Fast Recovery of Compressive Sensed Images via Multiple Thresholding Operators

Evgeny Belyaev
ITMO University
Saint-Petersburg, Russia
eabelyaev@itmo.ru

***Abstract*—The problem of fast compressive sensed image recovery via iterative thresholding algorithm is considered. At each iteration of the algorithm, we propose to randomly select high or low complex thresholding operator in order to provide a good trade-off between the overall computational complexity and the quality of the recovery. As an example, well-known Block-matching and 3D filtering (BM3D) is used as a high complex thresholding operator, while simple 2-D DCT based thresholding is used as a low complex one. Experimental resuts show that the resulting approach is around 5 times less complex than the recovery based on BM3D only providing similar quality of reconstruction.**

I. Introduction

Compressive sensing [1], [2] is a framework which can be used for development of new cheap image sensors which observe a small number (e.g., 10-20%) of random linear measurements instead of all image pixels. A linear reconstruction by inverse transform, cannot, in general, recover the signal from a small number of measurements. However, in [1], [2], it was shown that if the signal is sparse in some known transform domain, then a stable reconstruction is possible. However, a high computational complexity of the reconstruction is still an important issue restricting the use of the compressive sensing applications in real-life.

In this paper we propose a fast recovery of compressive sensed images via multiple thresholding operators which allows to achieve a good trade-off between the overall computational complexity and the quality of the recovery. The rest of the paper is organized as follows. In Section II, we shortly describe the sensing model and recovery via iterative thresholding. Then, we introduce the proposed approach and show its efficiency via experimental results. Section III concludes the paper.

We use the following notations. The column vectors and matrices are denoted by boldfaced lowercase and uppercase letters, respectively, e.g., $\mathbf{v}$ and $\mathbf{A}$. The superscript $(.)^T$ denotes the transpose operation for a vector or a matrix, $(.)^{-1}$ is inverse operation for matrix, $\text{vec}(\mathbf{A})$ concatenates columns of $\mathbf{A}$ into a vector, $\hat{\mathbf{v}}$ means estimation of a vector $\mathbf{v}$, $\leftarrow$ means assignment operation, $|.|$ denotes a number of elements in vector or matrix.

II. Proposed approach

A. Sensing model and recovery via iterative thresholding

Let us consider a 2-D image $\mathbf{X} \in \mathbb{R}^{N \times N}$ of size $N \times N$ pixels. Following the compressive sensing framework [1], [3], the linear measurements are acquired as

$$\mathbf{y} = \mathbf{\Phi x}, \tag{1}$$

where $\mathbf{x} = \text{vec}(\mathbf{X})$, $\mathbf{\Phi} \in \mathbb{R}^{M \times N^2}$, $M < N^2$, denotes the measurement matrix. The ratio $\frac{M}{N^2}$ is known as *sensing rate*. We assume that the image has a sparse representation in a known basis, i.e. $\mathbf{x} = \mathbf{\Omega}^{-1}\boldsymbol{\vartheta}$, where $\mathbf{\Omega}$ is a $N^2 \times N^2$ representation matrix and $\boldsymbol{\vartheta}$ is the sparse vector of the transform coefficients. Under this assumption, the recovery is formulated as searching for the sparsest vector $\boldsymbol{\vartheta}$ which satisfies $\mathbf{y} = \mathbf{\Phi\Omega}^{-1}\boldsymbol{\vartheta}$. Herewith, the l_1–norm can be used as a sparsity metric. Following our previous work [4], we reconstruct the image by solving the l_1 minimization problem using *iterative soft thresholding* (IST) [5], [6], [7]. An image at iteration k is estimated as $\hat{\mathbf{x}}^k = soft\left(\hat{\mathbf{x}}^{k-1} + \Delta\hat{\mathbf{x}}^k, \sigma_k\right)$, where $\Delta\mathbf{x}^k = \mathbf{\Phi}^T\left(\mathbf{y} - \mathbf{\Phi}\hat{\mathbf{x}}^{k-1}\right)$, the initial estimate $\hat{\mathbf{x}}^0$ is zero-vector or an image provided by another recovery method, and the operator $soft(\mathbf{x}, \sigma)$ includes three main steps:

1) A sparsifying transform with matrix $\mathbf{\Omega}$ is applied for an image as $\boldsymbol{\vartheta} = \mathbf{\Omega x}$.
2) Soft thresholded transform coefficients $\tilde{\boldsymbol{\vartheta}} = \{\tilde{\vartheta}_i\}$ are calculated as

$$\tilde{\vartheta}_i = \begin{cases} 0, & \text{if } |\vartheta_i| < \sigma, \\ \left(1 - \dfrac{\sigma}{|\vartheta_i|}\right)\vartheta_i, & \text{otherwise.} \end{cases} \tag{2}$$

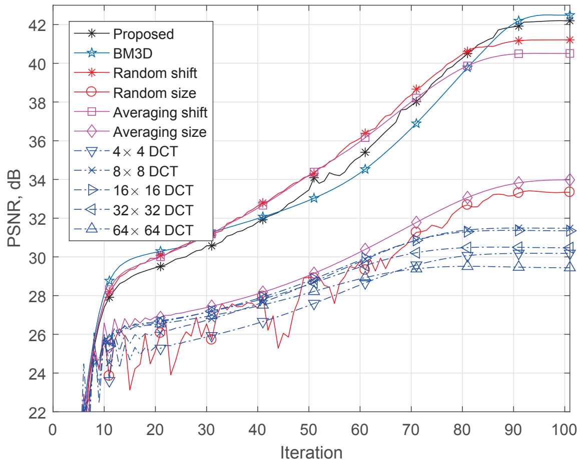
3) A soft thresholded image is calculated as $\tilde{\mathbf{x}} = \mathbf{\Omega}^{-1}\tilde{\boldsymbol{\vartheta}}$.

B. Recovery utilizing multiple thresholding transforms

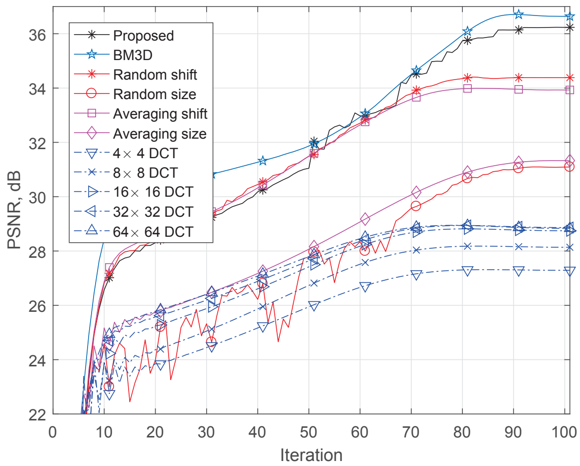
From computational complexity point of view, two-dimensional dicrete cosine transform (2-D DCT) could be considered as the fast sparsifying transform. In this case, at each iteration, an input frame $\mathbf{X}$ is divided into non overlapped blocks size of $L \times L$. And the operator $soft(\mathbf{x}, \sigma)$ is performed separately for each block, i.e., $\mathbf{\Omega}_{L \times L}$ is 2-D DCT transform size of $L \times L$.

The sparsity level provided by $\mathbf{\Omega}_{L \times L}$ depends on image properties: for some image areas, such as a flat areas, relatively large L provides better sparsity, while smaller L are more efficient for areas with many details. In order to improve the recovery performance, similar to [8], we could perform the thresholding for several block sizes from set $\mathcal{L}$ and then

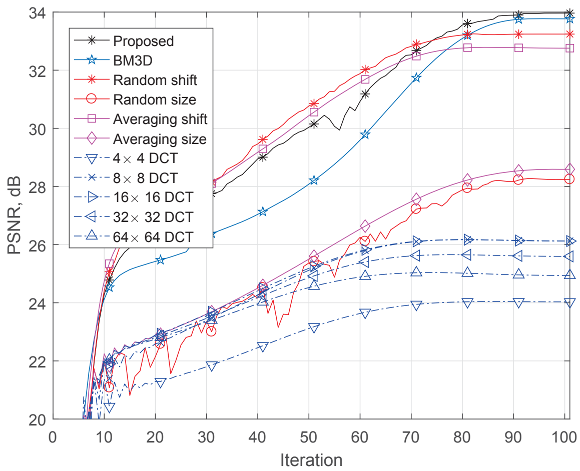
ISSN 2305-7254



(a) Akiyo



(b) Foreman



(c) Container

Fig. 1. Performance comparison of different recovery methods for sensing rate 20%

compute an average as follows:

$$\tilde{\mathbf{x}}_{\mathcal{L}} = \frac{1}{|\mathcal{L}|}\sum_{L\in\mathcal{L}} \mathbf{\Omega}^{-1}_{L\times L}\tilde{\boldsymbol{\theta}}_{L\times L}. \quad (3)$$

This approach allows to achieve a higher sparsity level with a price of higher computational complexity, since at each iteration we need to compute $|\mathcal{L}|$ transforms instead of one. In order to reduce the computational complexity, in [4] we proposed to use at each iteration a pseudo random value for L chosen with uniform probabilities from the set $\mathcal{L}$, i.e., perform only one randomly selected transform at each iteration. Fig. 1 and Table I show Peak Signal-to-Noise Ratio (PSNR) for each iteration, when 4×4 DCT, ..., 64×64 DCT used at each iteration, and when the averaging (3) is performed with set $\mathcal{L} = \{4, 8, 16, 32, 64\}$ (called *Averaging size*), and when the block size is randomly selected from the same set $\mathcal{L}$ (called *Random size*). One can see that the both the averaging and random selection provide much higher PSNR values than any single 2-D DCT. Herewith, execution time (all the presented methods are implemented in MATLAB) of the averaging is around 5 times higher than that of the random selection (see in Table I).

The same idea can be used when a single sparsifying transform $L\times L$ DCT is performed with shift of blocks grid by a random vector $(s_x, s_y) \in \{-S/2, ..., S/2\}$. In our experiments, we used 8×8 DCT with $S = 1$ for the averaging approach (see *Averaging shift*). In this case it need to perform $|\mathcal{L}| = 9$ transforms at each iteration which makes it the slowest method among the considered ones. Herewith, the approach with the random selection (see *Random shift*) allows to perform the recovery with $S = L/2$, i.e., for all possible combinations of shifts. As a result, it provides even higher performance with much lower complexity comparing to the averaging approach.

However, considered above 2-D DCT based approach exploits only local redundancy of images, therefore, it achieves only moderate sparsity level of an image representation. In order to achieve higher sparsity level, non-local self-similarity of images should be exploited as well. In [9], an image denoising algorithm based on sparse 3-D transform-domain collaborative filtering has been proposed. It utilizes block matching and 3-D transform (BM3D) in order to exploit both local and non-local image similarities. Moreover, as it was shown in [10], BM3D can be also used as a thresholding operator in other applications, e.g., for image super-resolution or compressive sensed image recovery. However, the main drawback of BM3D is its high computational complexity caused by block matching, 3-D transforms computation and thresholding within 3-D transform domain.

In order to reduce the recovery complexity, in this paper we propose to combine the recovery via 2-D DCT with random shift with the BM3D in the following way. First, at iterations $i = 1, ..., i_1$, only the 2-D DCT with random shift is used. Then at iterations $i = i_1 + 1, ..., i_{max}$, we run BM3D with probability ρ, and the 2-D DCT with random shift with probability $(1-\rho)$. As a result, in average, we use BM3D only $\rho(i_{max} - i_1)$ times instead of i_{max} as in the original recovery approach. In our experiments (see Fig. 1 and Table I) we performed the proposed approach (called *Proposed*) with $i_1 = 50$ and $\rho = 0.2$. In all cases, i_{max} was set to 100, i.e., in average, the proposed approach uses the BM3D only 10 times.

TABLE I. PEAK SIGNAL-TO-NOISE RATIO (PSNR) AND EXECUTION TIME FOR DIFFERENT SENSING RATES

Recovery method	*Akiyo*				*Foreman*				*Container*			
	PSNR, dB				*PSNR*, dB				*PSNR*, dB			
	20%	15%	10%	Time,sec	20%	15%	10%	Time, sec	20%	15%	10%	Time, sec
4×4 DCT	30.18	27.64	25.62	61.9	27.29	25.32	23.29	61.5	24.03	21.72	20.00	59.2
8×8 DCT	31.48	29.51	27.45	16.7	28.13	26.55	24.89	17.2	26.1	24.45	22.39	17.5
16×16 DCT	31.35	29.62	27.88	6.0	28.74	27.50	26.09	6.0	26.1	24.57	22.70	6.8
32×32 DCT	30.47	29.04	27.69	3.3	28.86	27.78	26.44	3.3	25.59	24.15	22.45	3.2
64×64 DCT	29.44	28.35	27.28	2.4	28.83	27.91	26.69	2.4	24.93	23.82	22.38	2.4
Averaging size	33.98	31.91	29.72	86.2	31.32	29.98	28.32	85.0	28.59	26.82	24.65	82.0
Averaging shift	40.51	37.86	34.41	247.8	33.92	32.40	30.35	251.7	32.75	30.92	28.39	237.0
Random size	33.34	31.47	29.48	17.3	31.09	29.82	28.24	16.7	28.24	26.57	24.51	17.1
Random shift	41.21	38.77	35.36	16.5	34.38	32.94	30.94	16.8	33.24	31.38	29.01	15.8
BM3D	42.49	40.12	36.52	76.2	36.64	35.51	33.81	73.2	33.76	31.74	29.27	83.1
Proposed	**42.19**	**39.77**	**36.10**	**16.4**	**36.23**	**34.88**	**32.74**	**16.0**	**33.96**	**31.91**	**29.39**	**17.9**

One can see that the resulting recovery approach is around 5 times faster and provides similar quality of recovery measured in PSNR.

III. CONCLUSION

In this paper we presented fast recovery of compressive sensed images via multiple thresholding operators. This approach randomly switches between high and low complex thresholding operators providing a good trade-off between the overall computational complexity and the quality of the recovery. Therefore, it can be attractive for real-time recovery applications.

REFERENCES

[1] E. J. Candes, J. Romberg, and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, Feb 2006.

[2] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[3] J. Romberg, "Imaging via compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 14–20, March 2008.

[4] E. Belyaev, M. Codreanu, M. Juntti, and K. Egiazarian, "Compressive sensed video recovery via iterative thresholding with random transforms," *IET Image Processing*, January 2020.

[5] I. Daubechies and L. Massimo and, "Accelerated projected gradient method for linear inverse problems with sparsity constraints," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 764–792, 2008.

[6] C. Cartis and A. Thompson, "A new and improved quantitative recovery analysis for iterative hard thresholding algorithms in compressed sensing," *IEEE Transactions on Information Theory*, vol. 61, no. 4, pp. 2019–2042, April 2015.

[7] E. Belyaev, S. Forchhammer, and M. Codreanu, "Error concealment for 3-d dwt based video codec using iterative thresholding," *IEEE Communications Letters*, vol. 21, no. 8, pp. 1731–1734, Aug 2017.

[8] U. L. Wijewardhana, E. Belyaev, M. Codreanu, and M. Latva-Aho, "Signal recovery in compressive sensing via multiple sparsifying bases," in *2017 Data Compression Conference (DCC)*, April 2017, pp. 141–150.

[9] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, Aug 2007.

[10] K. Egiazarian, A. Foi, and V. Katkovnik, "Compressed sensing image reconstruction via recursive spatially adaptive filtering," in *2007 IEEE International Conference on Image Processing*, Sep. 2007, vol. 1, pp. 549–552.