Foundations and Trends® in
Signal Processing
Vol. 4, No. 4 (2010) 297–416

DOI: 10.1561/2000000033

Block-Based Compressed Sensing of Images and Video

By James E. Fowler, Sungkwang Mun
and Eric W. Tramel

Contents

Foundations and Trends® in
Signal Processing
Vol. 4, No. 4 (2010) 297–416

DOI: 10.1561/2000000033

Block-Based Compressed Sensing of Images and Video

James E. Fowler[1], **Sungkwang Mun**[2] **and Eric W. Tramel**[3]

[1] *Department of Electrical & Computer Engineering, Geosystems Research Institute, Mississippi State University, Mississippi State, MS 39762, USA, fowler@ece.msstate.edu*
[2] *Department of Electrical & Computer Engineering, Geosystems Research Institute, Mississippi State University, Mississippi State, MS 39762, USA, sm655@msstate.edu*
[3] *Department of Electrical & Computer Engineering, Geosystems Research Institute, Mississippi State University, Mississippi State, MS 39762, USA, ewt16@msstate.edu*

Abstract

A number of techniques for the compressed sensing of imagery are surveyed. Various imaging media are considered, including still images, motion video, as well as multiview image sets and multiview video. A particular emphasis is placed on block-based compressed sensing due to its advantages in terms of both lightweight reconstruction complexity as well as a reduced memory burden for the random-projection measurement operator. For multiple-image scenarios, including video and multiview imagery, motion and disparity compensation is employed to exploit frame-to-frame redundancies due to object motion and parallax, resulting in residual frames which are more compressible

and thus more easily reconstructed from compressed-sensing measurements. Extensive experimental comparisons evaluate various prominent reconstruction algorithms for still-image, motion-video, and multiview scenarios in terms of both reconstruction quality as well as computational complexity.

Acronyms

3D-BCS-SPL three-dimensional BCS-SPL. 347, 355, 360–362, 365

BCS block-based compressed sensing. 301, 318–321, 324–329, 331, 334, 335, 338, 339, 343, 345, 348, 373, 377, 395

BCS-GPSR-DCT block-based compressed sensing with GPSR reconstruction using a block-based DCT sparsity basis. 324, 328

BCS-GPSR-DWT block-based compressed sensing with GPSR reconstruction using a whole-image DWT sparsity basis. 324, 328

BCS-SPL block-based compressed sensing with smooth projected Landweber reconstruction. 321–329, 331, 332, 334–336, 338, 339, 346–350, 352–355, 357, 360–362, 365–368, 370, 377, 378, 380, 381, 383, 388, 390–392, 394, 395

BCS-SPL-CT block-based compressed sensing with smooth projected Landweber reconstruction using a CT sparsity basis. 327, 328

BCS-SPL-DCT block-based compressed sensing with smooth projected Landweber reconstruction using a block-based DCT sparsity basis. 326–328

BCS-SPL-DDWT block-based compressed sensing with smooth projected Landweber reconstruction using a DDWT sparsity basis. 327, 328
BCS-SPL-DWT block-based compressed sensing with smooth projected Landweber reconstruction using a whole-image DWT sparsity basis. 326–328
BCS-TV block-based compressed sensing with total-variation reconstruction. 320, 325, 327–329
BP basis pursuit. 306, 307
BPDN basis-pursuit denoising. 307, 317

CoSaMP compressive sampling matching pursuit. 307, 333
CS compressed sensing. 300–318, 322, 324, 326, 328, 329, 331, 333–335, 338–342, 344–348, 355–360, 366, 368–379, 381–386, 388–392, 395, 398–401
CT contourlet transform. 322, 323, 326

DC disparity compensation. 390–394, 397–399
DC-BCS-SPL disparity-compensated BCS-SPL. 390–395, 398
DCT discrete cosine transform. 316, 324, 326, 335, 346, 360, 361
DDWT dual-tree discrete wavelet transform. 323, 326, 335, 381, 395
DE disparity estimation. 390–395, 397–399
DISCOS distributed compressed video sensing. 358, 371, 376
DLP digital-light-processing. 311
DMD digital micromirror device. 311, 312, 319, 342, 345
DWT discrete wavelet transform. 316, 320, 322–324, 326, 329–331, 333–335, 359, 377

GOP group of pictures. 346, 347, 351–355, 357–363, 366, 381, 388
GPSR gradient projection for sparse reconstruction. 307, 324–326, 333, 335, 357, 378

IST iterative splitting and thresholding. 307

JL Johnson-Lindenstrauss. 372

k–*t* **FOCUSS** focal underdetermined system solver in *k*–*t* space. 358, 359, 368, 371, 372, 381, 382, 386–388

LASSO least absolute shrinkage and selection operator. 306
LDS linear dynamical system. 356
LIMAT lifting-based invertible motion adaptive transform. 359
LSQ least-squares. 374, 375

MARX model-based adaptive recovery of compressive sensing. 333, 334
MC motion compensation. 341, 347–349, 351, 355–360, 363, 366, 367, 369, 371, 372, 374, 377, 379, 381, 388, 390, 391, 393, 394, 398, 399
MC-BCS-SPL motion-compensated BCS-SPL. 347, 349–368, 371, 372, 381, 384, 386, 387, 390, 391
MCTF motion-compensated temporal filtering. 359, 360
ME motion estimation. 341, 347–349, 355–360, 363, 366, 367, 369, 371, 372, 377, 379, 381, 388, 390, 391, 393, 394, 397–399
MEMS microelectromechanical systems. 342
MH multihypothesis. 369, 374–381, 386, 388
MH-BCS-SPL multihypothesis BCS-SPL. 381, 384–388
MRI magnetic resonance imaging. 300, 355–359, 382, 385, 388
MS multiscale. 329, 383
MS-BCS-SPL multiscale, wavelet-domain block-based compressed sensing with smooth projected Landweber reconstruction. 330–332, 334–336, 338, 383, 385–387
MS-GPSR multiscale GPSR. 333, 335–338

OMP orthogonal matching pursuits. 307, 333

PAR piecewise autoregressive. 333
PL projected Landweber. 308, 321, 326
PSNR peak signal-to-noise ratio. 324–328, 335, 336, 361–366, 378, 380, 382–386, 388, 396, 397

SALSA split augmented Lagrangian shrinkage algorithm. 334–338
SAMP sparsity adaptive matching pursuits. 333
SH single-hypothesis. 369, 373, 378, 380
SpaRSA sparse reconstruction via separable approximation. 307
SPL smoothed projected Landweber. 321, 323

SRM structurally random matrix. 305, 312, 315, 317, 318, 329, 334, 335, 337–339, 383

StOMP stagewise orthogonal matching pursuit. 333

TSW-CS tree-structured wavelet compressed sensing. 333, 334, 338

TV total variation. 318, 320, 325, 327, 328, 334–339, 383–387

1

Introduction

The *sampling theorem* is arguably the best known component of the theoretical foundations of the signal-processing and communications fields; its importance is paramount in that it underlies all modern signal-acquisition, sampling, sensing, and analog-to-digital conversion devices. Although introduced to the signal-processing and communications communities by Shannon in 1949 [109], the sampling theorem can be traced to earlier work by telegraphers and mathematicians (see, e.g., [82]). In essence, the sampling theorem states that, if a signal contains no frequencies higher than bandlimit W, then it can be perfectly reconstructed from samples acquired at a rate of at least $2W$. This latter quantity, commonly known as the *Nyquist rate*, thus represents the slowest rate at which sampling of any bandlimited signal can be acquired and still permit perfect reconstruction.

However, this traditional sampling theory is founded on relatively minimal prior knowledge on the signal being sampled — i.e., its bandlimit W. While traditional sampling theory has the advantage of applying to *any* signal satisfying this bandlimit constraint, we are commonly interested in more restricted classes of signals, i.e., those that are known to possess much more structure, and thus fewer degrees of freedom, than dictated by the signal bandlimit [8]. A well-known example is that of bandpass signals in which the signal is present over only a limited band of frequencies — under such bandpass structure, bandpass sampling (e.g., [129]) can acquire the signal with a sampling rate slower than $2W$. More recent literature has witnessed an explosion of interest in sensing that exploits structured prior knowledge in the general form of *sparsity*, meaning that signals can be represented by only a few coefficients in some transform basis. Like bandpass sampling, exploitation of such sparse structure within signals can effectively permit sampling at rates well below $2W$.

Central to much of this recent work is the paradigm of compressed sensing (CS)[1] (e.g., [18, 22, 38]) which permits relatively few measurements of the signal to be acquired in a linear fashion while still permitting exact reconstruction via a relatively complex and nonlinear recovery process. While much CS literature is rather generic in that it is not tied to any specific class of signal beyond a general assumption of sparsity, there has been significant interest in CS specifically tailored to imaging applications. Indeed, recent work in the CS field has seen proposals for not only sensor devices but also reconstruction algorithms designed specifically for a variety of imagery signals.

The goal of this monograph is to overview some of these methods. A primary focus is an examination of the state of the art in CS reconstruction for various imaging modalities, including still images, motion video, and multiview imagery. Throughout, we focus on photographic imagery which is acquired in the spatial domain of the image, a paradigm which is ubiquitous throughout image-processing applications. This stands in contrast to a significant portion of existing CS literature that has targeted, with substantial success, specific medical-imaging applications — in particular, magnetic resonance imaging (MRI), which is acquired directly in a Fourier-transform space. The potential for CS to significantly expedite MRI acquisition is relatively well established and already well covered tutorially in the literature (e.g., [83, 84]). On the other hand, CS for photographic imagery

---

[1] Also known as *compressive sampling* or *compressive sensing.*

acquired in the spatial domain is a comparatively emerging area and, thus, the topic of the present monograph.

An additional focus of this monograph is on CS reconstruction as applied on image blocks. In such block-based compressed sensing (BCS), an image is partitioned into small non-overlapping blocks which are acquired independently but reconstructed jointly. BCS is motivated primarily for reasons of reduced computational complexity and memory burdens. These can become impractically large for the CS of images and video as a result of the increased dimensionality (i.e., 2D and 3D) of such signals.

We note also that our discussion is not intended to serve as an in-depth tutorial on the theory or mathematics of CS; rather, there exist several excellent overviews on this subject (e.g., [7, 20, 22]). Instead, our coverage of CS theory here will be brief, while the specifics of the application of BCS to natural imagery will consume the bulk of the discussion.

The remainder of the monograph is organized as follows. Section 2 briefly overviews CS theory, including acquisition and reconstruction processes. Section 3 then considers the CS of a single still image, focusing on a variety of techniques to reconstruct such images from random CS measurements. Section 4 extends these concepts to the CS of video with an emphasis on reconstruction from motion-compensated residuals, and then Section 5 adds multihypothesis prediction to such motion-based CS reconstruction. Section 6 finally considers the CS of multiview images and video in which a scene is imaged from several viewpoints simultaneously. We end the monograph by making several concluding remarks.

2

Compressed Sensing

2.1 An Overview of CS Theory

In brief, compressed sensing (CS) [18, 22, 38] is an emerging mathematical paradigm which permits, under certain conditions, linear projection of a signal into a dimension much lower than that of the original signal while allowing exact recovery of the signal from the projections. More specifically, suppose that we want to recover real-valued signal $\mathbf{x}$ with length N from M measurements such that $M \ll N$. In other words, we want to recover $\mathbf{x} \in \Re^N$ from

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}, \tag{2.1}$$

where $\mathbf{y}$ has length M, and $\mathbf{\Phi}$ is an $M \times N$ measurement matrix with subsampling rate, or *subrate*, being $S = M/N$. Because the number of unknowns is much larger than the number of observations, recovering every $\mathbf{x} \in \Re^N$ from its corresponding $\mathbf{y} \in \Re^M$ is impossible in general; however, if $\mathbf{x}$ is known to be sufficiently *sparse* in some domain, then *exact* recovery of $\mathbf{x}$ is possible — this is the fundamental tenet of CS theory.

In CS, sparsity is the key aspect that enables recovery of $\mathbf{x}$ from $\mathbf{y}$. Such sparsity can be with respect to some transform $\mathbf{\Psi}$ such that, when the transform is applied to $\mathbf{x}$, i.e.,

$$\check{\mathbf{x}} = \mathbf{\Psi}\mathbf{x}, \tag{2.2}$$

only K coefficients in the set of transform coefficients $\check{\mathbf{x}}$ are nonzero. We say then that $\mathbf{x}$ is K-sparse and that $\mathbf{\Psi}$ is the *sparsity basis* for $\mathbf{x}$.

We note that real-world signals — particularly the images and the video that we focus on in this text — are rarely, if ever, truly sparse in any basis. Rather, such real-world signals are often only *compressible* in the sense that, if $\check{x}_n$ are the coefficients of $\check{\mathbf{x}} = \mathbf{\Psi}\mathbf{x}$ sorted in order of decreasing magnitude such that

$$|\check{x}_n| \geq |\check{x}_{n+1}|, \tag{2.3}$$

then

$$|\check{x}_n| < Rn^{-r}, \tag{2.4}$$

where $r \geq 1$ and $R < \infty$ (see, e.g., [10, 17, 18, 20]). Define $\check{\mathbf{x}}_K$ as the set of partial coefficients derived from $\check{\mathbf{x}}$ by keeping the K largest coefficients and setting the rest to zero, and $\mathbf{x}_K = \mathbf{\Psi}^{-1}\check{\mathbf{x}}_K$. Then, if $\mathbf{x}$ is compressible in the sense of (2.4), $\mathbf{x}_K$ is close to $\mathbf{x}$ in the sense of

$$\|\mathbf{x} - \mathbf{x}_K\|_2 \leq C_r R K^{-r}, \tag{2.5}$$

where C_r is a constant depending only on r [18]. When the coefficient magnitudes have a power-law decay as in (2.4), CS theory holds that the resulting compressible $\mathbf{x}$ can be recovered approximately from $\mathbf{y} = \mathbf{\Phi}\mathbf{x}$. Specifically, the recovered signal, $\hat{\mathbf{x}}$, will be close to $\mathbf{x}_K$ in the sense of

$$\|\hat{\mathbf{x}} - \mathbf{x}\|_2 \leq C\frac{\|\mathbf{x} - \mathbf{x}_K\|_1}{\sqrt{K}} \tag{2.6}$$

for some well-behaved constant C (see, e.g, [17, 20]).

In the case of such approximate recovery of compressible signals, the subrate $S = M/N$ plays the crucial role in determining the quality of the reconstruction — a larger subrate will enable a sparse approximation containing more nonzero coefficients to be recovered, leading to a reconstruction that is closer to the original signal. As a consequence, in experimental results to come later, we will evaluate performance of image- and video-reconstruction strategies by examining an image-distortion measure over a range of subrate values, noting that perfect, distortionless, recovery is achieved for such compressible signals only when $S = 1$.

2.2 Approaches to CS-Based Signal Acquisition

In the CS framework, the signal-acquisition or sensing device acquires neither $\mathbf{x}$ nor its coefficients $\check{\mathbf{x}}$ directly. Instead, the sensing device measures linear projections of $\mathbf{x}$ onto the measurement basis $\mathbf{\Phi}$, thereby acquiring the signal directly in a reduced dimensionality. This CS measurement process is represented mathematically as (2.1).

The trick behind CS is, of course, to do this dimensionality-reducing signal acquisition in a reversible manner, and this imposes conditions on the measurement and sparsity bases. Specifically, CS theory dictates that it is sufficient that the sparsity basis and the measurement basis be mutually *incoherent* in the sense that the measurement basis $\mathbf{\Phi}$ cannot sparsely represent the columns of the sparsity basis $\mathbf{\Psi}$. The usual choice for the measurement basis $\mathbf{\Phi}$ is a random matrix, since it can be shown that a random basis will be incoherent with *any* sparsity basis with very high probability. In this sense, a random measurement basis works universally well for any signal regardless of the domain in which its sparsity exists. Random matrices that exhibit such incoherence with any sparsity basis include matrices populated with independent and identically distributed Gaussian or ±1 Rademacher[1] random variables [6]. An orthonormalized random matrix (formed by, e.g., orthonormalizing the rows of the above Gaussian matrix) also provides incoherence [22].

We observe that the most straightforward approach to representing the measurement matrix $\mathbf{\Phi}$ is simply an $M \times N$ dense array of values (e.g., floating point for Gaussian matrices, binary for Rademacher matrices); in fact, $\mathbf{\Phi}$ has been implemented this way in many applications and in much of the CS literature. However, if the dimensionality of $\mathbf{x}$ is large (as is often the case when multidimensional signals like images and video are considered), the memory required to store $\mathbf{\Phi}$ in both the sensing and reconstruction devices may be impractically

---

[1] A Rademacher distribution assigns probability $\frac{1}{2}$ to the values ± 1.

large; additionally, a large $\mathbf{\Phi}$ may result in a huge computation burden for the reconstruction process which typically involves a large number of multiplications with $\mathbf{\Phi}$. However, the use of a structurally random matrix (SRM) (e.g., [36, 54]) can significantly mitigate these issues. In essence, an SRM provides a signal-sensing process (operator $\mathbf{\Phi}$) consisting of a random permutation, a simple and computationally efficient transform (such as a block cosine or Hadamard transform), and a random subsampling process. Each of these components can be generated procedurally without requiring the storage of a large, dense matrix; additionally, an SRM can be applied to the input signal with little computation or memory. As a consequence, many CS-reconstruction implementations use SRMs in practice, and it is largely anticipated that large-dimensional CS signal-acquisition devices will as well.

Finally, we argue that CS makes the most sense in situations in which the acquisition of each measurement (i.e., each component of vector $\mathbf{y}$ in (2.1)) comes with some substantial associated cost. For example, in medical applications, each measurement may represent the exposure of the patient to a certain amount of radiation — clearly, signal-acquisition schemes that minimize the total radiation exposure are desirable. On the other hand, in many imaging applications, such as those we consider later, each measurement may represent a discrete sensing element — the fewer the elements used, the lower the overall device cost, particularly for wavelengths outside of the visible spectrum for which a single sensor element may be so costly such that a full array of millions of sensors may be prohibitively expensive.

Under this “measurements-are-costly” paradigm, the potential advantages of CS arise from a signal-sensing process in which $\mathbf{x}$ is acquired and *simultaneously* reduced in dimension in the form of $\mathbf{y}$ directly within the hardware of the sensing device. In this case, the matrix-vector multiplication of the measurement process, $\mathbf{\Phi x}$, is performed implicitly within the sensor rather than calculated explicitly. In this sense, CS-based signal acquisition offers computation-free dimensionality reduction at the sensor side of the system. In such a sensing device, the full-dimensional signal $\mathbf{x}$ does not exist at any point within the sensor, having never been sensed or acquired in its full dimensionality.

2.3 Approaches to CS Reconstruction

Given CS measurements $\mathbf{y}$ produced via (2.1) with measurement matrix $\boldsymbol{\Phi}$, the CS reconstruction problem is to find $\hat{\mathbf{x}}$ such that $\mathbf{y} = \boldsymbol{\Phi}\hat{\mathbf{x}}$ (either exactly or approximately) and such that the coefficients $\check{\mathbf{x}} = \boldsymbol{\Psi}\hat{\mathbf{x}}$ are sparse with respect to sparsity basis $\boldsymbol{\Psi}$. The most straightforward formulation of the reconstruction problem searches for the $\check{\mathbf{x}}$ with the smallest ℓ_0 norm[2] consistent with the observed $\mathbf{y}$; i.e.,

$$\min_{\check{\mathbf{x}}} \|\check{\mathbf{x}}\|_0, \quad \text{such that } \mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\Psi}^{-1}\check{\mathbf{x}}, \tag{2.7}$$

where $\boldsymbol{\Psi}^{-1}$ is the inverse sparsity transform. For the $\check{\mathbf{x}}$ solving (2.7), the final reconstruction is then

$$\hat{\mathbf{x}} = \boldsymbol{\Psi}^{-1}\check{\mathbf{x}}. \tag{2.8}$$

However, this ℓ_0 optimization is NP-hard (e.g., [21]) and thus computationally infeasible for all but the smallest of problems. As a consequence, there have been a large number of alternative optimizations proposed in recent literature. Perhaps the most prominent of these is basis pursuit (BP) [25] which applies a convex relaxation to the ℓ_0 problem resulting in an ℓ_1 optimization,

$$\min_{\check{\mathbf{x}}} \|\check{\mathbf{x}}\|_1, \quad \text{such that } \mathbf{y} = \boldsymbol{\Phi}\boldsymbol{\Psi}^{-1}\check{\mathbf{x}}. \tag{2.9}$$

Often, it is assumed that the CS measurements are acquired with some noise; i.e., (2.1) becomes

$$\mathbf{y} = \boldsymbol{\Phi}\mathbf{x} + \mathbf{n}, \tag{2.10}$$

where $\mathbf{n}$ is some vector of noise. In this case, one can relax the equality constraint in the ℓ_1 formulation of (2.9) to yield[3]

$$\min_{\check{\mathbf{x}}} \|\check{\mathbf{x}}\|_1, \quad \text{such that } \|\mathbf{y} - \boldsymbol{\Phi}\boldsymbol{\Psi}^{-1}\check{\mathbf{x}}\|_2^2 \leq \epsilon \tag{2.11}$$

---

[2] The ℓ_0 "norm," $\|\check{\mathbf{x}}\|_0$, is not truly a norm but is a pseudonorm; it is merely the number of nonzero coefficients in $\check{\mathbf{x}}$.

[3] The well-known least absolute shrinkage and selection operator (LASSO) [116] has a form similar to (2.11) with the ℓ_2 term as the minimization objective and ℓ_1 term as the constraint.

for some tolerance $\epsilon > 0$. This constrained optimization is closely related to the unconstrained Lagrangian formulation,

$$\min_{\check{\mathbf{x}}} \|\check{\mathbf{x}}\|_1 + \lambda \|\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}^{-1}\check{\mathbf{x}}\|_2^2, \tag{2.12}$$

known as basis-pursuit denoising (BPDN) [25], wherein the Lagrangian multiplier λ balances the ℓ_1-driven sparsity against the ℓ_2-based measure of distortion.

A variety of strategies have been proposed to solve the (2.9), (2.11), and (2.12) optimizations; see [124] for a survey. BP (2.9) and BPDN (2.12) can be solved effectively with linear and second-order-cone programs, respectively; the well-known ℓ_1-MAGIC[4] software package takes this approach, for example. However, such convex-programming methods have computational complexity that is often large for higher-dimensional signals such as images and video. As an alternative, gradient-descent methods solve BPDN (2.12) and tend to be faster in practice than corresponding interior-point solutions. Gradient-based algorithms include iterative splitting and thresholding (IST) [31], sparse reconstruction via separable approximation (SpaRSA) [135], and gradient projection for sparse reconstruction (GPSR) [47].

A number of greedy algorithms have also been proposed for the CS reconstruction problem. These include matching pursuits [87], orthogonal matching pursuits (OMP) [123], and compressive sampling matching pursuit (CoSaMP) [91]. In practice, such greedy algorithms significantly reduce computational complexity as compared to convex-programming approaches, albeit typically at the cost of lower reconstruction quality [124].

Iterative thresholding (e.g., [14, 15, 31, 62]) is a body of algorithms which serves as an alternative to the greedy pursuits class of CS reconstruction. Iterative-thresholding algorithms form $\check{\mathbf{x}}$ by successively projecting and thresholding; for example, the reconstruction in [62] starts from some initial approximation $\check{\mathbf{x}}^{(0)}$ and forms the approximation at

---

[4] `http://www.l1-magic.org`.

iteration $i+1$ as

$$\check{\check{\mathbf{x}}}^{(i)} = \check{\mathbf{x}}^{(i)} + \frac{1}{\gamma}\mathbf{\Psi}\mathbf{\Phi}^{\mathrm{T}}(\mathbf{y} - \mathbf{\Phi}\mathbf{\Psi}^{-1}\check{\mathbf{x}}^{(i)}), \tag{2.13}$$

$$\check{\mathbf{x}}^{(i+1)} = \begin{cases} \check{\check{\mathbf{x}}}^{(i)}, & |\check{\check{\mathbf{x}}}^{(i)}| \geq \tau^{(i)}, \\ 0, & \text{else}. \end{cases} \tag{2.14}$$

Here, γ is a scaling factor, while $\tau^{(i)}$ is a threshold set appropriately at each iteration. For γ, [62] proposes the largest eigenvalue of $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi}$; in this case, if $\mathbf{\Phi}$ is an orthonormal matrix, $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} = \mathbf{I}$, and $\gamma = 1$. It is straightforward to see that this procedure is a specific instance of a projected Landweber (PL) algorithm [79, 12]; similar approaches for CS reconstruction include [14, 15]. Like the greedy algorithms of the pursuits class, PL-based CS reconstruction also provides reduced computational complexity as compared to convex-programming-based reconstruction. Additionally, the PL formulation offers the possibility of easily incorporating additional optimization criteria. For example, later in this monograph, we will overview an image-reconstruction technique that incorporates Wiener filtering into the PL iteration to search for a CS reconstruction simultaneously achieving sparsity and smoothness.

2.4 CS versus Source Coding

In the field of information theory as established by Shannon [108, 110], *source coding* can be considered to be “the conversion of arbitrary signals into an efficient digital representation” [55]; this digital representation is often called a *bitstream*. Since, in most cases, it is desired that the resulting bitstream use as few bits as possible, source coding has often been referred to as *data compression*. The performance of such data compression is measured in terms of a *bitrate* — i.e., the number of bits contained in the bitstream per original source sample. The compression may result in an exact representation of the original source, in which case the compression is *lossless*; otherwise, it is *lossy*. In the case of lossy compression, performance is also measured in terms of the quality of the representation using some fidelity measure, typically called the *distortion*. Thus, in information theory, the rate-distortion performance characterizes the compression in the form of a tradeoff between bitrate and distortion.

It is important to note that CS does not necessarily imply compression in the information-theoretic, rate-distortion sense. That is, the expression "compressed" in the CS name more correctly refers to a process of dimensionality reduction rather than "compression" in the form of source coding as this term is commonly construed in the information-theory community. Similarly, "compressibility" in the CS domain is strictly in the sense of (2.4), and we emphasize that our use of this term in the remainder of our discussion will be specifically limited to this context.

3

Block-Based Compressed Sensing for Still Images

3.1 CS Acquisition of Still Images

There has been significant interest in compressed sensing (CS) in systems that acquire and process 2D still images. Widespread use of digital cameras has led to increasing demands for higher spatial resolution, lower power consumption, and lower overall device costs. In many existing digital cameras, images are acquired with several million sensor elements. CS offers a compelling alternative to this traditional image-acquisition paradigm — instead of sampling in high resolution, CS offers the possibility of directly acquiring the image in a reduced dimensionality. With this dimensionality reduction taking place implicitly within the hardware of the sensing device, it is speculated that CS may eventually yield camera architectures that are significantly cheaper and that consume less power, both as a result of using a number of discrete sensing elements that is greatly reduced as compared to the full sensor array. Such cameras may then be able to accommodate spectral wavelengths (e.g., infrared) for which a single sensing element is so costly that a multi-mega-sensor array is prohibitive. Cameras might also be tailored to wavelengths impossible to implement in conventional CCD and CMOS imagers [115].

3.1.1 The Single-Pixel Camera

Perhaps the most well-known architecture for the CS acquisition of still images is the so-called “single-pixel camera” [42, 115, 131, 132] developed at Rice University. This camera architecture is illustrated in Figure 3.1. In short, the single-pixel camera uses a digital micromirror device (DMD) to optically perform inner products in the CS measurement process.

Specifically, the DMD — normally an output device used in digital-light-processing (DLP) applications — forms a pixel array of electrostatically actuated mirrors that can each be oriented in one of two states (+12° or −12° from horizontal). When coupled with an analog photosensor, the DMD is used as an image sensor such that the image in question is focused onto the DMD and partially reflected toward the photosensor. By orienting the photosensor such that one of the DMD mirror states reflect light toward the photosensor while the other mirror state reflects away, the DMD, in effect, forms an inner product between the image being acquired and the binary pattern present on the DMD

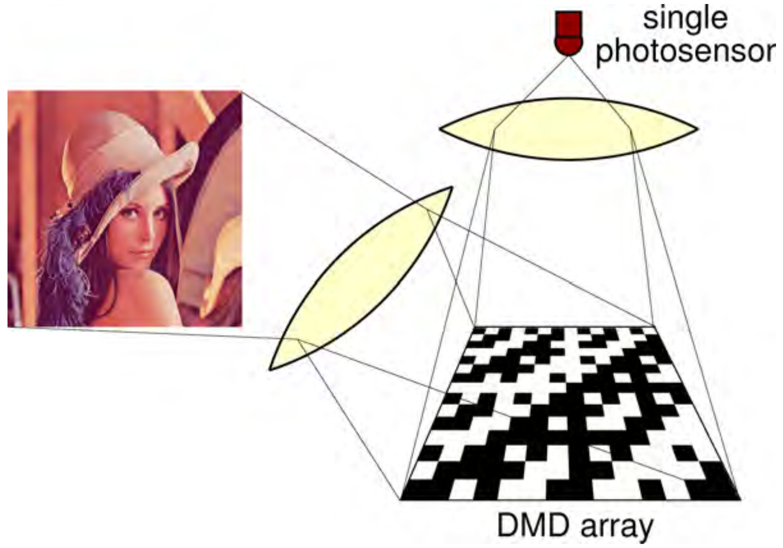


Fig. 3.1 The single-pixel camera from [115, 131, 132, 42] for the CS acquisition of a still image.

array. That is, the photosensor senses the analog sum of all the light reflected to it from the DMD, thereby outputting a continuous-valued analog measurement for each pattern on the DMD.

The resulting analog voltage reading of the photosensor provides a single (scalar) CS measurement value, $y_m \in \Re$. By repeating this process M times with multiple pseudorandom DMD patterns, a vector, $\mathbf{y} = \begin{bmatrix} y_1 & \cdots & y_M \end{bmatrix} \in \Re^M$, of M CS measurements is acquired. It is straightforward for the binary DMD patterns to effectuate a pseudorandom Rademacher ±1 measurement basis by taking an additional measurement with all the mirrors reflecting toward the sensor and then subtracting this mean measurement, $\bar{y}$, from the other measurements:

$$2\mathbf{y} - \bar{y} \cdot \mathbf{1} \to \mathbf{y}, \tag{3.1}$$

where $\mathbf{1}$ is an $M \times 1$ vector of all ones [42, 115]. Alternatively, by appropriately duty-cycling the mirrors, a Gaussian measurement matrix can be obtained [42, 115]. As mentioned previously, such Rademacher and Gaussian measurement bases are incoherent with any sparsity basis with high probability.

Sampling in the single-pixel camera may also be driven by a structurally random matrix (SRM) employing a block Hadamard transform as suggested in [54]. In this latter case, the DMD mirror patterns do not need to be explicitly stored in the sensing device; rather, they are generated procedurally "on the fly" as needed by the measurement process, thereby significantly reducing memory requirements on-board the sensing device as compared to dense Rademacher or Gaussian matrices, particularly when the image size is large.

The CS-based single-pixel camera has the potential to provide substantial benefits for hardware-constrained systems but does come at a cost when compared with traditional systems: acquisition time. A CS device requires less memory to store sampled signals and less power to transmit them, as well no computation required to calculate the dimensionality reduction (which is, in effect, accomplished optically), but the total time to acquire a given signal is increased by a factor of M — more, if duty cycling for a Gaussian matrix is needed (note, however, that such duty cycling is not required for a Hadamard-based SRM as in [54]). For the imaging of static scenes, this trade off is nominal, and the ability to use a single sensor instead of an entire array of sensors might outweigh the cost of the additional exposure time required for each measurement. For dynamic scenes, CS sampling is more complicated, and we will return to the issue of CS sampling for video later.

We note also that the single-pixel-camera architecture is perhaps best suited to applications at wavelengths outside of the visible spectrum. Indeed, current CCD and CMOS imagers already in widespread use in digital cameras provide a full array consisting of many millions of sensor elements in the visible spectrum — since such CCD/CMOS devices are extremely cheap, fast, low-powered, and high-quality, it may be difficult to justify the use of a single-pixel camera in visible-light applications. In contrast, however, sensors for extra-visible wavelengths typically entail significantly higher cost — for example, current infrared cameras are at least a thousand-fold more expensive than visible-light cameras at a comparable spatial resolution. On the other hand, the single-pixel camera, which requires only a single sensing element, may have the potential to significantly reduce the hardware costs of image acquisition in infrared and other extra-visible applications.

3.1.2 Quantization

We note that, in our discussion, and as is typical to CS literature, the image-acquisition device (exemplified here by the single-pixel camera of Figure 3.1), is assumed to be implemented entirely in analog hardware. As a consequence, the measurements acquired are considered to be real-valued; i.e., $\mathbf{y} \in \Re^M$. Yet, the reconstruction procedures to follow are necessarily implemented within digital computers. As a consequence, some form of analog-to-digital conversion is implicit between the image-acquisition and image-reconstruction processes. However, the issue of incorporating quantization into the CS paradigm is largely underdeveloped in CS literature to date.

It is tempting to apply a scalar quantizer to the individual components of the measurement vector $\mathbf{y}$ and concatenate the resulting quantizer indices into a bitstream. However, it is well-known (e.g., [48, 58]) that such a process yields abysmally poor performance (in an information-theoretic rate-distortion sense) as compared to traditional lossy source-coding algorithms (e.g., JPEG [66] and JPEG-2000 [69] for images). Several ameliorations (e.g., [28, 70, 133]) have been proposed in recent literature to improve the performance of quantized CS beyond that of straightforward reconstruction from scalar-quantized measurements. Still, these techniques yield rate-distortion performance well below that of existing source-coding algorithms.

As a consequence of the incipient nature of the coupling of quantization and CS, we do not consider the issue of quantization further in our discussion, treating both the original signal as well as its CS measurements conceptually as real-valued, with corresponding experimental implementation in floating-point arithmetic. It is anticipated that, as strategies for incorporating quantization into CS become more mature, adaptation of the algorithms discussed in the following to quantized measurements will be straightforwardly accomplished. Such is the case, for example, of the progressive-quantization methodology proposed in [133] which is employable with any image-reconstruction algorithm.

3.1.3 Other Image-Acquisition Architectures

In addition to the single-pixel camera, several other architectures for the CS acquisition of still imagery have been proposed; an overview of some of these devices is presented in [43]. These alternative architectures include devices which couple a sensor array with explicit calculation of a random projection in analog hardware. For example, [71] implements random convolution (convolution with a random filter followed by downsampling [103]) for CS image acquisition; on the other hand, [102] performs analog vector–matrix multiplication to effectuate CS measurements. Both [71] and [102] have the drawback that they require implementation of a full array of sensing elements (i.e., equal in number to the pixels in the image) and thus do not reduce the cost associated with the number of such sensing elements. Rather, the assumption is that the analog calculation of the dimensionality reduction is cheaper — in terms of device implementation or power consumption — than other means for accomplishing the same (e.g., digital calculation of an image transform). Such an advantage has not yet been clearly demonstrated for such devices to the best of our knowledge, however.

Alternatively, [88] proposes CS image acquisition via a coded aperture and a subsampled focal plane. Like the single-pixel camera, a reduced number of sensing elements is used as compared to the full sensor array, but still significantly more than just a single sensor as in the single-pixel camera. Like the single-pixel camera, coded-aperture image acquisition is likely to be most suited to applications residing outside the visible spectrum since current full-array CCD and CMOS imagers already provide low-cost image acquisition in the visible spectrum. Yet extra-visible wavelengths may require significantly more expensive sensor elements and therefore may benefit from the reduced number of sensor elements required by coded-aperture sensors.

Throughout the remainder of this monograph, we will assume that the single-pixel-camera architecture of Figure 3.1 is employed for CS acquisition of a still image, either through the use of a duty-cycled Gaussian measurement matrix or an SRM-driven sampling process. The primary concern is then how one reconstructs the image from the resulting CS measurements; we explore this issue next.

3.2 Straightforward Reconstruction for Images

A straightforward implementation of CS on 2D images recasts the 2D array-based problem as a 1D vector-based problem, typically using the following procedure:

(1) "Rasterize" $N \times N$ image $\mathbf{X}$ into an N^2-dimensional vector $\mathbf{x}$:

$$\mathbf{x} = \mathrm{Raster}(\mathbf{X}). \tag{3.2}$$

Here, $\mathrm{Raster}(\cdot)$ is a suitable rasterization operator — for example, the concatenation of the N rows of the image together, followed by a transpose, to produce an $N^2 \times 1$ vector.

(2) Apply $M \times N^2$ measurement matrix $\mathbf{\Phi}$:

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}. \tag{3.3}$$

(3) Apply some CS reconstruction algorithm. This vector-based reconstruction will employ $\mathbf{\Phi}$, the measurement matrix, as

well as $\mathbf{\Psi}$, the sparsity transform. In this context, $\mathbf{\Psi}$ is an $N^2 \times N^2$ matrix consisting of the N^2 basis images of some 2D image transform (discrete cosine transform (DCT), discrete wavelet transform (DWT), etc.), arranged such that each basis image has been rasterized into an $N^2 \times 1$ vector and placed columnwise into $\mathbf{\Psi}$. The reconstruction yields $\hat{\mathbf{x}} = \mathbf{\Psi}\check{\mathbf{x}}$:

$$\hat{\mathbf{x}} = \text{CS\_Reconstruction}\left(\mathbf{y}, \mathbf{\Phi}, \mathbf{\Psi}\right). \tag{3.4}$$

(4) Reassemble the image from the vectorized reconstruction:

$$\hat{\mathbf{X}} = \text{Unraster}\left(\hat{\mathbf{x}}\right). \tag{3.5}$$

This rasterized paradigm was used, for example, for reconstruction with the single-pixel-camera system in [115, 131, 132].

We note that the description here of the sparsity basis $\mathbf{\Psi}$ as an $N^2 \times N^2$ matrix is largely conceptual. While it would certainly be possible to represent any 2D image transform such as a DCT or DWT in this fashion, in practical implementation of CS reconstruction, one would almost certainly use some fast computation of both the forward transform $\mathbf{\Psi}\check{\mathbf{x}}$ and the inverse transform $\mathbf{\Psi}^{-1}\mathbf{x}$ within the CS_Reconstruction$(\cdot)$ process. For example, there are a number of fast algorithms for both the DCT (e.g., [46, 85]) and DWT (e.g., [32, 114]). Such algorithms typically offer a number of advantages for the calculation of an image transform as compared to matrix–vector multiplication, which, although conceptually simple, can entail significantly more computation and memory than the fast algorithms.

A primary concern with (3.4) is that, when a generic CS reconstruction is employed, it is somewhat “blind” to the fact that the data being processed represents an image beyond the fact that the sparsity transform $\mathbf{\Psi}$ is a 2D image transform. That is, generic CS reconstruction algorithms treat the problem as any other, simply searching for a sparse solution that is consistent with the observed random measurements. Unfortunately, imposing sparsity alone does not necessarily produce the most visually pleasing reconstructed images. For example, [17] found that the sparse solutions produced by a straightforward CS reconstruction often incurred visually unpleasant, high-frequency oscillations. In essence, the issue stems from the fact that generic CS reconstruction ignores attributes known to be widely possessed by images, such as smoothness. We will examine strategies for incorporating such image attributes into CS reconstruction below.

Another concern is that, from the perspective of practical implementation, the fact that the size of $\mathbf{\Phi}$ is $O(N^4)$ entails that the memory required to store this matrix grows very fast as the number of pixels in the image in question increase. This leads to a huge memory required to store the measurement operator when $\mathbf{\Phi}$ is implemented as a dense matrix (e.g., Gaussian or Rademacher) within the CS sensing process. Additionally, a large $\mathbf{\Phi}$ yields a huge memory and computational burden within the CS reconstruction process as well, since CS reconstruction involves numerous matrix computations with $\mathbf{\Phi}$. As a consequence of these difficulties in both memory and computation, many instances of image CS in prior literature consider images no larger than 256×256 pixels, and some focus on even smaller sizes. As mentioned previously, SRMs (e.g., [36, 54]) can effectively reduce these computation and memory burdens by using a procedurally-generated measurement operator in both the sensing and reconstruction processes. However, an alternative to SRMs is to impose some form of sparse structure (e.g., block diagonality) onto $\mathbf{\Phi}$, a possibility that we explore in detail below.

We now consider several strategies that attempt to ameliorate the difficulties identified above. Specifically, we first consider a reconstruction based on total variation that capitalizes on the fact that the underlying data to be reconstructed represents a natural 2D image. A primary focus of this effort is the promotion of smoothness of the end reconstruction. We then consider CS based on image blocks that effectively focuses on both smoothness of reconstruction as well as fast and memory-efficient implementation of the measurement operator.

3.3 Total-Variation Reconstruction

In [17], basis-pursuit denoising (BPDN) as given in (2.12) is recast in order to promote smoothness in the reconstructed images and to suppress the high-frequency artifacts encountered with the straightforward solution to (2.12). Specifically, it is proposed that, instead of seeking sparsity in the domain of some image transform, (2.12) is reformulated as

$$\min_{\mathbf{X}} \|\mathbf{X}\|_{\mathrm{TV}} + \lambda \|\mathbf{y} - \mathbf{\Phi}\mathbf{x}\|_2, \tag{3.6}$$

where $\mathbf{x}$ is the 1D rasterization of image $\mathbf{X}$. Here, the total variation (TV) of the image is

$$\|\mathbf{X}\|_{\mathrm{TV}} = \sum_{i,j} \sqrt{(x_{i+1,j} - x_{i,j})^2 + (x_{i,j+1} - x_{i,j})^2}, \tag{3.7}$$

where $x_{i,j}$ is the pixel in location (i,j) in the image $\mathbf{X}$. In [17], (3.6) is solved using a second-order-cone program that accommodates the TV-based norm.

In essence, the TV approach to CS reconstruction replaces sparsity in the domain of an image transform with sparsity in the domain of a discretized gradient, implicitly promoting smoothness in the resulting reconstruction. As a result, TV-based reconstruction ameliorates the problem associated with generic CS recovery of image data in that the reconstruction is no longer blind to the fact that the underlying signal is an image. Although the use of SRMs [36, 54] can greatly improve matters, TV reconstruction still tends to be very computationally complex as compared to other reconstruction algorithms. As a consequence, TV-based CS reconstruction of an image tends to be rather slow, perhaps even to the point of being infeasible if the image size is large.

3.4 CS with Blocks in the Spatial Domain

As an alternative to SRMs for alleviating the huge computation and memory burdens associated with a dense measurement matrix $\mathbf{\Phi}$ within both the sensing and reconstruction processes, one can adopt a philosophy long used in image-processing fields when an image is too large to be feasibly processed in its entirety — specifically, break the image into smaller blocks and process the blocks independently. Such an approach was proposed in [53] for block-based compressed sensing (BCS) for 2D images.

In BCS, an image is divided into $B \times B$ non-overlapping blocks and acquired 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

$$\mathbf{y}_j = \mathbf{\Phi}_B \mathbf{x}_j, \tag{3.8}$$

where $\mathbf{\Phi}_B$ is an $M_B \times B^2$ measurement matrix such that the target subrate for the image as a whole is $S = M_B/B^2$. It is straightforward to see that (3.8) applied block-by-block to an image is equivalent to a whole-image measurement matrix $\mathbf{\Phi}$ in (3.3) with a constrained structure; specifically, $\mathbf{\Phi}$ is block diagonal,

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{\Phi}_B & 0 & \cdots & 0 \\ 0 & \mathbf{\Phi}_B & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{\Phi}_B \end{bmatrix}. \tag{3.9}$$

We note that the single-pixel camera discussed above can easily accommodate BCS acquisition by simply driving the DMD array with this block-diagonal $\mathbf{\Phi}$ instead of a dense $\mathbf{\Phi}$ as was done originally in [42, 115, 131, 132].

There are several approaches that one can take to reconstruct an image that has been acquired using BCS. The most straightforward situation is when the sparsity transform $\mathbf{\Psi}$ is also a block-based operator of the same $B \times B$ size, i.e.,

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{\Psi}_B & 0 & \cdots & 0 \\ 0 & \mathbf{\Psi}_B & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{\Psi}_B \end{bmatrix}, \tag{3.10}$$

where $\mathbf{\Psi}_B$ is a $B^2 \times B^2$ matrix for the block-based transform. In this case, (3.4) can simply be applied for each block independently; i.e., for block j,

$$\hat{\mathbf{x}}_j = \text{CS\_Reconstruction}(\mathbf{y}_j, \mathbf{\Phi}_B, \mathbf{\Psi}_B). \tag{3.11}$$

We refer to this independent, block-by-block reconstruction as a "block-independent" reconstruction. In general, block-independent reconstruction will produce severe blocking artifacts and is thus not usually a reasonable solution.

Better results may arise when the block measurement matrix $\mathbf{\Phi}_B$ is placed into the block-diagonal $\mathbf{\Phi}$ as in (3.9) with $\mathbf{\Phi}$ then being used in the whole-image reconstruction of (3.4). We refer to this as "block-diagonal" reconstruction. Block-diagonal reconstruction permits the sparsity achieved in the reconstructed signal to be adjusted across all blocks simultaneously (in the block-independent solution, sparsity is imposed on each block independently). In this case, the sparsity transform $\mathbf{\Psi}$ can take the form of a full-image transform (such as a DWT) rather than being constrained to have a block-diagonal structure as is the case for block-independent reconstruction. Such a full-image transform will serve to ameliorate the blocking artifacts that naturally arise in BCS due to the block-diagonal structure of $\mathbf{\Phi}$. More importantly, since block-based transforms do not decorrelate across image blocks, the compressibility of the image in the sense of (2.4) is likely to be much greater in the domain of a full-image transform such as a DWT, achieving a much higher-fidelity reconstruction than the block-independent approach.

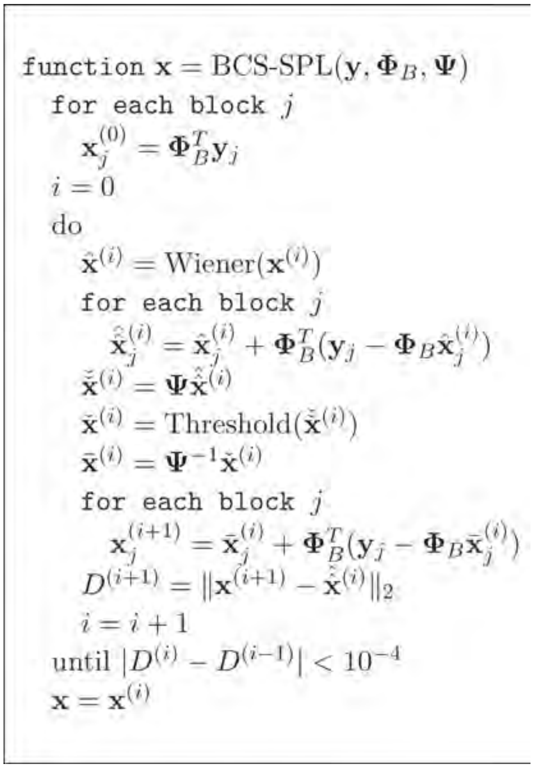
An alternative is to eliminate the sparsity transform $\mathbf{\Psi}$ altogether and opt instead to impose a sparse gradient, i.e., by adopting a TV-based reconstruction. In such a TV-based solution, we apply (3.6) using the block-diagonal $\mathbf{\Phi}$ of (3.9). We refer to this combination of BCS and TV-based reconstruction as "BCS-TV."

In experimental results to follow, we will evaluate the performance of each of these strategies and give examples of the visual quality of reconstructed images. First, however, we overview an alternative to BCS-TV that combines BCS acquisition, fast iterative reconstruction, and explicit smoothing in the form of Wiener filtering with the goal of producing high-quality, visual-pleasing reconstructions like BCS-TV without its heavy computational burden.

3.4.1 The BCS-SPL Algorithm

In [53], BCS was proposed wherein the acquisition of an image is driven by random matrices applied on a block-by-block basis, while the reconstruction is a variant of the projected Landweber (PL) reconstruction of (2.13)–(2.14) that incorporates a smoothing operation intended to reduce blocking artifacts. Since it combines BCS with a smoothed projected Landweber (SPL) reconstruction, in [89], the overall technique was called BCS-SPL.

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. Specifically, a Wiener-filtering step is interleaved with the PL projection of (2.13)–(2.14). The specific implementation we use here was initially described in [89] and is presented in Figure 3.2.



```
function $\mathbf{x} = \text{BCS-SPL}(\mathbf{y}, \mathbf{\Phi}_B, \mathbf{\Psi})$
  for each block $j$
    $\mathbf{x}_j^{(0)} = \mathbf{\Phi}_B^T \mathbf{y}_j$
  $i = 0$
  do
    $\hat{\mathbf{x}}^{(i)} = \text{Wiener}(\mathbf{x}^{(i)})$
    for each block $j$
      $\hat{\hat{\mathbf{x}}}_j^{(i)} = \hat{\mathbf{x}}_j^{(i)} + \mathbf{\Phi}_B^T(\mathbf{y}_j - \mathbf{\Phi}_B \hat{\mathbf{x}}_j^{(i)})$
    $\check{\check{\mathbf{x}}}^{(i)} = \mathbf{\Psi} \hat{\hat{\mathbf{x}}}^{(i)}$
    $\check{\mathbf{x}}^{(i)} = \text{Threshold}(\check{\check{\mathbf{x}}}^{(i)})$
    $\bar{\mathbf{x}}^{(i)} = \mathbf{\Psi}^{-1} \check{\mathbf{x}}^{(i)}$
    for each block $j$
      $\mathbf{x}_j^{(i+1)} = \bar{\mathbf{x}}_j^{(i)} + \mathbf{\Phi}_B^T(\mathbf{y}_j - \mathbf{\Phi}_B \bar{\mathbf{x}}_j^{(i)})$
    $D^{(i+1)} = \|\mathbf{x}^{(i+1)} - \hat{\hat{\mathbf{x}}}^{(i)}\|_2$
    $i = i + 1$
  until $|D^{(i)} - D^{(i-1)}| < 10^{-4}$
  $\mathbf{x} = \mathbf{x}^{(i)}$
```

Fig. 3.2 BCS-SPL reconstruction of a 2D image; Wiener$(\cdot)$ is pixel-wise adaptive Wiener filtering using a neighborhood of 3×3, while Threshold$(\cdot)$ is a thresholding process. Φ_B is assumed to be a random orthonormal matrix such that γ in (2.13) is unity (from [89]).

In [89], to set a proper τ for hard thresholding in BCS-SPL($\cdot$), we employ the so-called interference heuristic [86] which effectively assumes Gaussian marginal statistics for the insignificant transform coefficients (i.e., those coefficients that are zero in a sparse approximation to the original image). Then, the universal threshold of [37] is used. Specifically, in (2.14),

$$\tau^{(i)} = \lambda \sigma^{(i)} \sqrt{2 \log K}, \tag{3.12}$$

where λ is a constant control factor to manage convergence, and K is the number of the transform coefficients. As in [37], $\sigma^{(i)}$ is estimated using a robust median estimator,

$$\sigma^{(i)} = \frac{\mathrm{median}(|\check{\check{\mathbf{x}}}^{(i)}|)}{0.6745}. \tag{3.13}$$

As a final note, we observe that while (3.12) and (3.13) form a convenient heuristic for setting the threshold in the BCS-SPL algorithm, [39] points to theoretical shortcomings with the underlying Gaussian assumption in general iterative-thresholding algorithms for CS.

3.4.2 BCS-SPL with Directional Transforms

The BCS-SPL framework of [53, 89] 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 [89].

Although DWTs are widely used in traditional source coding of images (e.g., the JPEG-2000 standard [69]), DWTs in their traditional critically-sampled form are known to be somewhat deficient in several characteristics, lacking such properties as shift invariance and significant directional selectivity. As a result, there have been several recent proposals made for transforms that feature a much higher degree of directional representation than is obtained with traditional DWTs. Two prominent families of such directional transforms are contourlets and complex-valued DWTs. The contourlet transform (CT) [33] preserves interesting features of the traditional DWT, namely multiresolution and local characteristics of the signal, and, at the expense of spatial redundancy, it better represents the directional features of the image. The CT couples a Laplacian-pyramid decomposition with directional filterbanks, inheriting the redundancy of the Laplacian pyramid (i.e., 4/3). Alternatively, complex-valued wavelet transforms have been proposed to improve upon DWT deficiencies, with the dual-tree discrete wavelet transform (DDWT) [78] becoming a preferred approach due to the ease of its implementation. In the DDWT, real-valued wavelet filters produce the real and imaginary parts of the transform in parallel decomposition trees. The DDWT yields a decomposition with a much higher degree of directionality than that possessed by traditional DWTs; however, since both trees of the DDWT are themselves orthonormal or biorthogonal decompositions, the DDWT taken as a whole is a redundant tight frame. Albeit redundant, both the CT and DDWT have been effectively used in the source coding of images (e.g., [16, 23, 44, 45, 49, 122]); their use in BCS-SPL reconstruction was demonstrated in [89].

Hard thresholding inherently assumes independence between coefficients. However, bivariate shrinkage [107] is better suited to directional transforms in that it exploits statistical dependency between transform coefficients and their respective parent coefficients, yielding performance superior to that of hard thresholding. In [107], a non-Gaussian bivariate distribution was proposed for the current coefficient and its lower-resolution parent coefficient based on an empirical joint histogram of DWT coefficients. However, it is straightforward to apply this process to any transform having a multiple-level decomposition, such as the directional transforms we consider here. Specifically, [89] proposed that, given a specific transform coefficient ξ and its parent coefficient ξ_p in the next coarser scale, the Threshold$(\cdot)$ operator in SPL be the maximum *a posteriori* estimator of ξ,

$$\text{Threshold}(\xi, \lambda) = \frac{\left(\sqrt{\xi^2 + \xi_p^2} - \lambda \frac{\sqrt{3}\sigma^{(i)}}{\sigma_\xi}\right)_+}{\sqrt{\xi^2 + \xi_p^2}} \cdot \xi, \tag{3.14}$$

where $(g)_+ = 0$ for $g < 0$, $(g)_+ = g$ else; $\sigma^{(i)}$ is the median estimator of (3.13); and, again, λ is a convergence-control factor. Here, σ_ξ^2 is the marginal variance of coefficient ξ estimated in a local 3×3

neighborhood surrounding ξ as in [107]. We refer the reader to [107] for full details on the method of bivariate shrinkage and its underlying statistical models.

3.4.3 Experimental Observations on BCS-Based Reconstruction

We now present a battery of experimental evaluations to gauge the performance of the block-based still-image reconstruction techniques just previously discussed. Throughout, we employ several popular grayscale images[1] of size 512×512, and we use BCS on blocks of size 32×32. In all cases, the images are subjected to a BCS measurement process with $\mathbf{\Phi}_B$ in (3.8) being an orthonormalized dense Gaussian matrix. We measure reconstruction performance in terms of a peak signal-to-noise ratio (PSNR) between the reconstructed and original images.

We compare several reconstruction techniques as discussed above. Figures 3.3–3.5 depict typical reconstruction results for the “Lenna” image using a subrate $S = 0.2$; these figures show only a detailed portion from the center of the image. Figure 3.3(a) gives the original image, while Figures 3.3(b)–(d) present several straightforward reconstructions, all based on a generic gradient projection for sparse reconstruction (GPSR) CS reconstruction algorithm [47]. The first, in Figure 3.3(b), comes from a block-independent reconstruction using a block-based DCT as the sparsity transform; in this “block-independent BCS-GPSR-DCT” reconstruction, each 32×32 block is reconstructed independently via (3.11) with $\mathbf{\Psi}_B$ being the block-based DCT. Figure 3.3(c) presents a similar reconstruction, only this time, the reconstruction is block-diagonal. That is, “block-diagonal BCS-GPSR-DCT” instead performs whole-image reconstruction via (3.4) using the corresponding block-diagonal $\mathbf{\Phi}$ and $\mathbf{\Psi}$, the latter again being based on the 32×32 block DCT. Severe blocking artifacts are readily apparent in Figures 3.3(b) and (c). This blocking is ameliorated significantly in Figure 3.3(d) in which a whole-image DWT is used in the reconstruction via (3.4); i.e., in BCS-GPSR-DWT, $\mathbf{\Phi}$ still possesses a

---

[1] The images are available within the BCS-SPL package, `http://www.ece.mstate.edu/~fowler/BCSSPL/`.

(a) Original image

(b) Block-independent BCS-GPSR-DCT, 25.7 dB

(c) Block-diagonal BCS-GPSR-DCT, 25.9,dB

(d) BCS-GPSR-DWT, 26.0 dB

Fig. 3.3 Reconstructions of the 512 × 512 "Lenna" image (shown in detail) for a subrate of $S = 0.2$. BCS measurement uses 32 × 32 blocks.

block-diagonal structure, but $\mathbf{\Psi}$ does not. We note that PSNR figures for the three straightforward GPSR-based reconstructions are fairly similar, being around 26 dB for "Lenna."

Significantly improved PSNR performance, as well as visual quality, results when smoothing is included in the reconstruction process. To wit, Figures 3.4(a)–(c) present example reconstructions for a TV-based reconstruction as well as two BCS-SPL-based reconstructions. Specifically, Figure 3.4(a) illustrates the performance of BCS-TV wherein the TV reconstruction of (3.6) is applied to BCS-acquired imagery, promoting a sparse image gradient and thereby an implicitly smooth reconstruction. We note that we use ℓ_1-MAGIC[2] in the BCS-TV

---

[2] `http://www.l1-magic.org`.

(a) BCS-TV, 30.6 dB (b) BCS-SPL-DCT, 30.3 dB

(c) BCS-SPL-DWT, 30.9 dB

Fig. 3.4 Reconstructions of the 512 × 512 "Lenna" image (shown in detail) for a subrate of $S = 0.2$. BCS measurement uses 32 × 32 blocks.

implementation. On the other hand, Figures 3.4(b) and (c) shows the reconstruction due to BCS-SPL which features explicit smoothing in the form of a Wiener filter incorporated into the iterative PL reconstruction process. Shown are results for both a 32 × 32 block DCT (BCS-SPL-DCT, Figure 3.4(b)) and a full-frame DWT (BCS-SPL-DWT, Figure 3.4(c)). We use the BCS-SPL implementation available from the BCS-SPL website.[3] The results of the smoothing-based reconstructions of Figures 3.4(a)–(c) are similar, with PSNRs around 30.5 dB, some 4 dB higher than the straightforward GPSR-based reconstructions of Figures 3.3(b)–(d).

Figures 3.5(a) and (b) evaluate directional transforms for CS reconstruction, deploying both the CT and DDWT within the BCS-SPL

---

[3] `http://www.ece.mstate.edu/~fowler/BCSSPL/`.

(a) BCS-SPL-CT, 31.0 dB (b) BCS-SPL-DDWT, 31.4 dB

Fig. 3.5 Reconstructions of the 512 × 512 “Lenna” image (shown in detail) for a subrate of $S = 0.2$. BCS measurement uses 32 × 32 blocks.

framework. We refer to the resulting implementations as BCS-SPL-CT and BCS-SPL-DDWT, respectively. The results for the directional transforms are similar, with PSNRs around 31 dB being about 0.5 dB higher than the reconstructions of Figure 3.4.

We note that, in these results, we use bivariate shrinkage (3.14) with $\lambda = 10$, 25, and 25, respectively, for BCS-SPL-CT, BCS-SPL-DDWT, and BCS-SPL-DWT. Lacking parent–child relations, BCS-SPL-DCT uses hard thresholding (3.12) with $\lambda = 6$.

Table 3.1 compares PSNR for several 512 × 512 images at several subrates. We note that, since the quality of reconstruction can vary due to the randomness of the measurement matrix $\mathbf{\Phi}$, all PSNR figures are averaged over five independent trials. The results indicate that BCS-SPL with the directional transforms achieves the best performance at low measurement rates. At higher measurement rates, performance is more varied — BCS-TV is more competitive; however, the directional BCS-SPL techniques usually produce PSNR close to that of the TV-based algorithm.

The experimental results presented here demonstrate that imposing smoothness to the reconstruction — either implicitly via the gradient-based TV approach, or explicitly in the form BCS-SPL’s Wiener filtering — can improve PSNR and visual-quality performance significantly. BCS-SPL is advantageous in that its simple formulation based on Landweber iterations permits it to easily accommodate various sophistications such as directional transforms and bivariate shrinkage.

Table 3.1. BCS-based reconstruction performance as measured in PSNR in dB; figures averaged over five independent trials; images are 512 × 512.

	Subrate				
Algorithm	0.1	0.2	0.3	0.4	0.5
Lenna					
BCS-SPL-DDWT	**28.3**	**31.4**	**33.5**	**35.2**	**36.8**
BCS-SPL-CT	28.2	31.0	33.0	34.7	36.3
BCS-SPL-DWT	27.8	30.9	32.9	34.6	36.2
BCS-SPL-DCT	27.7	30.5	32.5	34.2	35.8
BCS-TV	27.9	30.6	32.6	34.3	35.9
BCS-GPSR-DWT	22.7	26.0	28.1	29.9	31.3
Block-diagonal BCS-GPSR-DCT	22.5	25.9	28.1	29.8	31.2
Block-independent BCS-GPSR-DCT	22.6	25.8	28.0	29.8	31.3
Barbara					
BCS-SPL-DDWT	**22.9**	24.3	**25.9**	**27.5**	29.1
BCS-SPL-CT	22.8	24.3	**25.9**	**27.5**	**29.4**
BCS-SPL-DWT	22.6	23.9	25.2	26.6	28.1
BCS-SPL-DCT	22.8	**24.4**	**25.9**	27.4	29.1
BCS-TV	22.5	23.6	24.6	25.6	26.7
BCS-GPSR-DWT	20.1	22.2	23.6	25.0	26.4
Block-diagonal BCS-GPSR-DCT	20.3	23.4	25.4	27.3	28.8
Block-independent BCS-GPSR-DCT	20.4	23.3	25.5	27.3	28.9
Mandrill					
BCS-SPL-DDWT	**22.9**	24.9	26.7	28.4	30.3
BCS-SPL-CT	**22.9**	**25.0**	**27.0**	**28.9**	**30.9**
BCS-SPL-DWT	22.5	24.3	26.0	27.7	29.4
BCS-SPL-DCT	22.3	24.2	25.9	27.8	29.7
BCS-TV	22.3	24.3	26.1	27.8	29.5
BCS-GPSR-DWT	18.6	19.8	20.9	21.9	23.1
Block-diagonal BCS-GPSR-DCT	18.5	19.8	20.9	22.1	23.2
Block-independent BCS-GPSR-DCT	18.5	19.8	20.9	22.1	23.3
Goldhill					
BCS-SPL-DDWT	**27.0**	28.9	30.5	31.8	33.1
BCS-SPL-CT	26.9	**29.0**	30.5	31.9	33.3
BCS-SPL-DWT	26.7	28.7	30.1	31.5	32.9
BCS-SPL-DCT	26.1	28.3	29.6	31.0	32.6
BCS-TV	26.5	28.9	**30.6**	**32.1**	**33.6**
BCS-GPSR-DWT	23.0	25.4	27.0	28.4	29.6
Block-diagonal BCS-GPSR-DCT	23.1	25.6	27.3	28.6	29.8
Block-independent BCS-GPSR-DCT	22.9	25.5	27.2	28.7	29.9

3.5 CS with Blocks in the Wavelet-Domain

As discussed above, techniques such as TV and BCS-SPL exploit smoothness that is anticipated to be present in the original image signal. However, other image characteristics can also be used to tailor CS reconstruction specifically to image data. For example, a number of recent CS strategies (e.g., [61, 63, 64, 77, 105]) are deployed assuming that the image is both acquired and reconstructed in the domain of a DWT. Such wavelet-domain CS permits known statistical models (e.g., [63, 64, 77]) for wavelet coefficients to be exploited in reconstruction. Additionally, the degree of CS subsampling can be adapted to the wavelet decomposition — often, the baseband is retained in full with no subsampling (e.g., [61, 105]), while the degree of subsampling is increased for successively higher-resolution decomposition levels (e.g., [105]). The subsampling in this case is often referred to as multiscale (MS) after [125].

Previously, we considered the block-based CS paradigm primarily as an alternative to SRMs to mitigate computational burdens by limiting the CS measurement process to relatively small blocks. However, the drawback of BCS is typically a reduced quality of image reconstruction due to the fact that CS measurement generally works better the more global it is. We now consider an amelioration of the reconstruction quality of BCS while retaining its light computational burden and extremely fast execution. Specifically, we overview the technique proposed in [50] for an MS algorithm that deploys BCS-SPL [89] in the domain of a wavelet transform.

3.5.1 Wavelet-Domain BCS

Ideally, the CS measurement operator should be "global" in the sense that the entire signal $\mathbf{x}$ should contribute to each and every measurement taken in producing $\mathbf{y}$ in (2.1). However, a block-diagonal structure as in (3.9) defeats such maximally holistic measurement. As a consequence, BCS-based techniques such as BCS-SPL and BCS-TV can be at a disadvantage in terms of reconstruction quality due to their reliance on a block-based measurement operator. We now consider a modification to the BCS-SPL algorithm as proposed in [50]; this variant of BCS-SPL is designed to improve its reconstruction-quality performance while maintaining its block-based measurement and corresponding fast reconstruction. Specifically, BCS-SPL is deployed within the wavelet domain of the image $\mathbf{x}$ to provide multiscale measurement and reconstruction.

The measurement operator $\mathbf{\Phi}$ for MS-BCS-SPL is split into two components — a multiscale transform $\mathbf{\Omega}$ (e.g., a DWT) and a multiscale block-based measurement process $\mathbf{\Phi}'$ such that $\mathbf{\Phi} = \mathbf{\Phi}'\mathbf{\Omega}$, and (2.1) becomes

$$\mathbf{y} = \mathbf{\Phi}'\mathbf{\Omega}\mathbf{x}. \tag{3.15}$$

Assume that $\mathbf{\Omega}$ produces L levels of wavelet decomposition; thus, $\mathbf{\Phi}'$ consists of L different block-based measurement operators, one for each level. That is, let the DWT of image $\mathbf{x}$ be

$$\tilde{\mathbf{x}} = \mathbf{\Omega}\mathbf{x}. \tag{3.16}$$

Subband s at level l of $\tilde{\mathbf{x}}$ is then divided into $B_l \times B_l$ blocks and measured using an appropriately sized $\mathbf{\Phi}_l$ (note that $l = L$ is the highest-resolution level). That is, suppose $\tilde{\mathbf{x}}_{l,s,j}$ is a vector representing, in raster-scan fashion, block j of subband s at level l, with $s \in \{H, V, D\}$, and $1 \leq l \leq L$. Then,

$$\mathbf{y}_{l,s,j} = \mathbf{\Phi}_l \tilde{\mathbf{x}}_{l,s,j}. \tag{3.17}$$

Since the different levels of wavelet decomposition have different importance to the final image reconstruction quality, we adjust the measurement process so as to yield a different subrate, S_l, at each level l. In all cases, we set the subrate of the DWT baseband to full measurement, i.e., $S_0 = 1$. Then, we let the subrate for level l be

$$S_l = W_l S', \tag{3.18}$$

such that the overall subrate becomes

$$S = \frac{1}{4^L} S_0 + \sum_{l=1}^{L} \frac{3}{4^{L-l+1}} W_l S'. \tag{3.19}$$

Given a target subrate S and a set of level weights W_l, one can easily solve (3.19) for S', yielding a set of level subrates S_l via (3.18). However, this process will typically produce one or more $S_l > 1$. Thus, we modify the solution to enforce $S_l \leq 1$ for all l. Specifically, after finding S' and S_1 via (3.19) and (3.18), we check if $S_1 > 1$. If so, we set $S_1 = 1$, remove

Table 3.2. Wavelet-domain BCS subrates S_l at level l for target overall subrate S for a DWT with $L = 3$ levels. In all cases, the baseband is given full measurement ($S_0 = 1.0$).

	Level subrates, S_l		
S	S_1	S_2	S_3
0.1	1.0000	0.1600	0.0100
0.2	1.0000	0.5867	0.0367
0.3	1.0000	1.0000	0.0667
0.4	1.0000	1.0000	0.2000
0.5	1.0000	1.0000	0.3333

its corresponding term from the sum in (3.19), and then solve

$$S = \frac{1}{4^L} S_0 + \frac{3}{4^L} S_1 + \sum_{l=2}^{L} \frac{3}{4^{L-l+1}} W_l S' \tag{3.20}$$

for S', again using (3.18) to redetermine S_l for $l = 2, \ldots, L$. We repeat this process as needed to ensure that all $S_l \leq 1$.

For the experimental results to follow later, we use level weights,

$$W_l = 16^{L-l+1}, \tag{3.21}$$

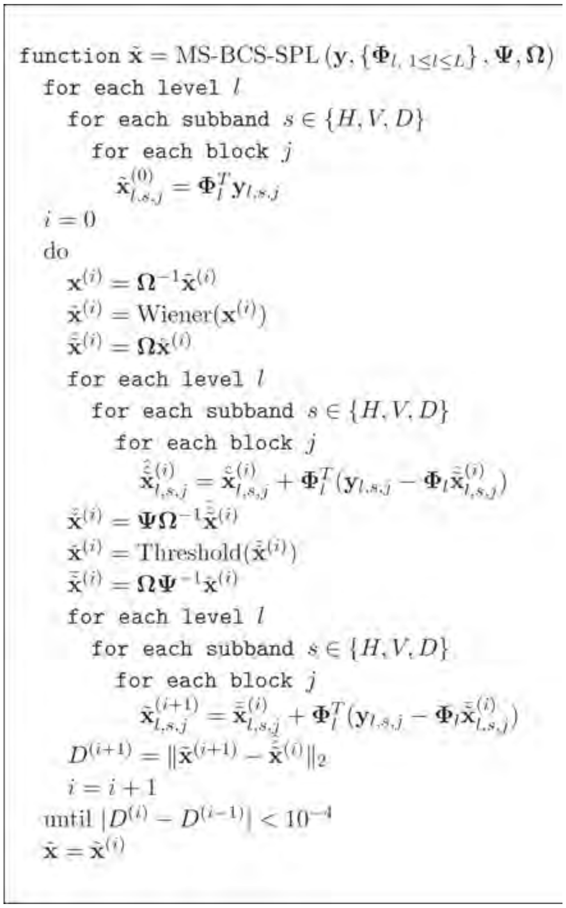
which we have found to perform well in practice. The resulting level subrates S_l for various target subrates S for a DWT with $L = 3$ levels are shown in Table 3.2.

3.5.2 Wavelet-Domain MS-BCS-SPL

The BCS-SPL reconstruction algorithm couples a full-image Wiener-filter smoothing process with a sparsity-enhancing thresholding process in the domain of some full-image sparsity transform $\mathbf{\Psi}$. Interleaved between the smoothing and thresholding operations lie Landweber steps in the form of

$$\mathbf{x} \leftarrow \mathbf{x} + \mathbf{\Phi}^{\mathrm{T}}(\mathbf{y} - \mathbf{\Phi}\mathbf{x}), \tag{3.22}$$

where $\mathbf{\Phi}$ is a measurement matrix. Figure 3.6 illustrates how the BCS-SPL reconstruction of Figure 3.2 is modified to accommodate the situation in which CS measurement takes place within a multiscale transform $\mathbf{\Omega}$ as in (3.15). In essence, the resulting MS-BCS-SPL reconstruction applies a Landweber step on each block of each subband in each decomposition level separately using the appropriate



```
function $\hat{\mathbf{x}} =$ MS-BCS-SPL $(\mathbf{y}, \{\mathbf{\Phi}_{l,\ 1 \le l \le L}\}, \mathbf{\Psi}, \mathbf{\Omega})$
  for each level $l$
    for each subband $s \in \{H, V, D\}$
      for each block $j$
        $\hat{\mathbf{x}}^{(0)}_{l,s,j} = \mathbf{\Phi}_l^T \mathbf{y}_{l,s,j}$
  $i = 0$
  do
    $\mathbf{x}^{(i)} = \mathbf{\Omega}^{-1}\hat{\mathbf{x}}^{(i)}$
    $\hat{\mathbf{x}}^{(i)} =$ Wiener$(\mathbf{x}^{(i)})$
    $\bar{\hat{\mathbf{x}}}^{(i)} = \mathbf{\Omega}\hat{\mathbf{x}}^{(i)}$
    for each level $l$
      for each subband $s \in \{H, V, D\}$
        for each block $j$
          $\hat{\hat{\mathbf{x}}}^{(i)}_{l,s,j} = \bar{\hat{\mathbf{x}}}^{(i)}_{l,s,j} + \mathbf{\Phi}_l^T(\mathbf{y}_{l,s,j} - \mathbf{\Phi}_l\bar{\hat{\mathbf{x}}}^{(i)}_{l,s,j})$
    $\check{\mathbf{x}}^{(i)} = \mathbf{\Psi}\mathbf{\Omega}^{-1}\hat{\hat{\mathbf{x}}}^{(i)}$
    $\check{\mathbf{x}}^{(i)} =$ Threshold$(\check{\mathbf{x}}^{(i)})$
    $\bar{\mathbf{x}}^{(i)} = \mathbf{\Omega}\mathbf{\Psi}^{-1}\check{\mathbf{x}}^{(i)}$
    for each level $l$
      for each subband $s \in \{H, V, D\}$
        for each block $j$
          $\hat{\mathbf{x}}^{(i+1)}_{l,s,j} = \bar{\mathbf{x}}^{(i)}_{l,s,j} + \mathbf{\Phi}_l^T(\mathbf{y}_{l,s,j} - \mathbf{\Phi}_l\bar{\mathbf{x}}^{(i)}_{l,s,j})$
    $D^{(i+1)} = \|\hat{\mathbf{x}}^{(i+1)} - \hat{\hat{\mathbf{x}}}^{(i)}\|_2$
    $i = i + 1$
  until $|D^{(i)} - D^{(i-1)}| < 10^{-4}$
  $\hat{\mathbf{x}} = \hat{\mathbf{x}}^{(i)}$
```

Fig. 3.6 MS-BCS-SPL reconstruction of a 2D image; Wiener$(\cdot)$ is pixel-wise adaptive Wiener filtering using a neighborhood of 3×3, while Threshold$(\cdot)$ is a thresholding process (from [50]).

block-based $\mathbf{\Phi}_l$ for the current level l. As in the original BCS-SPL, Wiener filtering takes place in the spatial domain of the image, while some thresholding operator is applied in the domain of full-frame sparsity transform $\mathbf{\Psi}$ to promote sparsity.

3.6 Other Approaches to CS Reconstruction of Images

After having considered block-based measurement and reconstruction for images in both the spatial and wavelet domains, we now overview several other prominent solutions for CS of images that have appeared in recent literature. Perhaps the most popular approach to the CS of images follows the MS paradigm [125] in which different decomposition levels in the wavelet domain are measured and reconstructed independently using a generic CS reconstruction. For example, in [5, 35, 63, 64, 77], this approach is applied using the generic reconstructions orthogonal matching pursuits (OMP) [123], stagewise orthogonal matching pursuit (StOMP) [40], sparsity adaptive matching pursuits (SAMP) [35], compressive sampling matching pursuit (CoSaMP) [91], and Bayesian compressive sensing [5, 72]. As representative of such algorithms, we consider a multiscale variant of GPSR [47] as described in [105]. In essence, in this multiscale GPSR (MS-GPSR), GPSR reconstruction is applied independently to each DWT level; subrates in the individual levels are varied such that the baseband is retained in full, and successively higher-resolution decomposition levels feature a reduced subrate.

An example of a wavelet-domain CS measurement and reconstruction that is more sophisticated than the simple generic reconstructions like MS-GPSR involves the use of the well-known fact that wavelet coefficients inhabit a cross-scale tree structure in the transform domain (e.g., [111]). This tree-structured prior model has been incorporated into CS reconstruction both as an explicit coefficient structure [9] as well as in the form of a statistical model [63, 64]. In this latter statistical approach, called tree-structured wavelet compressed sensing (TSW-CS) [63, 64], a hierarchical Bayesian model is imposed on the coefficients across the DWT scales.

Fitting an *a priori* signal model to the CS reconstruction model is a frequently adopted strategy for the CS of images. In [77], a Gaussian scale mixture (GSM) [98] model is incorporated into several wavelet-domain CS reconstructions including reweighted ℓ_1 minimization [19], iteratively reweighted least squares [30], and iterative hard thresholding [14, 15]. Additionally, a spatial-domain piecewise autoregressive (PAR) model was incorporated into a spatial-domain CS reconstruction in [136]. This latter technique, called model-based adaptive recovery of compressive sensing (MARX) in [136], adopts the PAR model to better handle nonstationarity of the sparsity support that images exhibit spatially.

Finally, there exist a number of algorithms for the general problem of inverse imaging that are often used in deconvolution or inpainting applications but can be applied to CS reconstruction as a special case. Such techniques include the algorithms of [11, 13, 135]. As representative of this general class of CS reconstruction, we consider the split augmented Lagrangian shrinkage algorithm (SALSA) of [2]. SALSA is a rather flexible solution to the CS reconstruction problem — when applied to CS, SALSA reconstructs from a full-image SRM-based measurement by enforcing either ℓ_1 sparsity in some transform domain (we use a DWT) or a minimum TV norm. We have found that DWT-based SALSA works better than its TV-based counterpart for the CS reconstruction of still images; additionally, we have found that the original SALSA implementation of [2] outperforms the later constrained variant of [3].

In the experimental results to follow, we compare several of the algorithms mentioned here to the BCS-SPL, MS-BCS-SPL, and TV algorithms discussed previously. Throughout our evaluations, we largely concern ourselves with algorithms that have implementations readily available. Additionally, we restrict our attention to only those implementations that can handle images of a relatively large size (i.e., 512 × 512). This constraint effectively rules out algorithms which cannot accommodate SRM-based measurement operators (e.g., Bayesian compressive sensing [5, 72] and TSW-CS [63, 64], both of which directly access rows/columns of the measurement matrix) or which require an excessively long time to reconstruct a single image (e.g., MARX [136]). By “excessively long,” we mean more than several hours on a modern computer — given that we will be subsequently considering reconstruction of multiple frames for video, it is imperative that realistic reconstructions for video be based on still-image reconstructions that are significantly more expeditious.

3.7 Comparison of Various CS Techniques for Images

We now evaluate the performance of various algorithms described above on several grayscale images of size 512 × 512. We compare the original BCS-SPL [89] with spatial-domain BCS, MS-BCS-SPL [50] with wavelet-domain BCS, TV reconstruction [17], MS-GPSR [105], and SALSA [2]. Both MS-BCS-SPL and BCS-SPL use a DDWT [78] as the sparsity transform $\mathbf{\Psi}$ with bivariate shrinkage [107] applied within the DDWT domain to enforce sparsity as described in [89]. MS-BCS-SPL uses a three-level DWT with the popular 9/7 biorthogonal wavelets [4] as the measurement-domain transform $\mathbf{\Omega}$. At decomposition level l of $\mathbf{\Omega}$, blocks of size $B_l \times B_l$ are individually measured in the DWT domain using the scrambled block-DCT SRM measurement operator of [36]; we use blocks of sizes $B_l = 16$, 32, and 64 for decomposition levels $l = 1, 2$, and 3, respectively ($l = 3$ is the highest-resolution level). On the other hand, BCS-SPL uses $B \times B$ block-based measurement applied directly on the image in its ambient domain; here, $B = 32$. TV and SALSA use the scrambled block-Hadamard SRM of [54] to provide a fast whole-image, spatial-domain CS measurement; additionally, SALSA employs a 9/7 biorthogonal DWT as a sparsity transform. Finally, MS-GPSR is implemented similarly to MS-BCS-SPL — GPSR reconstruction is applied independently to each DWT level using the same $\mathbf{\Omega}$ as MS-BCS-SPL; subrates in the individual levels follow Table 3.2 with measurement using a scrambled block-DCT SRM applied to the entire DWT level. We use our implementation[4] of BCS-SPL and MS-BCS-SPL, ℓ_1-MAGIC[5] for TV, and the GPSR[6] and SALSA[7] implementations from their respective authors.

The reconstruction performance of the various algorithms under consideration is presented in Table 3.3. In most cases, the wavelet-domain measurement and MS reconstruction of MS-BCS-SPL provides a substantial gain in reconstruction quality over the spatial-domain measurement of BCS-SPL, generally on the order of a 1- to 3-dB increase in PSNR. Additionally, MS-BCS-SPL outperforms TV reconstruction in most instances despite the fact that TV has the advantage of full-image measurement; the gains of MS-BCS-SPL over TV are particularly significant at the lowest subrates. MS-BCS-SPL also generally outperforms MS-GPSR even though the latter globally measures each

---

[4] `http://www.ece.msstate.edu/~fowler/BCSSPL/`.

[5] `http://www.l1-magic.org`.

[6] `http://www.lx.it.pt/~mtf/\gls{GPSR}/`.

[7] `http://cascais.lx.it.pt/~mafonso/salsa.html`.

Table 3.3. Reconstruction PSNR in dB; images are 512 × 512.

	Subrate				
Algorithm	0.1	0.2	0.3	0.4	0.5
Lenna					
MS-BCS-SPL	**31.6**	**34.7**	**36.7**	**37.9**	**39.0**
BCS-SPL	28.0	31.6	33.7	35.4	36.9
TV	29.9	32.9	35.0	36.8	38.4
MS-GPSR	30.3	33.6	35.2	36.3	37.8
SALSA	23.9	28.5	31.6	34.0	36.0
Barbara					
MS-BCS-SPL	23.8	25.1	26.1	27.4	28.8
BCS-SPL	22.4	23.8	25.4	27.0	28.7
TV	23.0	24.5	**26.3**	**28.4**	**30.8**
MS-GPSR	**24.0**	**25.3**	26.1	27.5	29.6
SALSA	19.7	22.7	25.1	27.7	30.4
Peppers					
MS-BCS-SPL	**31.1**	**34.2**	**35.7**	**36.8**	**37.7**
BCS-SPL	29.0	32.1	33.8	35.2	36.4
TV	30.4	33.1	34.7	35.9	37.0
MS-GPSR	29.3	31.9	33.1	34.2	35.8
SALSA	23.3	28.2	31.2	33.3	35.0
Mandrill					
MS-BCS-SPL	21.4	**23.0**	**24.6**	**25.5**	**26.5**
BCS-SPL	20.5	21.8	22.9	23.9	25.1
TV	20.5	22.0	23.4	24.9	**26.5**
MS-GPSR	**21.5**	22.9	24.3	25.1	26.3
SALSA	16.6	19.6	21.1	22.5	24.2
Goldhill					
MS-BCS-SPL	**29.0**	**31.1**	**32.8**	**33.7**	34.7
BCS-SPL	27.1	29.1	30.5	31.8	33.1
TV	27.5	29.9	31.6	33.2	**34.8**
MS-GPSR	28.5	30.4	32.2	33.0	34.1
SALSA	22.9	26.0	28.2	30.2	32.0

resolution level. The primary exception is the “Barbara” image — although MS-BCS-SPL outperforms TV at the lowest subrates, MS-GPSR is slightly better. However, TV dominates the performance comparison for “Barbara” at the higher subrates. Figure 3.7 depicts typical reconstruction results for the “Lenna” image using a subrate $S = 0.1$; these figures show only a detailed portion from the center of the image.

As can be seen in Table 3.4, in terms of execution times, reconstruction with MS-BCS-SPL is only slightly slower than BCS-SPL, each running for about half a minute on a dual-core 2.8-GHz machine. On the

(a) MS-BCS-SPL, 31.6 dB (b) BCS-SPL, 28.0 dB

(c) TV, 29.9 dB (d) MS-GPSR, 30.3 dB

(e) SALSA, 23.9 dB

Fig. 3.7 Reconstructions of the 512 × 512 "Lenna" image (shown in detail) for a subrate of $S = 0.1$.

other hand, SALSA is somewhat slower, while the execution times of both MS-GPSR and TV are some two orders of magnitude longer, with TV requiring nearly two hours to reconstruct a single image despite the use of a fast SRM measurement operator.

Table 3.4. Reconstruction time for the 512 × 512 “Lenna” image at subrate of 0.3.

Algorithm	Time (s.)
BCS-SPL	30
MS-BCS-SPL	46
SALSA	111
MS-GPSR	1173
TV	6584

3.8 Perspectives

In applying the CS paradigm to still images, two primary challenges must be addressed. First, the CS reconstruction should be specifically tailored to the fact that the underlying signal is an image, and, second, the measurement and reconstruction processes must accommodate the large-sized signals that accompany multidimensional data without imposing large computational or memory burdens. Above, we have discussed several solutions for the first issue, including the imposition of smoothness, directional sparsity transforms, as well as CS measurement in the wavelet domain. For the second issue, we have considered the use of both SRMs as well as block-based measurement operators in both the measurement and reconstruction processes.

Overall, the MS-BCS-SPL algorithm [50] effectively addresses both issues — MS-BCS-SPL retains the fast execution speed associated with block-based measurement while rivaling the quality of CS reconstructions such as TV that employ full-image measurement. However, there exist practical-implementation issues for any CS technique employing wavelet-domain measurement, including the MS-BCS-SPL and MS-GPSR [105] approaches considered here as well as other prominent CS reconstructions such as TSW-CS [63, 64].

Specifically, the general advantages of BCS are a reduced computational complexity in reconstruction as well as a greatly simplified measurement-operator implementation in both the reconstruction as well as sensing processes. A multiscale BCS in the wavelet domain like that used in MS-BCS-SPL retains these advantages for reconstruction; however, the decomposition of the measurement process as $\boldsymbol{\Phi} = \boldsymbol{\Phi}'\boldsymbol{\Omega}$ entails that the transform $\boldsymbol{\Omega}$ (a dense matrix) wrecks the block-diagonal structure of $\mathbf{\Phi}'$. As a consequence, the resulting $\mathbf{\Phi}$ becomes dense and thus a challenge to implement within a CS sensing device. In fact, it is not clear that this can be done without simply storing the dense $\mathbf{\Phi}$ within the sensing device. As a consequence, wavelet-domain measurement defeats the computation and memory advantages of BCS in the sensing device, although these advantages can still be exploited on the reconstruction side of the system. Furthermore, wavelet-domain measurement with most popular wavelets will require the measurement matrix $\mathbf{\Phi}$ to be real-valued even if the underlying $\mathbf{\Phi}'$ is binary or Rademacher, thereby requiring duty cycling in a single-pixel camera and further complicating the acquisition process. The situation is similar for SRMs — when $\mathbf{\Phi}'$ is implemented by an SRM, the dense transform $\mathbf{\Omega}$ prevents the procedural generation of the measurement process in the sensing device, again requiring dense storage of a real-valued $\mathbf{\Phi}$. Spatial-domain measurement — either block-based (as in BCS-SPL) or SRM-based (as in TV) — does not encounter this impediment to practical implementation. As a consequence, we will primarily focus on spatial-domain measurement from this point going forward.

In the remainder of this monograph, we turn our attention away from the reconstruction of a single still image and toward applications dealing with multiple images, namely, video sequences and multiview imagery. In each of these applications, we will find ourselves reconstructing a set of multiple images, a task that we could, of course, accomplish by reconstructing each image independently. However, we shall see that significant performance gains can be obtained through reconstructing the multiple images while capitalizing on any correlation that exists across the image set. In any event, we will require a very fast still-image recovery as the foundation of CS reconstruction for this more complicated, multiple-image data in the remainder of this monograph, since the still-image reconstruction will be employed numerous times. Due to its exceedingly fast computation, we therefore employ BCS-SPL as the still-image reconstruction engine upon which mechanisms for the exploitation of cross-image correlation are built.

4

Block-Based Compressed Sensing for Video

In much of the literature in the area of compressed sensing (CS), the primary goal has largely been the blind reconstruction of CS-acquired signals. That is, for the most part, many CS reconstruction strategies are oblivious to the structure of the signal being recovered beyond a general assumption of sparsity in some transform basis. Recently, however, several reconstruction approaches have focused on situations in which additional information about the signal content is available to aid signal reconstruction; sometimes this additional information is called "side information." Of specific interest here is the situation in which one or more predictions of the signal to be recovered are available to the CS reconstruction process. Video sequences are one form of data in particular in which predictions are commonly used in various forms of processing. Specifically, it is typical in video processing that one or more reference frames are used to make predictions of some current frame such that the resulting residual frame has dramatically lowered signal energy leading to more efficient representation and processing. This paradigm is, in fact, fundamental to the traditional source coding of video and an essential part of all modern video standards such as MPEG [67, 68] and H.264/AVC [65].

We now consider the CS reconstruction of video sequences in which frame-to-frame predictions are used to aid the CS-reconstruction process. In effect, we investigate CS reconstruction on the prediction residual which is, in most cases, significantly more compressible, in the sense of (2.4), than the original frame; such prediction thus results in a higher-quality CS reconstruction. Key to this prediction-driven residual reconstruction is the use of motion estimation (ME) and motion compensation (MC) such that the frame-to-frame predictions compensate for object motion between frames. Such use of ME/MC derives from traditional video-coding algorithms which make extensive use of sophisticated MC strategies.

4.1 CS Acquisition of Video

CS acquisition of video would ideally be global in the sense that CS measurements would span the entire spatial and temporal extent of a video sequence; however, such global CS acquisition of video is largely considered impractical to implement in a real device [41]. As a consequence, we focus on the case in which each video frame is acquired independently with still-image-based CS measurement, for instance, with successive applications of a single-pixel camera as was done in [132]. While other approaches to acquisition might eventually be possible, we consider the single-pixel camera to be a straightforward and realizable framework for capturing images and video via linear projection.

More specifically, CS theory dictates that it is possible to recover a signal of dimension N from a set of measurements of dimension M where $M \ll N$. In the canonical CS acquisition or measurement process,

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}, \tag{4.1}$$

we see that these measurements are calculated as a projection of the entire N-dimensional signal by an $M \times N$ random projection matrix, $\mathbf{\Phi}$. Complications in designing CS hardware arise due to the global nature of this dimensionality-reduction step. That is, a CS device must be able to simultaneously view the entirety of a signal (for video, this means in space as well as in time) and calculate its projection by $\mathbf{\Phi}$ non-computationally in the ambient signal domain. Because of this requirement, simultaneous spatial and temporal measurement of video appears impractical [41], and thus one opts for frame-by-frame measurement.

4.1.1 Hardware Limitations

In the case of natural image signals, we have already employed the single-pixel camera [42, 115, 131, 132] for static-scene measurement. However, the single-pixel camera entails multiple measurements conducted sequentially in time such that the total time to acquire a given signal is increased by a factor of M. Thus, for dynamic scenes, there is a potential for disagreement between successive measurements, as each measurement of the scene is at a different point in time. If object motion between measurements is significant, there could be blurring or other reconstruction errors when recovering the signal from the measurements. For this reason, a viable CS sensing device for dynamic scenes (i.e., video) which uses single-pixel acquisition must have a very short exposure time and delay between measurements.

More specifically, for the CS measurement of video at a target frame rate, R_{f}, using M measurements for each frame, each measurement must be captured within $1/(R_{\mathrm{f}}M)$ seconds. For high-resolution video, M can be somewhat large (though, of course, still much less than the number of pixels, N), and this puts a tight restriction on the sensing device when the measurements are captured sequentially. However, the latest micro-mirror arrays have attained very fast switching speeds, and these speeds are increasing each year as research into microelectromechanical systems (MEMS) continues. For example, [92] reports an MEMS device capable of switching mirror states in 225 ns; such a digital micromirror device (DMD) would permit CS measurement of video with 720 × 480 frames at a subrate of $S = 0.3$ with a frame rate of

$$R_{\mathrm{f}} = \frac{1}{720 \cdot 480 \cdot 0.3 \cdot 225 \times 10^{-9}} \approx 42\,\text{frames/s}. \tag{4.2}$$

While devices with such switching speeds are not yet in commercial production, even devices with much lower switching rates (e.g., on the order of 0.1–1 MHz) can accomplish video acquisition by trading-off spatial resolution, frame rate, or reconstruction quality. For example, at a switching rate of 0.5 MHz and a subrate $S = 0.2$, a video sequence could be captured at standard CIF resolution (352×288) at $R_{\mathrm{f}} \approx 24$ frames/s.

If the measurement matrix $\mathbf{\Phi}$ has real-valued, rather than binary, entries (as is the case with a dense Gaussian measurement operator), it is possible to use pulse-width modulation or dithering to simulate fractional transmittance from the mirror to the sensor. However, such approaches are problematic. Firstly, the fractional values are subject to quantization error induced by the accuracy of the pulse-width modulation. Secondly, for a fixed measurement subrate, the mirror-switching speed must be increased by a factor dependent on the quantization precision used.

To reduce some of the necessary tradeoffs caused by sequential measurement, multiple sensors operating in parallel may be used to increase the effective measurement subrate. A multiple-pixel device would operate in much the same manner as the single-pixel camera and would still maintain a low sensor density as compared to a full-resolution sensor.

4.1.2 Physical Limitations

The exposure time necessary to accurately measure the amount of radiation incident on the photosensor is another — and perhaps more significant — component to the time required for each measurement. Exposure time presents a challenge: if the exposure time is too short, then noise from dark current within the system or Poisson noise induced by the photon arrival rate can overwhelm the actual measurements, requiring more sophisticated — and costly — photosensors. However, the increased cost of such photosensors is offset by the fact that there would be need for only one photosensor rather than an entire array. Also, using the single-pixel framework, photons from all the mirrors that face the sensor are concentrated onto a single sensor during measurement. For example, consider block-based compressed sensing (BCS) using blocks of size 64×64. If we assume that, on average, half the mirrors point toward the sensor for any given measurement, the sensor is exposed to $64^2/2 \approx 2000$ times more light than a single sensing element would be in a traditional dense-array sensor. This focusing of energy increases the signal-to-noise performance of the single sensor as compared to the limited spatial binning capabilities of sensors in a dense-grid configuration.

These considerations make high spatial- and time-resolution video difficult but arguably not impossible. In a general sense, the design of CS acquisition devices would necessitate some tradeoff between the number of measurements acquired for each frame, the desired frame rate (temporal resolution), and the exposure time for each measurement. For example, applications such as distributed video networks, or other *ad hoc* distributed sensor networks, could make use of cheaper CS video-sensor systems in surveillance capacities wherein spatial resolution is not the top priority. With smaller-resolution frames, the number of measurements for each frame decreases for a given target subrate, thereby allowing greater exposure time for each measurement within a frame-rate constraint.

Many of these design constraints might be justified in settings wherein every measurement is costly or the sensors themselves are costly. Sensing signals in exotic spectra — such as in thermal, terahertz, and medical imaging — represent areas wherein CS can potentially reduce either device or acquisition cost. Except for perhaps niche applications, imaging within the visible domain using CS is not likely to be competitive with existing low-cost CCD or CMOS imagers. However, in the remainder of our discussion, we use visible-domain imagery in our experiments to explore potential recovery techniques for signals. Thermal, infrared, and medical images exhibit characteristics similar to natural, visible domain, images — most importantly, piecewise smoothness. Because of these similarities, we anticipate that the methods we demonstrate could also be applied to extra-visible spectra with similar effect.

4.1.3 Block-Based Acquisition of Video

So far, this discussion has considered only the case of a globalized, and therefore dense, structure of $\mathbf{\Phi}$. From prior discussion, we know there are some inherent drawbacks to such a dense measurement process, such as reconstruction time and the memory requirements of storing $\mathbf{\Phi}$. However, the hardware of the single-pixel camera can accommodate a BCS measurement procedure, and the practical considerations discussed here still apply in the BCS context. BCS also has the added advantage of decreasing the bandwidth required to transmit measurement vectors between system memory and the DMD array since a small measurement vector can be transmitted once and subsequently translated across the DMD array for each block.

For static images, BCS measurement is straightforward, requiring only a block-diagonal $\mathbf{\Phi}$ be employed. In the case of BCS of video, however, since each block is measured independently, each block represents a different point in the time of the scene, rather than each frame. On the one hand, because fewer measurements are required for a single block than for an entire image, the time duration between the first measurement and the last measurement in a given block is a multiplicative factor less than if we take global measurements of the entire frame. This decreases the possibility for blurring within a given block in a frame. On the other hand, since each block represents a different point in time, there could be some content drift between the blocks of a given frame if the dynamic content being represented is changing sufficiently fast. If it is more desirable to have blurring rather than drift, then the CS device could scan through the blocks repeatedly, taking a single measurement at a time. This would simulate the measurement timing of a global CS measurement of the frame and could be accomplished by simply reordering the rows of the block-diagonal $\mathbf{\Phi}$.

4.2 Straightforward CS Reconstruction for Video

The straightforward implementation of CS on video would involve, as for images, the vectorization of a 3D group of frames into a single 1D vector. However, the computation and memory issues associated with this vectorization approach for 2D images are exacerbated with the even greater dimensionality present in video data. Additionally, a global sensing simultaneously across the spatial and temporal extent of a group of frames is likely to be impractical as mentioned before. Consequently, we focus on the situation in which video frames are acquired independently in a 2D fashion, e.g., by applying a suitable image acquisition (e.g., single-pixel camera) in a frame-by-frame fashion as in [131, 132]. Again, to cut computation and memory, we focus on a block-based image measurement applied frame by frame. We now consider several strategies for the CS reconstruction of video frames acquired in this manner; these include both intraframe and 3D versions of BCS-SPL.

Given a frame-by-frame acquisition, the most straightforward reconstruction would be to reconstruct the individual frames independently using the BCS-SPL procedure introduced previously. However, such an intraframe BCS-SPL reconstruction ignores the fact that consecutive video frames are usually highly correlated. Nonetheless, intraframe 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 [131, 132]. Although such 3D reconstruction could be applied across a video volume as a whole (as in [131, 132]), for computation and memory issues (which are likely to be substantial given the increased dimensionality), we consider a 3D version of the BCS-SPL algorithm, 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 acquisition, the individual video frames are acquired using a 2D block-based measurement 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 $B \times B \times P$. For thresholding, we employ a variation on (3.13) wherein the median operator is replaced by the mean (i.e., $\sigma^{(i)} = \mathrm{mean}(|\check{\check{\mathbf{x}}}^{(i)}|)/0.6745$); we have observed empirically that this mean-based estimator yields faster convergence in the Landweber operation for video. We refer to the resulting technique as 3D-BCS-SPL reconstruction.

It was argued in [132] 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 intraframe 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 be incorporated into the reconstruction; we consider such ME/MC-based reconstruction next.

4.3 The Motion-Compensated BCS-SPL Algorithm

In video coding, 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-coding systems. In [90], this ME/MC framework was incorporated 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 coding. The BCS-SPL for video proposed in [90] uses the same simple measurement as was used for 2D images previously — block-based random CS measurements are applied frame by frame. The resulting motion-compensated version of BCS-SPL, or motion-compensated BCS-SPL (MC-BCS-SPL), consists of several main components: residual reconstruction, in which BCS-SPL is applied to an MC residual; multihypothesis initialization, in which intraframe BCS-SPL is used to initialize the MC-BCS-SPL reconstruction; and forward/backward MC, in which multiple reconstruction passes are performed in multiple directions across the GOP. These components are explored below.

4.3.1 Residual Reconstruction

Suppose we have two consecutive frames, the current frame, $\mathbf{x}$, and the reference frame, $\mathbf{x}_{\text{ref}}$. We assume that the reference frame has been previously reconstructed, while we have only random measurements, $\mathbf{y}$, of the current frame, using BCS measurement. 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 $\mathbf{y}$ (we consider a more sophisticated estimate of $\hat{\mathbf{x}}$ later). However, such intraframe reconstruction does not capitalize on our knowledge of reference frame $\mathbf{x}_{\text{ref}}$. Yet, at this point, we have approximations to the current frame as well as to 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 coding. 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 measurement operator to the motion-compensated frame; i.e., for each block j:

$$\mathbf{y}_{\text{r}_j} = \mathbf{y}_j - \mathbf{\Phi}_B \hat{\mathbf{x}}_{\text{mc}_j}. \tag{4.3}$$

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

$$\mathbf{y}_{\text{r}_j} = \mathbf{y}_j - \mathbf{\Phi}_B \hat{\mathbf{x}}_{\text{mc}_j} = \mathbf{\Phi}_B \big(\mathbf{x}_j - \hat{\mathbf{x}}_{\text{mc}_j}\big) = \mathbf{\Phi}_B \mathbf{x}_{\text{r}_j}. \tag{4.4}$$

If the MC process is reasonably accurate, the residual frame $\mathbf{x}_\text{r}$ should be more compressible — in the sense of (2.4) — than the original image $\mathbf{x}$. This is demonstrated empirically for a video sequence in Figure 4.1 wherein it is seen that, in reference to (2.4), the transform-coefficient magnitudes decay more quickly for the residual frame $\mathbf{x}_\text{r}$ than for the original frame $\mathbf{x}$. In such a case, CS reconstruction will thereby be more effective at recovering the residual $\mathbf{x}_\text{r}$ from $\mathbf{y}_\text{r}$ than it is at recovering $\mathbf{x}$ from $\mathbf{y}$. Let $\hat{\mathbf{x}}_\text{r}$ be such a BCS-SPL reconstruction from $\mathbf{y}_\text{r}$; consequently, we can form a new approximation to $\mathbf{x}$ as

$$\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{mc}} + \hat{\mathbf{x}}_\text{r}. \tag{4.5}$$

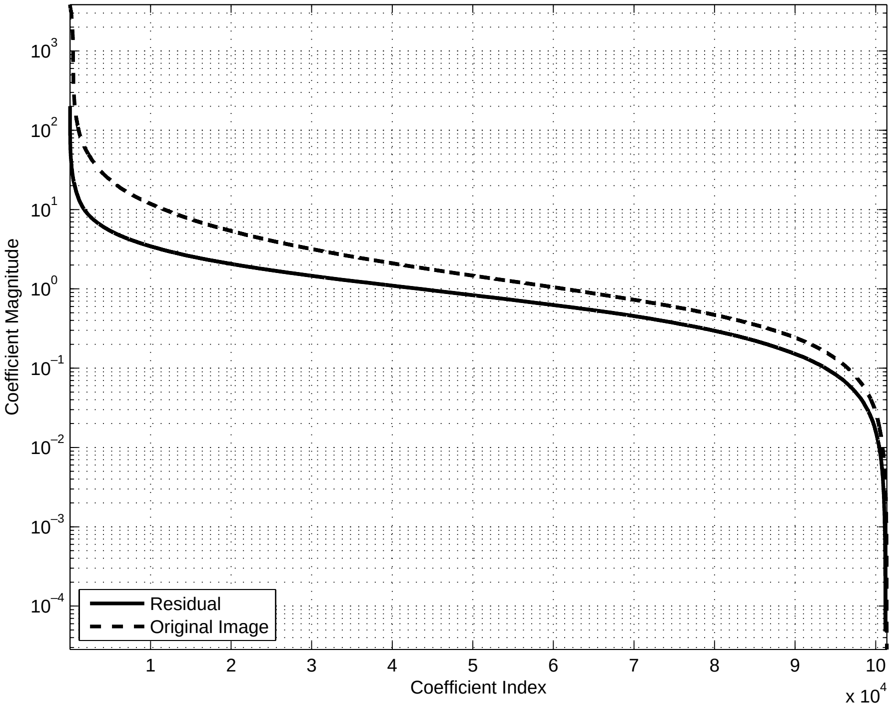
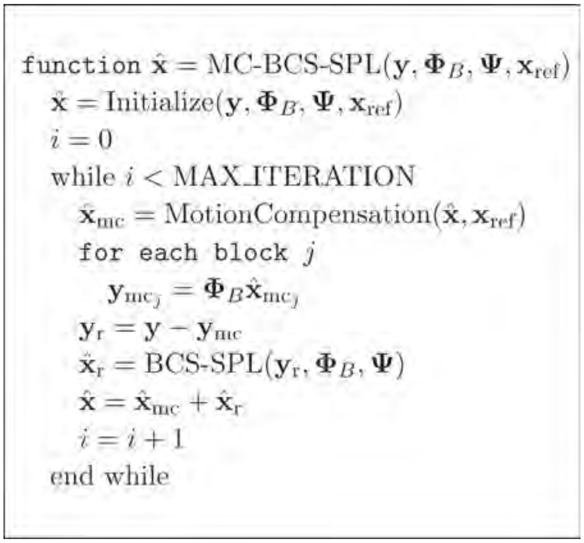


Fig. 4.1 Decay of the magnitudes of the transform coefficients for frame 1 of the "Foreman" video sequence as compared to that of the motion-compensated residual between frames 1 and 0. ME/MC is based on 16 × 16 blocks with quarter-pixel accuracy over a window of size 15 × 15 pixels; ME/MC is performed between the original frames of the sequence. The transform is a 4-level biorthogonal 9/7 DWT.

We now have a new approximation to the current frame that is of better quality than the initial approximation that we created from a direct BCS-SPL reconstruction from $\mathbf{y}$. Consequently, we will be able to produce a more accurate motion-vector field from ME applied to this new $\hat{\mathbf{x}}$ and $\mathbf{x}_{\text{ref}}$, and further enhancement can be expected by iteratively repeating the above process. The resulting MC-BCS-SPL algorithm [90] is summarized in Figure 4.2.

4.3.2 Multihypothesis Initialization

We now return to the issue of producing an initial estimate, $\hat{\mathbf{x}}$, of the current frame. Let us assume two frames again, reference frame, $\mathbf{x}_{\text{ref}}$,



```
function $\hat{\mathbf{x}}$ = MC-BCS-SPL($\mathbf{y}, \mathbf{\Phi}_B, \mathbf{\Psi}, \mathbf{x}_{\text{ref}}$)
  $\hat{\mathbf{x}}$ = Initialize($\mathbf{y}, \mathbf{\Phi}_B, \mathbf{\Psi}, \mathbf{x}_{\text{ref}}$)
  $i = 0$
  while $i <$ MAX_ITERATION
    $\hat{\mathbf{x}}_{\text{mc}}$ = MotionCompensation($\hat{\mathbf{x}}, \mathbf{x}_{\text{ref}}$)
    for each block $j$
      $\mathbf{y}_{\text{mc}_j} = \mathbf{\Phi}_B \hat{\mathbf{x}}_{\text{mc}_j}$
    $\mathbf{y}_{\text{r}} = \mathbf{y} - \mathbf{y}_{\text{mc}}$
    $\hat{\mathbf{x}}_{\text{r}}$ = BCS-SPL($\mathbf{y}_{\text{r}}, \mathbf{\Phi}_B, \mathbf{\Psi}$)
    $\hat{\mathbf{x}} = \hat{\mathbf{x}}_{\text{mc}} + \hat{\mathbf{x}}_{\text{r}}$
    $i = i + 1$
  end while
```

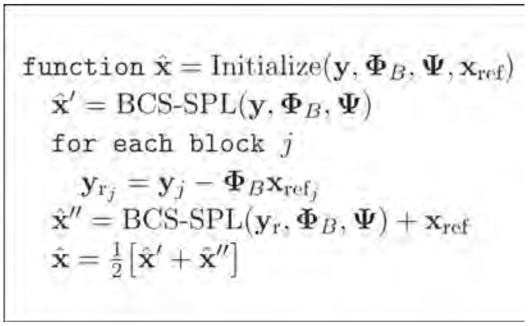
Fig. 4.2 MC-BCS-SPL reconstruction of the current frame from a single reference frame. BCS-SPL(·) is the BCS-SPL reconstruction from Figure 3.2; Initialize(·) is described in Figure 4.3 (from [90]).

and random measurement, $\mathbf{y}$, of the current frame, $\mathbf{x}$. As mentioned above, one possible way to obtain an initial guess for the current frame is to apply BCS-SPL; let this estimate be $\hat{\mathbf{x}}'$. An alternative approach would be a residual reconstruction of the frame-to-frame difference with the motion field set to zero. The former approximation might be suitable for a dynamic sequence with high-motion content. On the other hand, some natural video sequences are relatively stationary, such as newscasts and surveillance video whose motion vectors are mostly zero or close to zero; in these cases, a residual reconstruction with a zero motion field might be more appropriate as an initial reconstruction. In this zero-motion case, the residual reconstruction (4.3) can be rewritten as

$$\mathbf{y}_{\text{r}_j} = \mathbf{y}_j - \mathbf{\Phi}_B \mathbf{x}_{\text{ref}_j}, \tag{4.6}$$

leading to an alternate estimate, $\hat{\mathbf{x}}''$, of the current frame.

Both of these approaches — BCS-SPL reconstruction as well as zero-motion residual reconstruction — provide a guess for the current



```
function $\hat{\mathbf{x}} = \text{Initialize}(\mathbf{y}, \mathbf{\Phi}_B, \mathbf{\Psi}, \mathbf{x}_{\text{ref}})$
  $\hat{\mathbf{x}}' = \text{BCS-SPL}(\mathbf{y}, \mathbf{\Phi}_B, \mathbf{\Psi})$
  for each block $j$
    $\mathbf{y}_{\text{r}_j} = \mathbf{y}_j - \mathbf{\Phi}_B \mathbf{x}_{\text{ref}_j}$
  $\hat{\mathbf{x}}'' = \text{BCS-SPL}(\mathbf{y}_{\text{r}}, \mathbf{\Phi}_B, \mathbf{\Psi}) + \mathbf{x}_{\text{ref}}$
  $\hat{\mathbf{x}} = \frac{1}{2}\left[\hat{\mathbf{x}}' + \hat{\mathbf{x}}''\right]$
```

Fig. 4.3 Multihypothesis initialization in MC-BCS-SPL reconstruction.

frame, but neither is likely to be exactly correct, and each might be somewhat different than the true current frame. As a consequence, as the final initial frame, we average these two approximations as the initial guess for the MC-BCS-SPL reconstruction. This process, which is depicted algorithmically in Figure 4.3, is similar to multihypothesis MC (e.g., [57, 112]) in traditional video coding, so we refer to it as multihypothesis initialization. We have observed empirically over a variety of video sequences that this multihypothesis initialization outperforms either of the single hypotheses, $\hat{\mathbf{x}}'$ or $\hat{\mathbf{x}}''$, used alone; sample results are illustrated in Figure 4.4. In short, in the absence of significant motion, multihypothesis initialization permits the initial reconstruction to exploit the reference frame in the case that it has higher quality than the intraframe reconstruction of the current frame (which is, in fact, likely, due to the forward–backward reconstruction process which we discuss next).

4.3.3 Forward/Backward Motion Compensation

Thus far, we have 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."

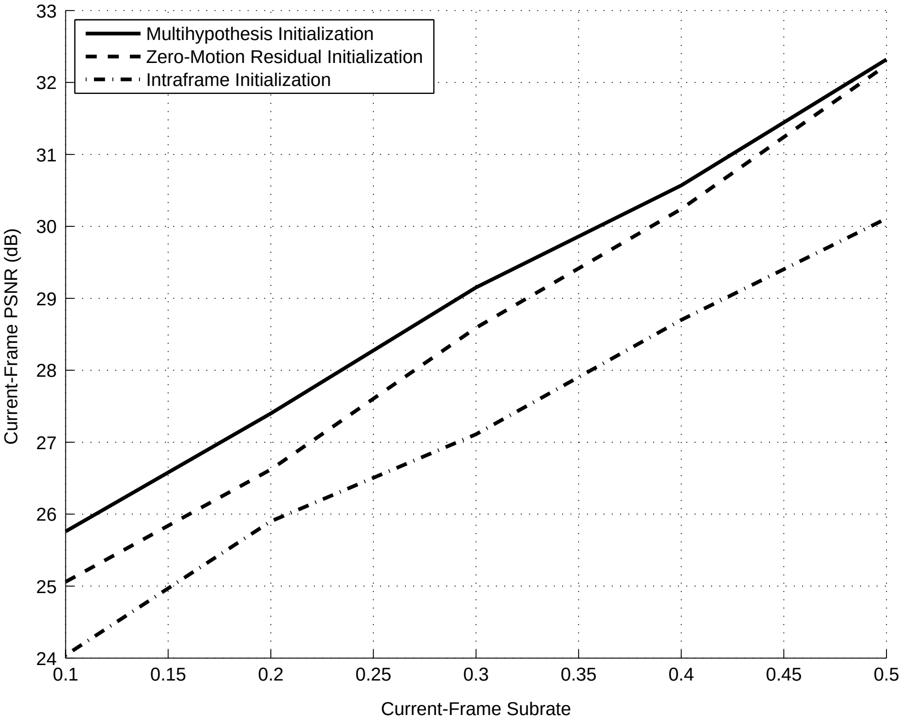


Fig. 4.4 Various strategies for the initial estimate of the current frame within MC-BCS-SPL. Results are for frame 1 of the "Coastguard" sequence with frame 0 as the reference frame $\mathbf{x}_{\mathrm{ref}}$; the reference frame is reconstructed independently using BCS-SPL with a subrate of 0.7. Performance is shown for MC-BCS-SPL reconstruction of the current frame using the designated initialization (see Figure 4.3) for the current frame: intraframe initialization ($\hat{\mathbf{x}}'$), zero-motion residual initialization ($\hat{\mathbf{x}}''$), or multihypothesis initialization ($[\hat{\mathbf{x}}' + \hat{\mathbf{x}}'']/2$).

In the MC-BCS-SPL setting, we have block-based random measurements of each of the frames of the GOP; i.e., $\mathbf{y}_p = \mathbf{\Phi}_p \mathbf{x}_p$ for $0 \leq p \leq P - 1$, where p is the frame number. $\mathbf{\Phi}_p$ is the random block-based measurement operator for frame p; we assume that this measurement operator has subrate of $S_p = M_p/N$ 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 acquired with the same subrate, and (2) the key frame, $\mathbf{x}_0$, is acquired at a relatively high subrate, while all the non-key frames are at an identical lower subrate.

Direct, intraframe 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 have observed a successively lower reconstruction quality as we progress through the non-key frames. That is, we have observed 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. In [90], it is proposed 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 (here, we abuse our notation by indexing this frame from the next GOP relative to the current 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 reach 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. Thus, in [90], it is proposed to use 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 $\frac{P}{2}+1 \leq p \leq P-1$. To reconstruct the center frame, $\hat{\mathbf{x}}_{P/2}$, we slightly modify the MC-BCS-SPL procedure of Figure 4.2 to incorporate bidirectional MC prediction. Specifically, MC-BCS-SPL iterates first using $\hat{\mathbf{x}}_{P/2-1}$ (the forward reference frame) as the reference frame for the MotionCompensation$(\cdot)$ operator, and then iterates using $\hat{\mathbf{x}}_{P/2+1}$ (the backward reference frame). This process is repeated for an additional set of both forward and backward iterations to yield the final reconstruction of the center frame. Figure 4.5 illustrates this forward/backward MC-BCS-SPL reconstruction process.

Because center frame $\hat{\mathbf{x}}_{P/2}$ is recovered using both forward and backward reconstruction, it turns out to usually have a relatively high reconstruction quality. As a second and final stage of reconstruction, we update the frames between the center frame ($\hat{\mathbf{x}}_{P/2}$) and the key frames ($\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_P$) by performing the same bidirectional reconstruction as was used on the center frame. A complete description of the two phases of reconstruction is given in Table 4.1. In this table, Phase 1 is forward/backward reconstruction from Figure 4.5, while Phase 2 is "enhancement" reconstruction from the center frame toward the two key frames in reverse direction.

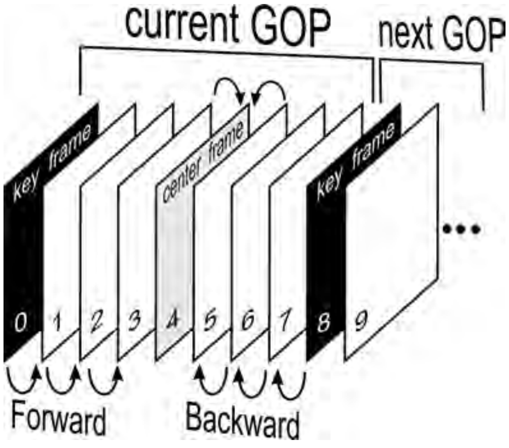


Fig. 4.5 MC-BCS-SPL reconstruction applied to a GOP of size $P = 8$ frames. Black frames: direct (intraframe) BCS-SPL reconstruction of key frames; white frames: forward/backward MC-BCS-SPL reconstruction of non-key frames (forward MC-BCS-SPL for frames 1–3, backward MC-BCS-SPL for frames 5–7); gray frame: bidirectional MC-BCS-SPL reconstruction of center non-key frame.

Table 4.1. Two-phase multiframe reconstruction for a GOP of size $P = 8$. Phase 1 — forward/backward reconstruction and bidirectional center-frame reconstruction; Phase 2 — enhancement reconstruction between the center frame and the two key frames.

	Current frame	Reference frame
Phase 1	1	0
	2	1
	3	2
	8	7
	7	6
	6	5
	4	$3, 5^*$
Phase 2	3	4
	5	4
	2	$1, 3^*$
	1	2
	3	2
	6	$5, 7^*$
	5	6
	7	6

*Bidirectional reconstruction.

4.4 Other Approaches to CS Reconstruction of Video

While the straightforward video reconstructions represented by intraframe BCS-SPL and 3D-BCS-SPL do not exploit ME/MC, the MC-BCS-SPL technique [90] discussed above does. In recent literature, there have been a number of strategies proposed for the CS reconstruction of video, and one finds both techniques that use explicit ME/MC as well as those that do not. Below, we survey recent literature in the area of CS reconstruction of video — we first consider techniques without ME/MC before considering several approaches that exploit explicit ME/MC processes.

4.4.1 Approaches Without Motion Estimation or Compensation

A number of approaches to the CS reconstruction of video were developed for the particular case of dynamic magnetic resonance imaging (MRI). This type of image sequence tends to have less motion, and the motion tends to be less of a strictly translational nature, than does video acquired from natural photographic scenes. Initial work adopted the volumetric reconstruction employed originally in [131, 132] — for example, [52] reconstructs a dynamic MRI volume using a temporal Fourier transform coupled optionally with a spatial wavelet transform as a 3D sparsity basis.

Vaswani et al. [81, 101, 126, 127, 128] have proposed a variety of related approaches for the CS reconstruction of dynamic MRI data. Fundamental to several of these techniques [81, 101, 127] is the general strategy of residual reconstruction from a prediction of the current frame as is employed in MC-BCS-SPL, the key difference being that, rather than using an ME/MC-based prediction, Vaswani et al. employ a least-squares [127] or Kalman-filtered [101] prediction. These predictions are driven by an explicit sparsity pattern for the current frame; the techniques attempt to track this sparsity pattern as it evolves from frame to frame. It is assumed that the sparsity pattern evolves slowly over time, an assumption that may not hold in general video with arbitrary object motion. Another variant of these algorithms, called "Modified-CS" [128] adopts the strategy of finding the ℓ_1 optimal solution *outside* the currently known sparsity pattern which is again tracked frame to frame. Finally, [81] coupled the modified-CS paradigm with the residual-reconstruction strategy; this latter variant, called "Modified-CS-Residual," is a prominent benchmark for gauging CS-reconstruction performance for not only dynamic MRI but also video as well.

Alternatively, [104] proposes another strategy that also attempts to explicitly track temporal changes in video. In this case, [104] deploys a linear dynamical system (LDS) that models the evolution of a video scene in terms of low-dimensional dynamic parameters and high-dimensional static parameters such that the CS measurement process is applied to only the dynamic portion of the signal. It is observed in [104], however, that this LDS-based strategy works well for relatively low-motion content of a largely textural nature, such as flames, water, and traffic, whereas the more complex and translational motion often associated with more arbitrary video content is not properly handled by the model.

This observation echoes the primary drawback that applies in general to the techniques for CS reconstruction of video that do not employ

ME/MC. Methods such as those considered above are typically best suited to video content that varies only quite slowly over time, such as dynamic MRI. For more complex temporal variation, particularly the non-stationary translational object motions that often occur in video of natural scenes, the use of explicit ME/MC is warranted. MC-BCS-SPL as previously described is a prime example; next, we consider several additional techniques that incorporate some forms of explicit ME/MC into CS reconstruction.

4.4.2 Approaches Using Motion Estimation and Compensation

In recent literature, there have been a handful of approaches that incorporate ME/MC for CS reconstruction of video. In general, ME/MC will accommodate more complex object motions than can the low-motion techniques considered above.

In [113], an extension of the dynamic-MRI reconstruction due to Vaswani and Lu [128] was proposed. In essence, rather than simply estimating updates to the time-varying sparsity pattern directly from the preceding frame, the preceding frame is first motion-compensated, allowing for more arbitrary object motions to be handled. The technique of [113] inherits, however, the drawback identified previously for [128] in that the temporal evolution of the sparsity pattern is assumed to be slow.

We now overview several techniques that are most closely related to MC-BCS-SPL [90]. Like MC-BCS-SPL, these techniques partition a GOP into key and non-key frames, with the key frames anchoring a ME/MC process of prediction of the non-key frames. Kang and Lu [76] exploit temporal correlation by constructing a motion-compensated interpolation between consecutive key frames. This motion-compensated interpolation is then used as the initialization point of a still-image CS reconstruction ([76] uses a full-frame gradient projection for sparse reconstruction (GPSR) [47] reconstruction). This motion-compensated interpolation replaces the linear initialization, $\mathbf{x} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{y}$, often used to commence still-frame reconstruction, e.g., such as in BCS-SPL (see Figure 3.2). Also proposed is a modified stopping criterion that terminates GPSR iterations to prevent excessive divergence of the reconstruction from the motion-compensated prediction of the frame. The key frames are reconstructed using an independent still-image CS reconstruction with a subrate higher than that used for the non-key frames.

Another reconstruction algorithm driven by ME/MC between high-quality key frames was considered in [74, 75]. This algorithm, called focal underdetermined system solver in *k*–*t* space (*k*–*t* FOCUSS) in [74], assumes that there exist one or two key frames obtained through some separate means, and then CS reconstruction is driven by residuals between each intervening non-key frame and a block-based bidirectional motion-compensated prediction from each of the key frames (or a single unidirectional motion-compensated prediction in the event that only one key frame is available). As in MC-BCS-SPL, full-search block matching is used for the ME/MC process. We note that *k*–*t* FOCUSS was designed specifically for dynamic MRI; consequently, [74, 75] uses relatively long GOPs (e.g., 25 frames for a cardiac cine sequence in [75]) with perfect key frames (i.e., subrate = 1.0) at each end.

Another ME/MC-based reconstruction was proposed by Do et al. [34]. This technique, called distributed compressed video sensing (DISCOS) in [34], also partitions the GOP into key and non-key frames; however, the key frames are coded using a traditional intraframe source coder such as H.264/AVC intracoding. The non-key frames are then reconstructed from both block-based and frame-based CS measurements; specifically, the block-based measurement of a 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 Figure 4.2, except that it is not iterative, and the measurement operator is a mixture of full-frame and block-based measurement rather than simply block-based.

Although having some similarities, there exist some substantial differences between MC-BCS-SPL and the techniques of [34, 74, 75, 76]. For one, both Kang and Lu [76] (GOP size: 3 frames) and DISCOS [34]

(GOP size: 4 frames) rely on short GOPs yielding frequent high-quality key frames due to the fact that their ME/MC processes, being based on only key frames, cannot support long temporal distances existing between key frames. Similarly, *k–t* FOCUSS [74, 75] implicitly assumes that image content changes slowly over time such that motion-compensated interpolation over longer GOPs is feasible, as is often the case with the dynamic MRI data for which *k–t* FOCUSS was designed. In contrast, MC-BCS-SPL is built on frame-to-frame ME/MC in which all frames, key and non-key, participate as reference frames in ME/MC. In this sense, the techniques in [34, 74, 75, 76] effectively ignore the motion information that can be derived from the intervening non-key frames. Exploiting this frame-to-frame motion permits MC-BCS-SPL to better handle the complex and rapid motions that often occur in video.

Along these lines, MC-BCS-SPL iterates ME/MC using the latest reconstructed frames to improve the estimate of the motion vectors which then leads to further improvement in the frame reconstruction. On the other hand, iterative improvement is not possible in the techniques in [34, 76] since the motion-vector fields are derived from only the key frames which are reconstructed only once.

4.4.3 Motion-Compensated Temporal Filtering

A final strategy to incorporating explicit ME/MC into CS reconstruction for video is represented by the technique proposed in [96]. In contrast to the ME/MC-based reconstructions like MC-BCS-SPL as well as those of [34, 74, 75] discussed above which are inspired by the traditional hybrid video-coding architecture, the technique of [96] draws from the strategy of motion-compensated temporal filtering (MCTF) (e.g., [24, 27, 51, 94, 97, 106]), adopting specifically the MCTF of [106] known as lifting-based invertible motion adaptive transform (LIMAT). In essence, MCTF is combined with a spatial discrete wavelet transform (DWT) to implement a motion-compensated 3D transform. 3D reconstruction simultaneously across all frames similar to [131] is then conducted. A key aspect of the proposed approach is that the 3D reconstruction is applied in each resolution level of the spatial DWT separately, using the reconstruction of the previous spatial resolution for determining the motion vectors between all the frames to drive the MCTF within the current resolution level. As a consequence, the technique of [96] has the advantage of explicit ME/MC like those methods surveyed above. However, it also inherits the computational drawback of the cross-frame volumetric reconstruction identified above as a significant impediment for techniques such as [131]; specifically, computational issues are compounded in [96] since a separate volumetric reconstruction, as well as ME/MC process, is conducted for each spatial resolution level. As a consequence, we will focus on the ME/MC-based reconstructions discussed above rather than that of [96].

4.5 Experimental Observations

We now examine the performance of MC-BCS-SPL reconstruction relative to its simple intraframe and 3D volumetric variants to demonstrate that significant gain results from the explicit exploitation of motion information within the CS reconstruction of video. We use the common video sequences “Coastguard” (296 frames), “Football” (120 frames), “Foreman” (296 frames), “Hall Monitor” (88 frames), “Mobile” (296 frames), “Mother and Daughter” (296 frames), “Stefan” (296 frames), and “Susie” (72 frames). These sequences have grayscale CIF frames of size 352 × 240 or 352 × 288. 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 an orthonormalized dense Gaussian measurement matrix 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 as well as in the multihypothesis initialization of Figure 4.3. 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 Figure 4.2 is set to 5.

We compare MC-BCS-SPL to the intraframe-BCS-SPL and 3D-BCS-SPL alternatives. The most straightforward reconstruction is intraframe 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 the peak signal-to-noise ratio (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 have the same subrate, i.e., $S_{\mathrm{K}} = S_{\mathrm{NK}}$. Alternatively, we also consider the case wherein the key frames have an increased subrate with respect to the non-key frames, i.e., $S_{\mathrm{K}} > S_{\mathrm{NK}}$. A summary of the results from both cases is presented in Tables 4.2 and 4.3.

Table 4.2. Average PSNR in dB for several video sequences.

			MC-BCS-SPL	
S	Intraframe BCS-SPL	3D-BCS-SPL	$S_{\mathrm{K}} = S_{\mathrm{NK}}$	$S_{\mathrm{K}} > S_{\mathrm{NK}}$
Coastguard				
0.1	22.69	22.76	23.06	24.19
0.2	24.70	24.76	25.78	27.12
0.3	26.37	26.45	28.29	29.62
0.4	27.99	27.95	30.88	32.21
0.5	29.60	29.57	33.58	34.73
Football				
0.1	20.68	20.84	20.86	20.73
0.2	22.39	22.67	23.35	23.36
0.3	24.08	24.35	25.51	25.86
0.4	25.68	25.93	27.44	28.05
0.5	27.31	27.53	29.40	30.11
Foreman				
0.1	25.99	26.47	27.58	28.00
0.2	28.83	29.18	31.01	32.27
0.3	31.21	31.36	33.50	34.95
0.4	33.18	33.29	35.76	37.41
0.5	35.07	35.13	38.05	39.83
Hall monitor				
0.1	22.55	22.78	22.79	26.43
0.2	24.78	25.22	25.31	30.76
0.3	26.89	27.26	27.40	33.20
0.4	28.81	29.06	29.35	35.54
0.5	30.73	30.96	31.34	38.12

Table 4.3. Average PSNR in dB for several video sequences.

			MC-BCS-SPL	
S	Intraframe BCS-SPL	3D-BCS-SPL	$S_{\mathrm{K}} = S_{\mathrm{NK}}$	$S_{\mathrm{K}} > S_{\mathrm{NK}}$
Mobile				
0.1	17.76	18.11	18.58	19.58
0.2	19.32	19.88	21.52	22.42
0.3	20.89	21.55	24.48	25.54
0.4	22.44	23.14	27.29	28.57
0.5	24.08	24.87	30.14	31.36
Mother and Daughter				
0.1	30.14	30.37	30.74	33.88
0.2	33.08	33.34	34.01	38.29
0.3	35.57	35.62	36.65	40.71
0.4	37.46	37.48	38.88	42.82
0.5	39.34	39.34	41.04	44.97
Stefan				
0.1	19.73	19.87	19.78	19.97
0.2	21.37	21.56	22.73	22.72
0.3	23.07	23.22	25.52	25.87
0.4	24.77	24.79	27.92	28.50
0.5	26.47	26.54	30.33	30.89
Susie				
0.1	30.01	30.37	31.19	30.58
0.2	32.82	32.92	34.37	34.78
0.3	34.86	34.82	36.80	37.32
0.4	36.45	36.42	38.82	39.48
0.5	37.99	38.01	40.75	41.46

4.5.1 MC-BCS-SPL with Equal Subrate

The simplest situation is when all frames of a GOP, including the key frame, have an identical subrate. In this case, the average subrate per frame is $S = S_{\mathrm{K}} = S_{\mathrm{NK}}$. The PSNR performance of MC-BCS-SPL is measured at various subrates and compared with intraframe BCS-SPL and 3D-BCS-SPL in Figures 4.6 and 4.7. In these graphs, we see that 3D-BCS-SPL yields similar or slightly superior performance as compared to intraframe BCS-SPL; this is as expected because the 3D reconstruction exploits what frame-to-frame correlation exists in the sequence. However, the advantage is not very large in most instances, so it would appear insufficient to reconstruct based simply on frame-to-frame correlation.

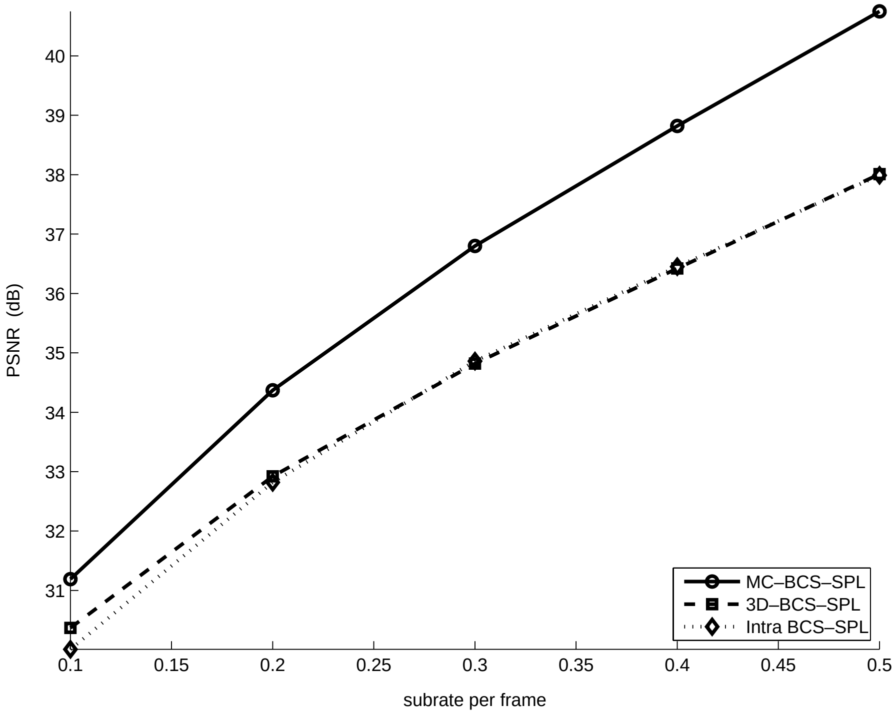


Fig. 4.6 Performance of MC-BCS-SPL on the "Susie" sequence for equal subrate, $S_{\mathrm{K}} = S_{\mathrm{NK}}$. PSNR is averaged over all frames of the sequence.

On the other hand, ME/MC in MC-BCS-SPL can largely track object motion and, hence, can increase compressibility in the sense of (2.4) in the resulting residual to yield higher-quality reconstruction. It is notable that MC-BCS-SPL achieves a 3–5 dB gain at the higher subrates for some sequences ("Coastguard," "Foreman," "Mobile," "Stefan"), as can be observed in Tables 4.2 and 4.3.

4.5.2 MC-BCS-SPL with Key Frames of Increased Subrate

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 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

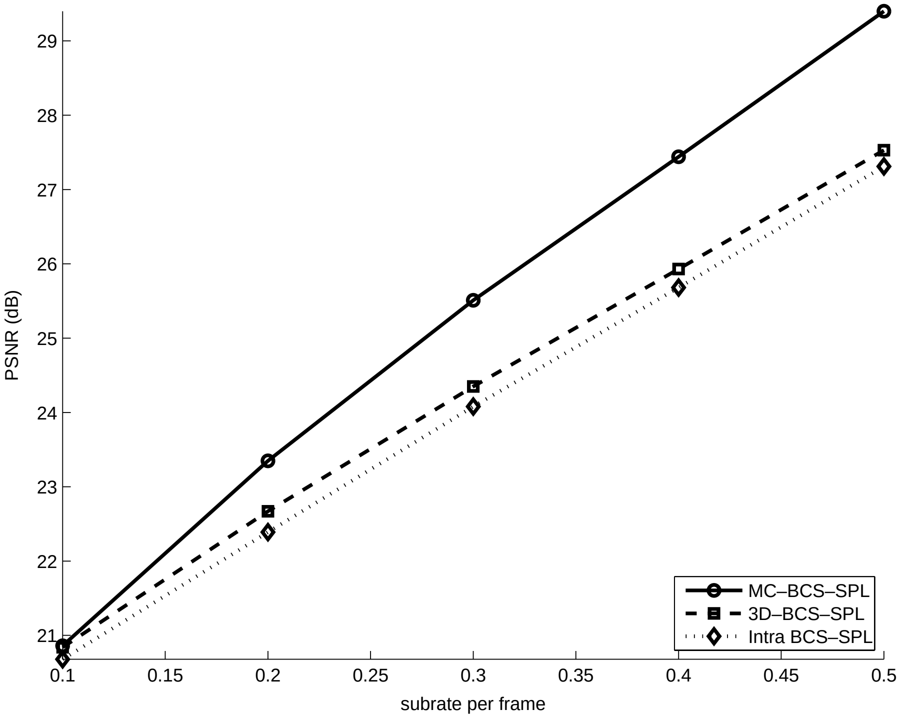


Fig. 4.7 Performance of MC-BCS-SPL on the “Football” sequence for equal subrate, $S_{\mathrm{K}} = S_{\mathrm{NK}}$. PSNR is averaged over all frames of the sequence.

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. In this situation, we set a target average subrate per frame to be S — we will have S range from 0.1 to 0.5 in the experiments. We then set the key-frame subrate to be incrementally higher than S; i.e., we use

$$S_{\mathrm{K}} = S + 0.4, \tag{4.7}$$

which we have found works well in practice. Then, the corresponding non-key subrate needed to produce the desired target S is

$$S_{\mathrm{NK}} = \frac{S \cdot P - S_{\mathrm{K}}}{P - 1}, \tag{4.8}$$

where P is the GOP size.

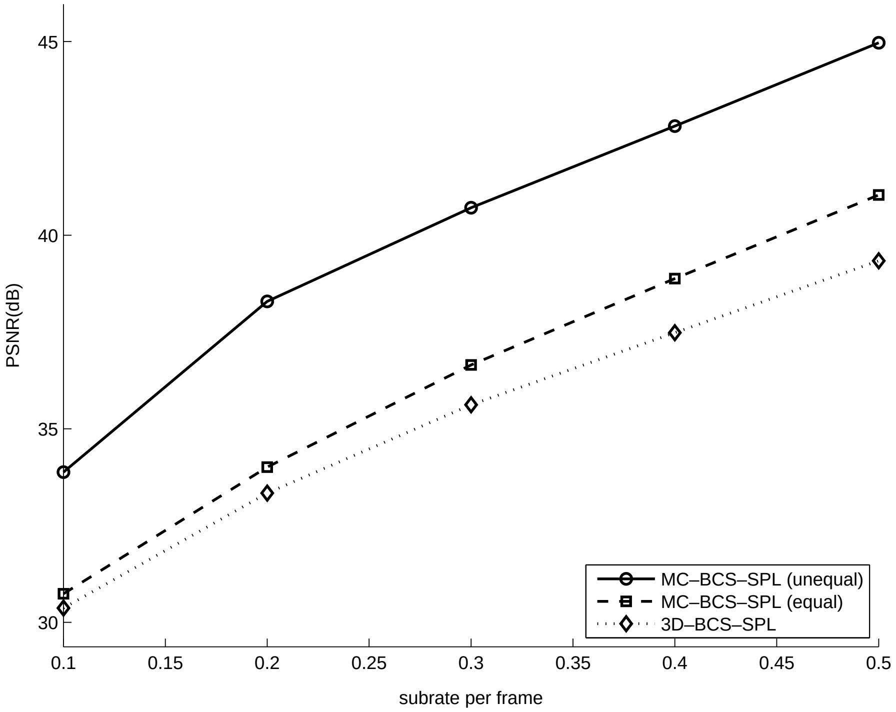


Fig. 4.8 Performance of MC-BCS-SPL on the "Mother and Daughter" sequence for unequal subrate; PSNR is averaged over all frames of the sequence. "MC-BCS-SPL (unequal)" refers to MC-BCS-SPL with $S_{\mathrm{K}} > S_{\mathrm{NK}}$; "MC-BCS-SPL (equal)" refers to MC-BCS-SPL with $S_{\mathrm{K}} = S_{\mathrm{NK}}$.

Figures 4.8 and 4.9 depict the performance of MC-BCS-SPL with increased S_{K}. In these graphs, MC-BCS-SPL with increased S_{K} is compared to MC-BCS-SPL with $S_{\mathrm{K}} = S_{\mathrm{NK}}$, i.e., the "equal subrate" paradigm as well as to 3D-BCS-SPL (intraframe 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 intraframe BCS-SPL in the previous, equal-subrate experiments.

Additionally, visual quality of the intraframe BCS-SPL and MC-BCS-SPL recoveries of the "Football" and "Susie" sequences are

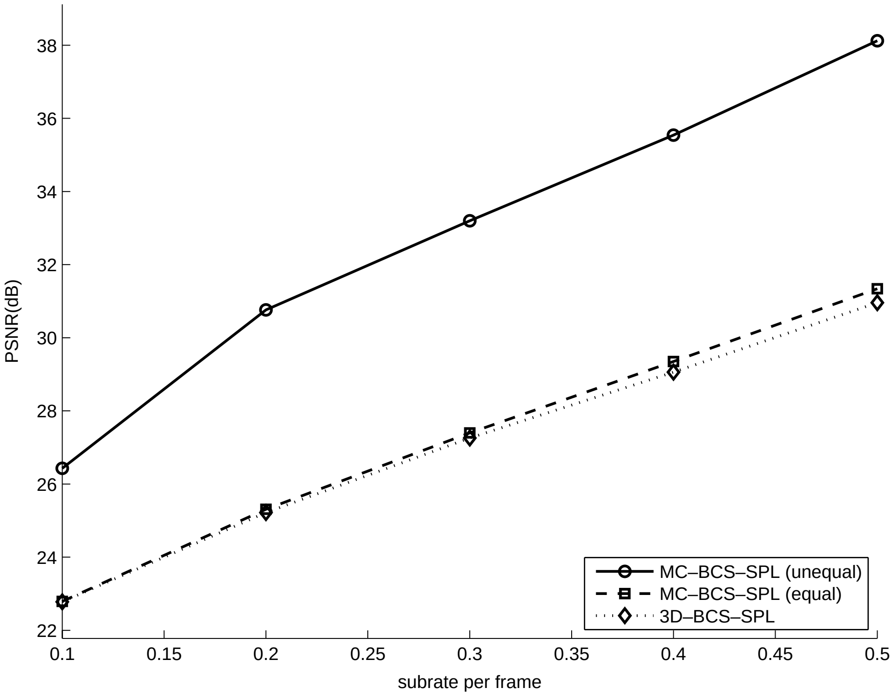
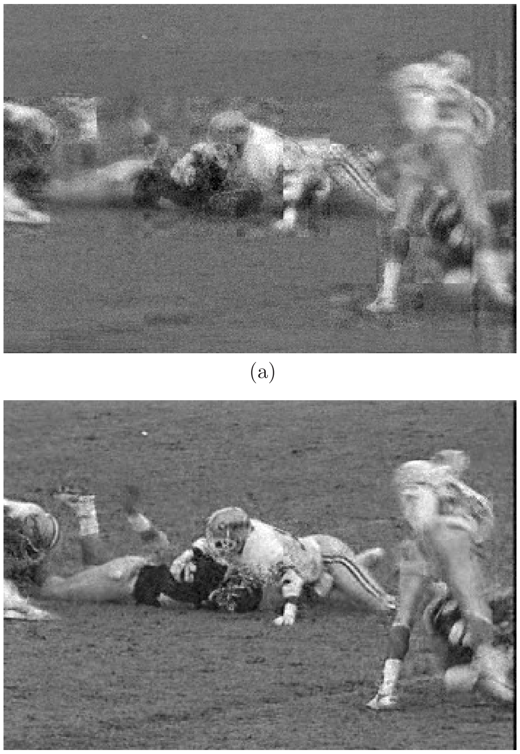


Fig. 4.9 Performance of MC-BCS-SPL on the "Hall Monitor" sequence for unequal subrate; PSNR is averaged over all frames of the sequence. "MC-BCS-SPL (unequal)" refers to MC-BCS-SPL with $S_{\mathrm{K}} > S_{\mathrm{NK}}$; "MC-BCS-SPL (equal)" refers to MC-BCS-SPL with $S_{\mathrm{K}} = S_{\mathrm{NK}}$.

compared in Figures 4.10 and 4.11 for the center frame ($p = 4$) of the first GOP. As can be seen, MC-BCS-SPL works especially well on static parts in the image due to ME/MC successfully reducing the residual energy between consecutive frames in such static regions. For more dynamic sequences, although MC-BCS-SPL still results in some degree of "mosquito noise" on moving objects, it provides visual quality significantly superior to that of intraframe BCS-SPL.

4.6 Perspectives

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,

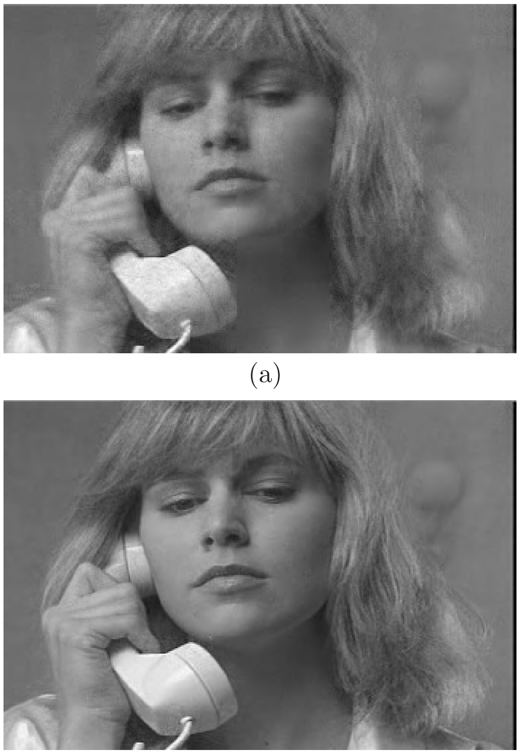


(a)

(b)

Fig. 4.10 Reconstructed center frame for “Football.” (a) Intraframe BCS-SPL with $S = 0.3$, 24.02 dB; (b) MC-BCS-SPL with unequal subrate and $S = 0.3$ ($S_{\mathrm{K}} = 0.7$, and $S_{\mathrm{NK}} = 0.24$).

we focus 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 results from exploiting the frame-to-frame motion of objects. Incorporating reconstruction from an ME/MC-based residual, the MC-BCS-SPL technique discussed here alternatively reconstructs frames 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 achieves significantly higher quality than does either intraframe BCS-SPL reconstruction or a 3D variant of BCS-SPL that features



(a)

(b)

Fig. 4.11 Reconstructed center frame for “Susie.” (a) Intraframe BCS-SPL with $S = 0.3$, 32.48 dB; (b) MC-BCS-SPL with unequal subrate and $S = 0.3$ ($S_{\mathrm{K}} = 0.7$, and $S_{\mathrm{NK}} = 0.24$).

temporal decorrelation merely in the form of a motion-agnostic 3D transform.

The experimental results above have focused on comparing MC-BCS-SPL to its intraframe and 3D variants. Subsequently, we will compare the performance of MC-BCS-SPL to several other alternative strategies for the CS of video, including the prominent benchmarks Modified-CS-Residual [81] and *k–t* FOCUSS [74, 75]. However, before we do, we first introduce an enhanced, multihypothesis approach for generating motion-based predictions, the topic that we take up next.

5

Multihypothesis Prediction for Compressed Sensing of Video

Previously, we explored the compressed sensing (CS) reconstruction of video with a focus on techniques that form an explicit prediction using motion estimation (ME) and motion compensation (MC). One form of MC widely employed in traditional video coding is that of multihypothesis (MH) prediction [112, 57] in which multiple, distinct predictions are created and then combined to yield a composite prediction superior to any of the constituent single-hypothesis (SH) predictions. Our discussion so far has been largely limited to SH prediction for CS, although [34] did propose a form of MH-based reconstruction.

In [118], an alternative strategy for incorporating MH prediction into the CS reconstruction of video so as to increase quality over equivalent SH-driven reconstruction was proposed. Central to [118] is a formulation of the MH prediction process in the domain of random CS projections; as this formulation results in an ill-posed optimization, [118] resorts to Tikhonov regularization [117] which is widely used to yield tractable solutions to such ill-posed problems. We now describe this Tikhonov-based regularization, and we compare it experimentally against the alternative strategy enforcing an ℓ_1-based sparsity constraint on the MH predictions considered in [34]. We find that the Tikhonov-regularized approach usually yields significantly superior reconstruction, particularly when the video frames are acquired at very low subrate. As before, we use block-based CS measurement of video frames and the block-based compressed sensing with smooth projected Landweber reconstruction (BCS-SPL) still-image CS reconstruction as the foundation upon which the video CS reconstruction is built.

5.1 Prediction Strategies for Residual Reconstruction

As discussed previously, residual reconstruction seeks a more compressible representation (in the sense of (2.4)) of a given signal by recovering the difference between the signal and some prediction. The philosophy is very similar to that of differential pulse code modulation (DPCM) in traditional source coding — if a prediction is similar to the signal it is intended to approximate, then the value of the residual over most of the support is insignificant in magnitude. In traditional video coding (e.g., the MPEG standards [67, 68]), this technique is used extensively to create low-energy residual frames which are then fed into a still-image coder.

Residual reconstruction can be easily integrated into the CS paradigm because it requires no change on the part of the signal acquisition and has a simple implementation on the reconstruction side. Suppose that we have a video frame $\mathbf{x}$ with a measurement basis $\mathbf{\Phi}$ such that measurements $\mathbf{y}$ are calculated via

$$\mathbf{y} = \mathbf{\Phi}\mathbf{x}. \tag{5.1}$$

If we are given some kind of prediction of $\mathbf{x}$ in the ambient domain of $\mathbf{x}$ — namely, $\tilde{\mathbf{x}}$, which we hope satisfies $\tilde{\mathbf{x}} \approx \mathbf{x}$ — then we can find the residual $\mathbf{r}$ between the two frames as $\mathbf{r} = \mathbf{x} - \tilde{\mathbf{x}}$. Because $\mathbf{y}$ is acquired simply by taking the inner products of $\mathbf{x}$ with the rows of $\mathbf{\Phi}$, the projection of $\mathbf{r}$ into the measurement basis is

$$\mathbf{q} = \mathbf{\Phi}\mathbf{r} = \mathbf{\Phi}\left(\mathbf{x} - \tilde{\mathbf{x}}\right) = \mathbf{y} - \mathbf{\Phi}\tilde{\mathbf{x}}. \tag{5.2}$$

Because of the linear nature of the signal-sampling process, a simple subtraction of a projection of $\tilde{\mathbf{x}}$ provides us with a projected residual signal at the reconstruction side without changing our signal-acquisition procedure. As demonstrated in Figure 4.1, the residual $\mathbf{r}$ will be more amenable to CS reconstruction because it is more compressible (in the sense of (2.4)) than $\mathbf{x}$ itself, leading to the final reconstruction of $\mathbf{y}$ being

$$\hat{\mathbf{x}} = \tilde{\mathbf{x}} + \mathrm{Reconstruct}(\mathbf{q}, \mathbf{\Phi}), \tag{5.3}$$

where Reconstruct$(\cdot)$ is some suitable CS reconstruction. In the residual reconstruction used by [81, 101, 127], the prediction $\tilde{\mathbf{x}}$ of the current frame is derived from the reference frame or frames using a least-squares or Kalman-filtering prediction. On the other hand, in motion-compensated BCS-SPL (MC-BCS-SPL) [90], as well as in focal underdetermined system solver in *k–t* space (*k–t* FOCUSS) [74, 75] and distributed compressed video sensing (DISCOS) [34], prediction is driven by an MC of reference frames. In these latter strategies, ME produces a motion-vector field which attempts to match the reference frames to the current frame.

That is, the key to the successful use of ME/MC in residual reconstruction is to create an MC prediction $\tilde{\mathbf{x}}$ from a given reference frame $\mathbf{x}_{\mathrm{ref}}$ that is as close as possible to $\mathbf{x}$ such that the resulting residual, $\mathbf{r}$, is highly compressible in the sense of (2.4). Thus, the goal is to carry out the optimization,

$$\tilde{\mathbf{x}} = \arg \min_{\mathbf{p} \in \mathcal{P}(\mathbf{x}_{\mathrm{ref}})} \|\mathbf{x} - \mathbf{p}\|_2^2, \tag{5.4}$$

where $\mathcal{P}(\mathbf{x}_{\mathrm{ref}})$ is the set of all possible motion-compensated frames producible from the given reference frame under the particular ME/MC model used. However, in the CS paradigm, the creation of the prediction $\tilde{\mathbf{x}}$ via ME/MC occurs during CS reconstruction; as a consequence, $\mathbf{x}$ is unknown, and (5.4) cannot be implemented as written. This is contrary to traditional source coding of video in which $\mathbf{x}$ is known to the video encoder such that it can effectively solve (5.4) directly.

There are two strategies to approximate (5.4) using only information known to the CS reconstruction. The first would be to approximate $\mathbf{x}$ with an initial CS reconstruction from $\mathbf{y}$ and use the resulting approximation to $\mathbf{x}$ to drive the prediction process; i.e.,

$$\tilde{\mathbf{x}} = \arg \min_{\mathbf{p} \in \mathcal{P}(\mathbf{x}_{\mathrm{ref}})} \| \mathrm{Reconstruct}(\mathbf{y}, \mathbf{\Phi}) - \mathbf{p}\|_2^2, \tag{5.5}$$

where Reconstruct$(\cdot)$ is some suitable CS reconstruction. The resulting $\tilde{\mathbf{x}}$ is then used in (5.3) to form $\hat{\mathbf{x}}$ using a CS reconstruction from the measurement-domain residual, $\mathbf{q} = \mathbf{y} - \mathbf{\Phi}\tilde{\mathbf{x}}$. This approach was used in MC-BCS-SPL [90] as well as in k–t FOCUSS [74, 75].

An alternative strategy proposed in [118] is to recast the optimization of (5.5) from the ambient signal domain of $\mathbf{x}$ into the measurement domain of $\mathbf{y}$; specifically,

$$\begin{aligned}\tilde{\mathbf{x}} &= \arg \min_{\mathbf{p} \in \mathcal{P}(\mathbf{x}_{\mathrm{ref}})} \|\mathbf{\Phi}(\mathbf{x} - \mathbf{p})\|_2^2 \\ &= \arg \min_{\mathbf{p} \in \mathcal{P}(\mathbf{x}_{\mathrm{ref}})} \|\mathbf{y} - \mathbf{\Phi}\mathbf{p}\|_2^2. \end{aligned} \tag{5.6}$$

Although (5.6) reformulates the search for the prediction into the measurement domain, the Johnson-Lindenstrauss (JL) lemma [1, 29, 73] suggests that the solution of (5.6) will likely coincide with that of (5.4). In brief, the JL lemma holds that L points in $\mathbb{R}^N$ can be projected into a K-dimensional subspace while approximately maintaining pairwise distances as long as $K \geq O(\log L)$. As a consequence, the $\tilde{\mathbf{x}}$ closest to $\mathbf{x}$ in (5.4) should map to the $\mathbf{\Phi}\tilde{\mathbf{x}}$ that is closest to $\mathbf{y}$ in (5.6), provided that the number of candidates searched in the minimizations is not too large.

Our experimental observations reveal that the measurement-domain prediction of (5.6) provides better predictions in general than the ambient-domain strategy represented by (5.5) (see Figure 5.1). This is due to the fact that (5.5) uses only a noisy approximation to $\mathbf{x}$, whereas the JL lemma suggests that (5.6) should nearly duplicate the targeted procedure of (5.4). As a consequence, we focus on measurement-domain predictions in the form of (5.6) in the following.

5.2 SH Frame Prediction for CS Reconstruction

In traditional video coding, frame predictions are calculated using ME/MC from temporally neighboring frames which are likely to have similar content to the target frame. Specifically, the frame at time t to be predicted, $\mathbf{x}_t$, is split into blocks of size $B \times B$. The chosen reference frame or frames are then searched within a spatial region surrounding the location of the target block within $\mathbf{x}_t$. The single best-matching

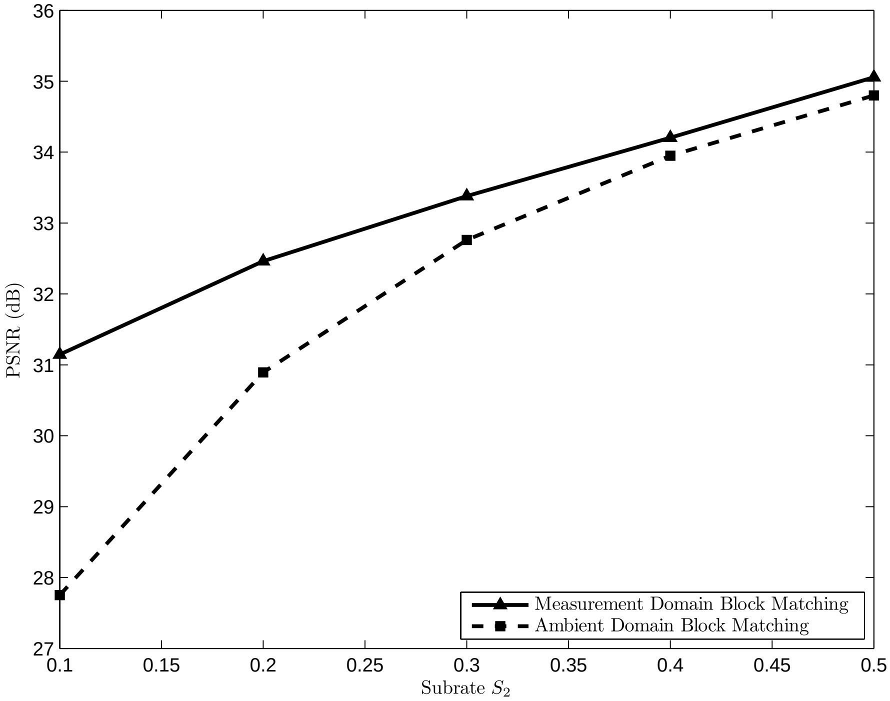


Fig. 5.1 Recovery of frame $\mathbf{x}_2$ of "Foreman" using frame $\mathbf{x}_1$ as reference.

block, chosen according to some distortion measure, in the reference frames then forms the prediction of the target block. This is known as SH prediction in the video-coding community since a single, best-matching hypothesis prediction (a block in one of the reference frames, in this case) is chosen to represent the target block.

In the CS reconstruction of video wherein each frame has been acquired using BCS applied frame by frame, the ensemble of measurements for frame $\mathbf{x}_t$ is

$$\mathbf{y}_{t,i} = \mathbf{\Phi}\mathbf{x}_{t,i}, \tag{5.7}$$

where i is a block index. In order to create a prediction of a given block, $\mathbf{x}_{t,i}$, we recast (5.6) as

$$\tilde{\mathbf{x}}_{t,i} = \arg\min_{\mathbf{p}\in\mathcal{H}_{t,i}} \|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{p}\|_2^2, \tag{5.8}$$

where $\mathcal{H}_{t,i}$ is the set of blocks culled from the reference frame or frames within the search space given for block $\mathbf{x}_{t,i}$ (typically a rectangular region about the spatial location of $\mathbf{x}_{t,i}$ in the frame).

5.3 MH Frame Prediction for CS Reconstruction

Video coding has long exploited MH methods to improve video-coding quality [57, 112]; common forms include subpixel-accurate MC [56], overlapped-block MC [93, 95], bidirectional MC (B-frames), and long-term-memory MC [134]. These techniques can be viewed as trade-offs specific to a bitrate-limited environment; that is, these techniques impose specific structures on the hypotheses that form the ultimate prediction in order to limit the amount of additional motion-vector rate overhead entailed by multiple predictions of a single block. However, in the context of CS reconstruction, the MH predictions are all calculated at the reconstruction side of the system, there is no associated rate burden, and we are able to consider more intensive forms of MH prediction, essentially combining all the best hypotheses available from the reference frames without the imposition of rate-limiting structure.

As proposed in [118], for an MH CS reconstruction, the goal is to reformulate (5.4) so that, instead of choosing a single hypothesis, we find an optimal linear combination of all hypotheses contained in the search set; i.e, (5.4) becomes

$$\mathbf{w}_{t,i} = \arg\min_{\mathbf{w}} \|\mathbf{x}_{t,i} - \mathbf{H}_{t,i}\mathbf{w}\|_2^2, \tag{5.9}$$

$$\tilde{\mathbf{x}}_{t,i} = \mathbf{H}_{t,i}\mathbf{w}_{t,i}, \tag{5.10}$$

where we have also recast (5.4) for block-based prediction with i being the block index. Here, $\mathbf{H}_{t,i}$ is a matrix of dimensionality $B^2 \times K$ whose columns are the rasterizations of the possible blocks within the search space of the reference frames, and $K = |\mathcal{H}_{t,i}|$. In this context, $\mathbf{w}_{t,i}$ is a column vector which represents the optimal linear combination of the columns of $\mathbf{H}_{t,i}$; the solution of this optimization can be calculated as a simple least-squares (LSQ) problem.

Of course, in the case of CS reconstruction, (5.10), like (5.4) before it, cannot be implemented — we cannot calculate $\mathbf{w}_{t,i}$ directly because we do not have access to $\mathbf{x}_{t,i}$; we have only its measurements, $\mathbf{y}_{t,i}$. We thus adopt the measurement-domain approach of (5.8), modifying it to the MH case. However, this makes the optimization a much more difficult, ill-posed problem, because we have to calculate the optimal linear combination within the projected space of $\mathbf{\Phi}$; i.e., combining (5.8) and (5.9) yields

$$\hat{\mathbf{w}}_{t,i} = \arg\min_{\mathbf{w}} \|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{H}_{t,i}\mathbf{w}\|_2^2. \tag{5.11}$$

In general, $\mathbf{w}_{t,i} \neq \hat{\mathbf{w}}_{t,i}$ unless $\mathbf{\Phi}$ is square, which is necessarily not the case for CS. The ill-posed nature of this problem requires some form of regularization of the LSQ optimization.

The most common approach to regularizing an LSQ problem is Tikhonov regularization [117] which imposes an ℓ_2 penalty on the norm of $\hat{\mathbf{w}}_{t,i}$,

$$\hat{\mathbf{w}}_{t,i} = \arg\min_{\mathbf{w}} \|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{H}_{t,i}\mathbf{w}\|_2^2 + \lambda^2\|\mathbf{\Gamma}\mathbf{w}\|_2^2, \tag{5.12}$$

where $\mathbf{\Gamma}$ is known as the Tikhonov matrix. The $\mathbf{\Gamma}$ term allows the imposition of prior knowledge on the solution; in some contexts, it may make sense to use a high-pass or difference operator for $\mathbf{\Gamma}$ to obtain a smooth result, or, in others, to set $\mathbf{\Gamma} = \mathbf{I}$ to impose an energy constraint on the solution. In our case, we take the approach that hypotheses which are the most dissimilar from the target block should be given less weight than hypotheses which are most similar. Specifically, we propose a diagonal $\mathbf{\Gamma}$ in the form of

$$\mathbf{\Gamma} = \begin{bmatrix} \|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{h}_1\|_2^2 & & 0 \\ & \ddots & \\ 0 & & \|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{h}_K\|_2^2 \end{bmatrix}, \tag{5.13}$$

where $\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_K$ are the columns of $\mathbf{H}_{t,i}$. With this structure, $\mathbf{\Gamma}$ penalizes weights of large magnitude assigned to hypotheses which have a significant distance from $\mathbf{y}_{t,i}$ when projected into the measurement domain. For each block, then, $\hat{\mathbf{w}}_{t,i}$ can be calculated directly by the usual Tikhonov solution,

$$\hat{\mathbf{w}}_{t,i} = ((\mathbf{\Phi}\mathbf{H}_{t,i})^{\mathrm{T}}(\mathbf{\Phi}\mathbf{H}_{t,i}) + \lambda^2\mathbf{\Gamma}^{\mathrm{T}}\mathbf{\Gamma})^{-1}(\mathbf{\Phi}\mathbf{H}_{t,i})^{\mathrm{T}}\mathbf{y}_{t,i}. \tag{5.14}$$

In this formulation, λ is a scale factor that controls the relative effect of the Tikhonov-regularization term in the optimization of (5.12). The choice of λ can have a large effect on the performance of the regularization, so it is important to find a value which imposes an adequate level of regularization without causing $\|\mathbf{y}_{t,i} - \mathbf{\Phi}\mathbf{H}_{t,i}\mathbf{w}\|_2^2$ to become too large. We found in practice that, over a large set of different frames, a value of $\lambda \in [0.1, 0.3]$ provided the best results; consequently, we use $\lambda = 0.25$ from this point on.

5.4 An Alternate ℓ_1-Based MH Regularization

The DISCOS algorithm of [34] can be viewed, in essence, as an alternative to the Tikhonov regularization used in (5.12)–(5.13). Specifically, it was assumed in [34] that the MH weights $\mathbf{w}_{t,i}$ in (5.9) are sparse; i.e., only a relative few of the possible hypotheses in $\mathbf{H}_{t,i}$ should contribute the prediction in (5.10). As a consequence of this assumption, [34] imposes an ℓ_1 penalty term on $\hat{\mathbf{w}}_{t,i}$ in the form of

$$\hat{\mathbf{w}}_{t,i} = \arg\min_{\mathbf{w}} \|\mathbf{\Phi}\mathbf{H}_{t,i}\mathbf{w} - \mathbf{y}_{t,i}\|_2^2 + \lambda \|\mathbf{w}\|_1 . \tag{5.15}$$

The intuition here is that only a few blocks within the search space should contribute significantly to the linear combination; this is reflective of the structure often imposed on MH prediction in traditional video coding, structure that is necessary to limit motion-vector bitrate overhead. However, in the context of CS reconstruction, a regularization enforcing sparsity is needlessly restrictive on the structure of $\hat{\mathbf{w}}_{t,i}$, which can potentially result in lower prediction quality. Furthermore, Tikhonov regularization in the form of (5.12)–(5.13) is a much more amenable solution than ℓ_1 regularization in terms of scalability and computation time, as well. That is, with the ℓ_1 penalty, the optimization in (5.15) is approached as a traditional CS problem using some generic CS solver. Such CS solvers are based on some kind of iterative search to arrive at a final solution and are thus strictly linear in the computation. Yet, the weights $\hat{\mathbf{w}}_{t,i}$ must be calculated for every block in $\mathbf{x}_t$, so the computation time can be very significant when using these linear solvers. On the other hand, the Tikhonov regularization we propose can be calculated directly at each block with simple matrix math in the form of (5.14).

We note that [99] also features an ℓ_1-regularized multihypothesis prediction. Briefly, [99] culls a selection of candidate blocks from previously reconstruction frames into an overcomplete dictionary which is then used in the ℓ_1-driven CS reconstruction of the current block. In essence, this process is equivalent to (5.15); the key difference with respect to [34] is that [99] uses the resulting prediction ($\tilde{\mathbf{x}}_{t,i}$ in (5.10)) as the reconstruction of current block directly, rather than employing a subsequent residual reconstruction as in (5.3). Finally, [60] is similar to [99], differing mainly in that the overcomplete dictionary is iteratively refined via multiple reconstructions of a frame.

A focus of experimental results which follow is an investigation into the relative performance of the Tikhonov-regularization approach to MH prediction discussed above as opposed to that of the ℓ_1-based approach of [34]. We explore these experimental results next.

5.5 Experimental Observations

Initially, we consider the first two consecutive frames, $\mathbf{x}_1$ and $\mathbf{x}_2$, of a given video sequence — the first frame, $\mathbf{x}_1$, is used as a reference frame, while the second frame, $\mathbf{x}_2$, is the "test frame" used to measure reconstruction performance. In all cases, the reference frame has a relatively high subrate of $S_1 = 0.5$ and is reconstructed using BCS-SPL. On the other hand, the test frame has a range of subrates, $S_2 \leq S_1$. This disparity in subrates is intended to reflect the situation in which the video sequence is acquired with relatively high subrates for certain "key frames" which anchor the ME/MC-driven reconstruction process for one or more intervening "non-key frames" (e.g., [34, 90]). Throughout, we use a block size of $B = 16$ for BCS with an orthonormalized dense Gaussian measurement matrix, and a discrete wavelet transform (DWT) with four levels of decomposition as the sparsity basis for BCS-SPL reconstruction.

The reconstructed reference frame is used to create a prediction of each block of the test frame; afterward, residual reconstruction (i.e., (5.2)–(5.3)) of the test frame is conducted. We investigate several prediction strategies as discussed above. Namely, we consider the Tikhonov-regularized MH prediction of (5.12)–(5.14) with $\lambda = 0.25$. We also use the ℓ_1-regularized MH prediction of (5.15) as proposed in [34]. For this ℓ_1-regularized MH prediction, we use gradient projection for sparse reconstruction (GPSR) [47] to find the weights, although other CS solvers could be used. For comparison, we also consider performance of the SH prediction of (5.8), as well as the straightforward BCS-SPL reconstruction of the test frame independently of the reference frame, which we refer to as "independent reconstruction" in the results to follow. In all cases, a spatial window size of ±15 pixels about the current block is used as the search space for finding the hypotheses for both the SH and MH predictions.

5.5.1 Measurement-Domain and Ambient-Domain Prediction

Above, we discussed two options for creation of a prediction of the test frame — an ambient-domain strategy via (5.5) driven by an initial CS reconstruction of the test frame, and a measurement-domain procedure via (5.6). We now compare the relative performance, as measured in peak signal-to-noise ratio (PSNR), of these two prediction approaches. For simplicity, we consider SH prediction in both cases; i.e., the measurement-domain strategy consists of the block-based SH prediction of (5.8), while the corresponding ambient-domain approach is a blocked version of (5.5). The resulting reconstruction performance for the "Foreman" sequence over a range of test-frame subrates is presented in Figure 5.1. The measurement-domain strategy significantly outperforms the equivalent ambient-domain approach at the lower subrates. At the higher subrates, the gain is less important due to the fact that the initial CS reconstruction used in (5.5) is of fairly high quality itself. Nonetheless, the measurement-domain approach is superior over the entire range of subrates examined; as a consequence, we will focus on it exclusively in the remainder of these results.

5.5.2 Single Reference Frame

We now consider the recovery of test frame $\mathbf{x}_2$ using $\mathbf{x}_1$ as reference and evaluate the various measurement-domain prediction approaches discussed above. The PSNR performance of the test-frame recovery as

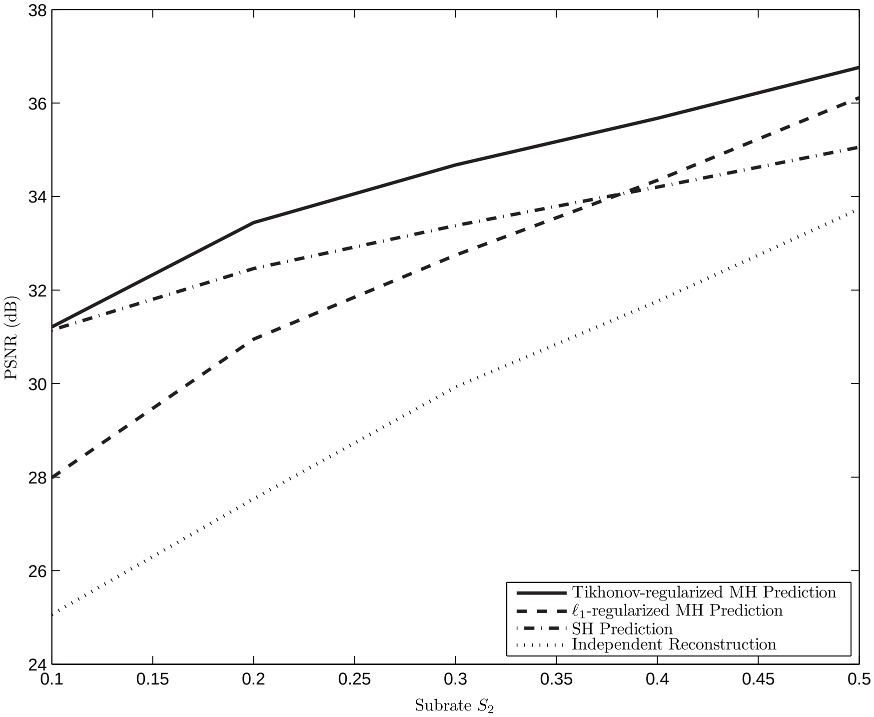


Fig. 5.2 Recovery of frame $\mathbf{x}_2$ of “Foreman” using frame $\mathbf{x}_1$ as reference.

the subrate, S_2, for the test frame varies is presented in Figure 5.2. As can be seen in Figure 5.2, the proposed Tikhonov-regularized MH prediction provides significantly superior recovery for $\mathbf{x}_2$ at low subrates as compared to the ℓ_1-regularized prediction of [34]. For higher subrates near $S_2 \approx 0.5$, the performance of the ℓ_1 regularization is generally more competitive.

5.5.3 Bidirectional Reference Frames

Bidirectional prediction is often used with ME/MC in traditional video coding to overcome occlusion by deriving temporal predictions from both future as well as past reference frames. Such bidirectional prediction is easily incorporated into the ME/MC-based CS reconstructions evaluated in the previous section. Specifically, we now consider the situation in which we reconstruct frame $\mathbf{x}_2$ using both frames $\mathbf{x}_1$ and

$\mathbf{x}_3$ as reference. This bidirectional prediction has the effect of simply doubling the size of the hypothesis set, which can significantly increase the computation time of the prediction, but can overcome certain forms of occlusion.

We repeat the experiment of the previous section for the bidirectional case, presenting the results in Figure 5.3. In this case, the second reference frame, $\mathbf{x}_3$, was reconstructed with a subrate of $S_3 = S_1 = 0.5$ using BCS-SPL independently from the other two frames.

In the bidirectional case, we see that the relative performance of each of the frame-prediction techniques is similar to that seen for the single-reference-frame case. However, the use of bidirectional prediction increases the PSNR by about 0.5 dB for the MH methods. We still have that the Tikhonov-regularized MH prediction always outperforms both the ℓ_1-regularized MH prediction and as well as the SH prediction.

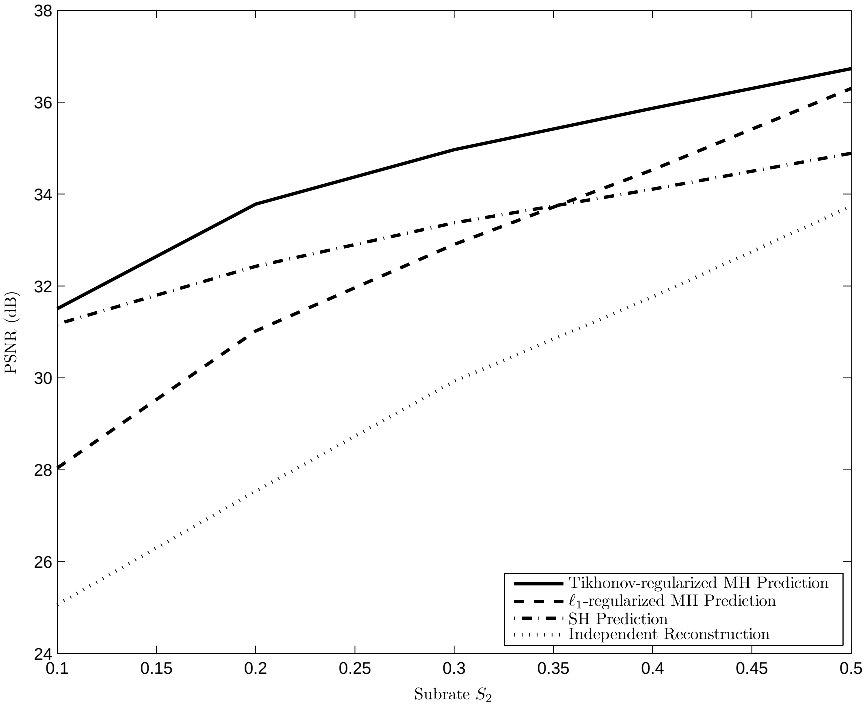


Fig. 5.3 Recovery of frame $\mathbf{x}_2$ of “Foreman” using frames $\mathbf{x}_1$ and $\mathbf{x}_3$ as reference.

5.6 Comparison of Various CS Techniques for Video

We now present a comprehensive comparison between several CS reconstruction algorithms for video. We use the first 88 frames of the "Foreman," "Coastguard," "Hall Monitor," and "Mother and Daughter" sequences. In all cases, we use a group of pictures (GOP) size of $P = 8$ frames with key frames starting each GOP having a subrate of $S_{\mathrm{K}} = 0.7$. The intervening non-key frames have subrate S_{NK} varying between 0.1 and 0.5.

We compare a video reconstruction that couples the MH prediction described above with the BCS-SPL still-image recovery considered previously. This technique, which we refer to as multihypothesis BCS-SPL (MH-BCS-SPL), performs an iterative reconstruction of the non-key frames of the GOP based upon bidirectional MH predictions driven from the neighboring key frames. Once the entire GOP has been reconstructed, the process is repeated using the newly reconstructed frames to refine the MH predictions.

Additionally, we compare to the motion-compensated version of BCS-SPL discussed previously (i.e., MC-BCS-SPL) which was our original, single-hypothesis extension of BCS-SPL from still images to video as proposed in [90]; we use the implementation available at the BCS-SPL website.[1] Being block-based techniques, both MH-BCS-SPL as well as MC-BCS-SPL feature block-based measurement in the spatial domain applied identically to each video frame; the block size for both techniques is 16 × 16. An orthonormalized dense Gaussian measurement matrix is used for BCS, and a dual-tree discrete wavelet transform (DDWT) [78] is used as the sparsity transform.

We also compare to two prominent CS reconstruction algorithms, Modified-CS-Residual [128] and *k–t* FOCUSS [74, 75], both of which we have described previously. As discussed before, *k–t* FOCUSS uses iterative recovery with ME/MC of non-key frames from the neighboring key frames. On the other hand, Modified-CS-Residual does not employ ME/MC but rather attempts to explicitly track the sparsity pattern frame to frame. We use the implementations of *k–t* FOCUSS[2]

---

[1] `http://www.ece.msstate.edu/~fowler/BCSSPL/`.

[2] `http://bisp.kaist.ac.kr/research_02.htm`.

and Modified-CS-Residual[3] available from their respective authors. Although both *k–t* FOCUSS and Modified-CS-Residual were originally designed for the reconstruction of dynamic magnetic resonance imaging (MRI) data, they are both largely considered to be benchmark algorithms in present literature for the reconstruction of video as well. Both techniques, being oriented toward dynamic MRI, feature frame-by-frame measurement driven by a 2D full-frame Fourier transform applied identically to each frame with low-frequency coefficients benefiting from a higher subrate.

Finally, we compare to straightforward, "intraframe" reconstruction of each frame of the sequence independently from the others. We

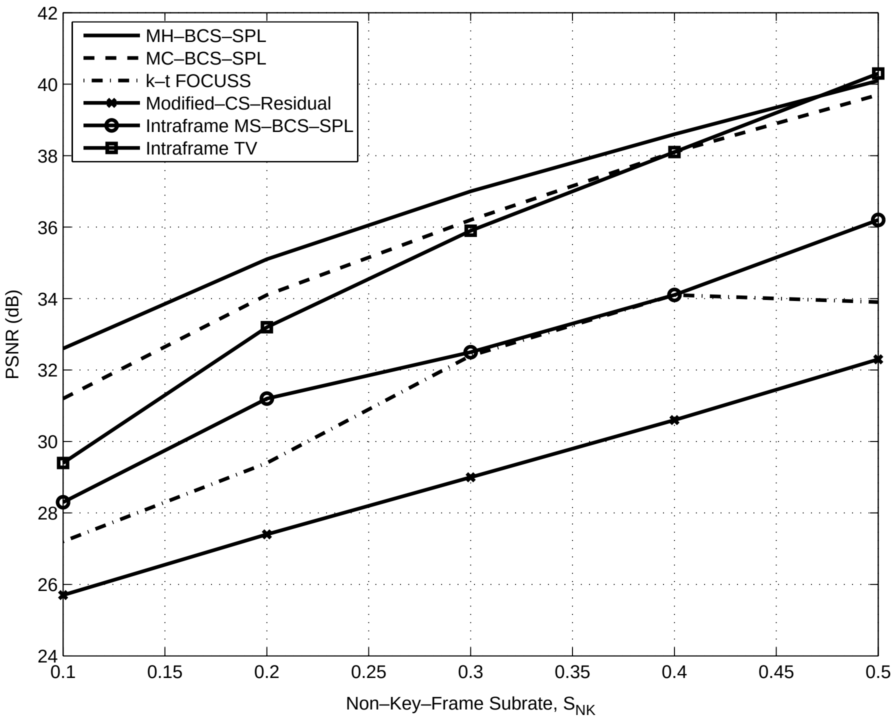


Fig. 5.4 Performance of various CS reconstruction algorithms on the 88-frame "Foreman" sequence for $S_{\mathrm{K}} = 0.7$. PSNR is averaged over all frames of the sequence.

---

[3] `http://home.engineering.iastate.edu/~luwei/modcs/`.

consider the multiscale (MS) variant of BCS-SPL discussed previously and originally proposed in [50]; in the results here, we refer to it as "intraframe MS-BCS-SPL." We also consider an intraframe implementation of total variation (TV) reconstruction [17] ("intraframe TV"). We note that, in the still-image results presented earlier, MS-BCS-SPL and TV outperformed other techniques in terms of reconstruction quality, with MS-BCS-SPL generally producing higher-quality reconstructions with much less computation, but TV being amenable to fast, spatial-domain measurement with a structurally random matrix (SRM). In these results, the intraframe MS-BCS-SPL features dense Gaussian block-based measurement in the wavelet domain with blocks of size 16 × 16, while intraframe TV uses a full-frame block-Hadamard SRM measurement [54].

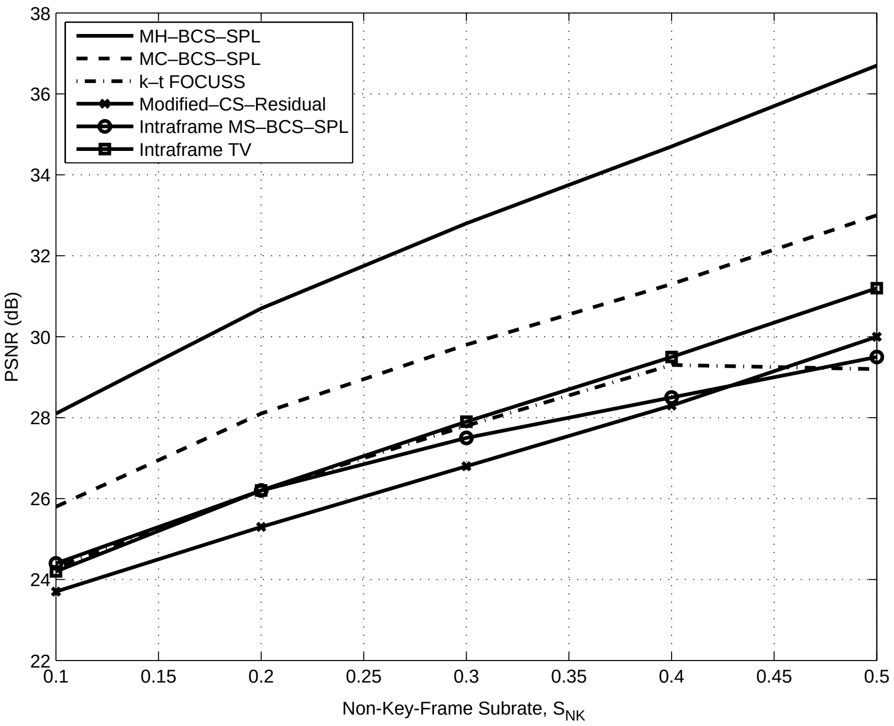


Fig. 5.5 Performance of various CS reconstruction algorithms on the 88-frame "Coastguard" sequence for $S_{\mathrm{K}} = 0.7$. PSNR is averaged over all frames of the sequence.

Although there are a number of other reconstruction algorithms for video in the literature, none of these, to our knowledge, have implementations readily available at the time of this writing. As a consequence, we present results only for those algorithms identified above.

Figures 5.4–5.7 illustrate the performance of the various reconstructions for varying non-key-frame subrate S_{NK}. Table 5.1 tabulates these same results as well. Visual results for a single frame of the "Foreman" sequence are given in Figure 5.8.

As can be seen notably in Table 5.1, the multihypothesis-driven MH-BCS-SPL almost always outperforms the other techniques considered, sometimes by as much as 2–3 dB. The performance of the other techniques is rather mixed — sometimes MC-BCS-SPL or intraframe TV will be somewhat competitive with MH-BCS-SPL for certain subrates

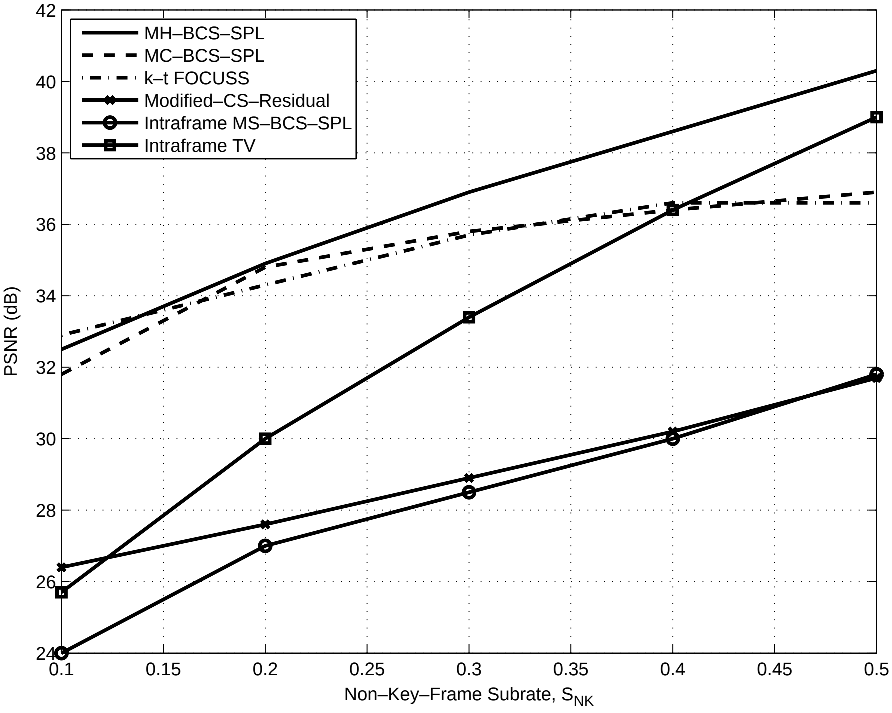


Fig. 5.6 Performance of various CS reconstruction algorithms on the 88-frame "Hall Monitor" sequence for $S_{\mathrm{K}} = 0.7$. PSNR is averaged over all frames of the sequence.

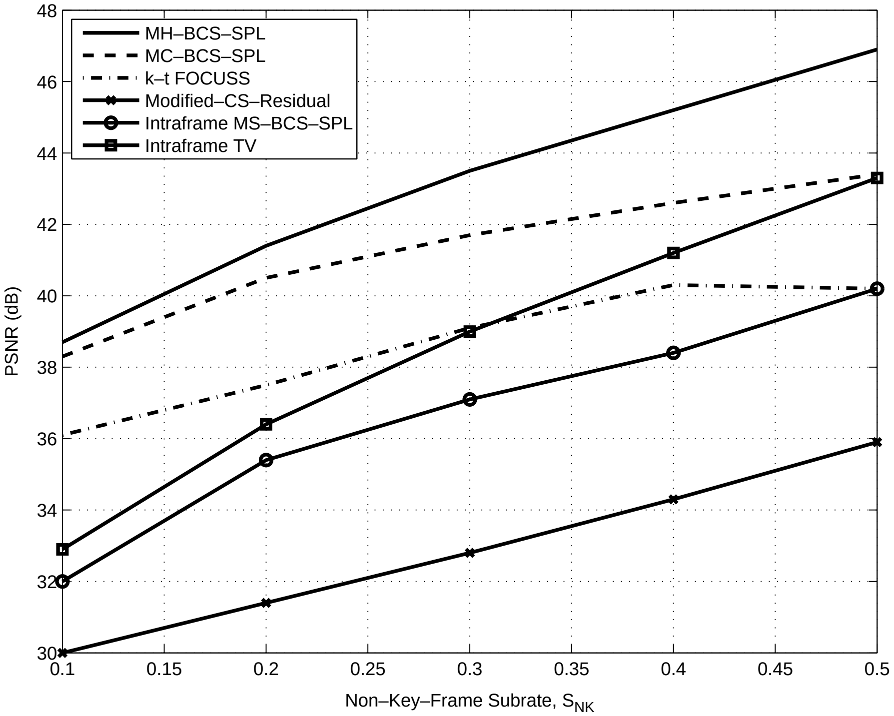


Fig. 5.7 Performance of various CS reconstruction algorithms on the 88-frame "Mother and Daughter" sequence for $S_{\mathrm{K}} = 0.7$. PSNR is averaged over all frames of the sequence.

and sequences. Additionally, the two techniques designed for dynamic MRI are typically rather distant in performance from MH-BCS-SPL with the exception of low subrates for the "Hall Monitor" sequence.

Although none of the implementations have been particularly optimized for execution speed, we present reconstruction times for the algorithms in Table 5.2. Here, we measure the typical length of time required to reconstruction one frame out of the sequence. We see that, while intraframe MS-BCS-SPL reconstruction is the fastest, intraframe TV reconstruction is the slowest, requiring some 20 minutes per frame.

5.7 Perspectives

In this discussion, we have considered how the high degree of frame-to-frame temporal correlation in video signals can be exploited to

Table 5.1. Average PSNR in dB for several video sequences; $S_{\mathrm{K}} = 0.7$, 88 frames.

	S_{NK}				
Algorithm	0.1	0.2	0.3	0.4	0.5
Foreman					
MH-BCS-SPL	**32.6**	**35.1**	**37.0**	**38.6**	40.1
MC-BCS-SPL	31.2	34.1	36.2	38.1	39.7
k–*t* FOCUSS	27.2	29.4	32.4	34.1	33.9
Modified-CS-Residual	25.7	27.4	29.0	30.6	32.3
Intraframe MS-BCS-SPL	28.3	31.2	32.5	34.1	36.2
Intraframe TV	29.4	33.2	35.9	38.1	**40.3**
Coastguard					
MH-BCS-SPL	**28.1**	**30.7**	**32.8**	**34.7**	**36.7**
MC-BCS-SPL	25.8	28.1	29.8	31.3	33.0
k–*t* FOCUSS	24.3	26.2	27.8	29.3	29.2
Modified-CS-Residual	23.7	25.3	26.8	28.3	30.0
Intraframe MS-BCS-SPL	24.4	26.2	27.5	28.5	29.5
Intraframe TV	24.2	26.2	27.9	29.5	31.2
Hall Monitor					
MH-BCS-SPL	32.5	**34.9**	**36.9**	**38.6**	**40.3**
MC-BCS-SPL	31.8	34.8	35.8	36.4	36.9
k–*t* FOCUSS	**32.9**	34.3	35.7	36.6	36.6
Modified-CS-Residual	26.4	27.6	28.9	30.2	31.7
Intraframe MS-BCS-SPL	24.0	27.0	28.5	30.0	31.8
Intraframe TV	25.7	30.0	33.4	36.4	39.0
Mother and Daughter					
MH-BCS-SPL	**38.7**	**41.4**	**43.5**	**45.2**	**46.9**
MC-BCS-SPL	38.3	40.5	41.7	42.6	43.4
k–*t* FOCUSS	36.1	37.5	39.1	40.3	40.2
Modified-CS-Residual	30.0	31.4	32.8	34.3	35.9
Intraframe MS-BCS-SPL	32.0	35.4	37.1	38.4	40.2
Intraframe TV	32.9	36.4	39.0	41.2	43.3

enhance CS recovery by forming MH predictions using a distance-weighted Tikhonov regularization to find the best linear combination of hypotheses. The MH predictions were used to create a measurement-domain residual of the frame to be recovered. Such a residual is typically more compressible — in the sense of (2.4) — than the original frame, rendering it more amenable to CS recovery. This procedure for video recovery shows a significant distortion-performance improvement over a straightforward recovery of the frames independently. We also demonstrated that distance-weighted Tikhonov regularization for MH prediction provides better performance than an equivalent ℓ_1-based

(b) MH-BCS-SPL, 37.0 dB (c) MC-BCS-SPL, 36.7 dB

(d) k-t FOCUSS, 32.7 dB (e) Modified-CS-Residual, 29.6 dB

(f) Intraframe MS-BCS-SPL, 33.1 dB (g) Intraframe TV, 36.7 dB

Fig. 5.8 Reconstructions of frame 4 of the "Foreman" sequence for $S_{\mathrm{K}} = 0.7$ and $S_{\mathrm{NK}} = 0.3$.

Table 5.2. Reconstruction time in seconds per frame (spf).

Algorithm	Time (spf)
Intraframe MS-BCS-SPL	10
k–t FOCUSS	46
MC-BCS-SPL	159
MH-BCS-SPL	324
Modified-CS-Residual	699
Intraframe TV	1223

regularization that assumes a sparse combination of hypotheses, especially at low subrates.

The MH prediction was coupled with the BCS-SPL [89] still-image recovery considered previously to create a video reconstruction employing bidirectional MH prediction from the key frames at either end of a GOP. The resulting MH-BCS-SPL technique was compared to several alternative CS reconstructions for video, including the prominent k–t FOCUSS [74, 75] and Modified-CS-Residual [128] methods that were initially designed for dynamic-MRI reconstruction. Overall, MH-BCS-SPL almost always outperformed the other reconstructions, yielding higher PSNR for a given non-key-frame subrate.

The discussion thus far has centered on the CS reconstruction of video, exploiting the fundamental concept of ME/MC-driven predictions and residual reconstruction. Next, we extend this same basic framework to a different application in which multiple views of a scene are captured from slightly different perspectives. In this multiview scenario, we find that we can employ the same residual-reconstruction paradigm to significantly improve performance over a simple, straightforward CS reconstruction of each view independently.

6

Compressed Sensing of Multiview Image and Video

The falling cost of high-quality video sensors coupled with their increasingly widespread use in surveillance, defense, and entertainment applications has led to heightened demand for multi-sensor video-acquisition systems. In surveillance applications, for example, the use of video-sensor networks has been widely investigated, but the memory and computation burden of capturing and encoding high-quality video for transmission and storage has served as an impediment to the adoption of multi-sensor technology in many applications [100]. In the area of entertainment, much work has been done recently to promote the production and consumption of 3D video content, which so far has largely taken the form of stereoscopic video-display systems with a fixed viewpoint. Future display technologies, such as holography, promise a more realistic and engaging viewing experience by permitting many different viewing angles; however, capturing such multiview-image data requires a system more sophisticated than the two-camera approach widely used today. In these and other applications, the excessively voluminous nature of multiview-image and video data causes a serious impediment to the continued development of these fields.

In the context of multiview images and multiview video, compressed sensing (CS) has the potential to greatly enhance multiview signal acquisition not only by decreasing the inherent memory cost by lowering the number of measurements taken, but also by decreasing the computational burden on the sensors. There have been several approaches proposed for CS reconstruction of multiview images (e.g., [26, 80, 130]). Although it is assumed that the images in the multiview set are acquired independently from one another (perhaps with multiple single-pixel cameras), the reconstruction typically targets a joint recovery of the entire set of multiview images, capitalizing on the joint sparsity of the multiview set as well as the correlation between images. However, such joint reconstruction is problematic in that the computation burdens imposed are likely to be significant, particularly so as the number of views increases.

As an alternative multiview reconstruction, [119, 120, 121] proposed the extension of the motion-compensated BCS-SPL (MC-BCS-SPL) [90] framework discussed previously to the multiview scenario. Specifically, [119, 120, 121] capitalized on disparity estimation (DE) and disparity compensation (DC) between adjacent views to provide a prediction of the current image to be reconstructed. The DE/DC prediction drives a residual-based CS reconstruction of the current view in a procedure quite similar to MC-BCS-SPL with DE/DC replacing motion estimation (ME) and motion compensation (MC).

Here, we overview this disparity-compensated variant of BCS-SPL which was called DC-BCS-SPL in [119, 120, 121]. We also extend this DC-BCS-SPL process to the case of multiview video, wherein DE/DC is coupled with ME/MC such that predictions for the current view are created both from adjacent views as well as from temporally neighboring frames. Experimental results show that the incorporation of DE/DC and, in the case of multiview video, ME/MC, into the CS-reconstruction process provides a significant increase in reconstruction quality as compared to the straightforward CS reconstruction of each individual view independently of the others.

6.1 Single-View Reconstruction

In order to provide a CS reconstruction of a single view within a set of multiview images, [119] couples BCS-SPL still-image reconstruction with a DE/DC-driven prediction process, calling the resulting algorithm DC-BCS-SPL. The DC-BCS-SPL algorithm is partitioned into two phases. In the first phase, a predictor $\mathbf{x}_p$ for current view $\mathbf{x}_d$ is created by bidirectionally interpolating the closest adjacent views,

$$\mathbf{x}_p = \mathrm{ImageInterpolation}(\mathbf{x}_{d-1}, \mathbf{x}_{d+1}), \tag{6.1}$$

where $\mathbf{x}_{d-1}$ and $\mathbf{x}_{d+1}$ are the “left”and “right” neighbors of $\mathbf{x}_d$, respectively. Next, the residual $\mathbf{r}$ is calculated between the original observation $\mathbf{y}_d$ and the observation resulting from the projection of $\mathbf{x}_p$ using the same measurement matrix, $\mathbf{\Phi}$; i.e.,

$$\mathbf{r} = \mathbf{y}_d - \mathbf{\Phi}\mathbf{x}_p. \tag{6.2}$$

This residual then drives the BCS-SPL reconstruction. We note that, alternatively, $\mathbf{x}_p$ could be produced by direct BCS-SPL reconstruction of the current image, i.e., $\mathbf{x}_p = \mathrm{CSReconstruction}(\mathbf{y}_d, \mathbf{\Phi})$. However, we have found that, usually, the quality of the interpolated image is much better than that of the direct BCS-SPL reconstruction.

In the second phase, the reconstructed residual $\hat{\mathbf{r}}$ produces the reconstruction $\hat{\mathbf{x}}_d = \hat{\mathbf{r}} + \mathbf{x}_p$. The prediction process then repeats, only this time, DE/DC is used instead of interpolation. Specifically, $\mathbf{DV}_{d-1}$ and $\mathbf{DV}_{d+1}$ are the fields of left and right disparity vectors, respectively; these are obtained from DE applied to the current reconstruction, $\hat{\mathbf{x}}_d$, of the current image and the left and right adjacent images. The disparity vectors then drive the DC of the current image to produce the current prediction, $\mathbf{x}_p$, and its corresponding residual, $\mathbf{r}$. This second phase of the algorithm is repeated k times. The complete algorithm is illustrated in Figure 6.1.

We note that there exist a variety of DE/DC methods of varying sophistication, with some producing high-quality predictions driven by depth or parallax information between views — any of these DE/DC strategies could be used in DC-BCS-SPL by simply placing them in the DE and DC blocks in Figure 6.1. However, we have opted instead to employ a simple block-based DE/DC process which is, in fact, quite similar to the traditional full-search block-based ME/MC process used in MC-BCS-SPL for the CS reconstruction of video. We have found that the more sophisticated depth- or parallax-driven DE/DC methods are

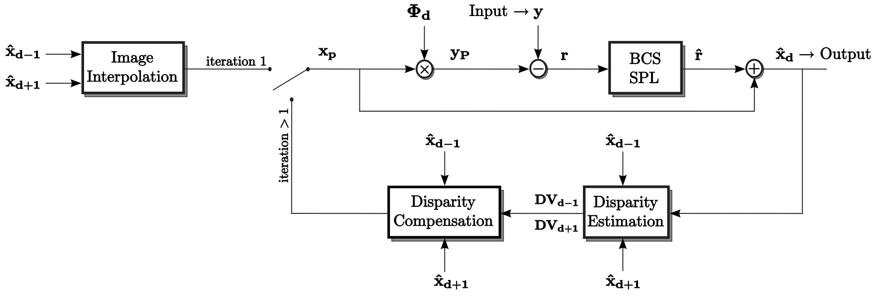


Fig. 6.1 The DC-BCS-SPL single-view reconstruction procedure (from [120]).

computationally cumbersome when compared to block-based DE/DC, which, on the other hand, provides adequate predictions for a much lower computational cost.

6.2 Multistage Reconstruction of Multiview Images

Above, we described the CS recovery of a single frame within a multiview dataset, given a disparity-compensated prediction made from adjacent views. It was implicitly assumed that these left and right views themselves had already been reconstructed via some process. However, we can see that, assuming that these left and right views are also recovered from CS measurements, the performance of the recovery of any given view is dependent upon the quality of the views used as references. The higher the distortion present in the reference frames, the higher the distortion will be in the recovery of the given view of current interest.

To reconstruct the entire multiview dataset from individual CS measurements of each of the constituent frames, [121] proposes the multistage process illustrated in Figure 6.2. We see that the reconstruction of the entire multiview set is partitioned into three stages. In the first, or initial, stage, each image in the multiview set is reconstructed individually from the received set of measurements using BCS-SPL. In the second stage (the "basic" stage), each image is reconstructed using the DC-BCS-SPL procedure of Figure 6.1 with the left and right reference views as obtained from the preceding initial stage.

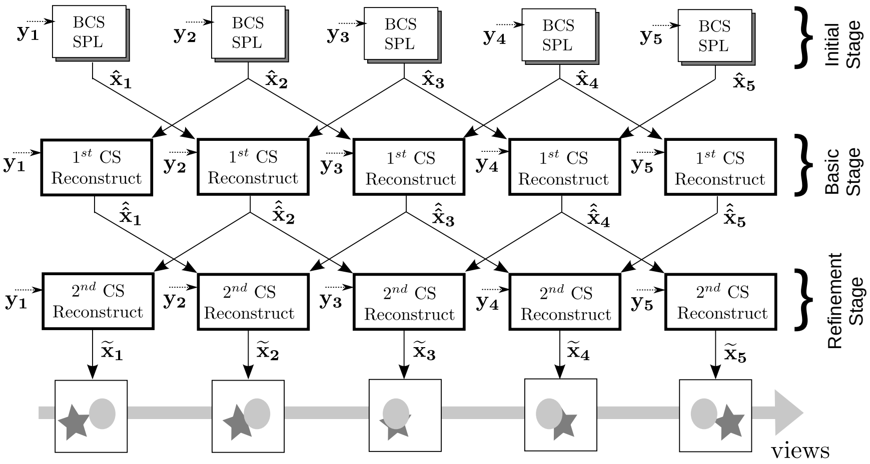


Fig. 6.2 The multistage DC-BCS-SPL multiview reconstruction procedure (from [121]).

Subsequently, one or more refinement stages are performed. A refinement stage of the algorithm is simply the repetition of the basic stage as described above with the results from the second stage substituted for the references used to drive the DE/DC-based reconstruction. The stages could conceivably be repeated until there is no significant difference between consecutive passes; however, in [121], only one refinement stage in order to minimize the overall computational complexity of the reconstruction.

6.3 Reconstruction of Multiview Video

The multistage DC-BCS-SPL procedure described above reconstructs an entire set of multiview images; however, the algorithm can be easily extended for use with multiview video in which we have multiple time instances of each view. To do so, we perform predictions not only along the disparity (or view) axis, but also along the temporal axis via ME/MC, as illustrated in Figure. 6.3. The algorithm is partitioned into three phases, much like DC-BCS-SPL for multiview images. In the initial stage, each frame in each view in the multiview video is reconstructed individually from the received set of measurements

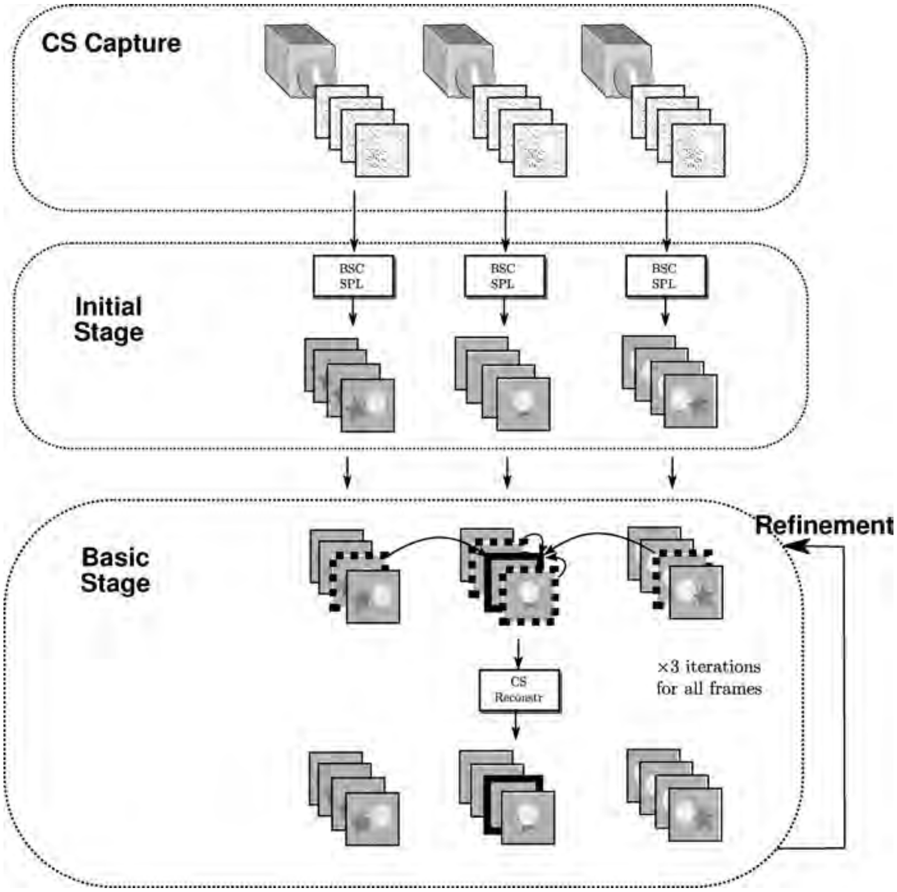


Fig. 6.3 The multistage DC-BCS-SPL reconstruction using ME/MC and DE/DC for multiview video.

using BCS-SPL. In the second stage, for each image $\mathbf{x}_d^t$ at time t and view d, a prediction is created by directionally interpolating the BCS-SPL reconstructions of the closest frames in both the temporal and disparity directions; we use the procedure in [59] for this directional interpolation. These four neighboring views/frames—$\mathbf{x}_{d-1}^t$, $\mathbf{x}_d^{t-1}$, $\mathbf{x}_{d+1}^t$, and $\mathbf{x}_d^{t+1}$—are first spatially lowpass filtered, then a classical forward block-matching DE or ME is performed between them, which is further refined in order to obtain a bidirectional view/temporal interpolation.

The initial predictor used for the image compensation is obtained by averaging the interpolations on temporal and disparity axes; i.e.,

$$\begin{aligned}\mathbf{x}_p = 0.5 \cdot \mathrm{TemporalInterpolation}(\hat{\mathbf{x}}_d^{t-1}, \hat{\mathbf{x}}_d^{t+1}) \\ + 0.5 \cdot \mathrm{DisparityInterpolation}(\hat{\mathbf{x}}_{d-1}^{t}, \hat{\mathbf{x}}_{d+1}^{t}).\end{aligned} \tag{6.3}$$

Next, we compute the residual between the measurements and the $\mathbf{\Phi}_d^t$ projection of the predicted frame. This residual in the measurement domain is then reconstructed using BCS-SPL and added back to the prediction to generate a reconstruction, $\hat{\mathbf{x}}_d^t$. $\hat{\mathbf{x}}_d^t$ is further refined in the basic stage by calculating fields of disparity vectors, $(\mathbf{DV}_{d-1}^t, \mathbf{DV}_{d+1}^t)$, and temporal motion vectors, $(\mathbf{MV}_d^{t-1}, \mathbf{MV}_d^{t+1})$. These vectors then drive the compensation to form both view and temporal predictions of the current frame from the neighboring frames. The final prediction is obtained by averaging these four predictions, and the procedure is repeated.

Subsequently, one or more refinement stages are performed. A refinement stage of the algorithm is simply the repetition of the basic stage as described above with the results from the second stage substituted for the references used to drive the compensated-CS reconstruction. The stages could conceivably be repeated until there is no significant difference between consecutive passes; however, we use only one refinement stage.

6.4 Experimental Observations

6.4.1 Multiview Images

In order to observe the effectiveness of DC-BCS-SPL recovery, we evaluate its performance at each stage of reconstruction shown in Figure 6.2 — i.e., at the initial stage, at the basic stage, and at the refinement stage. In our experiments, we use a dual-tree discrete wavelet transform (DDWT) [78] with six levels of decomposition as the sparse representation basis, $\mathbf{\Psi}$; we note that the performance of the DDWT within the BCS-SPL framework was found to be among the best of the transforms investigated in [89]. For BCS measurement, a block size of 64×64 pixels is used with an orthonormalized dense Gaussian matrix, and DE for each view is calculated using a block size of 16×16 pixels with a search window of 32 × 32 pixels. All results are averaged over 5 independent trials. For multiview image data, we use the multiview images from the Middlebury stereo-image database,[1] an example of which is depicted in Figure 6.4.

In this experimental framework, we evaluate the reconstruction performance in terms of a peak signal-to-noise ratio (PSNR) obtained for a variety of subrates S, with each view of the multiview dataset having

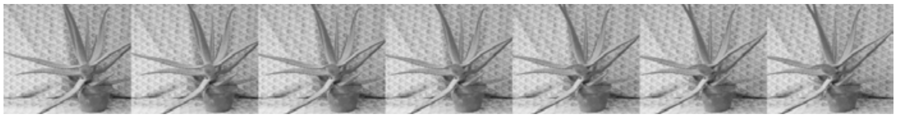


Fig. 6.4 The seven-image multiview image set "Aloe."

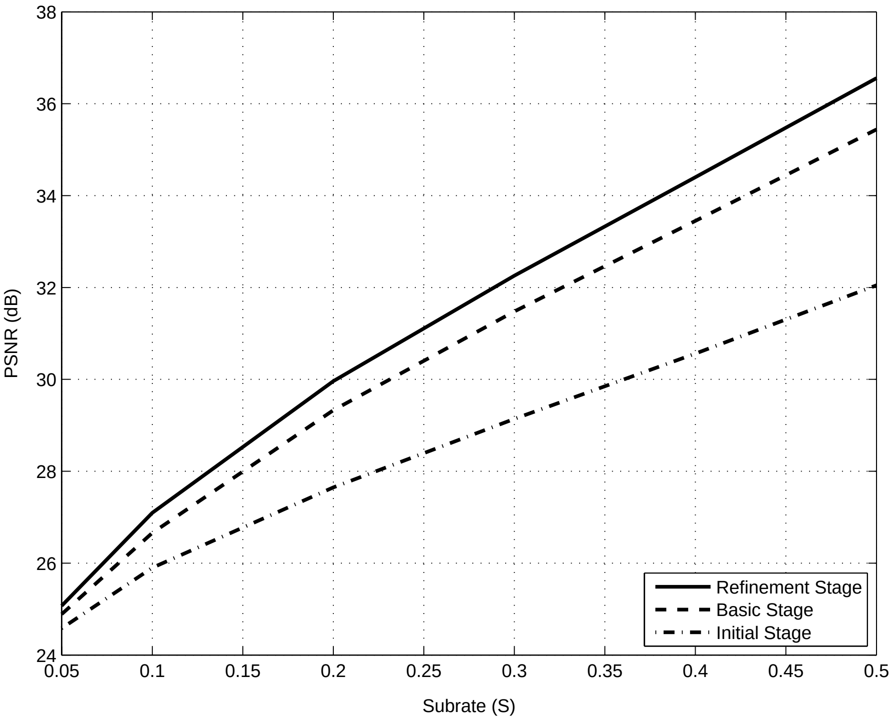


Fig. 6.5 Average recovery distortion performance over subrate for the "Aloe" multiview-image dataset.

---

[1] `http://cat.middlebury.edu/stereo/data.html`.

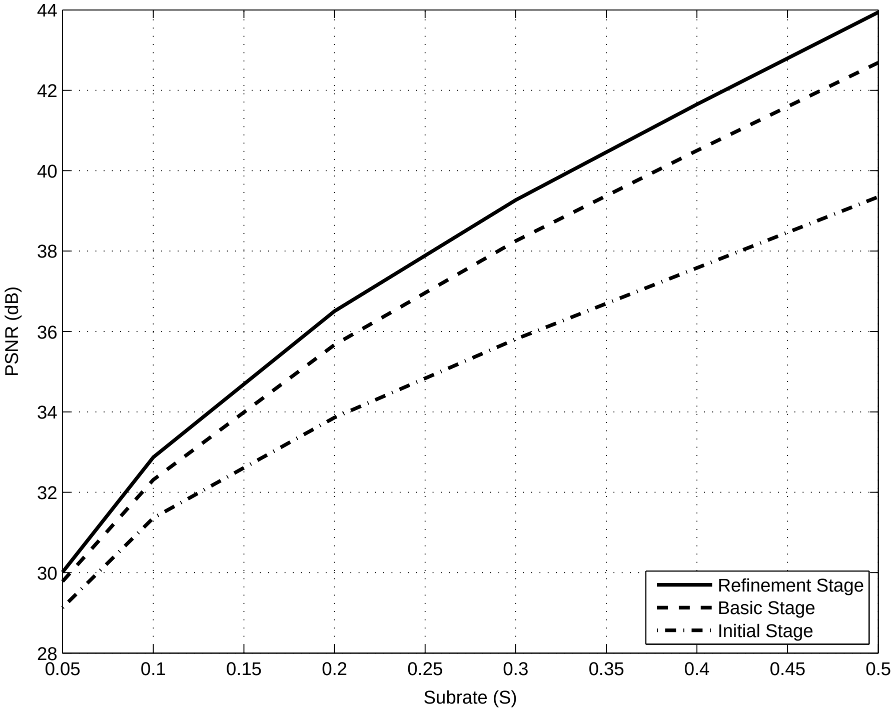


Fig. 6.6 Average recovery distortion performance over subrate for the "Baby" multiview-image dataset.

the same subrate; we average the PSNR obtained across all the views and report an average PSNR over the entire multiview dataset. The results are presented in Figures 6.5 and 6.6. We see that the incorporation of DE/DC into the reconstruction process as it occurs in the basic stage of Figure 6.2 provides a significant increase in reconstruction quality as opposed to the independent reconstruction of each view (the initial stage). Furthermore, the refinement stage further improves the reconstruction quality, especially as the subrate increases.

6.4.2 Multiview Video

For multiview video, each frame within each view of a multiview video sequence has the same subrate. ME uses the same process as DE — namely, full-search block ME with a block size of 16 × 16 and a search

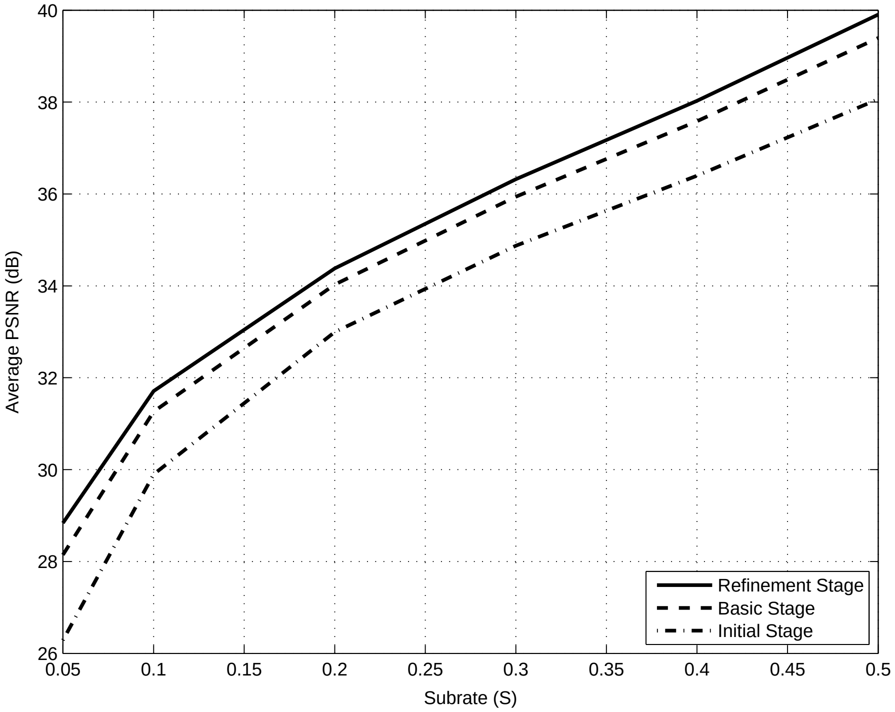


Fig. 6.7 Average recovery distortion performance over subrate for the "Ballet" multiview video.

window of 32 × 32. Figures 6.7 and 6.8 present the performance at each of the three stages of reconstruction for several 256 × 192 multiview sequences over various subrates. As before, DE/DC coupled with ME/MC in the DC-BCS-SPL reconstruction occurring in the basic stage improves reconstruction quality dramatically over the independent reconstruction in the initial stage, and the refinement stage produces even further quality improvement.

6.5 Perspectives

In this discussion, we have considered the CS recovery of multiview images as well as multiview video. Central to this reconstruction process was the creation of predictions of the current view from adjacent views via DE and DC, and, in the case of multiview video, predictions

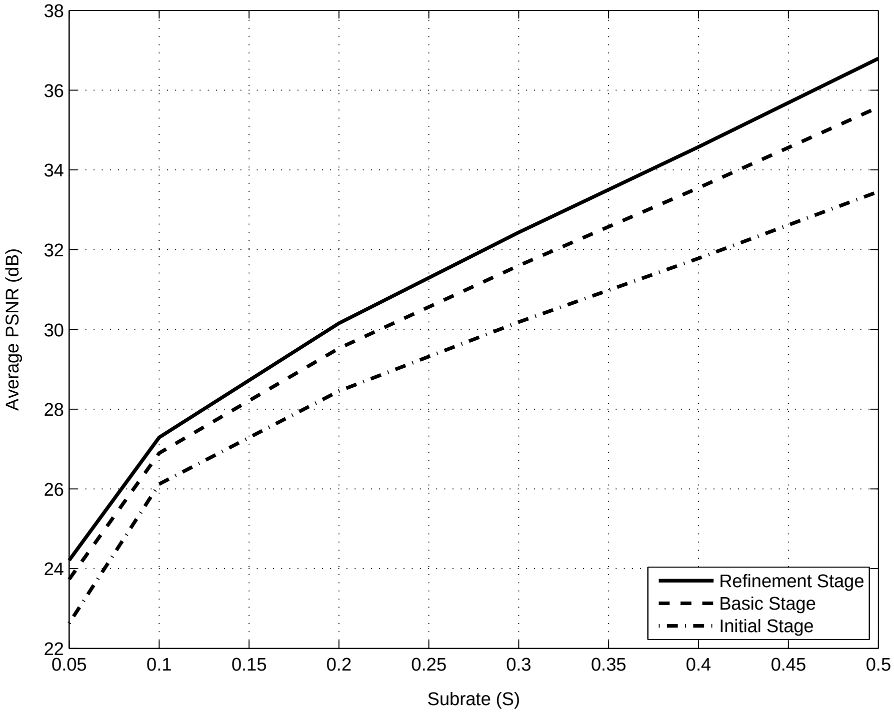


Fig. 6.8 Average recovery distortion performance over subrate for the "Book Arrival" multiview video.

from temporally neighboring frames via ME and MC. These DE/DC- and ME/MC-based predictions were used in a CS reconstruction of a residual rather than the frame directly. Experimental results displayed an increase in performance when using DE/DC and ME/MC predictions in comparison to recoveries which merely reconstruct each image independently from one another.

7

Conclusions

In this survey, we have overviewed the emerging concept of compressed sensing (CS) with a particular focus on recent proposals for its use with a variety of imaging media, including still images, motion video, as well as multiview images and video. Throughout, we have considered a variety of CS reconstruction techniques proposed in recent literature and have examined relative performance of several prominent reconstruction algorithms for each of the various imagery formats. We have given particular emphasis to block-based measurement and reconstruction which has the advantages of significantly reduced memory and computation with respect to other approaches relying on full-frame CS measurement operators.

For multiple-image scenarios, including video and multiview imagery, we have capitalized on established strategies in traditional source coding for motion estimation and compensation as well as disparity estimation and compensation. These predictive techniques were used to exploit frame-to-frame redundancies due to object motion and parallax to provide residual frames that are more compressible — in a sense of a rapid decay of coefficient magnitude within a sparsity transform — than their corresponding original frames. Reconstruction from such prediction residuals was seen to significantly improve performance as compared to straightforward reconstruction of each frame independently.

Ultimately, we argue that CS sensors that simultaneously reduce dimension during signal acquisition may eventually provide for image sampling devices at greatly reduced cost. We anticipate that certain applications — such as infrared imaging, in which single sensing elements are extremely expensive, or remote signal-sensing platforms that are severely resource-constrained in terms of power, storage, and communication bandwidth — stand to benefit greatly from CS. Effective CS reconstruction algorithms, such as those that we survey here, will thus be critical in enabling widespread adoption of such cheap and efficient CS sensors in a variety of imaging applications.

Acknowledgments

The authors thank M. Trocan, B. Pesquet-Popescu, and T. Maugey for numerous contributions to the multiview-image and video work described here. The research work of the authors reported in this survey was supported in part by the National Science Foundation under Grant No. CCF-0915307.

References

[1] D. Achlioptas, “Database-friendly random projections: Johnson-Lindenstrauss with binary coins,” *Journal of Computer and System Science*, vol. 66, no. 4, pp. 671–687, June 2003.

[2] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, “Fast image recovery using variable splitting and constrained optimization,” *IEEE Transactions on Information Theory*, vol. 19, no. 9, pp. 2345–2356, September 2010.

[3] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo, “An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems,” *IEEE Transactions on Information Theory*, vol. 20, no. 3, pp. 681–695, March 2011.

[4] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transform,” *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205–220, April 1992.

[5] S. D. Babacan, R. Molina, and A. K. Katsaggelos, “Bayesian compressive sensing using Laplace priors,” *IEEE Transactions on Image Processing*, vol. 19, no. 1, pp. 53–63, January 2010.

[6] R. Baraniuk, M. Davenport, R. DeVore, and M. Wakin, “A simple proof of the restricted isometry property for random matrices,” *Constructive Approximation*, vol. 28, no. 3, pp. 253–263, December 2008.

[7] R. G. Baraniuk, “Compressive sensing,” *IEEE Signal Processing Magazine*, vol. 24, no. 4, pp. 118–121, July 2007.

[8] R. G. Baraniuk, E. Candès, R. Nowak, and M. Vetterli, “Compressive sampling,” *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 12–13, March 2008.

[9] R. G. Baraniuk, V. Cevher, M. F. Duarte, and C. Hegde, "Model-based compressive sensing," *IEEE Transactions on Information Theory*, vol. 56, no. 4, pp. 1982–2001, April 2010.

[10] R. G. Baraniuk, V. Cevher, and M. B. Wakin, "Low-dimensional models for dimensionality reduction and signal recovery: A geometric perspective," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 959–971, June 2010.

[11] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[12] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging.* Bristol, UK: Institute of Physics Publishing, 1998.

[13] J. M. Bioucas-Dias and M. A. T. Figueiredo, "A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration," *IEEE Transactions on Image Processing*, vol. 16, no. 12, pp. 2992–3004, December 2007.

[14] T. Blumensath and M. E. Davies, "Iterative thresholding for sparse approximations," *The Journal of Fourier Analysis and Applications*, vol. 14, no. 5, pp. 629–654, December 2008.

[15] T. Blumensath and M. E. Davies, "Iterative hard thresholding for compressed sensing," *Applied and Computational Harmonic Analysis*, vol. 27, no. 3, pp. 265–274, November 2009.

[16] J. B. Boettcher and J. E. Fowler, "Video coding using a complex wavelet transform and set partitioning," *IEEE Signal Processing Letters*, vol. 14, no. 9, pp. 633–636, September 2007.

[17] 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.

[18] E. Candès and T. Tao, "Near-optimal signal recovery from random projections: Universal encoding strategies?," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, December 2006.

[19] E. Candès, M. B. Wakin, and S. P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, no. 5–6, pp. 877–905, December 2008.

[20] E. J. Candès, "Compressive sampling," in *Proceedings of the International Congress of Mathematicians*, vol. 3, pp. 1433–1452, Madrid, Spain, August 2006.

[21] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, December 2005.

[22] 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.

[23] V. Chappelier, C. Guillemot, and S. Marinkovi̇c, "Image coding with iterated contourlet and wavelet transforms," in *Proceedings of the International Conference on Image Processing*, vol. 5, pp. 3157–3160, Singapore, October 2004.

[24] P. Chen and J. W. Woods, "Bidirectional MC-EZBC with lifting implementation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 10, pp. 1183–1194, October 2004.

[25] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 33–61, August 1998.

[26] X. Chen and P. Frossard, "Joint reconstruction of compressed multi-view images," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 1005–1008, Taipei, Taiwan, April 2009.

[27] S.-J. Choi and J. W. Woods, "Motion-compensated 3-D subband coding of video," *IEEE Transactions on Image Processing*, vol. 8, no. 2, pp. 155–167, February 1999.

[28] W. Dai and O. Milenkovic, "Information theoretical and algorithmic approaches to quantized compressive sensing," *IEEE Transactions on Communications*, vol. 59, no. 7, pp. 1857–1866, July 2011.

[29] S. Dasgupta and A. Gupta, "An elementary proof of a theorem of Johnson and Lindenstrauss," *Random Structures and Algorithms*, vol. 22, no. 1, pp. 60–65, January 2003.

[30] I. Daubechies, R. De Vore, M. Fournasier, and C. S. Güntürk, "Iteratively reweighted least squares minimization for sparse recovery," *Communications on Pure and Applied Mathematics*, vol. 63, no. 1, pp. 1–38, January 2010.

[31] I. Daubechies, M. Defrise, and C. De Mol, "An interative thresholding algorithm for linear inverse problems with a sparsity constraint," *Communications on Pure and Applied Mathematics*, vol. 57, no. 11, pp. 1413–1457, November 2004.

[32] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *The Journal of Fourier Analysis and Applications*, vol. 4, no. 3, pp. 245–267, 1998.

[33] M. N. Do and M. Vetterli, "The contourlet transform: An efficient directional multiresolution image representation," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2091–2106, December 2005.

[34] 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*, pp. 1393–1396, Cairo, Egypt, November 2009.

[35] T. T. Do, L. Gan, N. Nguyen, and T. D. Tran, "Sparsity adaptive matching pursuit algorithm for practical compressed sensing," in *Proceedings of the 42th Asilomar Conference on Signals, Systems, and Computers*, pp. 581–587, Pacific Grove, California, October 2008.

[36] T. T. Do, T. D. Tran, and L. Gan, "Fast compressive sampling with structurally random matrices," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 3369–3372, Las Vegas, NV, March 2008.

[37] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, May 1995.

[38] D. L. Donoho, "Compressed sensing," *IEEE Transactions on Information Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.

[39] D. L. Donoho, A. Maleki, and A. Montanari, "Message-passing algorithms for compressed sensing," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 106, no. 45, pp. 18914–18919, November 2009.

[40] D. L. Donoho, Y. Tsaig, I. Drori, and J.-L. Starck, “Sparse solution of underdetermined linear equations by stagewise orthogonal matching pursuit,” Techinical Report, Stanford University, 2006.

[41] M. F. Duarte and R. G. Baraniuk, “Kronecker compressive sensing,” *IEEE Transactions on Image Processing*, to appear.

[42] 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.

[43] M. F. Duarte and Y. C. Eldar, “Structured compressed sensing: From theory to applications,” *IEEE Transactions on Signal Processing*, vol. 59, no. 9, pp. 4053–4085, September 2011.

[44] R. Eslami and H. Radha, “On low bit-rate coding using the contourlet transform,” in *Proceedings of the 37th Asilomar Conference on Signals, Systems, and Computers*, vol. 2, pp. 1524–1528, Pacific Grove, CA, November 2003.

[45] R. Eslami and H. Radha, “Wavelet-based contourlet transform and its application to image coding,” in *Proceedings of the International Conference on Image Processing*, vol. 5, pp. 3189–3192, Singapore, October 2004.

[46] E. Feig and S. Winograd, “Fast algorithms for the discrete cosine transform,” *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2174–2193, September 1992.

[47] M. A. T. Figueiredo, R. D. Nowak, and S. J. Wright, “Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems,” *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 4, pp. 586–597, December 2007.

[48] A. K. Fletcher, S. Rangan, and V. K. Goyal, “On the rate-distortion performance of compressed sensing,” in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 885–888, Honolulu, HI, April 2007.

[49] J. E. Fowler, J. B. Boettcher, and B. Pesquet-Popescu, “Image coding using a complex dual-tree wavelet transform,” in *Proceedings of the European Signal Processing Conference*, Poznań, Poland, September 2007.

[50] J. E. Fowler, S. Mun, and E. W. Tramel, “Multiscale block compressed sensing with smoother projected Landweber reconstruction,” in *Proceedings of the European Signal Processing Conference*, pp. 564–568, Barcelona, Spain, August 2011.

[51] J. E. Fowler and B. Pesquet-Popescu, “Wavelets in source coding, communications, and networks: An overview,” *EURASIP Journal on Image and Video Processing*, vol. 2007, p. 27, Article ID 60539 2007.

[52] U. Gamper, P. Boesiger, and S. Kozerke, “Compressed sensing in dynamic MRI,” *Magnetic Resonance in Medicine*, vol. 59, no. 2, pp. 365–373, February 2008.

[53] L. Gan, “Block compressed sensing of natural images,” in *Proceedings of the International Conference on Digital Signal Processing*, pp. 403–406, Cardiff, UK, July 2007.

[54] L. Gan, T. T. Do, and T. D. Tran, "Fast compressive imaging using scrambled block Hadamard ensemble," in *Proceedings of the European Signal Processing Conference*, Lausanne, Switzerland, August 2008.

[55] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Norwell, MA: Kluwer Academic Publishers, 1992.

[56] B. Girod, "Motion-compensating prediction with fractional-pel accuracy," *IEEE Transactions on Communications*, vol. 41, no. 4, pp. 604–612, April 1993.

[57] B. Girod, "Efficiency analysis of multihypothesis motion-compensated prediction for video coding," *IEEE Transactions on Image Processing*, vol. 9, no. 2, pp. 173–183, February 2000.

[58] 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, March 2008.

[59] C. Guillemot, F. Pereira, L. Torres, T. Ebrahimi, R. Leonardi, and J. Ostermann, "Distributed monoview and multiview video coding," *IEEE Signal Processing Magazine*, vol. 24, no. 5, pp. 67–76, September 2007.

[60] L. Haixiao, S. Bin, Q. Hao, and Q. Zhiliang, "A dictionary generation scheme for block-based compressed video sensing," in *Proceedings of the IEEE International Conference on Signal Processing, Communications and Computing*, pp. 1–5, Xi'an, China, September 2011.

[61] B. Han, F. Wu, and D. Wu, "Image representation by compressive sensing for visual sensor networks," *Journal of Visual Communication and Image Representation*, vol. 21, no. 4, pp. 325–333, May 2010.

[62] J. Haupt and R. Nowak, "Signal reconstruction from noisy random projections," *IEEE Transactions on Information Theory*, vol. 52, no. 9, pp. 4036–4048, September 2006.

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

[64] L. He, H. Chen, and L. Carin, "Tree-structured compressive sensing with variational Bayesian analysis," *IEEE Signal Processing Letters*, vol. 17, no. 3, pp. 233–236, March 2010.

[65] *Advanced Video Coding for Generic Audiovisual Services* ITU-T, May. ITU-T Recommendation H.264, 2003.

[66] *Digital Compression and Coding of Continuous-tone Still Image—Part 1: Requirements and guidelines* ISO/IEC 10918-1, JPEG Coding Standard, 1991.

[67] *Information Technology—Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbits/s* ISO/IEC 11172-2, MPEG-1 Video Coding Standard, 1993.

[68] *Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video* ISO/IEC 13818-2, MPEG-2 Video Coding Standard, 1995.

[69] *Information Technology—JPEG 2000 Image Coding System—Part 1: Core Coding System* ISO/IEC 15444-1, 2000.

[70] L. Jacques, D. K. Hammod, and J. M. Fadili, "Dequantizing compressed sensing: When oversampling and non-Gaussian constraints combine," *IEEE Transactions on Information Theory*, vol. 57, pp. 559–571, January 2011.

[71] L. Jacques, P. Vandergheynst, A. Bibet, V. Majidzadeh, A. Schmid, and Y. Leblebici, "CMOS compressed imaging by random convolution," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 1113–1116, Taipei, Taiwan, April 2009.

[72] S. Ji, Y. Xue, and L. Carin, "Bayesian compressive sensing," *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, June 2008.

[73] W. B. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," *Contemporary Mathematics*, vol. 26, pp. 189–206, 1984.

[74] H. Jung, K. Sung, K. S. Nayak, E. Y. Kim, and J. C. Ye, "$k{-}t$ FOCUSS: A general compressed sensing framework for high resolution dynamic MRI," *Magnetic Resonance in Medicine*, vol. 61, no. 1, pp. 103–116, January 2009.

[75] H. Jung and J. C. Ye, "Motion estimated and compensated compressed sensing dynamic magnetic resonance imaging: What we can learn from video compression techniques," *Imaging Systems and Technology*, vol. 20, no. 2, pp. 81–98, June 2010.

[76] L.-W. Kang and C.-S. Lu, "Distributed compressive video sensing," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 1169–1172, Taipei, Taiwan, April 2009.

[77] Y. Kim, M. S. Nadar, and A. Bilgin, "Compressed sensing using a Gaussian scale mixtures model in wavelet domain," in *Proceedings of the International Conference on Image Processing*, pp. 3365–3368, Hong Kong, September 2010.

[78] N. G. Kingsbury, "Complex wavelets for shift invariant analysis and filtering of signals," *Journal of Applied Computational Harmonic Analysis*, vol. 10, pp. 234–253, May 2001.

[79] L. Landweber, "An iteration formula for Fredholm integral equations of the first kind," *American Journal of Mathematics*, vol. 73, no. 3, pp. 615–624, July 1951.

[80] X. Li, Z. Wei, and L. Xiao, "Compressed sensing joint reconstruction for multiview images," *Electronics Letters*, vol. 46, no. 23, pp. 1548–1550, November 2010.

[81] W. Lu and N. Vaswani, "Modified compressive sensing for real-time dynamic MR imaging," in *Proceedings of the International Conference on Image Processing*, pp. 3045–3048, Cairo, Egypt, November 2009.

[82] H. D. Lüke, "The origins of the sampling theorem," *IEEE Comunnications Magazine*, vol. 37, no. 4, pp. 106–108, April 1999.

[83] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magnetic Resonance in Medicine*, vol. 58, no. 6, pp. 1182–1195, December 2007.

[84] M. Lustig, D. L. Donoho, J. M. Santos, and J. M. Pauly, "Compressed sensing MRI," *IEEE Signal Processing Magazine*, vol. 25, no. 2, pp. 72–82, March 2008.

[85] J. Makhoul, "A fast cosine transform in one and two dimensions," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 28, no. 1, pp. 27–34, February 1980.

[86] A. Maleki and D. L. Donoho, "Optimally tuned iterative reconstruction algorithms for compressed sensing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 330–341, April 2010.

[87] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3397–3415, December 1993.

[88] R. Marcia, Z. Harmany, and R. Willett, "Compressive coded aperture imaging," in *Computational Imaging III*, p. 72460G, San Jose, CA: Proc. SPIE 7246, January 2009.

[89] S. Mun and J. E. Fowler, "Block compressed sensing of images using directional transforms," in *Proceedings of the International Conference on Image Processing*, pp. 3021–3024, Cairo, Egypt, November 2009.

[90] S. Mun and J. E. Fowler, "Residual reconstruction for block-based compressed sensing of video," in *Proceedings of the IEEE Data Compression Conference*, (J. A. Storer and M. W. Marcellin, eds.), pp. 183–192, Snowbird, UT, March 2011.

[91] D. Needell and J. A. Tropp, "CoSAMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, no. 3, pp. 301–321, May 2009.

[92] G. N. Nielson, R. H. Olsson, P. R. Resnick, and O. B. Spahn, "High-speed MEMS micromirror switching," in *Proceedings of the Conference on Lasers and Electro-Optics*, pp. 1–2, Baltimore, MD, May 2007.

[93] S. Nogaki and M. Ohta, "An overlapped block motion compensation for high quality motion picture coding," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 1, pp. 184–187, San Diego, CA, May 1992.

[94] J.-R. Ohm, "Three-dimensional subband coding with motion compensation," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 559–571, September 1994.

[95] M. T. Orchard and G. J. Sullivan, "Overlapped block motion compensation: An estimation-theoretic approach," *IEEE Transactions on Image Processing*, vol. 3, no. 5, pp. 693–699, September 1994.

[96] J. Y. Park and M. B. Wakin, "A multiscale framework for compressive sensing of video," in *Proceedings of the Picture Coding Symposium*, Chicago, IL, May 2009.

[97] B. Pesquet-Popescu and V. Bottreau, "Three-dimensional lifting schemes for motion compensated video compression," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1793–1796, Salt Lake City, UT, May 2001.

[98] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Transactions on Image Processing*, vol. 12, no. 11, pp. 1338–1351, November 2003.

[99] J. Prades-Nebot, Y. Ma, and T. Huang, “Distributed video coding using compressive sampling,” in *Proceedings of the Picture Coding Symposium*, Chicago, IL, May 2009.

[100] R. Puri, A. Majumdar, P. Ishwar, and K. Ramchandran, “Distributed video coding in wireless sensor networks,” *IEEE Signal Processing Magazine*, vol. 23, no. 4, pp. 94–106, July 2006.

[101] C. Qiu, W. Lu, and N. Vaswani, “Real-time dynamic MR image reconstruction using Kalman filtered compressed sensing,” in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 393–396, Taipei, Taiwan, April 2009.

[102] R. Robucci, J. D. Gray, L. K. Chiu, J. Romberg, and P. Hasler, “Compressive sensing on a CMOS separable-transform image sensor,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1089–1101, June 2010.

[103] J. Romberg, “Compressive sensing by random convolution,” *SIAM Journal on Imaging Sciences*, vol. 2, no. 4, pp. 1098–1128, December 2009.

[104] A. C. Sankaranarayanan, P. K. Turaga, R. G. Baraniuk, and R. Chellappa, “Compressive acquisition of dynamic scenes,” in *Proceedings of the European Conference on Computer Vision, Part I*, pp. 129–142, Crete, Greece, September 2010.

[105] P. Schniter, L. C. Potter, and J. Ziniel, “Fast Bayesian matching pursuit: Model uncertainty and parameter estimation for sparse linear models,” *IEEE Transactions on Signal Processing*, submitted 2008.

[106] A. Secker and D. Taubman, “Lifting-based invertible motion adaptive transform (LIMAT) framework for highly scalable video compression,” *IEEE Transactions on Image Processing*, vol. 12, no. 12, pp. 1530–1542, December 2003.

[107] L. Şendur and I. W. Selesnick, “Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency,” *IEEE Transactions on Signal Processing*, vol. 50, no. 11, pp. 2744–2756, November 2002.

[108] C. E. Shannon, “A mathematical theory of communication,” *The Bell System Technical Journal*, vol. 27, pp. 379–423, 623–656, July, October 1948.

[109] C. E. Shannon, “Communication in the presence of noise,” *Proceedings of the IRE*, vol. 37, no. 1, pp. 10–21, January 1949.

[110] C. E. Shannon, “Coding theorems for a discrete source with a fidelity criterion,” *IRE International Convention Record, Part 4*, vol. 7, pp. 142–163, 1959.

[111] J. M. Shapiro, “Embedded image coding using zerotrees of wavelet coefficients,” *IEEE Transactions on Signal Processing*, vol. 41, no. 12, pp. 3445–3462, December 1993.

[112] G. J. Sullivan, “Multi-hypothesis motion compensation for low bit-rate video coding,” in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pp. 437–440, Minneapolis, MN, April 1993.

[113] R. Sundaresan, Y. Kim, M. S. Nadar, and A. Bilgin, “Motion-compensated compressed sensing for dynamic imaging,” in *Applications of Digital Image Processing XXXIII*, (A. G. Tescher, ed.), p. 77980A, San Diego: Proc. SPIE 7798, August 2010.

[114] W. Sweldens, “The lifting scheme: A new philosophy in biorthogonal wavelet constructions,” in *Wavelet Applications in Signal and Image Processing III*, (A. F. Laine, M. A. Unser, and M. V. Wickerhauser, eds.), pp. 68–79, San Diego, CA: Proc. SPIE 2569, July 1995.

[115] D. Takhar, J. N. Laska, M. B. Wakin, M. F. Duarte, D. Baron, S. Sarvotham, K. F. Kelly, and R. G. Baraniuk, “A new compressive imaging camera architecture using optical-domain compression,” in *Computational Imaging IV*, (C. A. Bouman, E. L. Miller, and I. Pollak, eds.), p. 606509, San Jose, CA: Proc. SPIE 6065, January 2006.

[116] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *Journal of the Royal Statistical Society, Series B*, vol. 58, no. 1, pp. 267–288, 1996.

[117] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems.* Washington, D.C.: V. H. Winston & Sons, 1977.

[118] E. W. Tramel and J. E. Fowler, “Video compressed sensing with multihypothesis,” in *Proceedings of the IEEE Data Compression Conference*, (J. A. Storer and M. W. Marcellin, eds.), pp. 193–202, Snowbird, UT, March 2011.

[119] M. Trocan, T. Maugey, J. E. Fowler, and B. Pesquet-Popescu, “Disparity-compensated compressed-sensing reconstruction for multiview images,” in *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 1225–1229, Singapore, July 2010.

[120] M. Trocan, T. Maugey, E. W. Tramel, J. E. Fowler, and B. Pesquet-Popescu, “Compressed sensing of multiview images using disparity compensation,” in *Proceedings of the International Conference on Image Processing*, pp. 3345–3348, Hong Kong, September 2010.

[121] M. Trocan, T. Maugey, E. W. Tramel, J. E. Fowler, and B. Pesquet-Popescu, “Multistage compressed-sensing reconstruction of multiview images,” in *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, pp. 111–115, Saint-Malo, France, October 2010.

[122] M. Trocan, B. Pesquet-Popescu, and J. E. Fowler, “Graph-cut rate distortion algorithm for contourlet-based image compression,” in *Proceedings of the International Conference on Image Processing*, pp. 169–172, San Antonio, TX, September 2007.

[123] J. Tropp and A. Gilbert, “Signal recovery from random measurements via orthogonal matching pursuit,” *IEEE Transactions on Information Theory*, vol. 53, no. 12, pp. 4655–4666, December 2007.

[124] J. A. Tropp and S. J. Wright, “Computational methods for sparse solution of linear inverse problems,” *Proceedings of the IEEE*, vol. 98, no. 6, pp. 948–958, June 2010.

[125] Y. Tsaig and D. L. Donoho, “Extensions of compressed sensing,” *Signal Processing*, vol. 86, no. 3, pp. 549–571, March 2006.

[126] N. Vaswani, “Kalman filtered compressed sensing,” in *Proceedings of the International Conference on Image Processing*, pp. 893–896, San Diego, CA, October 2008.

[127] N. Vaswani, “LS-CS-Residual (LS-CS): Compressive sensing on least squares residual,” *IEEE Transactions on Signal Processing*, vol. 57, no. 8, pp. 4108–4120, August 2010.

[128] N. Vaswani and W. Lu, “Modified-CS: Modifiying compressive sensing for problems with partially known support,” *IEEE Transactions on Signal Processing*, vol. 58, no. 9, pp. 4595–4607, September 2010.

[129] R. G. Vaughan, N. L. Scott, and D. R. White, “The theory of bandpass sampling,” *IEEE Transactions on Signal Processing*, vol. 39, no. 9, pp. 1973–1984, September 1991.

[130] M. B. Wakin, “A manifold lifting algorithm for multi-view compressive imaging,” in *Proceedings of the Picture Coding Symposium*, Chicago, IL, May 2009.

[131] M. B. Wakin, J. N. Laska, M. F. Duarte, D. Baron, S. Sarvotham, D. Takhar, K. F. Kelly, and R. G. Baraniuk, “An architecture for compressive imaging,” in *Proceedings of the International Conference on Image Processing*, pp. 1273–1276, Atlanta, GA, October 2006.

[132] 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.

[133] L. Wang, X. Wu, and G. Shi, “Progressive quantization of compressive sensing measurements,” in *Proceedings of the IEEE Data Compression Conference*, (J. A. Storer and M. W. Marcellin, eds.), pp. 233–242, Snowbird, UT, March 2011.

[134] T. Wiegand, X. Zhang, and B. Girod, “Long-term memory motion-compensated prediction,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 1, pp. 70–84, February 1999.

[135] S. J. Wright, R. D. Nowak, and M. A. T. Figueiredo, “Sparse reconstruction by separable approximation,” *IEEE Transactions on Signal Processing*, vol. 57, no. 7, pp. 2479–2493, July 2009.

[136] X. Wu, X. Zhang, and J. Wang, “Model-guided adaptive recovery of compressive sensing,” in *Proceedings of the IEEE Data Compression Conference*, (J. A. Storer and M. W. Marcellin, eds.), pp. 123–132, Snowbird, UT, March 2009.