

Adaptive Block Compressive Sensing for Noisy Images

Hui-huang Zhao, Paul L. Rosin, Yu-Kun Lai, Jing-hua Zheng and Yao-nan Wang

Abstract This paper develops a novel adaptive gradient-based block compressive sensing (AGbBCS_SP) methodology for noisy image compression and reconstruction. The AGbBCS_SP approach splits an image into blocks by maximizing their sparsity, and reconstructs images by solving a convex optimization problem. The main contribution is to provide an adaptive method for block shape selection, improving noisy image reconstruction performance. Experimental results with different image sets indicate that our AGbBCS_SP method is able to achieve better performance, in terms of peak signal to noise ratio (PSNR) and computational cost, than several classical algorithms.

Key words: Block Compressive Sensing (CS) Block Compressive Sensing (CS); Adaptive; Convex Optimization; Sparsity

1 Introduction

Compressive Sensing (CS) is a sampling paradigm that provides signal compression at a significantly lower rate than the Nyquist rate [8]. It has been successfully applied in a wide variety of applications in recent years, including image process-

Hui-huang Zhao and Jing-hua Zheng
Hunan Provincial Key Laboratory of Intelligent Information Processing and Application, Hunan, China, and College of Computer Science and Technology, Hengyang Normal University, Hengyang China e-mail: happyday.huihuang@gmail.com

Paul L. Rosin and Yu-Kun Lai
School of Computer Science and Informatics, Cardiff University, Cardiff, UK

Yao-nan Wang
College of Electrical and Information Engineering, Hunan University, Changsha, China e-mail: yaonan@hnu.cn

ing [5, 15, 18], Internet of things [23, 16], video [26, 17], and solder joint image compression [28].

In this paper, we develop a novel CS algorithm named AGbBCS_SP for image compression and reconstruction, which is particularly beneficial for noisy images. The main contributions of this paper are summarized as follows:

- We propose a multi-shape block splitting strategy for block Compressive Sensing. Besides splitting the image into square blocks, we also split it into rectangular blocks with different shapes.
- Our adaptive Compressive Sensing scheme makes a practical assumption that only a small, randomly chosen image part requires to be known. Our method automatically selects the appropriate block shape which maximizes the sparsity of the signal in the known region.
- We propose an adaptive approach to selecting a suitable control factor, by comparing the sparsity of the reconstruction results.

2 Related work

Compressive Sensing Algorithms: In recent years, many methods have been proposed which can be roughly divided into several categories: (1) Convex Optimization Algorithms. These techniques solve a convex problem which is used to approximate the target signal, including Greedy Basis Pursuit (GBP) [12]. (2) Greedy Iterative Algorithms. These methods include Orthogonal Matching Pursuit (OMP) [24], Compressive Sampling MP (CoSaMP) [20] and Subspace Pursuit (SP) [6]. (3) Iterative Thresholding Algorithms. Such as Hard thresholding [2]. (4) Combinatorial / Sub-linear Algorithms. such as Heavy Hitters on Steroids (HHS) [22]. (5) Non Convex Minimization Algorithms. This techniques recover compressive sensing signals from far less measurements by replacing the l_1 -norm by the l_p -norm where $p \leq 1$ [4]. (6) Bregman Iterative Algorithms. When applied to CS problems, the iterative approach using Bregman distance regularization achieves reconstruction in four to six iterations [21].

Block Based Compressive Sensing (BCS): In the methods above, a column or row of an image is normally viewed as a vector. But in many applications the nonzero elements of sparse vectors tend to cluster in blocks [9]. In order to improve the performance, [11] proposed and studied block compressive sensing for natural images and this method involves Wiener filtering and projection onto the convex set and hard thresholding in the transform domain. [19] proposed a BCS_SPL method with a variant of projected Landweber (PL) iteration and smoothing. [10] developed BCS_SPL methods based on a smoothed projected Landweber reconstruction algorithm. BCS_SPL has obvious defects since the Wiener filter and iterative projected Landweber discard partial information in the image. [25] proposed a block compressed sensing method based on iterative re-weighted l_1 norm minimization. [27] developed a block compressed sensing method for solder joint images based on CoSaMP.

3 Compressive Sensing Methodology

Given an image, the first step of CS is the construction of a k -sparse representation, where k is the number of the non-zero entries of the sparse signal. And most natural signals can be made sparse by applying orthogonal transforms, such as Wavelet Transform, Fast Fourier Transform and Discrete Cosine Transform (DCT) [3].

For a noisy image, Compressive Sensing can be represented as:

$$y = \Phi\Psi s + w, \quad (1)$$

where w is an N -dimensional noise signal (or measurement error), Ψ is an $N \times N$ orthogonal basis matrix and Φ is an $M \times N$ random measurement matrix ($M < N$). As expected, signal x Eq.(1) may be estimated from measurement y by solving the convex minimization problem [20] as follows.

$$\underset{x}{\operatorname{argmin}} \|\Phi x - y\|_2^2 + \lambda \|x\|_1. \quad (2)$$

Generally Eq.(2) is a constrained minimization problem of a convex function. One of the simplest methods for solving a convex minimization problem is the gradient-based algorithm which generates a sequence x_k via

$$x_0 \in \mathbb{R}^N, x_k = x_{k-1} - t_k \nabla g(x_{k-1}), \quad (3)$$

where $g(x)$ is a convex function, and $t_k > 0$ is a suitable step size. For a signal in Eq.(1), let us think about an objective function $F(x) = g(x) + f(x)$, where $g(x)$ is convex, and $f(x) = \lambda \|x\|_1$. In our method, it is more natural to study the closely related problem Eq.(2)

At point x_{k-1} , the function $F(x)$ can be approximated by a quadratic function

$$Q_L(x, x_{k-1}) = g(x_{k-1}) + \langle (x - x_{k-1}), \nabla g(x_{k-1}) \rangle + \frac{1}{2t_k} \|x - x_{k-1}\|_2^2, \quad (4)$$

This problem can be solved by the gradient-based method, in which t_k is replaced by a constant $1/L$ which is related to the Lipschitz constant [1].

4 The Adaptive Block Compressive Sensing with Sparsity

4.1 Multi-shape Block Split Strategy

Given an $N_1 \times N_2$ image, it is split into small blocks of size $n_1 \times n_2$. Let f_i represent the vectorized signal of the i -th block through raster scanning, $i=1, 2, \dots, K$, and $K = \frac{N_1 N_2}{n_1 n_2}$. One is able to get an m -dimensional sampled vector y_B through the following linear transformation,

$$y_B = \Phi_B f_i, \quad (5)$$

where Φ_B is an $m \times n_1 n_2$ measurement matrix, $m \ll n_1 n_2$. The block CS method is memory efficient as we just need to store an $m \times n_1 n_2$ Gaussian random matrix Φ_B , rather than a full $M \times N_1 N_2$ one. Small data requires less memory storage and allows faster processing, while large data produces more accurate reconstruction.

In existing methods, the blocks in the Block Compressive Sensing are fixed as squares. However, there are many different block aspect ratios with the same number of pixels. Unlike common methods, we split the image into different shapes. Given an $N \times N$ image (assuming N is a power of 2 for simplicity), the shape of block is $w \times h$, so

$$\begin{cases} w = 2^a, \\ h = 2^b, \\ a = 0, 1, 2, 3, \dots, \log_2 N. \\ b = \log_2 N - a, \end{cases} \quad (6)$$

For example, 9 aspect ratios are defined to split a 256×256 image with the following block-shapes: 1×256 , 2×128 , 4×64 , 8×32 , 16×16 , 32×8 , 64×4 , 128×2 and 256×1 . As we will discuss later in section 5.1, some block shapes (especially those closer to squares) are more likely to provide effective reconstruction. Also, using closer-to-square blocks also means that these blocks can be fit in smaller square regions, e.g. 8×32 , 16×16 , 32×8 blocks can be fit in 32×32 squares, whereas 1×256 blocks cannot. As we will discuss in section 4.2, this makes adaptive selection more effective. Detailed discussions will be presented in experimental results.

4.2 Adaptive Block Shape Selection

In most cases, the information of the entire signal (image) is unknown. It is hard to select one block shape from several shapes if the image content is unknown. So we make a practical assumption that only a small part of the image is known and propose a new approach based on sparsity for block shape selection. We highlight the block shape selection step in our approach.

First, we randomly select a small percentage of image pixels that make up known regions. These regions are then split into smaller block shapes considering the various aspect ratios specified in Eq.(6). We reconstruct them, calculate their sparsity, and then select the block shape which maximizes sparsity.

For an image, firstly, it is split into T non-overlapping regions with size $P \times Q$, where $K_1 = T \times p$ are known regions, and p is the proportion. So K_1 regions (size $P \times Q$) are selected. There are K_2 block sizes in Eq.(6) $w_k \times h_k, (k = 1, 2, \dots, K_2)$ that fit within $P \times Q$ regions. Then for K_1 regions (size $P \times Q$), they are split into K_3 blocks with size of $w_k \times h_k$. Given that \hat{x} is defined as the reconstructed result in Eq.(2), the summed sparsity of its blocks is defined as

Algorithm 1: Block Shape Selection with Sparsity

Input : An input image s , a percentage p ;
Output The selected block size $w \times h$
:
Procedure:
Step 1: Split s into T blocks with size of $P \times Q, K_1 = T \times p$
 K_1 regions (each of size $P \times Q$) are selected, and those regions collectively form \hat{s} .
Step 2: K_2 block shapes are considered: $w_1 \times h_1, w_2 \times h_2, \dots, w_{K_2} \times h_{K_2}$.
 \hat{s} is split into K_3 blocks altogether with $w_k \times h_k$ through Eq.(6)
For the k -th block size $\hat{s} = \{\hat{s}^{(k)}(1), \hat{s}^{(k)}(2), \dots, \hat{s}^{(k)}(K_3)\}$.
for $k = 1$ to K_2 **do**
 $\hat{s}^{(k)} = \emptyset$.
 for $j = 1$ to K_3 **do**
 Add a new signal $\hat{s}^{(k)}(j)$ to $\hat{s}^{(k)}$.
 end
end
for $k = 1$ to K_2 **do**
 Get \hat{x}^k through Eq.(2) with $\hat{s}^{(k)}$
 $S_{p_k} = I_{\varepsilon}^0(\hat{x}^k \leq \varepsilon)$ through Eq.(7),
end
 $S_{p_d} = \max\{S_{p_1}, S_{p_2}, \dots, S_{p_k}\}$
The d -th block shape is chosen, and the block size is $w_d \times h_d$.
Output w_d and h_d .

$$S_p = I_{\varepsilon}^0(\hat{x}_{i,j} \leq \varepsilon), \quad (7)$$

where $\hat{x}_{i,j}$ is the element at location (i, j) in \hat{x} the reconstructed result and $I_{\varepsilon}^0(\cdot)$ is a function defined in [13]. Thus, we propose the adaptive block shape selection with sparsity algorithm whose details are shown in Algorithm 1. For example, given a 256×256 image, we set $p = 0.25$. We consider splitting the image into $T = 64$ regions of size $P \times Q = 32 \times 32$, and $K_1 = 64 \times 0.25 = 16$ blocks are randomly selected, so that $K_3 = 16 \times 4 = 64$. With 32×32 regions, we consider $K_2 = 3$ block sizes 8×32 , 16×16 and 32×8 which fit within the region.

4.3 Adaptive Block Compressive Sensing with Sparsity Algorithm

During the minimization of Eq.(2), λ can be used to improve the result with different sampling rate. Usually $\lambda = M/4$, but in our proposed method, we set $\lambda \in [1, 100]$, and we adaptively choose λ such that the largest sparsity is achieved. Thus, we propose our AGbBCS_SP algorithm whose details are shown in Algorithm 2, where the basic sparse optimization is based on [28].

Algorithm 2: Adaptive Gradient-based Block Compressive Sensing

Input : An image I of size $N \times N$; a sparse signal transform matrix $\Psi \in \mathbb{R}^{N \times N}$; a measurement matrix $\Phi \in \mathbb{R}^{M \times N}$; M is the sample number; Lipschitz constant $L = 0.5$; the number of iterations $J = M/4$.

Output The reconstructed image s .

:

Procedure

Step 1: I is split into T regions, and p is a percentage, $K_1 = T \times p$ regions are selected.

One block shape $W \times H$ is chosen by Algorithm 1, I is split into K_4 blocks with $W \times H$ block size.

Step 2: Set the block counter $k=1$, $\lambda = 1$, and the iteration counter $j=1$, $S_{p_{max}} = 0$.

while $\lambda \leq 100$ **do**

while $k \leq K_2$ **do**

 Transform each block into a data vector; $y_0 = x_0 = 0 \in \mathbb{R}^N$, $t_1 = 1$;

while $j \leq J$ **do**

$z_j^k = PL(y_j^k)$, solved through [28].

$$t_{j+1}^k = \frac{1 + \sqrt{1 + 4t_j^{k2}}}{2}$$

$x_j^k = \operatorname{argmin}\{F(x^k) : x^k = z_j^k, x_{j-1}^k\}$

$$y_{j+1}^k = x_j^k + \frac{t_j^k}{t_{j+1}^k}(z_j^k - x_j^k) + \frac{t_j^{k-1}}{t_{j+1}^k}(x_j^k - x_{j-1}^k)$$

end

 Collect all the \hat{x}_j^k to form \hat{x} .

end

$S_p = l_\varepsilon^0(\hat{x} \leq \varepsilon)$ through Eq.(7),

 If $S_p > S_{p_{max}}$

$S_{p_{max}} = S_p$

$\hat{x} = \hat{x}$

Endif

end

$s' = \Psi^{-1} \hat{x}$.

For each one-dimensional data vector in s' , transform it into a $W \times H$ block.

Collect all the blocks to form the reconstructed image s .

5 Experiments and Discussion

In order to evaluate the quality of the reconstructed results, many researchers used the Peak Signal to Noise Rate (PSNR) to measure the result quality in image processing. In our study, the PSNR is also used to compare the experimental results. The experiments were implemented on a Intel Core i5 with 2.70 GHz CPU. The test images include some standard ones (such as *woman*), INRIA Holidays dataset (812 images) [14] to which *salt & pepper* noise is added with $\delta = 0.05$ by default. Since some methods require the image size to be a power of 2, we have cropped all the images to 256×256 .

5.1 Experiments with different block aspect ratios

Given a 256×256 image, the block-shapes 1×256 , 2×128 , 4×64 , 8×32 , 16×16 , 32×8 , 64×4 , 128×2 and 256×1 , are considered. We used the INRIA Holidays dataset, containing 812 images and the noise is set $\delta = 0.05$. With sample number $M = 128$ and $\lambda = M/4$, we run different block shapes. Then we select the best shape, and the number of times that each block shape is best is shown in figures 1 (a) and (b) for the two datasets, respectively.

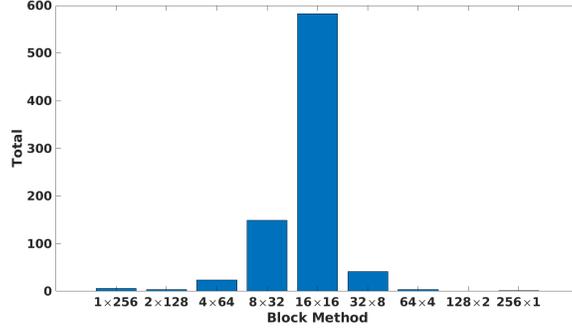


Fig. 1 The number of images that each block shape is best in the INRIA Holidays datasets.

We can find that a square block cannot always get the best results, and 8×32 , 16×16 , and 32×8 can achieve the top three results. So in our AGbBCS_SP method, three block shapes are chosen. As described above, we consider splitting a 256×256 image into 64 regions, each of size 32×32 , and $64 \times 0.25 = 16$ blocks are randomly selected to calculate sparsity in the three block shapes. Then we choose the block shape which can get maximum sparsity for the given image.

5.2 The comparison of reconstruction results

Now let us compare the proposed AGbBCS_SP with the popular methods SP [6], OMP [24], BOMP [9], CoSaMP [7], BCoSaMP [27] and BCS_SPL [10]. In BOMP, BCoSaMP and BCS_SPL, the block size is set to a square block (size 16×16). The test image *woman* is used (size 256×256) with added noise $\delta = 0.03$, as shown in figure 2 (a). The reconstruction results based on popular methods with sample number $M = 200$ are shown in figures 2 (b-g) and the reconstruction result based on our AGbBCS_SP with the same sample number, is shown in figure 2 (h).

We can see that our method can achieve a better result than SP, OMP, BOMP, CoSaMP, BCoSaMP and BCS_SPL. With more noise added and $M = 128$ in test image *woman*, the PSNR comparisons are shown in figure 3(a). One can see from

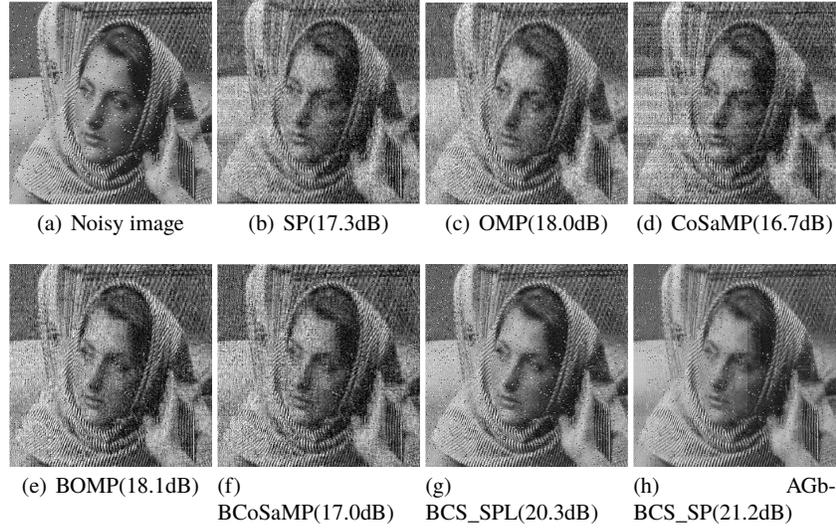


Fig. 2 Reconstruction results based on different methods

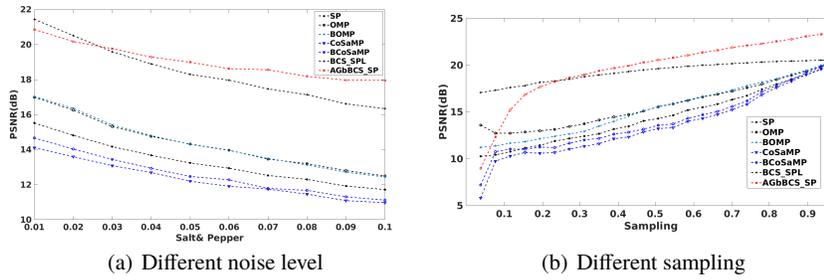


Fig. 3 PSNR comparison based on different noise level added

figure 3 (b) that when $\delta > 0.025$, our method achieves a better PSNR result than BCS_SPL. With increasing samples in the noisy image (see figure 2 (a)) the PSNR comparisons are shown in figure 3 (b) that. And compared to SP, OMP, BOMP, CoSaMP and BCoSaMP, our method can achieve best result.

In the next experiment, we used the INRIA Holidays datasets with added noise $\delta = 0.05$. The comparison results are shown through the experiments with different sample numbers (from 0.1 to 0.9). The results are shown in figures 4 (a) and (b).

From figure 4, one can see that the proposed approach can always obtain better results in terms of PSNR as compared to SP, OMP, BOMP, CoSaMP, and BCoSaMP. Increasing the number of samples can improve the reconstruction results. When the sampling rate $u = M/N > 0.3$, the proposed algorithm can achieve better results than BCS_SPL too. With an increasing number of samples, BCS_SPL gets worse reconstruction results.

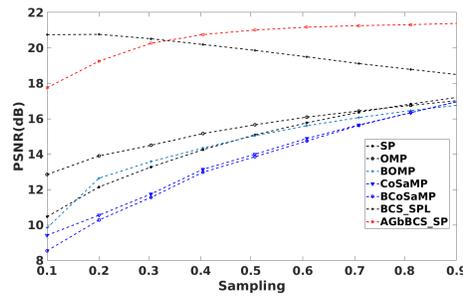


Fig. 4 Quantitative comparison based on different methods for INRIA datasets.

6 Conclusions

This paper proposes an adaptive gradient-based block compressive sensing (AG-bBCS_SP) approach on the basis of the sparsity of the image. Experiments reveal that, in block compressive sensing, the square block shape cannot always produce the best results. Our algorithm can adaptively achieve better results by using the sparsity of pixels to adaptively select block shape. The proposed algorithm can achieve the best results in average PSNR than classical algorithms, SP, OMP, BOMP, CoSaMP, BCS_SPL and BCoSaMP with different datasets.

Acknowledgements

This work was supported by National Natural Science Foundation of China (61733004, 61503128, 61602402), Science and Technology Plan Project of Hunan Province (2016TP102), Scientific Research Fund of Hunan Provincial Education Department (16C0311), and Hunan Provincial Natural Science Foundation (2017JJ4001). We would like to thank NVIDIA for the GPU donation.

References

1. Beck, A., Teboulle, M.: A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *Siam Journal on Imaging Sciences* **2**(1), 183–202 (2009)
2. Blumensath, T., Davies, M.E.: Iterative hard thresholding for compressed sensing. *Applied and Computational Harmonic Analysis* **27**(3), 265–274 (2009)
3. Cai, T.T., Wang, L.: Orthogonal matching pursuit for sparse signal recovery with noise. *Information Theory, IEEE Transactions on* **57**(7), 4680–4688 (2011)
4. Chartrand, R.: Exact reconstruction of sparse signals via nonconvex minimization. *Signal Processing Letters, IEEE* **14**(10), 707–710 (2007)

5. Cui, H., Zhang, S., Gan, X., Shen, M., Wang, X., Tian, X.: Information recovery via block compressed sensing in wireless sensor networks. In: 2016 IEEE International Conference on Communications (ICC), pp. 1–6 (2016)
6. Dai, W., Milenkovic, O.: Subspace pursuit for compressive sensing signal reconstruction. *Information Theory, IEEE Transactions on* **55**(5), 2230–2249 (2009)
7. Davenport, M.A., Needell, D., Wakin, M.B.: Signal space cosamp for sparse recovery with redundant dictionaries. *IEEE Transactions on Information Theory* **59**(10), 6820–6829 (2013)
8. Donoho, D.L.: Compressed sensing. *IEEE Transactions on information theory* **52**(4), 1289–1306 (2006)
9. Eldar, Y.C., Kuppinger, P., Băcilă, H.: Block-sparse signals: Uncertainty relations and efficient recovery. *IEEE Transactions on Signal Processing* **58**(6), 3042–3054 (2010)
10. Fowler, J.E., Mun, S., Tramel, E.W.: Multiscale block compressed sensing with smoothed projected landweber reconstruction. In: *Signal Processing Conference, 2011 European*, pp. 564–568 (2015)
11. Gan, L.: Block compressed sensing of natural images. In: *Digital Signal Processing, 2007 15th International Conference on*, pp. 403–406. IEEE (2007)
12. Huggins, P.S., Zucker, S.W.: Greedy basis pursuit. *Signal Processing, IEEE Transactions on* **55**(7), 3760–3772 (2007)
13. Hurley, N., Rickard, S.: Comparing measures of sparsity. *IEEE Transactions on Information Theory* **55**(10), 4723–4741 (2009)
14. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. *Computer Vision–ECCV 2008* pp. 304–317 (2008)
15. Lu, H., Li, B., Zhu, J., Li, Y., Li, Y., Xu, X., He, L., Li, X., Li, J., Serikawa, S.: Wound intensity correction and segmentation with convolutional neural networks. *Concurrency and computation: practice and experience* **29**(6), e3927 (2017)
16. Lu, H., Li, Y., Chen, M., Kim, H., Serikawa, S.: Brain intelligence: go beyond artificial intelligence. *Mobile Networks and Applications* **23**(2), 368–375 (2018)
17. Lu, H., Li, Y., Mu, S., Wang, D., Kim, H., Serikawa, S.: Motor anomaly detection for unmanned aerial vehicles using reinforcement learning. *IEEE Internet of Things Journal* (2017)
18. Lu, H., Li, Y., Uemura, T., Kim, H., Serikawa, S.: Low illumination underwater light field images reconstruction using deep convolutional neural networks. *Future Generation Computer Systems* (2018)
19. Mun, S., Fowler, J.E.: Block compressed sensing of images using directional transforms. In: *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pp. 3021–3024. IEEE (2009)
20. Needell, D., Tropp, J.A.: Cosamp: Iterative signal recovery from incomplete and inaccurate samples. *Applied and Computational Harmonic Analysis* **26**(3), 301–321 (2009)
21. Osher, S., Mao, Y., Dong, B., Yin, W.: Fast linearized bregman iteration for compressive sensing and sparse denoising. *arXiv preprint arXiv:1104.0262* (2011)
22. Qaisar, S., Bilal, R.M., Iqbal, W., Naureen, M., Lee, S.: Compressive sensing: from theory to applications, a survey. *Communications and Networks, Journal of* **15**(5), 443–456 (2013)
23. Serikawa, S., Lu, H.: Underwater image dehazing using joint trilateral filter. *Computers & Electrical Engineering* **40**(1), 41–50 (2014)
24. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. *Information Theory, IEEE Transactions on* **53**(12), 4655–4666 (2007)
25. Unde, A.S., Deepthi, P.: Block compressive sensing: Individual and joint reconstruction of correlated images. *Journal of Visual Communication and Image Representation* **44**, 187–197 (2017)
26. Zhao, C., Ma, S., Zhang, J., Xiong, R., Gao, W.: Video compressive sensing reconstruction via reweighted residual sparsity. *IEEE Transactions on Circuits and Systems for Video Technology* **27**(6), 1182–1195 (2017)
27. Zhao, H., Wang, Y., Qiao, Z., Fu, B.: Solder joint imagery compressing and recovery based on compressive sensing. *Soldering & Surface Mount Technology* **26**(3), 129–138 (2014)
28. Zhao, H., Zhao, H., Chen, J., Chen, J., Xu, S., Xu, S., Wang, Y., Wang, Y., Qiao, Z., Qiao, Z.: Compressive sensing for noisy solder joint imagery based on convex optimization. *Soldering & Surface Mount Technology* **28**(2), 114–122 (2016)