# ADAPTIVE MEASUREMENT RATE ALLOCATION FOR BLOCK-BASED COMPRESSED SENSING OF DEPTH MAPS

*Krishna Rao Vijayanagar, Ying Liu, Joohee Kim*

Dept. of Electrical and Computer Engineering, Illinois Institute of Technology, USA.

**ABSTRACT**

In recent years, compressed sensing (CS) has been used for compressing depth maps for the multi-view video plus depth. In these compression schemes, every block of the depth map is sampled with a fixed non-adaptive sensing matrix and the algorithms are generally incorporated into conventional codecs like H.264/AVC, resulting in high computational complexity both at the encoder and decoder. In this paper, we present a novel block-based CS codec for compressing depth maps that has two major features. First, an adaptive measurement rate allocation algorithm is introduced that computes the measurement rate for each compressively sensed block using rate-distortion optimization (RDO). Second, a simple block classification and frame-differencing module is utilized to reduce encoding complexity while maintaining good RD performance. Simulation results clearly show that the proposed codec has superior rate-distortion (RD) performance in comparison to H.264/AVC Baseline Profile (up to 3 dB gain) and an encoding time reduction of up to 97%.

***Index Terms—*** Compressed sensing, depth map compression, rate-distortion optimization, dynamic measurement rate.

## 1. INTRODUCTION

In the past few years, there has been a great deal of interest in compressed sensing (CS) which states that robust and accurate signal recovery is possible when a signal is sampled with sub-Nyquist rates if it is sparse in some orthonormal basis. Examples of such orthonormal basis sets are the wavelet basis and the discrete cosine transform (DCT). Theoretical foundations for CS can be found in [1] [2] [3]. In this paper, we consider the use of CS for compressing depth maps for the multi-view video plus depth format.

Recently, depth map compression methods using compressed sensing [5] [6] have been proposed which are largely based on the principles laid down for compressed sensing of images and video. However, depth maps have characteristics that are different from that of images. They contain large, smooth, piecewise-constant regions with sharp transitions at object boundaries and these characteristics have to be taken into consideration during compression. In [5], the measurement rate is the same for every block of the frame irrespective of the characteristics of the depth map. In [6], an attempt is made for classifying the blocks based on their depth characteristics. However, the classification scheme presented in [6] is the same as [4] and it uses convex optimization at the encoder to estimate the distortion and this increases the encoding complexity tremendously. Furthermore, the measurement rate at which the blocks are sensed is kept constant and this results in very little control over the overall coding bitrate and quality.

In this paper, we propose a CS-based depth map codec that computes the measurement rate for every compressively sensed block in an RD optimal sense. Encoding complexity is greatly reduced by eliminating motion estimation and compensation and by not using convex optimization at the encoder for classifying the blocks. It is also a full-fledged codec that does not have to be incorporated into H.264/AVC, thus reducing the encoding and decoding complexity.

The rest of the paper is organized as follows. In Section 2, we briefly describe the basics of CS. In Section 3, we introduce the proposed architecture and provide a detailed explanation of the encoder and decoder. In Section 4, we provide detailed simulation results. Finally, the paper is concluded in Section 5.

## 2. BASICS OF COMPRESSED SENSING

Let us consider a real-valued vector $\mathbf{x}$ of length $N \times 1$. Using an orthonormal basis set $\Psi$, we can represent $\mathbf{x}$ as $\mathbf{x} = \Psi\mathbf{s}$ where $\mathbf{s}$ is the coefficient vector. If the number of non-zero elements in $\mathbf{s}$ is $k$, then $\mathbf{x}$ is $k$-sparse with respect to $\Psi$. Compressed sensing (CS) claims that the signal $\mathbf{x}$ can be recovered accurately from very small number of samples (far less than the Nyquist rate). That is, if we are given $M$ measurements of $\mathbf{x}$, where $M \ll N$, then the exact recovery of $\mathbf{x}$ is possible under certain conditions. Hence, the idea is to recover $\mathbf{x} \in \Re^N$ from

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

where $\mathbf{y}$ has length $M$ and $\Phi$ is an $M \times N$ measurement matrix with sampling rate $S = \frac{M}{N}$ and

$$\mathbf{y} = \Phi\Psi\mathbf{s}. \tag{2}$$

Setting $A = \Phi\Psi$, we can re-write Eq. (2) as,

$$\mathbf{y} = A\mathbf{s}, \tag{3}$$

where $\mathbf{y}$ is the output vector of size $M \times 1$. Given the measurements $\mathbf{y}$, the sensing matrix $\Phi$, and the basis set $\Psi$, the problem at hand is to recover the estimate of the signal $\mathbf{x}$ denoted by $\tilde{\mathbf{x}}$. This can be done by exploiting sparsity in the gradient domain using Total Variational (TV) minimization. Most images are sparse in the gradient domain and thus we can use TV minimization to recover the pixels from the CS samples. That is, we compute,

$$\min_{\tilde{\mathbf{x}}} \|\tilde{\mathbf{x}}\|_{TV} + \lambda \|\mathbf{y} - \Phi\tilde{\mathbf{x}}\|_2 \leq \epsilon, \tag{4}$$

where $\tilde{\mathbf{x}}$ is the 1-D rasterization of the CS block. The total variation of the block is defined as

$$\|\tilde{\mathbf{x}}\|_{TV} = \sum_{i,j} \sqrt{(x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2}, \tag{5}$$

where $x_{i,j}$ is the pixel at coordinates $(i, j)$ in $\tilde{\mathbf{x}}$.

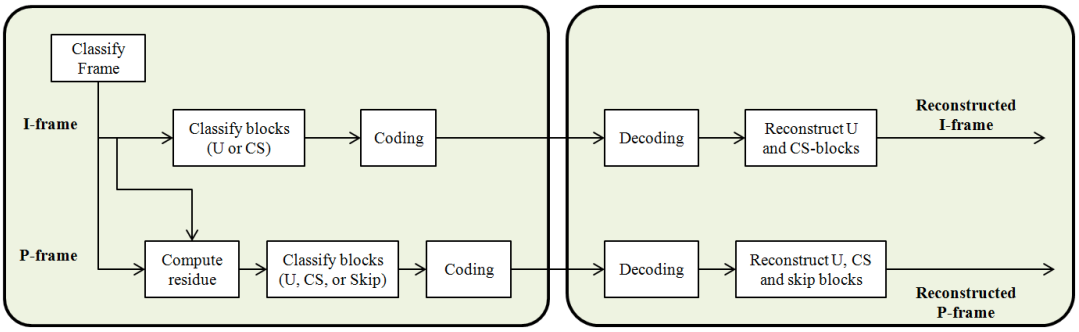



**Fig. 1**. Proposed hybrid CS codec architecture.

## 3. ARCHITECTURAL OVERVIEW

The architecture of the proposed codec is shown in Fig. 1. The codec classifies the first frame of every group-of-pictures (GOP) as an intra (I) frame with the remaining frames of the GOP being classified as inter (P) frames. Next, the blocks of an I frame are classified into uniform (U) and compressively sensed (CS) blocks. The blocks of a P frame are classified into either U, CS, or Skip blocks. These blocks are then compressed and sent to the decoder along with header information where the bitstream is decoded and the depth map is reconstructed. We look at each of these steps in detail by analyzing the encoder and decoder separately.

### 3.1. Encoder design

#### *3.1.1. Frame classification and residue generation*

The frame classification and residue generation step is a preprocessing stage to prepare the frame for compression. If the frame is an I frame, then there is nothing to be done and the depth map is sent as-is to the transform step. However, if it is a P frame, then a residue frame is generated by computing the frame difference between the current frame and the previous frame. This is done to remove temporal redundancy and we do not use the traditional motion estimation and compensation in order to keep the encoder complexity low.

#### *3.1.2. CS acquisition*

In this step, the depth map (or frame residue for P frames) is divided into non-overlapping $8 \times 8$ blocks and the pixels are compressed sensed with sub-Nyquist sampling. To achieve high coding efficiency, we propose to decorrelate a block of pixels with a subset of the 2D-DCT basis vectors and the resulting partial DCT coefficients are the sub-Nyquist rate CS samples which are transmitted for depth map reconstruction at the decoder. Following the transform step, all the blocks are quantized as follows. Given a dead-zone threshold $\Delta$, a quantization step size $q$ and the CS sample $y$, the quantized sample $y_q$ is given by Eq. (6). In our algorithm, the quantization step size is constant for DC measurements of both I and P frames and it is fixed at 8. However, for AC measurements of I and P frames, the quantization step sizes range from 2 to 16. We set $\Delta = q$ for simplification.

$$y_q = \begin{cases} 0, & \text{for } |y| \leq \Delta, \\ \left\lceil \dfrac{y-\Delta}{q} \right\rceil, & \text{for } y > +\Delta, \\ \left\lfloor \dfrac{y+\Delta}{q} \right\rfloor, & \text{for } y < -\Delta. \end{cases} \tag{6}$$

#### *3.1.3. Classification*

After quantization, we vectorize the quantized CS samples of every block using a zig-zag scan and create a matrix $\mathbf{D}$ whose columns are the blocks of the depth map in raster scan order. By examining the CS samples in $\mathbf{D}$, we classify every block into different coding categories. For an I frame, the blocks are classified as either U or CS blocks. If a particular block has all-zero AC measurements, then it is classified as a U block. Such a block need not be reconstructed using convex optimization at the decoder. All other blocks with non-zero AC measurements are classified as CS blocks.

For a P frame, the blocks are classified as either U, CS or Skip-blocks. If all the quantized measurements are zero-valued, then the block is identical to its co-located block in the previous frame and it is classified as a Skip-block. The classification of U and CS blocks is similar to what was done in the I frame. The classification procedure presented is much simpler than the method given in [6] which uses convex optimization at the encoder for classification. We later demonstrate that the simplicity of the proposed technique does not impact the RD performance negatively. We also note that all the classification decisions are recorded into a map that is later transmitted to the decoder.

#### *3.1.4. RD optimized measurement rate computation*

One of the goals of this paper is to present an RD optimized measurement rate allocation algorithm that computes the rate at which each CS block is sensed. In the proposed scheme, we first transform and quantize the blocks before they are compressively sensed. This reduces the complexity of the RDO, or else, the problem would become a joint optimization of quantization and measurement rates which is beyond the scope of this paper. Thus, assuming that the RDO module in the proposed codec receives quantized transform coefficients, we now explain the details of the RDO scheme.

Since the measurement rate decision determines how many AC measurements are retained in every CS block, we first estimate how the number of bits available for compressing the AC measurements of all the CS blocks. For this, the DC measurements of all the blocks are extracted and the difference between consecutive DC measurements is computed. The resulting differences, referred to as residual DC measurements are binarized using a $k^{th}$-order Exp-Golomb code. The value of $k = 0$ and this is a fixed value. The classification map is concatenated to this bitstream and compressed using an adaptive binary arithmetic code. Let the size of the compressed bitstream be denoted by $R_{DC} + R_{map}$ and let the bit-budget available for compressing the depth map be $R_{total}$, then the maximum

number of bits for encoding the AC measurements of the CS blocks is denoted by $R_{AC_{max}} = R_{total} - (R_{DC} + R_{map})$.

Let $\mathcal{X} = \{X_1, X_2, X_3, \ldots, X_P\}$ be the set of all CS blocks in the depth map, where $P$ is the total number of CS blocks in the depth map after classification. We specify that every block can be compressively sensed using only one of the measurement rates specified in $\mathcal{S} = \{S_1, S_2, S_3, \ldots, S_J\}$, where $0.0 < S_j \leq 1.0$ and $1 \leq j \leq J$. Here, $\mathcal{S}$ is a finite set of candidate measurement rates in order to ensure that the optimization problem is tractable. Let the measurement rate chosen for the $p^{th}$ block be denoted by $L_p$ where $1 \leq p \leq P$. Consequently, the set of all measurement rates for all the CS blocks is defined as $\mathcal{L} = \{L_1, L_2, L_3, \ldots, L_P\}$. We note that any given measurement rate $L_p \in \mathcal{L}$ should exist in $\mathcal{S}$. Now, the problem at hand is to estimate the best set of measurement rates for the CS blocks while reducing the overall distortion and ensuring that the number of bits used does not exceed $R_{AC_{max}}$. This can be formulated as a constrained optimization problem as

$$\begin{aligned} \mathcal{L}^* = \min_{\mathcal{L}} \quad & D(\mathcal{X}, \mathcal{L}) \\ \text{subject to} \quad & R(\mathcal{X}, \mathcal{L}) \leq R_{AC_{max}}. \end{aligned} \tag{7}$$

Here, $R(\mathcal{X}, \mathcal{L})$ and $D(\mathcal{X}, \mathcal{L})$ are the total rate and distortion, respectively, after compressively sensing all the CS blocks in the depth map using the measurement rates specified for each block in $\mathcal{L}$. In order to compute the rate $R(\mathcal{X}, \mathcal{L})$, we use Exp-Golomb coding to binarize the retained AC measurements of the CS blocks and then encode this bitstream using adaptive binary arithmetic coding. In order to estimate the distortion $D(\mathcal{X}, \mathcal{L})$, we first reconstruct every CS block using the inverse DCT operator. We note that at this stage, the CS blocks consist of only the retained AC measurements with the remaining coefficients set to zero. After this, we compute the sum of the mean square error (MSE) between the original CS blocks and the reconstructed blocks and set this as the distortion $D(\mathcal{X}, \mathcal{L})$. Here, we make an assumption that the distortion is representative of the distortion that would exist if convex optimization were to be employed to recover pixel information from the compressively sensed CS blocks. Assuming that the rate and distortion are additive, we can re-write Eq. (7) as an unconstrained Lagrangian formulation [7] [8] as

$$\mathcal{L}^* = \min_{\mathcal{L}} \quad \sum_{p=1}^{P} J(X_p, \mathcal{L}). \tag{8}$$

where $J(X_p, \mathcal{L})$ is the Lagrangian cost function for CS block $X_p$. Assuming that the blocks are encoded independently leading to additive rate and distortion terms and setting $\lambda_{opt} \geq 0$ as the Lagrangian multiplier, we can re-write Eq. (8) as

$$\mathcal{L}^* = \min_{\mathcal{L}} \quad \sum_{p=1}^{P} (D(X_p, L_p) + \lambda_{opt} R(X_p, L_p)), \tag{9}$$

$$\mathcal{L}^* = \quad \sum_{p=1}^{P} \min_{\mathcal{L}} (D(X_p, L_p) + \lambda_{opt} R(X_p, L_p)). \tag{10}$$

In this paper, we use the bisection algorithm to compute $\lambda_{opt}$. It should be noted that the proposed algorithm does not use convex optimization at the encoder for reconstructing the depth values while computing the distortion. Instead, we use the original DCT coefficients and the inverse DCT to obtain the pixel information.

After solving for $\lambda_{opt}$ and obtaining the RD optimal combination of measurement rates, we compressively sense each CS block using their corresponding measurement rates specified in $\mathcal{L}$. That is, the sensing matrix $\Psi$ for the $p^{th}$ CS block consists of only $64 \times \mathcal{L}_p$ 2D-DCT basis vectors corresponding to the most important $64 \times \mathcal{L}_p$ frequency components.

#### *3.1.5.* ***Entropy coding***

After sensing all the CS blocks, the codec needs to compress the block classification map, measurement rate chosen for each CS block, and the quantized CS samples for every block. Forward differences (or DPCM) is used to remove redundancy between DC coefficients, between the measurement rates, between classification codes in the map. The resultant residues are binarized using a $k^{th}$-order Exp-Golomb code to create a bitstream which is compressed using adaptive binary arithmetic coding and transmitted to the decoder.

### 3.2. Decoder design

The first step at the decoder is to decompress the bitstream and to reconstruct the various coding elements using inverse Exp-Golomb coding and inverse DPCM. Then, the I frames and P frames are reconstructed as follows.

#### *3.2.1.* ***I frame***

In an I frame, only U and CS blocks exist and they are decoded as follows. To recover the U blocks, firstly the quantized DC measurements belonging to the U blocks are inverse quantized (using the constant step size of 8). After this, we create an $8 \times 8$ transform coefficient matrix with the top-left entry being the DC measurement and the remaining entries set to zero. We apply an $8 \times 8$ IDCT transform to this matrix and recover the U block. The reconstruction of the CS blocks is done using pixel-domain TV minimization. The procedure for this has been described in Eqns. (4) and (5).

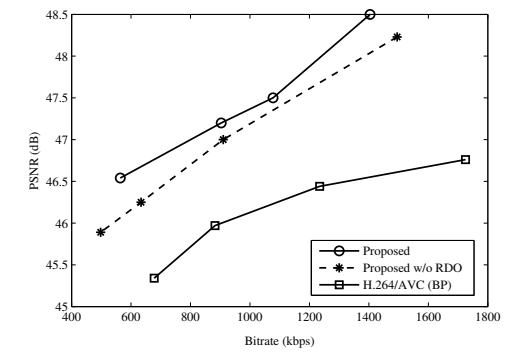
#### *3.2.2.* ***P frame***

In a P frame, we have Skip, U and CS blocks and they are decoded as follows. The reconstruction of skip blocks is simple and is done by copying the co-located block in the previously decoded frame. For recovering a U block belonging to a P frame, we first carry out the steps used for recovering a U block belonging to an I frame and this gives us the residue for that particular block. This is added to the co-located block of the previously decoded frame to obtain the final U block in the pixel domain.

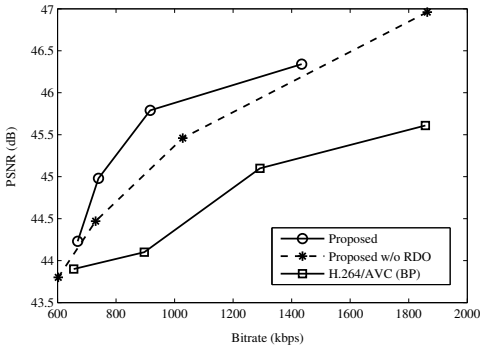
Similar to the recovery of the U blocks, we recognize that the CS measurements for the CS blocks are the results of projecting the residue to the frequency domain and not the depth values. The problem that arises here is that the residue data is not sparse in the gradient domain which is a prerequisite for TV minimization. To solve this problem, we have to make use of the co-located block in the previously decoded frame and transform the residue back to the pixel domain. The recovery of the CS blocks in P frames is carried out as follows. Assume that the current CS block is $X_p^t$ and the co-located block in the previous frame is $X_p^{t-1}$ where $P$ is the total number of CS blocks in the current frame and $1 \leq p \leq P$. Then, the residue block is given by,

$$X_{R_p}^t = X_p^t - X_p^{t-1}. \tag{11}$$

The residue is then compressively sensed to obtain $Y_{R_p}^t$ which is quantized, coded and transmitted to the decoder. Now, at the decoder, we have $\widehat{Y_{R_p}^t}$ which is the de-quantized version of $Y_{R_p}^t$ and due to lossy coding $\widehat{Y_{R_p}^t} \neq Y_{R_p}^t$. We also have access to $\widetilde{X_p^{t-1}}$ which is the co-located block in the previously decoded frame. We recognize that, due to lossy compression, $X_p^{t-1} \neq \widetilde{X_p^{t-1}}$. Using the measurement rate specified for $X_p^t$, we can compressively sense $\widetilde{X_p^{t-1}}$ to get $\widehat{Y_p^{t-1}}$. Now,




(a)




(b)

**Fig. 2**. RD simulation results for (a) Kendo and (b) Balloons multiview sequences.

$$\widehat{Y_p^t} = \widehat{Y_{R_p}^t} + \widehat{Y_p^{t-1}}, \tag{12}$$

where $\widehat{Y_p^t}$ is the result of compressively sensing, quantizing and dequantizing the desired output $\widetilde{X_p^t}$. With the help of Eq. (12), we can recover $\widetilde{X_p^t}$ by the use of TV-minimization as follows:

$$\min_{\widetilde{\mathbf{x}_p^t}} \left\| \widetilde{\mathbf{x}_p^t} \right\|_{TV} \text{ s.t. } \left\| \widehat{\mathbf{y}_p^t} - \Phi \widetilde{\mathbf{x}_p^t} \right\|_2 \leq \epsilon, \tag{13}$$

where $\widetilde{\mathbf{x}_p^t}$ and $\widehat{\mathbf{y}_p^t}$ are the vectorized versions of $\widetilde{X_p^t}$ and $\widehat{Y_p^t}$, respectively.

## 4. SIMULATION RESULTS

In this section, we evaluate the performance of the proposed codec. For the experiments, we use the first $50$ frames of views $1$ and $3$ of the Kendo and Balloons sequences sampled at $15$ fps and synthesize view $2$ at the decoder. The resolution of the sequences is $1024 \times 768$ pixels. The comparison is done with H.264/AVC Baseline Profile and a modification of the proposed codec that we refer to as Proposed w/o RDO. In Proposed w/o RDO, we disable RD optimized measurement rate selection and retain $40$ of the $64$ DCT coefficients of every CS block similar to [6]. However, the blocks are classified into the various categories in Proposed w/o RDO.

### 4.1. RD performance

We first compare the RD performance of the proposed codec with that of H.264/AVC Baseline Profile and Proposed w/o RDO. In every experiment, only the depth maps corresponding to views $1$ and $3$ are compressed and not the color views. The encoding structure used to compress the depth maps is I-P-I-P. . . (i.e., a GOP size of $2$) and the QPISlice and QPPSlice values for H.264/AVC are equal to each other and set to $24, 28, 32$, and $36$. The bitrates shown in the results correspond to the sum of the bitrates needed for encoding the left and right depth maps. The PSNR is computed with the virtual view synthesized using the transmitted depth maps and the reference virtual view generated using the ground truth depth maps. For view synthesis, we use VSRS 3.5 [9]. The resulting RD plots for the Kendo and Balloons sequences are shown in Figs. 2 (a) and (b), respectively. It is clear from Fig. 2 that the proposed codec has superior RD performance with respect to H.264/AVC Baseline Profile with a gain of up to $2 - 3.5$ dB in spite of having much lower encoder complexity. The excellent RD performance that can be attributed the intelligent block-classification system and the RD optimized adaptive measurement rate module. Further confirmation of the efficiency of the adaptive measurement rate module is obtained by the RD performance gap between the proposed codec and Proposed w/o RDO (which uses a fixed measurement rate similar to [6]). The results clearly demonstrate that the adaptive measurement rate module helps achieve superior RD performance because all the other coding tools are identical between the proposed codec and Proposed w/o RDO.

### 4.2. Complexity comparison

We evaluate the average time needed to encode the right and left depth map by the proposed codec and H.264/AVC Baseline Profile on an Intel Xeon processor (3.20 GHz) with 8 GB RAM. The proposed codec is written entirely in MATLAB 2010b and H.264/AVC is written in C. The encoding time comparison is given in Table 1 and it shows that the proposed codec takes $97\%$ less time to encode depth maps. This is because H.264/AVC needs to perform motion estimation and compensation and intra prediction for every block and also apply an inloop deblocking filter for each frame at the encoder. Whereas, the proposed codec only needs to compute the frame difference, classify the blocks, perform RD optimized measurement rate selection for the CS blocks. We can say that the proposed codec has superior RD performance at lower encoding complexity in comparison to H.264/AVC Baseline Profile.

**Table 1**. Complexity comparison.

| | Proposed | | H.264/AVC | | |
|---|---|---|---|---|---|
| Sequence | QP | Encoding time (sec) | QP | Encoding time (sec) | Speed -up |
| Kendo | 2 | 26.2 | 24 | 950 | 97.24% |
| | 4 | 25.0 | 32 | 995 | 97.48% |
| | 16 | 20.6 | 36 | 1007 | 97.95% |
| Balloons | 2 | 21.8 | 24 | 956 | 97.71% |
| | 4 | 20.3 | 32 | 917 | 97.78% |
| | 12 | 18.1 | 36 | 920 | 98.03% |

## 5. CONCLUSION

In this paper, we present a low-complexity CS codec for compressing depth maps. It incorporates an efficient block classification scheme that does not use convex optimization at the encoder unlike [4] [6]. We present an RD optimized measurement rate selection scheme for the CS blocks to dynamically vary the measurement rate based on its characteristics. Results shows that the proposed scheme is lower in complexity and has better RD performance in comparison to H.264/AVC Baseline Profile achieving close to a $3$ dB gain and a $97\%$ reduction in encoding time.

## 6. REFERENCES


[1] E. Candes and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies," *IEEE Trans. on Info. Theory*, vol. 52, no. 12, pp. 5406-5425, 2006.

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

[3] E. J. Candes and M. B. Wakin, "An introduction to compressive sampling," *IEEE Sig. Proc. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.

[4] T. T. Do, X. Lu, and J. Sole, "Compressive sensing with adaptive pixel domain reconstruction for block-based video coding," in *Proc. of IEEE Intl. Conf. on Image Proc.* (ICIP '10), pp. 3377–3380, 2010.

[5] M. Sarkis and K. Diepold, "Depth map compression via compressed sensing," in *Proc. of IEEE Intl. Conf. on Image Proc.* (ICIP '09), Nov. 2009.

[6] J. Duan, L. Zhang, R. Pan, and Y. Sun, "An improved video coding scheme for depth map sequences based on compressed sensing," in *Proc. of Intl. Conf. on Multimedia Tech.* (ICMT '11), pp. 3401–3404, Aug. 2011.

[7] G. J. Sullivan and T. Wiegand "Rate-distortion optimization for video compression," *IEEE Sig. Proc. Mag*, vol. 15, no. 6, pp. 74–90, 1998.

[8] T. Wiegand, M. Lightstone, T. G. Campbell, and S. K. Mitra, "Efficient mode selection for block-based motion compensated video coding," in *Proc. of IEEE Intl. Conf. on Image Proc.* (ICIP '95), vol. 2, pp. 559–562, Oct. 1995.

[9] "View synthesis reference software (VSRS 3.5)," in *Tech. Rep. ISO/IEC JTC1/SC29/WG11*, Mar. 2010.