Multiscale Block Compressed Sensing with Smoothed Projected Landweber Reconstruction

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

Department of Electrical & Computer Engineering
Geosystems Research Institute
Mississippi State University, MS USA

August 2011

MISSISSIPPI STATE UNIVERSITY™

Motivation

Image Recovery With Compressed Sensing (CS)

- Assumption: Image is sparse
- Prior knowledge can improve reconstruction; e.g.,
  - Exploit wavelet structure in both sampling and reconstruction (statistical coefficient models)
  - Mutiscale CS—Sampling rate adjusted with DWT level (baseband retained in full)

Block-Based Compressed Sensing (BCS)

- BCS: CS sampling within small image blocks
- Advantage: very fast, low memory
- Drawback: reduced reconstruction quality (CS works better with global sampling)
- Motivation: deploy BCS within multiscale framework

CS Overview

Goal

Recover $\mathbf{x} \in \Re^N$ from

$$\mathbf{y} = \mathbf{A}\mathbf{x} \in \Re^M$$

- $\mathbf{A}$: $M \times N$ random measurement matrix, $M \ll N$
- $S = \frac{M}{N}$: subsampling rate, or subrate

Fundamental Tenet of CS

If $\mathbf{x}$ is sufficiently sparse, recovery is exact from

$$M \geq O(K \cdot \log N)$$

measurements by solving tractable program.

- K: number of nonzero coefficients in some transform Ψ, $\check{\mathbf{x}} = \Psi\mathbf{x}$

Block Compressed Sensing (BCS)

Image partitioned into small blocks ($B \times B$)

$$\mathbf{y}_j = \mathbf{\Phi}\mathbf{x}_j$$

$\mathbf{\Phi}$: $M_B \times B^2$, $\mathbf{x}_j$: block j of image

thus:

$$\mathbf{A} = \begin{bmatrix} \mathbf{\Phi} & 0 & \cdots & 0 \\ 0 & \mathbf{\Phi} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & \mathbf{\Phi} \end{bmatrix}$$

Multiscale BCS (MS-BCS)

Multiscale Sampling

Sampling operator is split into

$$\mathbf{A} = \mathbf{\Phi}'\mathbf{\Omega}$$

thus,

$$\mathbf{y} = \mathbf{\Phi}'\mathbf{\Omega}\mathbf{x}$$

- $\mathbf{\Omega}$: multiscale transform (i.e., L-level DWT)
- $\mathbf{\Phi}'$: multiscale block-based sampling operator
  - set of BCS sampling operators with subrates varying with level
  - $\mathbf{\Phi}' = \{\mathbf{\Phi}_l\}$, $1 \leq l \leq L$

Multiscale BCS (MS-BCS)

Multiscale Transform and Sampling

- Ω produces L levels of wavelet decomposition:

$$\tilde{\mathbf{x}} = \Omega \mathbf{x}$$

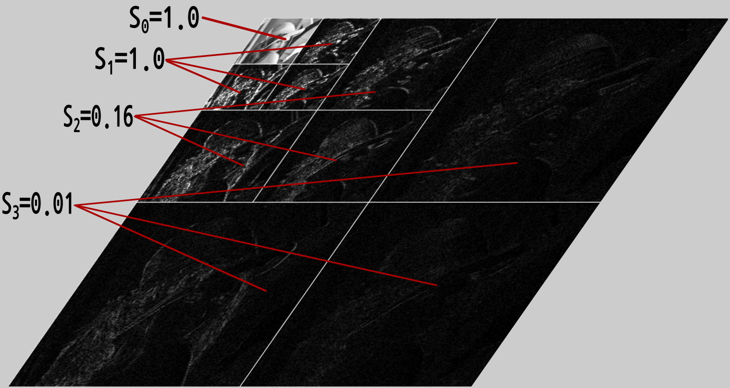
- Subband s at level l of $\tilde{\mathbf{x}}$ is divided into $B_l \times B_l$ blocks
- Each $B_l \times B_l$ block sampled with Φ_l of size $M_l \times B_l^2$:

$$\mathbf{y}_{l,s,j} = \Phi_l \tilde{\mathbf{x}}_{l,s,j}$$

  $\tilde{\mathbf{x}}_{l,s,j}$: raster-scan of block j of subband s at level l
- Φ_l yields subrate $S_l = M_l / B_l^2$ at level l

Multiscale BCS (MS-BCS)

Example: target subrate $S = 0.1$ with $L = 3$



$$S = \frac{1}{4^3} \times S_0 + \frac{3}{4^3} \times S_1 + \frac{3}{4^{3-2+1}} \times S_2 + \frac{3}{4^{3-3+1}} \times S_3 = 0.1$$

Multiscale BCS Reconstruction

MS-BCS-SPL reconstruction

- Adapt BCS-SPL algorithm (Gan 2007, Mun & Fowler 2009) to multiscale setting
- BCS-SPL:
  - Full-image smoothing with Wiener filter
  - Sparsity-enhancing thresholding in sparsity transform Ψ
  - Between smoothing and thresholding are Landweber steps:

$$\tilde{\mathbf{x}} \leftarrow \tilde{\mathbf{x}} + \mathbf{\Phi}^T \left(\mathbf{y} - \mathbf{\Phi}\tilde{\mathbf{x}}\right)$$

Multiscale BCS Reconstruction

MS-BCS-SPL algorithm

function $\tilde{\mathbf{x}}^{(i)} = \text{MS-BCS-SPL}\left(\mathbf{y}, \{\mathbf{\Phi}_{l,\ 1\le l\le L}\}, \mathbf{\Psi}, \mathbf{\Omega}, \tilde{\mathbf{x}}^{(0)}_{l,s,j}\right)$

do $\mathbf{x}^{(i)} = \mathbf{\Omega}^{-1}\tilde{\mathbf{x}}^{(i)}$

$\hat{\mathbf{x}}^{(i)} = \text{Wiener}(\mathbf{x}^{(i)})$

$\hat{\hat{\mathbf{x}}}^{(i)} = \mathbf{\Omega}\hat{\mathbf{x}}^{(i)}$

for each l, **for each** s, **for each** j

$\hat{\hat{\hat{\mathbf{x}}}}^{(i)}_{l,s,j} = \hat{\hat{\mathbf{x}}}^{(i)}_{l,s,j} + \mathbf{\Phi}_l^T(\mathbf{y}_{l,s,j} - \mathbf{\Phi}_l\hat{\hat{\mathbf{x}}}^{(i)}_{l,s,j})$

$\check{\check{\mathbf{x}}}^{(i)} = \mathbf{\Psi}\mathbf{\Omega}^{-1}\hat{\hat{\hat{\mathbf{x}}}}^{(i)}$

$\check{\mathbf{x}}^{(i)} = \text{Threshold}(\check{\check{\mathbf{x}}}^{(i)})$

$\bar{\check{\mathbf{x}}}^{(i)} = \mathbf{\Omega}\mathbf{\Psi}^{-1}\check{\mathbf{x}}^{(i)}$

for each l, **for each** s, **for each** j

$\tilde{\mathbf{x}}^{(i+1)}_{l,s,j} = \bar{\check{\mathbf{x}}}^{(i)}_{l,s,j} + \mathbf{\Phi}_l^T(\mathbf{y}_{l,s,j} - \mathbf{\Phi}_l\bar{\check{\mathbf{x}}}^{(i)}_{l,s,j})$

$D^{(i+1)} = \|\tilde{\mathbf{x}}^{(i+1)} - \hat{\hat{\hat{\mathbf{x}}}}^{(i)}\|_2$

$i = i + 1$

until $|D^{(i)} - D^{(i-1)}| < 10^{-2}$

Experimental Results

Experiment Setup

- Sparsity basis, Ψ: dual-tree DWT (DDWT)
- Sampling basis, Ω: 3-level 9/7 biorthogonal DWT
- $B_l \times B_l$ blocks sampled using structured random matrices (SRM)
- Block size $B_1 = 16$, $B_2 = 32$, $B_3 = 64$

Algorithms Compared

- Original BCS-SPL (Mun & Fowler 2009)
- TV (Candès *et al*. 2006)
- MS-GPSR (Schniter *et al*. 2008)—multiscale version of GPSR
- SALSA (Afonso *et al*. 2010) with DWT sparsity transform

Experimental Results

Reconstruction PSNR in dB

Algorithm	*Subrate* 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
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

Experimental Results

Reconstruction PSNR in dB

Algorithm	*Subrate* 0.1	0.2	0.3	0.4	0.5
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

Simulation Results for 2D Images

Lenna (512×512) with subrate $S = 0.1$

MS-BCS-SPL
31.6 dB
50 seconds

BCS-SPL
28.0 dB
30 seconds

Simulation Results for 2D Images

Lenna (512×512) with subrate $S = 0.1$

MS-GPSR
30.3 dB
20 minutes

TV
29.9 dB
1.8 hours

Conclusions

Conclusions

- MS-BCS-SPL achieves 1- to 3-dB gain over original BCS-SPL
- MS-BCS-SPL only slightly slower than original BCS-SPL
- MS-BCS-SPL rivals TV in PSNR, but significantly faster

Matlab Source Code

- MS-BCS-SPL Version 1.0

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