Signal Denoising With Random Refined Orthogonal Matching Pursuit

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

***Abstract*—In this paper, an efficient sparse recovery algorithm called random refined orthogonal matching pursuit (RROMP) is proposed for signal denoising. Given a noisy signal, the RROMP algorithm first generates several sparse representations of it by applying a multi-selection strategy and a false discovery rate (FDR) control, instead of seeking the sparsest one. The multi-selection strategy accelerates the whole process of generating the representations, while the FDR control enables each representation to be competitive. Then the generated representations are averaged to form a more accurate estimate in the sense of mean-square-error (MSE). Our experiments on both synthetically generated signals and natural images demonstrate the superiority of the RROMP algorithm.**

***Index Terms*—False discovery rate (FDR), minimum-mean-squared-error (MMSE), random refined orthogonal matching pursuit (RROMP), signal denoising, sparse representations.**

I. Introduction

SIGNAL denoising is one of the important topics in the field of instrumentation and measurement, because it can significantly reduce uncertainties in the measurement procedures, thus enhancing the accuracy and credibility of many systems. Numerous methods have been explored for this problem, such as adaptive filters [1], [2], statistical estimators [3], [4], transform-domain methods [5]–[10], etc.

Recently, sparse representation theory attracts many attentions. Sparse representation models signals as sparse linear combinations of atoms from a dictionary $\mathbf{D} \in \mathbb{R}^{N\times K}$ [11]–[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 this dictionary cannot sparsely represent its noisy parts. Formally, the sparse model for the denoising problem is given by

$$\hat{\alpha} = \arg\min_{\alpha} \|\alpha\|_0 \quad \text{subject to} \quad \|\mathrm{Y} - \mathbf{D}\alpha\|_2 \le \mu \qquad (1)$$

where $\hat{\alpha}$ is the sparse representation of Y, μ is the error tolerance, and the ℓ_0 penalty function $\|\cdot\|_0$ counts the non-zero coefficients in the representation.

Problem (1) is known to be nondeterministic polynomial-time hard (NP-hard) in general [14], and thus designing efficient computational methods for its approximation is a fundamental question in the field. Common strategies are typically based on convex relaxation and greedy pursuit. Convex relaxation methods, such as the basis pursuit (BP) [15] and least angle regression (LARS) [16], approximate the combinatorial ℓ_0 penalty term in (1) by a ℓ_1 norm. Greedy pursuit methods, which sequentially select atoms based on some greedy selection rules, include the matching pursuit (MP) [17], orthogonal matching pursuit (OMP) [18], stage-wise orthogonal matching pursuit (StOMP) [19], and compressive sampling matching pursuit (CoSaMP) [20], among others. The greedy methods appeal in their simplicity and low complexity. However, they become less effective when the number of atoms in the solution increases [11], [21]–[23]. This also limits the denoising performance of these algorithms [11], [21]–[23].

Recently, Elad and Yavneh [24] showed that sparse representations of a signal may be valuable even if they are *not the sparsest ones possible*, and moreover, that a plurality of such representations may be jointly more powerful than any single one. Following these ideas, the authors introduced the random orthogonal matching pursuit (RandOMP) algorithm which randomly computes a variety of sparse representations for a given signal. The algorithm produces random sparse representations which obey a specific prior distribution, and whose mean asymptotically converges to the minimum-mean-squared-error (MMSE) estimator [24], [25]. Thus, averaging the outputs of the RandOMP algorithm provides a good signal estimator in a mean-square-error (MSE) sense.

In this paper, we propose an improvement to the RandOMP algorithm, named random refined orthogonal matching pursuit (RROMP). The aim of the refined algorithm is to accelerate the process by improving the random selection step. The core of the RROMP algorithm is a stochastic selection process which allows for multiple atoms to be added each iteration, in a manner somewhat resembling the StOMP algorithm [19] which improves on the ordinary OMP by employing a deterministic multiple-selection process. At each RROMP iteration, a set of candidate atoms is randomly selected from the dictionary. Then, a false discovery rate (FDR) control [26]–[28] is applied to determine the most significant atoms in the set. The overall process requires less iterations and therefore is faster than the RandOMP, especially for solving large-scale problems. Furthermore, the resulting representations are more competitive

Manuscript received November 30, 2010; revised March 13, 2011; accepted April 27, 2011. Date of publication June 23, 2011; date of current version December 8, 2011. This work was supported in part by the National Natural Science Foundation of China (60871096 and 60835004), the Ph.D. Programs Foundation of Ministry of Education of China (200805320006), the Key Project of Chinese Ministry of Education (2009-120), and the Open Projects Program of National Laboratory of Pattern Recognition, China. The Associate Editor coordinating the review process for this paper was Dr. Domenico Grimaldi.

The authors are with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: shutao_li@yahoo.com.cn).

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/TIM.2011.2157547

than the ones generated by the RandOMP, thus improving the denoising results, as shown in our simulations.

The rest of this paper is organized as follows. In Section II, we introduce the RROMP algorithm and discuss our sampling strategy. Experimental results on both synthetic signals and natural images are presented in Section III. Finally, Section IV concludes the paper and suggests future works.

II. Random Refined Orthogonal Matching Pursuit

The signal denoising problem assumes a given measurement signal $\mathrm{Y} \in \mathbb{R}^N$ obtained from the clean signal $\mathrm{X} \in \mathbb{R}^N$ by a contamination ω of the form $\mathrm{Y} = \mathrm{X} + \omega$. In this paper, ω is restricted to be a white Gaussian noise with zero mean and standard deviation σ. We assume that the signal X has a sparse representation over the known dictionary $\mathbf{D} \in \mathbb{R}^{N\times K}$, and thus, there exists a sparse vector α such that $\mathrm{X} = \mathbf{D}\alpha$. In this setting, the typical denoising process seeks the sparsest representation that reproduces Y up to the allowed distortion.

Alternatively, the recently proposed RandOMP algorithm [24] suggests approximating X by computing *several* possible sparse representations $\{\alpha^i\}$ of Y, and averaging the estimates $\{\mathbf{D}\alpha^i\}$. The process produces an approximation of the MMSE estimator of X, and thus improves the MSE of the result compared to a deterministic selection method. This process, however, requires quite a few executions of the RandOMP method, and thus the overall process is relatively costly compared to simpler greedy options. The RROMP algorithm aims to reduce the complexity of this process by improving the RandOMP sampling procedure. This is achieved by employing a multiple-selection process which allows for several atoms to be added at once, thus reducing the number of iterations.

A. MMSE Estimator and RandOMP

Before introducing the RROMP algorithm, this section gives a brief review of the MMSE estimator and RandOMP. Recall that the denoising goal in terms of the MSE is to seek an estimation $\hat{\mathrm{X}}$ which can minimize the following formula:

$$\mathrm{MSE}_{\mathrm{Y}} = \mathrm{E}\left(\|\hat{\mathrm{X}} - \mathrm{X}\|_2^2 | \mathrm{Y}\right) \tag{2}$$

with E denoting expectation.

Given a dictionary $\mathbf{D} \in \mathbb{R}^{N\times K}$, let Ω denote the set of all 2^K sub-dictionaries, and $\mathbf{U}$ be one sub-dictionary. Based on Gaussian assumptions, Elad and Yavneh derive a MMSE estimator for $\hat{\mathrm{X}}$ [24]

$$\hat{\mathrm{X}}^{\mathrm{MMSE}} = \frac{a^2}{\sum_{\mathbf{U}'\in\Omega_g} \exp\left\{-\frac{a^2}{2\sigma^2}\cdot\|\mathrm{Y}-\mathrm{Y}_{\mathbf{U}'}\|_2^2\right\}} \cdot \sum_{\mathbf{U}\in\Omega_g} \exp\left\{-\frac{a^2}{2\sigma^2}\cdot\|\mathrm{Y}-\mathrm{Y}_{\mathbf{U}}\|_2^2\right\}\mathrm{Y}_{\mathbf{U}}. \tag{3}$$

In this expression, $\mathrm{Y}_{\mathbf{U}}$ denotes the orthogonal projection of the vector Y onto the subspace spanned by the columns of $\mathbf{U}$, Ω_g is a sub-set of Ω, g represents the sparsity factor which means each sub-dictionary $\mathbf{U}$ in Ω_g only has g atoms, and a^2 equals to $(\sigma_{\mathrm{X}}^2)/(\sigma_{\mathrm{X}}^2+\sigma^2)$, where σ_{X} is the variance of the non-zero entries of the representation of the original signal X. Then, the MSE of this estimator is given by

$$\mathrm{MSE}_{\mathrm{Y}}^{\mathrm{MMSE}} = ga^2\sigma^2 - \|\hat{\mathrm{X}}^{\mathrm{MMSE}}\|_2^2 + \sum_{\mathbf{U}\in\Omega_g}\|a^2\mathrm{Y}_{\mathbf{U}}\|_2^2 P(\mathbf{U}|\mathrm{Y}). \tag{4}$$

By observing (3), the MMSE estimator is a weighted average of the orthogonal projections of the noisy signal on all the feasible subspaces. Motivated by the sparsity and redundancy model, the MMSE estimator can also be regarded as a summation of various sparse representations weighted by $\exp\{-(a^2/2\sigma^2)\cdot\|\mathrm{Y}-\mathrm{Y}_{\mathbf{U}}\|_2^2\}$ in (3). That is to say, the atoms can be selected as candidate representations with a probability which is proportional to $\exp\{-(a^2/2\sigma^2)\cdot\|\mathrm{Y}-\mathrm{Y}_{\mathbf{U}}\|_2^2\}$. Since $\exp\{-a^2\cdot\|\mathrm{Y}\|_2^2\}$ is a constant and the columns in $\mathbf{U}$ are normalized, this probability is also proportional to $\exp\{(a^2/2\sigma^2)\cdot|\mathrm{Y}^T d_i|^2\}$, where d_i denotes the ith atom in $\mathbf{U}$, $i = 1,\ldots,g$.

Following this line of reasoning, the RandOMP algorithm slightly modifies the sampling strategy of the original OMP [18] by selecting one atom every iteration according to the probability $\exp\{(a^2/2\sigma^2)\cdot|\mathrm{Y}^T d_i|^2\}$. By running this algorithm several times, a collection of different representations $\{\alpha^i\}$ can be obtained, and averaging the estimates $\{\mathbf{D}\alpha^i\}$ is a way to approximate the MMSE estimator.

B. Proposed RROMP

The weights in (3) depend on the distances between the noisy signal Y and the sub-spaces $\mathbf{U}$, many of which are with small amplitude. This makes that the randomly selected atoms consist of a few "significant" atoms, and numerous "ordinary" ones. The "ordinary" atoms have little effect on the final results while still create large computational cost. Therefore, a key idea of the RROMP algorithm is to adopt an adaptive sampling strategy, which can further refine the selected atoms.

The scheme of the RROMP algorithm is presented in Fig. 1, which consists of the following eight steps.

Step 1: Set initial solution $\alpha_0 = 0$, initial residual $\mathrm{r}_0 = \mathrm{Y}$, initial estimate $\mathrm{I}_0 = \phi$, initial randomly selected set $\mathrm{H}_0 = \phi$, initial refined set $\mathrm{J}_0 = \phi$ and counter $s = 1$. The dictionary $\mathbf{D}$ is assumed to be known.

Step 2: Apply the $\mathbf{D}$ to the current residual r_{s-1}, getting a vector of residual correlations c_s

$$\mathrm{c}_s(k) = d_k^{\mathrm{T}}\mathrm{r}_{s-1}, \qquad k = 1,\ldots,K \tag{5}$$

for each corresponding atom (K is the number of atoms in $\mathbf{D}$, d_k denotes the kth atom in $\mathbf{D}$, $\mathrm{c}_s(k)$ stands for the kth element in c_s).

Step 3: Use the residual correlations c_s to form a set of probabilities $P(k) = \exp\{(a^2/2\sigma^2)\cdot(|\mathrm{c}_s(k)|^2/\|d_k\|_2^2)\}$. Then the probability P is employed to randomly select m elements[1] (representing the corresponding atoms) in c_s, yielding an index set H_s

$$\mathrm{H}_s = \{h|\mathrm{random}\,(\mathrm{c}_s(h))\}\,. \tag{6}$$

[1]Note that, one element cannot be chosen twice. To avoid this, if the element has been chosen, we would set its probability zero.

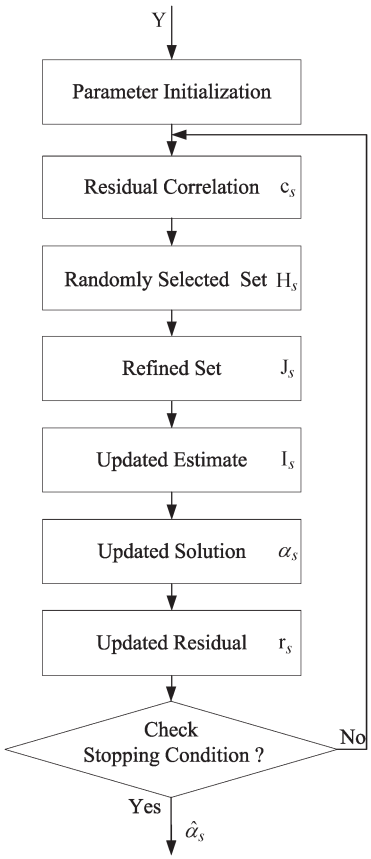


Fig. 1. Scheme of the RROMP algorithm.

Step 4: Apply the FDR control to the residual correlations c_s, obtaining an adaptive threshold T_s. Afterward, a set J_s of more significant indexes is obtained from the randomly selected set H_s with the threshold T_s

$$\mathrm{J}_s = \{j | \mathrm{c}_s\left(\mathrm{H}_s(j)\right) > T_s\}. \tag{7}$$

Step 5: Update the estimate I_s by merging the new index subset J_s with the previous estimate I_{s-1}

$$\mathrm{I}_s = \mathrm{I}_{s-1} \cup \mathrm{J}_s. \tag{8}$$

Step 6: Update the approximation α_s by projecting the vector Y on the columns of $\mathbf{D}$ belonging to the estimate I_s

$$\alpha_s = \left(\mathbf{D}_{\mathrm{I}_s}^T \mathbf{D}_{\mathrm{I}_s}\right)^{-1} \mathbf{D}_{\mathrm{I}_s}^T \mathrm{Y} \tag{9}$$

where $\mathbf{D}_{\mathrm{I}_s}$ denotes the $N \times |\mathrm{I}_s|$ matrix with columns chosen using index set I_s.

Step 7: Construct a new $\mathbf{D}\alpha_s$ using the α_s. Then, the current residual can be calculated by

$$\mathrm{r}_s = \mathrm{Y} - \mathbf{D}\alpha_s. \tag{10}$$

Step 8: Check the stopping condition. The procedure stops with the output of final solution $\hat{\alpha}_s$ when the ℓ_2 norm of the current residual r_s reaches an error goal or the number of iteration exceeds a fixed number. If it is not satisfied, we set $s = s + 1$ and go to the step 2.

It is important to note that the probability distribution P used in the step 3 is the same as the one employed in the RandOMP algorithm, which ensures that the mean of the produced representations will converge to the MMSE estimator. However, unlike the process in the RandOMP which only selects one atom each time, the step 3 selects many atoms according to the probability P, and then the FDR control is used to determine the significant ones. Thus, the RROMP algorithm can get multiple competitive atoms in each iteration and need less iterations than the RandOMP.

C. Refinement of the Random Set

In this section, we describe how to apply the FDR control to obtain an adaptive threshold T_s, which is used to refine the random set in the step 4 of the RROMP algorithm.

The FDR control is a statistical method that can separate "significant" elements from "ordinary" ones by setting a threshold [26]–[28]. The threshold is determined from the observed p-value distribution [29], and is adaptive to the test data. More specifically, the FDR threshold is obtained as follows.

First, normalize the residual correlation c_s to have unit ℓ_2-norm and rearrange it of the magnitudes

$$|\mathrm{c}_s|_{(1)} \geq |\mathrm{c}_s|_{(2)} \geq \ldots \geq |\mathrm{c}_s|_{(k)} \geq \ldots \geq |\mathrm{c}_s|_{(K)}. \tag{11}$$

Then, compute its corresponding ordered p-values $P_{(1)} \leq P_{(2)} \leq \cdots \leq P_{(k)} \leq \cdots \leq P_{(K)}$. After that, let k_s be the largest index k for which $P_{(k)} \leq (k/K)q$, where q denotes a fixed ratio. Finally, set $T_s = |\mathrm{c}_s|_{(k_s)}$ as the FDR threshold.

Fig. 2 gives a simple example showing how the FDR control works in the RROMP algorithm. We build a dictionary $\mathbf{D}$ of size 128×512 whose entries are drawn randomly from the normal distribution $\mathcal{N}(0, 1)$ and columns are normalized. We generate a random representation α with $g = 16$ non-zeros, having amplitudes uniformly distributed on $\mathcal{N}(0, 1)$. The clean signal is obtained by $\mathrm{X} = \mathbf{D}\alpha$, and the noisy signal Y is obtained by adding white Gaussian noise with entries drawn from $\mathcal{N}(0, 1)$. Fig. 2(a) depicts the output of residual correlations. In Fig. 2(b), 32 elements are randomly selected from residual correlations. As can be seen, there are a few "significant" elements coupled with some relatively inferior ones. Employing the FDR control $(q = 0.005)$ to set a threshold, the significant estimates can be extracted from all the candidates, as shown in Fig. 2(c).

Through the above FDR control, more competitive sparse representations $\{\alpha^i\}$ can be ultimately acquired with the RROMP algorithm. Since these competitive representations lead to very small projection error $\|\mathrm{Y} - \mathrm{Y}_{\mathbf{U}}\|_2^2$ and thus the corresponding weights in (3) are close to 1, all the resulting representations can be merged by a simple average

$$\alpha^{AVE} = \frac{1}{I}\sum_{i=1}^{I} \alpha^i, \qquad i = 1, \ldots, I \tag{12}$$

and its estimate $\{\mathbf{D}\alpha^{AVE}\}$ approaches the MMSE estimator.

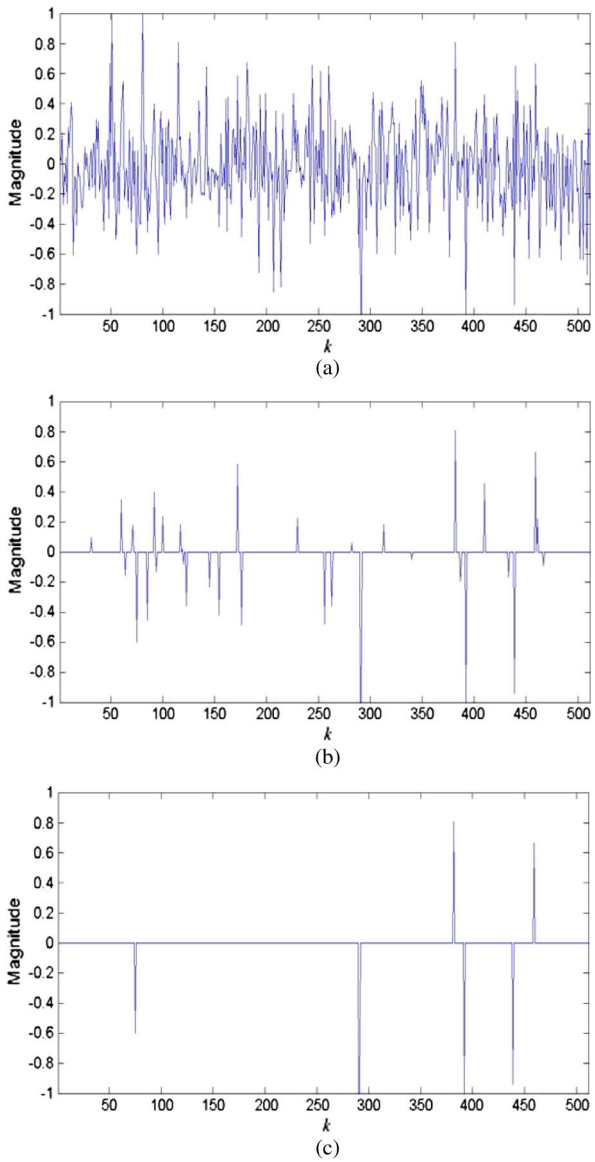


Fig. 2. FDR control in the RROMP algorithm. (a) Residual correlations c_s. (b) Randomly selected elements. (c) Refined elements using the FDR control.

III. Experimental Results

In this section, the proposed RROMP is first compared with the OMP [18] and RandOMP [24] on synthetically generated signals with white Gaussian noise. Then, the proposed method is tested on images corrupted by white Gaussian noise, and compared with other three well-known denoising methods [3], [4], [8]. Finally, the RROMP is extended to denoise images contaminated with speckle noise and compared with two recent speckle denoising approaches [9], [10].

A. Denoising Synthetic Signals With White Gaussian Noise

By problem suite (g, N, K), we mean a collection of sparse solution problems defined by two ingredients: a) a dictionary $\mathbf{D}$ of size N by K and b) a random representation α with sparsity level of g. By standard problem suite, the dictionary $\mathbf{D}$ is generated by normalizing a matrix with i.i.d. uniform random entries and the random representation α is created by sampling the g nonzeros from $\mathcal{N}(0, \sigma_{\mathrm{X}})$, $\sigma_{\mathrm{X}} = 1$. In these simulations, the performance of the RROMP is compared with the RandOMP and OMP on three standard problem suites: (25, 200, 500), (40, 500, 1000), (100, 1000, 2000).

The clean signal is generated by $\mathrm{X} = \mathbf{D}\alpha$, and its noisy version Y is obtained by adding white Gaussian noise with varying signal-to-noise ratio (SNR). The denoising effect is quantified by the relative mean square error (RMSE), which is a ratio between the mean square error obtained and the input noise level. Instead of placing a restriction on the iteration number, we choose the error goal $N(C\sigma)^2$ [30] as the stopping condition, and the parameter C for the RROMP, RandOMP and OMP is set to 1. For the RandOMP and RROMP, the number of generated representations I in (12) is chosen to 5, which has shown to be appropriate in our experiments on both synthetic signals and natural images. The number of randomly selected elements m in the step 3 of the RROMP is chosen to 10. The ratio q depends on the intensity values of the test signals and is set to 0.005 here. In fact, if the q increases, the RROMP will be accelerated due to more atoms selected in each iteration while its denoising performance may be slightly changed. However, we should note that when the intensity is fixed to a certain range, slight changes of the q will have little influence over the behavior of the RROMP.

The denoising results averaged on 1000 generated signals are shown in Fig. 3. As expected, we see in all these graphs that the RMSE gain of the RROMP over the RandOMP and OMP is consistent, though typically small in the problem suite (25, 200, 500). We also see that the denoising performances of both the RROMP and RandOMP are better than that of the OMP in general, which demonstrates the prominent effect of the MMSE estimator. However, another intriguing observation is that the behavior of the RandOMP is very close to that of the OMP for the SNR ≥ 15 in the problem suites: (40, 500, 1000), (100, 1000, 2000), whereas the advantage of the RROMP over the OMP and RandOMP is still obvious in these settings. It is worthwhile to note that since the MMSE estimator is very hard to compute in these large-scale signals, we do not add the MMSE estimator in these comparisons.

Fig. 4 shows how many iterations are needed for the above three algorithms to produce one representation of a test signal, which is randomly selected from the problem suite (40, 500, 1000). It can be seen that the RROMP indeed has much less iteration numbers than the OMP and RandOMP (about 1/4 of the number of iterations). In Table I, we also compare the computation time of the three algorithms for the above denoising simulations. The simulations are done in the environment of an AMD Athlon CPU 2.81 GHz with a 2.00 GB RAM PC, operating under Matlab 7.10.0. As we can see, the speed advantage of the RROMP over the RandOMP is obvious in all the problem suites. Especially in the problem suite (100, 1000, 2000), the RROMP runs approximately four times faster than the RandOMP, and even is close to the OMP. Therefore, in these high-dimensional signals, even though each iteration in the RROMP is more complex than that of the RandOMP due to the added FDR control, the computation complexity of the FDR control is relatively small compared with the complexity of

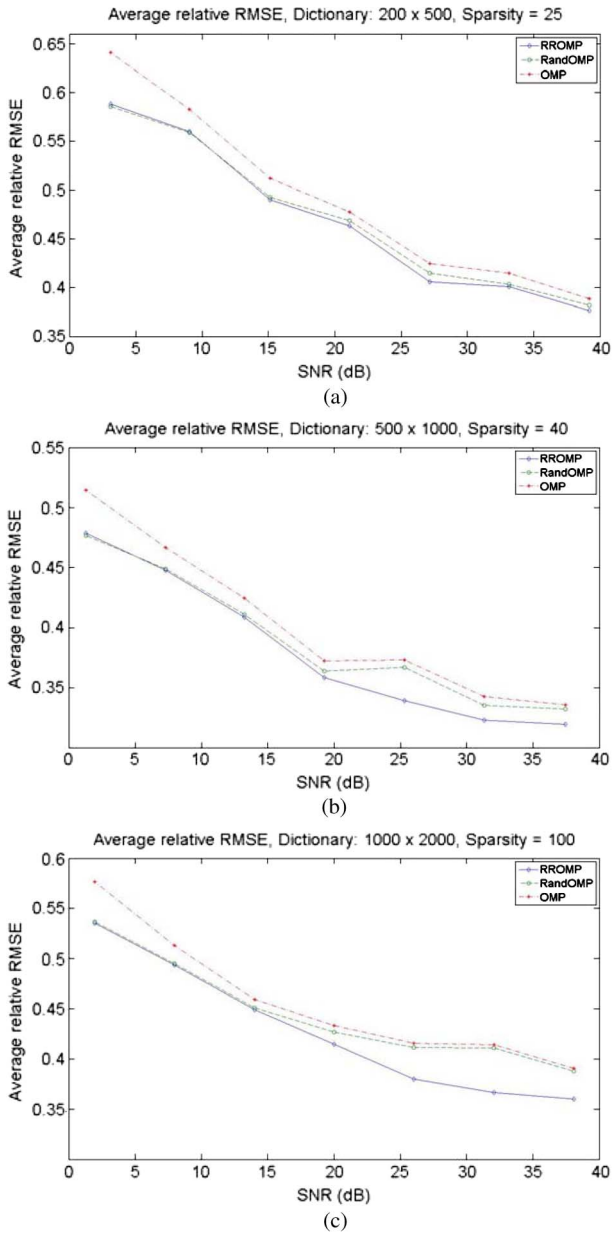


Fig. 3. Comparison of the denoising results for the OMP [18], RandOMP [24] and RROMP on different problem suites (g, N, K): (a) (25, 200, 500); (b) (40, 500, 1000); (c) (100, 1000, 2000).

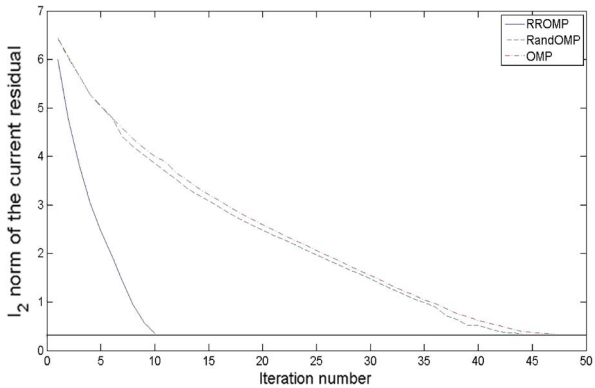


Fig. 4. Iteration numbers for the OMP [18], RandOMP [24] and RROMP to produce a representation of a test signal.

TABLE I
COMPARISON OF EXECUTION TIMES (IN SECONDS) FOR RUNNING 1000 GENERATED SIGNALS. THE EXECUTION TIME IN THIS TABLE IS AN AVERAGE OF ALL THE NOISE LEVELS

Problem suite (g,n,m)	OMP [18]	RandOMP [24]	RROMP
(25, 200, 400)	13.28	61.05	49.39
(40, 500, 1000)	73.75	395.13	141.21
(100, 1000, 2000)	693.67	3000.37	776.18

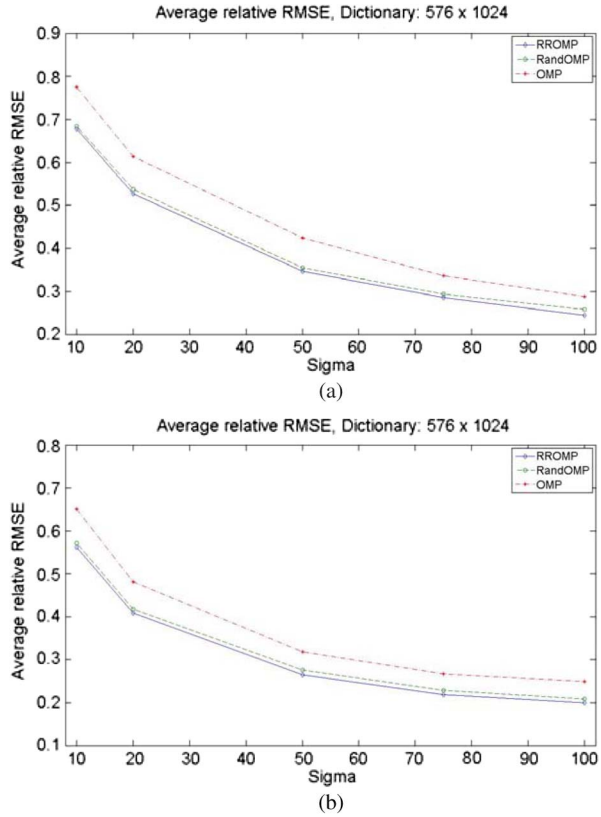


Fig. 5. Comparison of denoising results on the image-patches drawn from two images: (a) Barbara; (b) Lena.

the residual correlation correlation $(\mathbf{D}^{\mathrm{T}} \cdot \mathrm{r})$ and the orthogonal projection, which occur each iteration.

B. Denoising Images With White Gaussian Noise

In this subsection, our denoising experiments are divided into three parts. In the first part, we perform experiments on image patches drawn from some natural images, whose intensity values are fitted to the range [0, 255]. The test data-set is constructed by randomly selecting 1000 patches (of size 24×24) from images corrupted by white Gaussian noise, where the noise level σ is in the range between 10 and 100. Since the discrete cosine transform (DCT) dictionary[2] is simple and can serve natural image content well, it is applied in our experiments. The number of randomly selected elements m in the

[2]This dictionary is obtained by assigning $d[i, j] = \cos((i-1)(j-1)\pi/m)$ for $1 \leq i \leq n$ and $1 \leq j \leq m$removing the mean from all the atoms apart from the first, and normalizing each atom to unit ℓ^2-norm.

TABLE II
COMPARISON OF DENOISING RESULTS (IN PSNR) FROM THE FoE [3], SURE-LET [4], PROBSHRINK [8], OMP [18], RandOMP [24] AND RROMP. THE RESULTS REPORTED IN THE TABLE ARE AVERAGED ON FIVE IMAGES AND THE BEST RESULTS ARE LABELED IN BOLD

σ/Method	FoE [3]	SURE-LET [4]	ProbShrink [8]	OMP [18]	RandOMP [24]	RROMP
10	34.07	33.31	33.96	34.40	34.45	**34.51**
20	30.30	29.64	30.58	30.71	30.81	**31.01**
25	29.08	28.49	29.47	29.58	29.63	**29.94**
30	28.28	27.60	28.43	28.60	28.68	**28.98**
40	26.89	26.24	26.62	27.10	27.15	**27.49**
50	24.89	25.26	26.09	25.90	25.91	**26.33**
75	22.56	23.60	24.03	23.80	23.79	**24.10**
100	20.56	22.73	**23.08**	22.65	22.47	22.75

Fig. 6. Visual comparison of the reconstructed image ("Barbara") by the FoE [3], SURE-LET [4], ProbShrink [8], OMP [18], RandOMP [24] and RROMP. (a) Original image. (b) Noisy image (PSNR $= 20.19$ dB, $\sigma = 25$). (c) Denoising result using the FoE (PSNR $= 27.11$). (d) Denoising result using the SURE-LET (PSNR $= 26.74$). (e) Denoising result using the ProbShrink (PSNR $= 28.33$ dB). (f) Denoising result using the OMP (PSNR $= 28.63$ dB). (g) Denoising result using the RandOMP (PSNR $= 28.71$ dB). (h) Denoising result using the RROMP (PSNR $= 29.07$ dB).

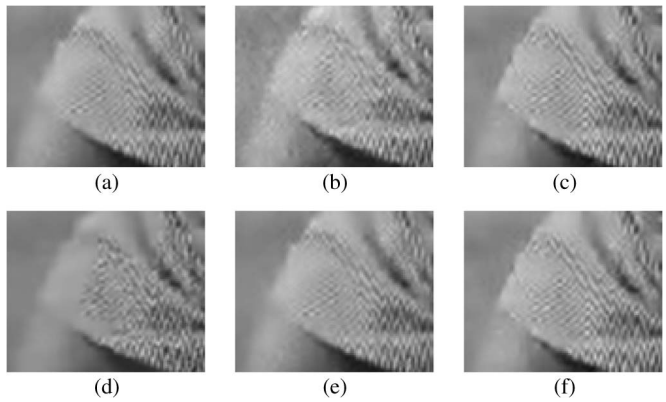


Fig. 7. Denoising result of the zooming area of the reconstructed image ("Barbara"): (a) FoE [3] (PSNR $= 27.11$). (b) SURE-LET [4] (PSNR $= 26.74$). (c) ProbShrink [8] (PSNR $= 28.33$ dB). (d) OMP [18] (PSNR $= 28.63$ dB). (e) RandOMP [24] (PSNR $= 28.71$ dB). (f) RROMP (PSNR $= 29.07$ dB).

RROMP is set the same as in the synthetic experiments. Considering that the intensity of the images is different with that of the synthetic signals, the ratio q in the FDR control is now modified to 0.2. The denoising results for the image-patches from image "Barbara" and "Lena" are shown in Fig. 5. As in the synthetic experiments, the performance of the RROMP still consistently outperforms that of the RandOMP and OMP in all the noise levels. In addition, we also observe that the RROMP saves about 10%–20% running time compared with the RandOMP on average.

The second part tests our method on grayscale images corrupted by white Gaussian noise. The experiments are very similar to those in [31], which first draw the image patches (of size 8×8) from the noisy image (the sliding distance between the patches is 1) and then return the processed patches to their original location to reconstruct the original image. In these experiments, the peak-signal-to-noise ratio (PNSR) is used as objective denoising measure. The parameter C in the stopping condition is chosen to 1.15, which is the same as in [31]. Five different grayscale images ("Barbara," "House," "Lena," "Peppers," and "Cameraman") are used and the added noise level σ is also in the range between 10 and 100. In Table II, the average (over all images) PSNRs of the RROMP are compared with the results from the RandOMP [24], OMP [18], and other three well-known denoising methods: FoE [3], SURE-LET [4], and ProbShrink [8]. It is worthy to note that the parameters of the FoE, SURE-LET and ProbShrink are set the same as in [3], [4], and [8] in our tests. We can see that our average results are obviously better than the other five methods when the noise level σ is between 10 and 75, whereas ProbShrink achieves better performances for the high noise level $\sigma = 100$. In Table II, it also should be noticed that the RandOMP does not show obvious advantage over the OMP, and even performs worse when the noise level σ is larger than 75. This is because there exists overlapping between the image-patches, and this overlapping reduces the effect of the MMSE estimator [25]. However, for the more competitive representations, the RROMP still outperforms the OMP by about 0.3 dB for the noise level σ between 20 and 75.

Fig. 6 provides a visual comparison of denoising results of image "barbara" using the FoE, SURE-LET, ProbShrink, OMP,

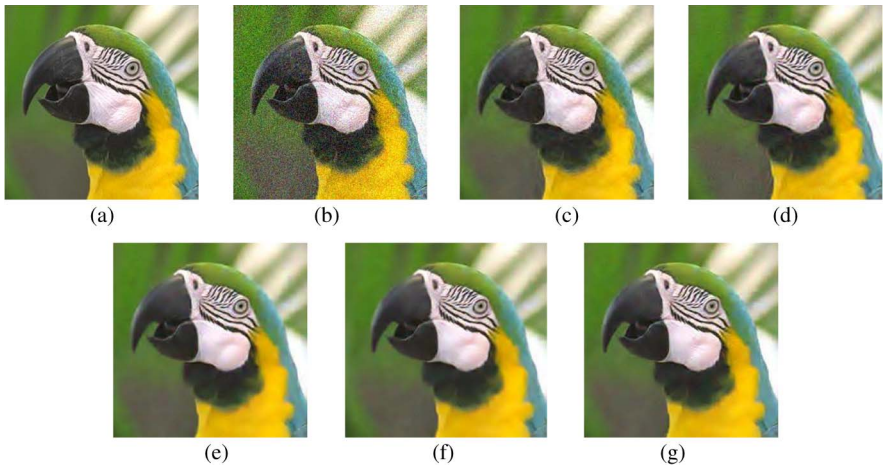


Fig. 8. Zooming area of the reconstructed color image ("Bird"). (a) Original image. (b) Noisy image (PSNR $= 20.17$ dB, $\sigma = 25$). (c) Denoising result using the SURE-LET [4] (PSNR $= 32.83$). (d) Denoising result using the ProbShrink [8] (PSNR $= 32.22$ dB). (e) Denoising result using the OMP [18] (PSNR $= 32.53$ dB). (f) Denoising result using the RandOMP [24] (PSNR $= 32.58$ dB). (g) Denoising result using the RROMP (PSNR $= 32.79$ dB).

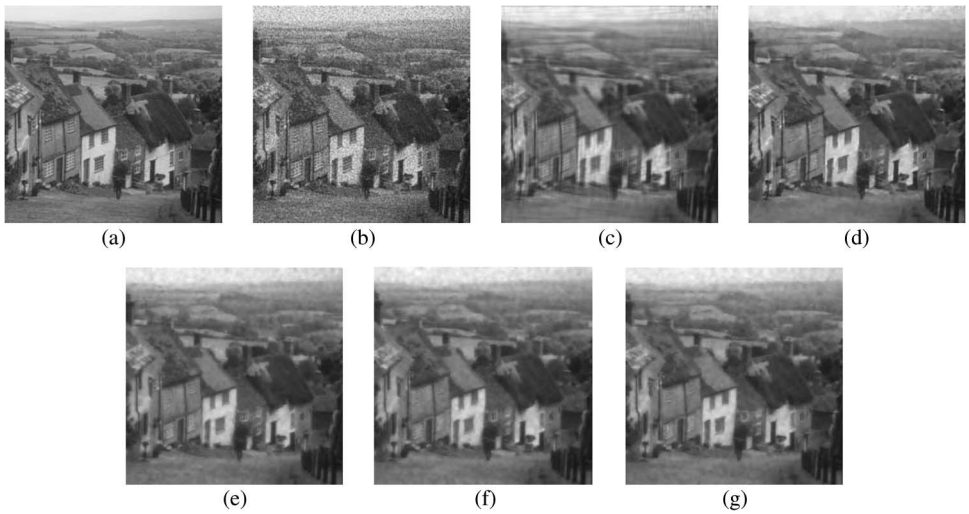


Fig. 9. Reconstructed image ("Hill"). (a) Original image. (b) Noisy image (PSNR $= 15.64$ dB). (c) Denoising result using the SteinBlock [9] (PSNR $= 26.02$). (d) Denoising result using the TVL1F [10] (PSNR $= 26.51$). (e) Denoising result using the OMP [18] (PSNR $= 26.15$ dB). (f) Denoising result using the RandOMP [24] (PSNR $= 26.13$ dB). (g) Denoising result using the RROMP (PSNR $= 26.38$ dB).

RandOMP and our method. It can be seen from these figures that the visual quality of our method is generally better than that of the other three methods. Furthermore, more detailed information (e.g., high-frequency texture information) can also be better preserved in our method, as shown in Fig. 7.

The third part conducts experiments on RGB color images corrupted by white Gaussian noise. For the OMP, RandOMP and RROMP, we just simply deal with each single RGB channel independently. Fig. 8 shows denoising results of several methods on the color image "Bird." As can be observed, our method and SURE-LET [4] obviously outperform other three approaches in term of PSNR. Moreover, we can see that our method outputs smoother surfaces in homogeneous regions and exhibits fewer artifacts, compared with SURE-LET and ProbShrink. We should note that both SURE-LET and ProbShrink adopt complex color denoising strategies, which integrate the dependences of each RGB channel to boost their performance. So, how to effectively exploit the correlations among each RGB channel in our method needs to be further studied.

C. Denoising Images With Speckle Noise

This subsection extends our method to denoise image contaminated with speckle noise. As indicated in [10], speckle noise is one kind of multiplicative noise and the noisy image $\mathrm{X}\eta$ is created by

$$\mathrm{X}\eta = \mathrm{X}\frac{1}{\mathrm{K}}\sum_{\mathrm{k}=1}^{\mathrm{K}}\eta_{\mathrm{k}} \tag{13}$$

where X is the clean image, η_{k} represents noise, which obeys one-side exponential distribution and K is set to 10. In this experiment, the noisy image is first processed by a logarithmic transform, which converts speckle noise into additive noise [32]. Then, the RROMP can be used to denoise the

log-transformed image with the same parameters in the above subsection and the denoised result in log-domain can be eventually transformed to the reconstructed image using an exponential function. Through the same operation, the OMP and RandOMP can also be used to denoise speckle noise. In Fig. 9, we compare our method with the OMP, RandOMP, and two recent speckle denoising approaches, SteinBlock [9] and TVL1F [10]. As can be seen from the denoised results, our method is very competitive with the TVL1F and performs better than the OMP, RandOMP, and SteinBlock in terms of both PSNR and visual quality. Likewise, we should note that our method can also be applied to Poisson denoising by a variance stability transform (VST) [33], [34] which can convert noise with Poisson distribution into noise with nearly Gaussian distribution. Actually, our preliminary tests in Poisson denoising have already provided promising results, but we do not add these results in this paper to save space. Another interesting question is how to extend our method to real noisy image denoising. Since the noise level of real noisy image is unknown, our future work will incorporate some noise automatic estimation models [35] into our denoising framework.

IV. Conclusion

In this paper, we present a new sparse coding algorithm called random refined orthogonal matching pursuit (RROMP) which considers an improved sampling strategy. The overall process requires less iterations and thus becomes faster, due to the multiple-selection strategy which allows for multiple atoms to be obtained each iteration. Besides, the generated representations are very competitive by applying the FDR control, thus resulting in better denoising performance. However, when the dimension of the signals is very low, the RROMP algorithm may turn to be less efficient, owing to the added complexity of the FDR control which operates each iteration. Thus, part of our ongoing work is to adopt some efficient techniques (e.g., using a look-up-table to compute the ordered p values instead of an exact computation) to reduce the complexity in the FDR control, which enable the RROMP to be more applicable for small signals. Furthermore, in the future, we can apply the RROMP algorithm to other applications, such as 3-D image denoising and compressive sensing reconstruction [36].

Acknowledgment

We thank Dr. R. Rubinstein and Prof. M. Elad (The Technion-Israel Institute of Technology) for their insightful suggestions and constructive comments on this work. We would also like to thank the excellent reviewers for their detailed reviews and valuable recommendations that improve the paper greatly.

References

[1] F. Russo, "Technique for image denoising based on adaptive piecewise linear filters and automatic parameter tuning," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 4, pp. 1362–1367, Aug. 2006.

[2] F. Russo, "An image-enhancement system based on noise estimation," *IEEE Trans. Instrum. Mens.*, vol. 56, no. 4, pp. 1435–1442, Aug. 2007.

[3] S. Roth and M. J. Black, "Fields of expert: A framework for learning image priors," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, San Diego, CA, 2005, vol. 2, pp. 860–867.

[4] F. Luisier and T. Blu, "SURE-LET multichannel image denoising: Interscale orthonormal wavelet thresholding," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 482–492, Apr. 2008.

[5] S. Neville and N. Dimopoulos, "Wavelet denoising of coarsely quantized signals," *IEEE Trans. Instrum. Meas.*, vol. 55, no. 3, pp. 892–901, Jun. 2006.

[6] I. Firoiu, C. Nafornita, and J.-M. Boucher, "Image denoising using a new implementation of the hyperanalytic wavelet transform," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 8, pp. 2410–2416, Aug. 2009.

[7] A. Mencattini, M. Salmeri, R. Lojacono, M. Frigerio, and F. Caselli, "Mammographic images enhancement and denoising for breast cancer detection using dyadic wavelet processing," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 7, pp. 1422–1430, Jul. 2008.

[8] A. Pižuriza and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single- and multiband image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 654–665, Mar. 2006.

[9] C. Chesneau, J. Fadili, and J.-L. Starck, "Stein block thresholding for image denoising," *Appl. Comput. Harmon. Anal.*, vol. 28, no. 1, pp. 67–88, Jan. 2010.

[10] S. Durand, J. Fadili, and M. Nikolova, "Multiplicative noise removal using L1 fidelity on frame coefficients," *J. Math. Imag. Vis.*, vol. 36, no. 3, pp. 201–226, Mar. 2010.

[11] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modelling of signals and images," *SIAM Rev.*, vol. 51, no. 1, pp. 34–81, Feb. 2009.

[12] S. Mallat, *A Wavelet Tour of Signal Processing*, 3rd ed. New York: Academic, 2009.

[13] B. Yang and S. T. Li, "Multifocus image fusion and restoration with sparse representation," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 4, pp. 884–892, Apr. 2010.

[14] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constr. Approx.*, vol. 13, no. 1, pp. 57–98, Mar. 1997.

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

[16] B. Efron, T. Hastie, I. Johnston, and R. Tibshirani, "Least angle regression," *Ann. Stat.*, vol. 32, no. 2, pp. 407–499, Apr. 2004.

[17] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[18] 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.*, Pacific Grove, CA, Nov. 1993, pp. 40–44.

[19] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, *Sparse Solution of Underdetermined Linear Equations by Stagewise Orthogonal Matching Pursuit*, 2006. [Online]. Available: http://www-stat.stanford.edu/~donoho/Reports/2006/StOMP-20060403.pdf

[20] D. Needell and J. A. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Appl. Comput. Harmon. Anal.*, vol. 26, no. 3, pp. 301–321, Apr. 2008.

[21] B. Wohlberg, "Noise sensitivity of sparse signal representations: Reconstruction error bounds for the inverse problem," *IEEE Trans. Signal Process.*, vol. 51, no. 12, pp. 3053–3060, Dec. 2003.

[22] D. L. Donoho, M. Elad, and V. Temlyyakov, "Stable recovery of sparse overcomplete representations in the presence of noise," *IEEE Trans. Inf. Theory*, vol. 52, no. 1, pp. 6–18, Jan. 2006.

[23] J. A. Tropp, "Just relax: Convex programming methods for subset selection and sparse approximation," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 1030–1051, Mar. 2006.

[24] M. Elad and I. Yavneh, "A plurality of sparse representations is better than the sparsest one alone," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4701–4714, Oct. 2009.

[25] M. Protter, I. Yavneh, and M. Elad, "Closed-Form MMSE estimation for signal denoising under sparse representation modeling over a unitary dictionary," *IEEE Trans. Signal Process.*, vol. 58, no. 7, pp. 3471–3484, Jul. 2010.

[26] F. Abramovich, Y. Benjamini, D. L. Donoho, and I. M. Johnstone, "Adapting to unknown sparsity by controlling the false discovery rate," *Ann. Stat.*, vol. 34, no. 2, pp. 584–653, Apr. 2006.

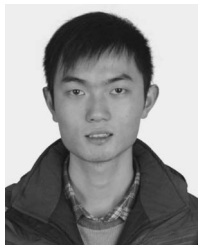
[27] Y. Benjamini and Y. Hochberg, "Controlling the false discovery rate: A practical and powerful approach to multiple testing," *J. Roy. Stat. Soc. Ser. B*, vol. 57, no. 1, pp. 289–300, Feb. 1995.

[28] D. L. Donoho and J. Jin, "Asymptotic minimaxity of false discovery rate thresholding for exponential data," *Ann. Stat.*, vol. 34, no. 6, pp. 2980–3018, Dec. 2006.
[29] T. T. Theodore, *Introduction to Engineering Statistics and Six Sigma: Statistical Quality Control and Design of Experiments and Systems*. London, U.K.: Springer-Verlag, 2006.
[30] 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.
[31] 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.
[32] H. Xie, L. Pierce, and F. Ulaby, "Statistical properties of logarithmically transformed speckle," *IEEE Trans. Geosci. Remote Sens.*, vol. 40, no. 3, pp. 721–727, Mar. 2002.
[33] F. J. Anscombe, "The transformation of poisson, binomial and negative-binomial data," *Biometrika*, vol. 35, no. 3/4, pp. 246–254, Dec. 1948.
[34] M. Freeman and J. Tukey, "Transformations related to the angular and the square root," *Ann. Math. Stat.*, vol. 21, no. 4, pp. 607–611, Dec. 1950.
[35] C. Liu, R. Szeliski, S. B. Kand, C. L. Zitnick, and W. T. Freeman, "Automatic estimation and removal of noise from a single image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 299–314, Feb. 2008.
[36] J. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Trans. Image Process.*, vol. 18, no. 7, pp. 1395–1408, Jul. 2009.

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.

He joined the College of Electrical and Information Engineering, Hunan University, in 2001. From May 2001 to October 2001, he was a Research Associate in the Department of Computer Science, Hong Kong University of Science and Technology, Kowloon, Hong Kong. From November 2002 to November 2003, he was a Postdoctoral Fellow at the Royal Holloway College, University of London, Egham, U.K., working with Prof. John-Shawe-Taylor. During April 2005 to June 2005, he has visited the Department of Computer Science, Hong Kong University of Science and Technology, as a visiting professor. Now, he is a full professor with the College of Electrical and Information Engineering, Hunan University. He has authored or coauthored more than 120 refereed papers. His professional interests are 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. He served as a member of IEEE's Neural Networks Technical Committee from 2007 to 2008.



Leyuan Fang (S'10) received the B.S. degree in electrical engineering from Hunan University of Science and Technology, Xiangtan, China, in 2008. He joined the College of Electrical and Information Engineering, Hunan University, Changsha, China, in 2008, for the Ph.D. degree program.

His technical interests include signal processing and sparse representation.