BLOCK COMPRESSED SENSING OF NATURAL IMAGES

Lu Gan

Dept. of Electrical Engineering and Electronics,
University of Liverpool, L69 3GJ
Email: lu.gan@liv.ac.uk

ABSTRACT

Compressed sensing (CS) is a new technique for simultaneous data sampling and compression. In this paper, we propose and study block compressed sensing for natural images, where image acquisition is conducted in a block-by-block manner through the *same operator*. While simpler and more efficient than other CS techniques, the proposed scheme can sufficiently capture the complicated geometric structures of natural images. Our image reconstruction algorithm involves both linear and nonlinear operations such as wiener filtering, projection onto the convex set and hard thresholding in the transform domain. Several numerical experiments demonstrate that the proposed block CS compares favorably with existing schemes at a much lower implementation cost.

***Index Terms*—** Compressed sensing, random projections, nonlinear reconstruction, sparsity

1. INTRODUCTION

In conventional imaging systems, natural images are often first sampled into the digital format at a higher rate and then compressed through the JPEG or the JPEG 2000 codec for efficient storage purpose. However, this approach is not applicable for low-power, low resolution imaging devices (e.g., those used in a sensor network) due to their limited computation capabilities. Over the past few years, a new framework called as compressive sampling (CS) has been developed for simultaneous sampling and compression. It builds upon the groundbreaking work by Candes *et al.* [1] and Donoho [2], who showed that under certain conditions, a signal can be precisely reconstructed from only a small set of measurements. The CS principle provides the potential of dramatic reduction of sampling rates, power consumption and computation complexity in digital data acquisitions. Due to its great practical potentials, it has stirred great excitements both in academia and industries in the past few years [3,4]. However, most of existing works in CS remain at the theoretical study. In particular, they are not suitable for real-time sensing of natural image as the sampling process requires to access the entire target at once [5]. In addition, the reconstruction algorithms are generally very expensive.

In this paper, we propose block-based sampling for fast CS of natural images, where the original image is divided into small blocks and each block is sampled independently using the *same measurement* operator. The possibility of exploiting block CS is motivated by the great success of block DCT coding systems which are widely used in the JPEG and the MPEG standards. The main advantages of our proposed system include: (a) Measurement operator can be easily stored and implemented through a random undersampled filter bank; (b) Block-based measurement is more advantageous for real-time applications as the encoder does not need to send the sampled data until the whole image is measured; (c) Since each block is processed independently, the initial solution can be easily obtained and the reconstruction process can be substantially speeded up;

For natural images, our preliminary results show that block CS systems offer comparable performances to existing CS schemes with much lower implementation cost. The rest of this paper is organized as follows. Section 2 provides a brief review of the CS principle. Section 3 describes the sensing operator along with the linear reconstruction method. Section 4 presents the non-linear reconstruction algorithms. Section 5 reports the simulation results for natural images followed by conclusions in Section 6.

2. BACKGROUND

In this paper, we focus on the problem of discrete CS. Consider a length-N, real valued signal x. Suppose that we are allowed to take n ($n \ll N$) linear, non-adaptive measurement of x through the following linear transformation [1, 2]:

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

where y represents an $n \times 1$ sampled vector and Φ is an $n \times N$ measurement matrix. Since $n \ll N$, the reconstruction of x from y is generally *ill-posed*. However, the CS theory is based on the fact that x has a sparse representation in a known transform domain Ψ (e.g., the DCT and the wavelet). In other words, the transform-domain signal $f = \Psi x$ can be well approximated using only $d < n \ll N$ nonzero entries. It was proved in [1, 2] that when Φ and Ψ are incoherent, x can be well recovered from $n = \mathcal{O}(d \log N)$ measurements through *non-linear* optimizations. In the study of CS, a couple of the most important issues include (a) the design of sampling operator Φ; (b) the development of fast nonlinear reconstruction algorithms.

For 1-D signals, Φ was usually taken to be a Gaussian i.i.d matrix [1]. However, for 2D images, N can be fairly large (at the order of $10^4 - 10^6$), which makes the storage and computations of a Gaussian ensemble very difficult. Thus, [6] suggested to apply a partial random Fourier matrix in the wavelet domain, i.e., Φ takes the form of $\Phi = F_\Omega \Psi_{WT}$, where F_Ω represents an $n \times N$ matrix from random selection of n rows out of the $N \times N$ Fourier matrix, while Ψ_{WT} stands for the matrix representation of a wavelet transform. In [3], the *multiscale CS* was proposed, where different scales of wavelet coefficients are segregated and sampled with different partial Fourier ensembles. Other choices of Φ can be found in [4, 7].

For non-linear reconstruction algorithms, the *basis pursuit* (BP) optimization [2, 3, 8] aims to minimize $\|\Psi x\|_{l_1}$ under the constraint of (1). BP was founded on a solid theoretical basis which shows that x can be *exactly* recovered if it is strictly sparse in a certain transform domain [2]. For 2D images, another well-known reconstruction algorithm is through the minimization of total variation (TV) [1, 6]. Unfortunately, both the BP and the TV minimizations require fairly heavy computations. Several fast greedy algorithms have also been proposed, such as the orthogonal matching pursuit (OMP) [9], the tree-based OMP [10] and the stage-wise orthogonal matching pursuit (StOMP) [4]. Other algorithms include iterative soft-thresholding [6, 11] and projection onto convex sets [6].

Despite above-mentioned works, there still exists a huge gap between the theory and applications, especially for CS of natural images. In this paper, we study block CS for real-time imaging and provide simple algorithms for large-scale reconstruction.

3. BLOCK COMPRESSED SAMPLING

Consider an $I_r \times I_c$ image with $N = I_r I_c$ pixels in total and suppose we want to take n CS measurements. In block CS, the image is divided into small blocks with size of $B \times B$ each and sampled with the same operator. Let x_i represent the vectorized signal of the i-th block through raster scanning. The corresponding output CS vector y_i can be written as

$$y_i = \Phi_B x_i \tag{2}$$

where Φ_B is an $n_B \times B^2$ matrix, with $n_B = \lfloor \frac{nB^2}{N} \rfloor$. In our current work, Φ_B is an orthonormalized i.i.d Gaussian matrix [6]. For the whole image, the equivalent sampling operator Φ in (1) is thus a block diagonal matrix taking the following form

$$\Phi = \begin{bmatrix} \Phi_B & & & \\ & \Phi_B & & \\ & & \ddots & \\ & & & \Phi_B \end{bmatrix} \tag{3}$$

Note that block CS is memory efficient as we just need to store an $n_B \times B^2$ Gaussian ensemble Φ_B, rather than a full $n \times N$ one. Besides, from a multi-rate signal processing point of view, block CS can be implemented as a random 2D filter bank as shown in Fig. 1. Here, each FIR filter $H_i(z_1, z_2)$ (for $0 \le i \le n_B - 1$) is supported in the region of $B \times B$. In the special case when $n_B = 1$, the proposed system boils down to the random filter system proposed in [5]. Obviously, there is a trade-off in the selection of block dimension B. Small B requires less memory in storage and faster implementation, while large B offers better reconstruction performance [1]. From empirical studies, we suggest block dimension $B = 32$ hereafter.

Not only does the block CS provide a simple structure at the sender side, it also leads to a good and fast initial solution of x. When Φ is a full matrix, most existing works take the initial solution as the result of l_2 optimization [6], i.e., $\hat{x} = \Phi^\dagger y$, where the superscript $\dagger$ denotes the pseudo inverse. In block CS, we propose to obtain the initial solution from the *minimum mean square error* (MMSE) linear estimation [12]. Let $\hat{x}_i$ represent the 1-D version of the reconstructed signal in the i-th block. To minimize $\|x_i - \hat{x}_i\|_2$, we have $\hat{x}_i = \hat{\Phi}_B y_i$ [12], where the reconstruction matrix $\hat{\Phi}_B$ can be written as

$$\hat{\Phi}_B = R_{xx}\Phi_B^T(\Phi_B R_{xx} \Phi_B^T)^{-1}, \tag{4}$$

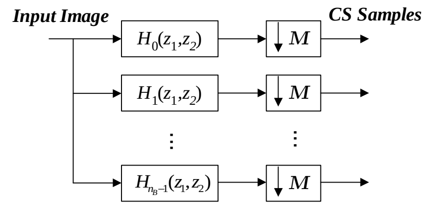


Fig. 1. Filter bank implementation of block CS. Here, $\mathbf{M}$ is a rectangular decimation matrix with $\det(\mathbf{M}) = B^2$.

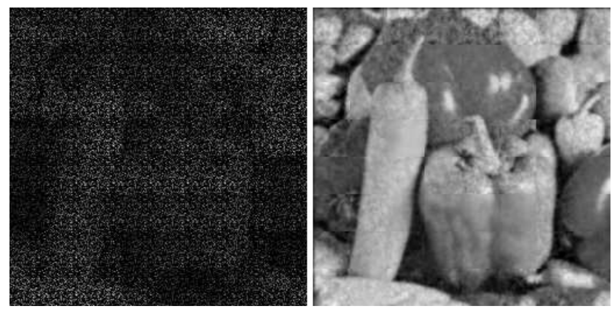


Fig. 2. Recovered 256×256 image *Peppers* from $n = 10000$ CS samples. Left: l_2 optimization; Right: MMSE linear reconstruction.

in which R_{xx} represents the autocorrelation function of the input signal. For natural images, we approximate R_{xx} using the AR(1) model with correlation coefficient $\rho = 0.95$. Such a model has been proved to work well in filter bank optimization for image coding [13, 14]. For a full Φ, the computation of the MMSE solution is prohibitively costly. In block CS, as the size of Φ_B is much smaller, $\hat{\Phi}_B$ can be easily calculated. As an example, Fig. 2 shows the reconstructed 256×256 image *Peppers*. It is obvious that the MMSE linear solution provides a much better reconstructed image than that of the l_2 optimization.

4. NON-LINEAR SIGNAL RECONSTRUCTION

To further improve the quality of the reconstructed images, we propose a 2-stage nonlinear reconstruction algorithm by exploiting the sparsity property. Our algorithm will use the techniques of hard thresholding and projection onto the convex set [6]. Before detailed explanation of the algorithm, let us first have a quick review of these techniques.

Projection onto convex set: Define $\mathcal{C}$ as the hyper-plane $\mathcal{C} = \{g : \Phi g = y\}$. For any arbitrary vector x, to find the closest vector $\mathcal{P}(x, y, \Phi)$ on $\mathcal{C}$, we can use the following formula [6]:

$$\mathcal{P}(x, y, \Phi) = x + \Phi^T(\Phi\Phi^T)^{-1}(y - \Phi x). \tag{5}$$

In the special case when Φ is an orthonormal matrix, i.e., $\Phi\Phi^T = \mathbf{I}$, (5) can be simplified into

$$\mathcal{P}(\Phi, y, x) = x + \Phi^T(y - \Phi x). \tag{6}$$

Hard thersholding: Hard thresholding [15] is widely used in the removal of Gaussian noise. The following function $\mathcal{H}(\Psi, x, K)$ describes this process. Simply speaking, the input signal is first transformed through Ψ to yield f. Then, the largest K coefficients of f

were kept, while the rest were set to zeros. After that, the inverse transform Ψ^{-1} is applied to yield the reconstructed signal.

---

function $x' = \mathcal{H}(\Psi, x, K)$
  $f = \Psi x$;
  Keep the largest K coefficients of f and set the remaining ones to zeros;
  $x' = \Psi^{-1} f$

4.1. Stage 1: Iterative Wiener Filtering and Hard Thresholding in the Lapped Transform Domain

As can be observed in Fig. 2, the MMSE linear reconstruction generates some blocking artifacts and noises. In our proposed non-linear method, Stage 1 is based on iterative spatial-frequency domain enhancement, as highlighted in Algorithm 1. In each iteration, the 3×3 Wiener filter is first applied in the *spatial domain* to reduce the blocking artifact and smooth the image. Then, the filtered signal is projected back to the convex set $\mathcal{C} = \{g : \ \Phi g = y\}$. After that, we apply the hard thresholding method [15] in the lapped transform (LT) domain to reduce Gaussian noise. And finally, the signal is again projected back to the convex set $\mathcal{C}$. Here, we use LT, rather than the wavelet, for frequency domain processing as the LT has much lower computational complexity [13]. Besides, it can simultaneously possess the linear-phase and the orthogonal properties, which is impossible for the wavelet (except for the Haar basis). Moreover, recent studies have shown that the LT can offer comparable image coding performance to the wavelet at various bit rates [14].

Through empirical studies, we suggest the maximum number of iterations in Stage 1 as $s_{max} = 5$. Besides, for hard-thresholding, the number of coefficients to be kept in the LT domain are $K_0 = K_1 = n/4$ and $K_2 = K_3 = K_4 = n/3$, in which n represents the total number of CS samples.

Algorithm 1 Iterative Wiener Filtering and Hard-thresholding in the Lapped Transform Domain

---

Input: Initial solution x_0: output from linear reconstruction;
  CS Sampling vectors: y;
  Maximum number of iterations: s_{max};
Output:
  for $s = 0$ to $s_{max} - 1$ **do**
    $x_{s,w} = wiener(x_s, [3, 3])$
    $\bar{x}_s = \mathcal{P}(\Phi, y, x_{s,w})$
    $\hat{x}_s = \mathcal{H}(\Psi_{LT}, \bar{x}_s, K_s)$
    $x_{s+1} = \mathcal{P}(\Phi, y, \hat{x}_s)$
  end for

4.2. Stage 2: Iterative Hard Thresholding through Frame expansions

In the application of signal denoising, it is well known that redundant frame expansions are superior to orthonormal bases. Some existing CS recovery algorithms have already applied frame expansions (e.g., using the curvelet [3]) for recovery of natural images. In Stage 2, we aim to exploit the sparsity of natural images in *two different classes* of frame expansions: the undecimated wavelet transform (UWT) [16] nand the oversampled lapped transform (OLT). The UWT provides a global description of the whole image, while the OLT offers a better representation of local structures through overlapped block-by-block processing. We implemented an iterative method as presented in Algorithm 2 through hard thresholding in frame expansions and projection onto the convex set. Our preliminary results indicate that by using two frame expansions instead of one, better reconstruction results can be obtained with faster convergence speed. In our simulations, "Daubechies-8" is used as the wavelet basis function. The OLT is implemented through the 8×16 basis functions in [13] with a decimation factor of 4. Besides, the maximum number of iteration is $s_{max} = 10$ and the values of K_s are $K_s = n/2$ ($0 \leq s \leq 2$) and $K_s = n/1.5$ ($3 \leq s \leq 9$), in which n is the number of CS samples.

Algorithm 2 Iterative Hard-thresholding using the UWT and the OLT

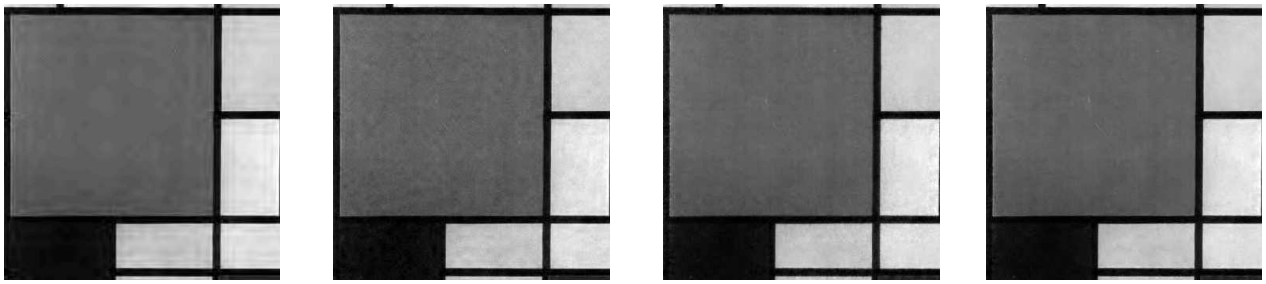
Input: Initial solution x_0: Results from Algorithm 1;
  CS Sampling vector: y;
  Maximum no. of iterations: s_{max};
Output: Reconstructed image: x_{smax};
  for $s = 0$ to $s_{max} - 1$ **do**
    $\hat{x}_s = \mathcal{H}(\Psi_{UWT}, x_s, K_s)$
    $\bar{x}_s = \mathcal{P}(\Phi, y, \hat{x}_s)$
    $\tilde{x}_s = \mathcal{H}(\Psi_{OLT}, \bar{x}_s, K_s)$
    $x_{s+1} = \mathcal{P}(\Phi, y, \tilde{x}_s)$
  end for

---

5. SIMULATION RESULTS

The proposed block-based CS sampling and reconstruction algorithms were implemented using *Matlab* on a 1.66GHz laptop computer. Fig. 3 shows the reconstructed results for a smooth 512×512 image *Mondrian* from a total of 69834 CS samples. For comparison purposes, we also include results from [4] using the multiscale CS with the StOMP and the BP reconstruction algorithms. The computation time of [4] is based on a 3G workstation, which is much more powerful than the laptop used in our simulations. As can be seen, due to the simplicity of block-based CS, our algorithms can produce fast reconstructions along with good visual qualities. In particular, compared with result of multiscale CS and StOMP [4], the proposed Alg. 1 yields the same PSNR with less computational time. The visual qualities of Fig. 3(a) and Fig. 3(c) are also similar: Algorithm 1 leads to a better reconstruction in the smooth area while the StOMP produces shaper edges. Compared with multiscale CS and the BP reconstruction, the proposed method (Algorithm 1 followed by Algorithm 2) provides a significant PSNR gain of 2dB with dramatic reduction of computation time, as testified in Fig. 3(b) and Fig. 3(d).

Table 1 tabulates the PSNR results of our algorithms on four 256×256 natural images *Lena*, *Peppers*, *Boats* and *Cameraman*. For the proposed method, the total reconstruction takes about 35 seconds to 1.5 minutes. For these images, we also presented results reported in [6], where random Fourier sampling matrices were applied in the wavelet domain of the whole image. The reconstruction algorithm



(a) (b) (c) (d)

Fig. 3. Portions of Reconstructed 512×512 image *Mondrian*. (a) Multiscale CS with StOMP reconstruction, PSNR=32.9dB, t_{StOMP} =64secs [4]; (b) Multiscale CS with BP reconstruction, PSNR=34.1dB, t_{BP}=30 hours [4]; (c) Block-based CS with Algorithm 1 only, PSNR=32.9dB, t_1=32 secs; (d) Block-based CS with both Algorithm 1 and Algorithm 2, PSNR=36.5dB, t_2=5 minutes;

Table 1. Objective coding performance (PSNR in dB)

No. of Samples n		10000	15000	20000	25000
Lenna	[6]	26.5	28.7	30.4	32.1
	Proposed	26.5	28.6	30.6	32.2
Peppers	[6]	21.6	25.3	27.5	29.4
	Proposed	27.2	30.3	32.7	34.7
Boats	[6]	26.7	29.8	31.8	33.7
	Proposed	27.0	29.9	32.5	34.8
Cameraman	[6]	26.2	28.7	30.9	33.0
	Proposed	24.0	26.1	27.9	29.4

in [6] was based on TV minimization [1], projection onto convex sets along with soft-thresholding in the wavelet domain. From this table, one can observe that for *Lena*, the PSNR results are roughly the same. For *Boats* and *Peppers*, our algorithms yield about 0.1-1.1dB and more than 5dB improvements, respectively. However, we lose about 2-3.6dB for *Cameraman*. In our opinion, such a big loss is mainly due to the fact that neither the UWT nor the OLT can fully characterize the directional information in *Cameraman*. By exploiting the recently developed directional transforms (e.g., the curvelet [17]), better results can be expected.

It should also be emphasized that our algorithms are much simpler than those in [6] both at the sender and the reconstruction sides. Moreover, the reconstruction in [6] requires additional l_1 information of wavelet coefficients at various scales (which is impractical to obtain in practical CS), while ours is completely based on the CS sampling vectors and operators.

6. CONCLUSIONS AND FUTURE WORKS

This paper has proposed a block compressed sensing framework for natural images. Due to the block-by-block processing mechanism, the sampling algorithm has very low complexities. It also offers a fast, good initial solution at the receiver side by using linear MMSE estimation. The quality of reconstructed image can be further improved using a 2-stage non-linear optimization. Despite the simplicity of our proposed algorithms, they compare favorably with existing more sophisticated algorithms.

As this paper is exploratory, there are many intriguing questions that future work should consider. First, the theory of block CS requires to be developed. Secondly, the optimization criteria of block sampling operator needs to be investigated and it would be valuable to understand the tradeoffs between block size and reconstruction performance. Thirdly, the convergence and computational complexity of the proposed algorithm needs to be analyzed; Fourthly, it is interesting to develop spatially adaptive reconstruction algorithm, where different basis or frame expansions should be used for regions with different characteristics. Finally, we hope to extend this work to CS of color images and videos.

7. REFERENCES

[1] E. Candes, J. Romberg, and T. Tao, “Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information,” *IEEE Trans. Inform. Theory*, vol. 52, pp. 489–509, Feb. 2006.

[2] D. L. Donoho, “Compressed sensing,” *IEEE Trans. Inform. Theory*, vol. 52, pp. 1289–1306, July 2006.

[3] Y. Tsaig and D. L. Donoho, “Extensions of compressed sensing,” *Signal Processing*, vol. 86, pp. 533–548, July 2006.

[4] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, “Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit,” Mar. 2006, preprint.

[5] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, “Random filters for compressive sampling and reconstrution,” in *Proc. Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, France, May 2006.

[6] E. Candes and J. Romberg, “Practical signal recovery from random projections,” 2005, preprint. [Online]. Available: http://www.dsp.ece.rice.edu/CS/

[7] D. Takhar, J. N. Laska, M. B. Wakin, M. F. Duarte, D. Baron, S. Sarvotham, K. F. Kelly, and R. G. Baraniuk, “A new compressive imaging camera architecture using optical-domain compression,” in *SPIE Proc. Computational Imaging IV*, Jan. 2006.

[8] S. Chen, D. Donoho, and M. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci Comp.*, vol. 20, Jan. 1999.

[9] J. A. Tropp, "Greed is good: Algorithmic results for sparse approximation," *IEEE Trans. Inform. Theory*, vol. 50, pp. 2231–2242, Oct. 2004.

[10] C. La and M. N. Do, "Signal reconstruction using sparse tree representations," in *SPIE Proc. Computational Imaging IV*, Jan. 2005.

[11] I. Daubechies, M. Defrise, and C. D. Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Comm. Pure Appl. Math*, vol. 57, pp. 1413–1541, 2004.

[12] S. Haykin, *Adaptive Filter Theory*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996.

[13] T. D. Tran, J. Liang, and C. Tu, "Lapped transform via time-domain pre- and post-filtering," *IEEE Trans. Signal Processing*, vol. 51, pp. 1557–1571, June 2003.

[14] C. Tu and T. D. Tran, "Context based entropy encoding of block coefficient transforms for image compression," *IEEE Trans. Image Processing*, vol. 11, pp. 1271–1283, Nov. 2002.

[15] D. Donoho and I. Johnstone, "Ideal spatial adaptation via wavelet shrinkage," *Biometrika*, vol. 81, p. 425455, 1994.

[16] I. Daubechies, "Ten lectures on wavelets," SIAM, Philadephia, CMBS conference Series, 1992.

[17] E. J. Cands and L. Demanet, "The curvelet representation of wave propagators is optimally sparse," *Comm. Pure Appl. Math*, vol. 58, pp. 1472–1528, 2004.