Compressive sensing based video object compression schemes for surveillance systems

Narayanan, Sathya; Makur, Anamitra

2015

Narayanan, S., & Makur, A. (2015). Compressive sensing based video object compression schemes for surveillance systems. Proceedings of SPIE - Video Surveillance and Transportation Imaging Applications 2015, 9407, 94070W-. doi:10.1117/12.2081806

<https://hdl.handle.net/10356/88460>

<https://doi.org/10.1117/12.2081806>

© 2015 Society of Photo-optical Instrumentation Engineers (SPIE). This paper was published in Proceedings of SPIE - Video Surveillance and Transportation Imaging Applications 2015 and is made available as an electronic reprint (preprint) with permission of Society of Photo-optical Instrumentation Engineers (SPIE). The published version is available at: [<http://dx.doi.org/10.1117/12.2081806>]. One print or electronic copy may be made for personal use only. Systematic or multiple reproduction, distribution to multiple locations via electronic or other means, duplication of any material in this paper for a fee or for commercial purposes, or modification of the content of the paper is prohibited and is subject to penalties under law.

Downloaded on 26 Apr 2023 20:15:12 SGT

Compressive Sensing based Video Object Compression Schemes for Surveillance Systems

Sathiya Narayanan and Anamitra Makur

School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore
email: sathiya3@e.ntu.edu.sg

ABSTRACT

In some surveillance videos, successive frames exhibit correlation in the sense that only a small portion changes (object motion). If the foreground moving objects are segmented from the background they can be coded independently requiring far fewer bits compared to frame-based coding. Huang et al proposed a Compressive Sensing (CS) based Video Object Error Coding (CS-VOEC) where the objects are segmented and coded via motion estimation and compensation. Since motion estimation might be computationally intensive, encoder can be kept simple by performing motion estimation the decoder rather than at the encoder. We propose a novel CS based Video Object Compression (CS-VOC) technique having a simple encoder in which the sensing mechanism is applied directly on the segmented moving objects using a CS matrix. At the decoder, the object motion is first estimated so that a CS reconstruction algorithm can efficiently recover the sparse motion-compensated video object error. In addition to simple encoding, simulation results show our coding scheme performs on par with the state-of-the-art CS based video object error coding scheme. If the object segmentation requires more computations, we propose to deploy a distributed CS framework called Distributed Compressive Video Sensing based Video Object Compression (DCVS-VOC) wherein the object segmentation is done only for key frames.

Keywords: Distributed Compressive Sensing, video object coding, object motion

1. INTRODUCTION

Video compression is achieved commonly by removing spatial, temporal and frequency redundancies. Standard video compression techniques consider video frame as a random signal and achieves compression by exploiting its stochastic properties. Though most video compression techniques fall in the above mentioned category, there is an extensive underway to produce video compression systems that use object models in order to achieve compression. This may not be suitable for general video as there might not be a chance of segmenting objects. However, in the case of video telephony, video surveillance and medical imaging, objects can certainly be segmented and coded efficiently.

An object based video compression framework was proposed in which the objects in the video frame are detected, tracked and then coded using the coefficients of the most significant principle components obtained via incremental principle component analysis.¹ Then emerged the Compressive Sensing (CS) theory which ensured the recovery of a sparse signal $f \in R^n$ using a small number of linear observations of the form

$$y = \Phi f \in R^m \quad (1)$$

where $\Phi \in R^{m \times n}$ is a known sensing matrix.² Exact reconstruction of f can be achieved using convex relaxation technique (e.g. Basis Pursuit) or greedy pursuit (e.g. Orthogonal Matching Pursuit). Motivated by CS, Huang et al proposed a Video Object Error Coding method (CS-VOEC) which assumed the foreground moving objects have been segmented and object-based motion compensated from the previous frame, and then the sparse object error is coded using CS principle.³

Motion estimation at the encoder might be computationally intensive and therefore, encoder can be kept simple by pushing the motion estimation from encoder to decoder. On the other hand, CS on video also aims at developing a compression technique that has a simple encoder. In this paper, we propose a novel CS

This work is technically supported by research funding MOE2014-T2-1-079.

Video Surveillance and Transportation Imaging Applications 2015, edited by Robert P. Loce, Eli Saber,
Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 9407, 94070W · © 2015 SPIE-IS&T
CCC code: 0277-786X/15/\$18 · doi: 10.1117/12.2081806

based Video Object Compression (CS-VOC) technique having a simple encoder. In the encoder, the sensing mechanism is applied directly on the segmented moving objects using a CS matrix. While reconstructing them at the decoder, the object motion is estimated prior to reconstruction so that a CS reconstruction algorithm can efficiently recover the sparse motion-compensated video object error. If the object segmentation requires more computations, we propose to deploy Distributed Compressive Video Sensing based Video Object Compression (DCVS-VOC) wherein the object segmentation is required only for key frames.

The rest of this paper is organized as follows. In section 2, we review the video object error coding technique based on CS. We discuss our proposed CS-VOC in section 3 and its variant DCVS-VOC in section 4. We present the simulation results of our proposed scheme in section 5. Section 6 concludes the paper.

2. LITERATURE REVIEW

In this section, we briefly discuss CS-VOEC and state two CS reconstruction algorithms.

2.1 Video Object Error Coding based on Compressive Sensing

CS-VOEC combines MPEG-type motion estimation with CS. The encoder has three major components: object segmentation, motion estimation and compensation and CS based error coding. Figure 1 gives the overview of the object based video encoding scheme.

Figure 1. CS based Video Object Error Coding - Encoder

In object based video compression, object segmentation is the first step and it has a direct impact on the number of bits required to code the video frame. In,³ edge-based object segmentation technique was used to separate the foreground moving objects from the background. Using the object mask resulting from object segmentation step, a rectangular bounding box (object block) is created around each segmented object so that the motion estimation and compensation can be done for the foreground objects alone. For a search window W , the object based motion estimation is performed as follows

$$u_j = \arg \min_{u \in W} \sum M_j [f_{n-1}(u + x_j) - f_n(x_j)]^2 \quad (2)$$

where j indicates the object number, u_j indicates the estimated motion vector, M_j is the binary object mask, x_j is the location of the object in the current frame, f_n is the current frame and f_{n-1} is the previous frame. Object motion estimation is followed by object motion compensation where the predicted frame $\tilde{f}_n$ is obtained for each object. From the motion compensated object block, object error is obtained. The motion compensated object error is sparse in nature, and therefore, CS procedure is applied for coding it. The 2-D object error block is first reshaped into 1-D signal and then the CS measurements are obtained by projecting the 1-D signal onto a sensing matrix. CS measurement vector can also be quantized to store or transmit. At the decoder, the motion compensated error signal is reconstructed using CS recovery algorithms thereby exploiting the sparsity of the error signal.

2.2 CS reconstruction algorithms for decoding

Given the CS measurement vector of the sparse signal x , exact recovery is possible via Basis Pursuit (BP):

$$\min_{\beta} \|\beta\|_{\ell_1} \text{ s.t. } \Phi\beta = y. \quad (3)$$

For problems with partially known support, a CS reconstruction procedure called Modified-CS (MOD-CS) was proposed in.⁴ MOD-CS aimed at estimating the sparsest possible signal estimate whose support contains T and the optimization problem is formulated as

$$\min_{\beta} \|\beta_{T^c}\|_{\ell_1} \text{ s.t. } \Phi\beta = y \quad (4)$$

where $T^c := [1 : n] \setminus T$ is the complement of T .

3. PROPOSED COMPRESSIVE SENSING BASED VIDEO OBJECT COMPRESSION (CS-VOC)

CS based video compression schemes aim at developing a simple encoder unlike MPEG which has complex encoder and simple decoder due to motion estimation at the encoder. In CS-VOEC, though CS in the place of IPCA results in a simpler encoding compared to that of the framework in,¹ motion estimation step at the encoder might be computationally intensive. The encoder might still be complex which is a contradiction to the purpose of CS on video. We propose a video compression scheme in which the motion estimation is done at the decoder rather than at the encoder. This matches with the initial motivation of CS on video. The encoder of our proposed video compression scheme is shown in figure 2.

3.1 Simple Encoder

The encoder has two major components: object segmentation and CS based object coding. At first, the foreground objects are segmented from the background using the edge-based object segmentation scheme similar to that of VOEC. Object block is obtained for each segmented object by creating a rectangular bounding box around the object using the object mask. The vectorized object block is sensed using a random sensing matrix Φ . The measurement vector y_n is transmitted. Therefore, computationally intensive motion estimation is not performed at the encoder. In order to achieve better compression, the CS measurements can also be quantized using a uniform or non-uniform quantizer. The parameters to be transmitted to the decoder are: location and shape of the objects, dimensions of the measurement matrix and the CS coefficients.

Figure 2. Proposed CS based Video Object Compression - Encoder

3.2 Decoding Scheme using Motion Estimation at the decoder

At the decoder, we consider the problem of reconstructing the object error block given its measurement vector y_n and the previous reconstructed frame $\hat{f}_{n-1}$. The measurement matrix Φ and the object mask M_j are also available at the decoder. Our proposed decoder has three components: motion estimation and compensation, CS reconstruction of object error and recovery of video frame.

Motion estimation and compensation: In order to perform motion estimation in the measurement domain, the object based motion estimation in (2) is modified as

$$U_j = \arg \min_{u \in W} \sum [\Phi(M_j \hat{f}_{n-1}(u + x_j)) - y_n]^2 \quad (5)$$

Upon obtaining U_j using (5), to obtain the predicted frame $\tilde{f}_n$, motion compensation is performed as follows,

$$\tilde{f}_n = Bg \star M_j^c + \sum_j M_j [\hat{f}_{n-1}(U_j + x_j)] \quad (6)$$

where Bg denotes the background frame, M_j^c denotes the binary complement of the object mask, u_j denotes the 2-D motion vector computed from U_j and $\star$ represents the pointwise multiplication operation.

CS reconstruction of object error: The measurement vector $\tilde{y}_n$ for the motion compensated object block can be obtained as

$$\tilde{y}_n = \Phi \times M_j[\tilde{f}_n(x_j)]. \quad (7)$$

By expressing the difference between the measurement vectors y_n and $\tilde{y}_n$ as

$$y_n - \tilde{y}_n = \Phi \times M_j[f_n(x_j) - \tilde{f}_n(x_j)] \quad (8)$$

the object error $f_n(x_j) - \tilde{f}_n(x_j)$ can be efficiently reconstructed using CS recovery algorithms like Basis Pursuit (BP).² Since we have additional information in the form of M_j , variants of traditional CS algorithms such as Modified-CS (MOD-CS)⁴ can be applied for solving (8). Algorithm 1 summarizes the step-by-step procedure involved in reconstructing j^{th} object block using our proposed decoding technique.

Recovery of video frame: The video frame can then be recovered using the reconstructed object blocks, object mask and the available background using

$$\hat{f}_n = Bg \star M_j^c + \sum_j M_j[\hat{f}_n(x_j)]. \quad (9)$$

Algorithm 1 CS-VOC decoder - Recovery of j^{th} object block

Input:

- Φ - CS matrix used at the encoder
- M_j - Mask of the object block
- x_j - Location of the object/mask
- y_n - Measurement vector of the object block to be recovered
- $\hat{f}_{n-1}$ - Previous reconstructed frame

Procedure:

- Step 1: Obtain the object's motion vector using (5).
- Step 2: Perform motion compensation as in (6) to obtain the predicted frame $\tilde{f}_n$.
- Step 3: Obtain the measurement vector $\tilde{y}_n$ of the motion compensated object block using (7).
- Step 4: The motion compensated object error can be obtained by solving the CS problem (8).
- Step 5: Video object block is recovered by adding the obtained motion compensated object error to the motion compensated object block.

Output:

- $\hat{f}_n(x_j)$ - Reconstructed video object block
-

3.3 Complexity Analysis

In CS-VOEC, motion estimation at the encoder requires $O(nS)$ computations where S is the number of search points; motion compensation requires $O(n)$ computations and the CS acquisition requires $O(nm)$ computations. In our proposed method, encoder has a complexity of $O(nm)$ computations. Therefore, our scheme has a simpler encoder with a computational savings of that many required for motion estimation compared to that of the video compression schemes performing motion estimation at the encoder. CS recovery algorithms have a complexity of $O(mnK)$ where K is the sparsity of the target signal. Therefore, the decoder in CS-VOEC has a complexity of $O(mnK)$. In our proposed decoding scheme, CS acquisition processes require $O(nm)$ computations, motion estimation requires $O(mnS)$ computations and motion compensation requires $O(n)$ computations. Therefore, the decoder complexity for CS-VOC is $O(nmS) + O(n) + O(nm) + O(mnK) \approx O(mn[K + S])$. Summary of computations involved in each compression scheme is given in Table I.

Table 1. Comparison of Computational Complexity

CS video compression method	Encoder	Decoder
CS-VOEC	$O(n(m + S))$	$O(nmK)$
Proposed CS-VOC	$O(nm)$	$O(mn[S + K])$

4. DISTRIBUTED COMPRESSIVE VIDEO SENSING FRAMEWORK FOR VIDEO OBJECT COMPRESSION

The encoder of CS-VOC does not include motion estimation and therefore, it is much simpler compared to that of CS-VOEC. However, the object segmentation at the encoder might also require lot of computations depending upon the number and size of objects. Distributed Compressive Video Sensing (DCVS) technique proposed in⁵ classifies video frames into two categories: Key frames (I-frames) and CS-frames (non-key frames). Key frames are transmitted periodically after a certain number of CS frames. This is similar to the Group-of-Pictures in video codecs. Therefore, we propose DCVS-VOC scheme in which the object segmentation at the encoder is carried out only for the key frames.

Simple Encoder: For key frames, object segmentation is performed in order to extract the foreground moving objects. These frames are then coded by conventional video compression such as the MPEG/H.26x. For CS frames, a tentative object mask M_j^t is obtained by dilating the object mask of the preceding key frame M_{key} . The tentative object mask can be expressed as

$$M_j^t = M_{key} \oplus B \quad (10)$$

where $\oplus$ denotes dilation operation and B is a disk-shaped structuring element of radius equal to GOP length. CS frames are then coded as in the case of CS-VOC by using M_j^t in the place of M_j .

Decoder: At the decoder key frames are reconstructed via conventional MPEG/H.26x decoding. For recovering CS frames, a decoding procedure similar to that of CS-VOC is applied. As given by (5), motion estimation in CS-VOC decoder is performed with respect to the reconstructed previous frame available. Motion estimation in DCVS-VOC decoder is performed with respect to both preceding key frame and succeeding key frame separately. Therefore, two motion vectors are obtained. The motion vector indicating the smallest motion and the corresponding key frame are considered for motion compensation. Reconstruction of sparse object error and the recovery of object block are done exactly in the same manner as in CS-VOC. It is worth noting that the decoder complexity still remains as $O(nm[S + K])$.

5. RESULTS AND DISCUSSIONS

Our main aim was to design a video object coding scheme having a simple encoder and then to propose an appropriate decoding scheme to reconstruct the video objects. In this section, we present the decoding performance in terms of both frame PSNR and object PSNR (PSNR considering object pixels only). The search window for motion estimation is fixed as ± 7 pixels. For our experiments we used video frames (of size 288×384) from the Walk sequence. For each frame, foreground objects are segmented from its background and then our proposed compression schemes are applied.

Experiment-1: For this experiment, 80 frames are taken from the ‘walk’ sequence and the video compression schemes (CS-VOEC, CS-VOC and DCVS-VOC) are applied. The undersampling ratio $\frac{m}{n}$ is varied from 0.1 to 0.5 in steps of 0.1. The ratio $\frac{m}{N}$ is much less than 1 where N is the total number of pixels in the frame. In the case of CS-VOC, object PSNR is obtained for the entire frame sequence, and the average object PSNR is computed. For DCVS-VOC, object PSNR is obtained for all CSframes in the sequence and its average value is computed. DCVS-VOC performance is evaluated for two GOP sizes (4 and 8). For GOP size of 8, the search window for motion estimation is set as ± 10 pixels. In this experiment, BP is used in the place of CS reconstruction algorithm. Figure 3 shows the variation of average object PSNR against the undersampling ratio. It can be seen that the performance of CS-VOC is much closer compared to that of CS-VOEC. Also, DCVS-VOC gives the best performance compared to CS-VOEC and CS-VOC. For an under-sampling ratio of 0.3, if the GOP size is fixed as 4, DCVS-VOC gives an average object PSNR of 26.84 dB whereas CS-VOEC and CS-VOC gives 23.16

Figure 3. Object PSNR (in decibels) versus undersampling ratio

dB and 22.71 dB respectively. Time taken by DCVS-VOC encoder for coding those 80 frames is 24.94 seconds whereas the times taken by CS-VOEC and CS-VOC encoders are 84.12 seconds and 81.95 seconds respectively.

Experiment-2: This experiment is to verify the improvement in CS reconstruction performance due to MOD-CS. Two CS reconstruction algorithms were applied: BP and MOD-CS. The object PSNR values are given in Table 2. MOD-CS (using the object mask information) gives an improved performance compared to BP. CS-VOC using BP results in an object PSNR of 22.72 dB for an under-sampling ratio of 0.3 whereas CS-VOC using MOD-CS results in 23.40 dB for a ratio of 0.2. Therefore, effective use of mask information reduces the number of measurements required to sense the object block.

Table 2. Comparison of object PSNR (in decibels) with BP and MOD-CS

Compression Scheme	$\frac{m}{n}=0.15$		$\frac{m}{n}=0.2$		$\frac{m}{n}=0.25$		$\frac{m}{n}=0.3$	
	BP	MOD-CS	BP	MOD-CS	BP	MOD-CS	BP	MOD-CS
CS-VOEC	19.92	22.49	20.88	23.85	21.91	25.51	23.23	27.42
CS-VOC	19.85	22.02	20.68	23.40	21.54	25.17	22.72	27.34
DCVS-VOC: GOP-size=4	24.56	25.47	25.58	26.68	26.32	27.64	27.43	29.23
DCVS-VOC: GOP-size=8	22.34	22.99	23.25	23.99	24.20	24.97	25.66	26.18

6. CONCLUSION

We proposed a novel CS based video object coding (CS-VOC) technique having a simple encoder in which the sensing mechanism is applied directly on the segmented moving objects using a CS matrix. Encoder was kept simple (requiring only $O(mn)$ computations) by performing motion estimation the decoder rather than at the encoder. At the decoder, the object motion is first estimated so that a CS reconstruction algorithm can efficiently recover the sparse motion-compensated video object error. Simulation results show CS-VOC performs on par with the state-of-the-art CS-VOEC. We proposed DCVS-VOC in which object segmentation is done only for key frames. DCVS-VOC gives better reconstruction performance compared to CS-VOC and CS-VOEC. For improving the reconstruction performances further, we applied MOD-CS in the place of traditional reconstruction algorithm (BP) making effective use of the mask information available.

REFERENCES

1. Hakeem, A., Shafique, K., Shah, M., "An Object based Video Coding Framework for Video Sequences Obtained From Static Cameras," MULTIMEDIA '05, Proceedings of the 13th annual ACM international conference on Multimedia, Nov. 2005.
2. Donoho, D., "Compressed sensing," IEEE Trans. Inform. Theory, vol. 52, no. 4, pp. 1289-1306, Apr. 2006.
3. Huang, H., Makur, A. and Venkatraman, D. "Video object error coding method based on compressive sensing," Control, Automation, Robotics and Vision, 2008. ICARCV 2008. 10th International Conference on , vol., no., pp.1287-1291, Dec. 2008.
4. Vaswani, N. and Lu, W., "Modified-cs: Modifying compressive sensing for problems with partially known support," IEEE Trans. on Sig. Proc., vol. 58(9), pp. 4595-4607, Sept. 2010.
5. Do, T.T., Chen, Y., Nguyen, D.T., Nguyen, N., Gan, L. and Tran, T.D. "Distributed compressed video sensing," Image Processing (ICIP), 2009. 16th IEEE International Conference on , vol., no., pp.1393-1396, 7-10 Nov. 2009.