28th Picture Coding Symposium, PCS2010, December 8-10, 2010, Nagoya, Japan

DICTIONARY LEARNING-BASED DISTRIBUTED COMPRESSIVE VIDEO SENSING+

Hung-Wei Chen, *Li-Wei Kang*, *and Chun-Shien Lu**
Institute of Information Science, Academia Sinica, Taipei, Taiwan
E-mail: {hungwei, lwkang, lcs}@iis.sinica.edu.tw

ABSTRACT

We address an important issue of fully low-cost and low-complex video compression for use in resource-extremely limited sensors/devices. Conventional motion estimation-based video compression or distributed video coding (DVC) techniques all rely on the high-cost mechanism, namely, sensing/sampling and compression are disjointedly performed, resulting in unnecessary consumption of resources. That is, most acquired raw video data will be discarded in the (possibly) complex compression stage. In this paper, we propose a dictionary learning-based distributed compressive video sensing (DCVS) framework to "directly" acquire compressed video data. Embedded in the compressive sensing (CS)-based single-pixel camera architecture, DCVS can compressively sense each video frame in a distributed manner. At DCVS decoder, video reconstruction can be formulated as an l_1-minimization problem via solving the sparse coefficients with respect to some basis functions. We investigate adaptive dictionary/basis learning for each frame based on the training samples extracted from previous reconstructed neighboring frames and argue that much better basis can be obtained to represent the frame, compared to fixed basis-based representation and recent popular "CS-based DVC" approaches without relying on dictionary learning.

Index Terms—Compressive sensing, sparse representation, dictionary learning, single-pixel camera, l_1-minimization.

1. INTRODUCTION

Conventional high-complexity video compression techniques [1] or recently popular low-complexity technique called distributed video coding (DVC) [2] all rely on the high-cost mechanism where video sensing and compression tasks are disjointedly performed. Most acquired raw pixel data in the sensing stage will be discarded in the (possibly) complex compression stage, which suffers from unnecessary memory wasting and power consumption, and is especially unfeasible for resource-extremely limited devices/ sensors. Recently, with the advent of the compressive sensing (CS)-based single-pixel camera architecture [3], based on the inherent sparse property of images, CS [4] can directly and efficiently acquire compressed image data via randomly projecting raw data to obtain linear and non-adaptive measurements. Image reconstruction can be formulated as solving an l_1-minimization problem [5]-[6] based on the acquired data measurements.

Recently, compressive video sensing integrating both video sensing and compression into a unified task has emerged as a new way to directly acquire compressed video data via random projection for each individual frame at a low-complexity encoder. Video reconstruction can be achieved via performing l_1-minimization together with exploiting correlations among successive frames at a high-complexity decoder [7]-[9]. In [7], we have proposed a distributed compressive video sensing (DCVS) framework, where an efficient initialization and several stopping criteria were designed to improve and speedup the employed l_1-algorithms using CS were proposed, where the major core is to assume each block in a frame can be sparsely represented with respect to the dictionary/basis formed from a set of spatially local neighboring blocks (without performing dictionary learning) of previous reconstructed neighboring frames, denoted as the "W/O dictionary learning"-based scheme in this paper.

In this paper, a DCVS framework via "dictionary learning"-based sparse representation is proposed. Our major contributions include: **(i) Single-pixel camera-compatible low-complexity video encoder:** only CS random projection will be individually performed for each frame, which can be fully compatible to the single-pixel camera [3]. In [8]-[9], it is required to support the H.264/AVC encoder to periodically encode each intra-frame, which is more complex. **(ii) Dictionary-learning based sparse representation:** a dictionary learned from a set of blocks globally extracted from the previous reconstructed neighboring frames together with the side information generated from them is used as the basis of each block in a frame. The major advantages are: (a) Extracting more blocks globally for dictionary learning can provide much better representation for blocks with large motions; and (b) Even if the qualities of the training blocks are not good enough (due to poorly reconstructed neighboring frames), the learned dictionary may still provide a good basis. The fact can be similarly explained by the image denoising approach via the dictionary learned from the patches extracted from a noisy image itself [10]. In contrast, the "W/O dictionary learning" approach [8]-[9] may not work well for: (a) blocks with (very) large motions; and (b) the use of non-learned dictionary formed from (possibly) low-quality blocks. Other technical comparisons can be found in Table 1 of Sec. 4.

2. COMPRESSIVE SENSING

Assume that an orthonormal basis matrix (or dictionary) $\Psi \in \mathrm{R}^{N\times N}$ (*e.g.*, DWT basis) can provide a K sparse representation for a signal $x \in \mathrm{R}^{N\times 1}$, *i.e.*, $x = \Psi\theta$, where $\theta \in \mathrm{R}^{N\times 1}$ can be well approximated using only $K << N$ non-zero entries. Compressive sensing (CS) [4] states that x can be accurately reconstructed by taking only $M = \mathrm{O}(K\times log(N/K))$, $K < M << N$, linear and non-adaptive measurements from the random projection as $y = \Phi x$, where $y \in \mathrm{R}^{M\times 1}$ is a measurement vector and $\Phi \in \mathrm{R}^{M\times N}$ is a measurement matrix that is incoherent with Ψ. More specifically, the M measurements in y are random linear combinations of the entries of x, which can be viewed as the compressed version of x. The reconstruction of θ (or x) can be formulated as an l_1-minimization problem. On the other hand, a basis matrix is actually not necessarily orthonormal. An overcomplete dictionary D learned from training some selected training samples [10] can be used as a basis for representing the original signal. In fact, by using a measurement matrix Φ randomly

+This work was supported in part by the National Science Council, Taiwan, under Grants NSC97-2628-E-001-011-MY3, NSC98-2631-H-001-013, NSC98-2811-E-001-008, NSC99-2218-E-001-010, and NSC99-2811-E-001-006.
*Corresponding author: lcs@iis.sinica.edu.tw.

978-1-4244-7135-5/10/$26.00 ©2010 IEEE

generated from some distribution, the incoherence between Φ and D should be usually high enough.

3. PROPOSED DCVS FRAMEWORK

3.1. Problem Formulation

In DCVS, a video sequence consists of several GOPs (group of pictures), where a GOP consists of a key frame followed by some CS frames. At DCVS encoder, each key frame or each block in a CS frame can be compressed via CS random projection to get its measurement vector. Here, the used measurement matrix is the scrambled block Hadamard ensemble (SBHE) matrix [5], which takes the partial block Hadamard transform, followed by randomly permuting its columns. At DCVS decoder, the reconstruction of a frame or a block can be formulated as an l_1-minimization problem. Here, the sparse coefficients with respect to different basis functions (or dictionaries) depending on various types of frames are solved via the "sparse reconstruction by separable approximation (SpaRSA)" algorithm [6]. Employed different basis functions or learned dictionaries will be described in Secs. 3.3~3.5.

3.2. DCVS Encoder

At DCVS encoder shown in Fig. 1, without acquiring complete raw video data and performing motion estimation, each key frame $x_t \in \mathrm{R}^{N\times 1}$ viewed as a column vector is compressed via frame-based random projection as $y_t = \Phi x_t$, where $y_t \in \mathrm{R}^{M_t\times 1}$ is the measurement vector, $M_t < N$, forming the compressed version of x_t, which will be transmitted to the decoder. $\Phi \in \mathrm{R}^{M_t\times N}$ is the measurement matrix [5]. On the other hand, each CS frame x_t consisting of B non-overlapping blocks, $b_{ti} \in \mathrm{R}^{N_b\times 1}$ viewed as a column vector, i = 1, 2, …, B, is compressed via block-based random projection by individually projecting each b_{ti} via $y_{ti} = \Phi b_{ti}$, where $y_{ti} \in \mathrm{R}^{M_{ti}\times 1}$ is the measurement vector, $M_{ti} < N_b$, and $\Phi \in \mathrm{R}^{M_{ti}\times N_b}$ is the measurement matrix [5]. The vectors y_{ti}, i = 1, 2, …, B, forming the compressed version of x_t, will be transmitted to the decoder.

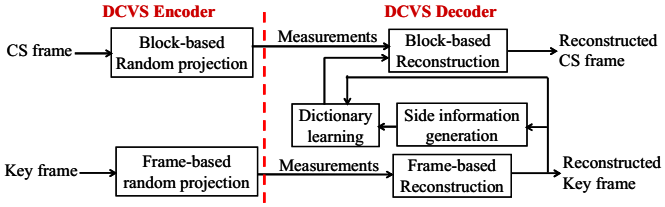

Fig. 1. Proposed DCVS with dictionary learning.

3.3. DCVS Decoder for Key Frame Reconstruction

At DCVS decoder, each key frame x_t can be reconstructed via solving the l_1-minimization problem as:

$$\min_{\theta_t} \frac{1}{2}\|y_t - A\theta_t\|_2^2 + \tau\|\theta_t\|_1, \qquad (1)$$

where y_t is the received measurement vector, $y_t = \Phi x_t$, $A = \Phi\Psi$, Φ is the measurement matrix [5], Ψ is the DWT basis, $\theta_t \in \mathrm{R}^{N\times 1}$ is the sparse coefficient vector to be solved via SpaRSA algorithm [6] with respect to Ψ, and τ is a non-negative parameter. Finally, the key frame x_t can be reconstructed via $\tilde{x}_t = \Psi\tilde{\theta}_t$, where $\tilde{\theta}_t$ is the solution of θ_t minimizing Eq. (1). For achieving the goal of independent reconstruction of a key frame, a general-purpose basis, DWT basis, for image representation is employed.

3.4. DCVS Decoder for CS Frame Reconstruction

At DCVS decoder, each CS frame x_t can also be reconstructed via solving the l_1-minimization problem for each block b_{ti}, i = 1, 2, …, B, in x_t as:

$$\min_{\alpha_{ti}} \frac{1}{2}\|y_{ti} - A_t\alpha_{ti}\|_2^2 + \tau\|\alpha_{ti}\|_1, \qquad (2)$$

where y_{ti} is the received measurement vector for b_{ti}, $y_{ti} = \Phi b_{ti}$, $A_t = \Phi \mathrm{D}_t$, Φ is the measurement matrix [5], $\mathrm{D}_t \in \mathrm{R}^{N_b\times P}$, $N_b \le P$, is the learned dictionary for x_t, described in Sec. 3.5, $\alpha_{ti} \in \mathrm{R}^{P\times 1}$ is the sparse coefficient vector to be solved via SpaRSA algorithm [6] with respect to the basis D_t, and τ is a non-negative parameter. Similarly, b_{ti} can be reconstructed via $\tilde{b}_{ti} = \mathrm{D}_t\tilde{\alpha}_{ti}$, where $\tilde{\alpha}_{ti}$ is the solution of α_{ti} minimizing Eq. (2). That is, each block b_{ti} in x_t can be sparsely represented as a linear combination of the atoms (column vectors) in D_t. Finally, the CS frame x_t can be reconstructed by integrating $\tilde{b}_{ti}$, i = 1, 2, …, B.

3.5. Dictionary Learning for CS Frame Reconstruction

If the basis/dictionary for an image can be learned based on the training samples/atoms extracted from the image itself, this basis should provide much sparser representation for the image. Although, it is impossible to obtain such the basis from an image itself to be reconstructed at decoder, a good dictionary learned from the training samples generated from the neighboring frames of a video frame to be reconstructed may be still obtained. Based on the general fact that the contents of successive frames in the same scene of a video should be similar, a frame can be well-predicted based on its side information possibly generated from the interpolation of its neighboring reconstructed frames.

At DCVS decoder, for a CS frame x_t, its side information I_t can be generated from the motion-compensated interpolation (MCI) of its previous and next reconstructed key frames, respectively, denoted by x_{t-j} and x_{t+j}. MCI technique has been successfully used for side information generation in DVC [2]. Then, we use the three frames, x_{t-j}, I_t, and x_{t+j} to learn the dictionary (basis) for this CS frame x_t as follows. First, we extract Q training patches $\mathrm{u}_i \in \mathrm{R}^{N_b}$, i = 1, 2, …, Q, from x_{t-j}, I_t, and x_{t+j}, where each frame is divided into several non-overlapping blocks. For each block in the three frames, we extract the 9 training patches including the nearest 8 blocks overlapping this block and this block itself, where each extracted patch $\mathrm{u}_i \in \mathrm{R}^{N_b}$ can be viewed as a column vector. Second, we apply the K-SVD algorithm [10] to these Q training patches to learn the dictionary $\mathrm{D}_t \in \mathrm{R}^{N_b\times P}$, $N_b \le P$, for x_t, where D_t is an overcomplete dictionary containing P atoms. With respect to D_t, each block b_{ti} in x_t can be sparsely represented as a sparse coefficient vector $\alpha_{ti} \in \mathrm{R}^{P\times 1}$ and, usually, $\|\alpha_{ti}\|_0 << N_b$. Using the learned dictionary for all the blocks of a CS frame can usually provide sparser representation for the frame than using a fixed DWT basis.

An illustrative example of the *Foreman* QCIF video sequence at measurement rate (*MR*, defined by the number of acquired measurements divided by the number of pixels of a frame) = 0.3 shown in Fig. 2 is used to demonstrate the efficiency of DCVS

decoder, where the parameter settings are described in Sec. 4. Fig. 2(a) and (b) show, respectively, an original CS frame (the 32nd frame), and its dictionary with size 256×256, where each atom (column vector) with length 256 in the dictionary is displayed as a block. Fig. 2(c) and (d), respectively, show the reconstructed CS frame using the dictionary shown in Fig. 2(b) and the frame-based DWT basis (treat this frame as a key frame). It can be observed from Fig. 2 that using the learned dictionary can provide better CS frame reconstruction than using the DWT basis at the same *MR*.

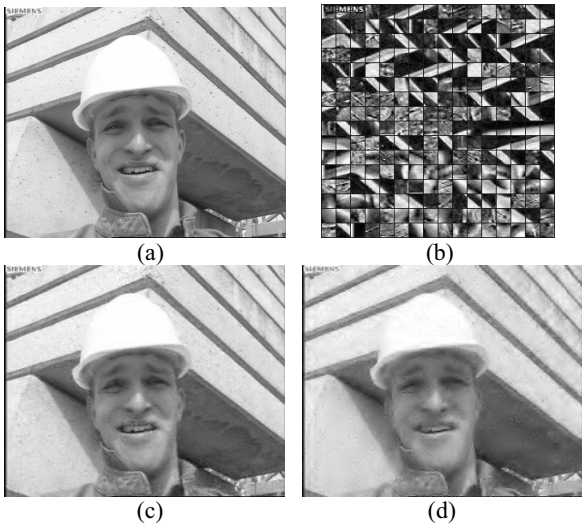

Fig. 2. Comparison of reconstructed CS frames with respective to learned and fixed dictionaries: (a) The original 32nd frame; (b) the dictionary learned for (a); (c) the reconstructed 32nd frame with respective to the dictionary shown in (b) (PSNR = 31.49dB); and (d) the reconstructed 32nd frame with respect to the frame-based DWT basis (PSNR=27.83dB).

4. SIMULATION RESULTS

In this paper, several QCIF (frame size: 176×144) video sequences (51 Y frames for each) with GOP size = 2, and different measurement rates (*MR*s) were employed to evaluate the proposed DVCS scheme. For learning the dictionary for each CS frame consisting of several non-overlapping 16×16 blocks, the parameter settings are described as follows. The dictionary size was set to 256×256, *i.e.*, N_b = 16×16 = 256 and P = 256 (atoms). In K-SVD [10], the number of training iterations was set to 10 while the target sparsity, denoted by S (number of nonzero coefficients used to represent each signal/block) was set to 10. According to our simulations, the performances will not exhibit significant changes when the two above-mentioned parameters for K-SVD are increased, which will increase the complexity of dictionary learning. Currently, the *MR* of all the frames in a video sequence are set to be the same as the target *MR*. In addition, to keep the encoding complexity to be as low as possible, the available measurements for each CS frame are equally allocated to each block without considering the complexity or sparsity of the block.

In this paper, two compressive video sensing schemes were used for comparison with our dictionary learning-based DCVS scheme (denoted by **Proposed**). The first one is the **Frame-DWT** scheme, in which under our DCVS architecture, all frames are treated as key frame (reconstructed with respect to the frame-based DWT basis). The second one is the "**W/O dictionary learning"** scheme, in which based on our DCVS architecture, each block in a CS frame is reconstructed with respect to its corresponding dictionary without learning. The second type is similar to the major core in [8]-[9]. Here, based on [8], the dictionary of each block in a CS frame includes the blocks extracted from the two spatially corresponding square 17×17 windows, respectively, in the two neighboring reconstructed key frames. The characteristics of the "Proposed" and the "W/O dictionary learning" schemes are summarized in Table 1. Please note that we only implemented the major core of the schemes proposed in [8]-[9] instead of the full system for comparison.

For reconstructing block b_{ti} in a CS frame x_t using SpaRSA, the computational complexity is approximately $O(P^{\beta})$, where P is decided by the dimension of $A_t \in \mathrm{R}^{M_{ti} \times P}$, and β is a constant. It has been shown that the complexity of SpaRSA is approximately linear (β is close to 1) [6]. In our parameter settings (Table 1), the dimension P (256) used by Proposed scheme is smaller than that (578) used by "W/O dictionary learning" scheme. Nevertheless, additional complexity for performing K-SVD dictionary learning [10] (approximately $Q \times (S^2 \times P + 2 \times N_b \times P)$ per training iteration [11], where Q is the number of training patches, S is the target sparsity, and $N_b \times P$ is the size of each dictionary D_t) is required for each CS frame in our scheme, which is, however, usually acceptable for a high-complexity decoder supported in a server or in cloud.

Table 1. Comparisons of the Proposed and "W/O dictionary learning" schemes.

Scheme	Proposed	W/O dictionary learning
Ingredients of dictionary	Learning based on the extracted patches from neighboring key frames and side information	Spatially neighboring blocks from neighboring key frames without learning
Dictionary size	256 atoms	Size of spatially corresponding square window × Number of neighboring key frames (17×17×2 = 578 atoms)
Number of dictionaries per CS frame	1	Number of blocks per CS frame (99 dictionaries for a QCIF CS frame)
Dictionary type	Global with learning	Local w/o learning
Decoding complexity per CS frame	Dictionary learning by K-SVD + l_1-minimization solving 256 coefficients per block	l_1-minimization solving 578 coefficients per block

The average PSNR performances at different *MR*s for the *Foreman*, *Mobile*, and *Silent* sequences are shown in Figs. 3(a), 4(a), and 5(a), respectively, where it can be observed that the PSNR performances of the proposed DCVS can outperform or be comparable to the Frame-DWT and "W/O dictionary learning" schemes [8]-[9], especially at lower *MR*s and for sequences with large motion. It can also be observed from Fig. 4(a) that the PSNR performances obtained from the three schemes are somewhat poor. The major reasons include: (i) the frame contents of the *Mobile* sequence are very complex, which may not be exactly sparse with respect to most bases, and (ii) the motions of the sequence are very large so that it is hard to learn a good dictionary for a CS frame from its neighboring key frames. It is worth noting that the dictionary learning of our DCVS can reveal some "denoising"

capability to obtain a basis better than that of the "W/O dictionary learning" scheme without relying on dictionary learning.

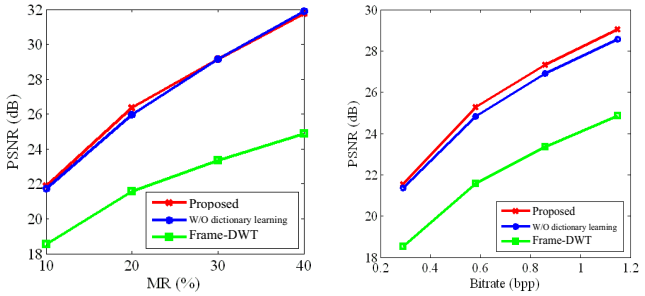

Fig. 3. The (a) MR-PSNR and (b) Bitrate-PSNR performances of the *Foreman* Sequence.

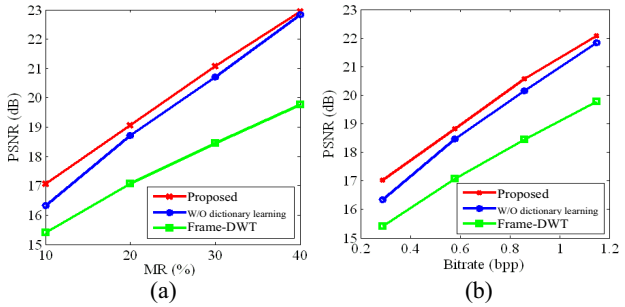

Fig. 4. The (a) MR-PSNR and (b) Bitrate-PSNR performances of the *Mobile* Sequence.

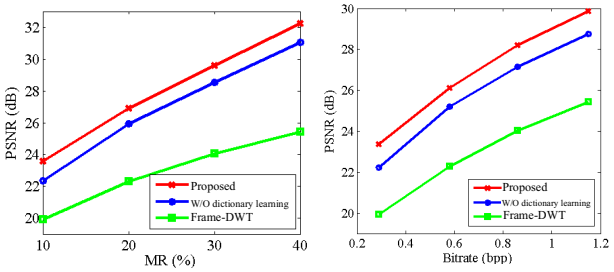

Fig. 5. The (a) MR-PSNR and (b) Bitrate-PSNR performances of the *Silent* Sequence.

On the other hand, to explore the compression efficiency in terms of PSNR-bitrate performances, we quantized each measurement via a nonuniform quantizer with 8 levels, generated using Lloyd's algorithm [12]. Then, we encoded each quantized measurement using an entropy encoder designed by Huffman coding, where each measurement was averagely encoded by 2.9 bits. The average PSNR performances at the four different bitrates (bits per pixel, *i.e.*, bpp), respectively, obtained by encoding the measurements for *MR* = 10%, 20%, 30%, and 40% for the three evaluated sequences, are shown in Figs. 3(b), 4(b), and 5(b), respectively, where it can be observed that the proposed DCVS can outperform or be comparable to the Frame-DWT and "W/O dictionary learning" schemes. That is, with some quantization noises, adaptive learned dictionaries can reveal some "denoising" capability and provide much better bases, resulting in better reconstructed quality. In addition, the average number of bits (2.9 bits) for encoding a measurement using the entropy encoder is very close to that (3 bits) using fixed-length encoder. The reason is that the measurement matrix spreads the energy of a signal uniformly across the measurements, so that each measurement is nearly allocated the same number of bits [13].

5. CONCLUSIONS

In this paper, a single-pixel camera-compatible dictionary learning-based distributed compressive video sensing (DCVS) framework is proposed to directly acquire compressed video. The simulation results have shown that the learned dictionary can provide better basis for video reconstruction than using the DWT basis and dictionary without learning-based basis. For the future works, several important issues need to be investigated in depth are described as follows. (i) Adaptive measurement matrix learning: (ii) Optimal measurement quantization and allocation. (iii) Bit allocation and entropy coding for measurements. (iv) Fast dictionary learning. (v) More efficient algorithm solving the l_1-minimization problem.

6. REFERENCES

[1] T. Wiegand and G. J. Sullivan, "The H.264/AVC video coding standard," *IEEE Signal Processing Magazine*, vol. 24, no. 2, pp. 148-153, March 2007.

[2] F. Dufaux, W. Gao, S. Tubaro, and A. Vetro, "Distributed video coding: trends and perspectives," *EURASIP Journal on Image and Video Processing*, Article ID 508167, 2009.

[3] 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 Magazine*, vol. 25, no. 2, pp. 83-91, March 2008.

[4] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21-30, March 2008.

[5] L. Gan, T. T. Do, and T. D. Tran, "Fast compressive imaging using scrambled hadamard ensemble," in *Proc. of European Signal Processing Conf.*, Switzerland, Aug. 2008.

[6] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, "Sparse reconstruction by separable approximation," *IEEE Trans. on Signal Processing*, vol. 57, no. 7, pp. 2479-2493, July 2009.

[7] L. W. Kang and C. S. Lu, "Distributed compressive video sensing," in *Proc. of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, Taipei, Taiwan, April 2009, pp. 1169-1172.

[8] J. Prades-Nebot, Y. Ma, and T. Huang, "Distributed video coding using compressive sampling," in *Proc. of Picture Coding Symposium*, Chicago, Illinois, USA, May 2009.

[9] T. T. Do, Y. Chen, D. T. Nguyen, N. Nguyen, L. Gan, and T. D. Tran, "Distributed compressed video sensing," in *Proc. of IEEE Int. Conf. on Image Processing*, Cairo, Egypt, Nov. 2009, pp. 1393-1396.

[10] 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*, vol. 54, no. 11, pp. 4311-4322, Nov. 2006.

[11] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," *CS Technical Report, Technion - Israel Institute of Technology*, 2008.

[12] W. Dai, H. V. Pham, and O. Milenkovic, "Distortion-rate functions for quantized compressive sensing," in *Proc. of IEEE Information Theory Workshop on Networking and Information Theory*, June 2009.

[13] V. K. Goyal, A. K. Fletcher, and S. Rangan, "Compressive sampling and lossy compression," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 48-56, 2008.