Learning Stable Multilevel Dictionaries for Sparse Representation of Images

Jayaraman J. Thiagarajan, Karthikeyan Natesan Ramamurthy, and Andreas Spanias
E-mail: {jjayaram,knatesan,spanias}@asu.edu.

***Abstract*—Dictionaries adapted to the data provide superior performance when compared to predefined dictionaries in applications involving sparse representations. Algorithmic stability and generalization are desirable characteristics for dictionary learning algorithms that aim to build global dictionaries which can efficiently model any test data similar to the training samples. In this paper, we propose an algorithm to learn dictionaries for sparse representation of image patches, and prove that the proposed learning algorithm is stable and generalizable asymptotically. The algorithm employs a 1-D subspace clustering procedure, the K-lines clustering, in order to learn a hierarchical dictionary with multiple levels. Furthermore, we propose a regularized pursuit scheme for computing sparse representations using a multilevel dictionary. Using simulations with natural image patches, we demonstrate the stability and generalization characteristics of the proposed algorithm. Experiments also show that improvements in denoising performance are obtained with multilevel dictionaries when compared to global K-SVD dictionaries. Furthermore, we propose a robust variant of multilevel learning for severe degradations that occur in applications like compressive sensing. Results with random projection-based compressive recovery show that the multilevel dictionary and its robust variant provide improved performances compared to a baseline K-SVD dictionary.**

I. Introduction

A. Dictionary Learning for Sparse Representations

THE statistical structure of naturally occurring signals and images allows for their efficient representation as a sparse linear combination of patterns, such as edges, lines and other elementary features [1]. A finite collection of normalized features is referred to as a dictionary. The linear model used for general sparse coding is given by

$$\mathbf{y} = \mathbf{\Psi}\mathbf{a} + \mathbf{n}, \tag{1}$$

where $\mathbf{y} \in \mathbb{R}^M$ is the data vector and $\mathbf{\Psi} = [\boldsymbol{\psi}_1 \boldsymbol{\psi}_2 \dots \boldsymbol{\psi}_K] \in \mathbb{R}^{M \times K}$ is the dictionary. Each column of the dictionary, referred to as an atom, is a representative pattern normalized to unit ℓ_2 norm. $\mathbf{a} \in \mathbb{R}^K$ is the sparse coefficient vector and $\mathbf{n}$ is a noise vector whose elements are independent realizations from the Gaussian distribution $\mathcal{N}(0, \sigma^2)$.

The sparse coding problem can be stated as

$$\hat{\mathbf{a}} = \underset{\mathbf{a}}{\operatorname{argmin}} \|\mathbf{a}\|_0 \text{ s.t. } \|\mathbf{y} - \mathbf{\Psi}\mathbf{a}\|_2^2 \leq \epsilon, \tag{2}$$

where $\|.\|_0$ indicates the ℓ_0 norm, $\|.\|_2$ denotes the ℓ_2 norm and ϵ is the error goal for the representation. However, exact ℓ_0 minimization is a combinatorial problem and hence its convex surrogate, the ℓ_1 norm, is often used. Some of the widely used methods for computing sparse representations include the Matching Pursuit (MP) [2], Orthogonal Matching Pursuit (OMP) [3], Basis Pursuit (BP) [4], FOCUSS [5] and iterated shrinkage algorithms [6], [7]. The sparse coding model has been successfully used for inverse problems in images [8]–[10], and also in machine learning applications such as classification and clustering [11]–[21]

Predefined dictionaries obtained using the discrete cosine transform (DCT), wavelet, and curvelet [22] bases have been used successfully for image reconstruction and compression. The dictionary $\mathbf{\Psi}$ can also be designed from a union of orthonormal bases [23] or structured as an overcomplete set of individual vectors optimized to the training data [24], [25]. A wide range of dictionary learning algorithms have been proposed in the literature [26]–[32], some of which are tailored for specific applications. The conditions under which a dictionary can be identified from the training data using an ℓ_1 minimization approach are derived in [33]. The joint optimization problem for dictionary learning and sparse coding with ℓ_0 sparsity constraints can be expressed as [8], [34], [35]

$$\min_{\mathbf{\Psi},\mathbf{A}} \|\mathbf{Y} - \mathbf{\Psi}\mathbf{A}\|_F^2 \text{ s.t. } \|\mathbf{a}_i\|_0 \leq S, \forall i, \|\boldsymbol{\psi}_j\|_2 = 1, \forall j, \tag{3}$$

where $\mathbf{Y} = [\mathbf{y}_1 \mathbf{y}_2 \dots \mathbf{y}_T]$ is a collection of T training vectors, $\mathbf{A} = [\mathbf{a}_1 \mathbf{a}_2 \dots \mathbf{a}_T]$ is the coefficient matrix, S is the sparsity of the coefficient vector and $\|.\|_F$ denotes the Frobenius norm. Learned dictionaries have been successfully applied to image compression, denoising and inpainting [9], [10].

In this paper, we propose a stable and generalizable learning algorithm for designing multilevel dictionaries that are particularly suited for sparse approximation of natural images. A simple example of learning a dictionary with two levels is demonstrated in Figure 1. The properties and performance of this learning algorithm will be analyzed in detail in this paper. The multilevel dictionary (MLD) learning algorithm is a hierarchical procedure where the dictionary atoms in each level are obtained using a 1-D subspace clustering algorithm, which we refer to as *K-lines clustering* [36] [1]. The proposed algorithm builds global dictionaries using a set of randomly chosen training patches obtained from a large collection of natural images that can generalize well to any test set of patches. For a learned dictionary to provide a good approximation, the test data must be similar to the data samples used

The authors are with the SenSIP Center, School of Electrical, Computer and Energy Engineering, Arizona State University, Tempe, AZ, 85287.

[1]Note that in the papers [36]–[38] the procedure has been referred to as K-hyperline clustering. But in this paper, we prefer to use the term K-lines clustering, since a 1-D subspace in any number of dimensions is referred only to as a line, and not as a hyperline.

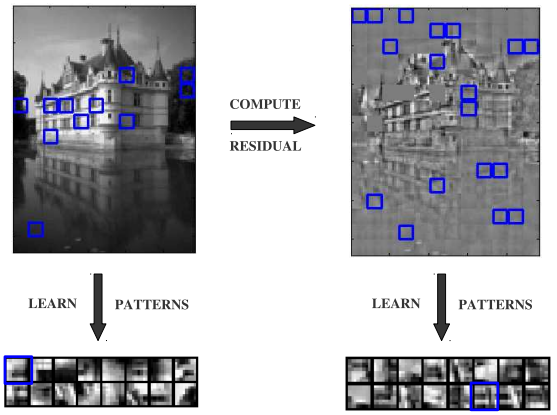


Fig. 1. Features learned at two levels from non-overlapping patches (8×8) of a 128×96 image. In each level, the patches that are highlighted in the image share similar information and hence jointly correspond to a learned pattern (also highlighted).

for training. Since local regions of natural images have high redundancy and consistent statistical properties [39], learning global dictionaries from a random collection of natural image patches will provide a good representation for patches from images not in the training set. The effectiveness of such dictionaries have been demonstrated in denoising [9] and compressed recovery [40].

B. Stability and Generalization in Learning

A learning algorithm is a map from the space of training examples to the hypothesis space of functional solutions. Algorithmic stability characterizes the behavior of a learning algorithm with respect to the perturbations of its training set [41], and generalization ensures that the expected error of the learned function with respect to the novel test data will be close to the average empirical training error [42]. In clustering, the learned function is completely characterized by the cluster centers. Stability of a clustering algorithm implies that the cluster centroids learned by the algorithm are not significantly different when different sets of i.i.d. samples from the same probability space are used for training [43]. When there is a unique minimizer to the clustering objective with respect to the underlying data distribution, stability of a clustering algorithm is guaranteed [44] and this analysis has been extended to characterize the stability of K-means clustering in terms of the number of minimizers [45]. In [38], the stability properties of the K-lines clustering algorithm have been analyzed and they have been shown to be similar to those of K-means clustering. Note that all the stability characterizations depend only on the underlying data distribution and the number of clusters, and not on the actual training data itself. Generalization implies that the average empirical training error becomes asymptotically close to the expected error with respect to the probability space of data, as the number of training samples $T \to \infty$. In [46], the generalization bound for sparse coding in terms of the number of samples T, also referred to as sample complexity, is derived and in [47] the bound is improved by assuming a class of dictionaries that are nearly orthogonal.

The algorithmic stability of dictionary learning methods has not been discussed in the literature until now, to the best of our knowledge. Given a sufficiently large training set, a stable learning algorithm will result in global dictionaries that will depend only on the probability space to which the training samples belong and not on the actual samples themselves. Generalization ensures that such global dictionaries learned result in a good performance with test data. In other words, the asymptotic stability and generalization of a dictionary learning algorithm provide a theoretical justification for the uniformly good performance of global dictionaries learned from an arbitrary training set.

C. Contributions

In this paper, we propose the MLD learning algorithm to design global representative dictionaries for image patches. We show that, for a sufficient number of levels, the proposed algorithm converges, and also demonstrate that a multilevel dictionary with a sufficient number of atoms per level exhibits energy hierarchy (Section III-C). Furthermore, we develop a Regularized Multilevel OMP (RM-OMP) procedure for computing sparse codes for test data using the proposed dictionary (Section III-D). Some preliminary algorithmic details and results obtained using MLD have been reported in [37].

Using the fact that the K-lines clustering algorithm is stable, we perform stability analysis of the MLD algorithm. For any two sets of i.i.d. training samples from the same probability space, as the number of training samples $T \to \infty$, we show that the dictionaries learned become close to each other asymptotically. When there is a unique minimizer to the objective in each level of learning, this holds true even if the training sets are completely disjoint. However, when there are multiple minimizers for the objective in at least one level, we prove that the learned dictionaries are asymptotically close when the difference between their corresponding training sets is $o(\sqrt{T})$. Instability of the algorithm when the difference between two training sets is $\Omega(\sqrt{T})$, is also shown for the case of multiple minimizers (Section IV-C). Furthermore, we prove the asymptotic generalization of the learning algorithm (Section IV-D).

The stability characteristics of MLD learning are experimentally demonstrated using natural image data (Section IV-E). We show that, the stability in terms of the learned dictionaries improves as the difference between their corresponding training sets becomes small and as the number of training samples increases. We train a global multilevel dictionary from a set of patches chosen randomly from a corpus of natural images and study its generalization behavior using several simulations. For comparison, we use a dictionary learned using the K-SVD algorithm, with similar training parameters, for the same training data set. We observe that the error in sparse approximation for the training and test data sets become comparable as the size of the training set increases. When compared to the K-SVD, the proposed algorithm exhibits much improved generalization by providing reduced test error even with a small number of training samples. The learned MLD results in a better denoising performance compared to

the global K-SVD dictionary (Section V). In order to improve recovery performance with severe degradations such as compressive sensing, we also propose a robust MLD (RMLD) procedure that uses multiple random subsets of data to obtain approximations in each level. Using compressive recovery of randomly projected data, we show that the RMLD improves over MLD, which in turn performs better than a baseline K-SVD dictionary (Section VI).

II. Background

In this section, we describe the K-lines clustering, a 1-D subspace clustering procedure proposed in [36], which forms a building block of the proposed dictionary learning algorithm. Furthermore, we briefly discuss the results for stability analysis of K-means and K-lines algorithms reported in [43] and [38] respectively. The ideas described in this section will be used in Section IV to study the stability characteristics of the proposed dictionary learning procedure.

A. K-lines Clustering Algorithm

The K-lines clustering algorithm is an iterative procedure that performs a least squares fit of K 1-D linear subspaces to the training data [36]. Note that the K-lines clustering is a special case of general subspace clustering methods proposed in [48]–[50], when the subspaces are $1-$dimensional and constrained to pass through the origin. In contrast with K-means, K-lines clustering allows each data sample to have an arbitrary coefficient value corresponding to the centroid of the cluster it belongs to. Furthermore, the cluster centroids are normalized to unit ℓ_2 norm. Given the set of T data samples $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^T$ and the number of clusters K, K-lines clustering proceeds in two stages after initialization: the cluster assignment and the cluster centroid update. In the cluster assignment stage, training vector $\mathbf{y}_i$ is assigned to a cluster j based on the minimum distortion criteria, $\mathcal{H}(\mathbf{y}_i) = \text{argmin}_j\, d(\mathbf{y}_i, \boldsymbol{\psi}_j)$, where the distortion measure is

$$d(\mathbf{y}, \boldsymbol{\psi}) = \|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y}^T\boldsymbol{\psi})\|_2^2. \tag{4}$$

In the cluster centroid update stage, we perform singular value decomposition (SVD) of $\mathbf{Y}_j = [\mathbf{y}_i]_{i\in\mathcal{C}_j}$, where $\mathcal{C}_j = \{i|\mathcal{H}(\mathbf{y}_i) = j\}$ contains indices of training vectors assigned to the cluster j. The left singular vector corresponding to the largest singular value of the decomposition, is the centroid of cluster j. Different strategies exist for initialization of cluster centroids and estimation of the number of hyperlines [36].

B. Stability Analysis of Clustering Algorithms

Analyzing the stability of unsupervised clustering algorithms can be valuable in terms of understanding their behavior with respect to perturbations in the training set. These algorithms extract the underlying structure in the training data and the quality of clustering is determined by an accompanying cost function. As a result, any clustering algorithm can be posed as a Empirical Risk Minimization (ERM) procedure, by defining a hypothesis class of loss functions to evaluate the possible cluster configurations and to measure their quality [51]. For example, K-lines clustering can be posed as an ERM problem over the distortion function class

$$\mathcal{G}_K = \left\{ g_{\boldsymbol{\Psi}}(\mathbf{y}) = d(\mathbf{y}, \boldsymbol{\psi}_j), j = \underset{l\in\{1,\cdots,K\}}{\text{argmax}} |\mathbf{y}^T\boldsymbol{\psi}_l| \right\}. \tag{5}$$

The class $\mathcal{G}_K$ is obtained by taking functions $g_{\boldsymbol{\Psi}}$ corresponding to all possible combinations of K unit length vectors from the $\mathbb{R}^M$ space for the set $\boldsymbol{\Psi}$. Let us define the probability space for the data in $\mathbb{R}^M$ as $(\mathcal{Y}, \boldsymbol{\Sigma}, P)$, where $\mathcal{Y}$ is the sample space and $\boldsymbol{\Sigma}$ is a sigma-algebra on $\mathcal{Y}$, i.e., the collection of subsets of $\mathcal{Y}$ over which the probability measure P is defined. The training samples, $\{\mathbf{y}_i\}_{i=1}^T$, are i.i.d. realizations from this space.

Ideally, we are interested in computing the cluster centroids $\hat{\boldsymbol{\Psi}}$ that minimize the expected distortion $\mathbb{E}[g_{\boldsymbol{\Psi}}]$ with respect to the probability measure P. However, the underlying distribution of the data samples is not known and hence we resort to minimizing the average empirical distortion with respect to the training samples $\{\mathbf{y}_i\}_{i=1}^T$ as

$$g_{\hat{\boldsymbol{\Psi}}} = \underset{g\in\mathcal{G}_K}{\text{argmin}} \frac{1}{T}\sum_{i=1}^{T} g_{\boldsymbol{\Psi}}(\mathbf{y}_i). \tag{6}$$

When the empirical averages of the distortion functions in $\mathcal{G}_K$ uniformly converge to the expected values over all probability measures P,

$$\lim_{T\to\infty} \sup_P \mathbb{P}\left(\sup_{g_{\boldsymbol{\Psi}}\in\mathcal{G}_K} \left| \mathbb{E}[g_{\boldsymbol{\Psi}}] - \frac{1}{T}\sum_{i=1}^{T} g_{\boldsymbol{\Psi}}(\mathbf{y}_i) \right| > \delta \right) = 0, \tag{7}$$

for any $\delta > 0$, we refer to the class $\mathcal{G}_K$ as uniform Glivenko-Cantelli (uGC). In addition to being uGC, if the class also satisfies a version of the central limit theorem, it is defined as uniform Donsker [41]. In order to determine if $\mathcal{G}_K$ is uniform Donsker, we have to verify if the covering number of $\mathcal{G}_K$ with respect to the supremum norm, $N_\infty(\gamma, \mathcal{G}_K)$, grows polynomially in the dimensions M [43]. Here, γ denotes the maximum L_∞ distance between an arbitrary distortion function in $\mathcal{G}_K$, and the function that covers it. For K-lines clustering, the covering number is upper bounded by [38, Lemma 2.1]

$$N_\infty(\gamma, \mathcal{G}_K) \leq \left(\frac{8R^3K + \gamma}{\gamma} \right)^{MK}, \tag{8}$$

where we assume that the data lies in an M-dimensional ℓ_2 ball of radius R centered at the origin. Therefore, $\mathcal{G}_K$ belongs to the uniform Donsker class.

The general idea behind stability of a clustering algorithm is that the algorithm should produce cluster centroids that are not significantly different when different i.i.d. training sets from the same probability space are used for training [43]–[45]. Stability is characterized based on the number of minimizers to the clustering objective with respect to the underlying data distribution. A minimizer corresponds to a function $g_{\boldsymbol{\Psi}} \in \mathcal{G}_K$ with the minimum expectation $\mathbb{E}[g_{\boldsymbol{\Psi}}]$. Stability analysis of the K-means algorithm has been reported in [43], [45].

Though the geometry of K-lines clustering is different from that of K-means, the stability characteristics of the two

clustering algorithms have been found to be similar [38]. Given two sets of cluster centroids $\mathbf{\Psi} = \{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_K\}$ and $\mathbf{\Lambda} = \{\boldsymbol{\lambda}_1, \ldots, \boldsymbol{\lambda}_K\}$ learned from training sets of T i.i.d. samples each realized from the same probability space, let us define the $L_1(P)$ distance between the corresponding clusterings as

$$\|g_{\mathbf{\Psi}} - g_{\mathbf{\Lambda}}\|_{L_1(P)} = \int |g_{\mathbf{\Psi}}(\mathbf{y}) - g_{\mathbf{\Lambda}}(\mathbf{y})| dP(\mathbf{y}). \quad (9)$$

When $T \to \infty$, and $\mathcal{G}_K$ is uniform Donsker, stability in terms of the distortion functions is expressed as

$$\|g_{\mathbf{\Psi}} - g_{\mathbf{\Lambda}}\|_{L_1(P)} \xrightarrow{P} 0, \quad (10)$$

where $\xrightarrow{P}$ denotes convergence in probability. This holds true even for $\mathbf{\Psi}$ and $\mathbf{\Lambda}$ learned from completely disjoint training sets, when there is a unique minimizer to the clustering objective. When there are multiple minimizers, (10) holds true with respect to a change in $o(\sqrt{T})$ samples between two training sets and fails to hold with respect to a change in $\Omega(\sqrt{T})$ samples [38]. The distance between the cluster centroids themselves is defined as [43]

$$\Delta(\mathbf{\Psi}, \mathbf{\Lambda}) = \max_{1 \le j \le K} \min_{1 \le l \le K} \left[(d(\boldsymbol{\psi}_j, \boldsymbol{\lambda}_l))^{1/2} + (d(\boldsymbol{\psi}_l, \boldsymbol{\lambda}_j))^{1/2} \right]. \quad (11)$$

Lemma 2.1 ([38]): If the $L_1(P)$ distance between the distortion functions for the clusterings $\mathbf{\Psi}$ and $\mathbf{\Lambda}$ is bounded as $\|g_{\mathbf{\Psi}} - g_{\mathbf{\Lambda}}\|_{L_1(P)} < \mu$, for some $\mu > 0$, and $dP(\mathbf{y})/d\mathbf{y} > C$, for some $C > 0$, then $\Delta(\mathbf{\Psi}, \mathbf{\Lambda}) \le 2\sin(\rho)$ where

$$\rho \le 2 \sin^{-1} \left[\frac{1}{r} \left(\frac{\mu}{\hat{C}_{C,M}} \right)^{\frac{1}{M+1}} \right]. \quad (12)$$

Here the training data is assumed to lie outside an M-dimensional ℓ_2 ball of radius r centered at the origin, and the constant $\hat{C}_{C,M}$ depends only on C and M.

When the clustering algorithm is stable according to (10), for admissible values of r, Lemma 2.1 indicates that the cluster centroids become arbitrarily close to each other, $\Delta(\mathbf{\Psi}, \mathbf{\Lambda}) \xrightarrow{P} 0$, which implies stability in terms of cluster centroids. From (12), it is also clear that the K-lines clustering cannot be stable if some training vectors have a norm close enough to 0, (i.e.) $r \to 0$.

III. Multilevel Dictionary Learning

In this section, we motivate and develop a multilevel dictionary learning approach for sparse representations, whose algorithmic stability and generalizability will be proved in Section IV. Furthermore, we propose the RM-OMP algorithm, that can be used to obtain sparse codes for a test image using the multilevel dictionary.

A. Motivation for Multilevel Learning

Our motivation for learning an MLD is two-fold. Firstly we require a global dictionary that can exploit, (a) the redundancy observed across local regions in natural images and, (b) the hierarchy of patterns found in training image patches. Secondly, the learning procedure must be provably stable, with respect to the notion of algorithmic stability, and generalizable.

The generative model in (1) is well suited for natural signals and images as they can be represented using a sparse linear combination of elementary features chosen from a dictionary [24]. The redundancy in the local regions of natural images [39] allows for the design of global dictionaries that can generalize well to a wide range of images. Global dictionaries learned from a set of randomly chosen patches from natural images have been successfully used for denoising [9], compressed sensing [40] and classification [52]. In addition to exhibiting redundancy, the natural image patches typically contain either geometric patterns or stochastic textures or a combination of both. This fact is demonstrated in [53], where the authors define two types of atomic subspaces to model image patches: subspaces of low dimensions (explicit manifolds) for primitive geometric patterns and subspaces of high dimensions (implicit manifolds) for stochastic textures. Since the image patches can contain both geometric and stochastic structures, a hybrid combination of explicit and implicit manifolds can be used for modeling them [53]. The proposed MLD algorithm learns global representative patterns in multiple levels, according to the order of their energy contribution. Since the geometric patterns usually are of higher energy when compared to stochastic textures in images, geometric patterns are learned in the first few levels and stochastic textures are learned in the last few levels.

Considering the dictionary learning formulation in (3), it can be seen that clustering algorithms such as the K-means and the K-lines can be obtained by constraining the desired sparsity to be 1. Since the stability characteristics of clustering algorithms are well understood, employing similar tools to analyze the more general dictionary learning can be beneficial. Note that the proposed algorithm poses dictionary learning as performing K-lines clustering in multiple levels and hence in this case we can use the stability characteristics of the clustering algorithm to study the stability of multilevel learning. Furthermore, by exploiting the fact that the distortion function class for each level of learning is uniform Donsker, the generalizability of the algorithm can also be proved. Note that multilevel learning is different from the work in [54], where multiple sub-dictionaries are designed and one of them is chosen for representing a group of patches.

B. Proposed MLD Learning Algorithm

We denote the MLD as $\mathbf{\Psi} = [\mathbf{\Psi}_1 \mathbf{\Psi}_2 ... \mathbf{\Psi}_L]$, and the coefficient matrix as $\mathbf{A} = [\mathbf{A}_1^T \mathbf{A}_2^T ... \mathbf{A}_L^T]^T$. Here, $\mathbf{\Psi}_l$ is the sub-dictionary and $\mathbf{A}_l$ is the coefficient matrix for level l. The approximation in level l is expressed as

$$\mathbf{R}_{l-1} = \mathbf{\Psi}_l \mathbf{A}_l + \mathbf{R}_l, \text{ for } l = 1, ..., L, \quad (13)$$

where $\mathbf{R}_{l-1}$, $\mathbf{R}_l$ are the residuals for the levels $l-1$ and l respectively and $\mathbf{R}_0 = \mathbf{Y}$, the matrix of training image patches. This implies that the residual matrix in level $l-1$ serves as the training data for level l. Note that the sparsity of the representation in each level is fixed at 1. Hence, the

TABLE I
ALGORITHM FOR BUILDING A MULTILEVEL DICTIONARY.

Input
$\mathbf{Y} = [\mathbf{y}_i]_{i=1}^T$, $M \times T$ matrix of training vectors.
L, maximum number of levels of the dictionary.
K_l, number of dictionary elements in level l, $l = \{1, 2, ..., L\}$.
ϵ, error goal of the representation.

Output
$\mathbf{\Psi}_l$, adapted sub-dictionary for level l.

Algorithm
Initialize: $l = 1$ and $\mathbf{R}_0 = \mathbf{Y}$.
$\Lambda_0 = \{i \mid \|\mathbf{r}_{0,i}\|_2^2 > \epsilon, 1 \leq i \leq T\}$, index of training vectors with squared norm greater than error goal.
$\hat{\mathbf{R}}_0 = [\mathbf{r}_{0,i}]_{i \in \Lambda_0}$.

while $\Lambda_{l-1} \neq \emptyset$ and $l \leq L$
  Initialize:
    $\mathbf{A}_l$, coefficient matrix, size $K_l \times M$, all zeros.
    $\mathbf{R}_l$, residual matrix for level l, size $M \times T$, all zeros.
  $\{\mathbf{\Psi}_l, \hat{\mathbf{A}}_l\} = \text{KLC}(\hat{\mathbf{R}}_{l-1}, K_l)$.
  $\mathbf{R}_l^t = \hat{\mathbf{R}}_{l-1} - \mathbf{\Psi}_l \hat{\mathbf{A}}_l$.
  $\mathbf{r}_{l,i} = \mathbf{r}_{l,j}^t$ where $i = \Lambda_{l-1}(j)$, $\forall j = 1, ..., |\Lambda_{l-1}|$.
  $\mathbf{a}_{l,i} = \hat{\mathbf{a}}_{l,j}$ where $i = \Lambda_{l-1}(j)$, $\forall j = 1, ..., |\Lambda_{l-1}|$.
  $\Lambda_l = \{i \mid \|\mathbf{r}_{l,i}\|_2^2 > \epsilon, 1 \leq i \leq T\}$.
  $\hat{\mathbf{R}}_l = [\mathbf{r}_{l,i}]_{i \in \Lambda_l}$.
  $l \leftarrow l + 1$.
end

overall approximation for all levels is

$$\mathbf{Y} = \sum_{l=1}^{L} \mathbf{\Psi}_l \mathbf{A}_l + \mathbf{R}_L. \tag{14}$$

MLD learning can be interpreted as a block-based dictionary learning problem with unit sparsity per block, where the sub-dictionary in each block can allow only a 1-sparse representation and each block corresponds to a level. The sub-dictionary for level l, $\mathbf{\Psi}_l$, is the set of cluster centroids learned from the training matrix for that level, $\mathbf{R}_{l-1}$, using K-lines clustering. MLD learning can be formally stated as an optimization problem that proceeds from the first level until the stopping criteria is reached. For level l, the optimization problem is

$$\underset{\mathbf{\Psi}_l, \mathbf{A}_l}{\text{argmin}} \|\mathbf{R}_{l-1} - \mathbf{\Psi}_l \mathbf{A}_l\|_F^2 \text{ subject to } \|\mathbf{a}_{l,i}\|_0 \leq 1, \quad \text{for } i = \{1, ..., T\}, \tag{15}$$

along with the constraint that the columns of $\mathbf{\Psi}_l$ have unit ℓ_2 norm, where $\mathbf{a}_{l,i}$ is the i^{th} column of $\mathbf{A}_l$ and T is the number of columns in $\mathbf{A}_l$. We adopt the notation $\{\mathbf{\Psi}_l, \mathbf{A}_l\} = \text{KLC}(\mathbf{R}_{l-1}, K_l)$ to denote the problem in (15) where K_l is the number of atoms in the sub-dictionary $\mathbf{\Psi}_l$. The stopping criteria is provided either by imposing a limit on the residual representation error or the maximum number of levels (L). Note that the total number of levels is the same as the maximum number of non-zero coefficients (sparsity) of the representation. The error constraint can be stated as, $\|\mathbf{r}_{l,i}\|_2^2 \leq \epsilon, \forall i = 1, ..., T$ for some level l, where ϵ is the error goal.

Table I lists the MLD learning algorithm with sparsity and error constraints. We use the notation $\Lambda_l(j)$ to denote the j^{th} element of the set Λ_l and $\mathbf{r}_{l,i}$ denotes the i^{th} column vector in the matrix $\mathbf{R}_l$. The set Λ_l contains the indices of the residual vectors of level l whose norm is greater than the error goal. The residual vectors indexed by Λ_l are stacked in the matrix, $\hat{\mathbf{R}}_l$, which in turn serves as the training matrix for the next level, $l+1$. In MLD learning, for a given level l, the residual $\mathbf{r}_{l,i}$ is orthogonal to the representation $\mathbf{\Psi}_l \mathbf{a}_{l,i}$. This implies that

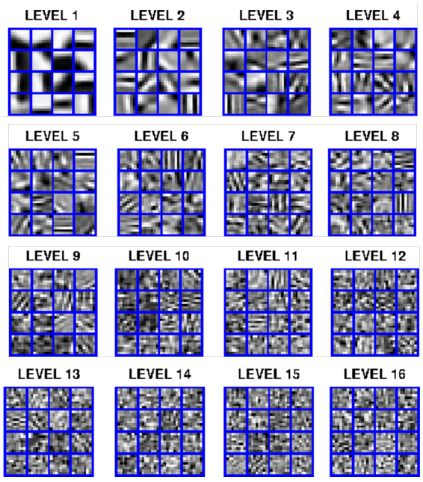


Fig. 2. Multilevel dictionary, with 16 levels of 16 atoms each, comprises of geometric patterns in the first few levels, stochastic textures in the last few levels and a combination of both in the middle levels.

$$\|\mathbf{r}_{l-1,i}\|_2^2 = \|\mathbf{\Psi}_l \mathbf{a}_{l,i}\|_2^2 + \|\mathbf{r}_{l,i}\|_2^2. \tag{16}$$

Combining this with the fact that $\mathbf{y}_i = \sum_{l=1}^{L} \mathbf{\Psi}_l \mathbf{a}_{l,i} + \mathbf{r}_{L,i}$, $\mathbf{a}_{l,i}$ is $1-$sparse, and the columns of $\mathbf{\Psi}_l$ are of unit ℓ_2 norm, we obtain the relation

$$\|\mathbf{y}_i\|_2^2 = \sum_{l=1}^{L} \|\mathbf{a}_{l,i}\|_2^2 + \|\mathbf{r}_{L,i}\|_2^2. \tag{17}$$

Equation (17) states that the energy of any training vector is equal to the sum of squares of its coefficients and the energy of its residual. From (16), we also have that,

$$\|\mathbf{R}_{l-1}\|_F^2 = \|\mathbf{\Psi}_l \mathbf{A}_l\|_F^2 + \|\mathbf{R}_l\|_F^2. \tag{18}$$

In our implementation of MLD learning, we include an additional step where the residual at each level is orthogonalized to the dictionary atoms chosen so far, and the coefficients are recomputed. Note that this does not affect any other behavior of the algorithm that is discussed in this section.

The training vectors for the first level of the algorithm, $\mathbf{r}_{0,i}$ lie in the ambient $\mathbb{R}^M$ space and the residuals, $\mathbf{r}_{1,i}$, lie in a finite union of $\mathbb{R}^{M-1}$ subspaces. This is because, for each dictionary atom in the first level, its residual lies in an $M-1$ dimensional space orthogonal to it. In the second level, the dictionary atoms can possibly lie anywhere in $\mathbb{R}^M$, and hence the residuals can lie in a finite union of $\mathbb{R}^{M-1}$ and $\mathbb{R}^{M-2}$ dimensional subspaces. Hence, we can generalize that the dictionary atoms for all levels lie in $\mathbb{R}^M$, whereas the training vectors of level $l \geq 2$, lie in finite unions of $\mathbb{R}^{M-1}, \ldots, \mathbb{R}^{M-l+1}$ dimensional subspaces of the $\mathbb{R}^M$ space.

C. Convergence

The convergence of MLD learning and the energy hierarchy in the representation obtained using an MLD can be shown by providing two guarantees. The first guarantee is that for a fixed number of atoms per level, the algorithm will converge to the required error within a sufficient number of levels. This is because the K-lines clustering makes the residual energy of the representation smaller than the energy of the training matrix at each level (i.e.) $\|\mathbf{R}_l\|_F^2 < \|\mathbf{R}_{l-1}\|_F^2$. This follows from (18) and the fact that $\|\mathbf{\Psi}_l\mathbf{A}_l\|_F^2 > 0$.

The second guarantee is that for a sufficient number of atoms per level, the representation energy in level $l-1$ will be less than the representation energy in level l. To show this, we first state that for a sufficient number of dictionary atoms per level, $\|\mathbf{\Psi}_l\mathbf{A}_l\|_F^2 > \|\mathbf{R}_l\|_F^2$. This means that for every l

$$\|\mathbf{R}_l\|_F^2 < \|\mathbf{\Psi}_l\mathbf{A}_l\|_F^2 < \|\mathbf{R}_{l-1}\|_F^2, \tag{19}$$

because of (18). This implies that $\|\mathbf{\Psi}_l\mathbf{A}_l\|_F^2 < \|\mathbf{\Psi}_{l-1}\mathbf{A}_{l-1}\|_F^2$, i.e., the energy of the representation in each level reduces progressively from $l=1$ to $l=L$.

D. Sparse Approximation using an MLD

In order to compute sparse codes for novel test data using a multilevel dictionary, we propose to perform reconstruction using a *Multilevel Orthogonal Matching Pursuit* (M-OMP) procedure which evaluates a 1-sparse representation for each level using the dictionary atoms from that level, and orthogonalizes the residual to the dictionary atoms chosen so far. Though asymptotic generalization of the M-OMP method will be shown in Section IV-D, imposing the energy hierarchy observed in the training process to any test data might result in poor generalization. Hence, there is a need to regularize this procedure such that there is more flexibility in choosing dictionary atoms for representing the test data. Hence, we propose to build a sub-dictionary with atoms selected from the current level as well as the u immediately preceding and following levels, $\tilde{\mathbf{\Phi}}_l = \left[\mathbf{\Phi}_{l-u}\mathbf{\Phi}_{l-(u-1)}\dots\mathbf{\Phi}_l\dots\mathbf{\Phi}_{l+(u-1)}\mathbf{\Phi}_{l+u}\right]$, in every step of the pursuit algorithm. In our implementation, we fix $u=2$ and also reduce the size of the sub-dictionary appropriately when $l \leq u$ and $l > L-u$. The dictionary $\tilde{\mathbf{\Phi}}_l$ is used to compute a 1-sparse representation for that step of the pursuit. It was observed from simulations that the RM-OMP scheme performs better than M-OMP, particularly when the training set is small.

E. Demonstration of MLD Learning

All simulation results presented in this paper were obtained with dictionaries learned using randomly chosen patches of size 8×8, extracted from the grayscale images in the training set of the Berkeley segmentation dataset (BSDS) [55]. The number of grayscale patches used for training will be clearly stated for each simulation. As a preprocessing step, the mean value of each training patch was removed. In this section, we will demonstrate the characteristics of MLD learning using an example dictionary learned using $50,000$ patches. Note that, the number of atoms was fixed at 16 per level and the number of levels was fixed at 16, which leads to a total of 256 atoms. For comparison, a global K-SVD dictionary of size 64×256 atoms was learned, with the same training set, using the MATLAB toolbox available online [56]. In this case, the desired sparsity, which refers to the number of non-zero coefficients (S), was fixed at 16. Initial dictionary atoms for the K-SVD algorithm and for each level of MLD learning were obtained using the K-means clustering procedure.

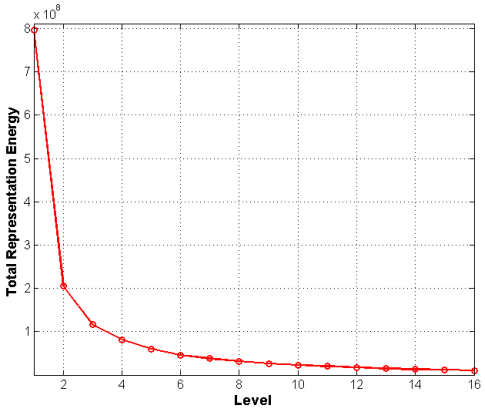


Fig. 3. Levelwise representation energy for the learned MLD with the BSDS training data set

Figure 2 illustrates the multilevel dictionary designed using the algorithm in Table I. Note that no noise was added to the image patches during learning. As it can be observed, the learned MLD contains geometric patterns in the first few levels, stochastic textures in the last few levels and a combination of both in the middle levels. The representation energy, $\|\mathbf{\Psi}_l\mathbf{A}_l\|_F^2$, captured across all the levels in MLD is shown in Figure 3, where the energy hierarchy in learning can be clearly seen.

Given a multilevel dictionary, an $S-$sparse representation for a test sample can be evaluated using the M-OMP or the RM-OMP procedures described in Section III-D. For the learned K-SVD and multilevel dictionaries, we computed the sparse codes for patches from a test dataset, by varying the desired sparsity. The test dataset consisted of $120,000$ non-overlapping 8×8 patches extracted from images in the BSDS test images. The illustration in Figure 4 shows the mean squared error (MSE) of the representation as a function of the number of non-zero coefficients. For the case of MLD, the results obtained using both the M-OMP and the RM-OMP schemes are shown. The OMP algorithm was employed to compute the sparse coefficients with the K-SVD dictionary. It can be observed that the MSE obtained using the M-OMP procedure is higher in all cases of sparsity, when compared to RM-OMP. Since the RM-OMP procedure considers dictionary atoms from the neighboring levels when computing a coefficient, it results in an improved generalization. When compared to K-SVD, multilevel dictionaries lead to a more accurate reconstruction when the sparsity level $S \geq 4$, which is the range typically used in several applications.

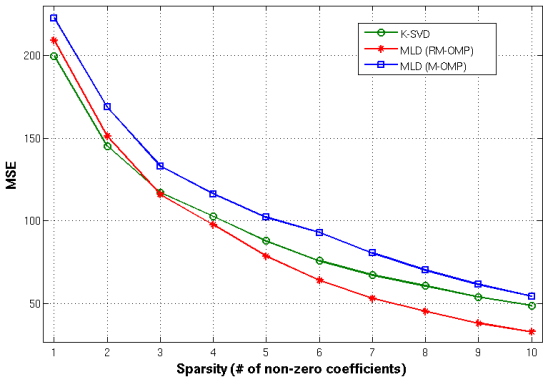


Fig. 4. Comparison of the MSE obtained with the BSDS test dataset using the K-SVD and the MLD dictionaries at different levels of sparsity.

IV. Stability and Generalization

In this section, the behavior of the proposed dictionary learning algorithm is considered from the viewpoint of algorithmic stability: the behavior of the algorithm with respect to the perturbations in the training set. It will be shown that the dictionary atoms learned by the algorithm from two different training sets whose samples are realized from the same probability space, become arbitrarily close to each other, as the number of training samples $T \rightarrow \infty$. Since the proposed MLD learning is equivalent to learning K-lines cluster centroids in multiple levels, the stability analysis of K-lines clustering [38], briefly discussed in Section II-B, will be utilized in order to prove its stability. For each level of learning, the cases of single and multiple minimizers to the clustering objective will be considered. Proving that the learning algorithm is stable will show that the global dictionaries learned from the data depend only on the probability space to which the training samples belong and not on the actual samples themselves, as $T \rightarrow \infty$. We also show that the MLD learning generalizes asymptotically, i.e., the difference between expected error and average empirical error in training approaches zero, as $T \rightarrow \infty$. Therefore, the expected error for novel test data, drawn from the same distribution as the training data, will be close to the average empirical training error.

The stability analysis of the MLD algorithm will be performed by considering two different dictionaries $\mathbf{\Psi}$ and $\mathbf{\Lambda}$ with L levels each. Each level consists of K_l dictionary atoms and the sub-dictionaries in each level are indicated by $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ respectively. Note that the sub-dictionaries $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ are the cluster centers learned using K-lines clustering on the training data for level l. The steps involved in proving the overall stability of the algorithm are: (a) showing that each level of the algorithm is stable in terms of $L_1(P)$ distance between the distortion functions, defined in (9), as the number of training samples $T \rightarrow \infty$ (Section IV-A), (b) proving that stability in terms of $L_1(P)$ distances indicates closeness of the centers of the two clusterings (Section IV-B), in terms of the metric defined in (11), and (c) showing that level-wise stability leads to overall stability of the dictionary learning algorithm (Section IV-C).

A. Level-wise Stability

Let us define a probability space $(\mathcal{Y}_l, \mathbf{\Sigma}_l, P_l)$ where $\mathcal{Y}_l$ is the data that lies in $\mathbb{R}^M$, and P_l is the probability measure. The training samples for the sub-dictionaries $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ are two different sets of T i.i.d. realizations from the probability space. We also assume that the ℓ_2 norm of the training samples is bounded from above and below (i.e.), $0 < r \leq \|\mathbf{y}\|_2 \leq R < \infty$. Note that, in a general case, the data will lie in $\mathbb{R}^M$ for the first level of dictionary learning and in a finite union of lower-dimensional subspaces of $\mathbb{R}^M$ for the subsequent levels. In both cases, the following argument on stability will hold. This is because when the training data lies in a union of lower dimensional subspaces of $\mathbb{R}^M$, we can assume it to be still lying in $\mathbb{R}^M$, but assign the probabilities outside the union of subspaces to be zero.

In each level, $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ are learned using the K-lines clustering algorithm on two different i.i.d. sets of training data. The distortion function class for the clusterings, defined similar to (5), is uniform Donsker because the covering number with respect to the supremum norm grows polynomially, according to (8). When a unique minimizer exists for the clustering objective, the distortion functions corresponding to the different clusterings $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ become arbitrarily close, $\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} \xrightarrow{P} 0$, even for completely disjoint training sets, as $T \rightarrow \infty$. However, in the case of multiple minimizers, $\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} \xrightarrow{P} 0$ holds only with respect to a change of $o(\sqrt{T})$ training samples between the two clusterings, and fails to hold when there is a change of $\Omega(\sqrt{T})$ samples [38], [43].

B. Distance between Cluster Centers for a Stable Clustering

For each cluster center in the clustering $\mathbf{\Psi}_l$, we pick the closest cluster center from $\mathbf{\Lambda}_l$, in terms of the distortion measure (4), and form pairs. Let us indicate the j^{th} pair of cluster centers by $\boldsymbol{\psi}_{l,j}$ and $\boldsymbol{\lambda}_{l,j}$. Let us define τ disjoint sets $\{A_i\}_{i=1}^{\tau}$, in which the training data for the clusterings exist, such that $P_l(\cup_{i=1}^{\tau} A_i) = 1$. By defining such disjoint sets, we can formalize the notion of training data lying in a union of subspaces of $\mathbb{R}^M$. The intuitive fact that the cluster centers of two clusterings are close to each other in $\mathbb{R}^M$ space, given that their distortion functions are close, is proved in the lemma below.

Lemma 4.1: Consider two sub-dictionaries (clusterings) $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ with K_l atoms each obtained using the T training samples that exist in the τ disjoint sets $\{A_i\}_{i=1}^{\tau}$ in the $\mathbb{R}^M$ space, with $0 < r \leq \|\mathbf{y}\|_2 \leq R < \infty$, and $dP_l(\mathbf{y})/d\mathbf{y} > C$ in each of the sets. When the distortion functions become arbitrarily close to each other, $\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} \xrightarrow{P} 0$ as $T \rightarrow \infty$, the smallest angle between the subspaces spanned by the cluster centers becomes arbitrarily close to zero, i.e.,

$$\angle(\boldsymbol{\psi}_{l,j}, \boldsymbol{\lambda}_{l,j}) \xrightarrow{P} 0, , \forall j \in 1, \ldots, K_l. \tag{20}$$

Proof: Denote the smallest angle between the subspaces represented by $\boldsymbol{\psi}_{l,j}$ and $\lambda_{l,j}$ as $\angle(\boldsymbol{\psi}_{l,j}, \lambda_{l,j}) = \rho_{l,j}$ and define a region $S(\boldsymbol{\psi}_{l,j}, \rho_{l,j}/2) = \{\mathbf{y} | \angle(\boldsymbol{\psi}_{l,j}, \mathbf{y}) \leq \rho_{l,j}/2, 0 < r \leq \|\mathbf{y}\|_2 \leq R < \infty\}$. If $\mathbf{y} \in S(\boldsymbol{\psi}_{l,j}, \rho_{l,j}/2)$, then $\mathbf{y}^T(\mathbf{I} - \boldsymbol{\psi}_{l,j}\boldsymbol{\psi}_{l,j}^T)\mathbf{y} \leq \mathbf{y}^T(\mathbf{I} - \boldsymbol{\lambda}_{l,j}\boldsymbol{\lambda}_{l,j}^T)\mathbf{y}$. An illustration of this

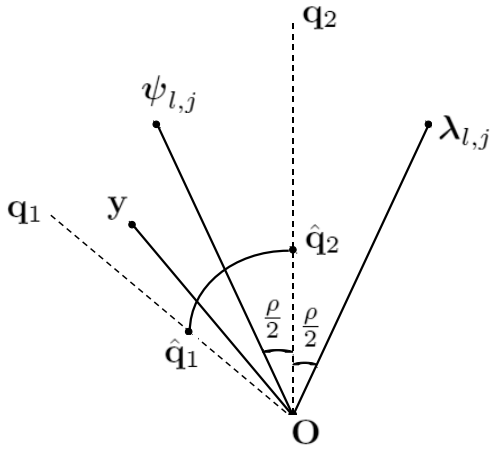


Fig. 5. Illustration for showing the stability of cluster centroids from the stability of distortion function.

setup for a 2-D case is given in Figure 5. In this figure, the arc $\hat{\mathbf{q}}_1\hat{\mathbf{q}}_2$ is of radius r and represents the minimum value of $\|\mathbf{y}\|_2$. By definition, the $L_1(P_l)$ distance between the distortion functions of the clusterings for data that exists in the disjoint sets $\{A_i\}_{i=1}^{\tau}$ is

$$\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} = \sum_{i=1}^{\tau} \int_{A_i} |g_{\mathbf{\Psi}_l}(\mathbf{y}) - g_{\mathbf{\Lambda}_l}(\mathbf{y})| dP_l(\mathbf{y}). \quad (21)$$

For any j and i with a non-empty $B_{l,i,j} = S(\boldsymbol{\psi}_{l,j}, \rho_{l,j}/2) \cap A_i$ we have,

$$\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} \geq \int_{B_{l,i,j}} |g_{\mathbf{\Psi}_l}(\mathbf{y}) - g_{\mathbf{\Lambda}_l}(\mathbf{y})| dP_l(\mathbf{y}), \quad (22)$$

$$= \int_{B_{l,i,j}} [\mathbf{y}^T \left(\mathbf{I} - \boldsymbol{\lambda}_{l,j}\boldsymbol{\lambda}_{l,j}^T\right) \mathbf{y} - \sum_{k=1}^{K} \mathbf{y}^T \left(\mathbf{I} - \boldsymbol{\psi}_{l,k}\boldsymbol{\psi}_{l,k}^T\right) \mathbf{y} \; \mathbb{I}\left(\mathbf{y} \text{ closest to } \boldsymbol{\psi}_{l,k}\right)] dP_l(\mathbf{y}), \quad (23)$$

$$\geq \int_{B_{l,i,j}} [\mathbf{y}^T \left(\mathbf{I} - \boldsymbol{\lambda}_{l,j}\boldsymbol{\lambda}_{l,j}^T\right) \mathbf{y} - \mathbf{y}^T \left(\mathbf{I} - \boldsymbol{\psi}_{l,j}\boldsymbol{\psi}_{l,j}^T\right) \mathbf{y}] dP_l(\mathbf{y}), \quad (24)$$

$$\geq C \int_{B_{l,i,j}} \left[\left(\mathbf{y}^T\boldsymbol{\psi}_{l,j}\right)^2 - \left(\mathbf{y}^T\boldsymbol{\lambda}_{l,j}\right)^2\right] d\mathbf{y}. \quad (25)$$

We have $g_{\mathbf{\Lambda}_l}(\mathbf{y}) = \mathbf{y}^T \left(\mathbf{I} - \boldsymbol{\lambda}_{l,j}\boldsymbol{\lambda}_{l,j}^T\right) \mathbf{y}$ in (23), since $\boldsymbol{\lambda}_{l,j}$ is the closest cluster center to the data in $S(\boldsymbol{\psi}_{l,j}, \rho_{l,j}/2) \cap A_i$ in terms of the distortion measure (4). Note that $\mathbb{I}$ is the indicator function and (25) follows from (24) because $dP_l(\mathbf{y})/d\mathbf{y} > C$. Since by assumption, $\|g_{\mathbf{\Psi}_l} - g_{\mathbf{\Lambda}_l}\|_{L_1(P_l)} \xrightarrow{P} 0$, from (25), we have

$$\left(\mathbf{y}^T\boldsymbol{\psi}_{l,j}\right)^2 - \left(\mathbf{y}^T\boldsymbol{\lambda}_{l,j}\right)^2 \xrightarrow{P} 0, \quad (26)$$

because the integrand in (25) is a continuous non-negative function in the region of integration.

Denoting the smallest angles between $\mathbf{y}$ and the subspaces spanned by $\boldsymbol{\psi}_{l,j}$ and $\boldsymbol{\lambda}_{l,j}$ to be $\theta_{\boldsymbol{\psi}_{l,j}}$ and $\theta_{\boldsymbol{\lambda}_{l,j}}$ respectively, from (26), we have $\|\mathbf{y}\|_2^2(\cos^2\theta_{\boldsymbol{\psi}_{l,j}} - \cos^2\theta_{\boldsymbol{\lambda}_{l,j}}) \xrightarrow{P} 0$, for all $\mathbf{y}$. By definition of the region $B_{l,i,j}$, we have $\theta_{\boldsymbol{\psi}_{l,j}} \leq \theta_{\boldsymbol{\lambda}_{l,j}}$. Since $\|\mathbf{y}\|_2$ is bounded away from zero and infinity, if $(\cos^2\theta_{\boldsymbol{\psi}_{l,j}} - \cos^2\theta_{\boldsymbol{\lambda}_{l,j}}) \xrightarrow{P} 0$ holds for all $\mathbf{y} \in B_{l,i,j}$, then we have $\angle(\boldsymbol{\psi}_{l,j}, \boldsymbol{\lambda}_{l,j}) \xrightarrow{P} 0$. This is true for all cluster center pairs as we have shown this for an arbitrary i and j. ■

C. Stability of the MLD Algorithm

The stability of the MLD algorithm as a whole, is proved in Theorem 4.3 from its level-wise stability by using an induction argument. The proof will depend on the following lemma which shows that the residuals from two stable clusterings belong to the same probability space.

Lemma 4.2: When the training vectors for the sub-dictionaries (clusterings) $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ are obtained from the probability space $(\mathcal{Y}_l, \mathbf{\Sigma}_l, P_l)$, and the cluster center pairs become arbitrarily close to each other as $T \to \infty$, the residual vectors from both the clusterings belong to an identical probability space $(\mathcal{Y}_{l+1}, \mathbf{\Sigma}_{l+1}, P_{l+1})$.

Proof: For the j^{th} cluster center pair $\boldsymbol{\psi}_{l,j}$, $\boldsymbol{\lambda}_{l,j}$, define $\bar{\mathbf{\Psi}}_{l,j}$ and $\bar{\mathbf{\Lambda}}_{l,j}$ as the projection matrices for their respective orthogonal complement subspaces $\boldsymbol{\psi}_{l,j}^{\perp}$ and $\boldsymbol{\lambda}_{l,j}^{\perp}$. Define the sets $D_{\boldsymbol{\psi}_{l,j}} = \{\mathbf{y} \in \bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta}) + \boldsymbol{\psi}_{l,j}\alpha\}$ and $D_{\boldsymbol{\lambda}_{l,j}} = \{\mathbf{y} \in \bar{\mathbf{\Lambda}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta}) + \boldsymbol{\lambda}_{l,j}\alpha\}$, where $-\infty < \alpha < \infty$, $\boldsymbol{\beta}$ is an arbitrary fixed vector, not orthogonal to both $\boldsymbol{\psi}_{l,j}$ and $\boldsymbol{\lambda}_{l,j}$, and $d\boldsymbol{\beta}$ is a differential element. The residual vector set for the cluster $\boldsymbol{\psi}_{l,j}$, when $\mathbf{y} \in D_{\boldsymbol{\psi}_{l,j}}$ is given by, $\mathbf{r}_{\boldsymbol{\psi}_{l,j}} \in \{\bar{\mathbf{\Psi}}_{l,j}\mathbf{y} | \mathbf{y} \in D_{\boldsymbol{\psi}_{l,j}}\}$, or equivalently $\mathbf{r}_{\boldsymbol{\psi}_{l,j}} \in \{\bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}$. Similarly for the cluster $\boldsymbol{\lambda}_{l,j}$, we have $\mathbf{r}_{\boldsymbol{\lambda}_{l,j}} \in \{\bar{\mathbf{\Lambda}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}$. For a 2-D case, Figure 6 shows the 1-D subspace $\boldsymbol{\psi}_{l,j}$, its orthogonal complement $\boldsymbol{\psi}_{l,j}^{\perp}$, the set $D_{\boldsymbol{\psi}_{l,j}}$ and the residual set $\{\bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}$.

In terms of probabilities, we also have that $P_l(\mathbf{y} \in D_{\boldsymbol{\psi}_{l,j}}) = P_{l+1}(\mathbf{r}_{\boldsymbol{\psi}_{l,j}} \in \{\bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\})$, because the residual set $\{\bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}$ is obtained by a linear transformation of $D_{\boldsymbol{\psi}_{l,j}}$. Here P_l and P_{l+1} are probability measures defined on the training data for levels l and $l+1$ respectively. Similarly, $P_l(\mathbf{y} \in D_{\boldsymbol{\lambda}_{l,j}}) = P_{l+1}(\mathbf{r}_{\boldsymbol{\lambda}_{l,j}} \in \{\bar{\mathbf{\Lambda}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\})$. When $T \to \infty$, the cluster center pairs become arbitrarily close to each other, i.e., $\angle(\boldsymbol{\psi}_{l,j}, \boldsymbol{\lambda}_{l,j}) \xrightarrow{P} 0$, by assumption. Therefore, the symmetric difference between the sets $D_{\boldsymbol{\psi}_{l,j}}$ and $D_{\boldsymbol{\lambda}_{l,j}}$ approaches the null set, which implies that $P_l(\mathbf{y} \in D_{\boldsymbol{\psi}_{l,j}}) - P_l(\mathbf{y} \in D_{\boldsymbol{\lambda}_{l,j}}) \to 0$. This implies,

$$P_{l+1}(\mathbf{r}_{\boldsymbol{\psi}_{l,j}} \in \{\bar{\mathbf{\Psi}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}) - P_{l+1}(\mathbf{r}_{\boldsymbol{\lambda}_{l,j}} \in \{\bar{\mathbf{\Lambda}}_{l,j}(\boldsymbol{\beta} + d\boldsymbol{\beta})\}) \to 0, \quad (27)$$

for an arbitrary $\boldsymbol{\beta}$ and $d\boldsymbol{\beta}$, as $T \to \infty$. This means that the residuals of $\boldsymbol{\psi}_{l,j}$ and $\boldsymbol{\lambda}_{l,j}$ belong to a unique but identical probability space. Since we proved this for an arbitrary l and j, we can say that the residuals of clusterings $\mathbf{\Psi}_l$ and $\mathbf{\Lambda}_l$ belong to an identical probability space given by $(\mathcal{Y}_{l+1}, \mathbf{\Sigma}_{l+1}, P_{l+1})$. ■

Theorem 4.3: Given that the training vectors for the first level are generated from the probability space $(\mathcal{Y}_1, \mathbf{\Sigma}_1, P_1)$, and the norms of training vectors for each level are bounded as $0 < r \leq \|\mathbf{y}\|_2 \leq R < \infty$, the MLD learning algorithm is stable as a whole.

Proof: The level-wise stability of MLD was shown in Section IV-A, for two cases: (a) when a unique minimizer

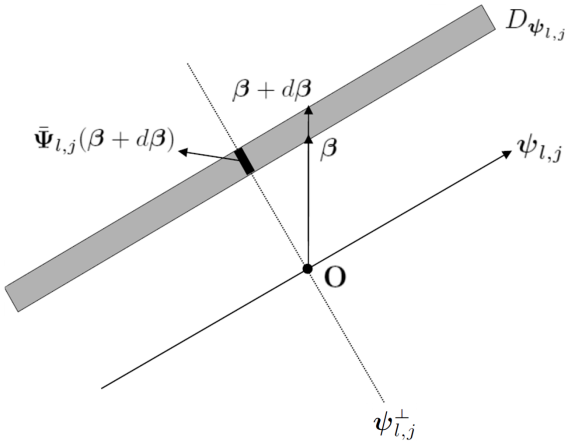


Fig. 6. The residual set $\{\bar{\boldsymbol{\Psi}}_{l,j}(\boldsymbol{\beta}+d\boldsymbol{\beta})\}$, for the 1-D subspace $\boldsymbol{\psi}_{l,j}$, lying in its orthogonal complement subspace $\boldsymbol{\psi}^{\perp}_{l,j}$.

exists for the distortion function and (b) when a unique minimizer does not exist. Lemma 4.1 proved that the stability in terms of closeness of distortion functions implied stability in terms of learned cluster centers. For showing the level-wise stability, we assumed that the training vectors in level l for clusterings $\boldsymbol{\Psi}_l$ and $\boldsymbol{\Lambda}_l$ belonged to the same probability space. However, when learning the dictionary, this is true only for the first level, as we supply the algorithm with training vectors from the probability space $(\mathcal{Y}_1, \boldsymbol{\Sigma}_1, P_1)$.

We note that the training vectors for level $l+1$ are residuals of the clusterings $\boldsymbol{\Psi}_l$ and $\boldsymbol{\Lambda}_l$. Lemma 4.2 showed that the residuals of level l for both the clusterings belong to an identical probability space $(\mathcal{Y}_{l+1}, \boldsymbol{\Sigma}_{l+1}, P_{l+1})$, given that the training vectors of level l are realizations from the probability space $(\mathcal{Y}_l, \boldsymbol{\Sigma}_l, P_l)$ and $T \to \infty$. By induction, this along with the fact that the training vectors for level 1 belong to the same probability space $(\mathcal{Y}_1, \boldsymbol{\Sigma}_1, P_1)$, shows that all the training vectors of both the dictionaries for any level l indeed belong to a probability space $(\mathcal{Y}_l, \boldsymbol{\Sigma}_l, P_l)$ corresponding to that level. Hence all the levels of the dictionary learning are stable and the MLD learning is stable as a whole. ■

If there is a unique minimizer to the clustering objective in all levels of MLD learning, then the MLD algorithm is stable even for completely disjoint training sets, as $T \to \infty$. However, if there are multiple minimizers in at least one level, the algorithm is stable only with respect to a change of $o(\sqrt{T})$ training samples between the two clusterings. In particular, a change in $\Omega(\sqrt{T})$ samples makes the algorithm unstable.

D. Generalization Analysis

Since our learning algorithm consists of multiple levels, and cannot be expressed as an ERM on a whole, the algorithm can be said to generalize asymptotically if the sum of empirical errors for all levels converge to the sum of expected errors, as $T \to \infty$. This can be expressed as

$$\left| \frac{1}{T} \sum_{l=1}^{L} \sum_{i=1}^{T} g_{\boldsymbol{\Psi}_l}(\mathbf{y}_{l,i}) - \sum_{l=1}^{L} \mathbb{E}_{P_l}[g_{\boldsymbol{\Psi}_l}] \right| \xrightarrow{P} 0, \qquad (28)$$

where the training samples for level l given by $\{\mathbf{y}_{l,i}\}_{i=1}^{T}$ are obtained from the probability space $(\mathcal{Y}_l, \boldsymbol{\Sigma}_l, P_l)$. When (28) holds and the learning algorithm generalizes, it can be seen that the expected error for test data which is drawn from the same probability space as that of the training data, is close to the average empirical error. Therefore, when the cluster centers for each level are obtained by minimizing the empirical error, we are guaranteed that the expected test error will also be small.

In order to show that (28) holds, we use the fact that each level of MLD learning is obtained using K-lines clustering. Hence, from (7), the average empirical distortion in each level converges to the expected distortion as $T \to \infty$,

$$\left| \frac{1}{T} \sum_{i=1}^{T} g_{\boldsymbol{\Psi}_l}(\mathbf{y}_{l,i}) - \mathbb{E}_{P_l}[g_{\boldsymbol{\Psi}_l}] \right| \xrightarrow{P} 0. \qquad (29)$$

The validity of the condition in (28) follows directly from the triangle inequality,

$$\begin{aligned} &\left| \frac{1}{T} \sum_{l=1}^{L} \sum_{i=1}^{T} g_{\boldsymbol{\Psi}_l}(\mathbf{y}_{l,i}) - \sum_{l=1}^{L} \mathbb{E}_{P_l}[g_{\boldsymbol{\Psi}_l}] \right| \\ &\leq \sum_{l=1}^{L} \left| \frac{1}{T} \sum_{i=1}^{T} g_{\boldsymbol{\Psi}_l}(\mathbf{y}_{l,i}) - \mathbb{E}_{P_l}[g_{\boldsymbol{\Psi}_l}] \right|. \qquad (30) \end{aligned}$$

If the M-OMP coding scheme is used for test data, and the training and test data for level 1 are obtained from the probability space $(\mathcal{Y}_1, \boldsymbol{\Sigma}_1, P_1)$, the probability space for both training and test data in level l will be $(\mathcal{Y}_l, \boldsymbol{\Sigma}_l, P_l)$. This is because, both the M-OMP coding scheme and the MLD learning associate the data to a dictionary atom using the maximum absolute correlation measure and create a residual that is orthogonal to the atoms chosen so far. Hence, the assumption that training and test data are drawn from the same probability space in all levels hold and the expected test error will be similar to the average empirical training error.

E. Simulations

Both stability and generalization are crucial for building effective global dictionaries to model natural image patches. Although it is not possible to demonstrate the asymptotic behavior experimentally, we study the changes in the behavior of the learning algorithm with increase in the number of samples used for training.

In order to illustrate the stability characteristics of MLD learning, we setup an experiment where we consider a multilevel dictionary of 4 levels, with 8 atoms in each level. We extracted patches of size 8×8 from the BSDS training images and trained multilevel dictionaries using different number of training patches T. As we showed in Section IV, asymptotic stability is guaranteed when the training set is changed by not more than $o(\sqrt{T})$ samples. The inferred dictionary atoms will not vary significantly, if this condition is satisfied.

We fixed the size of the training set at different values $T = \{1000, 5000, 10000, 50{,}000, 100000\}$ and learned an initial set of dictionaries using the proposed algorithm. The second set of dictionaries were obtained by replacing different

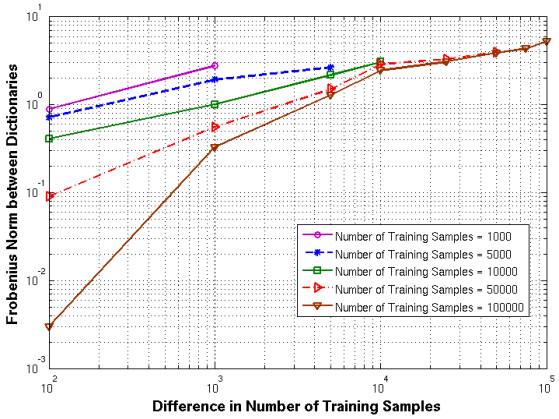


Fig. 7. Demonstration of the stability behavior of the proposed MLD learning algorithm. The minimum Frobenius norm between difference of two dictionaries with respect to permutation of their columns and signs is shown. The second dictionary is obtained by replacing different number of samples in the training set, used for training the original dictionary, with new data samples.

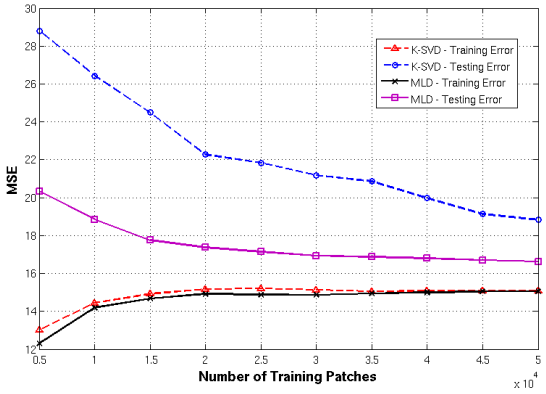


Fig. 8. Demonstration of the generalization characteristics of the proposed algorithm compared to K-SVD. We plot the MSE obtained by representing patches from the BSDS test dataset, using dictionaries learned with different number of training patches. For comparison, we show the training error obtained in each case.

number of samples from the original training set. For each case of T, the number of replaced samples was varied between 100 and T. For example, when $T = 10000$, the number of replaced training samples were $100, 1000, 5000$, and 10000. The amount of change between the initial and the second set of dictionaries was quantified using the minimum Frobenius norm of their difference with respect to permutations of their columns and sign changes. In Figure 7, we plot this quantity for different values of T as a function of the number of samples replaced in the training set. For each case of T, the difference between the dictionaries increases as we increase the replaced number of training samples. Furthermore, for a fixed number of replaced samples (say 100), the difference reduces with the increase in the number of training samples, since it becomes closer to asymptotic behavior.

Generalization of a dictionary learning algorithm guarantees a small approximation error for a test data sample, if the training samples are well approximated by the dictionary. In order to demonstrate the generalization characteristics of MLD learning, we designed dictionaries using different number of training image patches, of size 8×8, from the BSDS training dataset and evaluated the sparse codes for patches in the BSDS test dataset (Section III-E). The dictionaries were learned at 16 levels with 16 atoms per level. Figure 8 shows the approximation error (MSE) for both the training and test datasets obtained using multilevel dictionaries. Furthermore, the corresponding MSE for the case of similarly designed K-SVD dictionaries are included for comparison. In all cases, the sparsity in training and testing was fixed at $S = 16$. As it can be observed, with MLD, the difference between the MSE for training and test data is small even for a small training set. However, the K-SVD dictionaries resulted in much higher MSE difference for a small training set, although the MSE with training data is similar for both MLD and KSVD. Note that, in both cases, the approximation error for the test data reduces with the increase in the size of the training set.

V. Application: Denoising

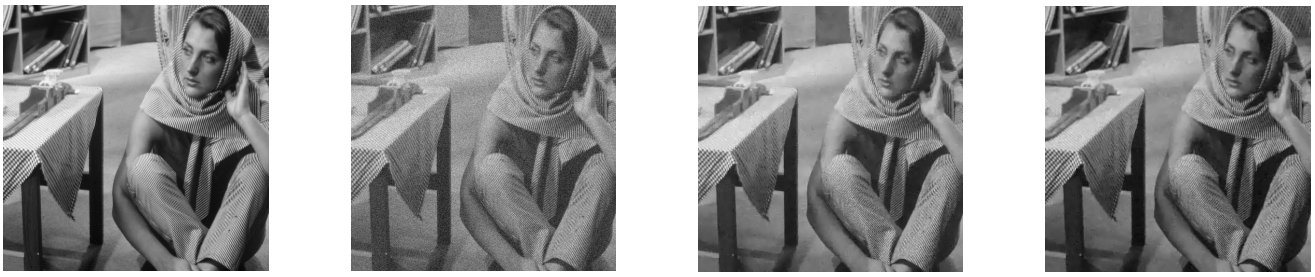
Our goal in denoising is to recover the clean image Y from the noisy observed image X. The image X is divided into patches of size 8×8 with an overlap of 1 pixel, and these patches are vectorized and stacked in the matrix $\mathbf{X}$. A noisy observation $\mathbf{x}$ (a column in $\mathbf{X}$), can be represented as a corrupted version of its corresponding clean patch, $\mathbf{x} = \mathbf{y} + \boldsymbol{\eta}$, where $\boldsymbol{\eta}$ is the AWGN vector with standard deviation σ. Patch-wise recovery was performed using RM-OMP with the global MLD dictionary learned with $50,000$ patches as described in Section III-E. Patchwise error goal was fixed and image-level reconstruction constraints were posed as described in [9]. All results were averaged over 5 iterations. Note that, under low-noise conditions dictionaries learned from the noisy test image itself perform better than global dictionaries. However, under high-noise conditions, global dictionaries perform comparably to image-specific dictionaries. This results in a significant computational advantage since it is not necessary to train a separate dictionary for each noisy image. Furthermore, we focus on global dictionary learning in this paper and hence we compare the results of global MLD and K-SVD dictionaries in Table II for high-noise conditions ($\sigma \geq 20$). For global K-SVD dictionary, the results reported in [9] were used. It can be seen that, in almost all the cases, global MLD performs better than global K-SVD dictionaries. The denoised *Lena* and *Fingerprint* images are shown in Figure 9, for $\sigma = 20$ and $sigma = 50$ respectively, where a clear improvement in reconstruction performance is observed. Computationally, denoising using MLD is less expensive compared to using K-SVD as seen from Table III. All the times reported in this paper are obtained using MATLAB 2012a on a 2.8 GHz, 8-core Intel i7 Linux machine.

VI. Application: Compressed Recovery

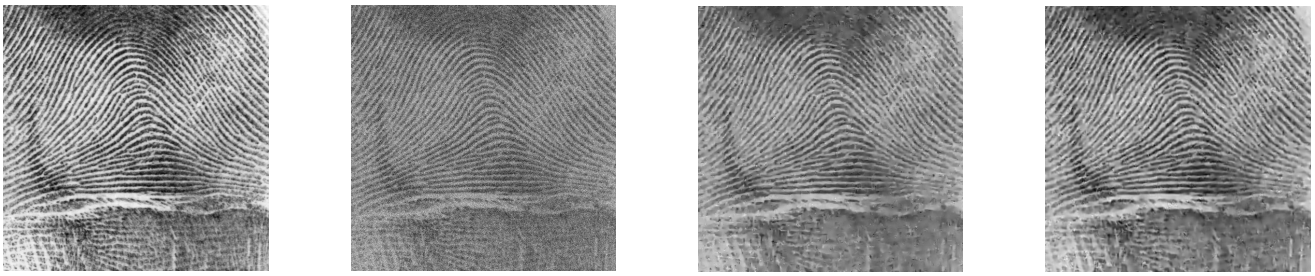
In compressed recovery, the test image is recovered using low-dimensional random projections obtained from its patches. The finite size of the training set and the lack of robustness in the initialization of K-lines clustering can affect the generalization of multilevel dictionaries to test observations, under

TABLE II
PSNR (DB) OF THE DENOISED STANDARD IMAGES CORRUPTED WITH AWGN OF STANDARD DEVIATION σ. IN EACH CASE, THE AVERAGE OF 5 TRIALS IS PROVIDED. HIGHER PERFORMANCE IS SHOWN IN BOLD FONT.

Noise (σ)	**Image**									
	Barbara		**Boat**		**Fingerprint**		**House**		**Lena**	
	K-SVD	**MLD**	**K-SVD**	**MLD**	**K-SVD**	**MLD**	**K-SVD**	**MLD**	**K-SVD**	**MLD**
20	28.87	**29.15**	30.24	**30.31**	28.21	**28.37**	32.88	**32.93**	32.27	**32.44**
25	27.57	**27.91**	29.17	**29.26**	26.94	**27.22**	31.82	**31.99**	31.2	**31.37**
50	24.06	**24.15**	25.91	**25.97**	22.68	**23.36**	27.91	**27.98**	27.77	**27.89**
75	22.54	**22.57**	24.02	**24.06**	19.73	**20.24**	25.33	**25.42**	25.81	**25.92**
100	**21.73**	21.72	22.83	**22.92**	18.23	**18.72**	23.86	**24.06**	24.45	**24.51**



(a) Original Image (b) Noisy Image (22.11 dB) (c) K-SVD (28.89 dB) (d) MLD (29.19 dB)



(a) Original Image (b) Noisy Image (14.15 dB) (c) K-SVD (22.78 dB) (d) MLD (23.38 dB)

Fig. 9. Original, noisy and denoised *Lena* and *Fingerprint* images with their respective PSNRs. Reconstructed images for global K-SVD dictionaries are obtained using the *K-SVD toolbox* [56].

TABLE III
AVERAGE TIME(SECONDS) TAKEN IN MATLAB FOR DENOISING IMAGES OF SIZE 512×512 UNDER DIFFERENT NOISE CONDITIONS.

Method	$\sigma = 20$	$\sigma = 25$	$\sigma = 50$	$\sigma = 75$	$\sigma = 100$
K-SVD [56]	16.47	14.93	11.43	9.58	8.61
MLD	8.79	8.52	7.96	7.54	7.11

such severe degradation. The learning procedure can be made robust by using multiple clusterings in each level, where each clustering is learned from a random subset of the training samples for that level.

Robust Multilevel Dictionaries: Learning robust multilevel dictionaries (RMLD) is closely related to *Rvotes* [57], a supervised ensemble learning method. The *Rvotes* scheme randomly samples the training set to create D sets of T_D samples each, where $T_D \ll T$. The final prediction is obtained by averaging the predictions from the multiple hypotheses learned from the training sets. For learning level l in RMLD, we draw D subsets of randomly chosen training samples, $\{\mathbf{Y}_l^{(d)}\}_{d=1}^{D}$ from the original training set $\mathbf{Y}_l$ of size T, allowing for overlap across the subsets. The superscript here denotes the index of the subset. For each subset $\mathbf{Y}_l^{(d)}$ of size $T_D \ll T$, we learn a sub-dictionary $\mathbf{\Psi}_l^{(d)}$ with K atoms using K-lines clustering. For each training sample in $\mathbf{Y}_l$, we compute $1-$sparse representations using all the D sub-dictionaries, and denote the set of coefficient matrices as $\{\mathbf{A}_l^{(d)}\}_{d=1}^{D}$. The approximation for the i^{th} training sample in level l, $\mathbf{y}_{l,i}$, is computed as the average of approximations using all D sub-dictionaries, $\frac{1}{D}\sum_d \mathbf{\Psi}_l^{(d)}\mathbf{a}_{l,i}^{(d)}$. The ensemble approximations for all training samples in the level can be used to compute the set of residuals, and this process is repeated for a desired number of levels, to obtain a robust multilevel dictionary (RMLD). Because of its improved robustness, reconstruction of test data with an RMLD can be performed using simple level-wise approximation, in contrast to the RM-OMP procedure with an MLD. We obtain $1-$sparse approximations on sub-dictionaries in each level, average the approximations, compute the residual, and repeat this process for the subsequent levels.

Results: The performance of compressed recovery based on

TABLE IV
PSNR (DB) OF THE IMAGES RECOVERED FROM COMPRESSED MEASUREMENTS OBTAINED USING GAUSSIAN RANDOM MEASUREMENT MATRICES. RESULTS OBTAINED USING THE PROPOSED MLD, AND RMLD DICTIONARIES, ALONG WITH K-SVD, ARE SHOWN FOR DIFFERENT MEASUREMENT NOISE CONDITIONS AND NUMBER OF MEASUREMENTS. HIGHER PSNR FOR EACH CASE IS INDICATED IN BOLD FONT.

Measurement SNR (dB)	**Method**	**Image**														
		Barbara			**Boat**			**House**			**Lena**			**Man**		
		N=8	**N=16**	**N=32**	**N=8**	**N=16**	**N=32**	**N=8**	**N=16**	**N=32**	**N=8**	**N=16**	**N=32**	**N=8**	**N=16**	**N=32**
0	K-SVD	19.8	20.54	21.51	21.48	22.07	23.42	22.57	23.91	25.48	23.28	24.23	26.16	22.23	23.2	24.9
	MLD	19.96	20.63	21.9	21.68	22.38	23.56	22.73	23.98	25.54	23.3	24.51	26.43	22.41	23.59	25.18
	RMLD	**21.55**	**22.02**	**22.6**	**23.05**	**23.76**	**24.25**	**24.2**	**24.97**	**26.44**	**24.1**	**25.38**	**26.11**	**23.89**	**25.15**	**25.66**
15	K-SVD	20.93	21.89	24.34	23.03	25.02	27.39	24.91	26.87	31.01	25.01	28.08	31.42	24.02	26.02	28.54
	MLD	21.17	22.41	24.95	23.42	25.26	27.83	25.06	27.15	31.37	25.29	28.29	31.55	24.31	26.19	28.82
	RMLD	**22.58**	**24.16**	**26.17**	**24.82**	**26.69**	**29.48**	**26.41**	**28.79**	**31.38**	**26.89**	**29.11**	**31.4**	**25.51**	**27.62**	**30.04**
25	K-SVD	21.43	22.09	24.99	23.45	25.88	28.7	25.1	27.1	31.6	25.27	29.03	31.83	24.17	26.59	29.36
	MLD	21.56	22.62	25.26	23.69	25.27	28.82	25.36	27.31	31.78	25.48	28.64	32	24.42	26.71	29.59
	RMLD	**22.65**	**24.33**	**26.72**	**25.12**	**27.07**	**29.68**	**26.94**	**29.04**	**32.38**	**27.58**	**29.55**	**32.36**	**25.92**	**27.79**	**30.38**

(a) K-SVD (26.25 dB) (b) MLD (26.59 dB) (c) RMLD (27.81 dB)

Fig. 10. Compressed recovery of images from random measurements ($N = 16$, SNR of measurement process $= 15$dB) using the different dictionaries. In each case the PSNR of the recovered image is also shown.

TABLE V
AVERAGE TIME(SECONDS) TAKEN IN MATLAB FOR TRAINING DICTIONARIES, WITH $50,000$ SAMPLES, AND RECOVERING IMAGES OF SIZE 512×512 USING DIFFERENT NUMBER OF RANDOM MEASUREMENTS.

Method	**Training**	**N = 8**	**N = 16**	**N = 32**
K-SVD [56]	675	0.07	0.09	0.10
MLD	502	0.05	0.11	0.19
RMLD	1980	0.45	1.05	2.31

random measurement systems is compared for global MLD, RMLD and K-SVD dictionaries. Sensing and recovery were performed on a patch-by-patch basis, on non-overlapping patches of size 8×8. MLD and K-SVD dictionaries were learned with $50,000$ BSDS patches as described in Section III-E. For learning the RMLD, we fix $K = 16$ and obtain $D = 20$ rounds of K-lines dictionaries in each level ($L = 16$) using random sets of training data. The measurement process is described as $\mathbf{x} = \mathbf{\Phi\Psi a} + \boldsymbol{\eta}$ where $\mathbf{\Psi}$ is the dictionary, $\mathbf{\Phi}$ is the measurement or projection matrix, $\boldsymbol{\eta}$ is the AWGN vector added to the measurement process, $\mathbf{x}$ is the output of the measurement process, and $\mathbf{a}$ is the sparse coefficient vector such that $\mathbf{y} = \mathbf{\Psi a}$. The size of the data vector $\mathbf{y}$ is $M \times 1$, that of $\mathbf{\Psi}$ is $M \times K$, that of the measurement matrix $\mathbf{\Phi}$ is $N \times M$, where $N < M$, and that of the measured vector $\mathbf{x}$ is $N \times 1$. The entries in the random measurement matrix were independent realizations from a standard normal distribution. We recover the underlying image from its compressed measurements, using the K-SVD, MLD, and RMLD dictionaries. For each case, we present average results from 100 trial runs, each time with a different measurement matrix. The recovery performance was evaluated for several standard images and reported in Table IV. MLD outperforms K-SVD dictionaries in all cases. Furthermore, the proposed RMLD algorithm results in much improved recovery, for increased complexity during training and testing phases. The average time taken for recovering a 512×512 image using the three proposed dictionaries are listed in Table V. Figure 10 illustrates the recovered images obtained using different dictionaries with random measurements.

VII. CONCLUSIONS

We presented a multilevel learning algorithm to design global dictionaries that exploit the redundancy and energy hierarchy found in natural image patches. The proposed algorithm employs K-lines clustering to learn atoms for each level. We showed that the algorithm converges for a sufficient number of levels and that energy hierarchy is exhibited for a sufficient number of atoms per level. We also showed that the dictionaries learned using different sets of training data, from

the same probability space, are arbitrarily close to each other, for a sufficiently large number of data samples. Furthermore, we proved the asymptotic generalization characteristics, and demonstrated the stability and generalization behavior using simulations. Simulation results for denoising and compressed sensing clearly demonstrated that the learned MLD provide superior performance when compared to K-SVD dictionaries.

REFERENCES

[1] D. J. Field, "What is the goal of sensory coding?" *Neural Comp.*, vol. 6, pp. 559–601, 1994.

[2] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE TSP*, vol. 41, no. 12, pp. 3397–3415, 1993.

[3] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, October 2004.

[4] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[5] I. F. Gorodnitsky and B. D. Rao, "Sparse signal reconstruction from limited data using FOCUSS: A re-weighted norm minimization algorithm," *IEEE TSP*, vol. 45, no. 3, pp. 600–616, March 1997.

[6] M. Elad, "Why simple shrinkage is still relevant for redundant representations?" *IEEE Trans. Inf. Theory*, vol. 52, no. 12, pp. 5559 –5569, December 2006.

[7] M. Elad *et.al.*, "A wide-angle view at iterated shrinkage algorithms," in *SPIE*, 2007.

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

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

[10] M. Elad, *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer, 2010.

[11] K. Huang and S. Aviyente, "Sparse representation for signal classification," in *NIPS*, 2006.

[12] J. J. Thiagarajan, K. N. Ramamurthy, and A. Spanias, "Sparse representations for pattern classification using learned dictionaries," *SGAI Int. Conf. on AI*, 2008.

[13] J. J. Thiagarajan, K. N. Ramamurthy, P. Knee, and A. Spanias, "Sparse representations for automatic target classification in SAR images," in *ISCCSP*, 2010.

[14] J. Wright *et.al.*, "Robust face recognition via sparse representation," *IEEE TPAMI*, vol. 31, no. 2, pp. 210–227, 2001.

[15] I. Ramirez, P. Sprechmann, and G. Sapiro, "Classification and clustering via dictionary learning with structured incoherence and shared features," in *IEEE CVPR*, Jun. 2010, pp. 3501 –3508.

[16] J. Yang *et.al.*, "Linear spatial pyramid matching using sparse coding for image classification," in *IEEE CVPR*, 2009.

[17] G. Yu, G. Sapiro, and S. Mallat, "Image modeling and enhancement via structured sparse model selection," in *IEEE ICIP*, Sep. 2010, pp. 1641 –1644.

[18] Q. Zhang and B. Li, "Discriminative K-SVD for dictionary learning in face recognition," in *IEEE CVPR*, 2010.

[19] Z. Jiang *et.al.*, "Learning a discriminative dictionary for sparse coding via label consistent K-SVD," in *IEEE CVPR*, 2011.

[20] J. J. Thiagarajan and A. Spanias, "Learning dictionaries for local sparse coding in image classification," in *Asilomar SSC*, 2011.

[21] J. J. Thiagarajan, K. N. Ramamurthy, P. Sattigeri, and A. Spanias, "Supervised local sparse coding of sub-image features for image retrieval," in *IEEE ICIP*, 2012.

[22] J. Starck, F. Murtagh, and J. Fadili, *Sparse image and signal processing: wavelets, curvelets, morphological diversity*. Cambridge Univ Pr, 2010.

[23] R. Gribonval and M. Nielsen, "Sparse representations in unions of bases," *IEEE Trans. Inf. Theory*, vol. 49, no. 12, pp. 3320–3325, 2003.

[24] B. A. Olshausen and D. J. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision Research*, vol. 37, no. 23, pp. 3311–3325, December 1997.

[25] M. S. Lewicki and T. J. Sejnowski, "Learning overcomplete representations," *Neural Comp.*, vol. 12, no. 2, pp. 337–365, 2000.

[26] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE TPAMI*, vol. 27, no. 12, pp. 1945–1959, 2005.

[27] J. J. Thiagarajan, K. N. Ramamurthy, and A. Spanias, "Shift-invariant sparse representation of images using learned dictionaries," *IEEE MLSP*, pp. 145–150, 2008.

[28] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online learning for matrix factorization and sparse coding," *JMLR*, vol. 11, no. 1, pp. 19–60, 2009.

[29] J. Mairal, G. Sapiro, and M. Elad, "Multiscale sparse image representation with learned dictionaries," in *IEEE ICIP*, 2007.

[30] R. Jenatton, J. Mairal, G. Obozinski, and F. Bach, "Proximal methods for sparse hierarchical dictionary learning," in *ICML*, J. Frankranz and T. Joachims, Eds. Omnipress, 2010, pp. 487–494.

[31] L. Bar and G. Sapiro, "Hierarchical dictionary learning for invariant classification," in *IEEE ICASSP*, March 2010, pp. 3578–3581.

[32] R. Rubinstein, A. Bruckstein, and M. Elad, "Dictionaries for sparse representation modeling," *Proc. IEEE*, vol. 98, no. 6, pp. 1045–1057, 2010.

[33] R. Gribonval and K. Schnass, "Dictionary Identification - Sparse Matrix-Factorisation via ℓ_1-Minimisation," *IEEE Trans. on Inf. Theory*, vol. 56, no. 7, pp. 3523–3539, Jul. 2010.

[34] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *IEEE ICASSP*, 1999.

[35] K. Engan, B. Rao, and K. Kreutz-Delgado, "Frame design using FOCUSS with method of optimal directions (MoD)," in *Norwegian Signal Processing Symposium*, 1999.

[36] Z. He *et.al.*, "K-hyperline clustering learning for sparse component analysis," *Sig. Proc.*, vol. 89, pp. 1011–1022, 2009.

[37] J. J. Thiagarajan, K. N. Ramamurthy, and A. Spanias, "Multilevel dictionary learning for sparse representation of images," in *IEEE DSP Workshop*, 2011.

[38] ——, "Optimality and stability of the K-hyperline clustering algorithm," *Patt. Rec. Letters*, 2010.

[39] D. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *Journal of the Optical Society of America*, vol. 4, pp. 2379–2394, 1987.

[40] J. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE TIP*, vol. 18, no. 7, pp. 1395–1408, 2009.

[41] A. Caponnetto and A. Rakhlin, "Stability properties of empirical risk minimization over Donsker classes," *JMLR*, vol. 7, pp. 2565–2583, 2006.

[42] T. Poggio, R. Rifkin, S. Mukherjee, and P. Niyogi, "General conditions for predictivity in learning theory," *Nature*, vol. 428, no. 6981, pp. 419–422, 2004.

[43] A. Rakhlin and A. Caponnetto, "Stability of K-means clustering," in *Advances in Neural Information Processing Systems*, vol. 19. Cambridge, MA: MIT Press, 2007.

[44] S. Ben-David, U. von Luxburg, and D. Pál, "A sober look at clustering stability," *Conference on Computational Learning Theory*, pp. 5–19, 2006.

[45] S. Ben-David, D. Pál, and H. U. Simon, "Stability of K-means clustering," ser. Lecture Notes in Computer Science, vol. 4539. Springer, 2007, pp. 20–34.

[46] A. Maurer and M. Pontil, "K-Dimensional Coding Schemes in Hilbert Spaces," *IEEE Trans. on Inf. Theory*, vol. 56, no. 11, pp. 5839–5846, 2010.

[47] D. Vainsencher and A. M. Bruckstein, "The Sample Complexity of Dictionary Learning," *JMLR*, vol. 12, pp. 3259–3281, 2011.

[48] P. Bradley and O. Mangasarian, "K-plane clustering," *Journal of Global Opt.*, vol. 16, no. 1, pp. 23–32, 2000.

[49] P. Agarwal and N. Mustafa, "K-means projective clustering," in *ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 2004, pp. 155–165.

[50] P. Tseng, "Nearest q-flat to m points," *Journal of Optimization Theory and Applications*, vol. 105, no. 1, pp. 249–252, 2000.

[51] J. M. Buhmann, "Empirical risk approximation: An induction principle for unsupervised learning," The University of Bonn, Tech. Rep. IAI-TR-98-3, 1998.

[52] R. Raina *et.al.*, "Self-taught learning: Transfer learning from unlabeled data," in *ICML*, 2007.

[53] S. Zhu, K. Shi, and Z. Si, "Learning explicit and implicit visual manifolds by information projection," *Patt. Rec. Letters*, vol. 31, pp. 667–685, 2010.

[54] G. Yu, G. Sapiro, and S. Mallat, "Image modeling and enhancement via structured sparse model selection," in *IEEE ICIP*, 2010.

[55] "Berkeley segmentation dataset," Available at http://www.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/.

[56] "K-SVD Matlab toolbox," Available at http://www.cs.technion.ac.il/∼elad/software/.

[57] L. Breiman, “Pasting small votes for classification in large databases and on-line,” *Machine Learning*, vol. 36, no. 1, pp. 85–103, 1999.