{bmatrix} \sqrt{\tau}\, I \\ W \end{bmatrix}^{\dagger} \begin{bmatrix} \sqrt{\tau}\, v_i \\ H_{s_i}(W v_i) \end{bmatrix} = G_1 v_i + G_2 H_{s_i}(W v_i) \tag{7}$$

where G_1 and G_2 are appropriate matrices in the above decomposition, and $v_i \triangleq \Phi_k R_i\, y$ are the rotated noisy patches, which can be pre-computed in each iteration. We choose the optimal s_i to be the smallest integer that makes the reconstructed x_i satisfy the error condition $\|v_i - \Phi_k x_i\|_2^2 \leq nC^2\sigma^2$, where C is a constant parameter [31]. Once step (ii) is completed, we proceed to the transform update based on the method in Section 3.1. Once the iterations complete, the denoised image patches $\{x_i\}$ are obtained obtained using (7). They are restricted to their range (e.g., 0-255 for unsigned 8-bit integer class) [23]. The denoised image is reconstructed by averaging the denoised patches at their respective image locations.

For improved denoising, the algorithm for (P5) is repeated for several passes by replacing y with the most recent denoised image estimate in each pass. The noise level in each such pass is set empirically.

4.2. Image Inpainting

The goal of image inpainting is to recover missing pixels in an image. The given image measurement, with missing pixel intensities set to zero, is denoted as $y = \Xi x + \varepsilon$, where ε is the additive noise on the available pixels, and $\Xi \in \mathbb{R}^{P\times P}$ is a diagonal binary matrix with zeros only at locations corresponding to missing pixels. We propose the following patch-based image inpainting formulation using FRIST learning,

$$\text{(P6)} \quad \min_{W,\{x_i,\alpha_i,C_k\}} \sum_{k=1}^{K} \sum_{i\in C_k} \left\{ \|W\Phi_k x_i - \alpha_i\|_2^2 + \tau^2 \|\alpha_i\|_0 + \gamma \|P_i x_i - y_i\|_2^2 \right\} + \lambda\, Q(W)$$

where $y_i = R_i y$ and $x_i = R_i x$. The diagonal binary matrix $P_i \in \mathbb{R}^{n\times n}$ captures the available (non-missing) pixels in y_i. The sparsity penalty $\tau \|\alpha_i\|_0$ is imposed, and $\gamma \|P_i x_i - y_i\|_2^2$ is the fidelity term for the ith patch, with the coefficient γ that is inversely proportional to the noise standard deviation σ. The threshold τ is proportional to the noise level σ, and also increases as more pixels are removed in y.

Our proposed iterative algorithm for solving (P6) involves the following steps: (i) sparse coding and clustering, and (ii) transform update. Once the iterations complete, we have a (iii) patch reconstruction step. The sparse coding problem with sparse penalty has closed-form solution [36], and thus Step (i) is equivalent to solving the following problem,

$$\min_{1\le k\le K} \|W\Phi_k x_i - T_\tau(W\Phi_k x_i)\|_2^2 \quad \forall\ i \tag{8}$$

where the hard thresholding operator $T_\tau(\cdot)$ is defined as

$$\left(T_\tau(b)\right)_j = \begin{cases} 0 & , \ |b_j| < \tau \\ b_j & , \ |b_j| \ge \tau \end{cases} \tag{9}$$

where the vector $b \in \mathbb{R}^n$, and the subscript j indexes its vector entries. Step (ii) is similar to that in the denoising algorithm in Section 5.4.

Ideal image inpainting without noise. In the ideal case when the noise ε is absent, i.e., $\sigma = 0$, the coefficient of the fidelity term $\gamma \to \infty$. Thus the fidelity term can be replaced with hard constraints $P_i\, x_i = y_i\ \forall\, i$. In the noiseless reconstruction step, with fixed $\{\alpha_i, C_k\}$ and W, we first reconstruct each image patch x_i by solving the following problem:

$$\min_{x_i} \|W\Phi_{k_i} x_i - \alpha_i\|_2^2 \quad s.t.\ \ P_i\, x_i = y_i \tag{10}$$

We define $y_i = P_i x_i \triangleq x_i - e_i$, where $e_i = (I_n - P_i)x_i$. Because Φ_k only rearranges pixels, $\Phi_k e_i$ has the support $\Omega_i = \text{supp}(\Phi_k e_i) = \{j |\, (\Phi_k e_i)_j \neq 0\}$, which is complementary to $\text{supp}(\Phi_k y_i)$. Since the constraint leads to the relationship $x_i = y_i + e_i$ with y_i given, we solve the equivalent minimization problem over e_i as follow,

$$\min_{e_i} \|W\Phi_k e_i - (\alpha_i - W\Phi_k\, y_i)\|_2^2 \quad s.t.\ \ \text{supp}(\Phi_k e_i) = \Omega_i \tag{11}$$

Here, we define W_{Ω_i} to be the submatrix of W formed by columns indexed in Ω_i, and $(\Phi_k e_i)_{\Omega_i}$ to be the vector containing the non-zero entries of $\Phi_k e_i$. Thus, $W\Phi_k e_i = W_{\Omega_i}(\Phi_k e_i)_{\Omega_i}$, and we define $\xi^i \triangleq \Phi_k e_i$. The reconstruction problem is then re-written as the following unconstrained problem,

$$\min_{\xi^i_{\Omega_i}} \left\| W_{\Omega_i}\xi^i_{\Omega_i} - (\alpha_i - W\Phi_k\, y_i)\right\|_2^2 \quad \forall\, i \tag{12}$$

The above least squares problem has a simple solution given as $\hat{\xi}^i_{\Omega_i} = W^\dagger_{\Omega_i}(\alpha_i - W\Phi_k\, y_i)$. Accordingly, we can calculate $\hat{e}_i = \Phi_k^T\hat{\xi}^i$, and thus the reconstructed patches $\hat{x}_i = \hat{e}_i + y_i$.

Robust image inpainting. We now extend to noisy y, and propose the robust inpainting algorithm. This is useful because real image measurements are inevitably corrupted with noise [14]. The robust reconstruction step for each patch is to solve the following problem,

$$\min_{x_i} \|W\Phi_{k_i}x_i - \alpha_i\|_2^2 + \gamma\, \|P_i x_i - y_i\|_2^2 \tag{13}$$

We define $\tilde{y}_i \triangleq \Phi_{k_i}y_i$, $u_i \triangleq \Phi_{k_i}x_i$, and $\tilde{P}_i \triangleq \Phi_{k_i}P_i\Phi_{k_i}^T$, where Φ_{k_i} is permutation matrix which preserves the norm. Thus the optimization problem (13) is equivalent to

$$\min_{u_i} \|Wu_i - \alpha_i\|_2^2 + \gamma \left\|\tilde{P}_i u_i - \tilde{y}_i\right\|_2^2 \tag{14}$$

which has a least square solution $\hat{u}_i = (W^TW + \gamma\tilde{P}^T\tilde{P})^{-1}(W^T\alpha_i + \gamma\tilde{P}^T\tilde{y}_i)$. As the matrix inversion $(W^TW + \gamma\tilde{P}^T\tilde{P})^{-1}$ is expensive with a cost of $O(n^3)$ for each patch reconstruction, we apply Woodbury Matrix Identity [37] and derive equivalent solution to (14) as

$$\hat{u}_i = [B - (F_iB)^T(\frac{1}{\gamma}I_{q^i} + F_iBF_i^T)^{-1}(F_iB)](W^T\alpha_i + \gamma\tilde{P}_i^T\tilde{y}_i) \tag{15}$$

where $F_i \triangleq (\tilde{P}_i)_{\Upsilon_i}$ and $\Upsilon_i \triangleq \mathrm{supp}(\tilde{y}_i)$. The scalar $q^i = |\Upsilon_i|$ counts the number of available pixels in y_i ($q^i < n$ for inpainting problem), and $B \triangleq (W^TW)^{-1}$ can be pre-computed. Since F_iB can be easily calculated as B_{Υ_i}, the matrix inversion $(\frac{1}{\gamma}I_{q^i} + F_iBF_i^T)^{-1}$ is less expensive, with a cost of $O((q^i)^3)$. Once $\hat{u}_i$ is computed, the patch is recovered as $\hat{x}_i = \Phi_{k_i}^T\hat{u}_i$.

The reconstructed patches are all restricted to their range (e.g., 0-255 for unsigned 8-bit integer class) [23]. Eventually, we output the inpainted image by averaging the reconstructed patches at their respective image locations. We perform multiple passes in the inpainting algorithm for (P6) for improved inpainting. In each pass, we initialize $\{x_i\}$ using patches extracted from the most recent inpainted image. By doing so, we indirectly reinforce the dependency between overlapping patches in each pass.

4.3. BCS-based MRI

Compressed Sensing (CS) enables accurate MRI reconstruction from far fewer measurements than required by Nyquist sampling [43, 19, 44]. However, CS-based

MRI suffers from various artifacts at high undersampling rate, using non-adaptive analytical transforms [43]. Recent works [19] proposed BCS-based MRI methods using adaptively learned sparsifying transform, and generated superior reconstruction results. Furthermore, MRI image patches normally contain various orientations [26], which have recently been shown to be well sparsifiable by directional wavelets [30]. Compared to directional analytical transforms, FRIST can adapt to the MRI data by supervised learning, while clustering the image patches simultaneously based on their geometric orientations, which leads to more accurate sparse modeling of MRI image.

Based on the previous TL-MRI work [19], we propose a BCS-based MRI imaging scheme using adaptively learned FRIST, dubbed FRIST-MRI. We restrict the parent W to be unitary transform, instead of well-conditioned transform, which leads to a more efficient algorithm. The FRIST-MRI problem with sparsity constraint is formulated as

$$\text{(P7)} \qquad \min_{W,x,\{\alpha_i,C_k\}} \mu \left\|F_u x - y\right\|_2^2 + \sum_{k=1}^{K} \sum_{i \in C_k} \left\|W \Phi_k R_i x - \alpha_i\right\|_2^2 \\ s.t.\ W^H W = I,\ \left\|A\right\|_0 \leq s,\ \left\|x\right\|_2 \leq L,\ \{C_k\} \in \Gamma$$

Here $W^H W = I$ is the unitary constraint, $x \in \mathbb{C}^P$ is the vectorized MRI image representation, and $y \in \mathbb{C}^M$ denotes the measurements with the sensing matrix $F_u \in \mathbb{C}^{M \times P}$, which is the undersampled Fourier encoding matrix. Here $M \ll P$, as Problem (P7) is aimed to reconstruct MRI image x from highly undersampled measurements y. The sparsity term $\left\|A\right\|_0$ counts the number of non-zeros in the entire sparse matrix $A \in \mathbb{C}^{n \times P}$, whose columns are the sparse codes $\{\alpha_i\}$. Such sparsity constraint enables variable sparsity levels for each specific patches [19].

We use the block coordinate descent-type approach [19] to solve the FRIST-MRI reconstruction problem (P7). The proposed algorithm alternates between (i) sparse coding and clustering, (ii) parent transform update, and (iii) MRI image reconstruction. We initialize the FRIST-MRI algorithm with the zero-filled Fourier reconstruction $F_u^H y$ for x. Step (i) solves Problem (P7) for $\{\alpha_i, C_k\}$ with fixed W and x as

$$\min_{\{\alpha_i,C_k\}} \sum_{k=1}^{K} \sum_{i \in C_k} \left\|W \Phi_k R_i x - \alpha_i\right\|_2^2 \qquad s.t.\ \ \left\|A\right\|_0 \leq s \tag{16}$$

The exact solution to Problem (16) requires calculating the sparsification error with each possible clustering. The cost scales as $O(n^3 K^P)$, which is computationally infeasible. Instead, we provide an approximate solution which computes the sparsification error SE_k^i for each extracted patch $R_i x$, associated with each Φ_k, by solving the following problem for each k,

$$\min_{\left\{\beta_i^k\right\}} \sum_{i=1}^{P} SE_k^i = \min_{\left\{\beta_i^k\right\}} \sum_{i=1}^{P} \left\|W \Phi_k R_i x - \beta_i^k\right\|_2^2 \qquad s.t.\ \ \left\|B^k\right\|_0 \leq s \tag{17}$$

where the columns of B^k are $\left\{\beta_i^k\right\}$. The clusters $\{C_k\}$ are approximately computed, by assigning $i \in C_{\hat{k}}$ where $\hat{k} = \arg\min_k SE_k^i$. We then perform exact sparse coding, which was described in Section 3.1, to calculate $\{\alpha_i\}$ given clusters.

Cam. Pep. Man Couple House Finger. Lena

Figure 1: Testing images used in the image denoising and image inpainting the experiments.

Step (*ii*) updates the parent transform W with unitary constraint. The solution, which is similar to previous work [22], is exact with fixed $\{\alpha_i, C_k\}$ and x. We first calculate the full SVD $\Delta A = \tilde{S}\tilde{\Sigma}\tilde{V}^H$, where the columns of Δ are $\{\Phi_k R_i x\}$. The optimal unitary parent transform is then $\hat{W} = \tilde{V}\tilde{S}^H$.

Step (*iii*) solves for x with fixed W and $\{\alpha_i, C_k\}$ as

$$\min_x \sum_{k=1}^{K} \sum_{i \in C_k} \|W\Phi_k R_i x - \alpha_i\|_2^2 + \mu \|F_u x - y\|_2^2 \qquad s.t. \;\; \|x\|_2 \leq L \tag{18}$$

As Problem (18) is a least squares problem with ℓ_2 constraint, it can solved exactly using the Lagrange multiplier method [45, 19], which is equivalently to solving

$$\min_x \sum_{k=1}^{K} \sum_{i \in C_k} \|W\Phi_k R_i x - \alpha_i\|_2^2 + \mu \|F_u x - y\|_2^2 + \rho(\|x\|_2^2 - L) \tag{19}$$

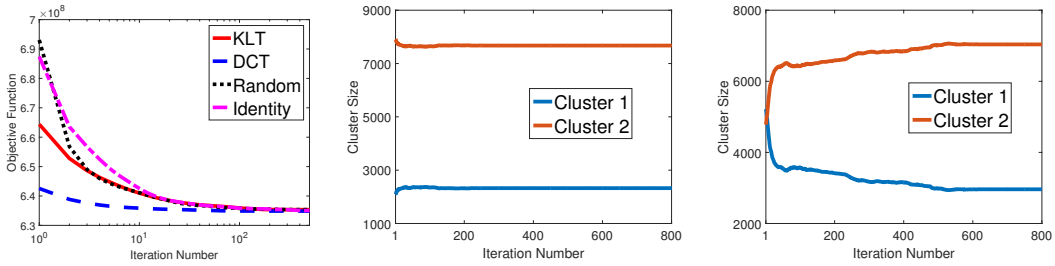
where $\rho \geq 0$ is the Lagrange multiplier. Similar to the simplification that was proposed in previous TL-MRI work [19], the normal equation of Problem 19 can be simplified as

$$(FEF^H + \mu F F_u^H F_u F^H + \rho I)Fx = F \sum_{k=1}^{K} \sum_{i \in C_k} R_i^H \Phi_k^H W^H \alpha_i + \mu F F_u^H y \tag{20}$$

where $E \triangleq \sum_{k=1}^{K} \sum_{i \in C_k} R_i^H \Phi_k^H W^H W \Phi_k R_i = \sum_{i=1}^{P} R_i^H R_i$. As FEF^H, $\mu F F_u^H F_u F^H$, and ρI are all diagonal matrices, the matrix which pre-multiplies Fx is diagonal and invertible. Thus, x can be reconstructed efficiently using the method proposed by TL-MRI work [19].

5. Experiments

We present numerical convergence result of the FRIST learning algorithm, image segmentation, as well as some preliminary results demonstrating the promise of FRIST learning in applications including image sparse representation, denoising, robust inpainting, and MRI reconstruction. We work with 8×8 non-overlapping patches for convergence and sparse representation, 8×8 overlapping patches for the image segmentation, denoising, and robust inpainting, and 6×6 overlapping patches (including wrap-around patches) for MRI experiments. Figure.1 lists the testing images which are used in image denoising and inpainting experiments.



(a) FRIST Objective (b) DCT initialization (c) KLT initialization

Figure 2: Convergence of FRIST objective and cluster size with various parent transform initializations.

5.1. Empirical convergence results

We first illustrate the convergence behavior of FRIST learning. We randomly extract 10^4 non-overlapping patches from the 44 images in the USC-SIPI database [38] (the color images are converted to gray-scale images), and learn a FRIST, with a 64×64 parent transform W, from the randomly selected patches using fixed sparsity level $s = 10$. We set $K = 2$, and $\lambda_0 = 3.1 \times 10^{-3}$ for visualization simplicity. In the experiment, we initialize the learning algorithm with different square 64×64 parent transform W's, including (i) Karhunen-Loève Transform (KLT), (ii) 2D DCT, (iii) random matrix with i.i.d. Gaussian entries (zero mean and standard deviation 0.2), and (iv) identity matrix. Figure 2(a) illustrates the convergence behavior of the objective functions over iterations, with different parent W initializations. The final values of the objective are identical for all the initializations. Figure 2(b) and Figure 2(c) show the cluster size changes over iterations for 2D DCT and KLT initializations. The final values of the cluster sizes are similar (although, not necessarily identical) for various initializations.

The numerical results demonstrate that our FRIST learning algorithm is reasonably robust, or insensitive to initialization. Good initialization for the parent transform W, such as DCT, leads to faster convergence of learning. Thus, we initialize parent transform W using 2D DCT, in the rest of the experiments.

5.2. Image Segmentation and Clustering Behavior

FRIST learning algorithm is capable of clustering image patches according to their orientations. In this subsection, we illustrate the FRIST clustering behavior by image segmentation experiment. First, we consider natural images *Wave* (512×512) and *Field* (512×512) as inputs, shown in Fig. 3(a) and Fig. 4(a). Both images contains directional textures, and we aim to cluster the pixels of the images into one of the four classes, which represent different orientations. For each input image, we convert it into gray-scale, and extract the overlapping mean-subtracted patches, and learn a FRIST while clustering the patches using algorithm in Section 3.1. As the overlapping patches are used, each pixel in the image may belong to several classes. We cluster a pixel into

Table 2: PSNR values for reconstruction of images from sparse representation obtained using the 2D DCT, SST, OCTOBOS, square and overcomplete K-SVD, and our proposed FRIST method. The first row of the table provides average PSNR values computed over the 44 images from USC-SIPI database. The best PSNR values are marked in bold.

Methods	2D DCT	SST	OCTOBOS	K-SVD		**FRIST**
Model Size		64×64	128×64	64×64	64×128	64×64
USC-SIPI	34.36	34.20	33.62	34.11	35.08	**35.14**
Cameraman	29.49	29.43	29.03	29.09	30.16	**30.63**
House	36.89	36.36	35.38	36.31	37.41	**37.71**

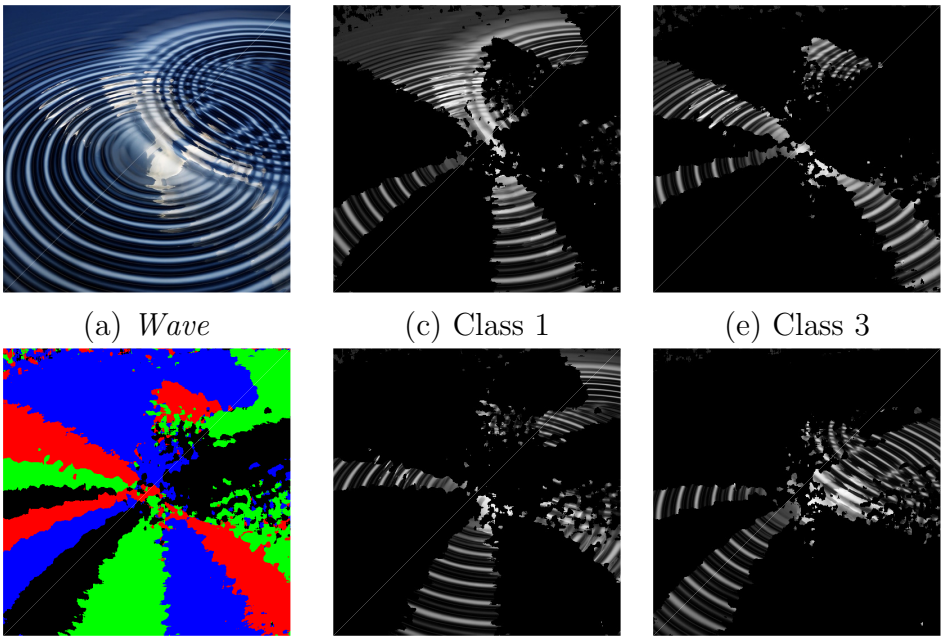
a particular class by majority voting.

We set $s = 10$, and $K = 4$ in the clustering experiments. Figure 3 and Fig.4 illustrate the segmentation results of images *Wave* and *Field*. Figure 3(b) and Fig.4(b) illustrates the pixel membership with four different colors (blue, red, green, and black, for class 1 to 4 respectively). Figures 3(c)-(f) and Figs. 4(c)-(f) each visualize the image pixels clustered into a specific class in gray-scale, and the pixels that are not clustered into that class are shown in black. The parent transform W and its child transforms W_k's in the learned FRIST for *Wave* are visualized in Fig. 5 with rows of each W_k displayed as patches. We observe that each child transform contains distinct directional features that were adaptively learned, in order to sparsify edges with various orientations better. The parent W turns out to be identical to the child transform shown in Fig. 4(d), meaning that the corresponding FR operator is identity matrix. We also observed reasonable clustering / image segmentation results with other images.

The preliminary image segmentation results demonstrate some potential for FRIST scheme in directional classification. More importantly, we wish to illustrate why FRIST can provide improvement over SST or OCTOBOS in various inverse problems. As natural images usually contain directional textures, FRIST is capable of grouping those patches with similar orientations, and thus provide better sparsification in each cluster by learning directional child transform correspondingly.

5.3. Image Sparse Representation

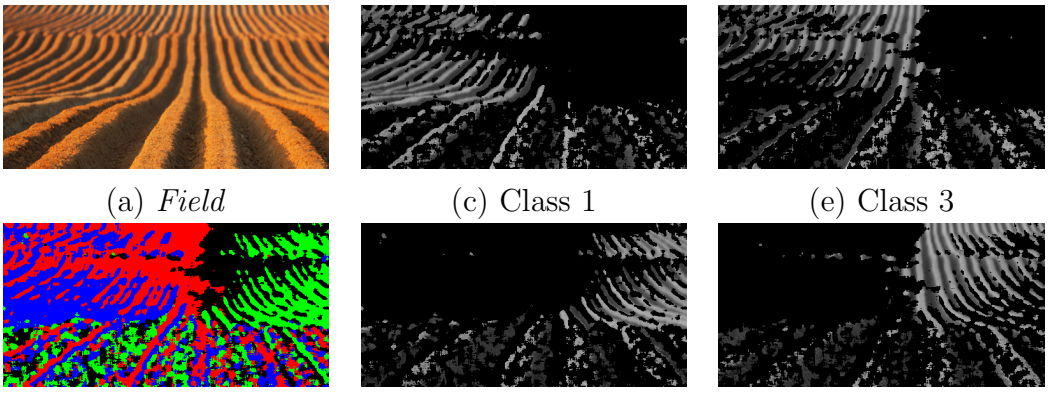
Most of the popular image compression methods make use of analytical sparsifying transforms. In particular, the commonly used JPEG adopts the 2D DCT to sparsify image patches. Data-driven adaptation of dictionaries using K-SVD scheme has also been shown to be beneficial for image compression, compared to fixed analytical transforms [39]. In this section, we do not attempt to demonstrate a complete image compression scheme providing state-of-the-art performance, but rather show that the proposed FRIST learning scheme provides improved image sparse representation, which



(a) *Wave* (c) Class 1 (e) Class 3

(b) Pixel memberships (d) Class 2 (f) Class 4

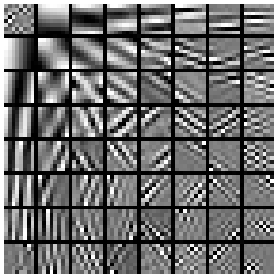
Figure 3: Image segmentation result of *Wave* (512×512) using FRIST learning on the gray-scale version of the image. The Four different colors present pixels that are belong to the four classes. Pixels which are clustered into a specific class are shown in the gray-scale, while pixels which are not clustered into that class, are shown in black for (c)-(f).



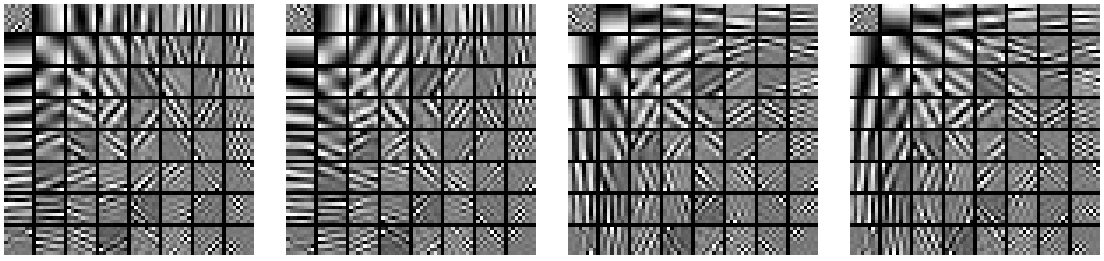
(a) *Field* (c) Class 1 (e) Class 3

(b) Pixel memberships (d) Class 2 (f) Class 4

Figure 4: Image segmentation result of *Field* (256×512) using FRIST learning on the gray-scale version of the image. The Four different colors present pixels that are belong to the four classes. Pixels which are clustered into a specific class are shown in the gray-scale, while pixels which are not clustered into that class, are shown in black for (c)-(f).



(a) Parent transform



(b) (c) (d) (e)

Figure 5: Visualization of the learned (a) parent transform, and (b)-(e) child transforms in FRIST for image *Wave*. The rows of each child transform is displayed as patches.

can be potentially applied in adaptive image compression framework.

We learn a FRIST, with 64×64 parent transform, from the 10^4 randomly selected patches (from USC-SIPI images) used in Section 5.1. We set $K = 32$, $s = 10$ and $\lambda_0 = 3.1 \times 10^{-3}$. To compare with other popular adaptive sparse models, we also train a 64×64 SST [22], 128×64 OCTOBOS [23], as well as a 64×64 square (synthesis) dictionary and a 64×128 overcomplete dictionary using KSVD [12] using the same training patches.

With the learned models, we represent each image from the USC-SIPI database compactly, by storing its sparse representation, including (*i*) non-zeros in the sparse codes of the 8×8 non-overlapping patches, (*ii*) locations of the non-zeros (plus the cluster membership if necessary) which needs only small overhead cost, and (*iii*) the adaptive sparse model. For each method, the patch sparsity (or equivalently, the number of (*i*) non-zeros in each patch) is set to be consistently as $s = 10$. The adaptive SST, square KSVD, and FRIST methods stores only 64×64 square matrix, whereas the overcomplete KSVD and OCTOBOS methods stores 128×64 matrix.

The images are then reconstructed from their sparse representations in a least squares sense, and the reconstruction quality for each image is evaluated using Peak-Signal-to-Noise Ratio (PSNR), expressed in decibels (dB). We use the average of the PSNR values over all 44 images as the indicator of the quality of compression of the USC-SIPI database. Additionally, we apply the learned sparse models to compress some standard images that are not included in the USC-SIPI database.

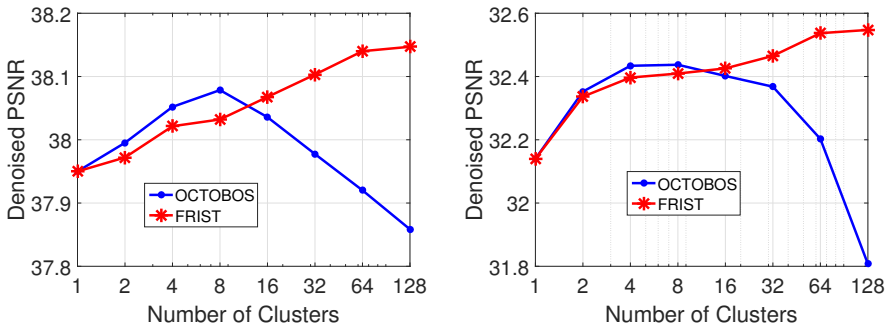
Table 3: PSNR values (in dB) for denoising with 64×64 adaptive FRIST along with the corresponding PSNR values for denoising using the 64×64 2D DCT, the 64×64 SST, the 64×256 overcomplete K-SVD, the 256×64 OCTOBOS, and BM3D. The best PSNR values are marked in bold.

Image	σ	Noisy PSNR	DCT	SST	K-SVD	OCTOBOS	BM3D	FRIST
Peppers	5	34.14	37.70	37.95	37.78	38.09	38.09	**38.16**
	10	28.10	34.00	34.37	34.24	34.57	34.66	**34.68**
(256×256)	15	24.58	31.83	32.14	32.18	32.43	**32.69**	32.54
Cameraman	5	34.12	37.77	38.01	37.82	38.16	**38.21**	38.16
	10	28.14	33.63	33.90	33.72	34.13	34.15	**34.16**
(256×256)	15	24.61	31.33	31.65	31.51	31.95	31.91	**31.97**
Man	5	34.15	36.59	36.64	36.47	36.73	36.76	**36.82**
	10	28.13	32.86	32.95	32.71	32.98	**33.18**	33.06
(768×768)	15	24.63	30.88	30.96	30.78	31.07	**31.32**	31.10
Lena	5	34.16	38.52	38.62	38.61	38.72	38.70	**38.72**
	10	28.12	35.36	35.52	35.49	35.64	**35.88**	35.67
(512×512)	15	24.64	33.51	33.68	33.72	33.91	**34.26**	33.93
Couple	5	34.16	37.25	37.32	37.29	37.40	**37.48**	37.43
	10	28.11	33.48	33.60	33.50	33.73	**34.01**	33.78
(512×512)	15	24.59	31.35	31.47	31.44	31.71	**32.08**	31.71

Table 2 lists the sparse representation reconstruction results. We observe that our learned FRIST provides the best reconstruction quality compared to other adaptive sparse models or analytical 2D DCT, for both the images in the database and the external images. Compared to unstructured overcomplete models such as KSVD and OCTOBOS, our proposed FRIST provides promising image sparse representation quality, while requiring fewer bits to represent the model itself by maintaining lower model richness. We only need to store a small parent transform, and the cluster membership indices for the FRIST model. Additionally, dictionary learning based representation requires synthesis sparse coding, which is more expensive compared to the cheap and exact sparse coding in transform model based methods [18]. The investigation of a complete image compression scheme using FRIST, with quantitative experiments and analysis is left to future work.

5.4. Image Denoising

We present denoising results using our FRIST-based framework in Section 4.1. We simulate i.i.d. Gaussian noise at 3 different noise levels ($\sigma = 5,\ 10,\ 15$) for five standard images. Denoising results obtained by our proposed algorithm in Section



(a) $\sigma = 5$ (b) $\sigma = 15$

Figure 6: Denoising PSNR for *Peppers* as a function of the number of cluster K.

4.1 are compared, with those obtained by the adaptive overcomplete K-SVD denoising scheme [12], adaptive SST denoising scheme [22], adaptive OCTOBOS denoising scheme [23], and BM3D [35], which is a state-of-the-art image denoising method. We also compare to the denoising result using the SST method, but with fixed 2D DCT.

We set $K = 64$, $n = 64$, $C = 1.04$ for the FRIST denoising method. For adaptive SST, and OCTOBOS denoising methods, we follow the same parameter settings proposed in the previous works [22, 23]. The same parameter settings for SST method is used for DCT based denoising algorithm. A corresponding 64×256 synthesis dictionary is used in the synthesis K-SVD denoising method, to match with the same model richness as the 256×64 OCTOBOS. For the K-SVD, and BM3D methods, we use the publicly available implementations in this experiment.

Table 3 lists the denoised PSNR results. The proposed FRIST scheme constantly provides better PSNRs compared to other fixed or adaptive sparse modeling methods including DCT, SST, K-SVD, and OCTOBOS for all testing bases. Figure 6 plot the denoising PSNRs for *Peppers* as a function of the number of child transforms (i.e., the number of clusters) K for $\sigma = 5$ and $\sigma = 15$. In both cases, the denoising PSNRs of OCTOBOS and FRIST schemes increase with K initially. Beyond an optimal value of K, OCTOBOS denoising scheme suffers from overfitting which causes the denoising PSNR decreases [23]. This effect is more pronounced the higher the noise level, by comparing Fig. 6(a) and Fig. 6(b). Instead, FRIST based denoising scheme, which has constant degree of freedom while K increases, provides monotonically increasing denoising PSNR. Compared to BM3D, FRIST can also provide comparable denoising PSNRs. We expect the denoising PSNRs for FRIST to improve further with optimal parameter tuning.

5.5. Image Inpainting

We present preliminary results for our adaptive FRIST-based inpainting framework (based on (P6)). We randomly remove 80% and 90% of the pixels of the entire image, and simulate i.i.d. additive Gaussian noise with $\sigma = 0$, 5, 10, and 15. We set $K = 64$, $n = 64$,

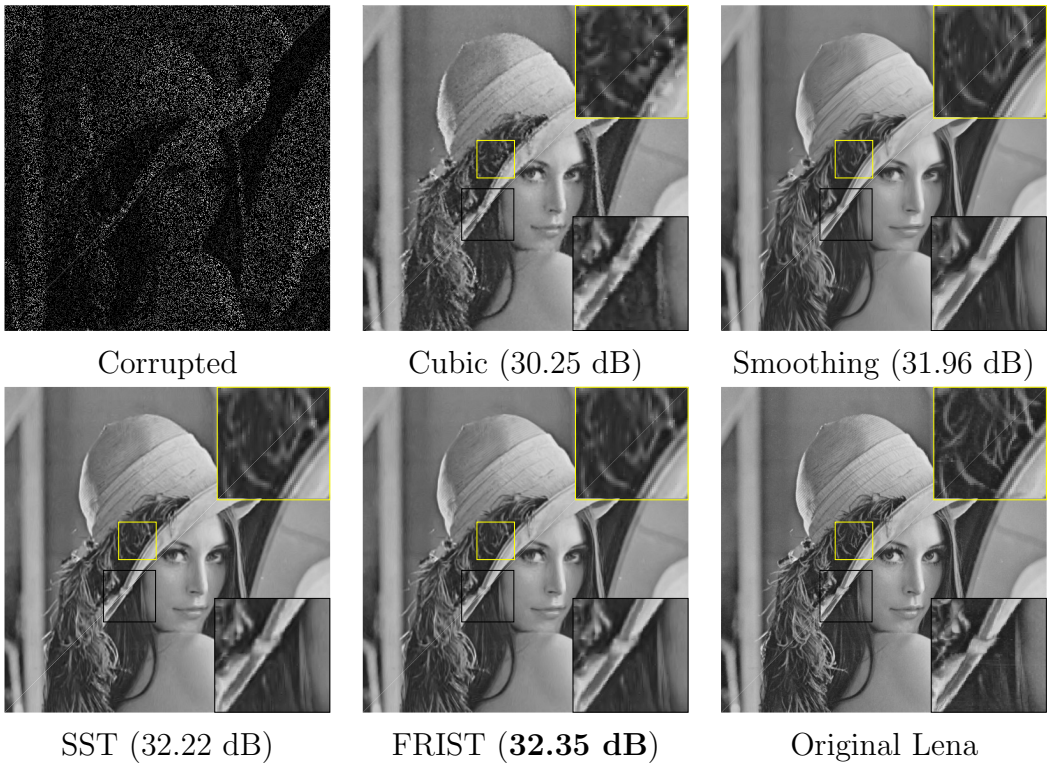


Figure 7: Illutration of image inpainting results for *Lena*, with regional zoom-in comparisons.

and apply the proposed FRIST inpaiting algorithm to reconstruct the image from the highly corrupted and noisy measurements. We additionally replace the adaptive FRIST in the proposed inpainting algorithm with fixed 2D DCT, adaptive SST [22], and OCTOBOS [23], and evaluate the inpainting performance using the corresponding models for comparison. The image inpainting results obtained by the FRIST based methods are also compared with those obtained by the cubic interpolation [40, 41], and patch smoothing method [42]. We used the Matlab function "griddata" to implement the cubic interpolation, and use the publicly available implementations of the patch smoothing methods. For the DCT, SST, OCTOBOS and FRIST based methods, we initialize the image patches using Cubic method in noiseless cases, and using Smooth method in noisy cases.

Table 4 lists the image inpainting PSNR results, with various amount of missing pixels and noise levels. The proposed FRIST inpainting scheme provides better PSNRs compared to all other inpainting methods based on interpolation, transform-domain sparsity, and spatial similarity. Figure 7 provides an illustration of the inpainting results, with regional zoom-in for visual comparison. We observe that the cubic interpolation produces blurry effects in various locations. Cubic interpolation method is extremely sensitive to noise, whereas the FRIST based method is the most robust. Compared to the competitors, FRIST method provides larger inpainting PSNR improvement as the

Table 4: PSNR values for image inpainting, averaged over six images, using adaptive FRIST based method, along with the corresponding values obtained using cubic interpolation (Cubic), patch smooth ordering (Smooth), patch-based DCT, adaptive SST, and adaptive OCTOBOS based methods, with various fractions of available pixels and noise levels. The best PSNR value in each row is marked in bold.

Available pixels	σ	Corrupted PSNR	Cubic	Smooth	DCT	SST	OCTOBOS	FRIST
20%	0	6.40	26.56	28.87	29.23	29.25	29.27	**29.33**
	5	6.39	6.40	28.64	29.21	29.24	29.26	**29.31**
	10	6.37	6.37	27.07	28.13	28.73	28.99	**29.16**
	15	6.33	6.33	25.52	26.94	28.07	28.44	**28.67**
10%	0	5.89	24.02	25.77	26.14	26.16	26.14	**26.20**
	5	5.87	5.88	25.46	25.59	25.85	25.96	**26.08**
	10	5.86	5.86	24.67	25.02	24.98	25.27	**25.46**
	15	5.82	5.82	23.73	24.19	24.47	24.68	**24.88**

noise level and the amount of missing pixels increase, by using a highly constrained adaptive overcomplete sparse model.

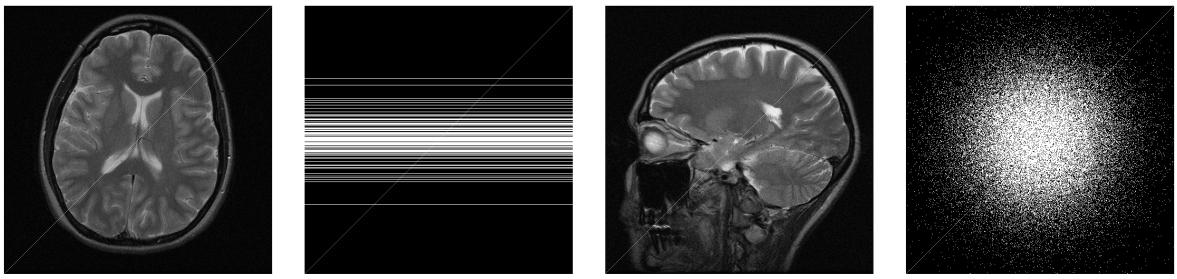
5.6. MRI Reconstruction

We present preliminary MRI reconstruction results using the proposed FRIST-MRI algorithm. The three complex images and the corresponding k-spaces sampling masks (with various sampling schemes and undersampling rate), which we work with in this section, are shown in Fig. 8, Fig. 9 (a), and Fig. 9 (b) ‖. We set $K = 32$, the sparsity level $s = 0.05 \times nP$, and other parameters similar to those used in [19] for reconstructing Image ***1*** and Image ***2***. We used higher sparsity level $s < 0.085 \times nN$ for reconstructing Image ***3*** which contains more smooth regions. To make the algorithm converge faster, lower sparsity level is used in the initial iterations [19]. We compare our FRIST-MRI reconstruction results to those obtained using conventional or popular methods, including Zero-filling, Sparse MRI [44], DL-MRI [48], PBDWS [46], PANO [47], and TL-MRI [19]. In order to make fair comparison to the TL-MRI method, we tune the sparsity parameter for constructing Image ***3*** ¶. The comparison of the reconstruction PNSRs are listed in Table 5.

First, our proposed FRIST-MRI algorithm provides significant improvements over

‖ The testing image data in this section were used and included in previous works [19, 26] with the data sources.

¶ We observed improved reconstruction PSNR compared to the result obtained using the sparsity level described in [19].



(a) (b) (c) (d)

Figure 8: Testing MRI images and their k-space sampling masks: (a) Image ***1***; (b) k-space sampling mask (Cartesian with 7× undersampling) for Image ***1***; (c) Image ***2***; (d) k-space sampling mask (2D random with 5× undersampling) for Image ***2***.

Table 5: Comparison of the PSNRs, corresponding to the Zero-filling, Sparse MRI [44], DL-MRI [48], PBDWS [46], PANO [47], TL-MRI [19], and the proposed FRIST-MRI reconstructions for various images, sampling schemes, and undersampling factors. The best PSNR for each MRI image is marked in bold.

Image	Sampling Scheme	Under-sampl.	Zero-filling	Sparse MRI	DL-MRI	PBDWS	PANO	TL-MRI	FRIST-MRI
1	Cartesian	7×	27.9	28.6	30.9	31.1	31.1	31.2	**31.4**
2	2D Random	5×	26.9	27.9	30.5	30.3	30.4	30.6	**30.7**
3	Cartesian	2.5×	24.9	29.9	36.6	35.8	34.8	36.3	**36.7**

the Zero-filling results (the initialization of the algorithm) with 6.4dB higher PSNR, as well as the reconstruction results using Sparse MRI with 4.2dB higher PSNR, averaged over all testing data. Comparing to recently proposed popular MRI reconstruction methods, FRIST-MRI algorithm demonstrates reasonably better performance for each testing case. We obtained an average PSNR improvement of 0.8dB, 0.5dB, and 0.3dB for FRIST-MRI over the non-local patch similarity-based PANO method, the partially adaptive PBDWS method, and the adaptive dictionary-based DL-MRI method. We observed that the MRI methods with adaptively learned regularizer usually provide much better reconstruction quality, compared to those using fixed models.

The proposed FRIST-MRI reconstruction quality is somewhat better than TL-MRI, with an 0.2dB PSNR improvement in average. As we followed the similar reconstruction framework and parameters used by the TL-MRI method, the quality improvement using FRIST-MRI is solely because the learned FRIST can serve as a better regularizer for MRI image reconstruction, compared to the single square transform. We also observed that FRIST-MRI used lower average sparsity level than TL-MRI for each testing case, and thus it provides better sparse representations for MRI images. Figure 9 visualizes the

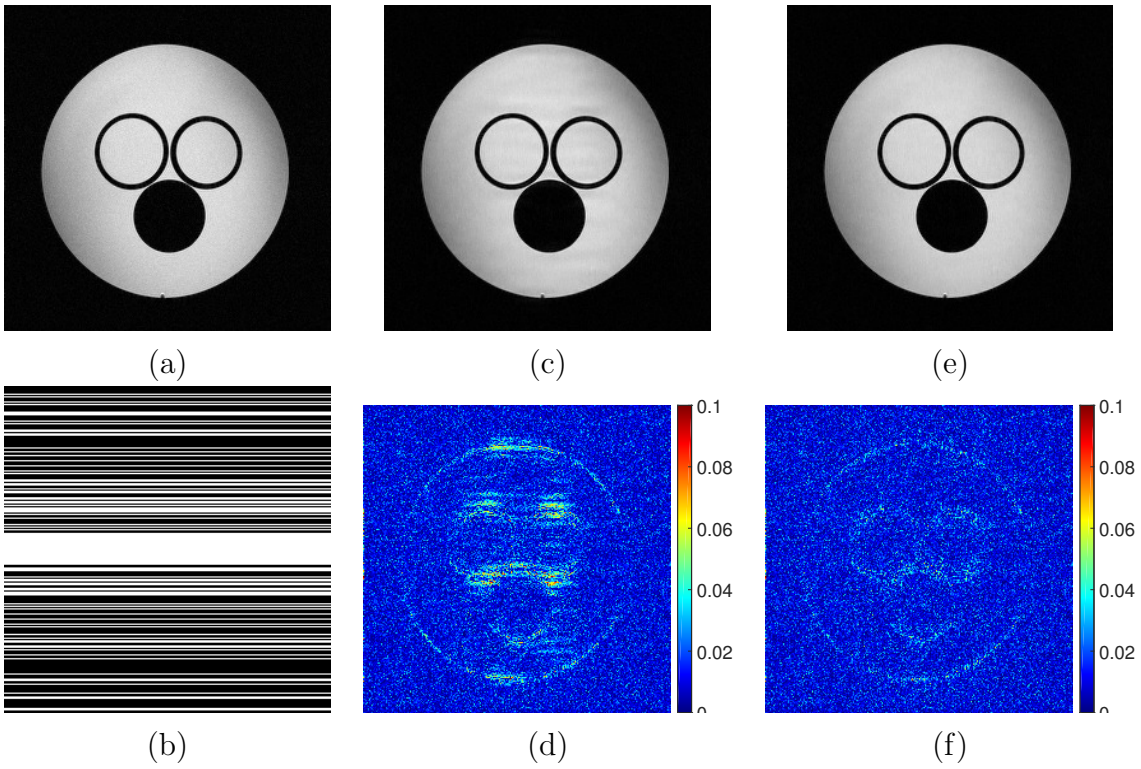


Figure 9: Visualization of reconstruction result of Image ***3*** using Cartesian and 2.5× undersampling: (a) Image ***3***; (b) sampling mask in k-space; (c) TL-MRI reconstruction (36.3dB); (d) magnitude of TL-MRI reconstruction error; (e) FRIST-MRI reconstruction (36.7dB); (f) magnitude of FRIST-MRI reconstruction error.

reconstruction quality comparison between FRIST-MRI and TL-MRI. The magnitude of FRIST-MRI reconstruction error clearly shows fewer artifacts, especially along the edges of the circles, compared to TL-MRI.

6. Conclusion

In this paper, we presented a novel framework for rotation and flipping invariant sparsifying transform learning. In particular, we proposed a method for learning a structured union-of-transforms model, dubbed FRIST. The collection of transforms in FRIST are related to a generating parent transform, by flipping and rotation operations. Our algorithm for FRIST learning is highly efficient, and involves closed-form solution with convergence guarantee. We have demonstrated the ability of FRIST learning in extracting directional features in images. FRIST is insensitive to initialization, and performs better than several prior methods in applications, including sparse image representation, image denoising, image inpainting, and MRI reconstruction (blind compressed sensing).

References

[1] Bruckstein A M, Donoho D L and Elad M 2009 *SIAM Review* **51** 34–81
[2] Elad M, Milanfar P and Rubinstein R 2007 *Inverse Problems* **23** 947–968
[3] Pratt W K, Kane J and Andrews H C 1969 *Proc. IEEE* **57** 58–68
[4] Engan K, Aase S and Hakon-Husoy J 1999 Method of optimal directions for frame design *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing* pp 2443–2446
[5] Aharon M, Elad M and Bruckstein A 2006 *IEEE Transactions on signal processing* **54** 4311–4322
[6] Davis G, Mallat S and Avellaneda M 1997 *Journal of Constructive Approximation* **13** 57–98
[7] Pati Y, Rezaiifar R and Krishnaprasad P 1993 Orthogonal matching pursuit : recursive function approximation with applications to wavelet decomposition *Asilomar Conf. on Signals, Systems and Comput.* pp 40–44 vol.1
[8] Mallat S G and Zhang Z 1993 *IEEE Transactions on Signal Processing* **41** 3397–3415
[9] Chen S S, Donoho D L and Saunders M A 1998 *SIAM J. Sci. Comput.* **20** 33–61
[10] Yaghoobi M, Blumensath T and Davies M 2009 *IEEE Transaction on Signal Processing* **57** 2178–2191
[11] Skretting K and Engan K 2010 *IEEE Transactions on Signal Processing* **58** 2121–2130
[12] Elad M and Aharon M 2006 *IEEE Trans. Image Process.* **15** 3736–3745
[13] Mairal J, Bach F, Ponce J and Sapiro G 2010 *J. Mach. Learn. Res.* **11** 19–60
[14] Mairal J, Elad M and Sapiro G 2008 *IEEE Trans. on Image Processing* **17** 53–69
[15] Aharon M and Elad M 2008 *SIAM Journal on Imaging Sciences* **1** 228–247
[16] Rubinstein R, Bruckstein A M and Elad M 2010 *Proceedings of the IEEE* **98** 1045–1057
[17] Mallat S 1999 *A Wavelet Tour of Signal Processing* (Academic Press)
[18] Ravishankar S and Bresler Y 2013 *IEEE Trans. Signal Process.* **61** 1072–1086
[19] Ravishankar S and Bresler Y 2015 Efficient blind compressed sensing using sparsifying transforms with convergence guarantees and application to magnetic resonance imaging vol 8 pp 2519–2557
[20] Pfister L and Bresler Y 2014 Adaptive sparsifying transforms for iterative tomographic reconstruction *International Conference on Image Formation in X-Ray Computed Tomography*
[21] Pfister L and Bresler Y 2015 Learning sparsifying filter banks *Proc. SPIE Wavelets & Sparsity XVI*
[22] Ravishankar S and Bresler Y 2014 *ell_{0}* sparsifying transform learning with efficient optimal updates and convergence guarantees vol 63 pp 2389–2404
[23] Wen B, Ravishankar S and Bresler Y 2015 *Int. J. Computer Vision* **114** 137–167
[24] Wen B, Ravishankar S and Bresler Y 2014 Learning overcomplete sparsifying transforms with block cosparsity *IEEE International Conference on Image Processing (ICIP)*
[25] Wersing H, Eggert J and Körner E 2003 Sparse coding with invariance constraints *Artificial Neural Networks and Neural Information ProcessingICANN/ICONIP* pp 385–392
[26] Zhan Z, Cai J, Guo D, Liu Y, Chen Z and Qu X 2015 *arXiv preprint arXiv:1503.02945*
[27] Lowe D G 1999 Object recognition from local scale-invariant features *IEEE International Conference on Computer vision (ICCV)* vol 2 pp 1150–1157
[28] Ke Y and Sukthankar R 2004 Pca-sift: A more distinctive representation for local image descriptors *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* vol 2 pp II–506
[29] Pennec E L and Mallat S 2005 *Multiscale Modeling & Simulation* **4** 992–1039
[30] Qu X, Guo D, Ning B, Hou Y, Lin Y, Cai S and Chen Z 2012 *Magnetic resonance imaging* **30** 964–977
[31] Ravishankar S and Bresler Y 2013 *IEEE Trans. Image Process.* **22** 4598–4612
[32] Zelnik-Manor L, Rosenblum K and Eldar Y 2012 *IEEE Transactions on Signal Processing* **60** 2386–2395
[33] Ravishankar S and Bresler Y 2013 Learning overcomplete sparsifying transforms for signal processing *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*

pp 3088–3092
[34] Yu G and Sapiro G 2011 *Image Processing On Line* **1**
[35] Dabov K, Foi A, Katkovnik V and Egiazarian K 2007 *IEEE Trans. on Image Processing* **16** 2080–2095
[36] Ravishankar S, Wen B and Bresler Y 2015 *IEEE Journal of Selected Topics in Signal Process.* **9** 625–636
[37] Woodbury M 1950 *Memorandum report* **42** 106
[38] The USC-SIPI Image Database [Online: http://sipi.usc.edu/database/database.php?volume=misc; accessed July-2014]
[39] Bryt O and Elad M 2008 *Journal of Visual Communication and Image Representation* **19** 270–282
[40] Yang T 1986 *Finite element structural analysis* vol 2 (Prentice Hall)
[41] Watson D 2013 *Contouring: a guide to the analysis and display of spatial data* (Elsevier)
[42] Ram I, Elad M and Cohen I 2013 *IEEE Transactions on Image Processing* **22** 2764–2774
[43] Trzasko J and Manduca A 2009 *iEEE Transactions on Medical imaging* **28** 106–121
[44] Lustig M, Donoho D and Pauly J M 2007 *Magnetic resonance in medicine* **58** 1182–1195
[45] Gleichman S and Eldar Y C 2011 *IEEE Transactions on Information Theory* **57** 6958–6975
[46] Ning B, Qu X, Guo D, Hu C and Chen Z 2013 *Magnetic resonance imaging* **31** 1611–1622
[47] Qu X, Hou Y, Lam F, Guo D, Zhong J and Chen Z 2014 *Medical image analysis* **18** 843–856
[48] Ravishankar S and Bresler Y 2011 *Medical Imaging, IEEE Transactions on* **30** 1028–1041