中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-11-005

Compressive Image Sensing: Turbo Fast Recovery with Lower-FrequencyMeasurement Sampling

Chun-Shien Lu, Hung-Wei Chen, Soo-Chang Pei

Sep. 14, 2011 || Technical Report No. TR-IIS-11-005

http://www.iis.sinica.edu.tw/page/library/TechReport/tr2011/tr11.html

Compressive Image Sensing: Turbo Fast Recovery with Lower-Frequency Measurement Sampling

Chun-Shien Lu(呂俊賢)
Institute of Information Science,
Academia Sinica, Taipei 115, Taiwan
Email: lcs@iis.sinica.edu.tw

Hung-Wei Chen(陳泓瑋)
Inst. Info. Sci., Academia Sinica and
Graduate Inst. Comm. Eng., NTU, Taiwan
Email: hungwei@iis.sinica.edu.tw

Soo-Chang Pei(貝蘇章)
Graduate Inst. Comm. Eng.,
Nat'l Taiwan Uni., Taipei, Taiwan
Email: pei@cc.ee.ntu.edu.tw

***Abstract*—In order to get better reconstruction quality from compressive sensing of images, exploitation of the dependency or correlation patterns among the transform coefficients has been popularly employed. Nevertheless, both recovery quality and recovery speed are not compromised well. In this paper, we study a new image sensing technique, called turbo fast compression image sensing, with computational complexity $O(m^2)$, where m denotes the length of a measurement vector $y = \phi x$ that is sampled from the signal x of length n via the sampling matrix ϕ with dimensionality $m \times n$. In order to leverage between reconstruction quality and recovery speed, a new and novel sampling matrix is designed. Our method has the following characteristics: (i) recovery speed is extremely fast due to a closed-form solution is derived; (ii) certain reconstruction accuracy is preserved because significant components of x can be reconstructed with higher priority via an elaborately designed ϕ. Our method is particularly different from those presented in the literature in that we focus on the design of a sampling matrix without relying on exploiting certain sparsity patterns. Simulations and comparisons with state-of-the-art CS methodologies are provided and demonstrate the feasibility of the proposed method in terms of reconstruction quality and computational complexity.**

keywords: Compressive sensing, Measurement, Reconstruction, Sampling, Sparsity

I. Introduction

We first describe the background regarding compressive sensing in Sec. I-A and then discuss related work in Sec. I-B.

A. Background

Compressive sensing (CS) has received much attention recently due to its revolutionary development in simultaneously sensing and compressing signals with certain sparsity. Moreover, the architecture of the so-called single-pixel camera [14], [27] has promoted the practicability of compressive image sensing (CIS). CS is mainly composed of two steps. Let x denote a k-sparse signal of length n to be sensed, let ϕ of dimensionality $m \times n$ represent a sampling matrix, and let y be the measurement of length m. At the encoder, a signal x is simultaneously sensed and compressed via random projection and the obtained samples are called measurements y in the context of compressive sensing. They are related via random projection as:

$$y = \phi x. \tag{1}$$

The measurement rate is defined as $0 < \frac{m}{n} < 1$ or $0 < \frac{m}{n} << 1$, which indicates the compression ratio (without quantization) without storing the original signal of length n. At the decoder, the original signal x to be sensed can be perfectly recovered by means of convex optimization or greedy algorithms if the relationship between m and k is satisfied [6].

For convex optimization-based CS algorithms, sparse signal recovery will be time-consuming and intractable if ℓ_0-minimization is adopted. ℓ_0-minimization seeks to find k nonzero entries of a signal if the signal is k-sparse in either the time/space or transform (*e.g.*, DCT, wavelet, etc) domain. The solution can become more tractable if the constraint of ℓ_0-minimization is relaxed and ℓ_1-minimization is used instead. Several algorithms relying on ℓ_1-minimization have been presented in the literature.

On the other hand, ℓ_2-norm solution is easy to calculate but the solution is not sparse, which violates the fundamental assumption of compressive sensing. Nevertheless, some approaches [19], [20], [25], [29] try to iteratively approximate ℓ_0-norm or ℓ_1-norm solution from their ℓ_2-norm counterpart. We have also studied a fast sparse signal recovery algorithm based on iteratively refining ℓ_2-norm solution [23].

In addition to convex optimization, non-convex programming (or greedy) algorithms like Orthogonal Matching Pursuit (OMP) [37] is an alternative for sparse signal recovery. Basically, OMP has been recognized as a "fast" algorithm with time complexity $O(kmn)$ with reasonable reconstruction quality in some cases. The problem here is that can we come up with a new solution to sparse signal recovery that is simultaneously faster and more accurate than OMP and other state-of-the-art compressive sensing recovery algorithms?

On the other hand, in the context of compressive sensing (CS) [11], the constraint of sparsity enables the possibility of sparse signal recovery from measurements (far) fewer than the original signal length. Moreover, the measurements generated from random projection of the original signal via a sampling matrix are equally weighted; *i.e.*, no one is more significant than the others. Thus, CS is inherently weakened in handling less-sparse signals such as highly textured images. The problem here is that can we yield weighted measurements so that less-sparse signals can be fast reconstructed while maintaining good reconstruction quality?

In this paper, we address the above problems of achieving fast and accurate CS recovery with less sparsity constraint. The idea is to investigate an elaborate design of the sampling matrix ϕ that can directly capture "important" measurements. With these important measurements, the quality original signal

can be sparsely reconstructed based on the important (corresponding to low-frequency) components in some transform domain. In our method, the qualities of reconstructed signals mimic those of JPEG compressed images.

The designed sampling matrix can be readily embedded to single-pixel cameras [14], [27] or mobile devices with camera functionality to simultaneously capture and compress signals, and the sampled measurements can be efficiently transmitted to the decoder or remote server for turbo fast recovery of the captured signals. In particular, our turbo fast CS recovery algorithm can be applied to a scenario where mobile device to mobile device (M2M) is considered. In addition, the bottleneck of distributed compressive video sensing (DCVS) [26] at decoder can now be solved if the proposed method is used.

B. Related Work

In the compressive sensing literature, efforts have been made to explore the structure or correlation inherent in the transformed coefficients to better reconstruct the signal from its corresponding measurement vector. Inspired from the concept of JPEG2000 compression, the tree-structure of wavelet transform has been popularly exploited.

In [12], [13], instead of capturing non-adaptive or universal measurements, the authors propose to gain adaptive transform coefficients from exploiting the tree-structure of Haar wavelet. In terms of image quality and recovery speed, the so-called adaptive compressive sensing framework demonstrates its superiority over the non-adaptive counterparts.

In [21], a tree-structured Bayesian compressive sensing framework is proposed, wherein the hierarchical statistical models of wavelet and DCT are adopted, and Markov chain Monte Carlo (MCMC) inference is employed. The computationally inefficient MCMC mechanism is later replaced with variational analysis in [22] to speed up recovery. Results show that their methods can achieve both accurate and fast CS recovery. The paradigm in [21], [22] belongs to probabilistic structured sparsity [4].

On the other hand, the concept of clustered sparsity has received considerable attention in compressive sensing. As summarized in [4] and Table I of [45], many existing CS algorithms [3], [8], [9], [15], [16], [24], [36] exploiting clustered sparsity need to know some pre-defined information, such as numbers, sizes, and positions of clusters, and the degree of sparsity. In [45], the proposed Bayesian compressive sensing method, a kind of nonparametric recovery algorithm, can make use of clustered sparsity without relying on prior information. Basically, the work [45] is inspired by [22] in that variational analysis is used in place of MCMC for Bayesian inference in order to guarantee convergence within finite iterations. The major difference between [22] and [45] is that the former employs a directional graphical model for tree-structure of wavelet coefficients, while the latter uses a undirectional graphical model. Furthermore, in order to target the problem of reconstructing structured-sparse signals, belief propagation is employed in [35], which resembles turbo equalization from digital communications. The clustered sparsity-based compressive sensing methods mentioned above belong to deterministic structured sparsity [4].

It should be noted that in [3], both tree structure and structured sparsity are considered and incorporated into two state-of-the-art CS algorithms, which are CoSaMP [32] and iterative hard thresholding (IHT) [5].

In addition to the aforementioned sparsity patterns, including tree structure and clustered sparsity, other models of transform coefficients, including Laplacian scale mixtures [7], piecewise autogressive model [40], Laplace prior [2], and Gaussian Mixture Models [44] are also employed within the compressive sensing framework.

The nice property of structured sparsity mentioned above has been applied to a number of image processing applications beyond reconstruction. In [42], the inverse problems of inpainting and deblurring are solved via a proposed structured sparse model selection algorithm. The key is that the sparsity of local windows partitioned from an image can be better controlled. Basically, stable inversion can be achieved because the degree of freedom in selecting models is equal to the number of bases, and is considerably lower than overcomplete dictionary methods. Further, a work, called piecewise linear estimator (PLE), extended from [42] is presented in [43].

C. Outline of this Paper

The rest of this paper is organized as follows. The use of compressive sensing for capturing natural images is called compressive image sensing (CIS) and discussed in Sec. II. In Sec. III, the idea behind our method and the proposed turbo fast CIS recovery algorithm are described. Some characteristics of our method are discussed in Sec. IV. In Sec. V, we provide extensive experiments to verify the proposed method in terms of reconstruction quality and computation speed. Finally, conclusions are given in Sec. VI.

II. Compressive Image Sensing

Inspired by the development of compressive sensing [11] and single-pixel cameras [14], [27], it is possible to sense and recover an image with as few measurements as possible if the image to be sensed is sufficiently sparse. As an illustrative example, consider the size of a sampling matrix required to sense an image of size 128×128 is as huge as 16384×16384 (suppose the use of 1D sensing as in Eq. (1)), which occupies 4×2^{28}=1G bytes for 32-bit single precision floating point, and most current desktop machines cannot afford to this storage overhead of storing sensing matrix. In addition, large sensing matrix will of course incur computation overhead during the process of random projection (Eq. (1)).

There are two solutions to this problem despite almost all compressive sensing algorithms are developed for 1D sensing. For purpose of compressive image sensing (CIS), one common strategy adopted is to divide an image into several patches/blocks with reasonable sizes and each patch is arranged in terms of 1D form so as to adapt to the existing CS algorithms. This is called block sensing [17], [18], [30],

[31]. In fact, each block signal is treated as a 1D signal for sensing and recovery.

Although block-based image sensing seems to be feasible, it still incurs the sensor calibration problem. The other is to employ 2D separate sensing [33], [34]. That is, separate sensing is conducted along the row and column directions, separately. Moreover, due to separate sensing strategy, the problem of storage overhead for storing a sampling matrix in a resource-limited sensor can be efficiently solved. As can be seen later, the storage overhead for 2D sensing of an image of size 1024×1024 is the same as that for 1D sensing of an image patch of size 32×32. Thus, it is apparent that due to the constraint of storage overhead, 1D block-based sensing is unfavorable to sense images/patches of large sizes.

In this paper, we study a new compressive image sensing algorithm via an elaborate design of sampling matrices. The paper is an extended and complete version of our prior work [10] in terms of methodological descriptions and analyses, technical comparisons with related works, and extensive experiments. The details will be described in the following sections.

III. Proposed Method: Turbo Fast Compressive Image Sensing

We first describe the idea behind our method in III-A. Then, the proposed turbo fast CS recovery algorithm based on an elaborately designed sampling matrix is presented in Sec. III-B. We investigate how it can be conducted via 1D block sensing and 2D separate sensing. The computation complexity of recovery and reconstruction quality will be studied.

A. The Idea

Although it is promising to take the concept of clustered sparsity or tree-structure of transform coefficients into consideration within the compressive sensing framework, we find two weaknesses for this paradigm of compressive sensing.

The first we notice is that the CS inversion time is still not computationally efficient. The crux is that the inference for exploiting some specific sparsity patterns is time-consuming. In view of this, we seek an alternative so that our CS recovery time can be significantly reduced while the reconstruction quality obtained from our method and state-of-the-art algorithms can be comparable.

In order not to spend time in tracing larger transform coefficients, we propose to sample only those transform coefficients that are situated at lower frequencies. This is reasonable because in image compression like JPEG the higher frequency components will be quantized with larger quantization intervals while the lower frequency components can be preserved with higher priority. Inspired by the principle of JPEG compression, our CS recovery quality will be designed to mimic JPEG compressed images. That is, we do not seek "perfect reconstruction," which is practically hard to achieve, due to natural images are usually not sparse.

The most important but unique characteristic that distinguishes our method from the existing ones is that our method can only sample m important measurements with m equal to the desired degree of sparsity k, and can turbo fast reconstruct the original signal x approximately from the sampled measurement vector y with computational complexity $O(m^2)$. The extensive experimental results indicate that our method indeed is very fast and can keep reconstruction quality up to the degree of JPEG compression approximately.

The second concern is that compressive sensing conventionally relies on the assumption of sparsity to reconstruct the original signal from (far) fewer measurements. As studied in [39], CS is only suitable for so-called sparse signals while principal component analysis (PCA) is more suitable to deal with non-sparse or noisy signals.

However, many natural signals inherently containing textured components are a kind of non-sparse signals. The assumption of sparsity and the exploitation of structured sparsity do not conform to the property of less-sparse signals. In view of this, another goal of our method is to target this problem. Our strategy is intuitive and empirical observations [41] again suggest that it is better to preserve important measurements sampled from lower frequency components in some transform domains.

Our method is particularly different from those reviewed in Sec. I-B in that we focus on the design of a sampling matrix while others concern to exploit the dependency or correlation patterns among the transform coefficients.

B. Turbo Fast Compressive Sensing Recovery via Sampling Matrix Design

We start from the random projection, $y = \phi x$, and observe that if important information of x can be sampled and stored in y, then it is possible to approximately reconstruct x with fewer important measurements in a fast way. The goal is feasible by designing a new and novel sampling matrix. Our method has been readily incorporated with single-pixel cameras for compressive image sensing.

In this subsection, we describe the proposed 1D sensing of an image patch/block and 2D separate sensing of a whole image, respectively.

1) 1D Block Sensing: For 1D sensing of an image patch, we introduce a 1D linear operator T and impose it to random projection to obtain:

$$Ty = T(\phi x), \tag{2}$$

where x is regarded as a small image or an image patch of reasonable size. Eq. (2) is further derived based on the principle of linear operations [28] as:

$$Ty = T(\phi x) = (T^2\phi)(Tx), \tag{3}$$

where T^2 denotes 2D linear operator. Please refer to Appendix A for details.

Eq. (3) reveals that the positions at lower frequencies of transformed vector Tx indicate important transformed coefficients and Ty indicates important measurements since they are linear combinations of significant transformed coefficients.

In order to sample "important" transformed coefficients from Tx and speed up recovery, we design a new sampling

matrix, $(T^2\phi)^z$, by setting the last $n-m$ columns of $T^2\phi$ to be zeros. This implies that the non-zero columns of $(T^2\phi)^z$ form a full-rank matrix with rank m. Once $(T^2\phi)^z$ is built in the transform domain, it is inversely transformed back to the time/space domain and an elaborately designed sampling matrix can be expressed as:

$$\Phi = (T^2)^{-1}(T^2\phi)^z, \tag{4}$$

where our designed sampling matrix Φ involves a random matrix ϕ and 2D linear operator T^2.

Now, Φ is stored in the sensors for the purpose of compressive sensing. According to Eq. (3) and Eq. (4), we have the following derivations:

$$y = \Phi x \;\Rightarrow\; Ty = (T^2\Phi)(Tx) = (T^2\phi)^z(Tx). \tag{5}$$

Recall that the last $n-m$ columns of $(T^2\phi)^z$ are set to zeros. This means that we only sample the lower-frequency components in Tx by truncating the remaining higher-frequency components.

In order to speed up sparse signal recovery, let Φ^s denote the submatrix of dimensionality $m \times m$ by discarding the zero columns of $(T^2\phi)^z$, and let $(Tx)^s$ denote the $m \times 1$ vector by discarding the last $n-m$ transformed coefficients. Therefore, we can derive from Eq. (5) to obtain:

$$\begin{aligned} Ty &= \Phi^s(Tx)^s \Rightarrow \\ (\Phi^s)^{-1}Ty &= (\Phi^s)^{-1}\Phi^s(Tx)^s = (Tx)^s. \end{aligned} \tag{6}$$

It is now evident that the signal x can be approximately and fast recovered via the following signal sensing and signal recovery processes. At the encoder for purpose of sensing, the measurement y is available via random projection in Eq. (5). At the decoder for purpose of recovery, y is first processed via Eq. (6), and then $(\Phi^s)^{-1}Ty$ in Eq. (6) is padded with $n-m$ zero values (to obtain Tx) and inversely transformed via T^{-1}.

We have to clarify that $(\Phi^s)^{-1}T$ can be calculated in advance and is fixed for use at the decoder. Therefore, the computational complexity of our CS recovery algorithm merely comes from processing y via $(\Phi^s)^{-1}Ty$.

Basically, it is observed that our method is proposed to execute conventional DCT or wavelet transform within the framework of compressive sensing if T is selected to be DCT or wavelet transform. But it is not suitable to just compute transformation (via T) of a signal to be sensed and then stored a proportion of low-frequency components (by discarding high-frequency components as done in this paper). This is because some parts to be deleted are still sampled in advance, which follows the same mechanism of conventional sampling. On the contrary, the ultimate goal of our method can, in fact, achieve the same effect of sampling+compression within the framework of compressive sensing, as explained in the previous paragraphs.

2) 2D Separate Sensing: Here, we investigate how to directly sense a whole 2D image via separate sensing in order to alleviate storage overhead of storing a sampling matrix. According to Eq. (11) of Appendix A, 2D transform is conducted on the 2D image, which is no longer divided into patches and is no longer arranged in terms of 1D form. Therefore, we can get 2D sensing of an image $x_{n\times n}$ via the sampling matrix $\phi_{m\times n}$ as:

$$y_{m\times m} = \phi_{m\times n} x_{n\times n} \phi^t_{n\times m}, \tag{7}$$

where $\phi^t_{n\times m}$ denotes the transpose of $\phi_{m\times n}$. Eq. (7) also reveals the measurement rate of $\frac{m^2}{n^2}$ in two dimensional sensing.

Similar to Eq. (11), let $S_{m\times m}$ and $S_{n\times n}$ stand for two 1D linear operators with respective transpose represented as $S^t_{m\times m}$ and $S^t_{n\times n}$. We have $S_{n\times n}S^t_{n\times n} = I_{n\times n} = S^t_{n\times n}S_{n\times n}$, where $I_{n\times n}$ denotes an identity matrix of dimensionality $n\times n$. Then, we can further derive by imposing 2D transform on the measurement matrix $y_{m\times m}$ of Eq. (7) to get:

$$\begin{aligned} T^2(y) &= S_{m\times m} y_{m\times m} S^t_{m\times m} \\ &= S_{m\times m}\phi_{m\times n} x_{n\times n}\phi^t_{n\times m} S^t_{m\times m} \\ &= S_{m\times m}\phi_{m\times n}(S^t_{n\times n}S_{n\times n})x_{n\times n} \\ &\quad (S^t_{n\times n}S_{n\times n})\phi^t_{n\times m}S^t_{m\times m} \\ &= (S_{m\times m}\phi_{m\times n}S^t_{n\times n})(S_{n\times n}x_{n\times n}S^t_{n\times n}) \\ &\quad (S_{n\times n}\phi^t_{n\times m}S^t_{m\times m}) \\ &= T^2(\phi)T^2(x)T^2(\phi^t). \end{aligned} \tag{8}$$

Then, the sampling matrix can be derived in similar to Eq. (4) for 2D separate sensing. Another merit of 2D sensing is that it enables compressive sensing of large images without resorting to block-based sensing.

IV. Analysis

In this section, we will discuss the following issues, including A) difference between our method and frequency/space-frequency transforms; B) computation complexity of CS recovery; C) mutual incoherence between the sampling matrix and dictionary, D) sparsity vs. reconstruction quality, and (E) approximate recovery vs. perfect reconstruction, for the proposed compressive image sensing methods.

A. Difference between Our Method and Frequency/Space-Frequency Transforms

It is worth noting that the principle of designing Φ in Eq. (4) still starts with a random sampling matrix ϕ conventionally adopted in CS, which is selected as the foundation for our design. The characteristic unique to our method is that a 2D linear operator T^2, as described in Appendix A, is applied to ϕ, followed by setting the last $n-m$ columns to be zeros. In addition, our CIS methods are built within the framework of CS while other conventional frequency or frequency-space transforms are not.

B. Computation Complexity of CS Recovery

The principle of our method is to preserve the top k-lowest frequency components of Tx. We have $m = k$ and only two linear transforms, $(\Phi^s)^{-1}T$ and T^{-1}, as described in the last paragraph of Sec. III-B, are required for approximate signal

recovery. Thus, the computation complexity of recovery is in the order of $O(m^2)$. As we shall see later in Sec. V, our method is the fastest CS recovery algorithm among the methods used for comparisons.

C. Mutual Incoherence

In the literature, a sampling matrix is usually a Gaussian random matrix or a Bernoulli random matrix taking value ± 1 with equal probability. A good sampling matrix, which preserves incoherence [6] with a dictionary or transform basis, is desired for efficient recovery. Mutual incoherence leads to perfect recover with a higher probability under the condition that $m \geq ck \cdot O(log(n/k))$ holds.

Our results show that random matrix conventionally adopted in the compressive sensing literature is more incoherent than the sampling matrix proposed in this paper with either K-SVD [1] or DCT. We have to, however, point out that since the sparsity assumption is rarely satsified with respect to natural images, which are usually less sparse, our method theoretically sacrifices mutual incoherence between the sampling matrix and the corresponding dictionary but is practically feasible in reconstructing signals with higher quality and speed.

D. Sparsity vs. Reconstruction Quality

Following the above descriptions, the impact of sparsity on reconstruction quality is discussed here. According to our empirical observations, almost all compressive sensing algorithms proposed so far fail to reconstruct a natural image with quality (in terms of PSNR or SSIM [38]) superior to simple interpolation. For example, consider a natural image, Peppers, of size 50×50. When a bicubic interpolation technique is adopted to generate an enlarged image of size 100×100, the obtained PSNR is 26.76dB (between the original Peppers and the resultant interpolated Peppers). However, within the framework of compressive sensing, if the original image has the size of 100×100 and the number of corresponding measurements is 50×50, the obtained image obtained from the measurements via OMP has PSNR 17.06dB, which is far lower than the one obtained using bicubic interpolation. Similar results can also be observed from many other CS recovery algorithms.

In view of these results, we propose a new CS recovery algorithm that does not follow the convention of CS. More specifically, our method is an alternative of CS in the sense that we seek to pursue approximate recovery instead of theoretic perfect reconstruction that is the ultimate goal of CS. Our observations and results are strongly supported by the fact that natural images are not highly or sufficiently sparse, which somewhat violates the fundamental assumption of CS. However, we also have to clarify that for those applications exhibiting sufficient sparsity the random sampling matrix inherently used in compressive sensing may be better than the one presented in this paper.

E. Approximate Recovery vs. Perfect Recpnstruction

It is known that under the constraint of $m \geq ck \cdot O(log\,\frac{n}{k})$ conventional compressive sensing methods can achieve perfect reconstruction if the available number of measurements and the sparsity of the signal to be reconstructed satisfy the above relation. However, perfect reconstruction is usually not practical because for many signals with large k, sometimes only small M is available, leading to violation of $m \geq ck \cdot O(log\,\frac{n}{k})$.

In this paper, we aim to explore a more practical solution to CS recovery. More specifically, based on the elaborate design of the sampling matrix, we do not seek perfect reconstruction but approximate reconstruction. Depending on the selected m, which is assumed to be k, turbo fast and approximate reconstruction can be achieved. Approximate reconstruction has been readily useful for many CS-based image/video applications [42], [43].

In this paper, as described in Appendix A, a Haar matrix is exploited as T to design the sampling matrix Φ such that the original signal x can be approximately reconstructed from as many measurements as the number of transform coefficients sampled via Eq. (5) and Eq. (6).

V. Experimental Results

Several experiments were conducted to verify the performance of the proposed turbo fast CIS methods in terms of reconstruction quality and speed.

State-of-the-art CS algorithms [11], including orthogonal matching pursuit (OMP) [37], Lasso, TS-BCS-MCMC [21], TS-BCS-VB [22], and model-based CS (MCS) [3][1], were chosen for comparisons under different measurement rates (MRs). There are two signal models, wavelet trees and block sparsity, used in MCS. In our experiments conducted here, 2D wavelet trees and block sparsity in [3] were adopted. The default settings of all source codes were employed in our experiments to better guarantee the good performances of the aforementioned methods for fair comparisons.

All experiments were conducted in Matlab 7.11 (R2010b) with Intel CPU Core i7 930 (2.80 GHz) and 6 GB RAM under OS Windows 7 Enterprise edition 64-bit. For simulations of image sensing, several images with different sizes and sparsities, including Baboon, Barbara, Cameraman, Flintstones, Lena, and Peppers, were adopted.

For 1D sensing, all the methods used for comparisons were conducted in a block-wise manner. That is, image sensing and recovery were executed for each 32×32 block [30]. In our method, as derived in Appendix A, two Haar matrices, $H_{1024\times 1024}$ and $H_{(1024\times MR)\times(1024\times MR)}$ were employed. However, if RAM is larger, then the block size can be permitted to be larger.

For 2D sensing in our method, we currently only employ 2D DCT because the design of 2D wavelet filters in terms of matrix forms need to be further studied.

A. Reconstruction Quality

The recovery quality is measured in terms of PSNR (in dB) and structural similarity (SSIM, $0 \leq \text{SSIM} \leq 1$) indexing [38], respectively. The bigger, the better. Table I and Table II show

[1]Our experiments conducted on MCS show that block sparsity is better than 2D tree in both reconstruction quality and speed.

the comparisons of reconstruction qualities under different measurement rates (MRs) for images of different sparsities, including a less-sparse image, Barbara, of size 512×512 and a sparse image, Cameraman, of size 256×256.

It is surprising to find from Table I and Table II that our method significantly outperforms or is comparable to all the algorithms used for comparisons no matter whether either structured sparsity or tree structure is taken into consideration or not. DCT- and Haar wavelet-based 1D sensing in our method exhibit comparable reconstruction results. Fig. 1 shows some of the reconstructed Barbara images using measurement rate 12.5%. Nevertheless, our method spends considerably less time than all the methods used for comparisons, as will be shown in the next subsection.

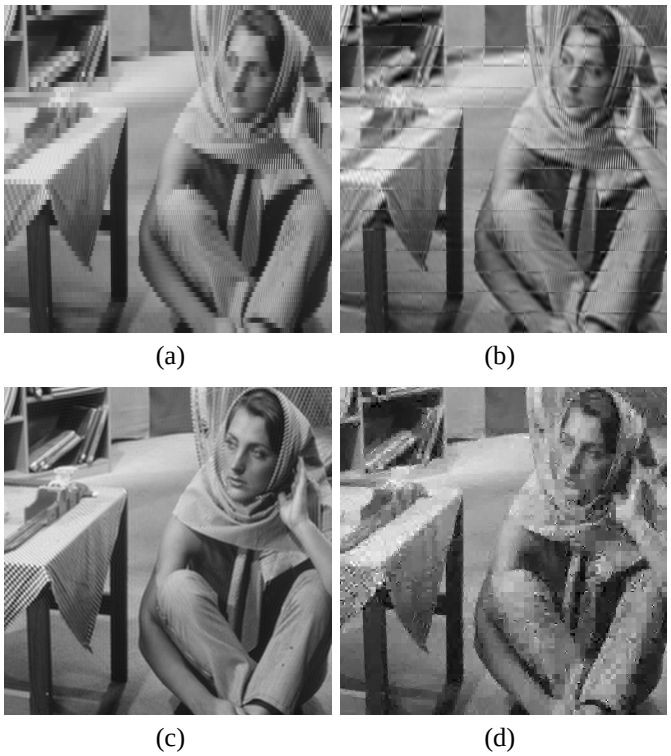


Fig. 1. CS recovered Barbara images with MR 12.5%: (a) our result with Haar wavelet-based 1D sensing (PSNR: 22.98dB; SSIM: 0.65); (b) our result with DCT-based 1D sensing (PSNR: 22.60dB; SSIM: 0.61); (c) our result with DCT-based 2D separate sensing (PSNR: 24.54dB; SSIM: 0.73); and (d) He *et al.*'s result [22] (PSNR: 21.37dB; SSIM: 0.54).

It is also interesting to find that for most cases the reconstruction quality of the Barbara image is better than that of the Cameraman image in terms of PSNR, which violates the intuition that Cameraman is sparser than Barbara in the transform (either DCT or wavelet) domain. However, when SSIM is adopted as the image quality evaluation metric, the obtained results indeed indicate that Cameraman is better reconstructed than Barbara for our method and those CS methods exploiting sparsity patterns.

B. Reconstruction Speed

The reconstruction speed is measured in terms of execution time (abbreviated as Exe.) and CPU time. We only provide in Table III the comparison of reconstruction speed under different measurement rates (MRs) for the Barbara image. Nevertheless, similar results can also be observed for other images with different sparsity levels. As the results indicate, our method finds its usefulness in real-time image sensing and recovery due to its extremely fast CS recovery. For the CS algorithms used for comparison, only the model-based CS with (CoSaMP+block sparsity) [3] is as fast as ours. It can be, however, found from Table I and Table II that our method obtains far better reconstruction quality than [3], in particular, for measurement rates smaller than 12.5%.

C. Fast Compressive Sensing and Recovery of Large Images

Fast compressive sensing and recovery of large-scale images is still challenging for the existing CS algorithms that employ 1D sensing strategy. The feasibility of our 2D separate sensing algorithm is demonstrated using a Shepp-Logan image of size 2048×2048. The reconstruction qualities obtained using our 1D sensing and 2D separate sensing strategies are shown in Fig. 2 under the measurement rate 1.5625%. The execution time for all strategies is between 2 and 3 seconds for measurement rate ranging from 1.5625% to 50%.

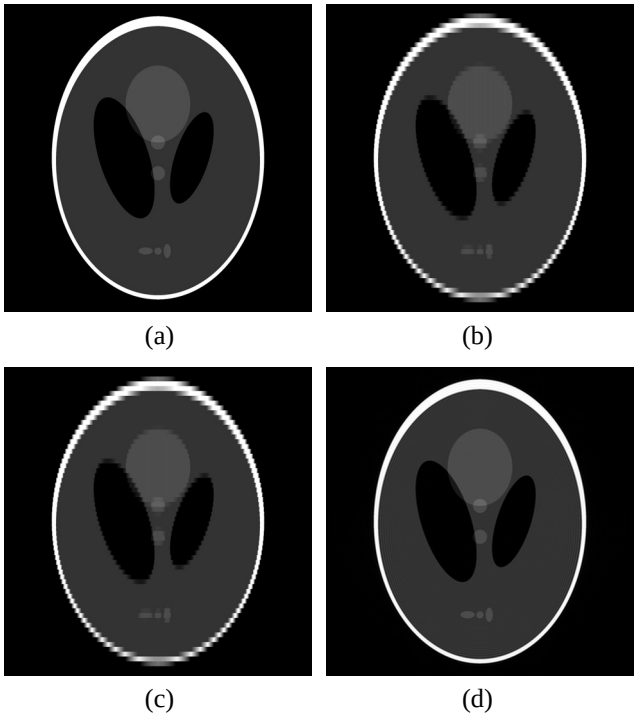


Fig. 2. CS recovered Shepp-Logan images with MR 1.5625%: (a) original image of size 2048×2048; (b) our result with Haar wavelet-based 1D sensing (PSNR: 22.18dB; SSIM: 0.95); (c) our result with DCT-based 1D sensing (PSNR: 22.22dB; SSIM: 0.94); and (d) our result with DCT-based 2D separate sensing (PSNR: 28.69dB; SSIM: 0.92).

D. Remarks

It should be noted that both the model-based CS method [3] and tree structure-based CS method [21], [22] need to perform wavelet decomposition prior to sensing. It is, however,

TABLE I
RECOVERY QUALITY COMPARISON OF CS ALGORITHMS UNDER DIFFERENT MEASUREMENT RATES (MRS) FOR CAMERAMAN IMAGE.

Methods	Metrics	MR (1.56%)	MR (3.13%)	MR (6.25%)	MR (12.5%)	MR (25.0%)	MR (50.0%)
OMP	PSNR(dB)	7.41	14.67	15.86	17.47	20.06	23.68
(Sparsify toolbox)	SSIM	0.04	0.32	0.35	0.41	0.53	0.69
Lasso	PSNR(dB)	13.82	15.39	17.24	19.04	21.76	25.96
(Sparsify Lab)	SSIM	0.33	0.38	0.43	0.49	0.62	0.77
Model-based CS	PSNR(dB)	6.38	7.42	9.18	14.92	21.99	23.05
(CoSaMP+block sparsity) [3]	SSIM	0.06	0.07	0.08	0.26	0.64	0.70
TS-BCS-VB	PSNR(dB)	17.11	17.98	19.10	20.20	22.62	26.78
[22]	SSIM	0.46	0.50	0.56	0.63	0.73	0.85
TS-BCS-MCMC	PSNR(dB)	16.87	17.90	19.33	20.59	23.11	27.39
(wavelet tree) [21]	SSIM	0.46	0.51	0.55	0.64	0.74	0.85
TS-BCS-MCMC	PSNR(dB)	7.68	11.64	18.10	21.08	23.87	28.15
(DCT tree) [21]	SSIM	0.08	0.26	0.48	0.60	0.75	0.87
Our Method (1D sensing:	PSNR(dB)	18.32	18.39	19.84	22.15	24.82	29.06
Haar wavelet-based)	SSIM	0.57	0.58	0.64	0.73	0.82	0.92
Our Method (1D sensing:	PSNR(dB)	18.42	18.46	19.38	22.36	25.60	30.22
DCT-based)	SSIM	0.58	0.58	0.61	0.69	0.79	0.90
Our Method (2D sensing:	PSNR(dB)	20.74	21.84	23.26	24.94	27.46	31.42
DCT-based)	SSIM	0.58	0.63	0.70	0.78	0.87	0.94

TABLE II
RECOVERY QUALITY COMPARISON OF CS ALGORITHMS UNDER DIFFERENT MEASUREMENT RATES (MRS) FOR BARBARA IMAGE.

Methods	Metrics	MR (1.56%)	MR (3.13%)	MR (6.25%)	MR (12.5%)	MR (25.0%)	MR (50.0%)
OMP	PSNR(dB)	15.84	16.55	18.00	20.26	23.12	27.76
(Sparsify toolbox)	SSIM	0.19	0.22	0.31	0.47	0.64	0.82
Lasso	PSNR(dB)	14.76	16.70	19.15	21.70	24.89	29.99
(Sparsify Lab)	SSIM	0.19	0.26	0.38	0.54	0.72	0.88
Model-based CS	PSNR(dB)	7.13	7.82	9.81	16.32	23.82	25.03
(CoSaMP+block sparsity) [3]	SSIM	0.04	0.05	0.06	0.31	0.69	0.74
TS-BCS-VB	PSNR(dB)	18.46	19.30	20.31	21.37	22.35	23.85
[22]	SSIM	0.37	0.41	0.46	0.54	0.62	0.74
TS-BCS-MCMC	PSNR(dB)	18.19	19.48	20.85	22.04	23.31	25.19
(wavelet tree) [21]	SSIM	0.36	0.42	0.48	0.55	0.65	0.77
TS-BCS-MCMC	PSNR(dB)	8.81	11.41	18.07	22.49	24.57	29.07
(DCT tree) [21]	SSIM	0.06	0.20	0.43	0.59	0.73	0.89
Our Method (1D sensing:	PSNR(dB)	19.25	19.32	21.18	22.98	25.49	30.24
Haar wavelet-based)	SSIM	0.48	0.50	0.56	0.65	0.79	0.93
Our Method (1D sensing:	PSNR(dB)	19.31	19.38	20.27	22.60	25.41	33.03
DCT-based)	SSIM	0.48	0.49	0.52	0.61	0.77	0.93
Our Method (2D sensing:	PSNR(dB)	22.47	23.24	23.92	24.54	25.70	30.88
DCT-based)	SSIM	0.56	0.61	0.67	0.73	0.82	0.93

impractical because before sensing we have no image data that can be used for wavelet transform. On the contrary, our method can directly sample signals in the time/space domain without needing such a pre-processing step.

Another characteristic of our method is that the original signal x can be approximately reconstructed from as many measurements as the number of transform coefficients sampled via Eq. (5) and Eq. (6).

VI. CONCLUSIONS AND FUTURE WORK

Fast and accurate compressive sensing recovery is still a challenging issue, and has received considerable attention in the literature. In this paper, we do not follow the tradition of imposing certain sparsity patterns on a CS recovery algorithm. On the contrary, we propose to design a new and novel sampling matrix for the purpose of preserving important measurements. Under this circumstance, turbo fast and accurate CIS recovery with closed-form solution of computation complexity $O(m^2)$ can be achieved.

We have also studied 1D sensing and 2D separate sensing strategies for 1D signals and 2D images, respectively. Compared to 1D sensing, 2D separate sensing is found to be particularly feasible in compressive sensing of large-scale images in terms of storage and computation overhead reduction and reconstruction quality improvement.

The issues, including the impact of noisy measurements on our methods and the exploration of CS characteristics for our method, deserve further studying.

ACKNOWLEDGMENT

This work was supported by National Science Council, Taiwan, under grants NSC 97-2628-E-001-011-MY3 and NSC

TABLE III
RECOVERY SPEED (IN SECONDS) COMPARISON OF CS ALGORITHMS UNDER DIFFERENT MEASUREMENT RATES (MRS) FOR BARBARA IMAGE.

Methods	time	MR (1.56%)	MR (3.13%)	MR (6.25%)	MR (12.5%)	MR (25.0%)	MR (50.0%)
OMP	Exe.	0.32	0.42	0.70	1.36	3.56	11.91
(Sparsify toolbox)	CPU	1.25	1.68	2.81	5.43	14.20	47.67
Lasso	Exe.	1.88	3.74	8.97	24.22	85.18	702.04
(Sparsify Lab)	CPU	7.60	14.94	35.83	96.86	339.83	2787.05
Model-based CS [3]	Exe.	1.13	1.25	1.49	2.60	5.75	4.86
(CoSaMP+block sparsity)	CPU	4.49	4.99	5.91	10.42	22.84	186.39
TS-BCS-VB [22]	Exe.	219.94	236.28	244.47	247.23	267.77	298.33
	CPU	220.66	907.33	937.05	949.77	1022.48	1135.52
TS-BCS-MCMC [21]	Exe.	2243.21	2450.64	2492.97	2580.08	2742.51	3088.34
(wavelet tree)	CPU	2241.52	9263.53	9411.93	9213.51	10279.05	11469.12
TS-BCS-MCMC [21]	Exe.	2585.22	2748.26	2842.79	2848.40	2978.40	3233.30
(DCT tree)	CPU	2579.59	10305.24	10650.77	10670.41	11109.13	11962.47
Our Method	Exe.	0.02	0.02	0.02	0.02	0.03	0.03
(1D sensing: Haar wavelet-based)	CPU	0.06	0.06	0.06	0.06	0.12	0.12
Our Method	Exe.	0.02	0.02	0.02	0.02	0.02	0.04
(1D sensing: DCT-based)	CPU	0.06	0.06	0.05	0.06	0.06	0.19
Our Method	Exe.	0.06	0.06	0.06	0.06	0.07	0.07
(2D sensing: DCT-based)	CPU	0.25	0.16	0.27	0.25	0.31	0.39

98-2221-E-001-004-MY3.

APPENDIX A
LINEAR TRANSFORM OF RANDOM PROJECTION

The linear operator introduced in [28] is employed to derive the linear transform relationship among x, y, and ϕ. Let S be a linear orthogonal transform operator and let x denote a 1D signal with dimensionality $m \times 1$. The vector X of transform coefficients corresponding to x is represented as: X=Sx, where S is a matrix with dimensionality $m \times m$. The original signal x can be reconstructed as:

$$\hat{x} = S^t X = S^t S x, \tag{9}$$

where S^t is the transpose of S, and $S^t S = S S^t = I$. Similarly, if 2D case, *i.e.*, x of size $m \times m$, is considered, then we have:

$$\begin{aligned} X &= SxS^t, \\ \hat{x} &= S^t X S = S^t S x S^t S. \end{aligned} \tag{10}$$

According to the above characteristics, the linear transform relationship among x, y, and ϕ can be, respectively, derived under the cases of DCT and discrete Haar wavelet transform (DHWT). For the DCT case, let S_m and S_n, respectively, denote $m \times m$ and $n \times n$ DCT matrices. Also let $T[\cdot]$ and $T^2[\cdot]$, respectively, denote the 1D-DCT and 2D-DCT operations. Let's start from $y = \phi x$, and we can derive:

$$\begin{aligned} &S_m y = S_m \phi x = (S_m \phi S_n^t)(S_n x) \Rightarrow \\ &T[y] = T[\phi x] = T^2[\phi] T[x]. \end{aligned} \tag{11}$$

Thus, Eq. (11) explains the rationality of Eq. (3).

For the wavelet case, the simple structure inherent in the Haar wavelet is adopted. Let H denote a 2D Haar wavelet transform. Then, the Haar transform of x can be derived, similar to Eq. (10), as:

$$X = HxH^t. \tag{12}$$

However, it is not straightforward to use the conventional Haar wavelet to achieve Eq. (12) because if, for example, the 8×8 Haar wavelet, as shown in Eq. (13), is used, only one wavelet decomposition is allowed.

$$H = \frac{\sqrt{2}}{2} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}. \tag{13}$$

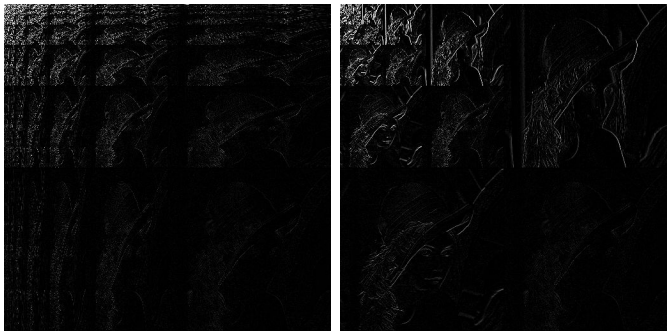
To handle this problem, another type of Haar transform suitable for Eq. (12) is designed as:

$$H = \begin{bmatrix} c^3 & c^3 & c^3 & c^3 & c^3 & c^3 & c^3 & c^3 \\ c^3 & c^3 & c^3 & c^3 & -c^3 & -c^3 & -c^3 & -c^3 \\ c^2 & c^2 & -c^2 & -c^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c^2 & c^2 & -c^2 & -c^2 \\ c & -c & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & c & -c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c & -c & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c & -c \end{bmatrix}, \tag{14}$$

where $c = \frac{\sqrt{2}}{2}$.

Different from conventional wavelet transforms, it is worth noting that the use of the Haar matrix shown in Eq. (14) allows multi-scale wavelet decomposition finished within one matrix operation. Fig. 3(a) and (b), respectively, illustrate the results of Haar wavelet decomposition using the designed Haar matrix (via Eq. (12) and Eq. (14)) and conventional Haar filter. It can be observed that the Haar matrix will continue to decompose the LH-band and HL-band at each scale while the conventional Haar wavelet will not. Furthermore, we want to clarify that Fig. 3 is merely used to illustrate the difference between the Haar matrix and conventional Haar wavelet. In

fact, the Haar matrix is used in our method to decompose the sampling matrix, as indicated in Eq. (3).



(a) (b)

Fig. 3. Comparison of Haar Transforms: (a) Haar matrix decomposition; (b) conventional Haar decomposition.

REFERENCES

[1] M. Aharon, M. Elad, and A. M. Bruckstein, "The K-SVD: an algorithm for designing of overcomplete dictionaries for sparse representation," *IEEE Trans. on Signal Processing*, 2006.

[2] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Bayesian compressive sensing using Laplacr prior," *IEEE Trans. on Image Processing*, vol. 19, no. 1, pp. 53-63, 2010.

[3] R. Baraniuk, V. Cevher, M. F. Duarte, and C. Hedge, "Model-based Compressive Sensing," *IEEE Trans. on Information Theory*, vol. 56, no. 4, pp. 1982-2001, 2010.

[4] R. Baraniuk, V. Cevher, and M. B. Wakin, "Low-Dimensional Models for Dimensionality Reduction and signal Recovery: A Geometric Perspective," *Proceedings of The IEEE*, vol. 98, no. 6, pp. 959-971, 2010.

[5] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *preprint*, 2008.

[6] E. Candes and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Problems*, vol. 23, no. 3, pp. 969-985, 2007.

[7] P. J. Carrgues, "Sparse coding models of natural images: algorithms for efficient inference and learning of higher-order structure," *Ph. D. dissertation*, 2009.

[8] V. Cevher, C. Hedge, M. F. Duarte, and R. G. Baraniuk, "Sparse signal recovery using Markov random fields," *Proc. Workshop on Neural Info. Proc. Sys. (NIPS)*, 2008.

[9] V. Cevher, P. Indyk, C. Hedge, and R. G. Baraniuk, "Recovery of clustered sparse signals from compressive measurements," *Proc. Sampling Theory Appl. (SAMPTA)*, 2009.

[10] H. W. Chen, C. S. Lu, and S. C. Pei, "Fast Compressive Sensing Recovery with Transform-based Sampling," *Proc. Workshop on Signal Processing with Adaptive Sparse Structured Representations*, pp. 110, June 27-30, Edinburgh, UK, 2011.

[11] Compressive Sensing Resources: `http://dsp.rice.edu/cs`

[12] S. Dekel, "Adaptive compressed image sensing based on wavelet-trees," *report*, 2008.

[13] S. Deutsch, A. Averbuch, and S. Dekel, "Adaptive compressed image sensing based on wavelet modeling and direct sampling," *Proc. SAMPTA*, 2009.

[14] M. F. Duarte, M. A. Davenport, D. Takhar, J. N. Laska, T. Sun, K. F. Kelly, and R. G. Baraniuk, "Single-pixel imaging via compressive sampling," *IEEE Signal Processing Mag.*, vol. 25, pp. 83-91, 2008.

[15] Y. C. Eldar and M. Mishali, "Robust recovery of signals from a structured union of subspaces," *IEEE Trans. on Information Theory*, vol. 55, no. 11, pp. 5302-5316, 2009.

[16] Y. C. Eldar, P. Kuppinger, and H. Bolcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. on Signal Processing*, vol. 58, no. 6, pp. 3042-3054, 2010.

[17] L. Gan, "Block compressed sensing of natural images," *Conf. on Digital Signal Processing (DSP)*, Cardiff, UK, July 2007.

[18] L. Gan, T. T. Do, and T. D. Tran, "Fast Compressive Imaging Using Scrambled Block Hadamard Ensemble," *Proc. 16th European Signal Processing Conference (EUSIPCO)*, Lausanne, Switzerland, August 2008.

[19] I.F. Gorodnitsky, and B.D. Rao, "Sparse Signal Reconstruction from Limited Data Using FOCUSS: A Re-weighted Minimum Norm Algorithm," *IEEE Trans. on Signal Processing*, vol. 45, no. 3, pp. 600-616, 1997.

[20] Z. Harmany, D. Thompson, R. Willett, and R.F. Marcia, "Gradient Projection for Linearly Constrained Convex Optimization in Sparse Signal Recovery," *Proc. IEEE ICIP*, pp. 3361-3364, 2010.

[21] L. He and L. Carin, "Exploiting structure in wavelet-based bayesian compressed sensing," *IEEE Transactions on Signal Processing*, vol. 57, no. 9, pp. 3488-3497, 2009.

[22] L. He, H. Chen, and L. Carin, "Tree-Structured Compressive Sensing with Variational Bayesian Analysis," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 233-236, 2010.

[23] S. H. Hsieh and C. S. Lu, "Fast Sparse Signal Recovery via Iteratively Refining L2-Norm Solution," submitted.

[24] J. Huang, T. Zhang, and D. Metaxas, "Learning with structured sparsity," *Proc. Int. Conf. on Machine Learning (ICML)*, 2009.

[25] M. Hyder, and K. Mahata, "An Approximate L0 Norm Minimization Algorithm for Compressed Sensing," *Proc. IEEE ICASSP*, pp. 3365-3368, 2009.

[26] L. W. Kang and C. S. Lu, "Distributed Compressive Video Sensing," *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, pp. 1169-1172, Taipei, Taiwan, 2009.

[27] F. Magalhaes, F. M. Araujo, M. V. Correia, M. Abolbashari, and F. Farahi, "Active Illumination Single-pixel Camera Based on Compressive Sensing," *Journals of Applied Optics*, vol. 50, no. 4, pp.405-414, 2011.

[28] N. Merhav and V. Bhaskaran, "A transform domain approach to spatial domain image," *HPL-94-116*, Technion City, Haifa 32000, Israel, 1994.

[29] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A Fast Approach for Overcomplete Sparse Decomposition Based on Smoothed L0 Norm," *IEEE Trans. on Signal Processing*, vol. 57, no. 1, pp. 289-301, 2009.

[30] S. Mun and J. E. Fowler, "Block compressed sensing of images using directional transforms," *Proc. IEEE Int. Conf. Image Processing*, pp. 3021-3024, 2009.

[31] S. Mun and J. E. Fowler, "Residual reconstruction for block-based compressed sensing of video," *Proc. Data Compression Conference (DCC)*, pp. 183-192, 2011.

[32] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301-321, 2009.

[33] Y. Rivenson and A. Stern, "Compressed Imaging with a Separable Sensing Operator," *IEEE Signal Processing Letters*, vol. 16, no. 6, pp. 449-452, 2009.

[34] Y. Rivenson and A. Stern, "Practical compressive sensing of large images," *16th Int'l Conf. on Digital Signal Processing*, July 2009.

[35] P. Schniter, "Turbo reconstruction of structured sparse signals," *Proc. 44th Annual Conf. on Information Sciences and Systems (CISS)*, 2010.

[36] M. Stojnic, F. Parvaresh, and B. Hassibi, "On the reconstruction of block-sparse signals with an optimal number of measurements," *IEEE Trans. on Signal Processing*, vol. 57, no. 8, pp. 3075-3085, 2009.

[37] J. A. Tropp and A. C. Gilbert, "Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit," *IEEE Trans. on Information Theory*, vol. 53, no. 12, pp. 4655-4666, 2007.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. on Image Processing*, vol. 13, no. 4, pp. 600-612, 2004.

[39] Y. Weiss, H.-S. Chang, and W. T. Freeman, "Learning compressed sensing," *Proc. Allerton*, 2007.

[40] X. Wu, X. Zhang, and J. Wang, "Model-guided adaptive recovery of compressive sensing," *Proc. Data Compression Conf.*, pp. 123-132, 2009.

[41] E. Y. Yam and J. W. Goodman, "A Methematical Analysis of the DCT Coefficient Distributions for Images," *IEEE Trans. on Image Processing*, vol. 9, no. 10, pp. 1661-1666, 2000.

[42] G. Yu, G. Sapiro, and S. Mallat, "Image Modeling and Enhancement via Structured Sparse Model Selection," *Proc. IEEE International Conference on Image Processing (ICIP)*, Hong Kong, 2010.

[43] G. Yu, G. Sapiro, and S. Mallat, "Solving Inverse Problems with Piecewise Linear Estimators: From Gaussian Mixture Models to Structured Sparsity," submitted, arxiv.org/abs/1006.3056, 2010.

[44] G. Yu and G. Sapiro, "Statistical Compressive Sensing of Gaussian Mixture Models," submitted, arxiv.org/abs/1101.5785, 2011.

[45] L. Yu, J. -P. Barbot, G. Zheng, and H. Sun, "Compressive Sensing for Cluster Structured Sparse Signals: Variational Bayes Approach," 2011.