LEARNING OVERCOMPLETE SPARSIFYING TRANSFORMS WITH BLOCK COSPARSITY

Bihan Wen†*, Saiprasad Ravishankar*†*, and Yoram Bresler*

Department of Electrical and Computer Engineering and the Coordinated Science Laboratory,
University of Illinois at Urbana-Champaign, IL, USA

ABSTRACT

The sparsity of images in a transform domain or dictionary has been widely exploited in image processing. Compared to the synthesis dictionary model, sparse coding in the (single) transform model is cheap. However, natural images typically contain diverse textures that cannot be sparsified well by a single transform. Hence, we propose a union of sparsifying transforms model, which is equivalent to an overcomplete transform model with block cosparsity (OCTOBOS). Our alternating algorithm for transform learning involves simple closed-form updates. When applied to images, our algorithm learns a collection of well-conditioned transforms, and a good clustering of the patches or textures. Our learnt transforms provide better image representations than learned square transforms. We also show the promising denoising performance and speedups provided by the proposed method compared to synthesis dictionary-based denoising.

Index Terms— Sparsifying transform learning, Overcomplete representation, Sparse representation, Clustering, Image denoising.

1. INTRODUCTION

The sparsity of signals and images in a certain transform domain or dictionary has been heavily exploited in signal and image processing. Well-known models for sparsity include the synthesis [1, 2], analysis [2, 3], and transform models [4, 5].

The popular *synthesis model* suggests that a signal $y \in \mathbb{R}^n$ can be sparsely synthesized as $y = Dx$, where $D \in \mathbb{R}^{n \times K}$ is a synthesis dictionary and $x \in \mathbb{R}^K$ is sparse, i.e., $\|x\|_0 \ll K$. The l_0 quasi norm counts the number of non-zeros in x. Real-world signals usually satisfy $y = Dx + e$, where e is an approximation error in the signal domain [1]. The alternative *analysis model* [2] suggests that given the signal y and analysis dictionary $\Omega \in \mathbb{R}^{m \times n}$, Ωy is sparse, i.e., $\|\Omega y\|_0 \ll m$ [3]. A more general *noisy signal analysis model* [3, 6] has also been studied, where the signal y is modeled as $y = z + e$, with Ωz sparse, and e a (small) noise term in the signal domain.

In this paper, we focus instead on the sparsifying transform model [5], which suggests that a 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 is an approximation error in the transform domain (rather than signal domain) and is assumed to be small. When $m = n$, $W \in \mathbb{R}^{n \times n}$ is called a square transform. When $m > n$, the transform is said to be tall or overcomplete. Various analytical transforms are known to approximately sparsify natural signals, such as the discrete cosine transform (DCT), and Wavelets [7]. The transform model has been shown to be more general than the analysis and noisy signal analysis models (cf. [2, 5] for more details on the distinctions between the various models).

An important advantage of the transform model compared to prior sparse models is the ease of sparse coding. When a transform W is known for the signal y, *transform sparse coding* finds a sparse code x of sparsity s by minimizing $\|Wy - x\|_2^2$ subject to $\|x\|_0 \leq s$. This problem is easy and its solution is obtained exactly as $\hat{x} = H_s(Wy)$, where $H_s(\cdot)$ is the projector onto the s-ℓ_0 ball [8], i.e., $H_s(b)$ zeros out all but the s elements of largest magnitude in $b \in \mathbb{R}^m$. In contrast, sparse coding with synthesis or analysis dictionaries involves solving NP-hard problems approximately [5]. Given W and sparse code x, one can also recover an estimate of the signal y by minimizing the residual $\|Wy - x\|_2^2$ over y. The recovered signal is $\hat{y} = W^\dagger x$, with $W^\dagger$ denoting the pseudo-inverse of W [5].

Recent research has focused on the adaptation of sparse models to data [3, 5, 9–15], which turns out to be advantageous in applications. In particular, the learning of transform models has been shown to be much cheaper than synthesis, or analysis dictionary learning [5, 8]. Adaptive transforms also provide comparable or better signal reconstruction quality in applications [5, 8, 16].

In this paper, we further explore the subject of sparsifying transform learning. Given a matrix $Y \in \mathbb{R}^{n \times N}$, whose columns represent training signals, the problem of learning an adaptive (single) square sparsifying transform W is formulated as follows [5, 17]

$$\text{(P0)} \quad \min_{W,X} \|WY - X\|_F^2 + \lambda Q(W) \quad s.t. \quad \|X_i\|_0 \leq s \ \forall \ i$$

where $Q(W) = -\log |\det W| + \|W\|_F^2$. Here, the subscript i denotes the i^{th} column of the sparse code matrix X. (P0) minimizes the sparsification error given by $\|WY - X\|_F^2$. The sparsification error is the modeling error in the transform model, and hence we minimize it in order to learn the best possible transform model. Problem (P0) has $Q(W)$ as a regularizer in the objective to prevent trivial solutions [5]. Specifically, the log determinant penalty enforces full rank on W and eliminates degenerate solutions such as those with zero, or repeated rows. The $\|W\|_F^2$ penalty helps remove a 'scale ambiguity' [5] in the solution. Together, the log determinant and Frobenius norm penalty terms fully control the condition number and scaling of the learnt transform [5]. This eliminates badly conditioned transforms, which typically convey little information and may degrade performance in applications. To make the two terms in (P0) scale similarly, we set $\lambda = \lambda_0 \|Y\|_F^2$ with constant λ_0.

Although prior work in transform learning has focused on the learning of (single) square transforms, natural images need not be sufficiently sparsifiable by a single transform. For example, image patches from different regions of an image usually contain different features, or textures. Hence, in this work, we study a union-of-transforms model, and show that it can sparsify the diverse features, or textures seen in natural images much better than a single transform. For the synthesis model, learning a union of dictionaries has been studied before [18–20], but with focus on signal classification.

2. OCTOBOS MODEL AND ITS LEARNING

The union-of-transforms model suggests that a signal $y \in \mathbb{R}^n$ is approximately sparsifiable by a particular transform in the collection $\{W_k\}_{k=1}^K$, where $W_k \in \mathbb{R}^{n \times n} \ \forall \ k$, are square transforms. Thus,

† Equal contributors. This work was supported in part by the National Science Foundation (NSF) under grants CCF-1018660 and CCF-1320953.

Copyright (c) 2014 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

there exists a particular W_k such that $W_k y = x + e$, with $x \in \mathbb{R}^n$ sparse, and e small. The sparse coding problem in this model is then

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

Here, z^k denotes a sparse representation of y in the transform W_k, with maximum allowed sparsity s. We assume that the W_k's are all identically scaled in (P1). In order to solve (P1), we first find the optimal sparse code $\hat{z}^k$ for each k (the inner optimization) as $\hat{z}^k = H_s(W_k y)$. We then compute the sparsification error (using the optimal sparse code) for each k and choose the best transform $W_{\hat{k}}$ (with sparse code $\hat{z}^{\hat{k}}$) as the one that provides the smallest sparsfication error among all the W_k's. Given the sparse code [1] $\hat{z}^{\hat{k}}$, one can also recover a least squares estimate of the signal as $\hat{y} = W_{\hat{k}}^{-1} \hat{z}^{\hat{k}}$. Since Problem (P1) matches a signal y to a specific transform, it can be potentially used to cluster a collection of signals according to their transform models.

Alternatively, we can interpret the union-of-transforms model as an OverComplete TransfOrm model with BlOck coSparsity (OCTOBOS). The equivalent overcomplete transform is obtained by stacking the collection of transforms as $W = \left[W_1^T \mid W_2^T \mid ... \mid W_K^T \right]^T$. The matrix $W \in \mathbb{R}^{m \times n}$, with $m = Kn$, and thus, $m > n$ (overcomplete transform) for $K > 1$. Here, the signal y obeys $Wy = x + e$, where $x \in \mathbb{R}^m$ is "block cosparse", and e is a small residual. The block cosparsity of x is defined as $\|x\|_{0,s} = \sum_{k=1}^{K} I(\left\|x^k\right\|_0 \leq s)$, where $x^k \in \mathbb{R}^n$ is the block of x corresponding to the transform W_k in the tall W, and s is a given sparsity level. The operator $I(\cdot)$ above is an indicator function with $I(S) = 1$ when statement S is true, and $I(S) = 0$ otherwise. We say that x is p-block cosparse if there are exactly p blocks of x each with at least $n - s$ zeros, i.e., $\|x\|_{0,s} = p$. In the OCTOBOS model, the sparse coding problem is formulated as follows

$$\text{(P2)} \quad \min_{x} \; \|Wy - x\|_2^2 \;\; s.t. \;\; \|x\|_{0,s} \geq 1$$

Problem (P2) finds an x that is at least 1-block cosparse. It is easy to show that the minimum values of the sparsification errors (i.e., the objectives) in Problems (P1) and (P2) are identical. Moreover, the optimal $\hat{x}$ in (P2) satisfies $\hat{x}^k = H_s(W_k y)$ for one $k = k_0$, and $\hat{x}^k = W_k y$ for all $k \neq k_0$. The chosen k_0 is the one that provides the smallest individual sparsification (at sparsity s) error (i.e., solves (P1)). Thus, the optimal sparse code(s) in (P1) is equal to the block(s) of the optimal $\hat{x}$ in (P2) satisfying $\left\|\hat{x}^k\right\|_0 \leq s$. The full proof of this result is presented elsewhere [21]. The preceding arguments establish the equivalence between the union-of-transforms model and the corresponding OCTOBOS model.

Given the data $Y \in \mathbb{R}^{n \times N}$, we formulate the union-of-transforms, or OCTOBOS learning problem as follows

$$\text{(P3)} \min_{\{W_k\},\{X_i\},\{C_k\}} \sum_{k=1}^{K} \left\{ \sum_{i \in C_k} \|W_k Y_i - X_i\|_2^2 + \lambda_k Q(W_k) \right\} \\ s.t. \;\; \|X_i\|_0 \leq s \;\; \forall \; i, \;\; \{C_k\} \in G$$

Here, the set $\{C_k\} \triangleq \{C_k\}_{k=1}^K$ denotes a clustering of the signals $\{Y_i\}_{i=1}^N$. The cluster C_k contains the indices i corresponding to the signals Y_i in the k^{th} cluster. The set G is the set of all possible partitionings (into K disjoint subsets) of the set of integers $\{1, 2, ..., N\}$. The term $\sum_{k=1}^{K} \sum_{i \in C_k} \|W_k Y_i - X_i\|_2^2$ in (P3) is the sparsification error for Y in the OCTOBOS (or, union-of-transforms) model.

The regularizer $Q'(W) = \sum_{k=1}^{K} \lambda_k Q(W_k)$ controls the condition numbers and scalings of the square blocks W_k. The weights λ_k in (P3) are chosen as $\lambda_k = \lambda_0 \|Y_{C_k}\|_F^2$, where Y_{C_k} is a matrix whose columns are the columns of Y in the k^{th} cluster. The rationale for this choice of λ_k is similar to that presented earlier for the λ weighting in (P0). This setting also implies that λ_k itself is a function of the unknown C_k (function of the signal energy in cluster C_k) in (P3). It can be shown that (similar to Corollary 2 in [5]) as $\lambda_0 \to \infty$ in (P3), the condition numbers of the optimal transforms tend to 1, and their spectral norms (scaling) tend to $1/\sqrt{2}$.

2.1. OCTOBOS Learning Algorithm

We propose an efficient algorithm for (P3) that alternates between a sparse coding and clustering step, and a transform update step.

2.1.1. Sparse Coding and Clustering Step

Here, we solve (P3) with fixed $\{W_k\}$ to determine the $\{C_k\}$, $\{X_i\}$.

$$\text{(P4)} \quad \min_{\{C_k\},\{X_i\}} \sum_{k=1}^{K} \sum_{i \in C_k} \left\{ \|W_k Y_i - X_i\|_2^2 + \eta_k \|Y_i\|_2^2 \right\} \\ s.t. \;\; \|X_i\|_0 \leq s \;\; \forall \; i, \;\; \{C_k\} \in G$$

The weight $\eta_k = \lambda_0 Q(W_k)$ above. The term $\|W_k Y_i - X_i\|_2^2 + \eta_k \|Y_i\|_2^2$, with $X_i = H_s(W_k Y_i)$ (i.e., the optimal sparse code of Y_i in transform W_k), is the clustering measure corresponding to the signal Y_i. This is a modified version of the measure in (P1), and includes the additional penalty $\eta_k \|Y_i\|_2^2$ determined by the regularizer (i.e., determined by the conditioning of W_k [2]). It is easy to observe that the objective in (P4) involves the summation of N such 'clustering measure' terms (one for each signal). Therefore, we can construct the following equivalent optimization problem.

$$\sum_{i=1}^{N} \min_{1 \leq k \leq K} \left\{ \|W_k Y_i - H_s(W_k Y_i)\|_2^2 + \eta_k \|Y_i\|_2^2 \right\} \qquad (1)$$

The minimization over k for each Y_i above determines the cluster C_k (in (P4)) to which Y_i belongs. For each Y_i, the optimal cluster index $\hat{k}$ is the one that provides the smallest value of the clustering measure above. The optimal $\hat{X}_i$ in (P4) is then $H_s(W_{\hat{k}} Y_i)$.

2.1.2. Transform Update Step

Here, we solve for $\{W_k\}$ in (P3) with fixed $\{C_k\}$, $\{X_i\}$. This optimization problem is separable (due to the objective being in summation form) into K unconstrained problems, each involving a particular transform W_k as follows

$$\text{(P5)} \quad \min_{W_k} \sum_{i \in C_k} \|W_k Y_i - X_i\|_2^2 + \lambda_k Q(W_k)$$

where $\lambda_k = \lambda_0 \|Y_{C_k}\|_F^2$ is a fixed weight. (P5) is solved for each k similarly to the transform update step of (P0) [17]. Let $U = Y_{C_k}$, and $V = X_{C_k}$. Then, we first decompose the positive-definite matrix $UU^T + \lambda_k I_n$ as LL^T (e.g., Cholesky decomposition). Next, we obtain the full singular value decomposiiton (SVD) of the matrix $L^{-1} U V^T$ as $Q \Sigma R^T$, where Q, Σ, and R are all $n \times n$ matrices. Then, the optimal transform $\hat{W}_k$ in (P5) is

$$\hat{W}_k = \frac{R}{2} \left(\Sigma + (\Sigma^2 + 2\lambda_k I_n)^{\frac{1}{2}} \right) Q^T L^{-1} \qquad (2)$$

[1] One needs to store the index $\hat{k}$ as part of the sparse code. This adds just $\log_2 K$ bits per index to the sparse code.

[2] This clustering measure will encourage the shrinking of clusters corresponding to any badly conditioned, or badly scaled transforms.

where the $(\cdot)^{\frac{1}{2}}$ notation in (2) denotes the positive definite square root. The closed-form solution (2) is guaranteed to be a global optimum of Problem (P5) [17].

Since we solve (P3) by exact alternating minimization, our objective function is monotone decreasing over the algorithm alternations. The objective in our algorithm being monotone decreasing and lower bounded [21], it converges. The computational cost per iteration (of sparse coding and clustering, and transform update) for learning an $m \times n$ ($m = Kn$) OCTOBOS transform using our algorithm scales as $O(mnN)$. This is much lower than the per-iteration cost of learning an $n \times m$ synthesis dictionary D using K-SVD [10], which (assuming synthesis sparsity $s \propto n$) scales as $O(mn^2N)$.

2.2. Image Denoising

The goal of denoising is to recover an estimate of a 2D image represented as a vector $x \in \mathbb{R}^P$ from its measurement $y = x + h$ corrupted by noise h. Similar to the prior work [17] on adaptive square transform-based denoising, we propose the following patch-based denoising formulation that exploits the OCTOBOS model.

$$\min_{\{W_k, x_i, \alpha_i, C_k\}} \sum_{k=1}^{K} \sum_{i \in C_k} \left\{ \|W_k x_i - \alpha_i\|_2^2 + \lambda_i' Q(W_k) \right\} + \tau \sum_{i=1}^{N} \|R_i\, y - x_i\|_2^2 \quad s.t. \;\; \|\alpha_i\|_0 \leq s_i \;\; \forall \; i, \;\; \{C_k\} \in G \tag{P6}$$

Here, $R_i \in \mathbb{R}^{n \times P}$ is a patch extraction operator, i.e., $R_i y \in \mathbb{R}^n$ denotes the i$^{\text{th}}$ patch (N overlapping patches assumed) of the image y as a vector. Vector $x_i \in \mathbb{R}^n$ denotes a denoised version of $R_i y$, that satisfies the OCTOBOS model. The weight τ is chosen inversely proportional to the noise level σ [8, 11]. Vector $\alpha_i \in \mathbb{R}^n$ in (P6) denotes the sparse representation of x_i in a specific cluster transform W_k, with an apriori unknown sparsity level s_i. The weighting λ_i' is set based on the given noisy data $R_i y$ as $\lambda_0 \|R_i y\|_2^2$. The net weighting on the $Q(W_k)$ regularizer in (P6) is then $\lambda_k = \sum_{i \in C_k} \lambda_i'$, which varies depending on C_k.

Our simple iterative algorithm for (P6) involves learning on noisy patches, and additionally estimating the unknown sparsity levels s_i. Each algorithm iteration involves the following steps: *(i)* intra-cluster transform learning, *(ii)* variable sparsity level update, and *(iii)* clustering. In Step *(i)*, we solve for the cluster transforms $\{W_k\}$ and the corresponding sparse codes $\{\alpha_i\}$ in (P6), with fixed $\{x_i\}$ ($x_i = R_i y \; \forall \; i$), $\{s_i\}$, and $\{C_k\}$. This problem separates out into K independent single transform learning problems (similar to (P0)), each involving a particular W_k and the sparse codes corresponding to cluster C_k. Each of these problems is solved by alternating between sparse coding and transform update steps [17].

In Step *(ii)* of our denoising algorithm, we update only the sparsity levels s_i for all i. For each $i \in C_k$ (the k$^{\text{th}}$ cluster), the update is performed using the same efficient procedure as in [8]. We choose s_i to be the smallest integer such that the x_i in (3) below satisfies $\|R_i\, y - x_i\|_2^2 \leq nC^2\sigma^2$.

$$x_i = \begin{bmatrix} \sqrt{\tau}\, I_n \\ W_k \end{bmatrix}^{\dagger} \begin{bmatrix} \sqrt{\tau} R_i y \\ \alpha_i \end{bmatrix} = G_1 R_i y + G_2 \alpha_i \tag{3}$$

where $\alpha_i = H_{s_i}(W_k R_i y)$. The corresponding x_i's computed in (3) represent the denoised patches.

In Step *(iii)* of each iteration of our denoising algorithm, we solve (P6) with respect to the clusters $\{C_k\}$ and sparse codes $\{\alpha_i\}$, with fixed $\{s_i\}$, $\{W_k\}$, and $\{x_i\}$ ($x_i = R_i y \; \forall \; i$). This problem is similar to (P4), and is solved similarly. The denoised patches $\{x_i\}$ obtained (by (3)) in the last iteration of our aforementioned iterative scheme are restricted to their range (e.g., 0-255), and averaged at their respective locations in the image to generate the denoised image estimate.

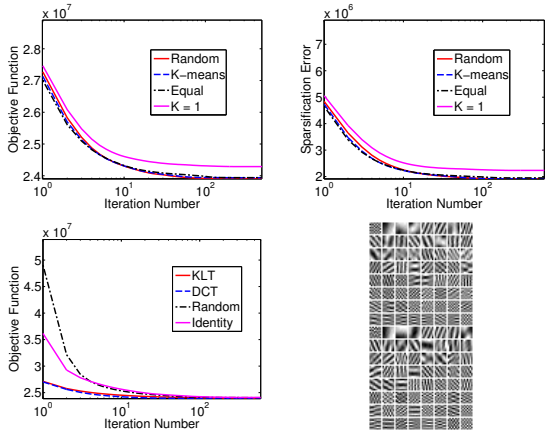

Fig. 1. Top: Objective function (left) and Sparsification error (right) with different $\{C_k\}$ initializations, along with the results for the single square transform ($K = 1$) case. Bottom: Objective function with different $\{W_k\}$ initializations (left), rows of learnt overcomplete W shown as patches for the case of KLT initialization (right).

3. NUMERICAL EXPERIMENTS

In this section, we present results demonstrating the promise of the OCTOBOS framework in applications. We work with the images Cameraman (256×256) and Barbara (512×512) in our experiments. To evaluate the quality of the learnt transforms for our approach, we compute the normalized sparsification error (NSE) as $\sum_{k=1}^{K} \sum_{i \in C_k} \|W_k Y_i - X_i\|_2^2 \; / \; \sum_{k=1}^{K} \sum_{i \in C_k} \|W_k Y_i\|_2^2$. Here, the W_k's are all normalized (e.g., to unit spectral norm), and $X_i = H_s(W_k Y_i)$. When $K = 1$, the above definition is identical to the previously proposed NSE [5] metric for a single transform. For image representation, the recovery PSNR [5] (in dB) is redefined in terms of the clusters as $255\sqrt{P} \; / \sqrt{\sum_{k=1}^{K} \sum_{i \in C_k} \left\|Y_i - W_k^{-1} X_i\right\|_2^2}$.

3.1. Convergence and Learning

We learn an OCTOBOS transform for the 8×8 non-overlapping mean-subtracted patches (as vectors) of the image Barbara, by solving (P3). We set $\lambda_0 = 3.1 \times 10^{-3}$, $s = 11$, and $K = 2$. In all our experiments, we initialize the algorithm for (P3) with the $\{C_k\}$ and $\{W_k\}$. The X_i in (P3) are then computed, and the alternating algorithm is executed beginning with the transform update step.

We illustrate the convergence of our OCTOBOS learning scheme for various initializations. First, we fix the initial $\{W_k\}$ to be the 2D DCT [5], and vary the initialization for the $\{C_k\}$ as: a) k-means, b) random clustering, and c) 'equal' clustering (clustering patches on the left half of the image to one cluster). Next, we fix the initial $\{C_k\}$ by random clustering (each patch assigned uniformly at random to one of 2 clusters), and vary the initialization for the $\{W_k\}$ as: a) 2D DCT, b) Karhunen-Loève Transforms (one KLT constructed for each cluster), c) identity, and d) random matrix with i.i.d. gaussian (zero mean, $\sigma = 0.2$) entries.

Fig. 1 shows the objective function and sparsification error converging quickly for our algorithm for the various initializations. Importantly, the final values of the objective and sparsification error

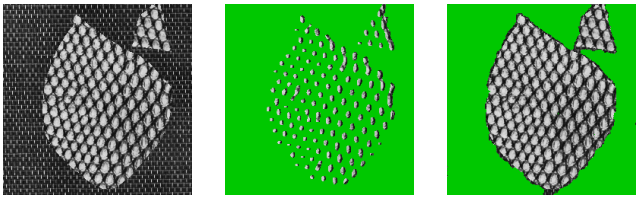

Fig. 2. Clustering example: Input image (left), Input image with pixels classified into Class 1 shown in Green for K-means (center), and OCTOBOS (right).

(the sparsification error for different transform initializations (not shown) behaves similarly) are nearly identical for all the initializations. Thus, although the proof of convergence to a global optimum is not provided for our algorithm, the result here indicates that the learning scheme is reasonably robust, or insensitive to initialization. For comparison, Fig. 1 also shows the objective and sparsification error for the single square transform learning (i.e., $K = 1$ case, with same parameters as before) algorithm [17], which converge to worse values than OCTOBOS.

Fig. 1 (bottom right image) visualizes the rows of the learnt OCTOBOS transform, for the case of KLT initialization (for the W_k's). The transform shows texture and frequency like structures, that sparsify the patches of Barbara. The learnt W_k's have similar condition numbers (≈ 1.4), and Frobenius norms (≈ 5) for all initializations.

3.2. Clustering/Classification Behavior

Here, we illustrate the potential of the proposed OCTOBOS learning for unsupervised classification. We consider the 251×249 input image shown in Fig. 2, which was formed by combining two textures from the Brodatz database [22]. We work with the 9×9 overlapping mean-subtracted patches from the input image, and employ (P3) to learn an adaptive clustering of the patches, with $s = 10$, $K = 2$, and λ_0 the same as in Section 3.1. A pixel is classified into a particular class C_k ($k = 1, 2$), if the majority of the overlapping patches that it belongs to, are clustered into that class by (P3). We initialize OCTOBOS learning using the clustering result of the k-means algorithm, and the W_k's are initialized with the DCT. Fig. 2 (center) shows the image pixels classified into each class for the k-means initialization. Fig. 2 (right) is the classification using our adaptive OCTOBOS scheme, improving over the k-means result reasonably.

3.3. Sparse Image Representation and Denoising

We first study the potential of the proposed OCTOBOS learning scheme (P3) for sparse image representation. We work with 8×8 non-overlapping mean-subtracted patches, and learn OCTOBOS transforms for the patches of Barbara, and Cameraman, at various K. For comparison, we also learn a square transform (i.e., the $K = 1$ case) for the images. We set λ_0 and s as in Section 3.1.

Table 1 lists the NSE and recovery PSNR metrics for the learnt OCTOBOS transforms, along with the corresponding values for the learnt (single) square transforms, the fixed (patch-based) 2D DCT, and KLT (constructed from all patches). The learnt transforms provide significantly better sparsification and recovery compared to the DCT and KLT. Importantly, as K increases, the learnt OCTOBOS transforms provide increasingly better image representation compared to the learnt square transform. The recovery PSNR increases monotonically, and NSE decreases likewise, as K increases.

Next, we present results for our adaptive OCTOBOS-based denoising framework (P6). We add i.i.d. Gaussian noise at 3 different noise levels to Barbara and Cameraman. For the OCTOBOS scheme, we work with 8×8 overlapping patches, and consider $K = 2,\ 4,\ 8$. We set the initial sparsity levels $s_i = 10$, $C = 1.08$ [8], $\lambda_0 = 3.1 \times 10^{-2}$, and the number of denoising algorithm iterations to 15. We execute intra-cluster transform learning for 12 iterations. Our results are compared to K-SVD denoising [3] [10, 11, 24], and to square transform denoising [17] (parameters set as in [8]).

Image	DCT	KLT	SQ	OCTOBOS			
				$K = 2$	$K = 4$	$K = 8$	$K = 16$
Cameraman	30.0	29.7	32.0	33.3	35.9	40.9	**47.8**
	9.0	9.7	4.7	3.5	1.9	0.7	**0.1**
Barbara	32.9	31.7	34.5	35.4	36.0	36.6	**38.1**
	6.8	8.9	4.3	3.5	3.0	2.6	**1.8**

Table 1. The recovery PSNR (first row for each image) and NSE (second row for each image) for the learnt OCTOBOS transforms, the learnt single square ($K = 1$) transform (SQ), DCT, and KLT, at $s = 11$. NSE is shown as a percentage.

Image	σ	K-SVD	Square	OCTOBOS		
			W	$K = 2$	$K = 4$	$K = 8$
Cameraman	5	37.81	38.05	38.06	38.08	**38.11**
	10	33.72	33.93	33.98	34.02	**34.04**
	20	29.82	29.89	29.98	**30.06**	30.05
Barbara	5	38.08	38.16	38.23	38.26	**38.30**
	10	34.41	34.37	34.47	34.56	**34.61**
	20	30.83	30.53	30.72	30.85	**30.91**

Table 2. PSNR values for denoising with OCTOBOS transforms, 64×64 square transform [17], and 64×256 K-SVD [11].

Fig. 3. Left: Noisy image ($\sigma = 20$, PSNR $= 22.10$ dB). Right: Denoised image (PSNR $= 30.06$ dB) using OCTOBOS ($K = 4$).

Table 2 lists the denoising PSNRs for the various methods. Our OCTOBOS scheme provides better PSNRs than the K-SVD, or square transform-based schemes, at all noise levels. Fig. 3 shows an example of denoising by OCTOBOS ($K = 4$). We also compute the average denoising speedups over the synthesis K-SVD at each K. For each image and noise level, the ratio of the run times of K-SVD denoising and transform denoising is computed, and these speedups are averaged over the images and noise levels (fixed K) in Table 2. The speedups for the square transform ($K = 1$), and $K = 2, 4, 8$, over KSVD are $9.8\times$, $6.6\times$, $5.4\times$, and $4.5\times$, respectively. Thus, OCTOBOS denoising is also much faster than K-SVD denoising.

4. CONCLUSIONS

In this paper, we presented a novel union-of-transforms model, and established its equivalence to an overcomplete transform with block cosparsity, termed the OCTOBOS model. Our alternating algorithm for OCTOBOS learning involving simple closed-form updates has a convergent objective, and is insensitive to initialization. The learnt OCTOBOS transforms provide better image representations than learnt single square transforms, or analytical transforms. In image denoising, the proposed scheme denoises better than adaptive square transforms, and both better and much faster than adaptive overcomplete synthesis dictionaries.

[3] K-SVD denoising has been shown [11] to perform usually better than prior Wavelets-based schemes [23].

5. REFERENCES

[1] 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.

[2] M. Elad, P. Milanfar, and R. Rubinstein, "Analysis versus synthesis in signal priors," *Inverse Problems*, vol. 23, no. 3, pp. 947–968, 2007.

[3] R. Rubinstein, T. Faktor, and M. Elad, "K-SVD dictionary-learning for the analysis sparse model," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 5405–5408.

[4] W. K. Pratt, J. Kane, and H. C. Andrews, "Hadamard transform image coding," *Proc. IEEE*, vol. 57, no. 1, pp. 58–68, 1969.

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

[6] R. Rubinstein and M. Elad, "K-SVD dictionary-learning for analysis sparse models," in *Proc. SPARS11*, June 2011, p. 73.

[7] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1999.

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

[9] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[10] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, 2006.

[11] 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.

[12] K. Skretting and K. Engan, "Recursive least squares dictionary learning algorithm," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 2121–2130, 2010.

[13] M. Yaghoobi, S. Nam, R. Gribonval, and M. Davies, "Analysis operator learning for overcomplete cosparse representations," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2011, pp. 1470–1474.

[14] B. Ophir, M. Elad, N. Bertin, and M.D. Plumbley, "Sequential minimal eigenvalues - an approach to analysis dictionary learning," in *Proc. European Signal Processing Conference (EUSIPCO)*, 2011, pp. 1465–1469.

[15] M. Yaghoobi, S. Nam, R. Gribonval, and M. E. Davies, "Constrained overcomplete analysis operator learning for cosparse signal modelling," *IEEE Trans. Signal Process.*, vol. 61, no. 9, pp. 2341–2355, 2013.

[16] L. Pfister, "Tomographic reconstruction with adaptive sparsifying transforms," M.S. thesis, University of Illinois at Urbana-Champaign, Aug. 2013.

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

[18] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR) 2010*, 2010, pp. 3501–3508.

[19] S. Kong and D. Wang, "A dictionary learning approach for classification: Separating the particularity and the commonality," in *Proceedings of the 12th European Conference on Computer Vision*, 2012, pp. 186–199.

[20] Yi-Chen Chen, C. S. Sastry, V. M. Patel, P. J. Phillips, and R. Chellappa, "Rotation invariant simultaneous clustering and dictionary learning," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 1053–1056.

[21] B. Wen, S. Ravishankar, and Y. Bresler, "Structured overcomplete sparsifying transform learning and applications," 2014, submitted. Online: `https://uofi.box.com/WenRavishankarBresler-OCTOBOS`.

[22] P. Brodatz, *Textures: a Photographic Album for Artists and Designers*, Dover, New York, 1965.

[23] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, 2003.

[24] Michael Elad, "Michael Elad personal page," `http://www.cs.technion.ac.il/~elad/Various/KSVD_Matlab_ToolBox.zip`, 2009.