Dictionary Learning & Compressive Imaging

Lawrence Carin, Guillermo Sapiro and David Brady

Electrical & Computer Engineering Department
Duke, University
Durham, NC
lcarin@ece.duke.edu

Agenda

- Dictionary learning and massive downsampling on measurement
- Compressive hyperspectral camera
- Compressive video
- Summary

Bayesian Dictionary Learning

- For the specific application considered here, each "dish" in the buffet corresponds to a dictionary element $\boldsymbol{d}_k \in \mathbb{R}^P$
- Each data $\boldsymbol{x}_i \in \mathbb{R}^P$ is represented as

$$\boldsymbol{x}_i = \mathbf{D}(\boldsymbol{z}_i \odot \boldsymbol{s}_i) + \epsilon_i$$

  where the K columns of $\mathbf{D}$ defined by $\{\boldsymbol{d}_k\}_{k=1,K}$
- Impose $\boldsymbol{z}_i \in \{0,1\}^K$ is sparse, turning on/off elements of $\boldsymbol{s}_i \in \mathbb{R}^K$, via Hadamard vector product denoted by $\odot$
- Indian buffet process used to constitute $\{\boldsymbol{d}_k\}_{k=1,K}$ and $\{\boldsymbol{z}_i\}_{i=1,N}$

Feature ("dish") Probabilities

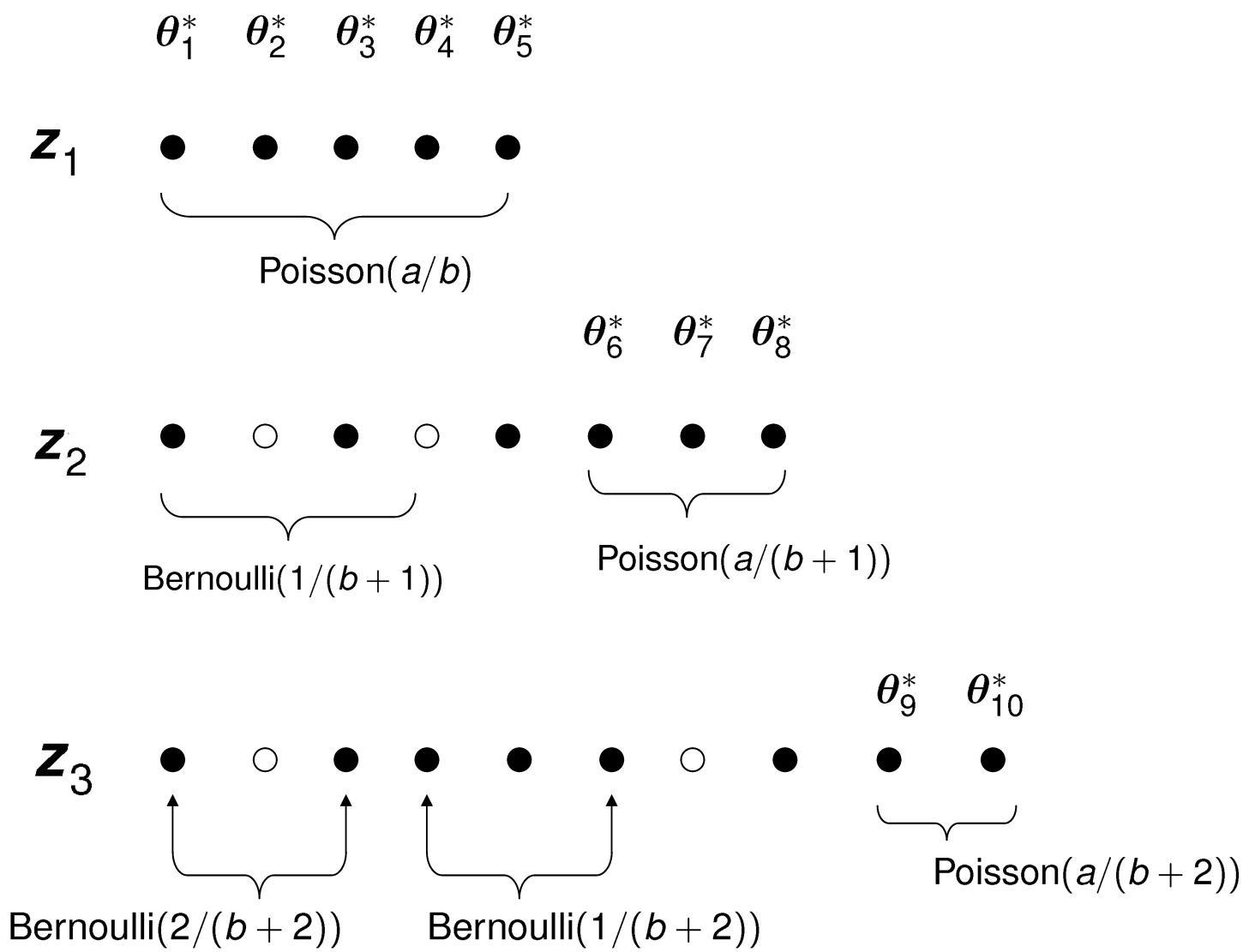
- Let π_k represent the probability of dish k, $k = 1, \ldots, K$
- We don't know the probability of each dish π_k, and wish to place a prior belief that most dishes/features are never used
- The beta distribution is a natural prior for probabilities π_k

$$\pi_k \sim \text{Beta}(a/K, b(K-1)/K) \quad , \quad k = 1, \ldots, K$$

- Each customer/data sample selects dish k as

$$z_{ik} \sim \text{Bernoulli}(\pi_k)$$

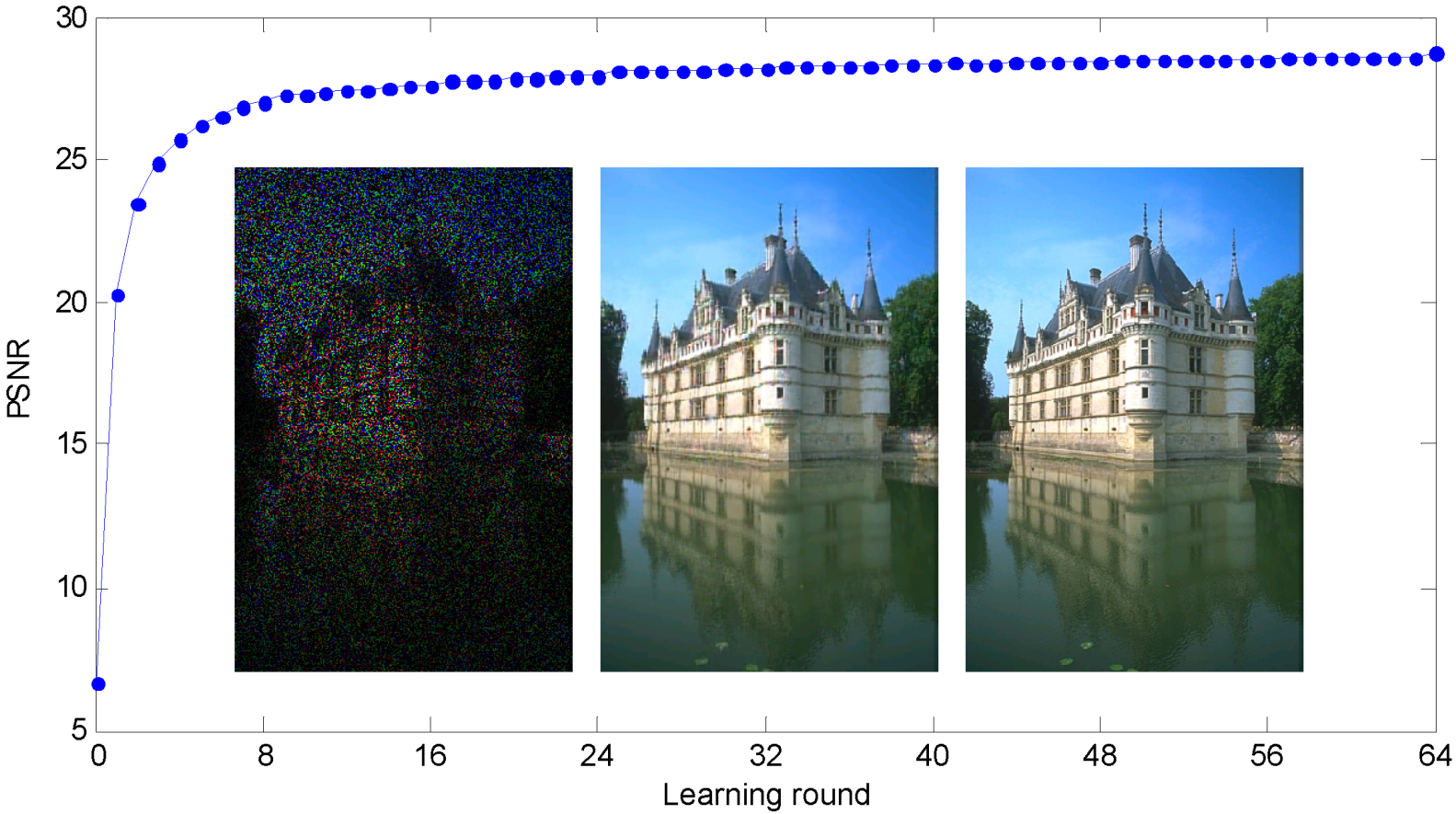
Indian Buffet Process Schematic, $K \rightarrow \infty$



IBP Model Imposes That:

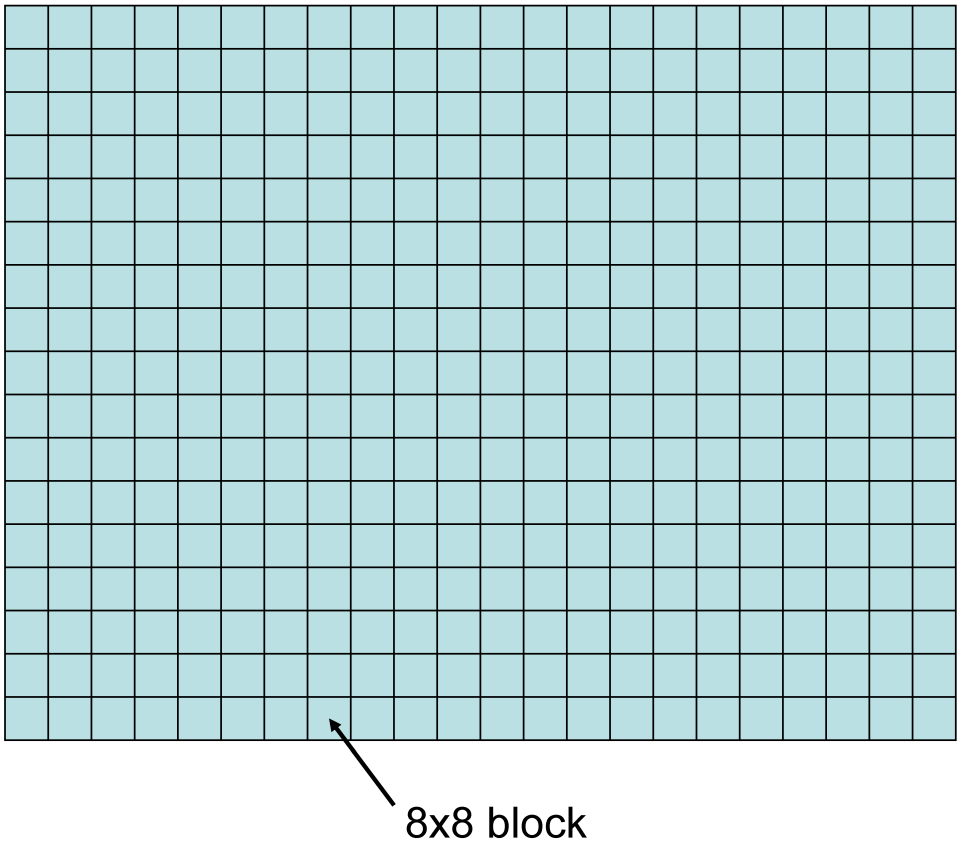
- Finite set of features/dishes needed to represent finite collection of data $\{\boldsymbol{x}_i\}_{i=1,N}$
- The number of features may expand with new data, to account for new characteristics of data
- The more "popular" features have been in the past, the more likely it is that they will be utilized in the future
- Always possible to add new features/dishes, but the probability of needing to do so decreases with increasing observed data N

IBP Application: Recovery of Missing Data



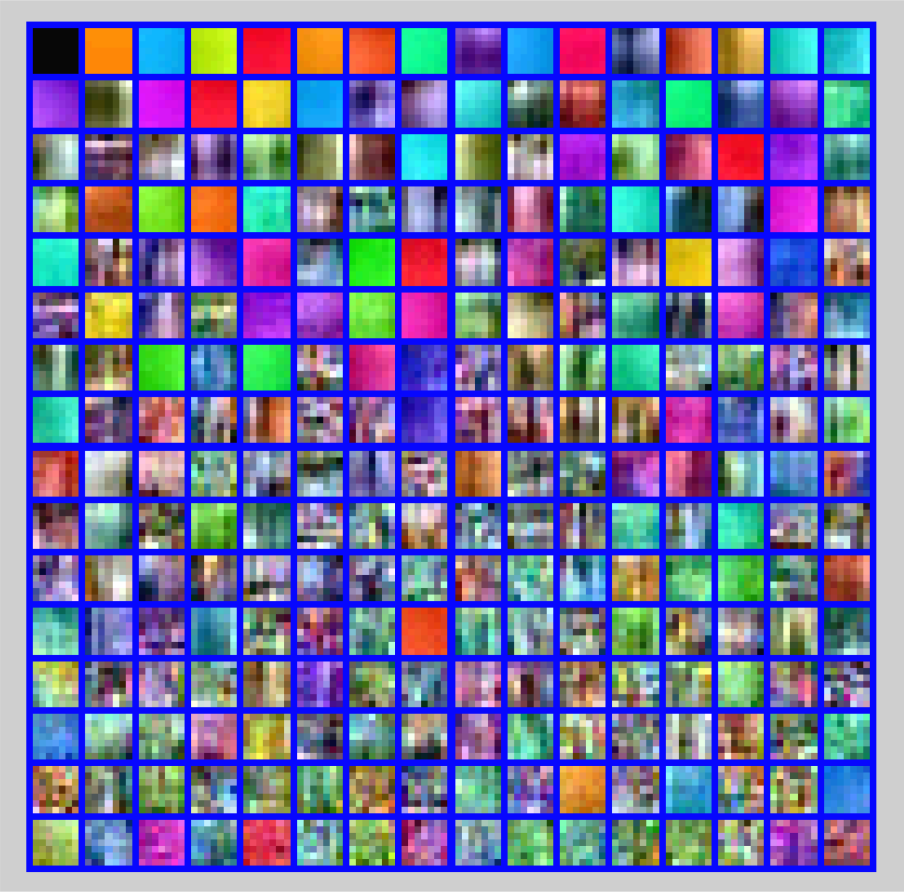
80% of RGB pixels missing at random

Different Data Samples Share Same Dictionary



$$\boldsymbol{x}_i = \sum_{k=1}^{\infty} w_{ik} z_{ik} \boldsymbol{d}_k + \epsilon_i \ , \ w_{ik} \sim \mathcal{N}(0, \alpha_0^{-1}) \ , \ z_{ik} \in \{0, 1\} \ , \ \text{finite} \ \|\boldsymbol{z}_i\|_0$$

Learned Dictionary



Inpainting

31.63 dB

Hierarchical Model

$$
\begin{aligned}
\boldsymbol{x}_i &\sim \mathcal{N}(\mathbf{D}(\boldsymbol{z}_i \odot \boldsymbol{s}_i), \gamma_e^{-1}\mathbf{I}_P) \\
\boldsymbol{s}_i &\sim \mathcal{N}_+(0, \gamma_s^{-1}\mathbf{I}_K) \\
\boldsymbol{z}_i &\sim \prod_{k=1}^{K} \text{Bernoulli}(\pi_k) \\
\boldsymbol{d}_k &\sim \mathcal{N}(0, \frac{1}{P}\mathbf{I}_P) \ , \quad k = 1, \ldots, K \\
\pi_k &\sim \text{Beta}(a/K, b(K-1)/K) \ , \quad k = 1, \ldots, K \\
\gamma_e &\sim \text{Gamma}(c, d) \\
\gamma_s &\sim \text{Gamma}(e, f)
\end{aligned}
$$

Alternative Shrinkage Hierarchical Model

$$
\begin{aligned}
\boldsymbol{x}_i &\sim \mathcal{N}(\mathbf{D}\boldsymbol{s}_i, \gamma_e^{-1}\mathbf{I}_P) \\
\boldsymbol{s}_i &\sim \frac{1}{(2\lambda)^K}\exp(-\|\boldsymbol{s}_i\|_1/\lambda) \\
\boldsymbol{d}_k &\sim \mathcal{N}(0, \frac{1}{P}\mathbf{I}_P)\ , \quad k = 1, \ldots, K \\
\gamma_e &\sim \text{Gamma}(c, d) \\
\lambda &\sim \text{Gamma}(e, f)
\end{aligned}
$$

Posterior for Shrinkage Representation

- The posterior of the model parameters may be expressed as

$$p(\mathbf{D}, \{\boldsymbol{s}_i\}, \gamma_e, \lambda | \{\boldsymbol{x}_i\}) =$$

$$\frac{\prod_{i=1}^{N} \mathcal{N}(\boldsymbol{x}_i; \mathbf{D}\boldsymbol{s}_i, \gamma_e^{-1}\mathbf{I}_P) \prod_{i=1}^{N} (2\lambda)^{-K} \exp(-\|\boldsymbol{s}_i\|_1/\lambda) \prod_{k=1}^{K} \mathcal{N}(\boldsymbol{d}_k; 0, \frac{1}{P}\mathbf{I}_P) \text{Ga}(\gamma_e; c, d) \text{Ga}(\lambda; e, f)}{p(\{\boldsymbol{x}_i\})}$$

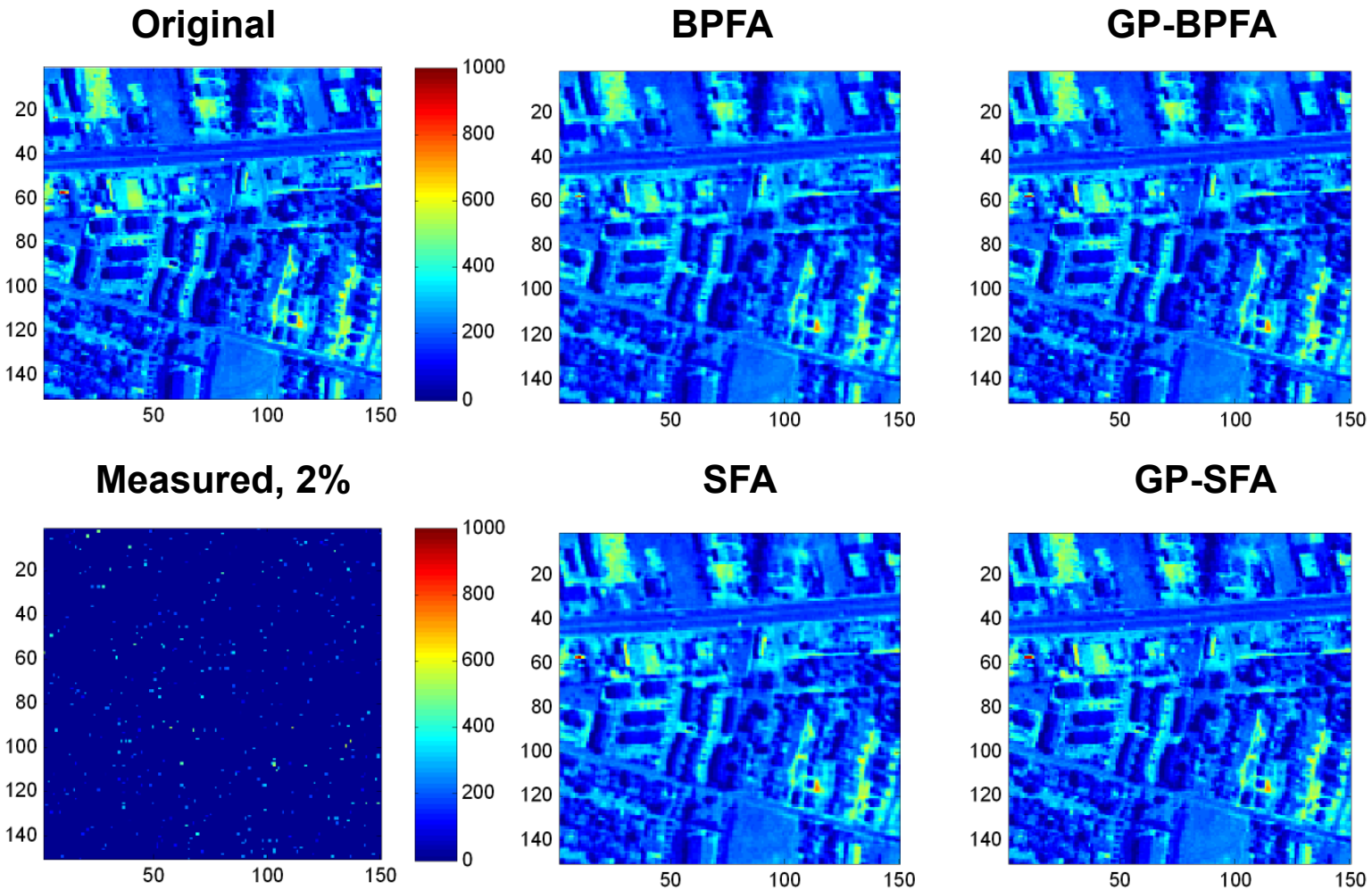
- Consider a point estimate for the parameters, that minimize the *negative* log posterior:

$$-\log p(\mathbf{D}, \{\boldsymbol{s}_i\}, \gamma_e, \lambda | \{\boldsymbol{x}_i\}) =$$

$$\sum_{i=1}^{N} \gamma_e \|\boldsymbol{x}_i - \mathbf{D}\boldsymbol{s}_i\|_2^2 + (1/\lambda) \sum_{i=1}^{N} \|\boldsymbol{s}_i\|_1 + (1/P) \sum_{k=1}^{K} \|\boldsymbol{d}_k\|_2^2$$

$$-\log \text{Ga}(\gamma_e; c, d) - \log \text{Ga}(\lambda; e, f) + \text{Const}$$

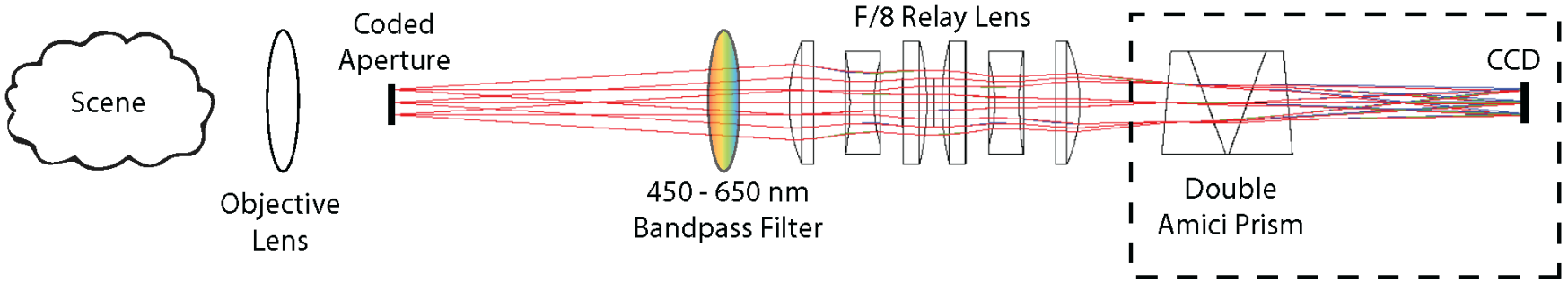
HSI Data, 162 Spectral Bands, Band 20 Shown



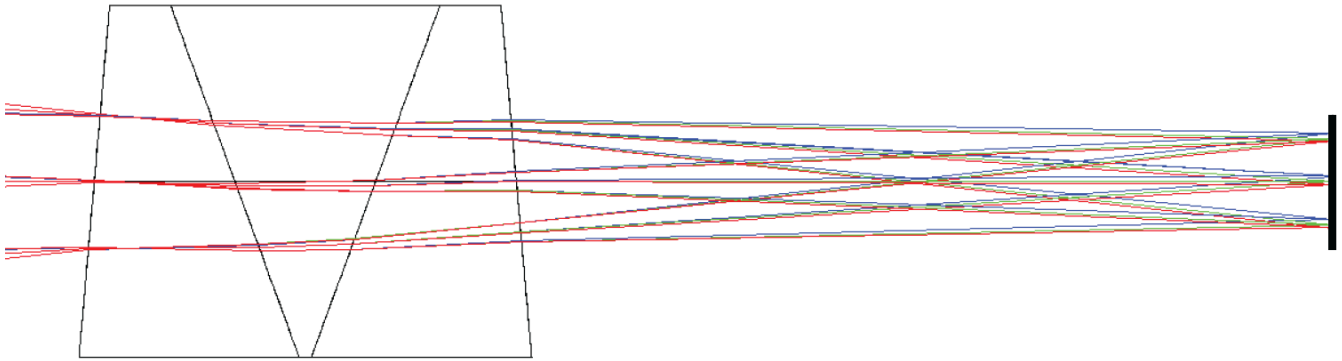
Agenda

- Dictionary learning and massive downsampling on measurement
- **Compressive hyperspectral camera**
- Compressive video
- Summary

CASSI Compressive HSI Camera

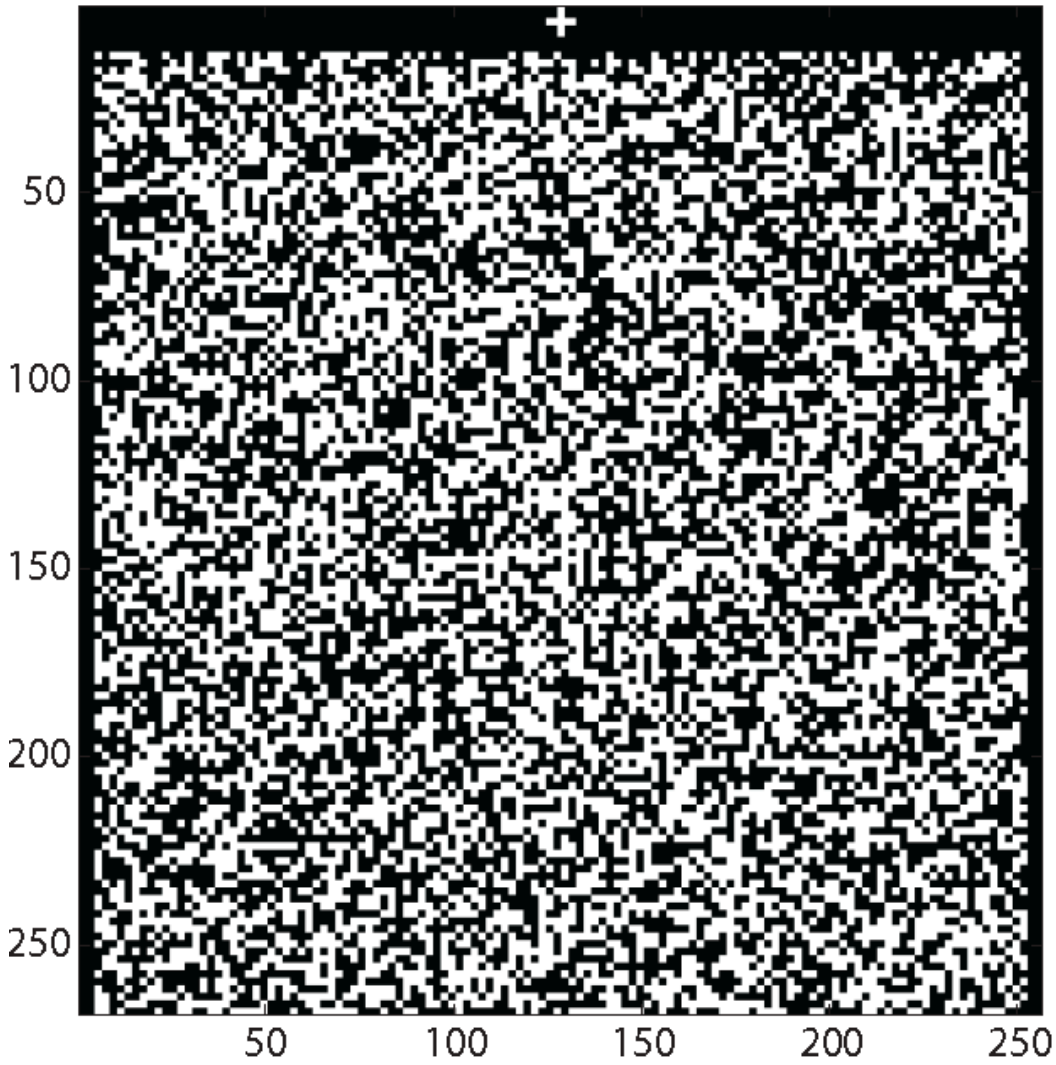


(a)

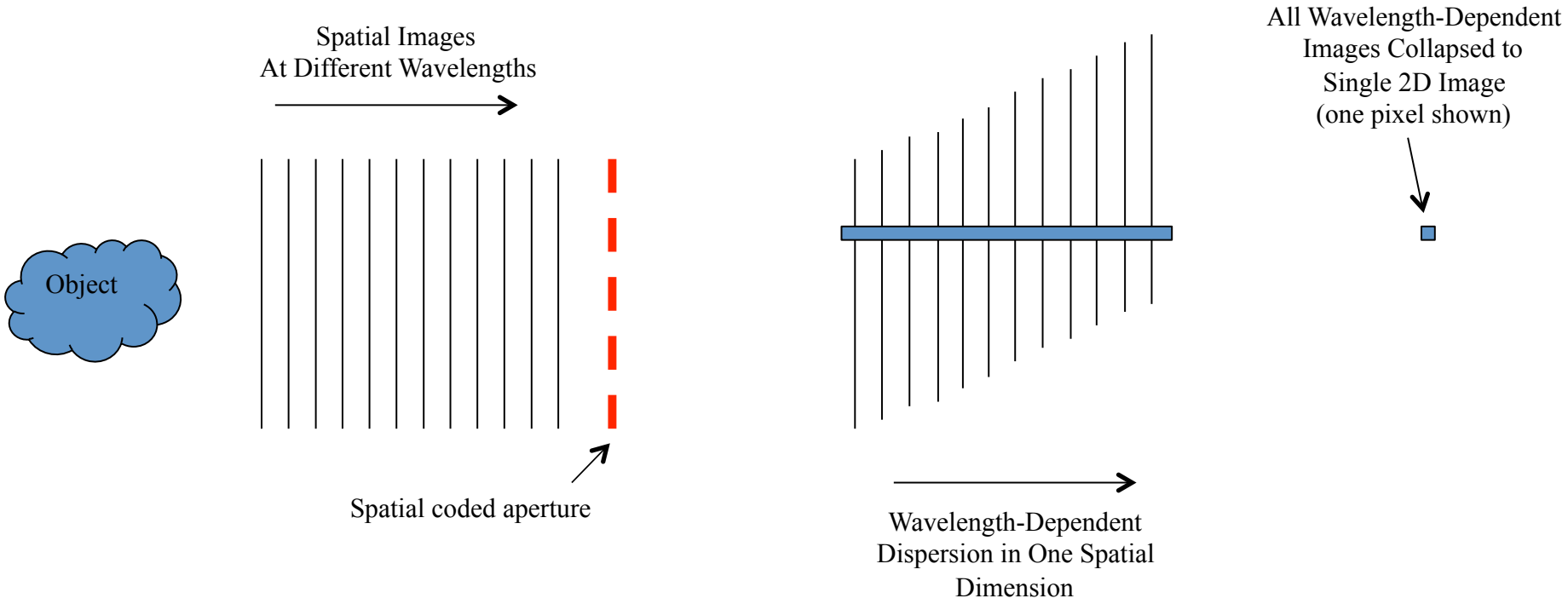


(b)

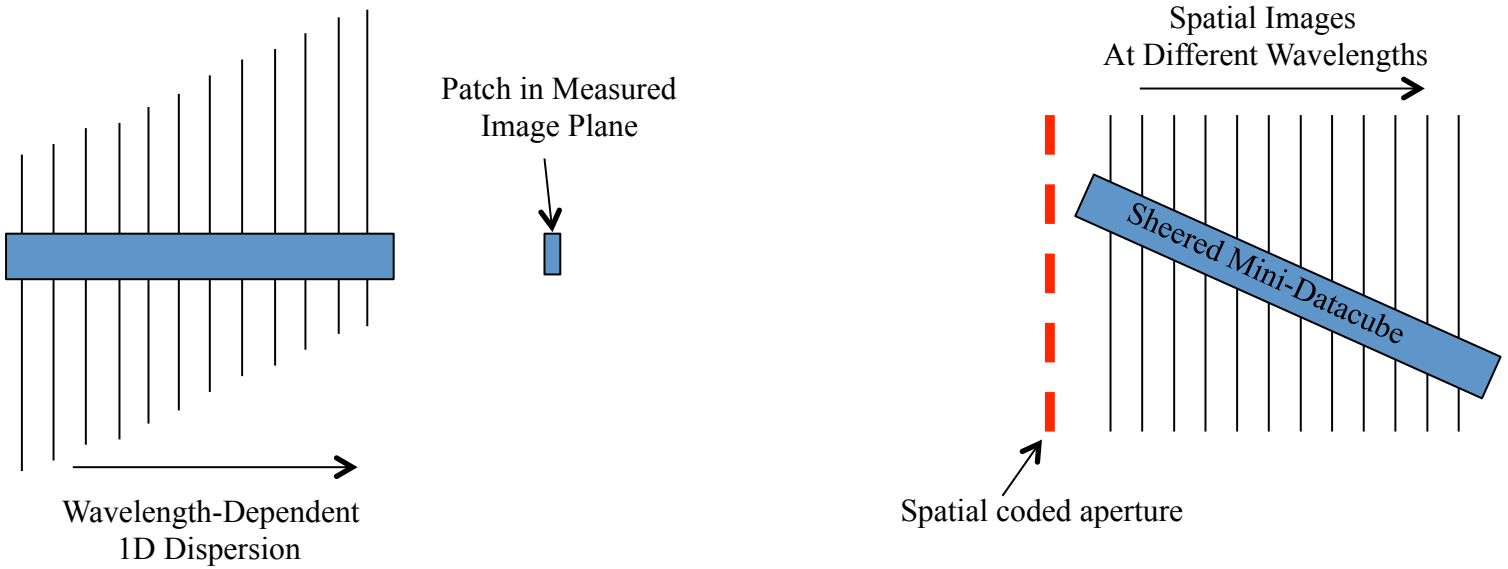
Coded Aperture



More on Measurement Construct



From Physics to Math



- Let $\boldsymbol{x}_i \in \mathbb{R}^P$ represent the pixels in the ith "sheered" mini-datacube, corresponding to the ith patch in the measurement plane
- Let $\boldsymbol{y}_i \in \mathbb{R}^p$ represent in the pixels in the measured patch, with $p \ll P$
- The measurement may be expressed as

$$\boldsymbol{y}_i = \Phi_i \boldsymbol{x}_i$$

- Definition of the sheering defined by dispersion, and Φ_i defined by coded aperture

Dictionary Learning and HSI Recovery

- From single CASSI measurement of the HSI datacube, we realize a large ensemble of patches, $\{\boldsymbol{y}_i\}_{i=1,N}$, and each may be expressed as $\boldsymbol{y}_i = \Phi_i \boldsymbol{x}_i$, where $\boldsymbol{x}_i$ is unknown sheered mini-datacube, with Φ_i known
- Assume that each mini-datacube $\boldsymbol{x}_i$ may be *sparsely* represented in an associated dictionary

$$\boldsymbol{x}_i = \mathbf{D}\boldsymbol{s}_i + \hat{\epsilon}_i$$

- Then each of the measurements may be expressed as

$$\boldsymbol{y}_i = \Phi_i \mathbf{D}\boldsymbol{s}_i + \epsilon_i$$

- Challenge: Based upon observed $\{\boldsymbol{y}_i\}_{i=1,N}$ and known $\{\Phi_i\}_{i=1,N}$, infer the *shared* dictionary $\mathbf{D}$ and the *sparse* weights $\{\boldsymbol{s}_i\}_{i=1,N}$

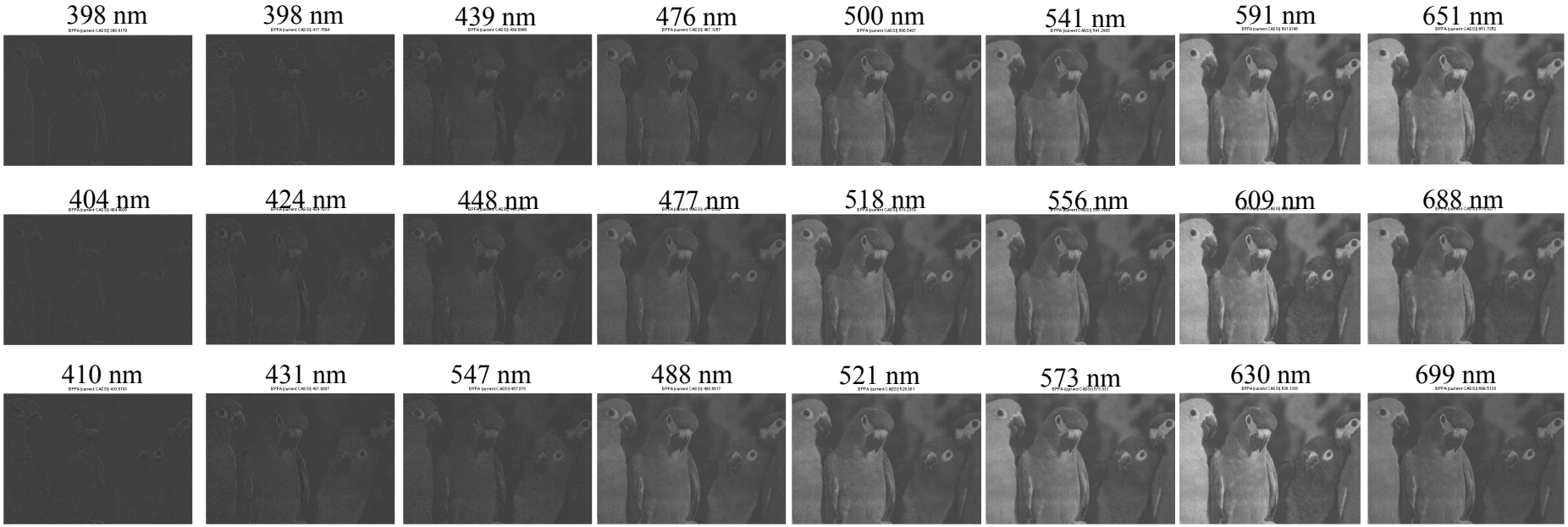
Nonparametric Bayesian Analysis

- Model setup:

$$\boldsymbol{y}_i = \Phi_i \mathbf{D} \boldsymbol{s}_i + \epsilon_i$$

- Use a beta-Bernoulli process to jointly learn the dictionary $\mathbf{D}$ and the sparse weights $\{\boldsymbol{s}_i\}$
- The fact that we share $\mathbf{D}$ for all $\{\boldsymbol{y}_i\}$ and that $\{\boldsymbol{s}_i\}$ are sparse is the key to making this analysis feasible
- Gaussian process imposed on the columns of $\mathbf{D}$, to impose spatial-spectral smoothness on the dictionary elements

Compressive Sensing Recovery



12:1 Compression

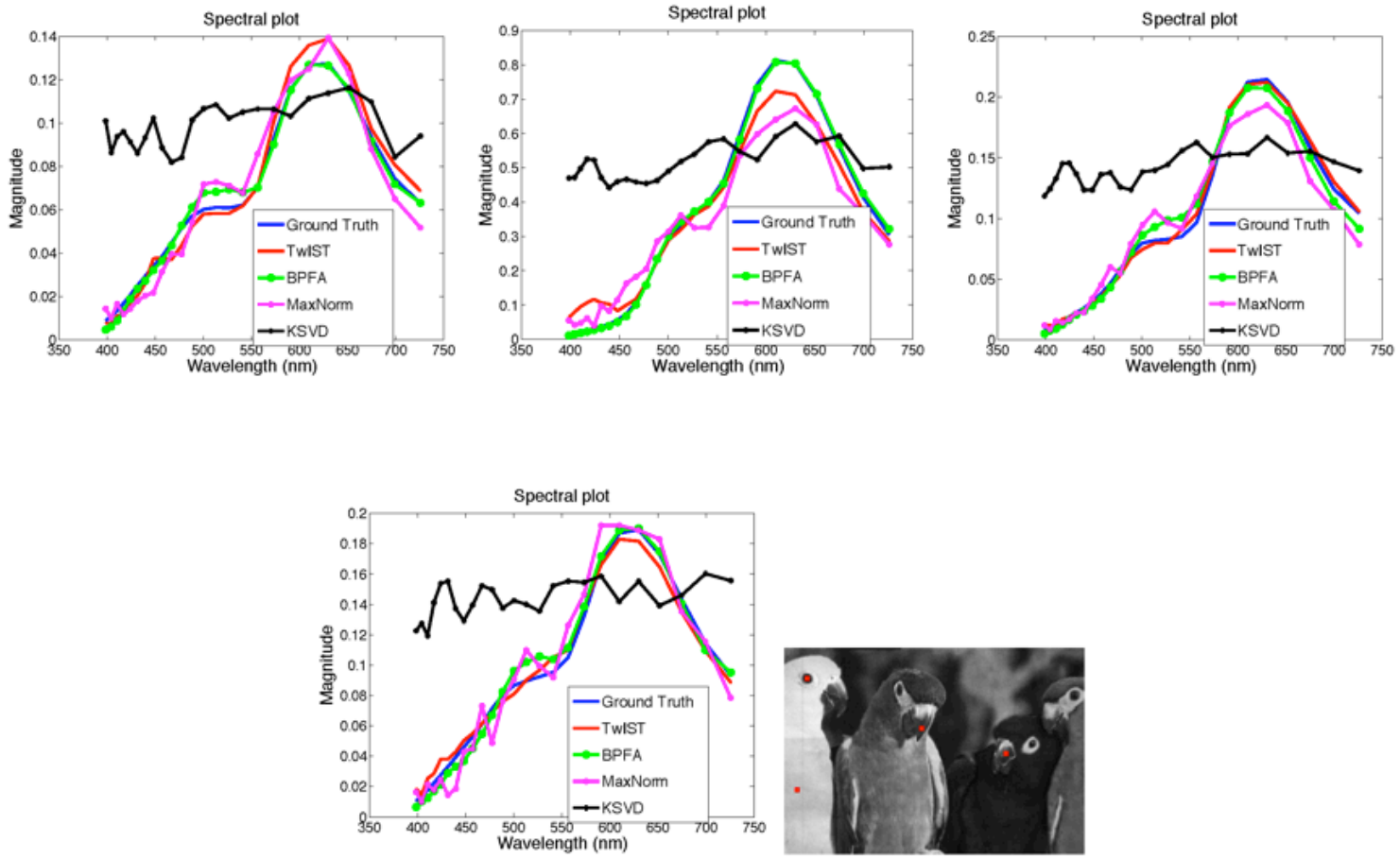


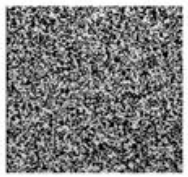
Figure: Comparison between average spectral patterns computed over small neighborhoods around four points (from synthetic birds dataset) and their reconstructions using BPFA, KSVD, MaxNorm and TwIST.

Agenda

- Dictionary learning and massive downsampling on measurement
- Compressive hyperspectral camera
- **Compressive video**
- Summary

Hardware

Objective lens

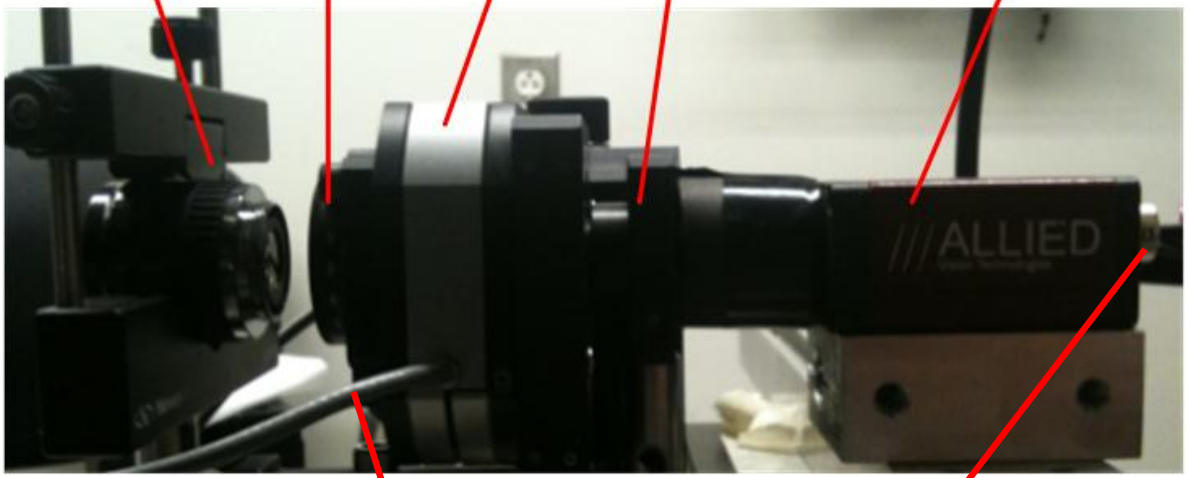


Coded aperture

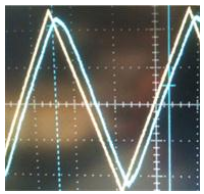
Piezo

Relay lens

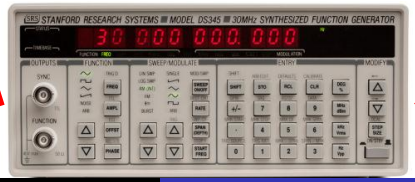
CCD



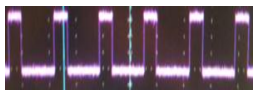
15Hz



Function generator



30fps



Formula of Physical Process

$$\mathbf{y}_{[i,j]} = \int_0^{[i,j]} \int_0^{\mathcal{T}} \Phi[\gamma - s(\gamma,t)]\boldsymbol{X}(\gamma,t) p_\gamma\left(\frac{\gamma}{\Delta}\right) p_t(t) d\gamma dt,$$
$$\forall i \in 1 \ldots m,\ \forall j \in 1 \ldots n$$

where:

(m,n)	the size of frame
$\gamma = [x,y]$	the spatial coordinates values
$\mathbf{y}_{[i,j]}$	the $[i,j]$-th pixel in the measurement $\mathbf{y}$
$s(\gamma,t)$	the instantaneous position of the coded aperture at time
Δ	the pixel area
$p_\gamma\left(\frac{\gamma}{\Delta}\right)$	the spatial sampling function
$p_t(t)$	the camera's integration window

Forward Model

For simplicity, we put the arguments of Φ and $\boldsymbol{X}$ lower as the subscripts.

$$\mathbf{y} = \begin{bmatrix} \mathbf{y}_{[1,1]} \\ \mathbf{y}_{[2,1]} \\ \vdots \\ \mathbf{y}_{[m,n]} \end{bmatrix} = \underbrace{\begin{bmatrix} \Phi_{[1,1],1} & 0 \cdots & \Phi_{[1,1],2} & 0 \cdots \cdots & 0 & \Phi_{[1,1],T} & 0 \cdots 0 \\ 0 & \ddots & \ddots & & \ddots & & 0 \\ 0 \cdots 0 & \Phi_{[m,n],1} & \cdots 0 & \Phi_{[m,n],2} & \cdots \cdots & 0 & \Phi_{[m,n],T} \end{bmatrix}}_{\Phi} \begin{bmatrix} \mathbf{x}_{[1,1],1} \\ \vdots \\ \mathbf{x}_{[m,n],1} \end{bmatrix} \text{subframe 1}$$

$$\begin{bmatrix} \mathbf{x}_{[1,1],2} \\ \vdots \\ \mathbf{x}_{[m,n],2} \end{bmatrix} \text{subframe 2}$$

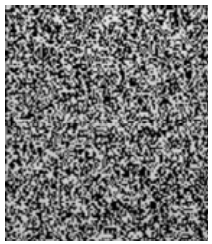
$$\begin{bmatrix} \mathbf{x}_{[1,1],3} \\ \vdots \\ \mathbf{x}_{[m,n],T-1} \end{bmatrix}$$

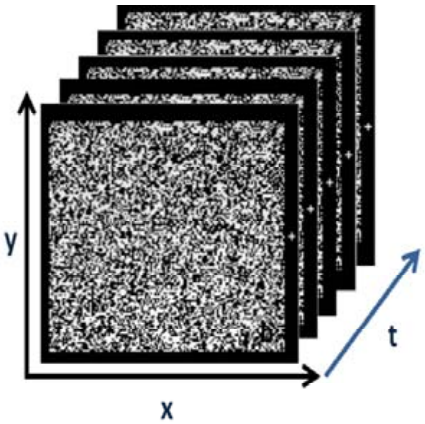
$$\begin{bmatrix} \mathbf{x}_{[1,1],T} \\ \vdots \\ \mathbf{x}_{[m,n],T} \end{bmatrix} \text{subframe } T$$

Sensed data

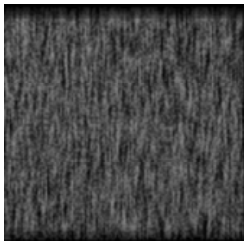
Forward matrix

Coded Aperture





N pixel shift = N times temporal resolution



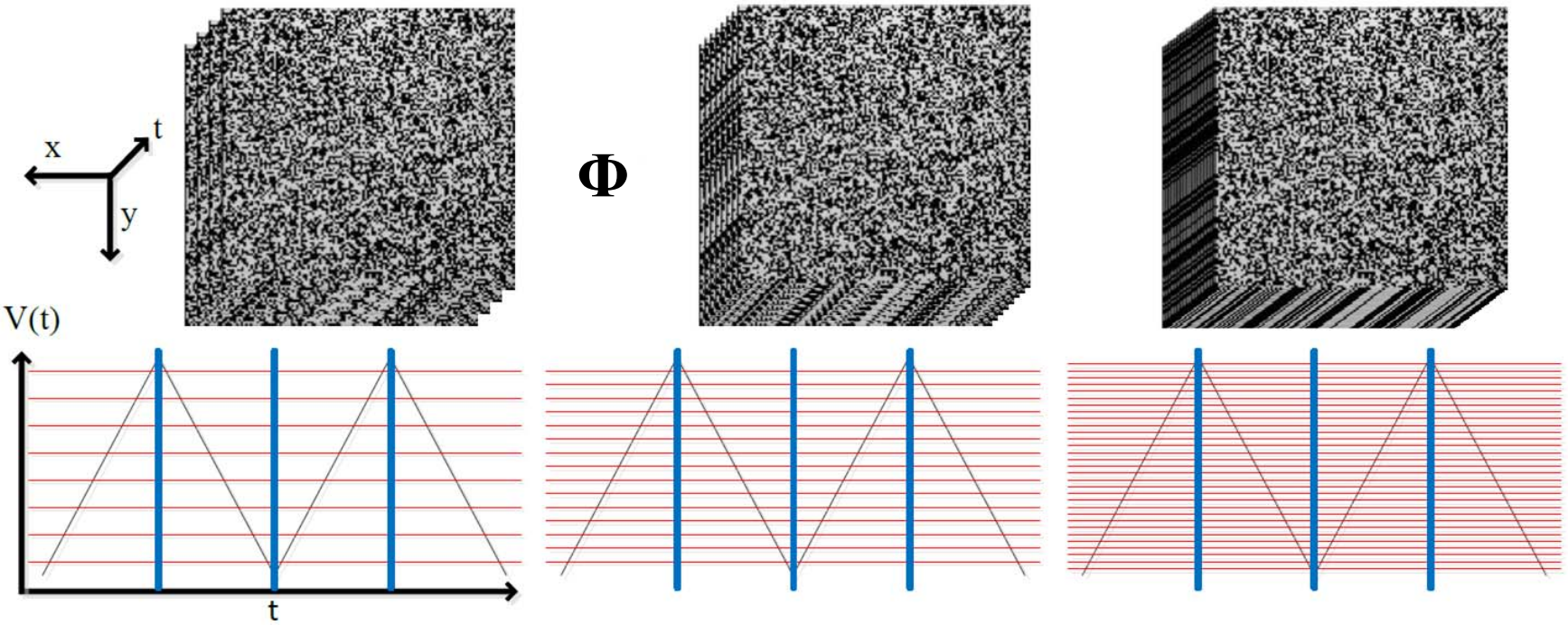
Spatial modulation

Calibration cube Φ

Spatiotemporal modulation

Illustration of mask movement in video.

Continuous Mask Movement



Coarse Temporal Sampling
T = 7

Moderate Temporal Sampling
T = 14, 28

Fine Temporal Sampling
T = 74, 148

Gaussian Mixture Model (GMM)

- GMM is a linear superposition of K Gaussian:

$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \text{ with } \sum_{k=1}^{K} \pi_k = 1, \quad 0 \leq \pi_k \leq 1. \quad (1)$$

  Variables $\pi_k, \boldsymbol{\mu}_k$ and $\boldsymbol{\Sigma}_k$ can be estimated by expectation-maximization (EM) algorithm.

- In a linear model $\mathbf{y} = \boldsymbol{\Phi}\mathbf{x} + \boldsymbol{\epsilon}$, $\boldsymbol{\epsilon} \in \mathcal{N}(\mathbf{0}, \mathbf{R})$, if $\mathbf{x} \sim p(\mathbf{x})$ in (1), then $p(\mathbf{x}|\mathbf{y})$ has the following analytical form

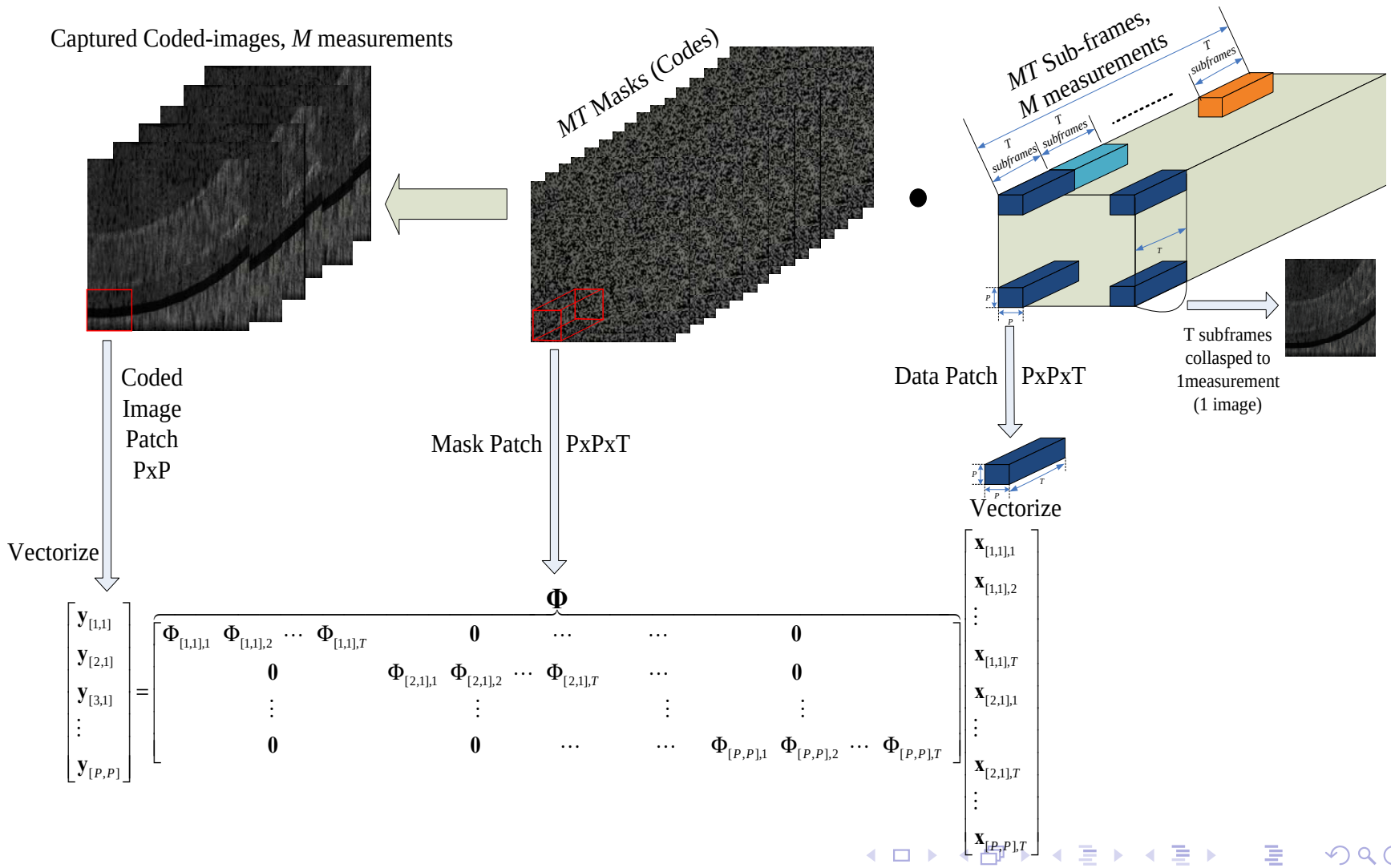
$$p(\mathbf{x}|\mathbf{y}) = \sum_{k=1}^{K} \tilde{\pi}_k \mathcal{N}(\mathbf{x}|\tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\Sigma}}_k) \quad (2)$$

  where

$$\tilde{\lambda}_k = \frac{\lambda_k \mathcal{N}(\mathbf{y}|\boldsymbol{\Phi}\mathbf{x}_k, \mathbf{R}^{-1} + \boldsymbol{\Phi}\boldsymbol{\Sigma}_k\boldsymbol{\Phi}^T)}{\sum_{l=1}^{K} \lambda_l \mathcal{N}(\mathbf{y}|\boldsymbol{\Phi}\mathbf{x}_l, \mathbf{R}^{-1} + \boldsymbol{\Phi}\boldsymbol{\Sigma}_l\boldsymbol{\Phi}^T)}$$

$$\tilde{\boldsymbol{\Sigma}}_k = (\boldsymbol{\Phi}^T\mathbf{R}\boldsymbol{\Phi} + \boldsymbol{\Sigma}_k^{-1})^{-1}, \ \tilde{\boldsymbol{\mu}}_k = \tilde{\boldsymbol{\Sigma}}_k(\boldsymbol{\Phi}^T\mathbf{R}\mathbf{y} + \boldsymbol{\Sigma}_k^{-1}\boldsymbol{\mu}_k)$$

Video Reconstruction via GMM



$$\begin{bmatrix} \mathbf{y}_{[1,1]} \\ \mathbf{y}_{[2,1]} \\ \mathbf{y}_{[3,1]} \\ \vdots \\ \mathbf{y}_{[P,P]} \end{bmatrix} = \begin{bmatrix} \Phi_{[1,1],1} & \Phi_{[1,1],2} & \cdots & \Phi_{[1,1],T} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\ \mathbf{0} & & & & \Phi_{[2,1],1} & \Phi_{[2,1],2} & \cdots & \Phi_{[2,1],T} & \cdots & \mathbf{0} \\ \vdots & & & & \vdots & & & \vdots & \vdots \\ \mathbf{0} & & & & \mathbf{0} & \cdots & \cdots & \Phi_{[P,P],1} & \Phi_{[P,P],2} & \cdots & \Phi_{[P,P],T} \end{bmatrix} \begin{bmatrix} \mathbf{x}_{[1,1],1} \\ \mathbf{x}_{[1,1],2} \\ \vdots \\ \mathbf{x}_{[1,1],T} \\ \mathbf{x}_{[2,1],1} \\ \vdots \\ \mathbf{x}_{[2,1],T} \\ \vdots \\ \mathbf{x}_{[P,P],T} \end{bmatrix}$$

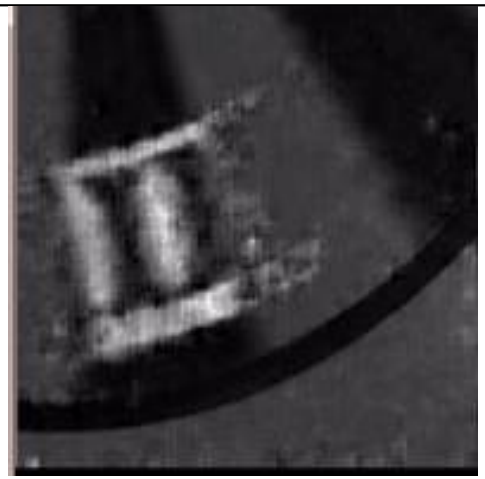
Algorithm 1:

1. Train GMM model (1) on arbitrary video patches using EM algorithm.
2. Reconstruct patches in the current time interval via (2).
3. Update GMM model (1) on the reconstructed patches using EM algorithm.
4. Go to step 2 with the next time interval, until the last interval is processed.

Note:

- *Online learning* strategy is used at Step 3.
- The algorithm can be implemented after all the acquisition, unless as discussed later we need it for adaptive.

Experimental Results

		
T=14	video	video
T=28	video	video

Note: The reconstruction results are continually improving due to *online training*.

Agenda

- Dictionary learning and massive downsampling on measurement
- Compressive hyperspectral camera
- Compressive video
- **Summary**