# Block Compressed Sensing for Images

Sungkwang Mun

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

November 2009

# Motivation

## Image Compression - File Size Comparison

Original image (Raw)
600K Byte

Compressed image (JPG)
30K Byte

- **Compression ratio :** $\frac{30K}{600K} = 0.05(5\%)$

# Motivation

## Concept of Image Compression

Original image
N: number of pixels

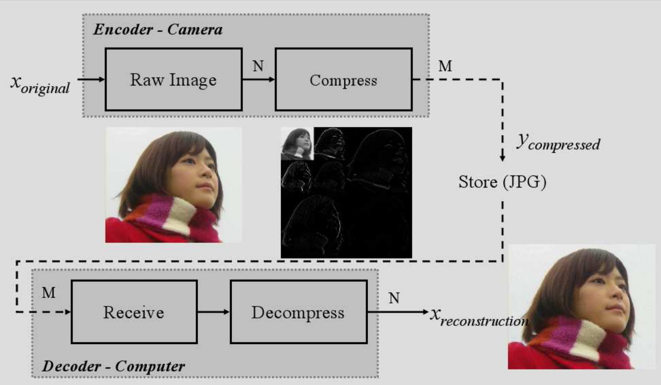
Wavelet Transform
M: number of significant coeff's

- Keep M significant coefficients only; N $>>$ M

# Motivation



## Conventional Compression




- Needs extra memory for storing raw image
- Needs fast CPU to compress

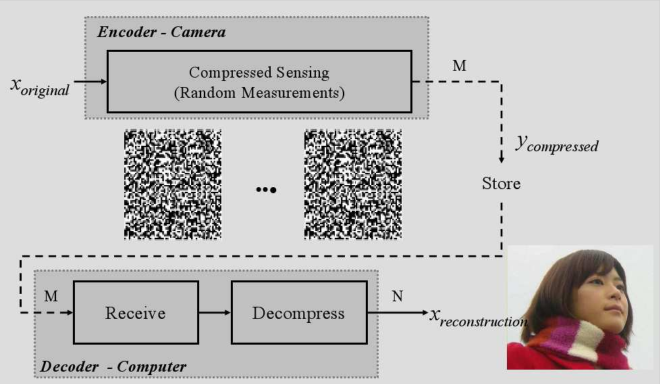
# Motivation

BCS for Images

S. Mun

- Motivation
- Background
- BCS-SPL
- Conclusions

## Compressed Sensing




- Only requires one-time matrix multiplication
- Fast acquisition without additional memory

# Outline

# Background—Compressed Sensing (CS)

## Goal

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

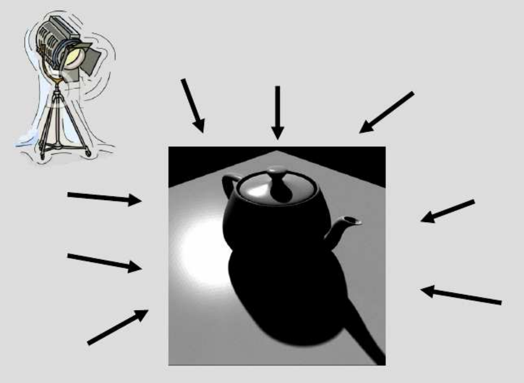
$$\mathbf{y} = \mathbf{\Phi x} \in \Re^M$$

- $\mathbf{\Phi}$: $M \times N$ random measurement matrix, $M \ll N$
- Theory:
  Recovery is exact if $\mathbf{x}$ is sufficiently sparse
- Mathematically proven by mathematicians (E.Candes and D.L. Donoho)

# Understanding Random Measurement



## Shadow example for CS



**Random Measurements:** $\Phi_{M\times N} = [\phi_1\phi_2\ldots\phi_M]^T$

# Background—Compressed Sensing (CS)

## Problems using CS in Practice

1. Huge measurement matrix
   - Thousands times larger than image (Giga bytes)
2. Complex recovery
   - Recovery (decompressing) is very complicated
   - Exponential time for ”ideal recovery”

# Solution for 1. Huge Measurement Matrix

## Block Compressed Sensing (BCS)

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

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

$\mathbf{\Phi}_B$: $\lfloor \frac{M}{N} B^2 \rfloor \times B^2$, $\mathbf{x}_j$: block $j$ of image

- Use only one small matrix on every image block
- Practical for the measurement matrix
- L. Gan, "Block compressed sensing of natural images," *Int. Conf. DSP*, 2007.

# Solution for 2. Complex Recovery

## Smooth Projected Landweber (SPL)

- **Projected Landweber**

$$\check{\check{\mathbf{x}}}^{(i)} = \check{\mathbf{x}}^{(i)} + \mathbf{\Phi}^T \left( \mathbf{y} - \mathbf{\Phi}\check{\mathbf{x}}^{(i)} \right),$$

$$\check{\mathbf{x}}^{(i+1)} = \begin{cases} \check{\check{\mathbf{x}}}^{(i)}, & \left|\check{\check{\mathbf{x}}}^{(i)}\right| \geq \tau^{(i)}, \tau : \textit{threshold value} \\ 0 & \textbf{else}. \end{cases}$$

- **Find nearly exact answer by allowing small error**
- **Greatly reduced complexity**
- **Easy to incorporate additional optimization criteria**
  - **Wiener filter (smoothing function)**
  - **Dual tree discrete wavelet transform**
  - **Bivariate shrinkage**

# Block CS - Smooth Projected Landweber



## BCS-SPL

**Finally, we propose the solution for two challenges**

1. Huge measurement matrix:
   - Block CS (BCS)
2. Complex recovery:
   - Smooth projected Landweber (SPL)

# Results for 2D Images

## Lenna for Compression Ratio $M/N = 20\%$

BCS-SPL
31.37 dB
1–5 mins

BCS-TV
30.59 dB
3–4 hrs

SAMP
28.54 dB
2–3 mins

- TV: Total-variation minimization, close to ideal recovery
- SAMP: Sparsity adaptive matching pursuit, variant of practical recovery

# Results for 2D Images

## PSNR performance comparison for Lenna image

| Algorithm | Compression Ratio ($M/N$) % | | | | |
|---|---|---|---|---|---|
| | 10 | 20 | 30 | 40 | 50 |
| BCS-SPL | 28.31 | 31.37 | 33.50 | 35.20 | 36.78 |
| BCS-TV | 27.86 | 30.60 | 32.56 | 34.25 | 35.89 |
| SAMP | 25.94 | 28.54 | 32.04 | 33.93 | 35.37 |
| GPSR | 24.69 | 28.54 | 31.53 | 33.69 | 35.82 |

# Conclusions



## Conclusions

BCS-SPL—Practical CS image reconstruction:

- Fast
- Better visual quality
- Promising video results

## References


- S. Mun and J. E. Fowler, “Block Compressed Sensing of Images Using Directional Transforms,” *ICIP 2009*, to appear.


## URL

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