An Efficient Dictionary Learning Algorithm and Its Application to 3-D Medical Image Denoising

Shutao Li*, *Member, IEEE*, Leyuan Fang, *Student Member, IEEE*, and Haitao Yin

***Abstract*—In this paper, we propose an efficient dictionary learning algorithm for sparse representation of given data and suggest a way to apply this algorithm to 3-D medical image denoising. Our learning approach is composed of two main parts: sparse coding and dictionary updating. On the sparse coding stage, an efficient algorithm named multiple clusters pursuit (MCP) is proposed. The MCP first applies a dictionary structuring strategy to cluster the atoms with high coherence together, and then employs a multiple-selection strategy to select several competitive atoms at each iteration. These two strategies can greatly reduce the computation complexity of the MCP and assist it to obtain better sparse solution. On the dictionary updating stage, the alternating optimization that efficiently approximates the singular value decomposition is introduced. Furthermore, in the 3-D medical image denoising application, a joint 3-D operation is proposed for taking the learning capabilities of the presented algorithm to simultaneously capture the correlations within each slice and correlations across the nearby slices, thereby obtaining better denoising results. The experiments on both synthetically generated data and real 3-D medical images demonstrate that the proposed approach has superior performance compared to some well-known methods.**

***Index Terms*—Dictionary learning, *k*-means clustering, multiple-selection strategy, sparse representation, 3-D medical image denoising.**

I. Introduction

MEDICAL imaging modalities, such as computed tomography (CT) and magnetic resonance (MR), have allowed clinicians and medical researchers to study the structural and functional features of the human body, thereby assisting the clinical diagnosis. However, due to the highly controlled imaging environment (e.g., limited light intensities), the imaging process often creates noise, which seriously affects the analysis of the medical image. Therefore, signal denoising remains an important problem for the biomedical engineering community. Actually, numerous methods have been explored for this problem, such as adaptive filters [1], [2], statistical estimators [3]–[5], transform-domain methods [6]–[9], etc.

Recently, sparse representation theory has attracted a lot of attentions. Sparse representation models signals as sparse linear combinations of atoms from a dictionary $\mathbf{D} \in \mathbb{R}^{N\times K}$ [10]–[13]. Given a noisy signal $\mathrm{Y} \in \mathbb{R}^N$, this model assumes that clean parts of the noisy signal have good sparse representations with respect to a predefined dictionary, whereas the dictionary cannot sparsely represent its noisy parts. A fundamental consideration in employing this sparse model is the choice of the dictionary $\mathbf{D}$. Using learning algorithms to train the dictionary from a set of training samples have led to methods with state-of-the-art performance for the 2-D image denoising application [14], [15]. However, the complexity constraints in these learning algorithms often limit the size of the dictionaries that can be trained, and, thus, prohibit their application to 3-D image denoising tasks. Therefore, it is important to develop efficient strategies to accelerate the dictionary learning process.

As we know, the main process of many dictionary learning algorithms [16]–[18] can be divided into two stages: sparse coding and dictionary updating. Sparse coding is to find the *best*, sparsest solution of the training signals, which dominates the complexity of the dictionary learning. The most commonly used strategy for the sparse coding is a greedy algorithm named orthogonal matching pursuit (OMP) [19], which selects an atom at each iteration based on a greedy selection rule. Works by Tropp [20] and Donoho *et al*. [21] have shown that if the atoms in $\mathbf{D}$ are adequately uncorrelated, the OMP algorithm is able to find sufficiently *good* representations. However, this is not the case for highly correlated redundant dictionaries, since high correlation can fool the pursuit and result in wrong solutions. Even though specific optimizations are possible for particular classes of dictionaries, the computational complexity of this algorithm remains very high.

In this paper, we propose an improvement to the OMP algorithm called multiple clusters pursuit (MCP) by adding a dictionary structuring strategy and a multiple-selection strategy. The dictionary structuring strategy uses the *k*-means clustering approach [22] to modify the structure of the dictionary by grouping the atoms with similar properties together and to calculate the prototype atom to represent each cluster. As the similarity measured here is based on the coherence, the clustering minimizes the coherence among prototype atoms and enables these atoms to constitute a subdictionary that is relatively quasi-incoherent (the cumulative coherence in this subdictionary grows slowly [20], [23]). Therefore, even for highly correlated dictionaries, the theoretical results of the greedy method

Manuscript received April 21, 2011; revised August 16, 2011 and September 29, 2011; accepted October 21, 2011. Date of publication October 27, 2011; date of current version January 20, 2012. This work was supported in part by the National Natural Science Foundation of China under Grant 60871096 and Grant 61172161, by the Key Project of Chinese Ministry of Education under Grant 2009-120, by the Scholarship Award for Excellent Doctoral Student granted by the Chinese Ministry of Education, and by the Fundamental Research Funds for the Central Universities, Hunan University. *Asterisk indicates corresponding author*.

*S. Li is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: shutao_li@ yahoo.com.cn).

L. Fang and H. Yin are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: fangleyuan@gmail.com; ocean_waves@126.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TBME.2011.2173935

based on the small coherence remain valid at the prototype atoms. Then, a multiple-selection strategy is then performed on the subdictionary, which allows for multiple prototype atoms (corresponding to its clusters) to be selected at each iteration. Finally, the search can be easily operated in the selected clusters to obtain multiple competitive atoms. In this way, the computational complexity of the searching process is greatly decreased due to the reduced searching space and the smaller number of iteration.

Since several recent pursuit approaches [23]–[26] are related to the proposed MCP algorithm, we will briefly review them and compare their difference. On the one hand, the similar idea of exploiting the dictionary properties to accelerate the atom searching process is also adopted in the tree-based pursuit (TBP) algorithm [23] by embedding a tree structure into the dictionary. In [23], the tree structure is constructed by a hierarchical clustering approach, which recursively clusters the most similar two atoms together and represents them with a molecule atom. It is worthwhile to note that the calculation of the prototype atom in the MCP is the same as the computation of the molecule atom. Moreover, the dictionary structure built by the *k*-means approach can be seen as a *special* case of the tree structure when the tree depth is two (leaf node atoms and one level of molecule atoms). However, the TBP algorithm only aims to select one atom at each iteration, and thus, its whole iteration number is larger than of the MCP. On the other hand, the stagewise OMP (StOMP) [24], M-term pursuit (MTP) [25], and fast MTP [26] all adopt the multiselection strategy in their atoms searching process. Compared with the proposed MCP algorithm, the atoms searching part in the StOMP and MTP algorithm takes more computation cost, since they both require searching all the atoms in the dictionary at each iteration. To accelerate the searching process, the fast MTP algorithm divides the process into two parts: the full search on all the atoms and local search on atoms in a subdictionary. Though such a process may be more efficient than the MTP, it is still different from that of the proposed MCP algorithm.

After the sparse representation is found, the updating of the dictionary is comparatively easier. The gradient descent with iterative projection [27], singular value decomposition (SVD) [18], and other least square methods [16], [17] are able to solve this dictionary updating problem. In this paper, an alternating optimization strategy [32], [33] is introduced to efficiently approximate the exact SVD computation and further accelerate the dictionary learning algorithm.

An intuitive way to extend the proposed dictionary learning algorithm to 3-D medical image denoising is to denoise each slice, respectively, with different learned dictionaries. However, the neighboring slices of medical image have strong correlations among them (termed as the interslice correlation), while there still exist meaningful correlations within each slice (termed as the intraslice correlation) [8], [28]. Therefore, a more reasonable way to denoise the 3-D image should consider the two kinds of correlations. According to this idea, we put forward a joint 3-D operation to process the patches from the same slice and nearby slices jointly by adopting two types of process blocks: 3-D intrablock and 3-D interblock. The intrablock is constructed by a fixed number of 2-D patches with the highest similarity to the currently processing one, and the interblock is composed of the 2-D patches extracted from the nearby slices. Then, the joint 3-D operation concatenates all the 2-D patches in both intrablocks and interblocks to a single vector and trains the dictionary on it. This takes the advantage of the learning capabilities of our training algorithm to fully capture both intraslice and interslice correlations in the medical slices, thereby getting better sparse representations of them.

It is worth noting that this paper is an extension of our previous work in [29]. However, compared with [29], this paper incorporates two new contributions: 1) combine the strength of the dictionary structuring strategy and the multipleselection strategy in the atom searching process; 2) propose the joint 3-D operation to sufficiently utilize both the intraslice and interslice correlations in the medical slices.

The rest of this paper is organized as follows. In Section II, we introduce the MCP and combine it with the alternating optimization to form our dictionary learning algorithm. Section III describes how to apply our dictionary learning algorithm to 3-D medical image denoising. The experimental results on both synthetic data and real 3-D medical images are presented in Section IV. Section V concludes this paper and suggests future works.

II. Dictionary Learning

Dictionary learning is the task of learning or training a dictionary such that it is well adapted to the training data. Usually, the objective is to give one sparse representation of the training set, making the total error as small as possible. Let the training data constitute the columns in the matrix $\mathbf{Y}$ and the sparse coefficient vectors are the columns in the matrix $\mathbf{X}$. The objective function of the dictionary learning can be stated formally as a minimization problem

$$\min_{\mathbf{D},\mathbf{X}} ||\mathbf{Y}-\mathbf{D}\mathbf{X}||_F^2 \text{ subject to } ||\mathbf{X}||_{\mathrm{P}} \leq T \tag{1}$$

where the function $||\cdot||_{\mathrm{P}}$ denotes the ℓ_{P}-norm.

A practical optimization strategy can be implemented by splitting the aforementioned problem into two parts which are alternately solved within an iterative loop [16]–[18]. The two parts are

1) sparse coding: keeping $\mathbf{D}$ fixed, find $\mathbf{X}$;
2) dictionary updating: based on the current $\mathbf{X}$ (can be updated within this step), find $\mathbf{D}$.

Since our dictionary learning algorithm is also composed of these two parts, the following sections will first give the new MCP algorithm to obtain the sparse solution of the training data, and then introduce the alternating optimization algorithm to update the dictionary.

A. Sparse Coding With MCP

Consider solving the optimization problem (1) with ℓ_{P}-norm penalty over the sparse matrix $\mathbf{X}$ while keeping the dictionary $\mathbf{D}$ fixed. This problem can be solved by optimizing each column

x of the sparse matrix $\mathbf{X}$ individually

$$\min_x ||y - \mathbf{D}x||_2^2 \text{ subject to } ||x||_{\mathrm{P}} \leq t \tag{2}$$

where y represents one signal of the training matrix $\mathbf{Y}$. Notice that the aforementioned optimization task can be easily transformed to be

$$\min_x ||y - \mathbf{D}\mathrm{x}||_2^2 + \lambda\, ||x||_{\mathrm{P}}. \tag{3}$$

For an appropriate parameter λ, the two problems are equivalent. If P in the function $||\cdot||_{\mathrm{P}}$ is set to 0, the aforementioned problem is known to be nondeterministic polynomial-time hard (NP-hard) in general [30]. Despite the difficulty to find the best solution, it is possible for greedy methods to seek for the solution that is nearly optimal, under a limited class of dictionaries: $\mathbf{D}$ has to be sufficiently incoherent [20], [21]. One typical greedy algorithm is known as OMP [19]. Assuming the residual signal R_y that is the difference between the training signal y and $\mathbf{D}x$, the OMP algorithm selects at each iteration the best atom $\mathbf{d}_g$ which has the highest correlation with the R_y by searching over the whole dictionary

$$\mathbf{d}_g \;=\; \arg\max_{\mathbf{d}_i} |\langle R_y, \mathbf{d}_i \rangle| \tag{4}$$

where $\mathbf{d}_i$ is the atom of dictionary $\mathbf{D}$ and $\langle \cdot\,,\, \cdot \rangle$ denotes the inner product. After that, constructing a subdictionary $\mathbf{D}_G$ by the selected atoms, the sparse representation x can be iteratively updated by projecting the signal y on the $\mathbf{D}_G$

$$x \;=\; (\mathbf{D}_G^T \mathbf{D}_G)^{-1} \mathbf{D}_G^T y. \tag{5}$$

The major computational cost of the OMP lies in calculating the inner products of all the atoms in the dictionary with the residual signal R_y to obtain *one best* atom in (4). However, a modification of the dictionary structure can greatly reduce the complexity of this part. Furthermore, the convergence of the OMP can be significantly accelerated if more atoms are selected at each iteration. Based on these two key ideas, the MCP algorithm introduces the efficient k-means clustering to construct the dictionary and applies a multiple-selection strategy into the searching process, which allows for several atoms to be chosen in each iteration.

Starting from the existing dictionary $\mathbf{D}$, the k-means approach divides the dictionary into k clusters by grouping similar atoms together, and calculates the prototype atoms to represent each cluster. Rather than using the traditional Euclidean distance, the square of the correlation is chosen as the similarity measure

$$S(\mathbf{d}_i, \mathbf{d}_j) \;=\; |\langle \mathbf{d}_i, \mathbf{d}_j \rangle|^2 \tag{6}$$

which enables each cluster to contain the atoms that are highly positively or negatively correlated. Given a cluster of atoms indexed by the set U_j, the prototype atom should be closest to all the atoms in the cluster using the similarity measure on average. Mathematically, we find an atom $\mathbf{c}_j$ to maximize the sum

$$\sum_{i \in \mathrm{U}_j} |\langle \mathbf{c}_j, \mathbf{d}_i \rangle|^2 = \sum_{i \in \mathrm{U}_j} \mathbf{c}_j^{\mathrm{T}} (\mathbf{d}_i \mathbf{d}_i^{\mathrm{T}}) \mathbf{c}_j = \mathbf{c}_j^T \left(\sum_{i \in \mathrm{U}_j} \mathbf{d}_i \mathbf{d}_i^T \right) \mathbf{c}_j.$$

Part 1 of the MCP algorithm: The k-means clustering process
Input: $\mathbf{D}$, the existing dictionary; k, the number of desired clusters Output: U_j, k clusters atoms' index set; $\mathbf{c}_j^*$, the prototype atoms
Initialization: set k clusters randomly and prototype atom $\mathbf{c}_j^*$ for each cluster by computing the dominant left singular vector of each cluster sub-matrix $\mathbf{D}_{\mathrm{U}_j}$, for $j = 1,\ldots,k$. 1) Re-compute all clusters: for each atom $\mathbf{d}_i$, find its new cluster index as $j^*(\mathbf{d}_i) = \arg\max_j \left\lvert \langle \mathbf{c}_j^*, \mathbf{d}_i \rangle \right\rvert^2$. Then, compute the new clusters U_j as $\mathrm{U}_j = \{i : j^*(\mathbf{d}_i) = j\}$. 2) Update the prototype atoms $\mathbf{c}_1^*,\ldots,\mathbf{c}_k^*$ by calculating the dominant left singular vectors of the new cluster sub-matrices $\mathbf{D}_{\mathrm{U}_1},\ldots,\mathbf{D}_{\mathrm{U}_k}$, respectively. 3) Repeat steps 1)-2) until there is no change for each cluster.

Fig. 1. k-means clustering process of the MCP algorithm.

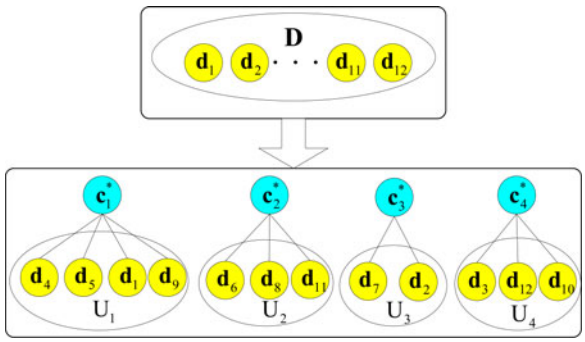


Fig. 2. Simple example of k-means to organize a dictionary $\mathbf{D}$ into four different clusters and represent each cluster by a prototype atom.

The prototype atom $\mathbf{c}_j^*$ is obtained when $\mathbf{c}_j$ equals to the dominant left singular vector of the cluster submatrix $\mathbf{D}_{\mathrm{U}_j}$, consisted of the atoms in the given cluster [31]. Note that the way to compute the prototype atom is derived from the TBP algorithm [23]. The main k-means procedure is to iteratively alternate between 1) reallocation of atoms $\mathbf{d}_i$ to the nearest clusters and 2) computation of prototype atoms $\mathbf{c}_j^*$ to represent the new clusters, as shown in Fig. 1. An illustrative example of k-means clustering to construct a dictionary $\mathbf{D}$ is presented in Fig. 2.

Like many greedy approaches [19], [23]–[26], the MCP algorithm iteratively searches for good atoms to approximate the residual signal R_y. Based on the organized dictionary structure, the searching part starts at the prototype atoms. In this step, we employ a multiple-selection strategy, which selects several prototype atoms whose inner products with the residue signal exceed a specially designed hard threshold. Then, the search goes down to the selected clusters, in which the atoms that lead to the highest amplitude of the inner product with the residual signal are finally chosen. The details of the MCP searching process are described in Fig. 3. As shown in Fig. 3, to obtain multiple atoms in each iteration, the searching space of the MCP algorithm is only on the prototype atoms and the atoms in a partition of clusters. Therefore, the overall atoms searching process is more efficient than the original OMP algorithm.

Part 2 of the MCP algorithm: The searching process

Input: y, the training signal; U_j, k clusters atoms' index set; $\mathbf{c}_j^*$, the prototype atoms
Output: x, the sparse solution of the training signal; I, the final estimate (the index set of the final chosen atoms)

Initialization: set the solution $x_0 = 0$, the residual signal $R_y^0 = y$, the prototype atoms' index set $F_0 = \varnothing$, the multiple-selection atoms' index set $J_s = \varnothing$, the estimate $I_0 = \varnothing$, the counter s = 1, the threshold λ_0 by computing the largest value of $\left|\left\langle R_y^0, \mathbf{c}_j^* \right\rangle\right|$, for $j = 1,\ldots,k$.

1) Compute the inner product M_j of the prototype atoms $\mathbf{c}_j^*$ with the current residual signal R_y^s:
$$M_j = \left|\left\langle R_y^s, \mathbf{c}_j^* \right\rangle\right|.$$
Then, choose several prototype atoms' indexes with the hard-threshold λ_s:
$$F_s = \left\{ j : |M_j| > \lambda_s \right\}.$$
The λ_s is calculated by $\lambda_s = \mu\lambda_{s-1}$, where the μ is a threshold decrease step.
2) Search in the selected clusters for the atoms, which are best correlated with residual signal R_y^s:
$$J_s = \left\{ i : \arg\max_{i \in U_l} \left|\left\langle R_y^s, \mathbf{d}_l \right\rangle\right|, \text{ for } l \in F_s \right\},$$
and merge the new selected atoms' index subset J_s with the previous estimate I_{s-1}:
$$I_s = I_{s-1} \cup J_s.$$
3) Update the sparse approximation x_s by projecting the signal y on the sub-dictionary $\mathbf{D}_{I_s}$:
$$x_s = (\mathbf{D}_{I_s}^T \mathbf{D}_{I_s})^{-1} \mathbf{D}_{I_s}^T y,$$
where $\mathbf{D}_{I_s}$ denotes the $N \times \text{card}(I_s)$ matrix with columns chosen using estimate I_s.
Afterward, update the residual signal by
$$R_y^s = y - \mathbf{D}x_s.$$
4) Check the stopping condition. The procedure stops with the output of final solution x_s when the ℓ_2-norm of the current residual R_y^s reaches an error goal or the number of chosen atoms exceeds a fixed number. If it is not satisfied, we set s = s+1 and go to the step 1).

Fig. 3. Searching process of the MCP algorithm.

Nowadays, many works in the field of sparse approximation have been done on highly redundant dictionaries, which allow for sparser representation of signal. However, the high correlation often exists in these redundant dictionaries, which will mislead the OMP to choose wrong *correlated* atoms. Interestingly, the *k*-means clustering of the dictionary can be thought as a way to artificially lower the coherence. After this clustering process, the atoms within the same cluster are highly correlated, whereas the atoms from different clusters are comparatively uncorrelated. Meanwhile, the correlations among the prototype atoms of different clusters are sufficiently low, ensuring them to constitute a nearly incoherent subdictionary. Thus, even for highly correlated dictionaries, the MCP algorithm can make right choices on the prototype atoms, and then select the atoms that are comparatively uncorrelated.

B. Dictionary Updating With Alternating Optimization

Given a sparse matrix $\mathbf{X}$, the dictionary $\mathbf{D}$ can be updated by solving the following problem:

$$\min_{\mathbf{D}} ||\mathbf{Y} - \mathbf{D}\mathbf{X}||_F^2. \tag{7}$$

In general, this problem can be solved using gradient descent with iterative projection [27] or other least square methods [16], [17], which update all the atoms in the dictionary as a whole. Recently, Aharon *et al*. [18] have adopted the SVD decomposition to optimize (7) by updating one atom together with the corresponding one coefficient vector in $\mathbf{X}$ at a time. More specifically, the quadratic term in (7) is first rewritten as

$$\begin{aligned}\{\mathrm{d}_h, \mathrm{g}_h\} &= \text{Arg}\min_{\mathbf{d}_h, \mathrm{g}_h} \left\| \mathbf{Y} - \sum_{j \neq h} \mathrm{d}_j \mathrm{g}_j^T - \mathrm{d}_h \mathrm{g}_h^T \right\|_F^2 \\ &= \text{Arg}\min_{\mathbf{d}_h, \mathrm{g}_h} ||\mathbf{E}_h - \mathrm{d}_h \mathrm{g}_h^T||_F^2, \quad \text{subject to} \, ||\mathrm{d}_h||_2 = 1\end{aligned} \tag{8}$$

where d_h is the updated atom, g_h^T is the coefficients row vector in $\mathbf{X}$, and $\mathbf{E}_h$ denotes the residual matrix. Then, (8) can be solved by the SVD decomposition of the matrix $\mathbf{E}_h$, and the atom d_h as well as g_h^T can be updated. To avoid the introduction of new nonzeros in $\mathbf{X}$, the update uses only the signal vectors whose current representations use the atom d_h.

In (8), the exact SVD decomposition of the matrix $\mathbf{E}_h$ is quite computationally demanding, especially for the large number of training signals, and therefore, the alternating optimization strategy [32], [33] which aims to efficiently approximate the exact computation is introduced to solve (8). This strategy minimizes the function (8) by alternatively restricting optimizations over the atom d_h and the sparse matrix row g_h^T, i.e., setting the derivation of the function $f(\mathrm{d}_h, \mathrm{g}_h) \equiv ||\mathbf{E}_h - \mathrm{d}_h \mathrm{g}_h^T||_F^2$ with respect to d_h and g_h to zero, i.e.,

$$\begin{cases} \dfrac{\partial f(\mathrm{d}_h, \mathrm{g}_h)}{\partial \mathrm{d}_h} = \left(\mathbf{E}_h - \mathrm{d}_h \mathrm{g}_h^T\right) \mathrm{g}_h = \mathbf{E}_h \mathrm{g}_h - \mathrm{d}_h \|\mathrm{g}_h\|_2^2 = 0 \\ \dfrac{\partial f(\mathrm{d}_h, \mathrm{g}_h)}{\partial \mathrm{g}_h} = \mathrm{d}_h^T \left(\mathbf{E}_h - \mathrm{d}_h \mathrm{g}_h^T\right) = \mathrm{d}_h^T \mathbf{E}_h - \mathrm{d}_h^T \mathrm{d}_h \mathrm{g}_h^T = 0. \end{cases} \tag{9}$$

Since $||\mathrm{d}_h||_2 = 1$, the optimizations over d_h and g_h are obtained by

$$\begin{cases} \mathrm{d}_h = \dfrac{\mathbf{E}_h \mathrm{g}_h}{\|\mathbf{E}_h \mathrm{g}_h\|_2} \\ \mathrm{g}_h = (\mathbf{E}_h)^T \mathrm{d}_h. \end{cases} \tag{10}$$

A single iteration of the aforementioned process can converge to the optimum [32], [33] and provides very close results to the full decomposition of $\mathbf{E}_h$. Besides, we can observe that the aforementioned operation avoids the SVD decomposition of the matrix $\mathbf{E}_h$, as only its products with vectors are needed.

III. 3-D Medical Image Denoising

For 2-D image, Elad and colleagues [14], [15] applied the K-SVD algorithm to learn the dictionary from the image patches of the noisy image directly and achieved state-of-the-art denoising performance. Following this denoising strategy, a simple way

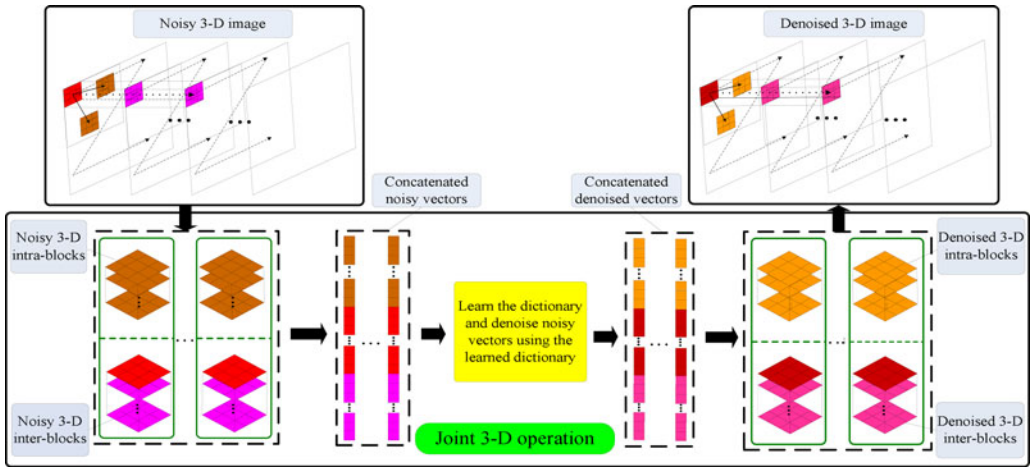


Fig. 4. Flowchart of the proposed joint 3-D operation.

to extend the proposed dictionary learning algorithm to the denoising of 3-D medical image that consists of a large number of slices is to denoise each single slice using separately learned dictionaries. However, in the nearby slices of the 3-D medical image, there exist strong interslice correlations, which can be exploited in 3-D image denoising. In addition, nonlocal methods [8], [28] have shown that the intraslice correlations within each slice could also be effectively employed to enhance the denoising performance. Therefore, this section proposes a joint 3-D operation that applies our learning algorithm to simultaneously utilize the intraslice and interslice correlations. Concretely, a noisy 3-D medical image $\mathrm{Z} \in \mathbb{R}^{Q\times P\times W}$ is composed of W slices $\mathrm{Z}_1,...,\mathrm{Z}_w,...,\mathrm{Z}_W \in \mathbb{R}^{Q\times P}$, which are assumed to be corrupted by white Gaussian noise with variance σ^2 and zero mean. Denote the ith pixel in the slice Z_w as $\mathrm{z}_w[i]$, the block of size $m \times m$ centered on this pixel as z_w^i, and the corresponding 3-D inter-block as $\mathrm{S}_w^{i,\mathrm{inter}}$ of size $m \times m \times l$, which consists of l patches $\mathrm{z}_w^i,...,\mathrm{z}_{w+l-1}^i$ from the nearby slices. The outline of the joint 3-D operation illustrated in Fig. 4 is as follows.

1) For each 2-D block z_w^i in the noisy image, find h blocks with the highest similarity to z_w^j, within a 2-D $N_s \times N_s$ searching window and construct a 3-D intra-block $\mathrm{S}_w^{i,\mathrm{intra}}$ of size $m \times m \times h$ with these found blocks. The similarity is measured as the squared ℓ_2-norm of the intensities difference between two blocks

$$d(\mathrm{z}_w^i, \mathrm{z}_w^j) = ||\mathrm{z}_w^i - \mathrm{z}_w^j||_2^2, j \subset \Lambda \tag{11}$$

where Λ is a set that contains the coordinates of all the searching blocks in the searching window, and the searching step is N_{step}. Meanwhile, extract the ith 3-D interblock $\mathrm{S}_w^{i,\mathrm{inter}}$ from the noisy image. Subsequently, transform each 2-D patch in both the 3-D intrablock and interblock into a vector, and concatenate all these vectors into a single vector y_i.

2) Use the proposed dictionary learning algorithm to train the dictionary on all the concatenated vectors $\{\mathrm{y}_i\}_{i=1}^M$ and simultaneously denoise the vectors by addressing the following optimization problem:

$$\min_{\mathbf{D},\mathrm{x}_i} ||\mathrm{x}_i||_0 \text{ subject to} ||\mathrm{y}_i - \mathbf{D}\mathrm{x}_i||_2^2 \le \varepsilon,\ \forall i = 1, 2, ..., M \tag{12}$$

where M is the total number of processed blocks z_w^i in the 3-D image, x_i is the sparse representation of vector y_i, and $\mathbf{D}\mathrm{x}_i$ is the estimate of the denoised vector. Following [14], [15], the error goal ε in (12) is chosen to be $N^*(C\sigma)^2$, where N^* is the dimension of the processing y_i. As in [15], the parameter C can be automatically tuned according to the rule

$$\mathbb{P}\left(\|\mathrm{y}_i\|_2 \le \sqrt{N^*}C\sigma\right) = 0.93 \tag{13}$$

where $\mathbb{P}$ represents the probability distribution.

3) Use the estimates of denoised vectors to obtain the estimates of each patch in the 3-D intrablocks and interblocks and return the estimates to their original positions.

Eventually, the denoised 3-D image can be reconstructed by averaging the estimates of each pixel. We note that there are two significant motivations to apply the proposed dictionary learning algorithm in the joint 3-D operation. First, since the proposed dictionary learning algorithm has a fast implementation, it is easy to process signals of relatively large size (e.g., 3-D vector here). In addition, the joint 3-D operation can take the advantage of the learning abilities of the dictionary learning algorithm to fully exploit both intraslice and interslice correlations in the 3-D medical image, thereby obtaining better denoising performance.

IV. Experimental Results

This section first evaluates the proposed dictionary learning algorithm on synthetic data. Using synthetic data with random dictionaries helps us to examine the ability of our learning algorithm to recover dictionaries exactly (within an acceptable squared error). Then, the proposed 3-D denoising method is tested on real 3-D medical images, corrupted by Gaussian, Poisson, or speckle noise. The denoising experiments demonstrate the superiority of our denoising method for the 3-D task.

A. Dictionary Learning on Synthetic Data

In these experiments, three random dictionaries of size 100×200, 150×300, and 200×400 are first generated by normalizing dictionaries with i.i.d. uniform random entries. Then, we produce 1500 training samples of dimension 100, 150, and 200, respectively. These samples are created by a linear combination of ten different dictionary atoms, with uniformly distributed i.i.d. coefficients in random and independent locations. After that, white Gaussian noise with varying signal-to-noise ratio (SNR) is added to the resulting training samples.

The dictionary is initialized with the training samples and the number of training iteration in these experiments is set to 50. If the squared error between a learned and true dictionary element is below 0.1, it is classified as correct recovery. To allow a fair comparison, the simulations are repeated eight times and the average dictionary recovery success rates are calculated. In Fig. 5, the average success rates of our method, for the aforementioned three different sizes of dictionaries, are compared with those of the K-SVD. During these experiments, the stopping condition for the OMP in the K-SVD and the MCP in our method is ten atoms. That is, they will stop when the number of selected atoms exceeds ten. The threshold decrease step μ, determining the number of chosen atoms in each iteration of the MCP algorithm, is chosen to 0.90. Through extensive experiments, we have found that as μ increases, the dictionary recovery rate will rise, while our method will become slightly slower. When $\mu = 0.90$, our method achieves a well balance between the dictionary recovery rate and the execution time. The number of the clusters in the k-means clustering is important. In these simulations, we define a variable $R = K/k$, which is a ratio between the number of atom in the dictionary K and the cluster number k. In order to analyze the influence of R to the recovery results, our method is performed under four different values of R on three different dictionaries, as shown in Fig. 5.

As we can see in Fig. 5, our method performs better when R is 3 and 4. In other words, the dictionary recovery rate of our method will rise as the number of cluster increases. In addition, we find that the behaviors of our method under the four different values of R are better than that of the K-SVD in general. Especially in the high noise range (SNR $<$20), the dictionary recovery rate of the K-SVD is very low, whereas our method can still recover the true dictionary in a strong possibility. This is due to the reason that a clustering of the highly coherent random dictionaries may assist the MCP algorithm to find better sparse solutions in the presence of high noise.

In Table I, we also compare the computation time of the K-SVD and our method for the aforementioned simulations. Simulations are done in the environment of an AMD Athlon CPU 2.81 GHz, operating under MATLAB 7.10.0. The execution time reported in this table has been averaged over all the noise levels. As can be seen, our method is about two to three times as fast as the K-SVD. Besides, comparing the computation time of our method on different values of R, it can be found that the running time will slightly increase, as R declines.

To show the complexity advantage of the MCP over the OMP, Fig. 6 gives their averaged execution time for implementing one iteration of the sparse coding operation. The parameter R here is chosen to be 4. It can be seen that the MCP algorithm, though containing the k-means process, runs faster than the OMP and this speed advantage becomes more obvious along with the increase of the dictionary size. In addition, the computation time of the k-means clustering is also added to this comparison, which

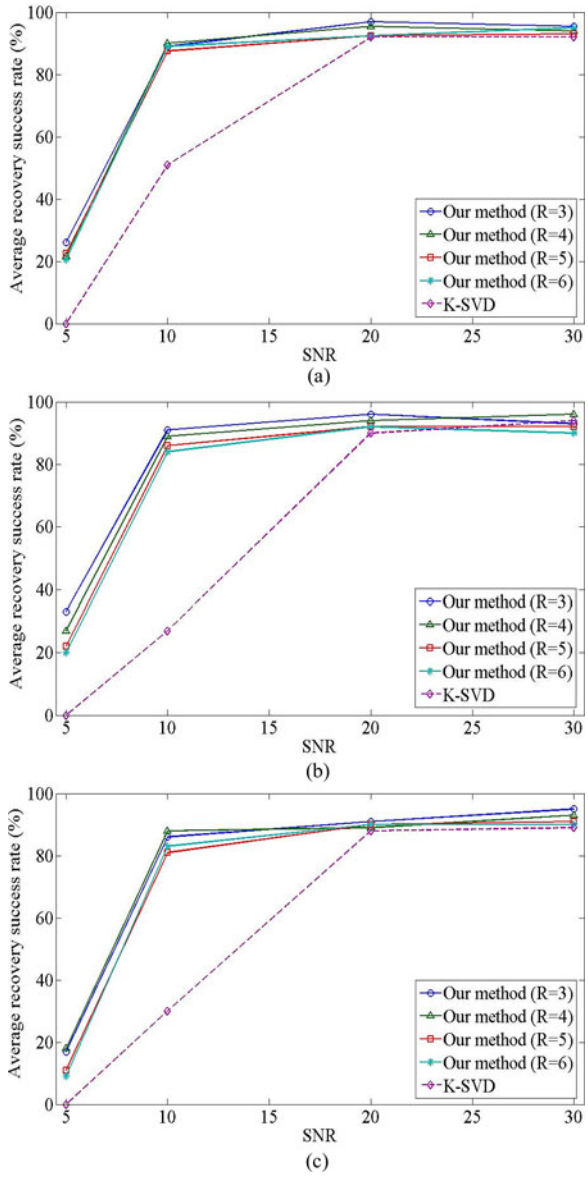


Fig. 5. Comparison of the dictionary recovery success rates for our method and the K-SVD on three different sizes of dictionaries. (a) Dictionary of size 100×200. (b) Dictionary of size 150×300. (c) Dictionary of size 200×400.

TABLE I
COMPARISON OF EXECUTION TIME (IN SECONDS) FOR THE K-SVD AND OUR METHOD ON THE FOUR VALUES OF R

Dictionary size	K-SVD	Our method (R=3)	Our method (R=4)	Our method (R=5)	Our method (R=6)
100×200	262	122	118	109	106
150×300	440	186	175	163	158
200×400	667	279	262	253	246

The execution time in this table is an average of all the noise levels.

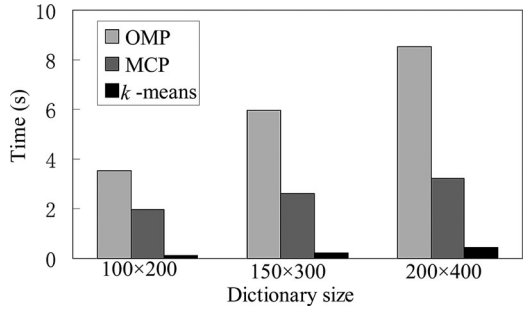


Fig. 6. Comparison of execution time for the OMP, MCP, and k-means clustering on three different sizes of dictionaries. The execution time is averaged on 50 iterations.

demonstrates its low complexity regarding to the whole MCP process. Furthermore, we should note that parallel techniques can be used to further accelerate this k-means clustering process [34].

B. Denoising 3-D Medical Images Corrupted by Gaussian Noise

In this section, we perform denoising experiments on two 3-D CT images[1]: "Male-Head" (size 260 × 190 × 206) and "Female-Ankle" (size 260 × 190 × 150), and two 3-D MR images[2]: "Brain" (size 200 × 200 × 175) and "Heart" (size 200 × 200 × 138). The intensity values of each test image are fitted to [0, 255] for compatibility with image denoising results, and then white Gaussian noise with varying standard deviations of $10 \leq \sigma \leq 100$ is added to the test images. The peak SNR (PNSR) is used as objective denoising measure and the denoising results in these experiments are averaged over eight executions.

First, the proposed method exploiting both the intraslice and interslice correlations is abbreviated as 3-D intraslice and interslice correlations (IAIRSC), and the simplified version of the proposed method, which only utilizes the interslice correlations, is denoted as 3-D intraslice correlations (IRSC). Then these two methods are compared with other four approaches: shape-adaptive discrete cosine transform (SA-DCT) [35], block matching 3-D (BM3D) [36], 2-D K-SVD [14], and 3-D K-SVD. In the 2-D image denoising application, SA-DCT and 2-D K-SVD are two state-of-the-art denoising algorithms. However, since the SA-DCT has not been designed for the 3-D case, our experiments apply the SA-DCT to denoise each slice independently. Likewise, the 2-D K-SVD first learns one dictionary from each slice and then separately deals with each slice using these learned dictionaries. By contrast, the 3-D K-SVD adopts the proposed joint 3-D operation to process a set of slices as a whole and utilizes both intraslice and interslice correlations with the original K-SVD algorithm. In addition, the BM3D adopted in this paper also exploits the intraslice and interslice correlations in the 3-D image through a predictive searching and collaborative filtering. In these tests, the parameters of the

[1]Download at: https://mri.radiology.uiowa.edu//VHDicom/.
[2]Download at: https://imaging.nci.nih.gov/ncia/sessionExpired.jsp.

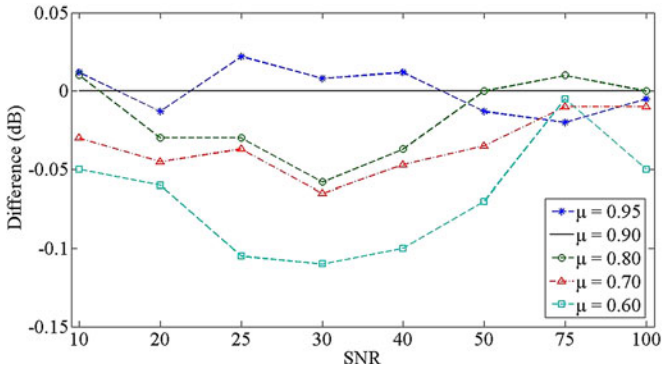


Fig. 7. Comparison of the denoising results (in PSNR) of the 3-D IAIRSC on values μ ranged from 0.60 to 0.95. The reported results are the difference compared to the results when $\mu = 0.90$.

TABLE II
PARAMETERS OF THE 2-D K-SVD, 3-D K-SVD, 3-D IRSC, AND 3-D IAIRSC

Parameters	2-D K-SVD	3-D K-SVD	3-D IRSC	3-D IAIRSC
Temporal block size	—	8×8×8	8×8×8	8×8×8
Spatial block size	8×8	8×8×2	—	8×8×2
Dictionary size	64×100	640×1000	512×1000	640×1000
Initial dictionary	DCT	DCT	DCT	DCT
Dictionary learning iterations	15	15	15	15
Threshold decrease step μ	—	—	0.90	0.90
Parameter R	—	—	4	4
Searching step size N_{step}	—	6	—	6
Searching window size $N_s \times N_s$	—	40×40	—	40×40

SA-DCT and BM3D are set the same as in [35] and [36]. Motivated by results in Fig. 5, the parameter R in our method is set to 4, which reaches a compromise between the performance and speed. The threshold decrease step μ is chosen to 0.90, and Fig. 7 illustrates the effect of the different choices of μ on the denoising performance of the 3-D IAIRSC. As can be observed, the 3-D IAIRSC perform better when μ is set to 0.90 or 0.95. Nevertheless, more atoms can be selected, and thus, the 3-D IAIRSC will be faster, as μ is chosen to 0.90. The choice for the size of 2-D patches ($m \times m$) in the 3-D intrablocks and interblocks is affected by the noise level. Specifically, our method with the large patch performs comparatively better in the high noise range, whereas the sized patch is more suitable for the low noise range. This is because the large patch will create a smoother effect for each pixel which is more appropriate for the high noise condition, and vice versa. However, when such patch size is chosen to 8×8, the proposed method achieves a well balance in the denoising performance across all the noise levels. Therefore, the best choice for the patch is 8×8. The number l of the interblock is set to 8. It is worthwhile to note that since the strong interslice correlations only exist in the nearby slices, our performance will be degraded with the over-small or over-large number of interblocks. The number h of the found intrablock in each searching window is only chosen to 2. This is due to the reason that when such number increases, there will be no obvious improvement in the performance while a high

TABLE III
COMPARISON OF DENOISING RESULTS USING THE SA-DCT [35], BM3D [36], 2-D K-SVD [14], 3-D K-SVD, 3-D IRSC, AND 3-D IAIRSC

Noise level σ	Visible Female-ankle						Visible Male-Head					
	SA-DCT	BM3D	2-D K-SVD	3-D K-SVD	3-D IRSC	3-D IAIRSC	SA-DCT	BM3D	2-D K-SVD	3-D K-SVD	3-D IRSC	3-D IAIRSC
10	39.91	**41.41**	39.70	41.13	41.22	41.10	40.28	**41.74**	40.13	41.48	41.51	41.43
20	36.48	**38.40**	36.12	37.99	38.16	38.18	38.25	**38.41**	36.08	38.00	38.27	38.24
30	34.40	**36.46**	33.76	36.05	36.25	36.32	33.99	**36.24**	33.13	35.95	**36.18**	**36.22**
40	32.95	**35.04**	32.21	34.67	34.90	**35.03**	32.49	**34.70**	31.20	34.50	34.58	**34.67**
50	32.01	33.83	30.43	33.68	33.78	**33.95**	31.39	33.32	29.67	33.28	33.27	**33.48**
75	30.15	29.03	27.84	31.78	31.81	**32.02**	29.59	28.81	27.75	30.72	30.78	**31.07**
100	28.88	27.85	26.31	30.33	30.27	**30.51**	28.36	26.83	26.40	29.57	29.54	**29.85**
Noise level σ	Brain						Heart					
	SA-DCT	BM3D	2-D K-SVD	3-D K-SVD	3-D IRSC	3-D IAIRSC	SA-DCT	BM3D	2-D K-SVD	3-D K-SVD	3-D IRSC	3-D IAIRSC
10	36.56	**37.15**	36.18	36.78	36.81	36.75	38.49	**39.52**	37.98	39.25	39.30	39.22
20	33.23	**34.02**	32.43	33.77	33.76	33.82	35.35	**36.46**	34.74	36.05	36.33	**36.37**
30	31.39	**32.31**	30.31	31.92	32.05	32.13	33.71	**34.75**	33.71	34.29	34.45	34.53
40	30.10	**31.12**	28.76	30.68	30.77	30.89	32.57	**33.63**	31.32	33.13	33.21	33.38
50	29.15	**29.93**	27.42	29.73	29.79	**29.98**	31.64	**32.70**	30.09	32.27	32.38	**32.61**
75	27.49	25.91	25.41	27.91	27.93	**28.15**	28.98	28.27	27.90	30.45	30.48	**30.73**
100	26.23	24.18	23.90	26.81	26.75	**26.97**	28.75	27.56	27.50	29.33	29.30	**29.58**

The best results in each test up to a 0.1 dB difference are labeled in bold.

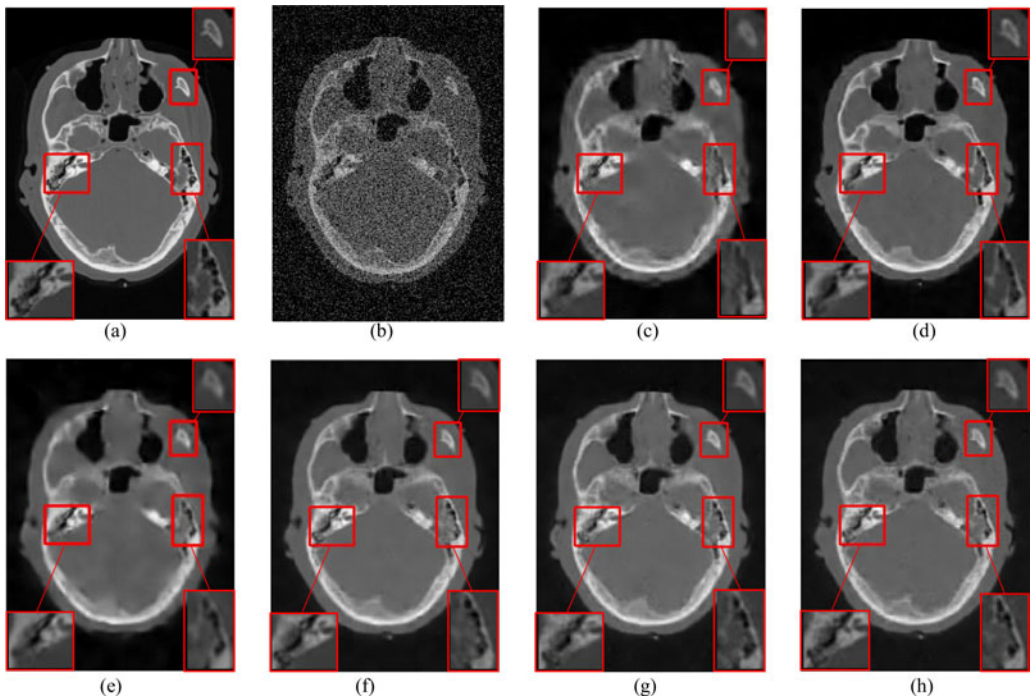


Fig. 8. Denoising results for one slice (#98) of the 3-D medical image "Male-Head" using the SA-DCT [35], BM3D [36], 2-D K-SVD [14], 3-D K-SVD, 3-D IRSC, and 3-D IAIRSC. (a) Original image. (b) Noisy image (PSNR = 18.58 dB, $\sigma = 30$). (c) Denoising result using the SA-DCT (PSNR = 31.52). (d) Denoising result using the BM3D (PSNR = 33.29). (e) Denoising result using the 2-D K-SVD (PSNR = 31.04 dB). (f) Denoising result using the 3-D K-SVD (PSNR = 32.80 dB). (g) Denoising result using the 3-D IRSC (PSNR = 33.26 dB). (h) Denoising result using 3-D IAIRSC (PSNR = 33.32 dB).

computational burden will be created for the dictionary learning part. For completeness, Table II lists the full set of parameters used in 2-D K-SVD, 3-D K-SVD, 3-D IRSC, and 3-D IAIRSC.

In Table III, the denoising results (in PSNR) of the 3-D IRSC and 3-D IAIRSC on the four test images are compared with those from the SA-DCT, BM3D, 2-D K-SVD, and 3-D K-SVD. As can be seen, for the noise level $\sigma \leq 30$, the BM3D provides the best result, while the 3-D IAIRSC is very competitive with the BM3D for the noise level $\sigma = 40,\ 50$, and performs better when the noise level σ is larger than 50. In addition, it is clear that the 3-D IAIRSC and 3-D K-SVD are substantially more effective than the 2-D K-SVD and SA-DCT, with significant gains of about 1 dB on average. These results demonstrate the prominent effect of the joint 3-D operation for these 3-D tasks. Also, a close observation is that for most of the noise levels, the 3-D IAIRSC is superior to the 3-D IRSC due to the utilization of intraslice correlations. Moreover, we can observe that the 3-D IAIRSC outperforms the 3-D K-SVD in general. This is due to the better dictionary trained by our dictionary learning method, especially for the comparatively high noise range ($\sigma \geq 20$). Another main appeal is that in the execution of the experiments of Table III, the 3-D IAIRSC runs about four times faster than the 3-D K-SVD on average, and thus, the high efficiency of our method further makes it comparatively suitable for these 3-D tasks.

In Fig. 8, we provide a visual comparison of various denoising results of one slice from the image "Male-Head". In these figures, the magnified red rectangles are used to show the regions that contain complex detailed information. It can be seen from these figures that the BM3D, 3-D IRSC, and 3-D IAIRSC are generally better than other three methods in terms of the preservation of fine details and reduction of the blurring. Furthermore, compared to the 3-D IRSC and BM3D, the 3-D IAIRSC outputs smoother surfaces in homogeneous regions and shows fewer artifacts.

C. Denoising 3-D CT Images Corrupted by Poisson Noise

This section extends our method to denoise the Poisson noise [37], which is easily created in imaging process of some modalities, such as CT. Unlike Gaussian noise, removing the Poisson noise in the signal is not an easy work, since the parameter of the Poisson probability density function is a function of the underlying signal intensity. To overcome this complication, a variance stability transform (VST) $S = 2\sqrt{Z + (3/8)}$ [38] is applied to the input image Z. Through the VST transformation, the noise with Poisson distribution is converted into noise with nearly Gaussian distribution. Then, our method can be used to denoise the transformed image with the same parameters in the aforementioned section and the denoised result can be eventually transformed to the reconstructed image using an inverse transformation $\hat{Z} = (1/4)\,\hat{S}^2 - (3/8)$. Similarly, we also apply the BM3D [36] to denoise Poisson noise with the same VST operation. Fig. 9 shows one slice from the CT image[3] "Male-Pelvis" corrupted by Poisson noise, and the corresponding slices from denoised results obtained by the BM3D and our method (3-D IAIRSC). As can be seen from the denoised results, our method outperforms the BM3D in terms of PSNR. In addition, our method exhibits fewer artifacts (e.g., regions within the red rectangles).

D. Denoising Real Noisy 3-D Ultrasound Images

In this section, we test the proposed method on a real noisy 3-D "liver" ultrasound image, which is available on Cambridge University website[4]. Since the ultrasound images are often assumed to be contaminated with the speckle noise [39], the test image is first processed by a logarithm transform, which converts the multiplicative speckle noise into the additive Gaussian noise [9], [40]. In the log-transformed image, the noise level can be simply estimated using a Gaussian noise estimation technique [41]. After that, our method is used to train the dictionary on the transform image and denoise this image. Finally, the enhanced image is obtained by applying an exponential function to the denoised result in the transformed domain. In this experiment, both the mean-to-standard-deviation ratio (MSR) [42] and contrast-to-noise ratio (CNR) [43] are adopted as the objective criteria to evaluate the quality of the denoised image. The MSR

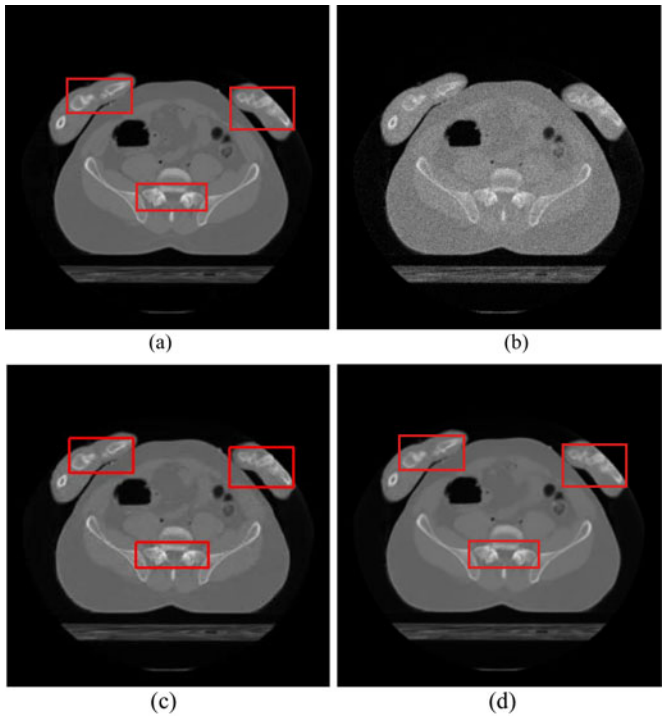


Fig. 9. Denoising results for one slice (#100) of the CT image "Male-Pelvis" using the BM3D [36] and our method (3-D IAIRSC). (a) Original image. (b) Noisy image (PSNR = 32.09 dB). (c) Denoising result using the BM3D (PSNR = 43.22). (d) Denoising result using our method (3-D IAIRSC) (PSNR = 43.88 dB).

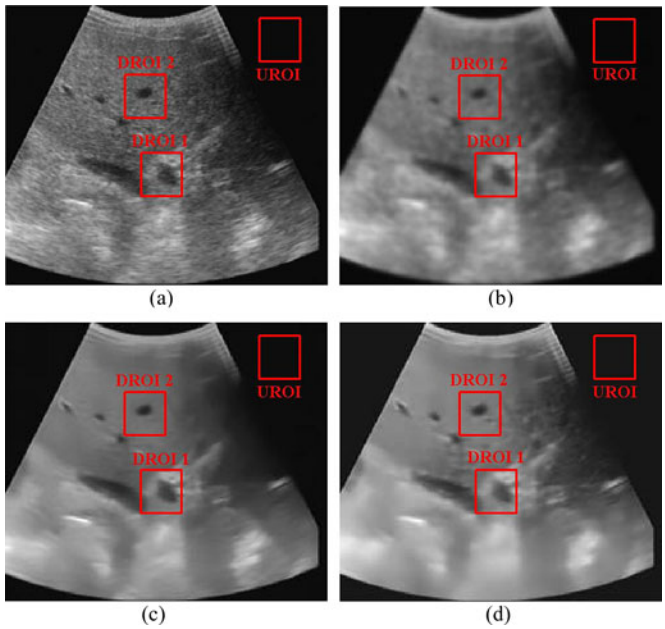


Fig. 10. Denoising results for one slice (#109) of the real noisy 3-D liver ultrasound image using the SRAD [44], OBNLM [45], and our method (3-D IAIRSC). The DROI and UROI used to compute the MSR and CNR indexes (listed in Table IV) are marked with red rectangles. (a) Original real noisy image. (b) Denoising result using the SRAD. (c) Denoising result using the OBNLM. (d) Denoising result using our method (3-D IAIRSC).

[3]Download at: https://mri.radiology.uiowa.edu//VHDicom/.
[4]Download at: http://mi.eng.cam.ac.uk/∼rwp/stradwin/.

TABLE IV
MSR AND CNR RESULTS OF ONE SLICE (#109) OF THE REAL NOISY 3-D LIVER ULTRASOUND IMAGE BY THE SRAD [44], OBNLM [45], AND OUR METHOD (3-D IAIRSC)

Method	DROI 1		DROI 2	
	CNR	MSR	CNR	MSR
Original region	4.88	4.30	6.36	5.64
SRAD [44]	5.82	4.66	8.23	6.61
OBNLM [45]	5.98	4.77	**8.78**	7.18
Our method (3-D IAIRSC)	**6.11**	**5.36**	8.56	**7.30**

The best results in this table are labeled in bold.

is computed in the desired region of interest (DROI)

$$\mathrm{MSR} = \frac{\mu_d}{\sigma_d} \tag{14}$$

where μ_d and σ_d are the mean and the standard deviation in the DROI. The CNR represents the contrast between the DROI and the undesired region of interest (UROI), which is defined as

$$\mathrm{CNR} = \frac{|\mu_d - \mu_u|}{\sqrt{0.5(\sigma_d^2 + \sigma_u^2)}} \tag{15}$$

where μ_u and σ_u are the mean and the standard deviation in the UROI.

In Fig. 10, we compare our method (3-D IAIRSC) with two well-known speckle denoising approaches: speckle reducing anisotropic diffusion (SRAD) [44] and optimized Bayesian nonlocal-means (OBNLM) [45]. In this figure, two DROIs and one UROI are sampled to compute the MSR and CNR, as shown in Table IV. It can be observed that our method is very competitive with the OBNLM and much better than the SRAD in terms of MSR, CNR, and visual quality.

We should note that our method can also be applied to denoise the 3-D MR images corrupted by the Rician noise through a VST [46], which converts noise with Rician distribution into noise with nearly Gaussian distribution. Actually, our preliminary tests on Rician denoising have already achieved promising results, but we do not report these results in this paper to save space.

V. CONCLUSION

In this paper, we have presented an efficient dictionary learning algorithm and applied it to 3-D medical image denoising with a join 3-D operation. Our learning algorithm is achieved by employing the MCP to perform the sparse coding and using the alternating optimization to update the dictionary. Compared to the OMP, the MCP algorithm gives better sparse solutions and achieves a great reduction on the computation complexity due to a dictionary structuring strategy and a multiple-selection strategy in the atom search process. The alternating optimization is a quick approximation way for updating dictionary, and, thus, further accelerates the whole dictionary learning algorithm. In addition, the joint 3-D operation takes the advantage of the learning abilities of our dictionary training algorithm to sufficiently exploit both intraslice and interslice correlations in the medical slices, thereby obtaining better estimates of them. Experiments on synthetic data demonstrate the effectiveness and efficiency of our dictionary learning algorithm, while real 3-D medical image experiments show the superiority of our denoising method in terms of both PSNR and visual quality.

Instead of adopting the fixed square patches, part of our ongoing work is to apply the shape-adaptive patches as in the SA-DCT [35] into our denoising method. Furthermore, in the future, there is a strong incentive to apply our dictionary learning algorithm to other large-scale applications (e.g., 3-D medical image deblurring, super-resolution, and segmentation).

ACKNOWLEDGMENT

The authors would like to thank excellent reviewers for their detailed reviews and valuable suggestions that greatly improved this paper.

REFERENCES

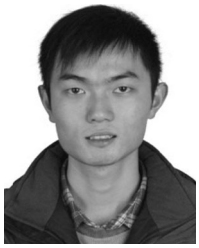
[1] N. V. Thakor and Y. S. Zhu, "Application filtering to EGG analysis: Noise cancellation and arrhythmia detection," *IEEE Trans. Biomed. Eng.*, vol. 38, no. 8, pp. 785–794, Aug. 1991.
[2] J. Oster, O. Pietquin, M. Kraemer, and J. Felblinger, "Nonlinear Bayesian filtering for denoising of electrocardiograms acquired in a magnetic resonance environment," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 7, pp. 1628–1638, Jul. 2010.
[3] A. Wong and A. K. Mishra, "Quasi-Monte Carlo estimation approach for denoising MRI data based on regional statistics," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 4, pp. 1076–1083, Apr. 2011.
[4] E. Causevic, R. E. Morley, M. V. Wickerhauser, and A. E. Jacquin, "Fast wavelet estimation of weak biosignals," *IEEE Trans. Biomed. Eng.*, vol. 52, no. 6, pp. 1021–1032, Jun. 2005.
[5] C. S. Drapaca, "A nonlinear total variance-based denoising method with two regularization parameters," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 3, pp. 582–586, Mar. 2009.
[6] H. Rabbani, R. Nezafat, and S. Gazor, "Wavelet-domain in medical image denoising using bivariate Laplacian mixture model," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 12, pp. 2826–2837, Nov. 2009.
[7] J. W. Lin, A. F. Laine, and S. R. Bergmann, "Improving PET-based physiological quantification through methods of wavelet denoising," *IEEE Trans. Biomed. Eng.*, vol. 48, no. 2, pp. 202–212, Feb. 2001.
[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
[9] H. Rabbani, M. Vafadust, P. Abolmaesumi, and S. Gazor, "Speckle noise reduction of medical ultrasound images in complex wavelet domain using mixture priors," *IEEE Trans. Biomed. Eng.*, vol. 55, no. 9, pp. 2152–2160, Sep. 2008.
[10] B. A. Olshausen and B. 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, Jun. 1996.
[11] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Jul. 2009.
[12] Y. Guo, S. Ruan, J. Landré, and J.-M. Constans, "A sparse representation method for magnetic resonance spectroscopy quantification," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 7, pp. 1620–1627, Jul. 2010.
[13] Y. Q. Li, P. Namburi, Z. L. Yu, C. Guan, J. F. Feng, and Z. H. Gu, "Voxel selection in fMRI data analysis based on sparse representation," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 10, pp. 2439–2451, Oct. 2009.
[14] 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, Dec. 2006.
[15] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Trans. Image Process.*, vol. 17, no. 1, pp. 53–69, Jan. 2008.
[16] K. Engan, S. O. Aase, and J. Hakon Husoy, "Method of optimal directions for frame design," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Washington, DC, Mar. 1999, vol. 5, pp. 2443–2446.
[17] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning unions of orthonormal bases with threshold thresholded singular value decomposition," in *Proc. IEEE Conf. Acoust., Speech, Signal Process.*, Philadelphia, PA, Mar. 2005, vol. 5, pp. 293–296.

[18] M. Aharon, M. Elad, and A. M. Bruckstein, "The K-SVD: An algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.
[19] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Annu. Asilomar Conf. Signals, Syst. Comput.*, Stanford, CA, Nov. 1993, vol. 1, pp. 40–44.
[20] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, Oct. 2004.
[21] D. L. Donoho, M. Elad, and V. N. Temlyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.
[22] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 129–137, Mar. 1982.
[23] P. Jost, P. Vandergheynst, and P. Frossard, "Tree-based pursuit: Algorithm and properties," *IEEE Trans. Signal Process.*, vol. 54, no. 12, pp. 4685–4697, Dec. 2006.
[24] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck. (2006) Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit [Online]. Available: http://www-stat.stanford.edu/∼donoho/Reports/2006/StOMP-20060403.pdf, Preprint.
[25] A. Rahmoune, P. Vandergheynst, and P. Frossard, "The M-term pursuit for image representation and progressive compression," in *Proc. IEEE Int. Conf. Image Process.*, Genoa, Italy, Sep. 2005, vol. 1, pp. 73–76.
[26] T. Gan, Y. He, and W. Zhu, "Fast M-term pursuit for sparse image representation," *IEEE Signal Process. Lett.*, vol. 15, no. 1, pp. 116–119, Jan. 2008.
[27] Y. Censor and S. A. Zenios, *Parallel Optimization: Theory, Algorithms and Applications*. New York: Oxford Univ. Press, 1997.
[28] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, Jul. 2005.
[29] L. Fang and S. Li, "An efficient dictionary learning algorithm for sparse representation," in *Proc. Chin. Conf. Pattern Recognit.*, Chongqing, China, Oct. 2010, pp. 1–5.
[30] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *J. Construct. Approx.*, vol. 13, no. 1, pp. 57–58, Jan. 1997.
[31] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, U.K.: Cambridge Univ. Press, 1985.
[32] J. C. Bezdek and R. J. Hathaway, "Convergence of alternating optimization," *Neural Parallel Sci. Comput.*, vol. 11, no. 4, pp. 351–368, Dec. 2003.
[33] J. C. Bezdek and R. J. Hathaway, "Some notes on alternating optimization," in *Proc. Annu. Fundam. Sci. Semin. Int. Conf. Fuzzy Syst.*, Calcutta, India, Feb. 2002, vol. 2275, pp. 187–195.
[34] K. Stoffel and A. Belkoniene, "Parallel K-means clustering for large data sets," in *Proc. EuroPar Parallel Process.*, Toulouse, France, Aug.1999, pp. 1451–1454.
[35] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images," *IEEE Trans. Image Process.*, vol. 16, no. 5, pp. 1395–1411, May 2007.
[36] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3D transform-domain collaborative filtering," in *Proc. 15th Eur. Signal Process. Conf.*, Poznan, Poland, Sep. 2007, pp. 145–149.
[37] B. R. Whiting, P. Massoumadeh, O. A. Earl, J. A. O'Sullivan, D. L. Snyder, and J. F. Williamson, "Properties of preprocessed sonogram data in X-ray computed tomography," *Med. Phys.*, vol. 33, no. 9, pp. 3290–3303, Sep. 2006.
[38] J.-L. Starck, F. Murtagh, and A. Bijioui, *Image Processing and Data Analysis: The Multiscale Approach*. Cambridge, U.K.: Cambridge Univ. Press, 1988.
[39] X. Zong, A. F. Laine, and E. A. Geiser, "Speckle reduction and contrast enhancement of echocardiograms via multi-scale nonlinear processing," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 532–540, Aug. 1998.
[40] A. Achim, A. Bezerianos, and P. Tsakalides, "Novel Bayesian multiscale method for speckle removal in medical ultrasound images," *IEEE Trans. Med. Imag.*, vol. 20, no. 5, pp. 772–783, May 2001.
[41] A. Danielyan and A. Foi, "Noise variance estimation in nonlocal transform domain," in *Proc. Int. Workshop Local Non-Local Approx. Image Process.*, Tuusula, Finland, Aug. 2009, pp. 41–45.
[42] G. Cincotti, G. Loi, and M. Pappalardo, "Frequency decomposition and compounding of ultrasound medical images with wavelets packets," *IEEE Trans. Med. Imag.*, vol. 20, no. 8, pp. 764–771, Aug. 2001.
[43] P. Bao and L. Zhang, "Noise reduction for magnetic resonance images via adaptive multiscale products thresholding," *IEEE Trans. Med. Imag.*, vol. 22, no. 9, pp. 1089–1099, Sep. 2003.
[44] Y. Yu and S. T. Acton, "Speckle reducing anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 11, no. 11, pp. 1260–1270, Nov. 2002.
[45] P. Coupé, P. Hellier, C. Kervrann, and C. Barillot, "Nonlocal means-based speckle filtering for ultrasound images," *IEEE Trans. Image Process.*, vol. 18, no. 10, pp. 2221–2229, Oct. 2009.
[46] A. Foi, "Noise estimation and removal in MR imaging: The variance-stabilization approach," in *Proc. IEEE Int. Symp. Biomed. Eng.*, Chicago, IL, Apr. 2011, pp. 1809–1814.

Shutao Li (M'07) received the B.S., M.S., and Ph.D. degrees in electrical engineering from the Hunan University, Changsha, China, in 1995, 1997, and 2001, respectively.

In 2001, he joined the College of Electrical and Information Engineering, Hunan University. From May 2001 to October 2001, he was a Research Associate in the Department of Computer Science, Hong Kong University of Science and Technology. From November 2002 to November 2003, he was a Postdoctoral Fellow at the Royal Holloway College, University of London, doing research with Prof. J. Shawe-Taylor. During April 2005 to June 2005, he was a Visiting Professor in the Department of Computer Science, Hong Kong University of Science and Technology. He is currently a Full Professor in the College of Electrical and Information Engineering, Hunan University. He has authored or coauthored more than 130 refereed papers. His research interests include information fusion, image processing, and pattern recognition.

Dr. Li has won two 2nd Grade National Awards at Science and Technology Progress of China in 2004 and 2006. From 2007 to 2008, he served as a member in Neural Networks Technical Committee.



Leyuan Fang (S'10) received the B.S. degree in electrical engineering from the Hunan University of Science and Technology, Xiangtan, China, in 2008. Since 2008, he has been working toward the Ph.D. degree in the College of Electrical and Information Engineering, Hunan University, Changsha, China

Since September 2011, he has been a Visiting Ph.D. Student in the Department of Ophthalmology, Duke University, Durham, NC, supported by the China Scholarship Council. His research interests include sparse representation and multiresolution analysis applied to biomedical signal/images and remote sensing images (denoising, change detection, and segmentation).

Mr. Fang has won the Scholarship Award for Excellent Doctoral Student granted by Chinese Ministry of Education in 2011.

Haitao Yin received the B.S. and M.S. degrees in applied mathematics from the College of Mathematics and Econometrics, Hunan University, Changsha, China, in 2007 and 2009, respectively. He is currently working toward the Ph.D. degree in the College of Electrical and Information Engineering, Hunan University.

His research interests include image processing, sparse representation, and pattern recognition.