Image Denoising via Nonlocally Sparse Coding and Tensor Decomposition

Wenrui Hu
Institute of Automation
Chinese Academy of Sciences
Beijing, 100190, China
wenrui.hu@ia.ac.cn

Yuan Xie
Institute of Automation
Chinese Academy of Sciences
Beijing, 100190, China
yuan.xie@ia.ac.cn

Wensheng Zhang
Institute of Automation
Chinese Academy of Sciences
Beijing, 100190, China
wszhang_casia@hotmail.com

Limin Zhu
Institute of Automation
Chinese Academy of Sciences
Beijing, 100190, China
raymanzhu@139.com

Yanyun Qu
School of Electronics and
Computer Science
Xiamen University
yyq@xmu.edu.cn

Yuanhua Tan
Karamay Hongyou Software
CO. LTD
Xinjiang, 834000, China
hyzhp@petrochina.com

ABSTRACT

The nonlocally sparse coding and collaborative filtering techniques have been proved very effective in image denoising, which yielded state-of-the-art performance at this time. In this paper, the two approaches are adaptively embedded into a Bayesian framework to perform denoising based on split Bregman iteration. In the proposed framework, a noise-free structure part of the latent image and a refined observation with less noise than the original observation are mixed as constraints to finely remove noise iteration by iteration. To reconstruct the structure part, we utilize the sparse coding method based on the proposed nonlocally orthogonal matching pursuit algorithm (NLOMP), which can improve the robustness and accuracy of sparse coding in present of noise. To get the refined observation, the collaborative filtering method are used based on Tucker tensor decomposition, which can takes full advantage of the multilinear data analysis. Experiments illustrate that the proposed denoising algorithm achieves highly competitive performance to the leading algorithms such as BM3D and NCSR.

Categories and Subject Descriptors

I.5.4 [**Pattern Recognition**]: Applications—*Computer vision, signal processing*

General Terms

Algorithms

Keywords

collaborative filtering, sparse coding, tensor decomposition, Bregman iteration

1. INTRODUCTION

Cleaning of noise from images is a classical and long studied problem in image processing. For an observed image $y \in \Re^{\sqrt{N}\times\sqrt{N}}$, the problem of image denoising can be generally formulated by

$$y = x + n \tag{1}$$

where $x \in \Re^{\sqrt{N}\times\sqrt{N}}$ is the latent image and $n \in \Re^{\sqrt{N}\times\sqrt{N}}$ is the additive Gaussian white noise. There are several image denoising techniques that have been developed in the past few decades, such as partial differential equations [1], spatially varying convolution [2], kernel regression [3], nonlocal techniques [4], transform-based techniques [5], and techniques based on sparse coding [6].

Among the above techniques, nonlocal techniques [4] which assume that there exists repeating structures in a given image have received increasingly more attention in recent years. One of reasons for this population is the nonlocal assumption greatly extends the ability of other methods. For example, nonlocal ROF model was developed in [7] to extend the traditional ROF model based on PEDs [1]. By combining the nonlocal and transform-domain approaches, Dabove et al. proposed a collaborative filtering algorithm named BM3D to perform denosing. The BM3D algorithm is well known due to its outstanding performance and can be considered to be the state of the art at this time. More recently, in [9], Rajwade et al. used the same framework to perform densoing but replaced the fixed transform bases (whether Haar/DCT/Biorthogonal wavelet) in BM3D by the spatially adaptive bases constructed by Tucker tensor decomposition. Inspired by the nonlocal and sparse coding approaches, Mairal et al. proposed a denoising algorithm based on the nonlocal sparse model (NLSM). More recently, in [11], Dong et al. introduced the concept of sparse coding noise and proposed a nonlocally centralized sparse representation (NCSR) to suppress the sparse coding noise for image denoisng. Both NLSM and NCSR have yielded state-of-the-art performance on par with the BM3D algorithm.

In this paper, we unify the nonlocal, transform-domain and sparse coding approaches to perform denoising based on split Bregman iteration [12]. In the proposed denoisng framework, we first reconstruct the noise-free structure part

x_S of the latent image by the nonlocally sparse coding method. Due to the potential instability of sparse decompositions in present of noise, we propose a nonlocal orthogonal matching pursuit algorithm (NLOMP) to reduce the coding error. Next, the collaborative filtering based on Tucker tensor decomposition is used to get a refined observation x_T with less noise than the original observation. Finally, the two parts x_S and x_T are mixed as constraints to finely remove noise iteration by iteration

This paper is organized as follows. Section 2 describes the proposed denoising algorithm in detail. Experimental results are given in Section 3 to verify our algorithm, and show the performance as compared with other algorithms. Finally, we conclude the algorithm in Section 4.

2. METHODS

2.1 Nonlocal Orthogonal Matching Pursuit

The denoising technique based on sparse coding is to denoised image by solving the following minimization [6]

$$\min_{x,\alpha,D} \frac{1}{2}||y-x||_2^2 + \sum_i \mu_i ||\alpha_i||_0 + \lambda \sum_i ||D\alpha_i - R_i x||_2^2 \quad (2)$$

where D is the overcomplete dictionary learned from the noisy image, μ_i and λ are regularization parameters, and R_i is an $n \times N$ matrix that extracts the i-th patch from the image. α_i is the sparse coefficients corresponding to the i-th patch. As the ℓ_0 problem is complicated in general, approximation methods are often employed. One such approximation technique is the orthogonal matching pursuit (OMP)[13], which is a greedy algorithm and can guarantee near-optimal results in some cases.

However, it has been shown that sparse coding with an overcomplete dictionary is unstable [14], which can result in noticeable reconstruction artifacts in the denoised image. Fortunately, multiple observations of a sparse signal can improve the ability to identify the underlying sparse representation [15]. To reduce the coding error in present of noise, we proposed a nonlocal orthogonal matching pursuit (NLOMP) algorithm based on the fact that natural images often contain repetitive structures, i.e., the rich amount of nonlocal redundancies [4].

Given the dictionary D, the objective function of our nonlocal sparse model is

$$\min_{\alpha_i} \ ||\alpha_i||_0, \quad s.t. \quad \sum_{q\in\Omega_i} ||D\alpha_i - y_q||_2^2 w_q \le \varepsilon \quad (3)$$

where Ω_i denotes a set of patches similar as the given patch y_i (including y_i), ε is a small constant controlling the approximation error, and w_q is the weight. We set the weights to be inversely proportional to the photometric distance between patches y_i and y_q

$$w_q = \frac{1}{W} exp(-||y_i - y_q||_2^2/h) \quad (4)$$

where h is the smooth parameter and W is the normalization constant. Then the algorithm of NLOMP is summarized in Alg. 1.

To verify the robustness and accuracy of NLOMP for sparse coding, we compare the proposed NLOMP algorithm with the traditional OMP algorithm. Fig. 1 (a) is the example image *House* with the noise level $\sigma = 20$, and Fig. 1 (b) shows the learned dictionary using the K-SVD algorithm [16]. Fig. 2 illustrates the coding results for different patches in Fig. 1 (a). The patch size is 8×8. The red stems correspond the sparse coding coefficients for the latent patches by OMP, and we take them as the latent sparse coefficients. The blue stems correspond the coefficients for the noisy patches by OMP while the green stems correspond the coefficients by our NLOMP. Obviously, the coding results of our NLOMP are more consistent with the latent coefficients than the ones of OMP.

Algorithm 1: Nonlocal Orthogonal Matching Pursuit

Input: The dictionary D, the similar signals $\{y_q\}$, the weights $\{w_q\}$, and the threshold ε.
Output: The sparse coefficient α.

Initialization: Initialize $k = 0$, and set
  The initial solution $\alpha^0 = 0$.
  The initial residuals $\{r_q^0 = y_q - D\alpha^0 = y_q\}$.
  The initial support $S^0 = Sup\{\alpha^0\} = \emptyset$.
while $\sum_{q\in\Omega} ||r_q^k||_2^2 w_q > \varepsilon$ **do**
  k = k+1.
  Sweep:
    Compute the optimal z_j of $\epsilon(j)$ for every d_j using
    $z_j^* = d_j^T r_w^{k-1}/||d_j||_2^2$, where $r_w = \sum_{q\in\Omega} w_q r_q$
    and $\epsilon(j) = \min \sum_{q\in\Omega} ||d_j z_j - r_q^{k-1}||_2^2 w_q$.
  Update Support:
    Find $j_0 : \forall\, j \notin S^{k-1}, \epsilon(j_0) < \epsilon(j)$,
    and update $S^k = S^{k-1} \cup \{j_0\}$.
  Update Solution:
    Compute the optimal α^k by
    $\min_\alpha \ \sum_{q\in\Omega} ||D\alpha - y_q||_2^2 w_q, \quad s.t. \ \ Sup\{\alpha\} = S^k$.
  Update Residuals:
    Compute $\{r_q^k = y_q - D\alpha^k\}$.
end
Return α;

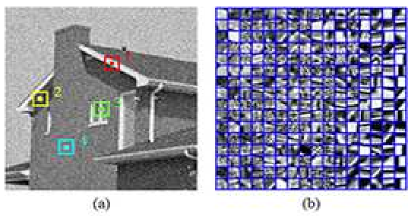

Figure 1: Example image and the dictionary. (a) the noisy image with variance $\sigma = 20$, (b) the corresponding adaptive dictionary.

It is worth noting that there exists a few small coefficients in the latent coding results which correspond to the degraded or lost texture of the image due to noise. In general, it is a big challenge to accurately restore these small coefficients in present of noise. Getting accurate restoration of big coefficients, however, is much easier when the structure part exists.

2.2 Denoising via Tensor Decomposition

Given a tensor $T \in \Re^{\sqrt{n}\times\sqrt{n}\times K}$, the Tucker tensor de-

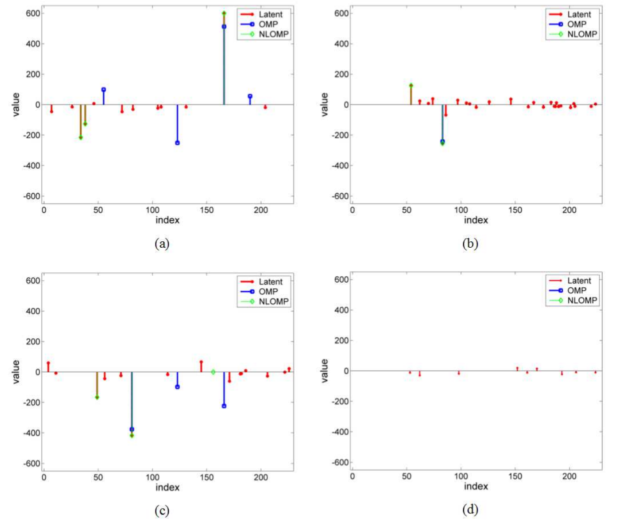

Figure 2: Sparse coding for different patches. (a) patch 1, (b) patch 2, (c) patch 3, (d) patch 4. The x axis denotes the index of coefficients and the y axis denotes the value of coefficients. (Color version shows clearly.)

composition of T is [18]

$$T = S \times_1 U^{(1)} \times_2 U^{(2)} \times_3 U^{(3)} \tag{5}$$

where $U^{(1)} \in \Re^{\sqrt{n}\times\sqrt{n}}$, $U^{(2)} \in \Re^{\sqrt{n}\times\sqrt{n}}$, and $U^{(3)} \in \Re^{\sqrt{n}\times\sqrt{n}}$ are ortho-normal matrices, which can be considered as 3D transform basis pairs, and S is a 3D coefficient array (core tensor) of size $\sqrt{n} \times \sqrt{n} \times K$. Here, the symbol $\times_n$ stands for the n-th mode tensor product.

In this work, followed [9], we use the tensor decomposition to finely remove noise while preserve texture at the same time. There are two advantages: firstly, due to the massive variety of the geometrical structure in the nature images, the data adaptive transform basis learned from the given noisy image [9, 17] is more suitable than the fixed basis in BM3D; secondly, tensor-based multilinear data analysis is capable of taking full advantage of the multilinear structures to provide better understanding and more precision[18].

Algorithm 2: Denoising via Tensor Decomposition

Input: The noisy image y. The noise variance σ.
Output: The denoised image x .

1 **Initialization:** $x = y$
2 **Patch clustering:** find the KNN for each exemplar patch and create tensor T_i for each cluster;
3 **Tensor decompostion:** $(S_i, U_i^{(1)}, U_i^{(2)}, U_i^{(3)},) = \mathrm{TD}(T_i)$;
4 **Thresholding:** $\tau = \eta\sigma\sqrt{2log(nK)}$, $\hat{S}_i = \mathrm{Thresh}(S_i, \tau)$;
5 **Tensor reconstruction:** $\hat{T}_i = \hat{S}_i \times_1 U_i^{(1)} \times_2 U_i^{(2)} \times_3 U_i^{(3)}$;
6 **Image update:** obtain the denoised image x by weighted averaging all denoised patches;
7 **Return** x.

Alg. 2 summarizes the image denoising algorithm using tensor decomposition. In the step of thresholding, τ is the threshold and η is a control parameter. The thresholding is a hard shrinkage operator as follow

$$\mathrm{Thresh}(z, \tau) = \begin{cases} z & |z| > \tau \\ 0 & |z| \le \tau \end{cases} \tag{6}$$

The choice of τ is of importance for the denoising algorithms which perform denoising by filtering the transform coefficients. If it is set too big, the texture will lost in the denoised image. If it is set too small, the noise will not be removed enough. In this work, we empirically set it less than 0.5 to remove a certain amount of the noise at each iteration.

2.3 Denoising Based on Iteration

In this section, we unify nonlocal orthogonal matching pursuit (NLOMP) and tensor decomposition (TD) to perform denoising based on split Bregman iteration.

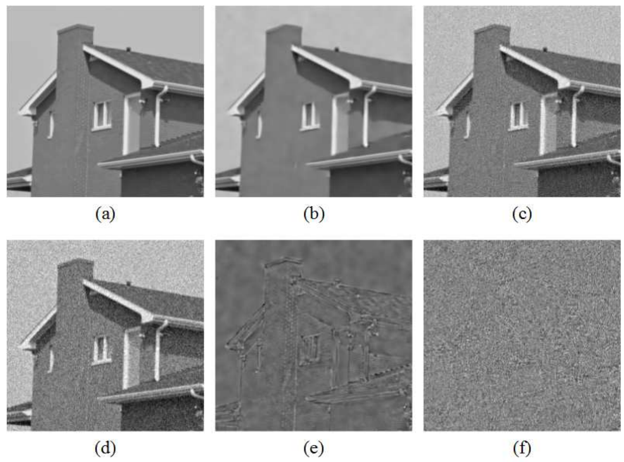

Figure 3: Image denoising using separately NLOMP and TD. (a) latent image x, (b) the denoised image x_S using NLOMP, (c)the denoised image x_T using TD, (d) the noisy image with $\sigma = 20$, (e) the difference r_1 between (a) and (b), (f) the difference r_2 between (a) and (c).

Fig. 3 shows the denoised results using separately NLOMP and TD. We can see that both noise and texture have been removed from the denoised image x_S using NLOMP (Fig. 3 (b)) and the difference r_1 (Fig. 3(e)) with the latent image well represents the texture. Fig. 3 (c) is the denoised result x_T by Alg. 2 using TD where we set the control parameter $\eta = 0.4$, and Fig. 3(d)) is the difference r_2 with the latent image. It is obvious that there remains a lot of noise in x_T. However, more textures are preserved at the same time.

The empirical distributions of the difference r_1 (black one) and r_2 (blue one) are plotted in Fig. 4. We can see that the distribution of r_1 can be well characterized by Laplacian distribution, while Gaussian distribution can well fit the distribution of r_2. This observation motivates us to model r_1 and r_2 with a Laplacian prior and a Gaussian prior respectively. Specifically, $P(x|x_S) \propto e^{-|x-x_S|}$ and $P(x|x_T) \propto e^{-||x-x_T||_2^2}$. Assumed x_S and x_D are independent, maximizing the posterior probability $P(x|x_T, x_S) \propto P(x|x_S)P(x|x_T)$ is equivalent to minimizing the following objective function

$$E(x) = \min_x \quad |x - x_S| + \frac{\lambda}{2}||x - x_T||_2^2 \tag{7}$$

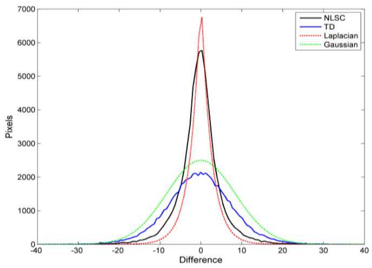

Figure 4: The empirical distributions of the difference. The x axis denotes the value of difference, and the y axis denotes the number of pixels. (Color version shows clearly.)

where λ is the regularization parameter. Eq. (7) is not patch-based but global denoising procedure mixing the constraints of x_S and x_T. We can investigate Eq. (7) in view of regularization other than Beyesian perspective, namely, x_S is the structure part and the ℓ_1-norm constraint forces the estimated image x enhanced just at textures, while x_T can be considered as a refined observation with less noise than the original noisy observation and the ℓ_2-norm constraint guarantees x consistent with the latent image.

We resort to the split Bregman method [12] to solve Eq. (7) duo to the rapid convergence in dealing with the ℓ_1-based optimization problem. The key idea of the split Bregman is that it "de-couple" the ℓ_1 and ℓ_2 portions of the energy in Eq. (7). By introducing a new variable, we can get the following problem equivalent to Eq. (7)

$$\min_{x,d} \quad |d| + \frac{\lambda}{2}||x - x_T||_2^2, \quad s.t. \;\; d = \Phi(x) = x - x_S \tag{8}$$

As shown in [12], the above iteration is equivalent to the simple version of the split Bregman iteration

$$(x^{k+1}, d^{k+1}) = \min_{x,d} \quad |d| + \frac{\lambda}{2}||x - x_T||_2^2 + \frac{\mu}{2}||d - \Phi(x) - b^k||_2^2 \tag{9}$$

$$b^{k+1} = b^k + \Phi(x^{k+1}) - d^{k+1} \tag{10}$$

where μ is the free parameter. We can perform the minimization problem Eq. (9) efficiently by iterative minimizing with respect to x and d separately. The two steps we must perform are

$$Step1: \; x^{k+1} = \min_{x} \quad \frac{\lambda}{2}||x - x_T||_2^2 + \frac{\mu}{2}||d^k - \Phi(x) - b^k||_2^2 \tag{11}$$

$$Step2: \; d^{k+1} = \min_{d} \quad |d| + \frac{\mu}{2}||d - \Phi(x^{k+1}) - b^k||_2^2 \tag{12}$$

To solve Step 1, note that because we have "de-couple" x from the ℓ_1 portion of the problem, the optimization problem that we must solve for x^{k+1} is now differentiable, and the analysis solution is

$$x^{k+1} = \frac{\lambda}{\lambda+\mu}x_T + \frac{\mu}{\lambda+\mu}x_S + \frac{\mu}{\lambda+\mu}(d^k - b^k) \tag{13}$$

In Step 2, there is no coupling between elements of d. The optimal value of d can be explicit computed by soft shrinkage operators

$$d^{k+1} = shrink(\Phi(x^{k+1}) + b^k, 1/\mu) \tag{14}$$

and the shrink operator is

$$shrink(z, \gamma) = \begin{cases} z - \gamma & z > \gamma \\ 0 & -\gamma \le z \le \gamma \\ z + \gamma & z < -\gamma \end{cases} \tag{15}$$

The final image denoising algorithm unifying NLOMP and TD based on split Bregman iteration is summarized in Alg. 3, where "Med" in re-estimating the noise variance stands for median operator. Note that we update the x_T with x^k and $\hat{\sigma}^k$ as inputs of Alg. 2 at each iteration.

Algorithm 3: Iterative Image Denoising

Input: The noisy image y. The noise variance σ.
Output: The denoised image x .

Initialization: $x = y, \; d = 0, b = 0$
Stage 1: Get x_S using NLOMP and x_T using TD
Stage 2: Denoising based on split Bregman iteration
while $||\hat{\sigma}^k - \hat{\sigma}^{k-1}||_2^2 > tol \; and \; k < Miter$ **do**
  Solve x^k using Eq. (13);
  Re-estimate noise variance:
  $\hat{\sigma}^k = \sqrt{\sigma^2 - \mathrm{Med}(||y - x^k||_2^2)}$;
  Update x_T with $\hat{x}^k$ and $\hat{\sigma}^k$ as inputs of Alg.2;
  Solve d^{k+1} using Eq. (14);
  Solve b^{k+1} using Eq. (10);
  k = k + 1;
end
Return x^k.

---

3. EXPERIMENTS

In this section we will illustrate the performance of the proposed approach. Several algorithms, such as BM3D [8], NCSR [11], and KSVD [6] were used for comparisons. Both peak signal to noise ratio (PSNR) and structural similarity (SSIM) indices are adopted to evaluate the objective quality of the denoised results.

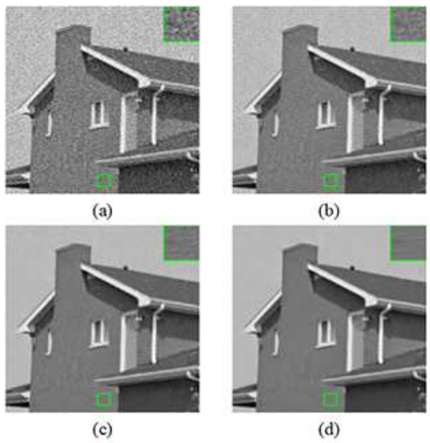

Figure 5: The evolution of the denoised result using the Alg. 3. (a) the noisy image with variance $\sigma = 20$ and PSNR 22.08 dB, (b) the denoised result after iterating once with PSNR 29.58 dB, (c) the denoised result after iterating twice with PSNR 34.47 dB, (d) the denoised result after iterating thrice with PSNR 34.90 dB.

The basic parameter setting of the proposed method is as follow: the patch size is 8×8; for Alg. 2, the number of similar patches is 30 in the step of patch clustering and the control parameter η is 0.4 in the step of thresholding; for Alg. 3, the regularization parameter λ of the objective function in Eq. (7) is 20σ , the auxiliary parameter μ for the Bregman iteration in Eq. (9) is 50, the stopping criterion "tol" is 0.1σ and the maximum number of iterations is 4. Note that the parameter settings mentioned above are fixed in all tests.

Fig. 5 illustrates the evolution of the denoised result using Alg. 3. The noise variance σ is 20. The image content inside the small green rectangle is zoomed in the upright corner. Compared with the denoised results in Fig. 3 (b) and Fig. 3 (c) which separately use NLOMP and TD, we can see that the noise is finely removed step by step and the degraded texture is preserved at the same time (see 5 (d)).

A set of 7 natural images commonly used in the literature of image denoising are used for comparison. The denoised results using different methods with difference noise variance are reported in Table 1. The highest PSNR and SSIM values are highlighted in each cell to facilitate the comparison.We can see that the proposed method achieves at least comparable denoising performance to the state-of-the-art algorithms BM3D and NCSR.

4. CONCLUSIONS

In this paper, we use the sparse coding based on the proposed NLOMP and the collaborative filtering based on tensor decomposition to perform denosing. The denoising framework based on split Bregman iteration has the Bayesian explanation, and at each iteration, the noise is finely removed from the current denoised image. Experimental results demonstrated that the proposed denoising algorithm can achieve the competitive performance to the state-of-the-art denoising algorithms.

5. ACKNOWLEDGMENTS

The work is supported by the National Natural Science Foundation of China (Nos. U1135005, 61305018, 61304224), the Key National Project of China (No. 0101050302).

6. REFERENCES

[1] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. Physica D, vol. 60, pp. 259 - 268, 1992.

[2] D. Tschumperle and R. Deriche. Vector-valued image regularization with PDEs: a common framework for different applications. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 27, pp. 506 - 517, 2005.

[3] H. Takeda, S. Farsiu, and P. Milanfar. Kernel regression for image processing and reconstruction. IEEE Transactions on Image Processing, vol. 16, pp. 349 - 366, 2007.

[4] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. Multiscale Model. Simul., vol. 4, pp. 490 - 530, 2005.

[5] L. Yaroslavsky, K. Egiazarian, and J. Astola. Transform domain image restoration methods: review, comparison and interpretation. Proc. SPIE Series, Nonlinear Processing and Pattern Analysis, pp. 1 - 15, 2001.

[6] M. Eladand and M. Aharon. Image denoising via sparse and redundant representations over learned dictionaries. IEEE Trans. on Image Processing, vol. 15, pp. 3736 - 3745, 2006.

[7] G. Gilboa and S. Osher. Nonlocal operators with applications to image processing. Multiscale Model and Simulation, vol. 7, pp. 1005-1028, 2008.

[8] K. Dabov, A. Foi, V. Katkovnik, and et al. Image denoising by sparse 3D transform-domain collaborative filtering. IEEE Trans. Image Processing, vol. 16, pp. 2080 - 2095, 2007.

[9] A. Rajwade, A. Rangarajan, and A. Banerjee. Image denoising using the higher order singular value decomposition. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 35, pp. 849 - 862, 2013.

[10] J. Mairal, F. Bach, J. Ponce, and et al. Nonlocal sparse models for image restoration. IEEE International Conference of Computer Vision, pp. 2272 - 2279, 2009.

[11] W. Dong, L. Zhang, G. Shi, and et al. Nonlocally centralized sparse representation for image restoration. IEEE Trans. on Image Processing, vol. 22, pp. 1620 - 1630, 2013.

[12] T. Goldstein and S. Osher. The split bregman method for L1 regularized problems. SIAM Journal on Imaging Sciences, vol. 2, pp. 323 - 343, 2008.

[13] Y. Pati, R. Rezaiifar, and P. Krishnaprasad. Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. Conf. Rec. 27th Asilomar Conf. Signals, Syst. Comput., vol. 1, 1993.

[14] M. Elad and I. Yavneh. A plurality of sparse representation is better than the sparsest one alone. IEEE Trans. Information Theory, vol. 55, pp. 4701 - 4714, 2009.

[15] J. A. Tropp. Algorithms for simultaneous sparse approximation. Signal Processing, vol. 86, pp. 572 - 602, 2006.

[16] M. Aharon, M. Elad, and A. Bruckstein. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans. on Signal Processing, vol. 54, pp. 4311 - 4322, 2006.

[17] P. Chatterjee and P. Milanfar. Clustering-based denoising with locally learned dictionaries. IEEE Trans. Image Processing, vol. 18, pp. 1438 - 1451, 2009.

[18] T. Kolda and B. Bader. Tensor decompositions and applications. SIAM Review, vol. 51, pp. 455 - 500, 2009.

Table 1: Summary of PSNR and SSIM

Images	*Methods*	$\sigma = 10$		$\sigma = 20$		$\sigma = 30$		$\sigma = 40$	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
House	Proposed	38.03	0.947	**34.89**	**0.912**	**32.96**	**0.891**	**31.41**	**0.873**
	BM3D	38.05	0.950	34.63	0.908	32.72	0.885	31.16	0.863
	NCSR	**38.18**	**0.951**	34.79	0.910	32.72	0.888	31.31	**0.873**
	KSVD	36.97	0.931	33.83	0.896	31.75	0.867	29.91	0.834
Barbara	Proposed	**36.18**	**0.968**	**32.54**	**0.935**	**30.19**	0.896	28.44	0.853
	BM3D	35.81	0.967	32.28	0.933	30.16	0.896	28.26	0.851
	NCSR	35.87	0.967	32.30	**0.935**	29.98	**0.897**	**28.48**	**0.858**
	KSVD	35.10	0.959	31.23	0.908	28.83	0.849	27.07	0.795
Boat	Proposed	**35.22**	**0.934**	31.60	0.875	29.54	0.828	28.09	0.788
	BM3D	35.16	**0.934**	**31.63**	**0.878**	**29.66**	**0.832**	**28.16**	**0.791**
	NCSR	35.13	0.933	31.54	0.875	29.47	0.828	28.06	0.787
	KSVD	34.71	0.926	31.02	0.855	28.90	0.798	27.43	0.752
Lena	Proposed	38.57	0.973	**34.79**	**0.944**	32.43	0.915	**30.82**	0.889
	BM3D	**38.64**	**0.974**	**34.79**	0.943	**32.50**	0.914	30.64	0.884
	NCSR	38.21	0.972	34.47	**0.944**	32.09	**0.916**	30.76	**0.893**
	KSVD	37.84	0.968	34.02	0.934	31.73	0.902	30.08	0.871
Hill	Proposed	34.76	**0.918**	31.39	0.841	29.56	0.785	28.32	0.742
	BM3D	34.72	0.917	**31.47**	**0.846**	**29.75**	**0.795**	**28.48**	**0.751**
	NCSR	**34.77**	**0.918**	31.39	0.844	29.53	0.789	28.30	0.743
	KSVD	34.29	0.907	30.83	0.817	28.93	0.753	27.61	0.704
Monarch	Proposed	34.98	**0.971**	**30.94**	**0.937**	**28.71**	**0.902**	**27.18**	0.869
	BM3D	34.56	0.969	30.63	0.932	28.59	0.897	26.88	0.859
	NCSR	**35.01**	**0.971**	30.93	0.936	28.69	**0.902**	27.03	**0.870**
	KSVD	34.02	0.961	30.12	0.920	28.03	0.881	26.61	0.846