Compressive Hyperspectral Imaging with Side Information

Xin Yuan, Tsung-Han Tsai, Ruoyu Zhu, Patrick Llull, David Brady, and Lawrence Carin

Department of Electrical and Computer Engineering, Duke University

Durham, NC 27708-0291 USA

Abstract—**A blind compressive sensing algorithm is proposed to reconstruct hyperspectral images from spectrally-compressed measurements. The wavelength-dependent data are coded and then superposed, mapping the three-dimensional hyperspectral datacube to a two-dimensional image. The inversion algorithm learns a dictionary *in situ* from the measurements via *global-local shrinkage* priors. By using RGB images as *side information* of the compressive sensing system, the proposed approach is extended to learn a coupled dictionary from the joint dataset of the compressed measurements and the corresponding RGB images, to improve reconstruction quality. A prototype camera is built using a liquid-crystal-on-silicon modulator. Experimental reconstructions of hyperspectral datacubes from both simulated and real compressed measurements demonstrate the efficacy of the proposed inversion algorithm, the feasibility of the camera and the benefit of side information.**

Index Terms—**Compressive sensing, hyperspectral image, side information, Bayesian shrinkage, dictionary learning, blind compressive sensing, computational photography, coded aperture snapshot spectral imaging (CASSI), spatial light modulation.**

I. INTRODUCTION

Hyperspectral imaging techniques have been widely applied in various fields such as astronomy [1], remote sensing [2], and biomedical imaging [3]. Unlike ordinary imaging systems, hyperspectral imagers capture a three-dimensional (3D) datacube, *i.e.*, a 2D array of vectors that contain the spectral information at each spatial location. Inspired by compressive sensing (CS) [4], [5], researchers have adopted joint sensing and reconstruction paradigms that measure a subset of the 3D spectral datacube [6]–[9] and utilize CS inversion algorithms to retrieve a 3D estimate of the underlying hyperspectral images. This sensing paradigm follows the traditional benefits of CS - reduced data rates and system complexity at the expense of computational algorithmic development and postprocessing.

The coded aperture snapshot spectral imaging (CASSI) [7], [8] systems are examples of CS hyperspectral imaging systems. The CASSI paradigm encodes each of the datacube's spectral channels with a unique 2D pattern, which is the underlying operating principle behind code division multiple access (CDMA). CASSI systems form an image onto a coded aperture placed at an intermediate image plane to spatially modulate the datacube with high-frequency patterns (Fig. 1). A disperser placed in a pupil plane behind the coded aperture spectrally shifts the coded image, effectively granting each spectral channel onto its own unique coding pattern when multiplexed onto a monochrome sensor. Unlike other hyperspectral

imaging techniques [10]–[12], CASSI can obtain a discrete 3D estimate of the target spectral datacube from as little as a single 2D measurement. Such operation renders the system's forward model highly underdetermined; inversion requires use of CS algorithms [13]–[18]. Compressive hyperspectral imaging requires the signal under estimation to be sparse in a basis that is incoherent with the system sensing matrix [7]. The reconstruction is accomplished using optimization algorithms, such as gradient projection for sparse reconstruction (GPSR) [7] or two-step iterative shrinkage/thresholding (TwIST) [19]. GPSR assumes sparsity of the entire datacube in a fixed (wavelet) basis, while TwIST is based on a piecewise-constant spatial intensity model (when the total variation norm is used) for hyperspectral images.

Distinct from the above optimization algorithms, blind CS algorithms [20] have been applied to CASSI systems by learning dictionaries from the measurements. Blind CS inversion strategies seek to recover 3D patches of the hyperspectral datacube jointly with the shared dictionary (inferred from the measurements). Each patch is a sparse combination of the dictionary atoms. Since the dictionary is *unknown a priori*, this is called *blind* CS [21]. This blind CS model shares statistical strengths among similar image patches at different spatial locations. Additionally, this CS approach is task-driven since the learned dictionary is appropriate for different tasks [22]. In this paper, we propose a new blind CS model that imposes *compressibility* [23], rather than sparsity, on the recovered dictionary coefficients. Specifically, we use *global-local shrinkage* priors [24]–[26] on the dictionary coefficients for each patch, under a Bayesian dictionary learning framework. This setting drives the reconstructed dictionary coefficient vector to contain many small (*i.e.*, close to zero) components while allowing for a few large components, but avoiding explicit sparsity enforcement. This model is feasible for high-dimensional hyperspectral signals since it can extract more information from the limited (*in situ* learned) dictionary atoms.

The quality of the reconstructed hyperspectral images relies on the conditioning of the sensing matrix. More accurate recovery is possible with additional measurements of the same scene [19] or by using digital micromirror (DMD) arrays [27]; however, these methods increase system cost and energy consumption. This paper features a blind CS algorithm that employs the RGB image of the target scene (as *side information*) and demonstrates substantial improvements on

reconstruction quality with a single CASSI measurement. Particularly, the proposed algorithm learns a joint dictionary from a single CASSI measurement and the corresponding RGB image (of the same scene) and then reconstructs the hyperspectral datacube. Since the RGB image is strongly correlated with the hyperspectral images, this joint model will dramatically aid the dictionary learning algorithm, thereby improving the reconstruction.

Furthermore, we propose a new camera using spatial-light modulation (SLM) [28], [29] for an active coding compressive spectral imager, to jointly modulate the spatial and spectral information of the scene on the intermediate image plane. This technology differs from CASSI and DMD-modulated CS imagers in that no dispersive element is required for multiplexing the spectral channels.

The reminder of the paper is organized as follows. Section II reviews the mathematical model of CASSI and introduces the new camera. Section III develops the new blind CS algorithm to reconstruct hyperspectral images. Experimental results are presented in Section IV, and Section V summarizes the paper.
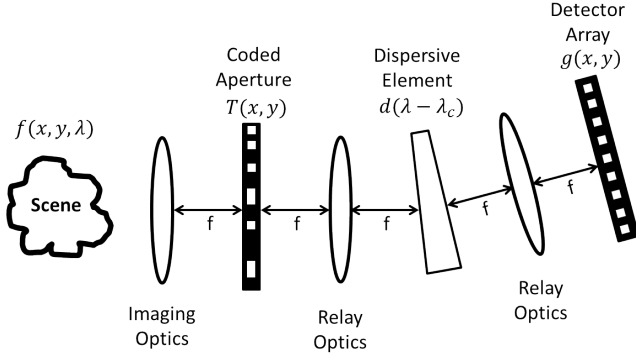

Fig. 1: Schematic of CASSI [7]. The imaging optics image the scene onto the coded aperture. The relay optics relay the image from the plane of the coded aperture to the detector through the dispersive element (figure revised from [7]).

II. HARDWARE

In this section, we first review the CASSI imaging process and then introduce the new camera. Finally, a shared mathematical model for both cameras is proposed.

A. Review of CASSI

A common CASSI design [7], [8], [19] in Fig. 1 adapts the available hardware to encode and multiplex the 3D spatiospectral information $f(x, y; \lambda)$ onto a 2D detector. This modulation process is based on physically shifting the hyperspectral datacube multiplied by the binary coding pattern $T(x, y)$ via chromatic dispersion. The coding pattern consists of an array of square, binary (100% or 0% transmission) apertures, which provide full or zero photon transmission. The camera's disperser (in traditional hyperspectral CS [6], [30], this is a prism or diffraction grating) laterally displaces the image as a

function of wavelength $d(\lambda - \lambda_c)$ of the coded aperture pattern, where $d(\lambda)$ is the lateral wavelength-dependent shift and λ_c represents the disperser's central wavelength. As previously mentioned, this coding process can be considered a form of CDMA, whereby each channel is modulated by an independent coding pattern (a shifted version of the single binary mask) on the detector plane. The detector integrates the spectrally-shifted planes along the spectral axis. Information can be recovered by separating channels based on their projected coding patterns. This multiplexing process can be represented as the following mathematical forward model:

$$g(x, y) = \int T(x + d(\lambda - \lambda_c), y) f(x + d(\lambda - \lambda_c), y; \lambda) \mathrm{d}\lambda, \quad (1)$$

where $g(x, y)$ is the continuous form of the detector measurement, representing the sum of dispersed coded images. Since the detector array has been spatially pixelated by the detector pixel pitch Δ, the discreatized detector measurement becomes:

$$g_{m,n} = \iint g(x, y) \mathrm{rect}(\frac{x}{\Delta} - m, \frac{y}{\Delta} - n) \mathrm{d}x \mathrm{d}y. \quad (2)$$

Finally, the discrete measurement for each pixel can be illustrated as:

$$g_{m,n} = \sum_j T_{(m+j-1),n} f_{(m+j-1),n,j} + \epsilon_{m,n}, \quad (3)$$

where $T_{m,n}$ is the spatially encoding pattern, $f_{m,n,j}$ represents the discretized spectral density of $f(x, y; \lambda)$, and $\epsilon_{m,n}$ is the noise.

B. A New Camera: SLM-CASSI

Here we use an SLM to encode the 3D spatial-spectral information on a 2D gray-scale detector - a similar CDMA strategy to CASSI for snapshot hyperspectral imaging. An SLM is an array of micro cells placed on a reflective layer; each cell has a nematic liquid crystal that adds pixel-level phase to the incident light as a function of voltage, thereby changing the polarization state on a per-pixel basis [28]. Each layer of the LC can be considered as a thin optical birefringence material; the orientation and the relative refractive index difference between the fast and slow axes determines the effective birefringence. Since most birefringent phase modulators are nominally sensitive to wavelength, this element can assign wavelength-dependent transmission patterns to multiplex every spectral channels for the compressive measurement. The hyperspectral slices can be separated from the coded data via CS inversion algorithms.

1) Experimental Setup: A schematic of this SLM-based CASSI system is shown in Fig. 2. An objective lens (L1) images the scene from a remote distance, and then the collimating lens (L2) and imaging lens (L3) relay the image through the polarizing beamsplitter and the achromatic quarter-wave retarder onto the SLM. The retarder serves to increase the contrast of the SLM polarization array by compensating the extra phase retardation in the liquid crystal device. The spatiospectral coding function is implemented on the SLM as an 8-bit, pseudo random, pixel-level grayscale phase retardation pattern. The phase retardation manifests
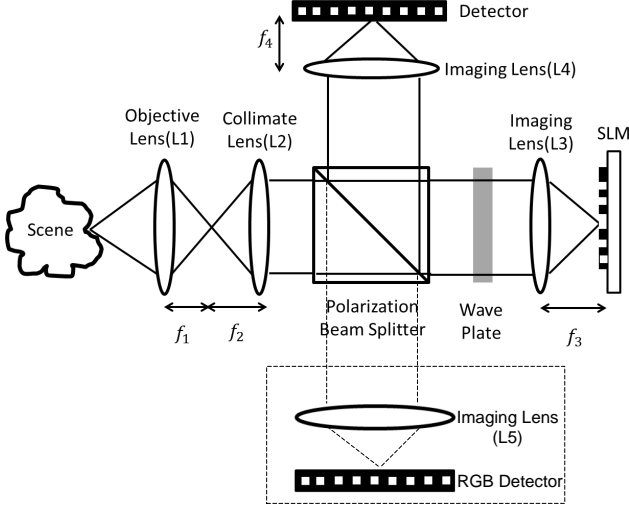
Fig. 2: The system schematic of CASSI with SLM. The parts in the dashed box represent ongoing work.

as wavelength-dependent irradiance modulation upon re-entry through the polarizing beamsplitter toward the camera. Finally, the modulated image is projected by the imaging lens (L4) onto the detector plane and then recorded by the detector. The following mathematical forward model can be used to describe the multiplexing process of the compressive sampling:

$$g(x,y) = \int T(x,y,\lambda)f(x,y;\lambda)\mathrm{d}\lambda, \qquad (4)$$

where $T(x,y,\lambda)$ are the wavelength dependent transmission patterns provided by the SLM, which can be calibrated by analyzing its electrically-controlled birefringence. Since the detector array is spatially pixellated, the $(m,n)^{th}$ detector measurement is given by:

$$g_{m,n} = \sum_j T_{m,n,j}f_{m,n,j} + \epsilon_{m,n}, \qquad (5)$$

where $T_{m,n,j}$ represents the discretized transmission patterns. The additional arm in Fig. 2 (in the dashed box) demonstrates an RGB detector used to observe the s-polarized light reflected by the polarization beam splitter. This *unmodulated* light is imaged to the RGB detector by L5. This imaging system enables concurrent measurement of a compressed spectral datacube alongside an RGB image, the latter of which is used as side information during the reconstruction. Fig. 3 is a photograph of the experimental setup. The setup includes a 60 mm objective lens (Jenoptik), a 75 mm achromatic relay lens (Edmound Optics), a polarization beam splitter (Newport), an achromatic quarter wave plate (Newport), a liquid crystal based SLM (Pluto, Holoeye), two 75 mm imaging lenses (Pentax) with 0.36 numerical aperture (NA), and a monochrome CCD camera (Pike F-421, AVT) with 2048×2048 pixels that are 7.4 μm square. A 450-680nm band pass filter (Badder) is mounted on the objective lens to block unwanted ultra-violet and infrared light (the SLM and camera are optimized for visible-range operation). The SLM provides strong modulation and effective multiplexing to fulfill the requirement of compressive sensing. This modulator is based

on a reflective Liquid Crystal on Silicon (LCoS) microdisplay technique[28], which has a 1920×1080-pixel active area with an 8 μm pixel pitch.
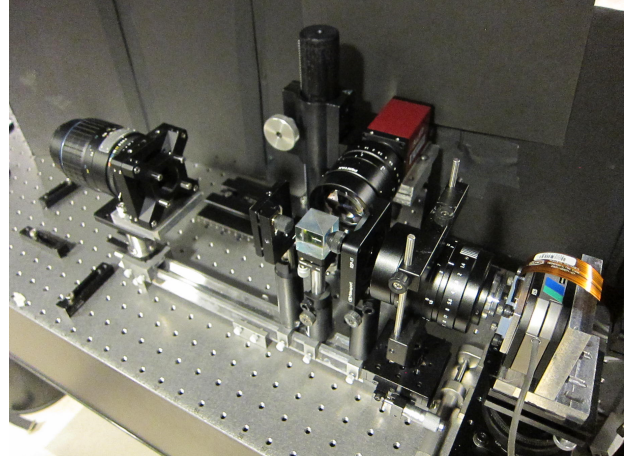

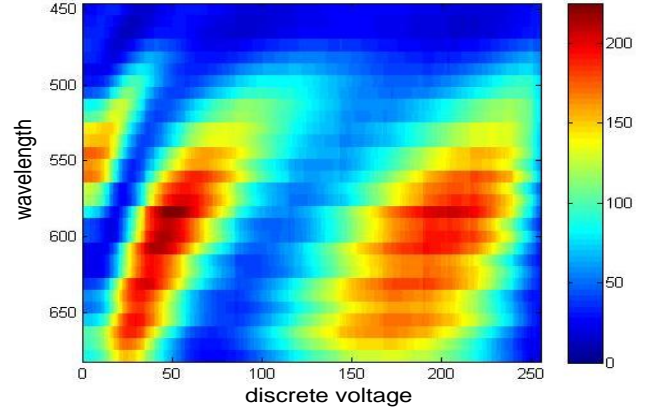Fig. 3: The experimental prototype of the system.

Fig. 4: Spectral dependence of SLM amplitude modulation. Given different voltages and incident wavelengths (in nm), the SLM combined with the optics generates corresponding transmission code. Within 450 nm and 680 nm, an 8-bit voltage variation on the SLM generates a gray scale transmission ratio between 0.08 and 0.96.

2) System Calibration: The effect of the 8-bit, pseudo-random voltage pattern on the spectral channels can be estimated by recording the system response generated by the SLM. Fig. 4 records the transmission under the monochromatic illumination and homogeneously-applied voltage on the modulator. We average the transmission across the center area of the SLM and calculate its relative transmisson. A potential advantage of this SLM is that we can change the code easily to get different modulations.[1]

---

[1]This also opens the door of adaptive compressive sensing [31]–[33]. However, after plenty experiments, we found the contribution of adaptive sensing is very limited in the CS inversion, partially due to the mismatch between the designed code and the calibration. We further found random coding performing well in our camera.

We assume that the system operator provides one-to-one mapping (1:1 magnification of the SLM pixels onto the detector) between the micro-display and the detector array. Theoretically, the system operator $T(x, y, \lambda)$ can be estimated by using SLM's calibration data and the applied voltage on the SLM. However, one might account the error in the real system projection. For example, optical aberrations and sub-pixel alignment discrepancies between the detector and SLM might break the ideal mapping and image some of the SLM pixels onto several detector pixels. Therefore, part of the transmission code might deviate from the ideal value, which results in an inaccurate system operator. Since the system operator dominates the quality of the object estimation in this inverse problem, a careful calibration of the $T(x, y, \lambda)$ is required.

A better representation of the system response can be acquired by recording the transmission pattern illuminated by each spectral band. The revised system operator $T(x, y, \lambda)$ includes all the possible coding patterns generated by the SLM, which can contribute to the data reconstruction. Here we combine the tungsten light source (LSH-T250, Horiba) with a monochromator (iHR320, Horiba) to quantize the spectral dimension into bands of finite width. Each band has a 7.5-8 nm full width at half maximum (FWHM). The scene's spectral irradiance is recorded by a fiber optics spectrometer (USB2000, Ocean Optics); these values are taken as ground truth in the experiments presented later in the paper. The number of spectral channels and their central wavelengths are determined by the grating period (1800 periods/mm) and the optical path length of the monochromator. To better represent a continuous light source, two adjacent spectral bands are separated by the monochromator's FWHM. At this calibration resolution, the system's spectrally-sensitive bandwidth (450-680nm) has been discretized into 30 spectral channels. Importantly, the spectral resolution of the reconstructed data is determined by the monochromator's resolving power during calibration; smaller FWHM values result in larger numbers of calibrated and reconstructed spectral channels. Compared to the spectral imagers that rely upon dispersive elements to spatially shear the physical code (in CASSI), this SLM-based spectral imager can adjust the compression ratio by controlling the number of spectral channels without revising the main camera's optical design. The spatial resolution of this camera is determined by the detector pixel resolution and the modulation degradation caused by the SLM. We tested the spatial resolution under another similar optical system [34] and estimated the performance of this camera. This camera can provide <0.024° angular resolution as a normal imager. After the compressed measurement, the angular resolution of the reconstructed image has been degraded to 0.027°. The spectral resolution is difficult to estimate experimentally. Base on a singular value analysis, this SLM modulation in Fig. 4 provides multiple measurement modes in spectral multiplexing. Therefore, it can effectively multiplex a spectrum which has up to several major spectral bands/peaks in the sensing region (450 to 680 nm) without losing information. This criterion can address most the spectral distributions of natural scenes, which are generally smooth and hence only require two or three modes to fully represent their major distribution in the spectral domain. The fine details can be recovered by the spectral constraint in the reconstruction algorithm, such as interpolating the relative spectral density to correct the reconstruction. However, spectrally complex objects might require more measurement modes to effectively multiplex the sampling. The RGB side information may be used as parallel sampling to provide the extra modes. Similarly, using multi-shot sampling with different SLM codes can increase the numbers of the modes as well. Finally, from a hardware perspective, increasing the thickness of the birefringence material can provide more spectrally complex modulation, which can increase the number of effective measurement modes.

C. Shared Mathematical Model of CASSI and SLM-CASSI

Assuming we are measuring a hyperspectral datacube $\mathbf{X} \in \mathbb{R}^{N_x \times N_y \times N_\lambda}$, where N_λ signifies the number of spectral channels of spatial extent $N_x \times N_y$. We can interpret both CASSI and SLM-CASSI measurements as a summation of the 3D Hadamard product of the datacube with the different modulation codes, $\mathbf{M} = \sum_{j=1}^{N_\lambda} \mathbf{X}_j \odot \mathbf{C}_j$. Here, $\mathbf{C}_j$ represents a shifted version of the same code in CASSI [20]; however, in SLM-CASSI each spectrally-varying modulation $\mathbf{C}_j$ depends on the SLM response rather than a spectral shift introduced by a dispersing prism. For each pixel, both (3) and (5) can be represented as:

$$\mathbf{M}(m, n) = \sum_{j=1}^{N_\lambda} \mathbf{X}_j(m, n) \mathbf{C}_j(m, n). \qquad (6)$$

The pixel-wise modulation of CASSI results in feasible patch-based reconstruction algorithms.

Since the SLM-CASSI sensing paradigm reconstructs N_λ spectral images from a single measurement $\mathbf{M}$, the compression ratio of the CASSI and SLM-CASSI systems discussed above is N_λ. The compressive sampling process in both spectral imaging systems can be represented by the same matrix described in the following section.

III. RECONSTRUCT HYPERSPECTRAL IMAGES WITH BLIND COMPRESSIVE SENSING

In this section, we develop a new blind Bayesian CS model to reconstruct hyperspectral images from CASSI or SLM-CASSI measurements. The proposed model is a generalized framework which can also be used in denoising and inpainting problems, among others [35].

A. Dictionary Learning Model

We decompose the 3D hyperspectral datacube into N small patches with size $n_x \times n_y \times N_\lambda$. In vector form, these patches are denoted $\boldsymbol{x}_n \in \mathbb{R}^P, \forall n = 1, \ldots, N$, with $P = n_x n_y N_\lambda$. We seek a dictionary $\mathbf{D} \in \mathbb{R}^{P \times K}$ with K atoms such that $\boldsymbol{x}_n = \mathbf{D}\boldsymbol{s}_n$, based on the captured image $\mathbf{M}$. For each patch, we have the corresponding 2D measurement patch $\{\mathbf{M}_n\}_{n=1}^N \in \mathbb{R}^{n_x \times n_y}$ and the 3D mask (coding) patch

$\{\mathbf{C}_n\}_{n=1}^{N} \in \{0,1\}^{n_x \times n_y \times N_\lambda}$. Considering the vectorized format of the measurement for n^{th} patch $\boldsymbol{y}_n \in \mathbb{R}^{n_x n_y}$, and the corresponding mask patches $\{\boldsymbol{W}_n^{(j)}\}_{n=1}^{N} \in \{0,1\}^{n_x n_y}, \forall j = 1, \ldots, N_\lambda$ (each $\boldsymbol{W}_n^{(j)}$ is the vector form of jth 2D slice in $\mathbf{C}_n$), we have the measurement model

$$\boldsymbol{y}_n = \boldsymbol{\Psi}_n \boldsymbol{x}_n + \boldsymbol{\epsilon}_n, \tag{7}$$

$$\boldsymbol{\Psi}_n = \left[\text{diag}\{\boldsymbol{W}_n^{(1)}\}, \ldots, \text{diag}\{\boldsymbol{W}_n^{(N_\lambda)}\}\right], \tag{8}$$

$$\mathbf{Y} = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_N], \quad \mathbf{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N], \tag{9}$$

where $\boldsymbol{\epsilon}_n$ represents the additive Gaussian noise, and we model it as $\boldsymbol{\epsilon}_n \sim \mathcal{N}(0, \alpha_0^{-1}\mathbf{I}_P)$, with α_0 denoting the noise precision.[2] We place a diffuse gamma prior on α_0

$$\alpha_0 \sim \text{Ga}(c_0, d_0), \tag{10}$$

where (c_0, d_0) are hyperparameters. Taking account of the dictionary learning model, (7) can be written as:

$$\boldsymbol{y}_n = \boldsymbol{\Psi}_n \mathbf{D} \boldsymbol{s}_n + \boldsymbol{\epsilon}_n, \tag{11}$$

where $\boldsymbol{s}_n$ is a vector of coefficients describing the decomposition of the signal $\boldsymbol{x}_n$ in terms of dictionary atoms. Given the measurement set $\mathbf{Y}$ and the forward matrices $\{\boldsymbol{\Psi}_n\}_{n=1}^{N}$, we aim to jointly learn $\mathbf{D}, \{\boldsymbol{s}_n\}_{n=1}^{N}$, and the noise precision parameter α_0 to recover $\mathbf{X}$. One key difference of our model compared to other blind CS work [21] is that each patch has a unique $\boldsymbol{\Psi}_n$, which is inspired from our cameras since the mask is generated randomly and $\boldsymbol{\Psi}_n$ also takes account of the system calibration.

We model each dictionary atom as a draw from a Gaussian distribution,

$$\mathbf{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K],$$
$$\boldsymbol{d}_k \sim \mathcal{N}(0, \frac{1}{P}\mathbf{I}_P), \quad \forall k = 1, \ldots, K. \tag{12}$$

The coefficients $s_{k,n}$ are assumed drawn from the marginalized distribution

$$p(s_{k,n}|\tau_n, \Phi_{k,n})$$
$$= \int \mathcal{N}(s_{k,n}; 0, \tau_n^{-1}\alpha^{-1})\text{InvGa}(\alpha; 1, (2\Phi_{k,n})^{-1})d\alpha \tag{13}$$
$$= \frac{1}{2}\sqrt{\frac{\tau_n}{\Phi_{k,n}}} \exp\left(-|s_{k,n}|\sqrt{\frac{\tau_n}{\Phi_{k,n}}}\right),$$

where $\text{InvGa}(\cdot)$ denotes the inverse-gamma distribution. The parameter $\tau_n > 0$ is a "global" scaling for all coefficients of the nth patch, and $\Phi_{k,n}$ is a "local" weight for the kth coefficient of that patch. We place a gamma prior $\text{Ga}(g_0, h_0)$ on $\Phi_{k,n}$, where one may set the hyperparameters (g_0, h_0) to ensure that most $\Phi_{k,n}$ are small. This encourages most $s_{k,n}$ to be small. By maximizing the log posterior to obtain a point estimate for the model parameters, one observes that the log of the prior in (13) corresponds to adaptive Lasso regularization [37].

[2]The extension of our model to non-uniform denoising problem [36] is straightforward, i.e., by imposing spatially-varying noise models for different patches.

Equivalently to (13), the model for $s_{k,n}$ may be represented in the hierarchical form

$$s_{k,n}|\tau_n, \alpha_{k,n} \sim \mathcal{N}(0, \tau_n^{-1}\alpha_{k,n}^{-1}), \tag{14}$$
$$\alpha_{k,n}|\Phi_{k,n} \sim \text{InvGa}(1, (2\Phi_{k,n})^{-1}), \tag{15}$$
$$\tau_n \sim \text{Ga}(a_0, b_0), \tag{16}$$
$$\Phi_{k,n} \sim \text{Ga}(g_0, h_0), \tag{17}$$

where latent variables $\{\alpha_{k,n}\}$ are included in the generative model, instead of marginalizing them out as in (13). A vague/diffuse gamma prior is placed on the scaling parameters τ_n. Despite introducing the latent variables $\{\alpha_{k,n}\}$, the form in (15) is convenient for computation, and with an appropriate choice of (g_0, h_0), most $\{\alpha_{k,n}\}$ are encouraged to be large. The large $\alpha_{k,n}$ corresponds to small $s_{k,n}$; this model imposes that most $s_{k,n}$ are small, i.e. it imposes *compressibility*. Note that (14) is different from the model used in [38], where a single τ is used for all patches; here, we impose different compressibility for each patch by inferring a unique τ_n, thus providing flexibility.

In order to automatically infer the number of necessary dictionary atoms, we can replace (11) with

$$\boldsymbol{y}_n = \boldsymbol{\Psi}_n \mathbf{D} \boldsymbol{\Lambda} \boldsymbol{s}_n + \boldsymbol{\epsilon}_n, \tag{18}$$
$$\boldsymbol{\Lambda} = \text{diag}[\nu_1, \ldots, \nu_K], \quad \nu_k \sim \mathcal{N}(0, \eta_k^{-1}), \tag{19}$$
$$\eta_k = \prod_{j=1}^{k} \tilde{\eta}_j, \quad \forall k = 1, \ldots, K; \quad \tilde{\eta}_j \sim \text{Ga}(e_0, f_0). \tag{20}$$

The multiplicative gamma prior (MGP) [39] used above is developed to stochastically increase the precision η_k as k increases. During inference, we observe that as k increases, ν_k tends to zero. This results in an approximate ordering of ν_k by weight. Based on this weight, we can infer the importance of the columns of $\mathbf{D}$. The other use of the ν_k's is to avoid over-fitting during the learning procedure. We can update a fraction of $\mathbf{D}$ by the weight of ν_k and discard the atoms with small weights to reduce the computational cost (refer to the Appendix for the inferred ν_k and learned dictionary $\mathbf{D}$).

B. The Statistical Model

The full statistical model of the proposed blind CS approach is:

$$\boldsymbol{y}_n \sim \mathcal{N}(\boldsymbol{\Psi}_n \mathbf{D} \boldsymbol{\Lambda} \boldsymbol{s}_n, \alpha_0^{-1}\mathbf{I}_P), \tag{21}$$
$$\mathbf{D} = [\boldsymbol{d}_1, \ldots, \boldsymbol{d}_K], \quad \boldsymbol{d}_k \sim \mathcal{N}(0, \frac{1}{P}\mathbf{I}_P), \tag{22}$$
$$s_{k,n} \sim \mathcal{N}(0, \tau_n^{-1}\alpha_{k,n}^{-1}\alpha_0^{-1}), \tag{23}$$
$$\alpha_{k,n} \sim \text{InvGa}(1, (2\Phi_{k,n})^{-1}), \tag{24}$$
$$\boldsymbol{\Lambda} = \text{diag}[\nu_1, \ldots, \nu_K], \tag{25}$$
$$\nu_k \sim \mathcal{N}(0, \eta_k^{-1}), \quad \eta_k = \prod_{j=1}^{k} \tilde{\eta}_j, \tag{26}$$
$$\tau_n \sim \text{Ga}(a_0, b_0), \quad \alpha_0 \sim \text{Ga}(c_0, d_0), \tag{27}$$
$$\tilde{\eta}_j \sim \text{Ga}(e_0, f_0), \quad \Phi_{k,n} \sim \text{Ga}(g_0, h_0), \tag{28}$$

where broad priors are placed on the hyperparameters $(a_0, b_0, c_0, d_0, e_0, f_0, g_0, h_0)$, i.e., $a_0 = \cdots = h_0 = 10^{-6}$.

Note in (23), the coefficients are also scaled to the noise precision α_0.

Let $\boldsymbol{\Theta} = \{\mathbf{D}, \mathbf{S}, \boldsymbol{\Lambda}, \boldsymbol{\epsilon}\}$ denote the parameters to be inferred. The log of the joint posterior may be expressed as:

$$
-\log p(\boldsymbol{\Theta}|\mathbf{Y})
$$
$$
= \frac{\alpha_0 \sum_n \|\boldsymbol{y}_n - \boldsymbol{\Psi}_n \mathbf{D}\boldsymbol{\Lambda}\boldsymbol{s}_n\|_2^2}{2} + \frac{P \sum_k \|\boldsymbol{d}_k\|_2^2}{2} + \log \mathrm{Ga}(\alpha_0) \tag{29}
$$
$$
+ \frac{\alpha_0 \sum_{n,k} \tau_n \alpha_{k,n} s_{k,n}^2}{2} + \sum_n \log \mathrm{Ga}(\tau_n)
$$
$$
+ \sum_{k,n} \log \mathrm{Ga}(\Phi_{k,n}) + \sum_{n,k} \log \mathrm{InvGa}(\alpha_{k,n}; 1, \frac{1}{2\Phi_{k,n}}) \tag{30}
$$
$$
+ \frac{\sum_k \eta_k \nu_k^2}{2} + \sum_k \log \mathrm{Ga}(\tilde{\eta}_k). \tag{31}
$$

The terms in (29) are widely employed in optimized-based dictionary learning [36], [40], [41]. The first term imposes an ℓ_2 fit between the model and the observed data $\boldsymbol{y}_n$ and the second term imposes an ℓ_2 regularization on the dictionary atoms. The third term regularizes the relative importance of the aforementioned two terms in (29) via the weighting α_0 (updated during the inference). The terms in (30) impose both compressibility (via the global-shrinkage parameter τ_n and local-shrinkage parameter $\alpha_{k,n}$) and an ℓ_2 regularization on $s_{k,n}$. Finally, the terms in (31) impose both ℓ_2 regularization and shrinkage on ν_k.

C. Related Models

The shrinkage manifested by (30) is the most distinct aspect of the proposed model relative to previous optimization-based (and Bayesian-based [20], [35]) approaches. These other approaches effectively impose sparsity through ℓ_1 regularization on s_n. This is done by replacing terms in (30) with $\gamma_s \sum_n \|\boldsymbol{s}_n\|_1$. To solve an optimization-based analog to the proposed approach, one seeks to minimize the objective function

$$
\mathcal{L}(\boldsymbol{\Theta}) = \frac{\alpha_0 \sum_n \|\boldsymbol{y}_n - \boldsymbol{\Psi}_n \mathbf{D}\boldsymbol{\Lambda}\boldsymbol{s}_n\|_2^2}{2}
$$
$$
+ \frac{\sum_k \|\boldsymbol{d}_k\|_2^2}{2} + \gamma_s \sum_n \|\boldsymbol{s}_n\|_1, \tag{32}
$$

the parameters α_0 and γ_s are typically set by hand (e.g., via cross-validation). One advantage of the Bayesian framework is that we infer posterior distributions for α_0 and γ_s (in our model this is $(\tau_n, \alpha_{k,n})$), along with similar posterior estimates for all model parameters without cross-validation. We also note that if we replace (30) by the spike-slab prior as used in [35], the model will reduce to the beta process factor analysis (BPFA) [42]. As opposed to this spike-slab prior, which imposes sparsity directly, our shrinkage prior imposes compressibility, and it is more appropriate to the high dimensional hyperspectral image patches. The *global-local* shrinkage prior used here can extract more useful information

from the limited dictionary atoms[3].

Other forms of shrinkage priors like Gaussian scale model and Laplace scale mode can be found in [43]–[46]. Aiming to better mimic the marginal behavior of discrete mixture priors, the global-local shrinkage priors [24], [25] have been developed to offer sufficient flexibility in high-dimensional settings, which inspires our model.

D. Inference

Due to local conjugacy, we can write the conditional posterior distribution for all parameters of our model in closed form, making the following Markov Chain Monte Carlo (MCMC) inference based on Gibbs sampling a straightforward procedure.

1) Sampling $\boldsymbol{d}_k$:

$$
p(\boldsymbol{d}_k|-) \propto \mathcal{N}(\boldsymbol{\mu}_{d_k}, \boldsymbol{\Sigma}_{d_k}); \tag{33}
$$
$$
\boldsymbol{\Sigma}_{d_k} = \left[P\mathbf{I}_P + \alpha_0 \nu_k^2 \sum_{n=1}^N s_{k,n}^2 \boldsymbol{\Psi}_n^T \boldsymbol{\Psi}_n \right]^{-1},
$$
$$
\boldsymbol{\mu}_{d_k} = \alpha_0 \boldsymbol{\Sigma}_{d_k} \nu_k \sum_{n=1}^N s_{k,n} \boldsymbol{\Psi}_n^T \boldsymbol{y}_{n,-k},
$$
$$
\boldsymbol{y}_{n,-k} = \boldsymbol{y}_n - \boldsymbol{\Psi}_n \mathbf{D}\boldsymbol{\Lambda}\boldsymbol{s}_n + \boldsymbol{\Psi}_n \boldsymbol{d}_k \nu_k s_{k,n}.
$$

2) Sampling $s_{k,n}$:

$$
p(s_{k,n}|-) \propto \mathcal{N}(\mu_{s_{k,n}}, \sigma_{s_{k,n}}^2); \tag{34}
$$
$$
\sigma_{s_{k,n}}^2 = \left(\tau_n \alpha_{k,n} \alpha_0 + \alpha_0 \lambda_k^2 \boldsymbol{d}_k^T \boldsymbol{\Psi}_n^T \boldsymbol{\Psi}_n \boldsymbol{d}_k \right)^{-1},
$$
$$
\mu_{s_{k,n}} = \alpha_0 \sigma_{s_{k,n}}^2 \nu_k \boldsymbol{d}_k^T \boldsymbol{\Psi}_n^T \boldsymbol{y}_{n,-k}.
$$

3) Sampling τ_n:

$$
p(\tau_n|-) \propto \mathrm{Ga}\left(a_0 + \frac{1}{2}K, b_0 + \frac{1}{2}\sum_{k=1}^K s_{k,n}^2 \alpha_{k,n}\alpha_0 \right). \tag{35}
$$

4) Sampling $\alpha_{k,n}$:

$$
p(\alpha_{k,n}|-) \propto \mathrm{IG}\left(\sqrt{\frac{1}{\Phi_{k,n}s_{k,n}^2 \tau_n \alpha_0}}, \frac{1}{\Phi_{k,n}} \right) \tag{36}
$$

where $\mathrm{IG}(\cdot)$ denotes the inverse-Gaussian distribution.

5) Sampling $\Phi_{k,n}$:

$$
p(\Phi_{k,n}|-) \propto \mathrm{GIG}(2h_0, \alpha_{k,n}^{-1}, g_0 - 1), \tag{37}
$$

where $\mathrm{GIG}(x; a, b, p)$ is the generalized inverse Gaussian distribution

$$
\mathrm{GIG}(x; a, b, p) = \frac{(a/b)^{\frac{p}{2}}}{2K_p(\sqrt{ab})} x^{p-1} \exp\left(-\frac{1}{2}\left(ax + \frac{b}{x}\right) \right),
$$

and $K_p(\theta)$ is the modified Bessel function of the second kind

$$
K_p(\theta) = \int_0^\infty \frac{1}{2}\theta^{-p} t^{p-1} \exp\left(-\frac{1}{2}(t + \frac{\theta^2}{t}) \right) dt.
$$

---

[3]We did experiments of denoising and inpainting with benchmark color images and compared with K-SVD [36] and BPFA [35]. The results are shown in the appendix. The proposed algorithm constantly performs better than the above two methods.

6) Sampling α_0:

$$p(\alpha_0|-) \quad \propto \quad \mathrm{Ga}(c_1, d_1); \qquad (38)$$

$$c_1 \quad = \quad c_0 + \frac{1}{2}\sum_n \|\mathbf{\Psi}_n\|_0 + \frac{1}{2}KN,$$

$$d_1 \quad = \quad d_0 + \frac{1}{2}\sum_n \|\boldsymbol{y}_n - \mathbf{\Psi}_n\mathbf{D}\mathbf{\Lambda}\boldsymbol{s}_n\|_2^2$$

$$+ \frac{1}{2}\sum_n\sum_k s_{k,n}^2\tau_n\alpha_{k,n},$$

where $\|\mathbf{\Psi}_n\|_0$ denotes the number of nonzero entries in $\mathbf{\Psi}_n$, and "-" refers to the conditioning parameters of the distributions.

The update equations of MGP related parameters $\{\nu_k, \eta_k, \tilde{\eta}_k\}$ can be found in [39] (also listed in the Appendix). In applications where speed is important, we can use all conditional posteriors including those above to derive a variational Bayes (VB) inference algorithm for our model, which loosely amounts to replace conditioning variables with their corresponding moments. Details of the VB procedure can be found in the Appendix.

E. RGB Images as Side Information

While reconstructing the hyperspectral images is challenging (the compression ratio is $N_\lambda : 1$ when a single measurement is available), an RGB image of the same scene can be obtained easily by an off-the-shelf color camera in the unused path of the system (i.e. directly reflecting off the polarizing beamsplitter) as shown in Fig. 2. We here consider using the RGB image as side information to aid the reconstruction.

In the case of an additional side RGB camera, the measurement is a joint dataset composed of the RGB image and the CASSI measurement; the compression ratio can be considered as $(N_\lambda + 3) : 4$. The new measurement model is:

$$\underbrace{\begin{bmatrix} \mathbf{Y} \\ \mathbf{Y}^{(\mathrm{rgb})} \end{bmatrix}}_{\overset{\mathrm{def}}{=}\tilde{\mathbf{Y}}} = \begin{bmatrix} \mathbf{\Psi} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \underbrace{\begin{bmatrix} \mathbf{X} \\ \mathbf{X}^{(\mathrm{rgb})} \end{bmatrix}}_{\overset{\mathrm{def}}{=}\tilde{\mathbf{X}}} + \boldsymbol{\epsilon}, \qquad (39)$$

where $\{\mathbf{Y}^{(\mathrm{rgb})}, \mathbf{X}^{(\mathrm{rgb})}\}$ are the patch format RGB image. Considering each patch,

$$\begin{bmatrix} \boldsymbol{y}_n \\ \boldsymbol{y}_n^{(\mathrm{rgb})} \end{bmatrix} = \begin{bmatrix} \mathbf{\Psi}_n & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}\begin{bmatrix} \boldsymbol{x}_n \\ \boldsymbol{x}_n^{(\mathrm{rgb})} \end{bmatrix} + \tilde{\boldsymbol{\epsilon}}_n, \qquad (40)$$

where $\{\boldsymbol{y}_n^{(\mathrm{rgb})}\}_{n=1}^N$ is the vectorized form of the RGB image patch corresponding to $\{\boldsymbol{y}_n\}_{n=1}^N$ and similar for $\{\boldsymbol{x}_n^{(\mathrm{rgb})}\}_{n=1}^N$. The dictionary learning method proposed in Section III-A is now performed on the joint dataset $\tilde{\mathbf{X}} = \tilde{\mathbf{D}}\tilde{\mathbf{S}}$ to learn the super-dictionary $\tilde{\mathbf{D}} \in \mathbb{R}^{\tilde{P} \times K}$, where $\tilde{P} = P + 3n_xn_y$.

Given $\tilde{\mathbf{Y}}$, the proposed model learns $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{S}}$ simultaneously. For the recovery of hyperspectral images, we select the top P rows from the product of $\tilde{\mathbf{D}}$ and $\tilde{\mathbf{S}}$. Averaging the overlapping patches and reformatting the data to 3D give us the desired hyperspectral images. We aim to get both clear images at each wavelength and correct spectra for each pixel, which are both embedded in the dictionary atoms. Since the RGB images are fully observed and embed the same scene as the hyperspectral images, this side information will facilitate the algorithm to learn a better dictionary and therefore improving the reconstruction quality.

Another way to use the RGB image is treating each R, G and B channel as a superposition of the hyperspectral images with different weights corresponding to the quantum efficiency of the R, G, and B sensors. However, these quantum efficiencies may be different for each camera. Here, we aim to propose a general and robust framework to collaborate RGB images with CASSI measurements. Therefore, the formulation in (39) is adopted.

In our experimental setup, a grayscale image may also be used as side information. As the RGB camera is inexpensive and carrying richer spectral information, we only consider the RGB case in the following experiments. In the results presented below, we consider that RGB image is measured separately by an additional camera. We are now modifying our SLM-CASSI system to capture the RGB image and the compressed hyperspectral image simultaneously as illustrated in Fig. 2.

IV. EXPERIMENTAL RESULTS

In this section, we verify the proposed blind CS inversion algorithm on diverse datasets. For now, we solely consider the case where a single measurement (*i.e.*, the compressed hyperspectral image) is available[4]. The proposed blind CS algorithm is compared with: i) two-step iterative shrinkage/thresholding (TwIST) [13] (using a 2D total variation norm on images at each wavelength[5]), and ii) the linearized Bregman algorithm [15]. The linearized Bregman approach respectively regularizes the spectral dimension and spatial dimensions with ℓ_1-norms of the discrete cosine transformation (DCT) and wavelet transformation coefficients. Note that TwIST and linearized Bregman are *global* algorithms; it is necessary to load the entire data into RAM when performing inversion. Conversely, our proposed method is *locally*-(patch) based; the online learning is feasible and the inversion can be performed in batch-mode [47]. We also performed the inversion via TwIST augmented with side information as used in the proposed algorithm. However, since the linearized Bregman requires the DCT for inversion, the RGB image is not appropriate to be mixed with the hyperspectral image channels (in the transformation). This precludes the comparison of inversion by linearized Bregman with side information.

We employ a spatial patch size $n_x = n_y = 8$. The compression ratio, N_λ, depends on the dataset. The proposed model can learn the dictionary atom number K from the data. During experimentation, we have found that setting $K = 64$ usually provides good results. We have verified that further increasing K does not significantly change the outcome of

---

[4]Please note that [20] did not show the algorithm proposed therein working with a single measurement. In this paper, we focus on the single measurement case and our model can readily be used in multiple measurements scenario. We also tried the method proposed in [20] and the results are worse than ours.

[5]We also tried the 3D total variation (TV) on the entire datacube, the results are a little bit worse than the 2D TV. Therefore, we only show the 2D TV results of TwIST.

TABLE I: Reconstruction PSNR (dB) of various scenes with different algorithms. Each feature shows the mean value and standard derivation of the PSNRs of reconstructed hyperspectral images across wavelength channels.

Scene					
Shrinkage w/o RGB	30.9673±5.0328	26.6319±5.5290	32.8013±4.1525	42.9004±5.4524	34.5532±5.1652
Shrinkage w/ RGB	**32.6175±4.0417**	**34.0775±4.3469**	**35.1950±2.9645**	**47.5275±4.7856**	**35.8182±2.0722**
TwIST w/o RGB	29.4249±2.9894	23.2787±5.8818	25.2030±3.4791	30.1709±3.4108	21.6566±2.2755
TwIST w/ RGB	31.6377±3.4455	30.3905±2.4102	31.6377±3.4455	31.1753±3.5349	31.6646±2.7473
Bregman	30.7027±4.6553	27.2907±5.4581	32.8595±3.8964	27.1390±4.0307	34.0448±4.5355
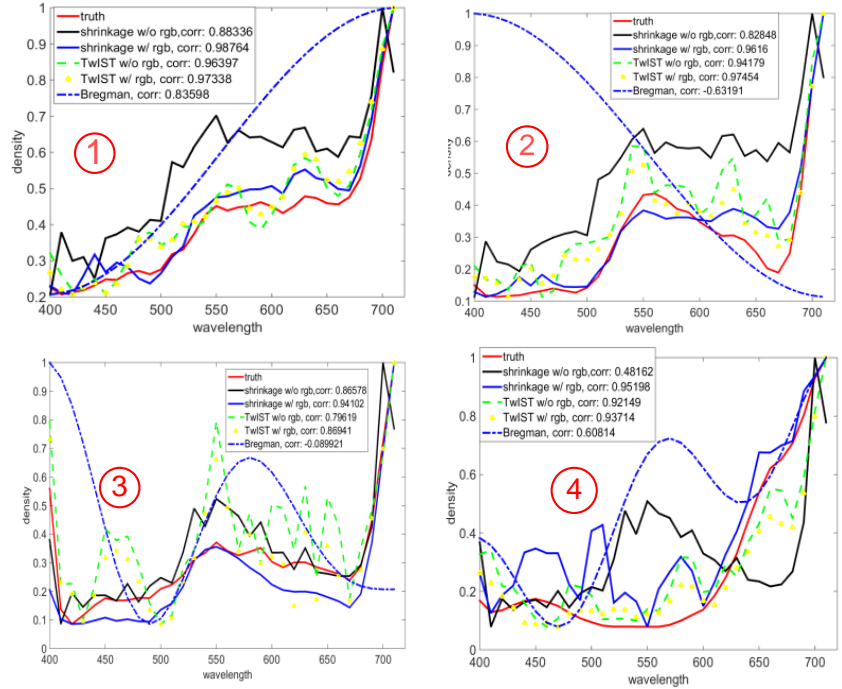
Fig. 5: Reconstructed spectra with different algorithms compared to ground truth. "corr" denotes the correlation between the reconstructed spectra and the truth.

our methods. All code used in the experiments was written in Matlab and executed on a 3.3GHz desktop with 16GB RAM.

We evaluate the proposed algorithm on synthetic data presented in Section IV-A and the real data captured both by the CASSI [8], [19] and the proposed SLM-CASSI camera Section IV-B. We denote our method as "shrinkage" in the experiments (in figures and ta bles) as the shrinkage prior is used in our model. To evaluate the algorithm's performance, we use the PSNR of the reconstructed images at different wavelengths and the correlation between the reconstructed spectrum and the true (reference) spectrum.

The computational time of our model is similar to the BPFA model used in [20] and linearized Bregman. TwIST provides faster, but generally worse results than our algorithm in terms of the metrics mentioned above. Quantitatively, linearized Bregman and TwIST require about 20 minutes and 14 minutes, respectively, to reconstruct the bird data (size $768 \times 1024 \times 24$).

A. Simulation Data

We use hyperspectral images encoded with a random binary (Bernoulli(0.5)) mask to simulate the CASSI measurements. The RGB images are available and aligned well with the hyperspectral images.

1) Nature scenes: We first consider the hyperspectral images of natural scenes used in [48][6]. There are $N_\lambda = 33$ channels (400-720nm with a 10nm interval) and we resize each image to $N_x = N_y = 512$. The PSNR of each reconstructed spectral channel with mean values and standard deviations (across wavelength channels) are presented for each algorithm in Table I. It can be seen that the proposed shrinkage method outperforms the others, especially once the RGB images are used as side information. Fig. 5 plots the reconstructed spectrum of selected blocks (the average spectrum of pixels inside the block is used) in the scene compared to the ground truth.

[6]http://personalpages.manchester.ac.uk/staff/d.h.foster/Hyperspectral_images_of_natural_scenes_04.html
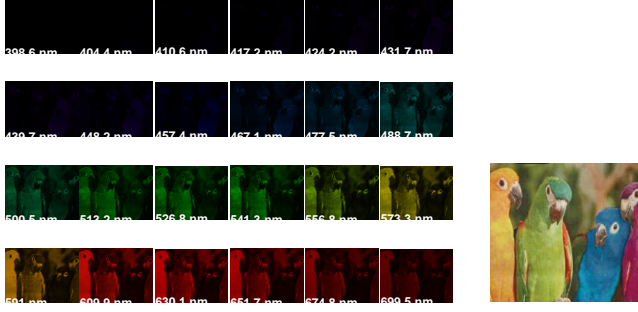
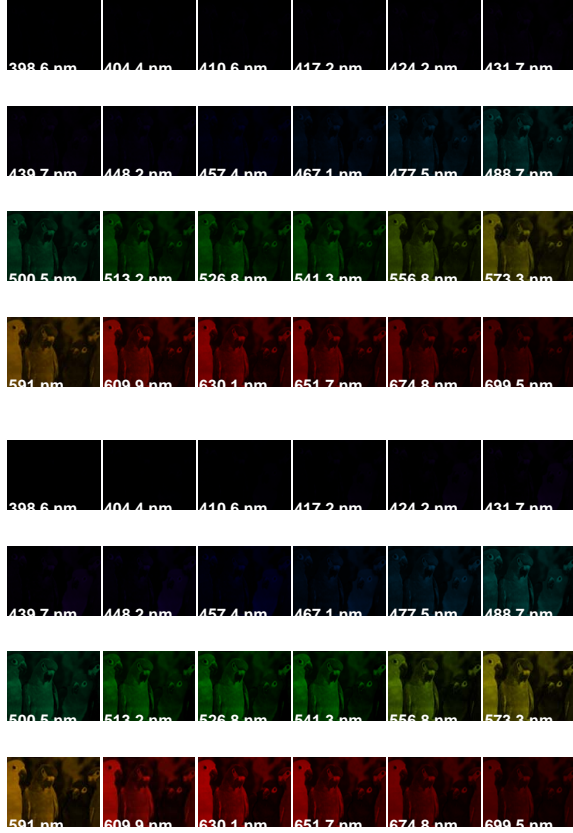Fig. 6: Left: the hyperspectral images (ground truth), right: the RGB image.

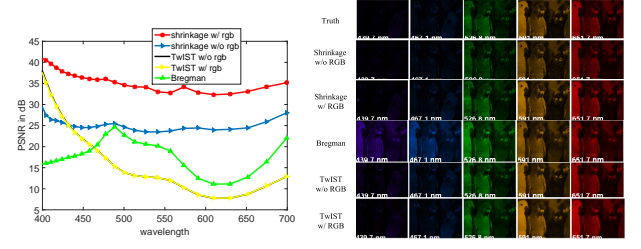Fig. 7: Reconstructed images without RGB image (top) and with RGB image (bottom).

Fig. 8: Left: PSNRs of the reconstructed images at each wavelength using different methods. Right: selected channels of reconstructed images compared to truth.
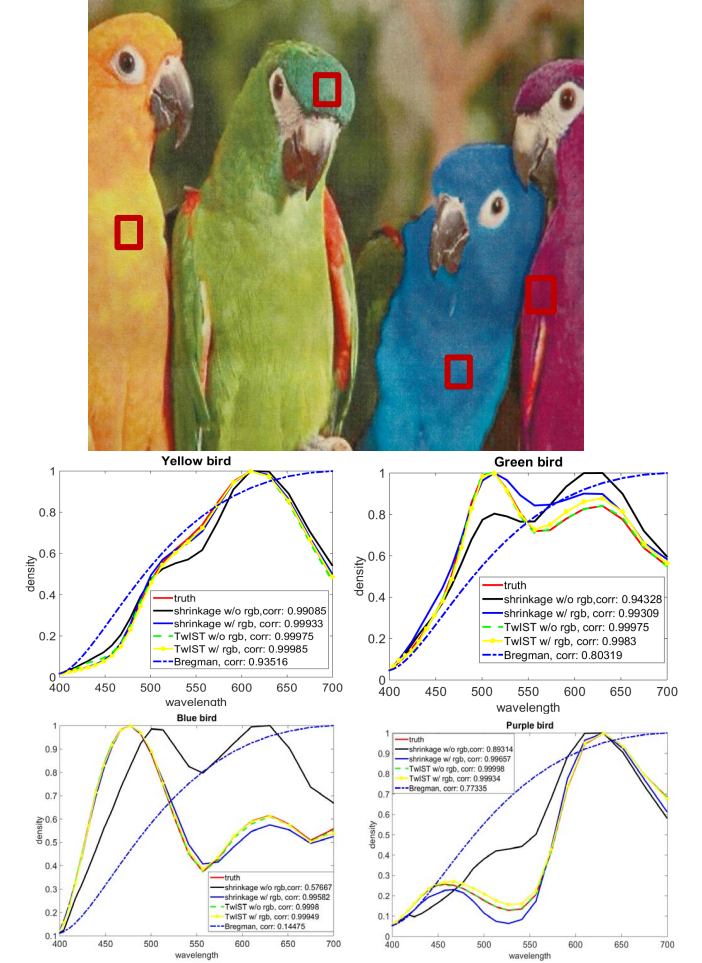

Fig. 9: Reconstructed spectra of different birds with various algorithms.

It can be seen that: i) though TwIST sometimes provides the lowest PSNR, the reconstructed spectrum is usually correct; ii) the spectrum reconstructed by the proposed algorithm is improved significantly when side information is provided; iii) the linearized Bregman presents the worst spectrum, mainly due to the DCT used in the spectral domain for inversion.

2) Bird data: Next we consider the 24-channel bird data (Fig. 6) measured by a hyperspectral camera in [20]. The RGB image is also available. Fig. 7 shows the reconstructed images using the proposed algorithm with and without side information. The left part of Fig. 8 plots the reconstruction PSNR of each channel using different algorithms. Again, our shrinkage blind CS method coupled with the RGB image

provides the best result. The right part of Fig. 8 shows the reconstructed images of five selected channels using the different algorithms. It can be seen that the TwIST results are characterized by lost details (over smoothed) of the birds; our model's results without using the RGB images appear noisy. Fig. 9 depicts the reconstructed spectra of different birds with different algorithms. It can be seen that both TwIST (both without and with the RGB image) and our algorithm with the RGB image provide very good matches to the ground truth, while the proposed model without RGB images and linearized
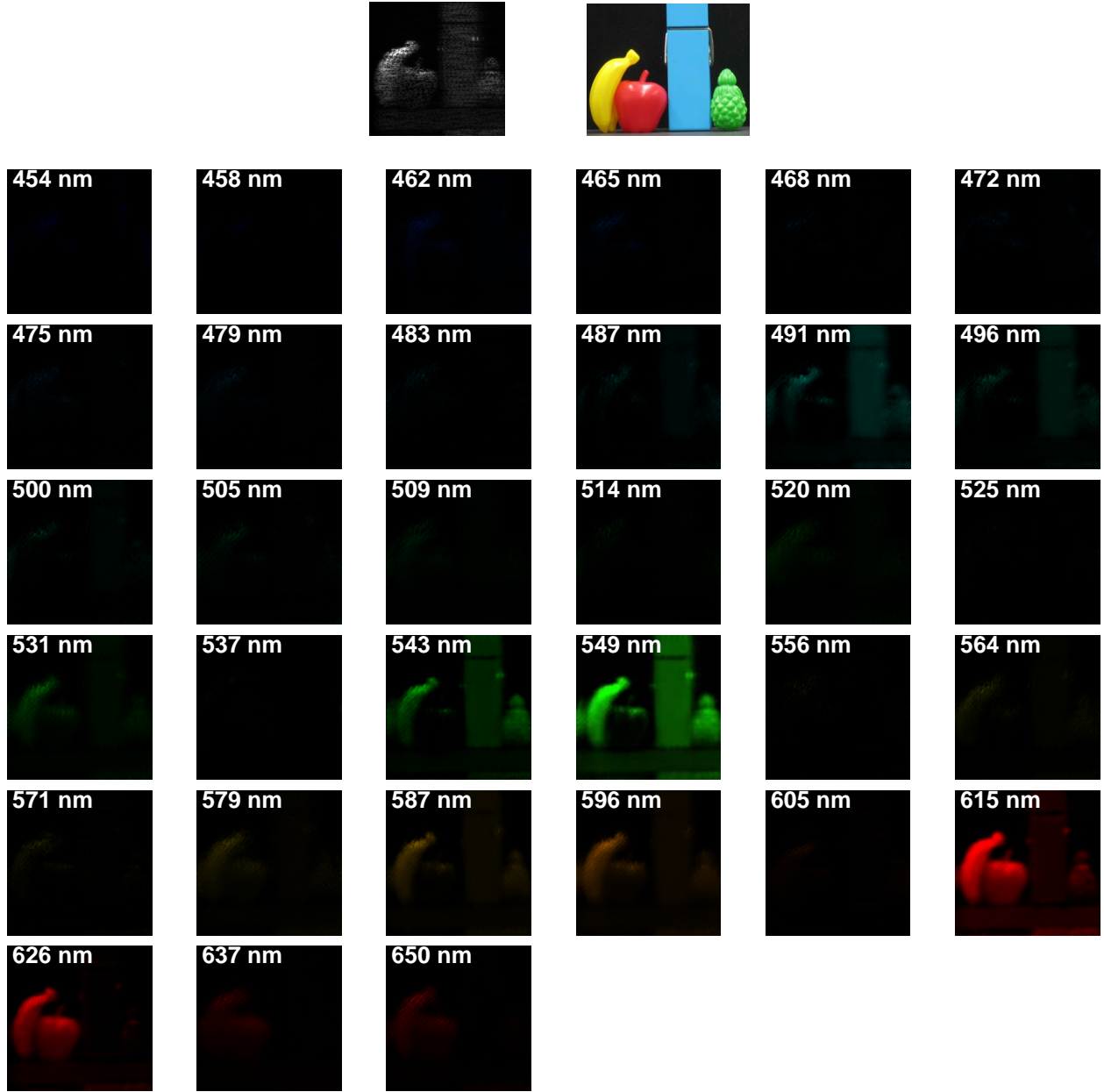
Fig. 10: Real data: measurement and reconstruction without side information (the RGB image).

Bregman do not represent the spectrum well.

B. Real Data

In this section, we apply the proposed algorithm to real data captured by our cameras (both the original CASSI camera and the new SLM-CASSI camera).

1) Object data: We first demonstrate our algorithm on data taken by the original CASSI system [8][7]. In these experiments, the reconstructions have 33 spectral channels (marked on the reconstruction in Fig. 10) and $N_x = N_y = 256$. There are 4 objects in the scene, a red apple, a yellow banana, a blue

[7]http://www.disp.duke.edu/projects/CASSI/experimentaldata/index.ptml

stapler and a green pineapple. Fig. 10 shows the reconstruction of the proposed algorithm without the RGB image. Since the RGB image is not well-aligned with the CASSI measurement, we don't show the result with side information and we found the reconstruction with the RGB image (not aligned) looks similar to this one due to the simple scene used in this experiment. Fig. 11 compares selected reconstructed images inverted by different algorithms. It can be seen that the proposed algorithm provides more detail of the scene (notice the clear apple stem). Fig. 12 plots the reconstructed spectra of the four objects. TwIST [8] has yielded accurate spectra (both without and with side information); our algorithm (without side information) provides similar results. The linearized Bregman algorithm does not reconstruct this real data well;
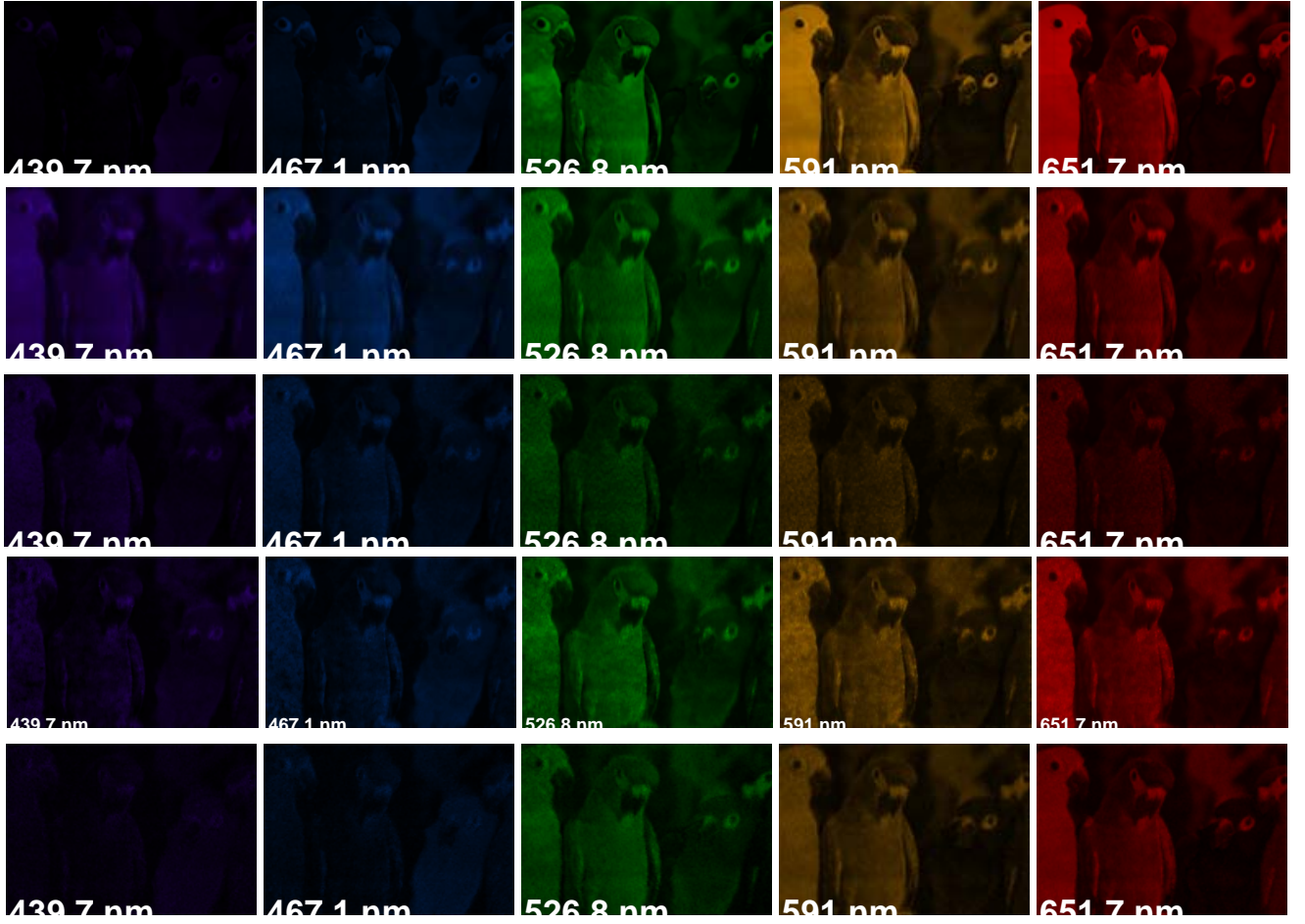
Fig. 13: Real data, row 1: reference; and reconstruction with, row 2: shrinkage without the RGB image; row 3: TwIST without the RGB image; row 4: TwIST with the RGB image; row 5: shrinkage with the RGB image. Notice the blue bird in the first two images for a good spectrum recovery.
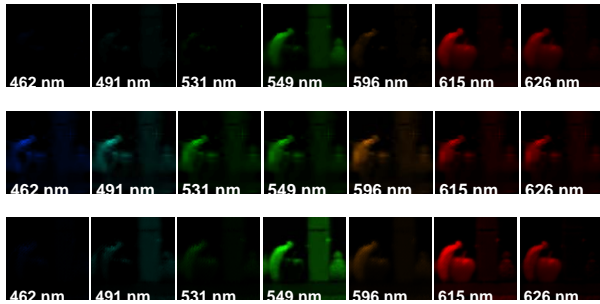

Fig. 11: Real data: selected channels of reconstructed hyperspectral images. Top row: TwIST, middle row: linearized Bregman, bottom row: shrinkage without RGB. Notice that only the bottom row provides the clear "apple stem".
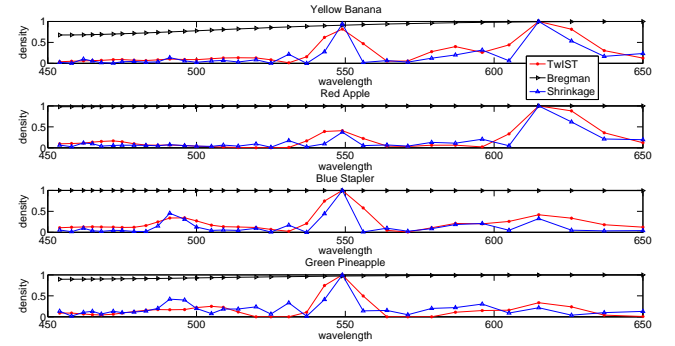

Fig. 12: Real data: spectra of the reconstructed hyperspectral images for the object data.

we don't not show its results in the following experiments.

2) Bird data: We again consider the bird data, now captured by the multiframe-CASSI camera [19]. The RGB image is aligned manually with the CASSI measurement ($N_x = 703, N_y = 1021, N_\lambda = 24$). We plot the reconstructed images in Fig. 13; the spectra are shown in Fig. 14. It can be

seen clearly that our proposed method with side information provides the best results with respect to both image clarity and spectral accuracy. Without the use of side information, the proposed model yields clear images but reconstructs the spectra poorly. This verifies the benefit of the side information in real data.

3) Data with CASSI-SLM : Now we consider the dataset captured by the proposed camera. Since no RGB image is available, we only show the results of our algorithm without side information. Fig. 15 shows the reconstructed spectrum and selected frames for the M&M dataset ($N_x = 512, N_y = 784, N_\lambda = 30$)[8]. As with the original CASSI experiments, we use a fiber optic spectrometer (USB2000, Ocean Optics) to provide the reference spectra for the targets. It can be seen again our algorithm provides better results than TwIST, both for images and spectrum.

As a last example, we show the reconstructed images with our method for the berry data ($N_x = N_y = 2048, N_\lambda = 40$) in Figure 16. Notice that the leaf reconstructs are prevalent in the green channels (520~590nm), while the berries (red) appear in the red portion of the spectrum (610~680nm).

C. Benchmark Images for Inpainting and Denoising

To demonstrate the general applications of our proposed dictionary learning model, we show denoising and inpainting results from the 8-bit benchmark color images tested in [35], [36]. For inpainting, we show PSNR values of the restored images at various observed data ratios (20% means 80% pixels are missing) in Table II. Noisy images corrupted with zero-mean Gaussian noise with different standard deviations σ (the image pixel values are in $\{0, \ldots, 255\}$) are restored with PSNR values shown in Table III. Our proposed algorithm consistently provides the highest PSNR. Importantly, few iterations of our VB inference algorithm are required to obtain good results; 20 iterations are used for denoising and 100 iterations are used for inpainting. The restored images are shown in the Appendix.
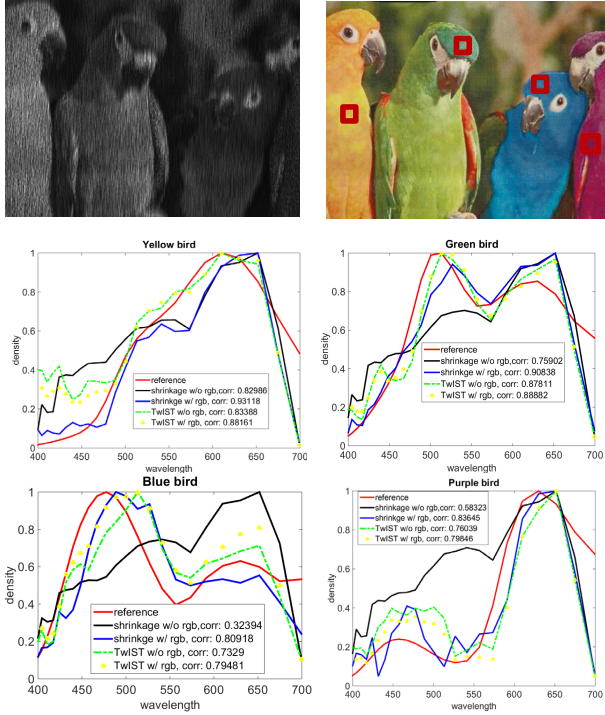

Fig. 14: Real data: measurement (top left) and reconstructed spectra of different birds with various algorithms compared to references.

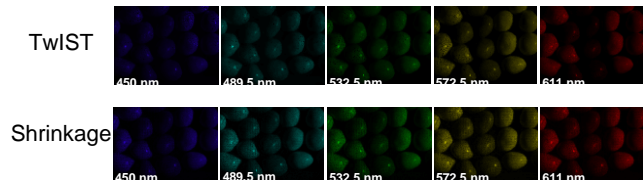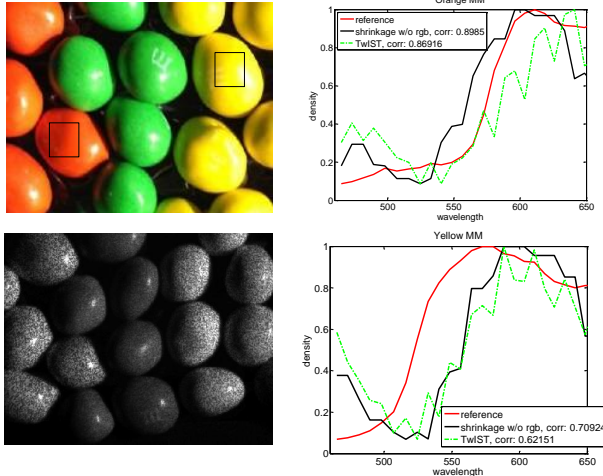Fig. 15: Real data: reconstructed spectra and images of M&M with TwIST and shrinkage without RGB image.

TABLE II: Inpainting results (PSNR in dB) of color images at various observed pixel percentages. We compare BPFA [35] (top row in each cell) with the proposed algorithm (bottom row in each cell). 8-bit RGB images are used.

data ratio	Castle	Mushroom	Train	Horses	Kangaroo
20%	29.12 **30.44**	31.56 **32.41**	24.59 **25.77**	29.99 **31.15**	29.59 **29.82**
30%	32.02 **33.65**	34.63 **35.23**	27.00 **28.51**	32.52 **33.78**	32.21 **32.29**
50%	36.45 **37.97**	38.88 **39.73**	32.00 **33.67**	37.27 **39.24**	37.34 **37.77**
80%	41.51 **44.10**	42.56 **44.79**	40.73 **43.88**	41.97 **45.29**	42.74 **49.39**

V. CONCLUSIONS

We have developed and tested a new Bayesian dictionary learning model for blind CS. Specifically, we have demonstrated high-quality inversion of compressed hyperspectral images captured by real cameras. Via the global-local shrinkage priors, our algorithm imposes compressibility on the dictionary

[8]The 30 wavelengths are 450nm, 458nm, 465nm, 473nm, 481.5nm, 489.5nm, 498nm, 507nm, 516nm, 524.5nm, 532.5nm, 540.5nm, 548.5nm, 556.5nm, 564.5nm, 572.5nm, 580.5nm, 588.5nm, 596nm, 603.5nm, 611nm, 618.5nm, 625.8nm, 633.5nm, 641nm, 648.5nm, 656nm, 663.5nm, 671nm, 678.5nm.
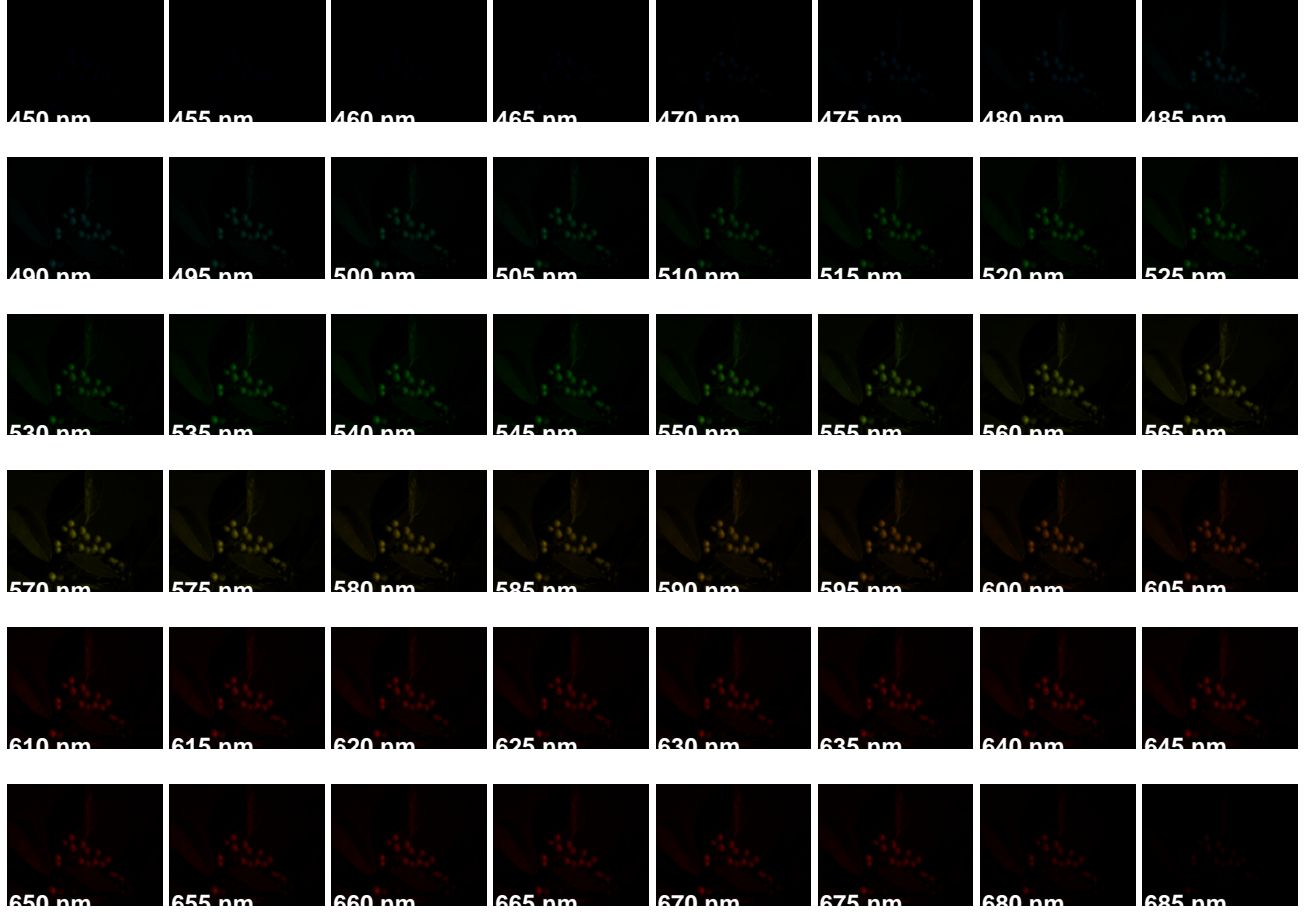
Fig. 16: Real data: reconstructed images of Berry data without RGB image.

TABLE III: Denoising results (PSNR in dB) of color images at various noise levels; σ denotes the noise standard deviations. We compare KSVD [36] (top row in each cell), with BPFA [35] (middle row in each cell) and the proposed algorithm (bottom row in each cell).

σ	Castle	Mushroom	Train	Horses	Kangaroo
	40.37	39.93	39.76	40.09	39.00
5	40.34	39.73	39.38	39.96	39.00
	41.24	**40.60**	**40.45**	**40.72**	**39.25**
	36.24	35.60	34.72	35.43	34.06
10	36.28	35.70	34.48	35.48	34.21
	37.11	**36.42**	**35.42**	**36.19**	**34.26**
	33.98	33.18	31.70	32.76	31.30
15	34.04	33.41	31.63	32.98	31.68
	34.51	**33.75**	**32.68**	**33.55**	**32.13**
	31.19	30.26	28.16	29.81	28.39
25	31.24	30.62	28.28	30.11	28.86
	31.74	**30.74**	**29.71**	**30.61**	**28.93**

coefficients to extract more information from the dictionary atoms. The reconstruction quality is improved significantly by integrating the compressed measurements with RGB images as side information under the blind CS framework. We also developed a new compressive hyperspectral imaging camera that uses an SLM to perform the spectral coding. Experimental results demonstrate the feasibility of the new camera, the superior performance of the algorithm, and the benefit of side information.

The original CASSI camera utilizes a separate disperser and encoder; the SLM-CASSI is capable of modulating both dimensions with one element. The SLM-CASSI also provides the potential of video rate compressive sensing and multi-frames adaptive sensing using the 60 Hz refresh rate of the SLM. SLM-CASSI offers greater flexibility of masking functions and an additional RGB camera for side information; however, it intertwines spectral and spatial multiplexing capabilities. The coded aperture/disperser architecture used in the original CASSI camera is more reliable, has a smaller form factor (i.e. fewer optical parts), and is less expensive, but lacks the coding flexibility and ability to use a separate camera for side information.

In this paper, we have shown that by using the RGB image as side information in our proposed blind CS framework, the reconstruction quality of hyperspectral images can be improved significantly. This framework can be generalized to other applications and algorithms. For instance, Gaussian mixture model based dictionary learning approaches [49]–[53] that learn a union of subspaces can also benefit from side information.

References

[1] K. Hege, D. O. Connell, W. Johnson, S. Basty, and E. Dereniak, "Hyperspectral imaging for astronomy and space surveillance," *Proc. SPIE*, vol. 5159, pp. 380–391, 2003.

[2] A. F. H. Goetz, G. Vane, J. E. Solomon, and B. N. Rock, "Imaging spectrometry for earth remote sensing," *Science*, vol. 228, pp. 1147–1153, 1985.

[3] B. S. Sorg, B. J. Moeller, O. Donovan, Y. Cao, and M. W. Dewhirst, "Hyperspectral imaging of hemoglobin saturation in tumor microvasculature and tumor hypoxia development," *J. Biomed. Opt.*, vol. 10, p. 044004, 2005.

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

[5] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, February 2006.

[6] M. E. Gehm, R. John, D. J. Brady, R. M. Willett, and T. J. Schulz, "Single-shot compressive spectral imaging with a dual-disperser architecture," *Optics Express*, vol. 15, pp. 14013–14027, 2007.

[7] A. Wagadarikar, R. John, R. Willett, and D. J. Brady, "Single disperser design for coded aperture snapshot spectral imaging," *Applied Optics*, vol. 47, no. 10, pp. B44–B51, 2008.

[8] A. Wagadarikar, N. Pitsianis, X. Sun, and D. Brady, "Video rate spectral imaging using a coded aperture snapshot spectral imager," *Optics Express*, vol. 17, no. 8, pp. 6368–6388, 2009.

[9] C. Li, T. Sun, K. F. Kelly, and Y. Zhang, "A compressive sensing and unmixing scheme for hyperspectral data processing," *IEEE Transactions on Image Processing*, vol. 21, no. 3, pp. 1200–1210, 2012.

[10] B. S. Sorg, B. J. Moeller, O. Donovan, Y. Cao, and M. W. Dewhirst, "Hyperspectral imaging of hemoglobin saturation in tumor microvasculature and tumor hypoxia development," *J. Biomed. Opt.*, vol. 10, p. 044004, 2005.

[11] E. Herrala, J. T. Okkonen, T. S. Hyvarinen, M. Aikio, and J. Lammasniemi, "Imaging spectrometer for process industry application," *Proc. SPIE*, vol. 2248, pp. 33–40, 1994.

[12] H. Morris, C. Hoyt, and P. Treado, "Imaging spectrometers for fluorescence and raman microscopy: acousto-optic and liquid crystal tunable filters," *Appl. Opt.*, vol. 34, pp. 4817–4826, 1995.

[13] J. Bioucas-Dias and M. 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] 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 of Selected Topics in Signal Processing, Tech. Rep., 2007.

[15] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing," *SIAM J. Imaging Sci*, pp. 143–168, 2008.

[16] X. Yuan, P. Llull, X. Liao, J. Yang, G. Sapiro, D. J. Brady, and L. Carin, "Low-cost compressive sensing for color video and depth," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[17] X. Yuan, J. Yang, X. Liao, P. Llull, G. Sapiro, D. J. Brady, and L. Carin, "Adaptive temporal compressive sensing for video," *IEEE International Conference on Image Processing*, pp. 1–4, 2013.

[18] 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.

[19] D. Kittle, K. Choi, A. Wagadarikar, and D. J. Brady, "Multiframe image estimation for coded aperture snapshot spectral imagers," *Applied Optics*, vol. 49, no. 36, pp. 6824–6833, December 2010.

[20] A. Rajwade, D. Kittle, T.-H. Tsai, D. Brady, and L. Carin, "Coded hyperspectral imaging and blind compressive sensing," *SIAM Journal on Image Science*, vol. 6, no. 2, pp. 782–812, 2013.

[21] S. Gleichman and Y. C. Eldar, "Blind compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, no. 10, pp. 6958–6975, 2011.

[22] J. M. Duarte-Carvajalino, G. Yu, L. Carin, and G. Sapiro, "Task-driven adaptive statistical compressive sensing of Gaussian mixture models," *IEEE Transactions on Signal Processing*, vol. 61, no. 3, pp. 585–600, February 2013.

[23] V. Cevher, "Learning with compressible priors," *Advances in Neural Information Processing Systems (NIPS)*, pp. 261–269, 2009.

[24] N. G. Polson and J. G. Scott, "Shrink globally, act locally: Sparse bayesian regularization and prediction," *Bayesian Statistics*, 2010.

[25] ——, "Local shrinkage rules, lévy processes and regularized regression," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 74, no. 2, pp. 287–311, 2012.

[26] X. Yuan, V. Rao, S. Han, and L. Carin, "Hierarchical infinite divisibility for multiscale shrinkage," *IEEE Transactions on Signal Processing*, vol. 62, no. 17, pp. 4363–4374, Sep. 1 2014.

[27] Y. Wu, I. O. Mirza, G. R. Arce, and D. W. Prather, "Development of a digital-micromirror-device-based multishot snapshot spectral imaging system," *Opt. Lett.*, vol. 36, pp. 2692–2694, 2011.

[28] G. Lazarev, A. Hermerschmidt, S. Kruger, and S. Osten, *Optical Imaging and Metrology:Advanced Technologies*. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim, Germeny, 2012.

[29] Y. August and A. Stern, "Compressive sensing spectrometry based on liquid crystal devices," *Optics Letters*, vol. 38, no. 23, pp. 4996–4999, 2013.

[30] M. E. Gehm and J. Kinast, "Adaptive spectroscopy: towards adaptive spectral imaging," *Proc. SPIE*, vol. 6978, p. 69780I, 2008.

[31] J. M. Duarte-Carvajalino and G. Sapiro, "Learning to sense sparse signals: Simultaneous sensing matrix and sparsifying dictionary optimization," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1395–1408, July 2009.

[32] S. D. A. Averbuch and S. Deutsch, "Adaptive compressed image sensing using dictionaries," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 57–89, 2012.

[33] W. Carson, M. Chen, M. Rodrigues, R. Calderbank, and L. Carin, "Communications inspired projection design with application to compressive sensing," *SIAM Journal on Imaging Sciences*, vol. 5, no. 4, pp. 1185–1212, 2012.

[34] R. Zhu, T. Tsai, and D. J. Brady, "Coded aperture snapshot spectral imager based on liquid crystal spatial light modulator," in *Frontiers in Optics*, 2013, pp. 1–2.

[35] M. Zhou, H. Chen, J. Paisley, L. Ren, L. Li, Z. Xing, D. Dunson, G. Sapiro, and L. Carin, "Nonparametric Bayesian dictionary learning for analysis of noisy and incomplete images," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 130–144, January 2012.

[36] J. Mairal, M. Elad, and G. Sapiro, "Sparse representation for color image restoration," *IEEE Transactions on Image Processing*, vol. 17, pp. 53–69, January 2008.

[37] H. Zou, "The adaptive lasso and its oracle properties," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1418–1429, Dec. 2006.

[38] Z. Xing, M. Zhou, A. Castrodad, G. Sapiro, and L. Carin, "Dictionary learning for noisy and incomplete hyperspectral images," *SIAM Journal on Imaging Sciences*, vol. 5, no. 1, pp. 33–56, Jan. 2012.

[39] A. Bhattacharya and D. B. Dunson, "Sparse bayesian infinite factor models," *Biometrika*, vol. 98, no. 2, pp. 291–306, 2011.

[40] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[41] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, pp. 3736–3745, December 2006.

[42] J. Paisley and L. Carin, "Nonparametric factor analysis with beta process priors," in *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009, pp. 777–784.

[43] A. Armagan, M. Clyde, and D. B. Dunson, "Generalized beta mixtures of gaussians," *Advances in Neural Information Processing Systems (NIPS)*, 2011.

[44] C. M. Carvalho, N. G. Polson, and J. G. Scott, "The horseshoe estimator for sparse signals," *Biometrika*, vol. 97, no. 2, pp. 465–480, 2010.

[45] P. Garrigues and B. A. Olshausen, "Group sparse coding with a laplacian scale mixture prior," *Advances in Neural Information Processing Systems (NIPS)*, pp. 676–684, 2010.

[46] J. E. Griffin and P. J. Brown, "Bayesian hyper-lassos with non-convex penalization," *Australian & New Zealand Journal of Statistics*, vol. 53, no. 4, pp. 423–442, 2011.

[47] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*, 2009, pp. 689–696.

[48] A. K. N. S. Foster, D.H. and M. Foster, "Frequency of metamerism in natural scenes," *Journal of the Optical Society of America A*, vol. 23, no. 12, pp. 2359–2372, 2006.

[49] M. Chen, J. Silva, J. Paisley, C. Wang, D. Dunson, and L. Carin, "Compressive sensing on manifolds using a nonparametric mixture of factor analyzers: Algorithm and performance bounds," *IEEE Transactions on Signal Processing*, vol. 58, no. 12, pp. 6140–6155, December 2010.

[50] J. Yang, X. Yuan, X. Liao, P. Llull, G. Sapiro, D. J. Brady, and L. Carin, "Video compressive sensing using Gaussian mixture models," *IEEE Transaction on Image Processing*, vol. 23, no. 11, pp. 4863–4878, November 2014.

[51] G. Yu and G. Sapiro, "Statistical compressed sensing of Gaussian mixture models," *IEEE Transactions on Signal Processing*, vol. 59, no. 12, pp. 5842–5858, 2011.

[52] J. Yang, X. Liao, X. Yuan, P. Llull, D. J. Brady, G. Sapiro, and L. Carin, "Compressive sensing by learning a Gaussian mixture model from measurements," *IEEE Transaction on Image Processing*, 2015.

[53] G. Yu, G. Sapiro, and S. Mallat, "Solving inverse problems with piecewise linear estimators: From Gaussian mixture models to structured sparsity," *IEEE Transactions on Image Processing*, 2012.