

Residual Reconstruction for Block-Based Compressed Sensing of Video

Sungkwang Mun and James E. Fowler

Department of Electrical and Computer Engineering
Geosystems Research Institute (GRI)
Mississippi State University, Starkville, Mississippi, USA

Abstract

A simple block-based compressed-sensing reconstruction for still images is adapted to video. Incorporating reconstruction from a residual arising from motion estimation and compensation, the proposed technique alternatively reconstructs frames of the video sequence and their corresponding motion fields in an iterative fashion. Experimental results reveal that the proposed technique achieves significantly higher quality than a straightforward reconstruction that applies a still-image reconstruction independently frame by frame; a 3D reconstruction that exploits temporal correlation between frames merely in the form of a motion-agnostic 3D transform; and a similar, yet non-iterative, motion-compensated residual reconstruction.

Introduction

Recent years have seen significant interest in the paradigm of compressed sensing (CS) (e.g., [1]) which permits, under certain conditions, signals to be sampled at sub-Nyquist rates via linear projection onto a random basis while still enabling exact reconstruction of the original signal. As applied to 2D images, however, CS faces several challenges including a computationally expensive reconstruction process and huge memory required to store the random sampling operator. For video, these problems can be even further exacerbated due to the increased dimensionality of the data. Recently, several fast algorithms have been developed for CS reconstruction of generic 1D signals, while the sampling-operator memory challenge was addressed for still images in [2] by using block-based sampling. Additionally in [2], projection-driven Landweber iterations were proposed to accomplish fast reconstruction while simultaneously imposing smoothing with the goal of improving the reconstructed-image quality by eliminating blocking artifacts. Subsequent investigations in [3] coupled recent directional transforms with statistically estimated thresholding within the framework introduced in [2]. The term block-based CS (BCS) sampling with smooth projected Landweber (SPL) reconstruction was used to describe the approach, and experimental results in [3] revealed that BCS-SPL reconstruction usually offers at least the same quality of recovery as does other state-of-the-art approaches with greatly increased execution speed.

In this paper, we adapt the BCS-SPL reconstruction framework for use with video. CS sampling of video would ideally be global in the sense that CS measurements

This material is based upon work supported by the National Science Foundation under Grant No. CCF-0915307.

would span the entire spatial and temporal extent of a video sequence; however, such global CS sampling of video is largely considered impractical to implement in a real device [4]. As a consequence, we focus on the case in which each video frame is acquired independently with still-image-based CS sampling, for instance, with successive applications of a single-pixel camera as in [5]. Upon signal recovery, however, the temporal redundancy between successive frames is exploited to improve the reconstruction quality over that of an independent frame-by-frame recovery. In this effort, we use the general philosophy of motion-estimation (ME) and motion-compensation (MC) that has been widely deployed for several decades in video-compression algorithms and standards. The centerpiece of our approach is the CS reconstruction from a residual between the current frame and its motion-compensated prediction as calculated in the domain of the random CS projections, and we refer to the resulting algorithm as motion-compensated BCS-SPL (MC-BCS-SPL). Experimental results exhibit a substantial performance improvement for MC-BCS-SPL as compared to the straightforward reconstruction in the form of applying BCS-SPL independently frame-by-frame without capitalizing on frame-to-frame motion. Significant gains are also observed when MC-BCS-SPL is compared to a non-iterative ME/MC-based CS reconstruction of video [6].

Background

CS for a Single Still Image

Suppose that we want to recover real-valued signal \mathbf{x} with length N from M samples, $M \ll N$; i.e., we want to recover \mathbf{x} from $\mathbf{y} = \Phi\mathbf{x}$, where \mathbf{y} has length M , and Φ is an $M \times N$ measurement matrix. Because the number of unknowns is much larger than observations, recovering every $\mathbf{x} \in \mathbb{R}^N$ from its corresponding \mathbf{y} is impossible in general; however, if \mathbf{x} is sufficiently *sparse*, exact recovery is possible—this is the fundamental tenet of CS theory; see, e.g., [1], for a more complete overview. The usual choice for Φ is a random matrix. We define the subsampling rate, or *subrate*, to be $S = M/N$.

Quite often, the requisite sparsity will exist with respect to some transform Ψ . In this case, the key to CS recovery is the production of a sparse set of significant transform coefficients, $\tilde{\mathbf{x}} = \Psi\mathbf{x}$, and the ideal recovery procedure searches for the $\tilde{\mathbf{x}}$ with the smallest ℓ_0 norm consistent with the observed \mathbf{y} . However, this ℓ_0 optimization being NP-complete, several alternative solution procedures have been proposed. Often, a convex relaxation is applied to the ℓ_0 problem resulting in an ℓ_1 optimization which can be solved via basis pursuits. Although such ℓ_1 -based recovery can be implemented effectively with linear programming, its computational complexity is often too high for use with higher-dimensional signals such as images and video.

Several reduced-complexity approaches for CS reconstruction have been proposed recently. For example, iterative-thresholding reconstructions start from some initial approximation and form the approximation at each iteration using a specific instance of a projected Landweber (PL) algorithm [7]. PL-based CS reconstruction typically provides reduced computational complexity as compared to pursuits-based reconstruction. Additionally, and perhaps more importantly, the PL formulation offers the

possibility of easily incorporating additional optimization criteria.

The straightforward application of CS to 2D imagery is simply to treat an $N \times N$ image as an N^2 -dimensional vector and apply existing 1D CS strategies directly to the vectorized problem. However, this straightforward framework faces several challenges including a computationally expensive reconstruction process and huge memory required to store the random sampling operator. One potential solution to these issues is to partition the image into smaller blocks. In [2], a such paradigm was proposed wherein the sampling of an image is driven by random matrices applied on a block-by-block basis, while the reconstruction is a variant of PL reconstruction that incorporates a smoothing operation intended to reducing blocking artifacts. In [3], the overall technique was called BCS-SPL.

In BCS-SPL, an image is divided into $B \times B$ blocks and sampled using an appropriately-sized measurement matrix. That is, suppose that \mathbf{x}_j is a vector representing, in raster-scan fashion, block j of input image \mathbf{x} . The corresponding \mathbf{y}_j is then $\mathbf{y}_j = \Phi_B \mathbf{x}_j$, where Φ_B is an $M_B \times B^2$ orthonormal measurement matrix such that the subrate is $S = M_B/B^2$. In BCS-SPL, Wiener filtering is incorporated into the basic PL framework in order to remove blocking artifacts. In essence, this operation imposes smoothness in addition to the sparsity inherent to PL. The specific implementation we use here was initially described in [3] and is presented in Fig. 1(a).

The BCS-SPL framework of [2, 3] is quite flexible thanks to its rather simple implementation. It is straightforward to incorporate sophisticated transforms and thresholding, as well as additional constraints into the process. For example, highly directional transforms with statistically estimated thresholding was investigated for BCS-SPL in [3]. Results there demonstrate that BCS-SPL with contourlets or complex-valued wavelets usually matches or exceeds the quality of total-variation reconstruction (e.g., [8]) whose gradient-based operation also promotes smoothing but runs several orders of magnitude slower than SPL algorithm. As a consequence, we adopt the BCS-SPL framework as the core image reconstruction used in our subsequent investigations of CS for video.

Extension of Still-Image CS to Video

The straightforward application of CS to video would involve, like for images, the vectorization of a 3D group of frames into a single 1D vector. However, the computation and memory issues associated with this approach for 2D images are exacerbated with the increased dimensionality coming from video. Additionally, a global sampling simultaneously across the spatial and temporal extent of a group of frames is likely to be impractical [4]. Consequently, we focus on the situation in which video frames are sampled independently in a 2D fashion, e.g., by applying a suitable image-acquisition sampler in a frame-by-frame fashion as in [5]. Again, to cut computation and memory, we focus on a block-based image sampling applied frame by frame. We now consider several strategies for the CS reconstruction of video frames sampled in this manner; these include both 2D and 3D versions of BCS-SPL (2D-BCS-SPL and 3D-BCS-SPL, respectively).

Given a frame-by-frame acquisition, the most straightforward reconstruction would be to reconstruct the individual frames independently using BCS-SPL. However, such

a 2D-BCS-SPL reconstruction ignores the fact that consecutive video frames are usually highly correlated. Nonetheless, 2D-BCS-SPL serves as a baseline against which to compare other techniques that do attempt to exploit such temporal correlation.

Another straightforward method of CS reconstruction for video that does make an effort to exploit temporal correlation is to treat the video frames as a 3D “volume” and reconstruct the 3D video volume by applying a suitable CS reconstruction algorithm using a 3D transform; this was done, for example, in [5]. Although such 3D reconstruction could be applied across a video volume as a whole (as in [5]), for computation and memory issues (which are likely to be substantial given the increased dimensionality), we consider a 3D version of BCS-SPL essentially extending BCS-SPL reconstruction for a single still image into three dimensions, with the video volume being partitioned into smaller, 3D cubes. Specifically, let us consider a group of P consecutive frames from a video sequence which we call a group of pictures (GOP). Again, for video sampling, the individual video frames are acquired using a 2D block-based sampling applied frame by frame. For reconstruction, the GOP is partitioned into $B \times B \times P$ cubes; i.e., 3D blocks which have a spatial size of $B \times B$ and a temporal size of P . The BCS-SPL reconstruction then uses a 3D block-based transform operator; e.g., a 3D discrete cosine transform (DCT) of size $B \times B \times P$. We refer to the resulting technique as 3D-BCS-SPL reconstruction.

It was argued in [5] that, in effect, CS reconstruction using a 3D transform—such as the 3D-BCS-SPL discussed here—attempts to exploit the frame-to-frame correlation that exists within a video GOP through the joint sparsity that occurs in the 3D transform. This is in contrast to the frame-by-frame 2D-BCS-SPL reconstruction which makes no attempt at all to exploit frame-to-frame correlation. However, neither of these two approaches takes full advantage of the temporal correlation that exists in the video sequence due to the frame-to-frame motion of objects. For this, some form of ME/MC must to be incorporated into the reconstruction; we propose such a ME/MC-based reconstruction next.

MC-BCS-SPL

In video compression, knowledge of object motion is used to make interframe predictions which are, in turn, used to drive an efficient coding of prediction residuals. As a result, ME/MC is a widely used and crucial component to traditional video-compression systems. Here, we incorporate the ME/MC framework into the reconstruction process of BCS-SPL. This implies that ME/MC resides at the reconstruction, or “decoder,” side of a BCS-SPL system for video, rather than at the sensing, or “encoder,” side as is the case in traditional video compression. The proposed BCS-SPL for video uses the same simple sampling as was used for 2D images—block-based random CS measurements are applied frame by frame. The resulting motion-compensated version of BCS-SPL, MC-BCS-SPL, consists of two main components: residual reconstruction, in which BCS-SPL is applied to an MC residual; and forward/backward MC, in which multiple reconstruction passes are performed in multiple directions across the GOP. These components are explored below.

Residual Reconstruction

Suppose we have two consecutive frames, the current frame, \mathbf{x} , and the reference frame, \mathbf{x}_{ref} . We assume that the reference frame has been previously reconstructed, while we have only random measurements, \mathbf{y} , of the current frame using block-based random sampling. As a first step, we could reconstruct an approximation, $\hat{\mathbf{x}}$, to current image \mathbf{x} by simply applying the BCS-SPL image reconstruction to \mathbf{y} . However, such intraframe reconstruction does not capitalize on our knowledge of reference frame \mathbf{x}_{ref} . Yet, at this point, we have an approximation to the current frame as well as the reference frame—we can perform ME on these two frames to estimate a motion field describing motion of objects between the two frames. Such ME could be performed in a variety of ways; for simplicity we consider full-search, block-based ME as is commonly used in video compression. Consequently, the approximation to the current frame, $\hat{\mathbf{x}}$, is partitioned into blocks whose motion from the reference frame is indicated by a field of motion vectors. This permits the production of a motion-compensated frame, $\hat{\mathbf{x}}_{\text{mc}}$, that forms a prediction of $\hat{\mathbf{x}}$ and thus also the still-unknown true current frame \mathbf{x} . Using this prediction, we can form a “projection-domain” residual by simply applying the known block-based random sampling operator to the motion-compensated frame; i.e., for each block j :

$$\mathbf{y}_{r_j} = \mathbf{y}_j - \Phi_B \hat{\mathbf{x}}_{\text{mc}_j}. \quad (1)$$

It is clear that \mathbf{y}_{r_j} is the random projection of the residual, \mathbf{x}_{r_j} , between our motion-compensated prediction $\hat{\mathbf{x}}_{\text{mc}_j}$ and the original and still-unknown block \mathbf{x}_j ; i.e.,

$$\mathbf{y}_{r_j} = \mathbf{y}_j - \Phi_B \hat{\mathbf{x}}_{\text{mc}_j} = \Phi_B (\mathbf{x}_j - \hat{\mathbf{x}}_{\text{mc}_j}) = \Phi_B \mathbf{x}_{r_j}. \quad (2)$$

If the MC process is reasonably accurate, the residual frame \mathbf{x}_r should be much sparser than the original image \mathbf{x} ; CS reconstruction should thereby be much more effective at recovering the residual \mathbf{x}_r from \mathbf{y}_r than it is at recovering \mathbf{x} from \mathbf{y} . Let $\hat{\mathbf{x}}_r$ be such a BCS-SPL recovery from \mathbf{y}_r ; consequently, we can form a new approximation to \mathbf{x} as

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{mc}} + \hat{\mathbf{x}}_r. \quad (3)$$

We now have a new approximation to the current frame that is likely to be of better quality than the initial approximation that we created from a direct BCS-SPL reconstruction from \mathbf{y} . Consequently, we should be able to produce a more accurate motion-vector field from ME applied to this new $\hat{\mathbf{x}}$ and \mathbf{x}_{ref} , and further enhancement can be expected by iteratively repeating the above process. The resulting MC-BCS-SPL algorithm is summarized in Fig. 1(b).

Forward/Backward MC

In the previous two sections, we considered the reconstruction of two consecutive frames of a video sequence. We now turn our attention to the more realistic problem of applying MC-BCS-SPL to multiple frames. Specifically, let us consider a GOP of P consecutive frames from a video sequence, consisting of one “key frame” (the first frame) followed by $P - 1$ “non-key frames.”

```

function  $\mathbf{x} = \text{BCS-SPL}(\mathbf{y}, \Phi_B, \Psi)$ 
  for each block  $j$ 
     $\mathbf{x}_j^{(0)} = \Phi_B^T \mathbf{y}_j$ 
   $i = 0$ 
  do
     $\hat{\mathbf{x}}^{(i)} = \text{Wiener}(\mathbf{x}^{(i)})$ 
    for each block  $j$ 
       $\hat{\mathbf{x}}_j^{(i)} = \hat{\mathbf{x}}_j^{(i)} + \Phi_B^T (\mathbf{y}_j - \Phi_B \hat{\mathbf{x}}_j^{(i)})$ 
     $\check{\mathbf{x}}^{(i)} = \Psi \hat{\mathbf{x}}^{(i)}$ 
     $\check{\mathbf{x}}^{(i)} = \text{Threshold}(\check{\mathbf{x}}^{(i)})$ 
     $\bar{\mathbf{x}}^{(i)} = \Psi^{-1} \check{\mathbf{x}}^{(i)}$ 
    for each block  $j$ 
       $\mathbf{x}_j^{(i+1)} = \bar{\mathbf{x}}_j^{(i)} + \Phi_B^T (\mathbf{y}_j - \Phi_B \bar{\mathbf{x}}_j^{(i)})$ 
     $D^{(i+1)} = \|\mathbf{x}^{(i+1)} - \hat{\mathbf{x}}^{(i)}\|_2$ 
     $i = i + 1$ 
  until  $|D^{(i)} - D^{(i-1)}| < 10^{-2}$ 
   $\mathbf{x} = \mathbf{x}^{(i)}$ 

```

(a)

```

function  $\hat{\mathbf{x}} = \text{MC-BCS-SPL}(\mathbf{y}, \Phi_B, \Psi, \mathbf{x}_{\text{ref}})$ 
   $\hat{\mathbf{x}} = \text{BCS-SPL}(\mathbf{y}, \Phi_B, \Psi)$ 
   $i = 0$ 
  while  $i < \text{MAX\_ITERATION}$ 
     $\hat{\mathbf{x}}_{\text{mc}} = \text{MotionCompensation}(\hat{\mathbf{x}}, \mathbf{x}_{\text{ref}})$ 
    for each block  $j$ 
       $\mathbf{y}_{\text{mc}j} = \Phi_B \hat{\mathbf{x}}_{\text{mc}j}$ 
     $\mathbf{y}_r = \mathbf{y} - \mathbf{y}_{\text{mc}}$ 
     $\hat{\mathbf{x}}_r = \text{BCS-SPL}(\mathbf{y}_r, \Phi_B, \Psi)$ 
     $\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{mc}} + \hat{\mathbf{x}}_r$ 
     $i = i + 1$ 
  end while

```

(b)

Figure 1: (a) BCS-SPL reconstruction of a 2D image; $\text{Wiener}(\cdot)$ is pixel-wise adaptive Wiener filtering using a neighborhood of 3×3 , while $\text{Threshold}(\cdot)$ is a thresholding process. (b) MC-BCS-SPL reconstruction of the current frame from a single reference frame.

In the MC-BCS-SPL setting, we have block-based random measurements of each of the frames of the GOP; i.e., $\mathbf{y}_p = \Phi_p \mathbf{x}_p$ for $0 \leq p \leq P - 1$, where p is the frame number. Φ_p is the random block-based sampling operator for frame p ; we assume that this sampling operator has subrate $S_p = M_p/N$ in that it reduces the N -dimensional \mathbf{x}_p image signal to an M_p -dimensional measurement signal \mathbf{y}_p . We will focus our attention on two cases: 1) all frames are sampled with the same subrate, and 2) the key frame, \mathbf{x}_0 , is sampled at a relatively high subrate, while all the non-key frames are at an identical lower subrate.

Direct BCS-SPL reconstruction from \mathbf{y}_0 produces $\hat{\mathbf{x}}_0$, a reconstruction of the key frame which is then used as the reference frame for the MC-BCS-SPL reconstruction of the second frame of the GOP. This MC-BCS-SPL process produces $\hat{\mathbf{x}}_1$ from \mathbf{y}_1 and $\hat{\mathbf{x}}_0$; $\hat{\mathbf{x}}_1$ is then in turn used as the reference frame in MC-BCS-SPL reconstruction of $\hat{\mathbf{x}}_2$ from \mathbf{y}_2 . This process continues through in this manner to reconstruct the remaining frames of the GOP.

We note, however, that, since each frame is used as a reference for a subsequent frame, we expect a successively lower reconstruction quality as we progress through the non-key frames. That is, we expect reconstruction quality to deteriorate with increasing frame number p , such that the quality of $\hat{\mathbf{x}}_{P-1}$, the reconstructed last frame of the GOP, will be much less than that of $\hat{\mathbf{x}}_0$, the first frame of the GOP. We propose to combat this quality deterioration by performing another iteration of MC-BCS-SPL reconstruction on the GOP, but this time running in the reverse temporal direction, from frame $P - 1$ to frame 0.

Specifically, we assume that there exists another GOP following the current GOP,

with \mathbf{x}_P being the first frame of that GOP. Direct BCS-SPL reconstruction from \mathbf{y}_P will produce a reconstruction $\hat{\mathbf{x}}_P$; this reconstruction can be used as the reference frame for the MC-BCS-SPL reconstruction of \mathbf{y}_{P-1} to produce $\hat{\mathbf{x}}_{P-1}$. We could continue MC-BCS-SPL reconstruction in this reverse temporal direction until we reached the start of the current GOP.

Running MC-BCS-SPL reconstruction on the current GOP in both the forward and backward temporal directions would yield two reconstructions for each non-key frame of the GOP. Our experimental observations have revealed that MC-BCS-SPL in the forward temporal direction yields the higher-quality reconstruction for the first half of the GOP, while MC-BCS-SPL in the backward temporal direction yields the better quality for the last half of the GOP. Assume that the GOP consists of an even number of frames. We thus propose using forward MC-BCS-SPL to reconstruct frames $\hat{\mathbf{x}}_p$ for $1 \leq p \leq \frac{P}{2} - 1$, the first half of the non-key frames, and backward MC-BCS-SPL (starting from the key frame of the next GOP) to reconstruct frames $\hat{\mathbf{x}}_p$ for $\frac{P}{2} + 1 \leq p \leq P - 1$.

Experimental Results

We now examine the performance of MC-BCS-SPL reconstruction on common video sequences. All of the video sequences are subject to block-based random projection applied frame by frame; i.e., by partitioning each frame p into $B \times B$ blocks and applying to each block a random sampling operator of size $M_{B_p} \times B^2$ such that the subrate for frame p is $S_p = M_{B_p}/B^2$. Unless otherwise stated, we use a block size of $B = 16$ and a GOP size of $P = 8$ frames. MC-BCS-SPL employs BCS-SPL for reconstruction of individual MC residual frames; for this BCS-SPL reconstruction, we use a 2D DCT as the transform operator and hard thresholding, for simplicity. For the ME/MC process in MC-BCS-SPL, we use full-search ME with quarter-pixel accuracy and a search window of ± 15 pixels. MAX_ITERATIONS in Fig. 1(b) is 5.

We compare MC-BCS-SPL to two alternatives, namely 2D-BCS-SPL and 3D-BCS-SPL. The most straightforward reconstruction is the “intraframe” 2D-BCS-SPL (we use a block-based $B \times B$ 2D DCT as the transform basis). Additionally, we consider 3D-BCS-SPL with the 3D transform operator being a $B \times B \times P$ 3D DCT. Additionally, we note that 3D-BCS-SPL uses a GOP size of $P = 4$ frames which we have observed empirically to yield results superior to a GOP size of $P = 8$.

As a primary measure of reconstruction quality, we calculate PSNR averaged over all the frames under consideration. For MC-BCS-SPL, various subrates are employed for the key frames as well as the non-key frames; thus, we have two subrates in use: the key-frame subrate (S_K), and the non-key-frame subrate (S_{NK}). First, we consider the case wherein all frames are subsampled at the same subrate, i.e., “equal” subsampling with $S_K = S_{NK}$. Then, we consider the case wherein the key frames have an increased subrate with respect to the non-key frames, i.e., $S_K > S_{NK}$. In this latter situation, PSNR results are averaged over only the non-key frames. A summary of the results from both cases is presented in Table 1.

Since the key frames constitute only a small number of the total frames in a sequence, and they serve somewhat as “anchors” to the forward/backward ME process

Table 1: Average PSNR in dB for several video sequences

S_{NK}	2D	3D	MC-BCS-SPL	
	BCS-SPL	BCS-SPL	$S_{\text{K}} = S_{\text{NK}}$	$S_{\text{K}} = 0.7$
Coastguard				
0.1	22.69	22.76	23.06	25.29
0.2	24.70	24.76	25.78	27.94
0.3	26.37	26.45	28.29	30.15
0.4	27.99	27.95	30.88	32.30
0.5	29.60	29.57	33.58	34.42
Foreman				
0.1	25.99	26.47	27.58	30.03
0.2	28.83	29.18	31.01	33.14
0.3	31.21	31.36	33.50	35.55
0.4	33.18	33.29	35.76	37.54
0.5	35.07	35.13	38.05	39.44
Hall Monitor				
0.1	22.55	22.78	22.79	29.61
0.2	24.78	25.22	25.31	32.22
0.3	26.89	27.26	27.40	33.44
0.4	28.81	29.06	29.35	34.25
0.5	30.73	30.96	31.34	34.85
Mother and Daughter				
0.1	30.14	30.37	30.74	36.90
0.2	33.08	33.34	34.01	39.66
0.3	35.57	35.62	36.65	41.20
0.4	37.46	37.48	38.88	42.41
0.5	39.34	39.34	41.04	43.47

of reconstruction in a GOP, it is reasonable to consider the situation in which key frames are given a higher subrate than the non-key frames such that they are reconstructed with high quality. Thus, we perform a battery of experiments which measure the PSNR when the subrate for key frames is increased beyond that of the non-key frames. Fig. 2 depicts the performance of MC-BCS-SPL when $S_{\text{K}} = 0.7$ while S_{NK} varies between 0.1 and 0.5. In these graphs, MC-BCS-SPL with $S_{\text{K}} = 0.7$ is compared to MC-BCS-SPL with $S_{\text{K}} = S_{\text{NK}}$, i.e., the “equal subsampling” paradigm as well as to 3D-BCS-SPL (2D-BCS-SPL is omitted due to similar performance to 3D-BCS-SPL). As expected, the higher-quality reconstruction of the key frames results in a significant performance improvement for MC-BCS-SPL. In particular, remarkable gains are exhibited for the “Hall Monitor” and “Mother and Daughter” sequences which did not show a large gain over 2D-BCS-SPL in the previous, equal-subsampling experiments. In an informal observation of computational complexity, MC-BCS-SPL reconstruction runs at a speed of about 40 seconds per frame (spf). On the other hand, 3D-BCS-SPL runs at 10spf, and 2D-BCS-SPL at 5spf. These times were measured on a 3.2-GHz quad-core processor.

We now compare the performance of MC-BCS-SPL to that of the technique of [6]. In this technique, the non-key frames are then reconstructed from both block-based and frame-based CS measurements; specifically, the block-based sampling of a

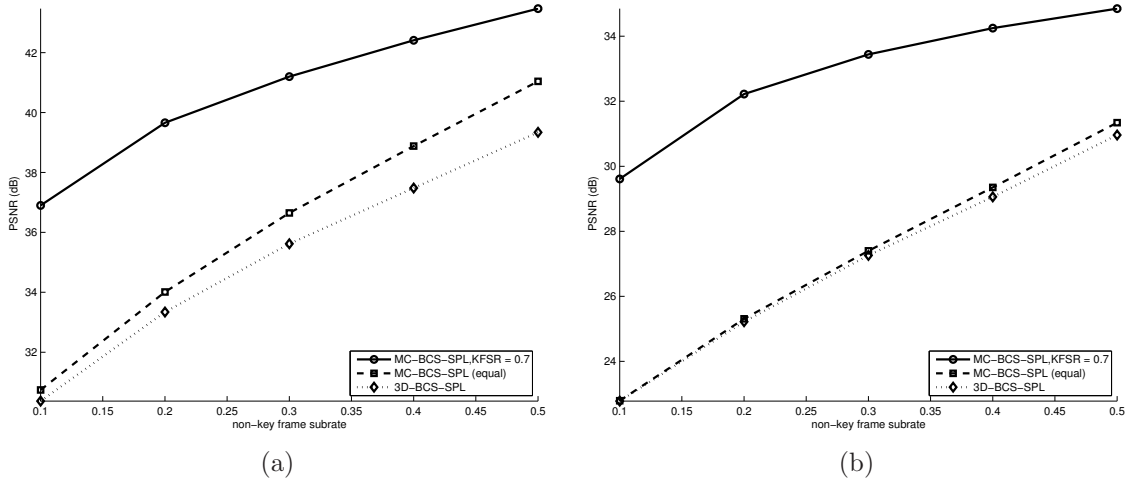


Figure 2: Performance of MC-BCS-SPL for $S_K = 0.7$. PSNR is averaged over all non-key frames of the sequence. “MC-BCS-SPL (equal)” refers to MC-BCS-SPL with equal subsampling, $S_K = S_{NK}$. (a) “Mother and Daughter.” (b) “Hall Monitor.”

non-key frame is used to form a block-by-block multihypothesis motion-compensated prediction of the non-key frame. For each non-key frame block, an ℓ_1 minimization selects which blocks in a spatial window in the neighboring key frames participate in the multihypothesis prediction of the current block. The motion-compensated prediction of the non-key frame is then used in a residual reconstruction based on the other, full-frame measurements of the non-key frame; this residual reconstruction is similar to that in MC-BCS-SPL, except that it is not iterative, and the sampling operator is full-frame rather than block-based. To compare to this technique, we depart from the experimental setup used above and instead match the test conditions used in [6]. Specifically, we change the GOP length to 4 frames with the first frame (i.e., the “key frame”) left uncoded such that the remaining 3 non-key frames are reconstructed from perfect-quality key frames as was done in the experimental results in [6]. The PSNR performance as a function of non-key frame subrate is presented in Fig. 3. We see that MC-BCS-SPL offers a PSNR gain of around 0.5 dB for the “Foreman” sequence, while a gain of 0.75–2 dB is exhibited for the “Coastguard” sequence.

Conclusions

CS reconstruction of video is a challenging task because computation and memory burdens grow quickly as the size of problem increases due to the inherent multidimensional nature of the data. As a consequence, in this paper, we focused on reconstruction driven by block-based processing, employing the BCS-SPL algorithm—a simple but powerful block-based still-image reconstruction—as the foundation necessary for video reconstruction. While independent frame-by-frame reconstruction is fast and straightforward, improved reconstruction quality resulted from exploiting the frame-to-frame motion of objects. Incorporating reconstruction from a ME/MC-based residual, the MC-BCS-SPL technique we proposed here alternatively reconstructs frames

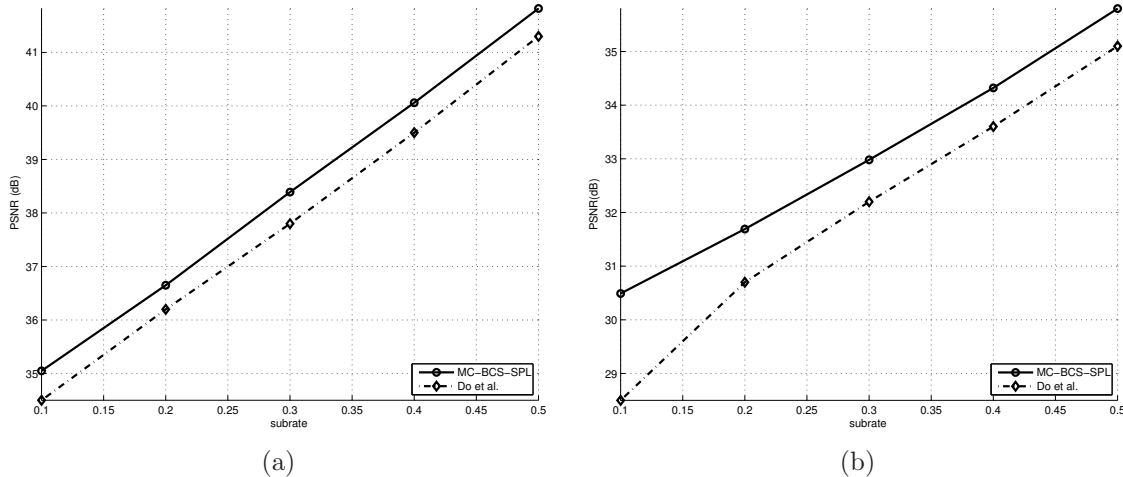


Figure 3: Comparison to the technique due to Do *et al.* [6]; GOP length = 4 frames with perfect-quality key frame; performance measured over non-key frames only. (a) “Foreman.” (b) “Coastguard.”

of the video sequence and their corresponding motion fields, using one to improve the quality of the other in an iterative fashion. The resulting reconstruction achieved significantly higher quality than did either frame-by-frame BCS-SPL reconstruction or a 3D variant of BCS-SPL that features temporal decorrelation merely in the form of a motion-agnostic 3D transform. Furthermore, significant performance gain was also observed over the alternative, non-iterative approach to CS video reconstruction of [6].

References

- [1] E. J. Candès and M. B. Wakin, “An introduction to compressive sampling,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 21–30, March 2008.
- [2] L. Gan, “Block compressed sensing of natural images,” in *Proceedings of the International Conference on Digital Signal Processing*, Cardiff, UK, July 2007, pp. 403–406.
- [3] S. Mun and J. E. Fowler, “Block compressed sensing of images using directional transforms,” in *Proceedings of the International Conference on Image Processing*, Cairo, Egypt, November 2009, pp. 3021–3024.
- [4] M. F. Duarte, “Compressive sensing for signal ensembles,” Ph.D. thesis, Rice University, 2009.
- [5] M. B. Wakin, J. N. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly, and R. G. Baraniuk, “Compressive imaging for video representation and coding,” in *Proceedings of the Picture Coding Symposium*, Beijing, China, April 2006.
- [6] T. T. Do, Y. Chen, D. T. Nguyen, N. Nguyen, L. Gan, and T. D. Tran, “Distributed compressed video sensing,” in *Proceedings of the International Conference on Image Processing*, Cairo, Egypt, November 2009, pp. 1393–1396.
- [7] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging*. Bristol, UK: Institute of Physics Publishing, 1998.
- [8] E. Candès, J. Romberg, and T. Tao, “Stable signal recovery from incomplete and inaccurate measurements,” *Communications on Pure and Applied Mathematics*, vol. 59, no. 8, pp. 1207–1223, August 2006.