High-Quality Hyperspectral Reconstruction Using a Spectral Prior

INCHANG CHOI, DANIEL S. JEON, and GILJOO NAM, KAIST
DIEGO GUTIERREZ, Universidad de Zaragoza, I3A
MIN H. KIM, KAIST

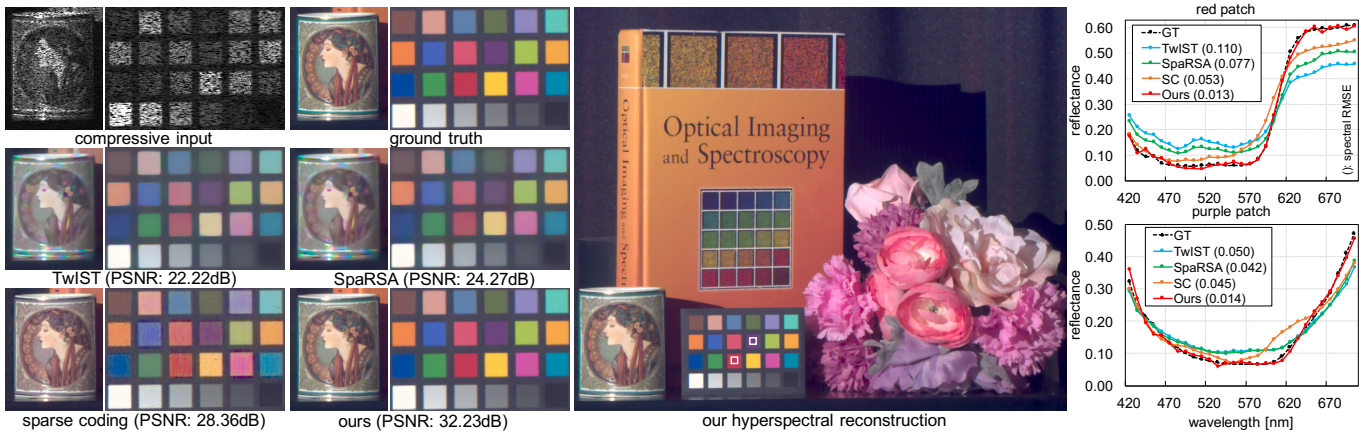


Fig. 1. Our novel hyperspectral reconstruction algorithm works with input from any existing compressive imaging architecture, and yields high-quality results, both in terms of spectral accuracy and spatial resolution. As the comparisons show, our results improve significantly over previous state-of-art methods. For instance, both TwIST and SpaRSA provide suboptimal spatial reconstruction in general, while sparse coding yields a noisy reconstruction of the color chart, and fails to accurately reconstruct the green border in the coffee mug. The charts on the right show how our reconstruction provides an excellent fit to the ground-truth data. In addition, we provide in this work a new high-resolution hyperspectral image dataset.

We present a novel hyperspectral image reconstruction algorithm, which overcomes the long-standing tradeoff between spectral accuracy and spatial resolution in existing compressive imaging approaches. Our method consists of two steps: First, we learn nonlinear spectral representations from real-world hyperspectral datasets; for this, we build a convolutional autoencoder, which allows reconstructing its own input through its encoder and decoder networks. Second, we introduce a novel optimization method, which jointly regularizes the fidelity of the learned nonlinear spectral representations and the sparsity of gradients in the spatial domain, by means of our new fidelity prior. Our technique can be applied to any existing compressive imaging architecture, and has been thoroughly tested both in simulation, and by building a prototype hyperspectral imaging system. It outperforms the state-of-the-art methods from each architecture, both in terms of spectral accuracy and spatial resolution, while its computational complexity is reduced by two orders of magnitude with respect to sparse coding techniques. Moreover, we present two additional applications of our method: hyperspectral interpolation and demosaicing. Last, we have created a new high-resolution hyperspectral dataset containing sharper images of more spectral variety than existing ones, available through our project website.

CCS Concepts: • **Computing methodologies → Hyperspectral imaging**;

Additional Key Words and Phrases: Hyperspectral imaging, image reconstruction

ACM Reference format:
Inchang Choi, Daniel S. Jeon, Giljoo Nam, Diego Gutierrez, and Min H. Kim. 2017. High-Quality Hyperspectral Reconstruction Using a Spectral Prior . *ACM Trans. Graph.* 36, 6, Article 218 (November 2017), 13 pages.
https://doi.org/10.1145/3130800.3130810

Authors' addresses: Inchang Choi, Daniel S. Jeon, and Giljoo Nam, KAIST, School of Computing, 291 Daehak-ro, Yuseong-gu, Daejeon, South Korea 34141; Diego Gutierrez, Universidad de Zaragoza, I3A, Maria de Luna 1, Zaragoza, Spain, 50018; Min H. Kim (corresponding author), KAIST, School of Computing, 291 Daehak-ro, Yuseong-gu, Daejeon, South Korea 34141; the corresponding author's email: minhkim@kaist.ac.kr.

0730-0301/2017/11-ART218 $15.00
https://doi.org/10.1145/3130800.3130810

1 INTRODUCTION

Different from conventional RGB cameras, hyperspectral images contain information from a much denser spectral sampling. This additional data can then be used in many applications, including appearance capture, environmental monitoring, scientific imaging, astronomy, etc [Attas et al. 2003; Gat 2000; Kim 2013; Kim et al. 2012a,b, 2014; Lin et al. 2014; Rapantzikos and Balas 2005] Earlier solutions focus on designing novel hardware architectures, including the use of liquid crystal bandpass filters, pushbroom scanners, micro-translation stages, or digital mirror devices, to name a few. These techniques share several limiting factors, such as the cost of engineering and building the hardware, or the need to capture static scenes only. Moreover, a common characteristic of these approaches is the tradeoff between spatial resolution and spectral accuracy in the captured results.

Several methods have been proposed to overcome this tradeoff. Optimization approaches define a data fidelity term and assume certain natural image priors, such as correlations in the spatial and the spectral domains. However, these hand-crafted priors are insufficient to represent the wide variety and nonlinear nature of real-world spectral data. Another recent trend relies on compressive imaging: the hyperspectral information is optically coded, either in the spatial or in the spatial-spectral domains, and the final signal is reconstructed from the captured coded information. This reconstruction represents a severely ill-posed problem (inferring dense spectral power distributions from a monochromatic, encoded image), for which many dictionary-based approaches exist. While sparse coding has provided good results in general, the reconstruction step imposes an extremely high computational cost. For instance, reconstruction of one hyperspectral image at VGA resolution in the recent spatial-spectral compressive technique by Lin et al. [2014] takes approximately 25 hours.

In this paper, we present a novel technique to address the reconstruction problem of hyperspectral images, consisting of two steps. First, we learn *nonlinear* spectral representations from real-world hyperspectral datasets, leveraging the encoding-decoding capabilities of a convolutional autoencoder. Second, we formulate a novel optimization problem that jointly regularizes the fidelity of nonlinear representations and the sparsity in spatial gradients. An important aspect in this step is our novel fidelity prior, relating the autoencoder with our optimization problem. The nonlinearity of our method provides a key advantage over previous sparse coding techniques, since it allows to reconstruct fine details with high spatial and spectral accuracy, outperforming state-of-the-art methods from the three existing compressive imaging architectures: SD-CASSI, SS-CASSI, and DD-CASSI. Moreover, our method can be applied to input from any of such architectures, and is two orders of magnitude faster than the state-of-the-art sparse coding approach.

We provide an in-depth analysis of all the factors and parameters of our reconstruction algorithm. We have additionally captured a new high-resolution hyperspectral image dataset, which fixes some limiting aspects of other existing datasets, where images suffer from low spatial resolution and are slightly out of focus, or present a limited spectral range. We make this new database publicly available at our project website[1]. together with our model and code. Moreover, we show results on a prototype hyperspectral camera, and present two additional applications of hyperspectral imaging.

2 RELATED WORK

There are many works that focus on developing hardware architectures to capture hyperspectral information. The most straightforward approach is temporal-spectral scanning, which isolates wavelength measurements using different bandpass or liquid crystal tunable filters (LCTF), while sequentially scanning the visible spectrum [Attas et al. 2003; Gat 2000; Lee and Kim 2014; Rapantzikos and Balas 2005]. The spectral resolution of this approach is limited by the number of filters used. Temporal multiple-sampling has been introduced using a micro-translation stage [Kim et al. 2012a; Kittle et al. 2010], or a digital micro-mirror device (DMD) [Wu et al. 2011]. Alternatively, spatial-spectral scanning approaches capture image columns for each wavelength through a slit, using for instance whiskbroom or pushbroom scanners [Brusco et al. 2006; Hoye and Fridman 2013; Porter and Enmark 1987]. Other recent approaches include kaleidoscope-based multiple sampling [Jeon et al. 2016], a reconfigurable camera [Manakov et al. 2013], or designs aimed to reduce hardware costs (e.g., [Alvarez et al. 2016; Baek et al. 2017; Cao et al. 2011; Habel et al. 2012]). In the rest of this section, we focus on *reconstruction* approaches from the captured data, which is the main goal of our paper.

Optimization Approaches. Optimization techniques aim to overcome the spatial-spectral tradeoff during reconstruction, usually defining a data fidelity term, and a total variation (TV) l_1-norm regularization term to emphasize sparsity of gradients. They rely on two main assumptions: First, hyperspectral components present a very high correlation in both the spatial and the spectral domains [Golbabaee et al. 2013; Zhang et al. 2011]. Second, hyperspectral vectors belong to a low-dimensional subspace [Li et al. 2012; Martin et al. 2015]. Zhang et al. [2011], and Golbabaee et al. [2013] assume that spectrally homogeneous segments exist in the spatial dimension, while Li et al. [2012] assume that spectral gradients are approximately piecewise smooth. Similarly, Martin et al. [2015] introduce a constrained optimization approach that infers spatial correlation. However, these approaches still exhibit artifacts in the reconstructed image structure and details. In our work, we replace hand-crafted priors with data-driven priors trained as neural networks. This reduces the ill-posedness of the problem, and allows us to introduce *nonlinear* representations of natural hyperspectral images.

Compressive Hyperspectral Imaging. Coded aperture snapshot spectral imaging (CASSI) is one of the most popular hyperspectral imaging approaches, allowing to capture dynamic scenes. The defining aspect of these compressive techniques is that the captured coded information needs to be reconstructed to yield the final image. CASSI can be divided into two classes, depending on how spectral signatures are encoded: (1) spatially-encoded CASSI, using a single disperser (SD-CASSI) [Kim et al. 2012a; Wagadarikar et al. 2008]; and (2) spatial-spectral CASSI, which codes information in both domains (SS-CASSI [Lin et al. 2014], or dual-disperser DD-CASSI [Gehm et al. 2007]). All these techniques share an intrinsic tradeoff between spatial resolution and spectral accuracy, so the reconstruction step defines the quality of the final image. Our novel spectral reconstruction algorithm can be applied to captured input from any compressive imaging technique, e.g., SD-CASSI, SS-CASSI, and DD-CASSI, providing better results with a significant speed-up factor over other existing data-driven approaches.

Dictionary-based Approaches. A few recent data-driven methods learn linear representations of natural spectral images as sparse coded dictionaries, using public hyperspectral image datasets (e.g., [Chakrabarti and Zickler 2011; Yasuma et al. 2010]). In this context, Peng et al. [2014] propose a denoising method during reconstruction. Wang et al. [2015] introduce a dual-camera system, combining information from panchromatic video at a high frame rate with hyperspectral information at a low frame rate; panchromatic information

[1]Project website: http://vclab.kaist.ac.kr/siggraphasia2017p1/

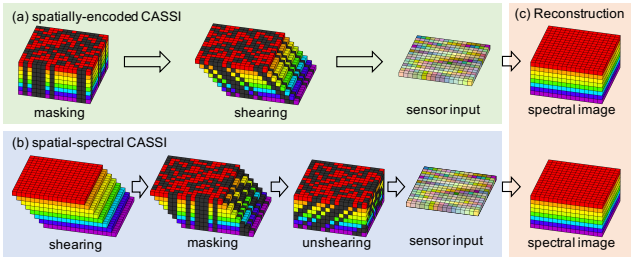


Fig. 2. Schematic diagram showing the spectral reconstruction process in (a) spatially-encoded CASSI, and (b) spatial-spectral CASSI. In spatially-encoded CASSI (SD-CASSI), a coded projection is created first, then dispersion creates a shear. In spatial-spectral CASSI (SS-CASSI and DD-CASSI), dispersion occurs first, then the sheared information is first coded, then unsheared with additional optics. A reconstruction step (c) yields the final spectral image in both cases.

is used to learn an overcomplete dictionary. Lin et al. [2014] introduce a spatial-spectral encoding hyperspectral imager, equipped with a diffraction grating. Dictionary-based techniques yield good results in general, although the required sparse reconstruction imposes a high computational cost (for instance, reconstructing hyperspectral images at less than VGA resolution takes about 25 hours [Lin et al. 2014]). Moreover, while sparse coding yields overcomplete dictionaries as *linear* representations of the scene, we make use of a *convolutional autoencoder*, which produces *nonlinear* representations. Coupled with our novel global optimization technique that jointly regularizes the fidelity of these nonlinear presentations, our reconstruction becomes more precise and efficient.

3 COMPRESSIVE HYPERSPECTRAL IMAGING

Background. The spectral signatures imprinted by a coded aperture are the fundamental building blocks in compressive hyperspectral imaging; from these, the image is reconstructed by means of optimization. There are two main ways to encode this spectral information: spatial encoding, and spatial-spectral encoding. Figure 2(a) depicts the former, used in SD-CASSI systems [Kim et al. 2012a; Kittle et al. 2010; Wagadarikar et al. 2008]; a coded projection of the spectrum is created first, then subsequently sheared by dispersion. The reconstruction step for SD-CASSI therefore reconstructs the image from sheared and coded information. On the other hand, Figure 2(b) shows spatial-spectral CASSI, used in SS-CASSI [Lin et al. 2014] and dual-disperser DD-CASSI [Gehm et al. 2007]. This approach disperses incident rays first, then the mask creates a coded projection; additional optics then unshear this information. As a result, the spectral reconstruction for SS-CASSI and DD-CASSI requires a simpler optimization than SD-CASSI, resulting in superior results, at the cost of a more complex optical setup.

In this work, we focus on hyperspectral image reconstruction from compressive input; as such, our method is agnostic to the particular encoding of the input spectral data. Considering the computational advantages, we use spatial-spectral encoding as our first choice to test our reconstruction algorithm (implementation details are given in Sections 4 and 5).

Image Formation. Let $h(x, y, \lambda)$ indicate the spectral intensity of light with wavelength λ at location (x, y). A mask creates coded patterns given by its transmission function $T(x, y)$, while dispersion creates a shear along the horizontal axis, according to a dispersion function $\phi(\lambda)$. In spatial encoding, e.g., SD-CASSI, the projected light intensity on the sensor $i(x, y)$ can be represented as an integral over all visible wavelengths Λ as:

$$i(x, y) = \int_{\Lambda} T(x + \phi(\lambda), y)\, h(x + \phi(\lambda), y, \lambda)\, d\lambda. \tag{1}$$

In contrast, in spatial-spectral encoding, e.g., DD-CASSI, the sheared spectrum $h(x + \phi(\lambda), y, \lambda)$ is modulated by the coded mask $T(x, y)$, and the result unsheared by $\phi(\lambda)$ in the opposite direction, resulting in:

$$i(x, y) = \int_{\Lambda} T(x - \phi(\lambda), y)\, h(x, y, \lambda)\, d\lambda. \tag{2}$$

Note that the sign of the horizontal dispersion function $\phi(\lambda)$ is reversed. In matrix-vector form, a hyperspectral image with C channels can be expressed as $\mathbf{h} \in \mathbb{R}^n$, where $n = H \times W \times C$, and H and W are the spatial dimensions of the image. Transmissivity can be expressed by means of a sparse modulation matrix $\mathbf{\Phi} \in \mathbb{R}^{m \times n}$, where $m = H \times W$ is the number of pixels in the sensor. This matrix is made up of $\mathbf{\Phi}_C \in \mathbb{R}^{m \times m}$ submatrices for each wavelength. The product of $\mathbf{\Phi}$ and $\mathbf{h}$ yields the captured image $\mathbf{i} \in \mathbb{R}^m$:

$$\mathbf{i} = \mathbf{\Phi h}. \tag{3}$$

This equation describes a highly under-determined system, since $m \ll n$. In the next section, we describe our novel reconstruction algorithm to restore the hyperspectral image $\mathbf{h}$. A key aspect is using a *nonlinear* operator learned through a convolutional autoencoder, instead of the common sparse coding approach of using linear combinations of overcomplete dictionaries.

General Compressive Sensing vs. Compressive Hyperspectral Imaging. Compressive hyperspectral imaging (HSI) could be considered within a general compressive sensing (CS) approach. However, compressive HSI has some particular characteristics that require a more specialized solution to obtain high-quality results like ours: CS reconstructs spatial image structures in 2D patches; color information is reconstructed implicitly by combining reconstructions from the three color channels, computed separately. On the contrary, HSI reconstructs spectral images as 3D tensors, with stronger compression along the spectral dimension, resulting in higher complexity; color cannot be reconstructed as in general CS, since it appears overlapped due to dispersion. In our HSI approach, a monochromatic sensor captures 31 spectral channels (measurement rate of just 3% of the spectral information, again with the additional problem of blur due to dispersion). It is dispersion that combines the spectral and the spatial domains (clearly visible in the captured coded information, top-left of Figure 1), leading to the common tradeoff between spatial and spectral resolution in HSI.

4 HYPERSPECTRAL IMAGE RECONSTRUCTION

Figure 3 shows an overview of our two-step process to reconstruct hyperspectral images from encoded sensor signals. First, we train a convolutional autoencoder to learn nonlinear representations of real hyperspectral image tensors. This nonlinearity is a key aspect of our reconstruction, since it will allow us to cover a wider range of real-world spectral features. Second, we reconstruct hyperspectral

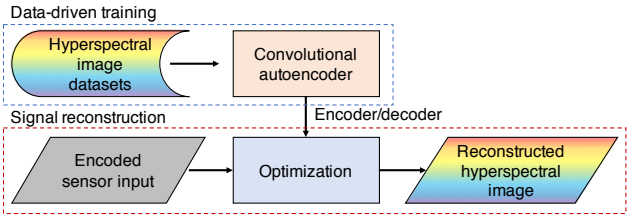


Fig. 3. Overview of our hyperspectral image reconstruction. Our convolutional autoencoder is first trained to learn nonlinear representations of hyperspectral image tensors. The final hyperspectral images are then reconstructed from the encoded sensor input by minimizing our objective function, which includes a novel prior to regularize the fidelity of nonlinear representations (refer to the text for details).

images from the encoded input by globally solving a nonlinear optimization problem. As an important aspect in our formulation, we introduce a novel prior term that enforces the data-driven autoencoder representations of the real world spectrum into reconstructed signals. Our objective function jointly regularizes this term and the sparsity of gradients to yield the final hyperspectral image.

4.1 Convolutional Autoencoder

Convolutional neural networks have been recently used in many tasks, such as spectral image classification [Li et al. 2017; Maggiori et al. 2016], or to extract features from images [Long et al. 2015; Taigman et al. 2014]. However, they are not intended to reconstruct the original signals from extracted features, which is precisely our goal. Autoencoders, on the other hand, are unsupervised neural networks where the output and input layers share the same number of nodes, and which can reconstruct its own inputs through encoder and decoder functions [Hinton and Salakhutdinov 2006]. They have been used for spectral image classification [Chen et al. 2014; Lin et al. 2013; Ma et al. 2016; Xing et al. 2016; Zabalza et al. 2016] or denoising [Vincent et al. 2010]. Masci et al. [2011] proposed the *convolutional* autoencoder, where both convolution operations and activation functions operate on each layer. It has been successfully applied for object retrieval [Leng et al. 2015], image classification [Zhang et al. 2016], denoising [Du et al. 2016], or real-time correction of multipath interference in time-of-flight imaging [Marco et al. 2017]. In this work, we leverage a convolutional autoencoder to first train an encoder network to learn a representation of hyperspectral images in a nonlinear space, then use the decoder network to reconstruct the final image from coded sensor data.

Similar to previous works [Li et al. 2012; Martin et al. 2015], we assume that hyperspectral vectors belong to a subspace of hidden representations. However, instead of using predetermined bases (such as discrete cosine transforms or wavelets) or dictionary-based sparse coding [Lin et al. 2014; Peng et al. 2014; Wang et al. 2015], we rely on the convolutional autoencoder to decompose input signals into a set of basis vectors and coefficients. Moreover, while common sparse coding approaches usually reconstruct signals by linear combination of the basis functions, the autoencoder allows for nonlinear reconstruction of hyperspectral information, which fits better the nonlinear nature of the problem, and thus leads to better results (see Section 5).

Our convolutional autoencoder consists of two subnetworks: an encoder network that transforms input training datasets into their nonlinear representation (green block in Figure 4), and a decoder network that generates the original datasets from these representations (red block). Formally, the convolutional autoencoder $\boldsymbol{A}()$ is thus a composition of two nonlinear functions: the encoder function $\boldsymbol{E}()$, and the decoder function $\boldsymbol{D}()$. After training the network, we can convert a hyperspectral image $\mathbf{h}$ into a nonlinear representation $\boldsymbol{\alpha}$ by using the encoder function $\boldsymbol{\alpha}=\boldsymbol{E}(\mathbf{h})$. We can then reconstruct the hyperspectral image $\mathbf{h}$ from $\boldsymbol{\alpha}$ using the decoder function $\mathbf{h}\approx\boldsymbol{D}(\boldsymbol{\alpha})$, which therefore acts as a hyperspectral image prior. This can be described as:

$$\boldsymbol{A}(\mathbf{h}) = \boldsymbol{D}(\boldsymbol{E}(\mathbf{h})) \approx \mathbf{h}. \tag{4}$$

Later, our signal reconstruction process searches for the nonlinear hyperspectral representation that satisfies our image formation model.

Network Architecture. As shown in Figure 4, our autoencoder consists of $(2\times d+1)$ layers, excluding the input and the output layers, where d is the number of hidden layers of each subnetwork. The encoder network $\boldsymbol{E}(\mathbf{h})$ is placed at the beginning of the autoencoder. Suppose an input of $H\times W\times C$ is fed into the encoder network. Then, the encoder outputs a nonlinear representation of the hyperspectral image, defined as:

$$\boldsymbol{E}(\mathbf{h}) = \mathbf{W}_E^{d+1} * \mathbf{F}_E^d + \mathbf{b}_E^{d+1}, \tag{5}$$

$$\mathbf{F}_E^l = \sigma(\mathbf{W}_E^l * \mathbf{F}_E^{l-1} + \mathbf{b}_E^l) \quad \text{for} \quad l \in \{1 \dots d\}, \tag{6}$$

where $\mathbf{W}_E^l$, $\mathbf{F}_E^l$, and $\mathbf{b}_E^l$ are the kernel weight, the intermediate feature representation, and the bias in layer l of the encoder network, respectively. The weights and the biases form an autoencoder. The subscript E refers to the encoder. We set $\mathbf{F}_E^0$ as the input hyperspectral image $\mathbf{h}$. In Equation (6), σ is a nonlinear activation function, so-called a rectified linear unit (ReLU), which is $\sigma(\cdot)=\max(0,\cdot)$. Note that in order not to impose any constraints on $\boldsymbol{\alpha}$ in the later reconstruction step, this activation function is not applied to the output layer in Equation (5). Refer to Section 4.3 for more details.

Similar to the architecture of the encoder network, the decoder network with d hidden layers is defined as:

$$\boldsymbol{D}(\boldsymbol{\alpha}) = \sigma(\mathbf{W}_D^{d+1} * \mathbf{F}_D^d + \mathbf{b}_D^{d+1}), \tag{7}$$

$$\mathbf{F}_D^l = \sigma(\mathbf{W}_D^l * \mathbf{F}_D^{l-1} + \mathbf{b}_D^l) \quad \text{for} \quad l \in \{1 \dots d\}, \tag{8}$$

where $\mathbf{F}_D^0$ is the nonlinear representation $\boldsymbol{\alpha}$ of the hyperspectral image. For the first convolutional layer of the encoder network, $\mathbf{F}_E^1$, we use $3\times3\times C$, but the other layers are convolved with kernels of $3\times3\times R$. The spatial resolution of the image and hidden layers remains the same. Note that our goal is to reconstruct the original signals, rather than to extract feature vectors like most existing applications of convolutional autoencoders. In that sense, our feature vectors can be seen as low-dimensional subspaces defining hyperspectral vectors, analogous to overcomplete dictionaries in sparse coding. We found that the number of feature vectors R has a significant impact on the accuracy of the reconstructed signal, with more feature vectors yielding better results, as Figure 5(a) shows. However, there is a practical tradeoff between performance and memory; in practice, we fix the number of feature vectors in

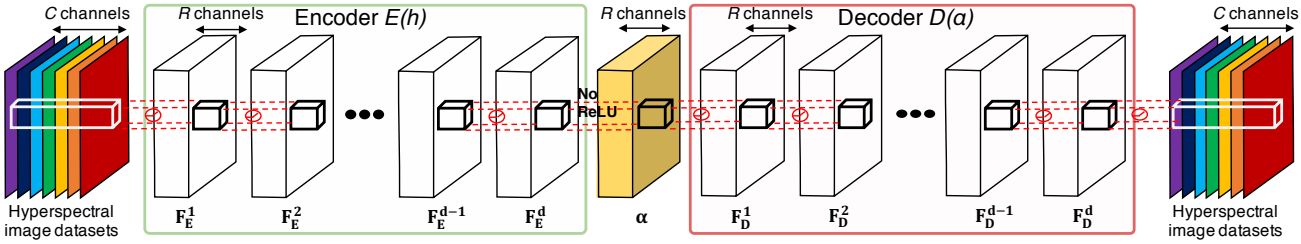


Fig. 4. Schematic diagram of our convolutional autoencoder. Our network learns from real hyperspectral image datasets with C channels, and approximates the same hyperspectral datasets as output. The model consists of an encoder network (green box) and a decoder network (red box). White tensors in hidden layers represent intermediate feature representations, and the yellow tensor in the middle represents the nonlinear representations of hyperspectral images $\boldsymbol{\alpha}$ (with R channels, $R \gg C$). Convolution operations and activation functions are preceded for each layer. The spatial resolution of the image and hidden layers remains the same.

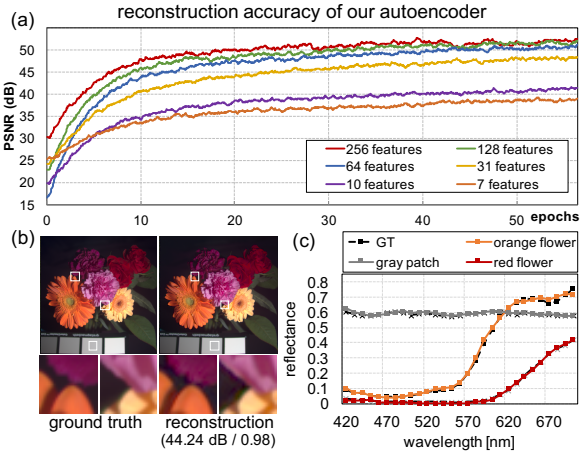


Fig. 5. Reconstruction accuracy of our autoencoder. (a) Impact of the number of feature vectors R when reconstructing the original signals. The original signal has C=31 channels: When $R \ll C$, (R=7 and R=10 in the figure), the reconstruction quality is poor, improving significantly when $R \approx C$. For $R \gg C$, accuracy increases only slightly. In our paper, we choose R= 64. (b) Comparison between the input ground truth and the reconstructed image. The average PSNR and SSIM values of the 31 channels from 400 nm to 700 nm are 44.24 dB and 0.98, respectively. (c) Spectral comparison between the ground truth and the reconstruction.

each layer to R=64 (larger than the original C=31). Figures 5(b) and (c) show an example of a hyperspectral reconstruction with R=64 feature vectors and 11 hidden layers. The reconstructed spectral information is virtually identical to the original.

Training Procedure. Our definition of the autoencoder includes a set of parameters $\boldsymbol{\theta}=\{(\mathbf{W}_E^l, \mathbf{b}_E^l), (\mathbf{W}_D^l, \mathbf{b}_D^l)\}_{l=1}^{d+1}$. To learn nonlinear representations of hyperspectral images, we train the autoencoder network and find the particular set $\boldsymbol{\theta}$ that minimizes a loss function. Given a set of k hyperspectral images $\mathbf{H}=\{\mathbf{h}^{(i)}\}_{i=1}^{k}$, our loss function $J(\mathbf{H}, \boldsymbol{\theta})$ including a decay term to avoid overfitting is:

$$\frac{1}{2k}\sum_{i=1}^{k}\left\|\boldsymbol{A}(\mathbf{h}^{(i)}) - \mathbf{h}^{(i)}\right\|^2 + \frac{\tau_w}{2}\sum_{l=1}^{d+1}\left(\|W_E^l\|^2 + \|W_D^l\|^2\right), \quad (9)$$

where $\boldsymbol{A}(\mathbf{h}^{(i)})$ can be alternatively expressed as $\boldsymbol{D}(\boldsymbol{E}(\mathbf{h}^{(i)}))$, and τ_w balances the relative importance between data fidelity and the regularization to avoid overfitting. Following Glorot and Bengio [2010], we initialize both $\mathbf{W}_E^l$ and $\mathbf{W}_D^l$ using the normalized initialization in order to maintain variances of back-propagated gradients and activation.

Implementation Details. We created augmented training datasets using 109 hyperspectral images obtained from the publicly available Harvard [Chakrabarti and Zickler 2011] and Columbia [Yasuma et al. 2010] datasets (77 images from the former, and 32 from the latter). Each hyperspectral image includes approximately 31 wavelength channels. We additionally augmented this initial image dataset following existing network training approaches [Simonyan and Zisserman 2015]. In order to achieve scale invariance for input images, we scaled the input dataset to two additional resolutions (half and double); this results in 327 hyperspectral images. We sampled 21,760 tensor patches of size 96×96×31 from this augmented dataset. We employ TensorFlow [Abadi et al. 2016] to implement our autoencoder, minimize the loss function in Equation (9) using the ADAM gradient descent method [Kingma and Ba 2014], and train it up to 60 epochs. The batch size is set to 64 with a learning rate of 10^{-4} for gradient descent; the weight τ_w for the decay term is set to 10^{-8}. With R = 64 feature channels and 11 hidden layers, it took approximately 30 hours to training the network, using a machine equipped with an i7-6770k CPU with 64GB of memory and an NVIDIA Titan X Pascal GPU with 12GB of memory.

4.2 Reconstruction via Optimization

As we have seen, we represent a hyperspectral image as $\mathbf{h} \approx \boldsymbol{D}(\boldsymbol{\alpha})$, with $\boldsymbol{\alpha} \in \mathbb{R}^q$, and $q=H\times W\times R$. Thus, the compressive image formation defined in Equation (3) can be re-written as:

$$\mathbf{i} = \boldsymbol{\Phi}\mathbf{h} \approx \boldsymbol{\Phi}\boldsymbol{D}(\boldsymbol{\alpha}). \quad (10)$$

Note that although this equation is similar to the linear combination of overcomplete dictionaries in sparse coding [Lin et al. 2014], our decoder $\boldsymbol{D}()$ is now a nonlinear operator.

Since $m \ll n$ in $\boldsymbol{\Phi} \in \mathbb{R}^{m\times n}$, Equation (10) defines an under-determined system. This makes the inverse problem of reconstructing a hyperspectral image $\mathbf{h}$ from an observation $\mathbf{i}$ severely ill-posed. We formulate our hyperspectral reconstruction by means of an objective

function as:

$$\min_{\boldsymbol{\alpha}} \underbrace{\|\mathbf{i} - \boldsymbol{\Phi} \boldsymbol{D}(\boldsymbol{\alpha})\|_2^2}_{\text{data terms}} + \underbrace{\tau_1 \|\boldsymbol{\alpha} - \boldsymbol{E}(\boldsymbol{D}(\boldsymbol{\alpha}))\|_2^2 + \tau_2 \|\boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha})\|_1}_{\text{prior terms}}, \quad (11)$$

where $\boldsymbol{E}: \mathbb{R}^n \rightarrow \mathbb{R}^q$ is the encoder introduced in Section 4.1, $\boldsymbol{\nabla}_{xy}$ denotes the spatial gradient operator, and τ_1 and τ_2 weigh the relative importance between the data fidelity and the prior terms. Our first prior term regularizes the fidelity of nonlinear representations using the encoder-decoder pair, while the second prior is a total variation (TV) l_1-norm regularizer, favoring sparsity of gradients in the spatial domain. The first prior term of $\boldsymbol{\alpha}$-fidelity is the key contribution in our objective function, since it allows us to relate autoencoder representations with our optimization problem. This has a large impact on the spectral accuracy of the reconstructed images, as Figure 9 shows.

Optimization. Since the gradient sparsity term of TV is not differentiable, we first split our objective function in Equation (11) into two problems:

$$\boldsymbol{f}(\boldsymbol{\alpha}) = \|\mathbf{i} - \boldsymbol{\Phi} \boldsymbol{D}(\boldsymbol{\alpha})\|_2^2 + \tau_1 \|\boldsymbol{\alpha} - \boldsymbol{E}(\boldsymbol{D}(\boldsymbol{\alpha}))\|_2^2, \quad (12)$$

$$\boldsymbol{g}(\mathbf{z}) = \tau_2 \|\mathbf{z}\|_1, \quad (13)$$

so that our optimization problem can be re-formulated as:

$$\min_{\boldsymbol{\alpha}} \boldsymbol{f}(\boldsymbol{\alpha}) + \boldsymbol{g}(\mathbf{z}) \quad \text{subject to } \boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha}) - \mathbf{z} = 0, \quad (14)$$

where $\mathbf{z}$ represents the spatial gradients of the reconstructed hyperspectral images. We iteratively solve this problem using the alternating direction method of multipliers (ADMM), as shown in Algorithm 1. We summarize here the key aspects of this solution, and refer the interested reader to other recent works for a more detailed explanation of this technique (e.g., [Afonso et al. 2011]). First, the l_2 terms are updated in Line 3, minimized by the ADAM optimizer [Kingma and Ba 2014]. We then minimize the l_1 term with an auxiliary variable $\mathbf{z}$ in Line 4, using proximal gradient descent; we update this term using an element-wise soft-thresholding function $S_{\tau_2/\rho}$, shown in Line 5. Parameters τ_2 and ρ in Algorithm 1 control the strength of the sparsity of gradients $\boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha})$ constraint as:

$$S_{\tau_2/\rho}(\mathbf{v}_i) = \begin{cases} \mathbf{v}_i - \tau_2/\rho, & \text{if } \mathbf{v}_i > \tau_2/\rho, \\ 0, & \text{if } |\mathbf{v}_i| \leq \tau_2/\rho, \\ \mathbf{v}_i + \tau_2/\rho, & \text{if } \mathbf{v}_i < -\tau_2/\rho. \end{cases} \quad (15)$$

The Lagrangian multipliers $\mathbf{u}$ are then updated in Line 6, via gradient ascent, to satisfy the constraint in Equation (14). This process is repeated until we reach the stopping criterium. Once we have obtained the solution representations $\boldsymbol{\alpha}_{\text{opt}}$, we recover the final hyperspectral image using the decoder as $\boldsymbol{D}(\boldsymbol{\alpha}_{\text{opt}})$.

ALGORITHM 1: ADMM solution of Equation (14)

1: **initialization**
2: **repeat**
3: $\quad \boldsymbol{\alpha}^{(k+1)} = \arg\min_{\boldsymbol{\alpha}} \left(\boldsymbol{f}(\boldsymbol{\alpha}) + \frac{\rho}{2} \left\| \boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha}) - \mathbf{z}^{(k)} + \mathbf{u}^{(k)} \right\|_2^2 \right)$
4: $\quad \mathbf{z}^{(k+1)} = \arg\min_{\mathbf{z}} \left(\boldsymbol{g}(\mathbf{z}) + \frac{\rho}{2} \left\| \boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha}^{(k+1)}) - \mathbf{z} + \mathbf{u}^{(k)} \right\|_2^2 \right)$
5: $\quad\quad = S_{\tau_2/\rho} \left(\boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha}^{(k+1)}) + \mathbf{u}^{(k)} \right)$
6: $\quad \mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \boldsymbol{\nabla}_{xy} \boldsymbol{D}(\boldsymbol{\alpha}^{(k+1)}) - \mathbf{z}^{(k+1)}$
7: **until** *the stopping criterion is satisfied.*

4.3 Discussion

Parameters. In Equation (11), we set τ_1 for the nonlinear representational fidelity to 0.1, while τ_2 and ρ in Algorithm 1 are set to 10^{-3} and 10^{-1} , respectively. Our optimizer performs approximately 20 ADMM iterations. The ADAM optimizer for $\boldsymbol{f}(\boldsymbol{\alpha})$ in Equation (14) iterates 200 steps with a learning rate of 5×10^{-2}.

Time Complexity. The time complexity of our hyperspectral image reconstruction is proportional to the number of multiplications performed in our convolutional autoencoder. When performing one-stride convolutions, the number of multiplications for a convolutional layer is $(H \times W) \times (w \times w \times R_i) \times R_o$, where w is the kernel size, and R_i and R_o are the number of feature maps for the input and output of the convolution. In our convolutional autoencoder (64 features with eleven hidden layers) with 3×3 kernels, the total number of multiplications is approximately $4.4 \times H \times W \times 10^5$. Compared to the current state-of-the-art data-driven approach [Lin et al. 2014], the sparse coding method can be considered as a shallow convolutional neural network without hidden layers nor activation functions. Using a dictionary with 6200 atoms of 10×10×31 hyperspectral image patches, the estimated number of multiplications is $(H \times W) \times (10 \times 10 \times 31) \times 6200 \approx 1.9 \times H \times W \times 10^7$, which is two orders of magnitude more.

Activation in the Encoder. As described in Section 4.1 and in Figure 4, a ReLU activation function is absent in the output layer of the encoder. This indicates that our nonlinear representation $\boldsymbol{\alpha}$ of hyperspectral images is not constrained to be sparse. Although we do not explicitly impose sparsity, the autoencoder makes the representation sparse while $\boldsymbol{\alpha}$ passes through other layers with a ReLU activation function. Another advantage of this absence is that it simplifies our nonlinear optimization. Adding a ReLU activation function in the output layer of the encoder would require an extra non-negative constraint satisfaction term in Equation (11); moreover, two more variables, an auxiliary variable and a Lagrangian variable, would have to be introduced in our ADMM formulation in Equation (14). As a result, convergence would be slower.

Global vs. Local Optimization. Global optimization approaches, such as TwIST and SpaRSA, are more effective in reconstructing spectral information, while local optimization techniques such as sparse coding operate on each patch independently, preserving image structures well. However, the amount of dispersion is limited by the patch size, which strongly affects computational costs. Our approach combines the benefits of both local and global optimization via the convolutional autoencoder and the total variation terms.

5 RESULTS

To evaluate the performance of our reconstruction algorithm, we have first created a test set of encoded input images from the existing Harvard [Chakrabarti and Zickler 2011] and Columbia [Yasuma et al.

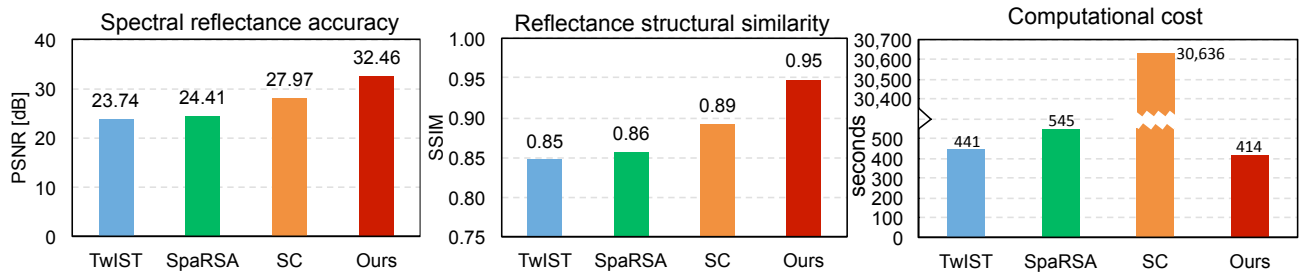
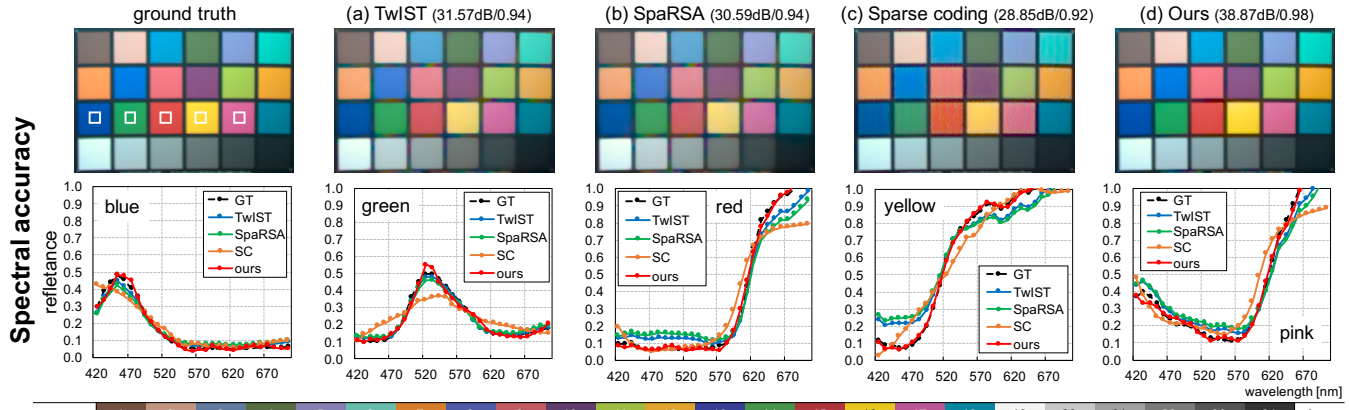


Fig. 6. Comparison of spectral reflectance accuracy (left), spatial reflectance accuracy (middle), and computational times (right) of four different methods: TwIST, SpaRSA, sparse coding, and ours (average of 32 spectral images from the Columbia spectral image dataset).



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	Avg.
TwIST	0.003	0.032	0.012	0.004	0.007	0.006	0.035	0.015	0.048	0.011	0.023	0.034	0.012	0.007	0.035	0.049	0.035	0.006	0.005	0.003	0.002	0.002	0.003	0.001	**0.016**
SpaRSA	0.005	0.041	0.019	0.007	0.011	0.013	0.048	0.025	0.062	0.019	0.032	0.045	0.019	0.012	0.051	0.060	0.048	0.012	0.005	0.004	0.003	0.002	0.002	0.001	**0.023**
SC	0.011	0.071	0.025	0.027	0.043	0.044	0.063	0.046	0.058	0.067	0.061	0.063	0.027	0.038	0.081	0.074	0.089	0.022	0.029	0.023	0.016	0.009	0.004	0.002	**0.041**
Ours	0.006	0.019	0.009	0.014	0.011	0.009	0.011	0.012	0.008	0.008	0.015	0.014	0.008	0.005	0.008	0.010	0.017	0.007	0.009	0.009	0.008	0.004	0.002	0.001	**0.009**

Fig. 7. Comparison of the spectral accuracy of our reconstruction against three state-of-the-art methods: (a) TwIST, (b) SpaRSA, (c) sparse coding (SC), and (d) our method. The numbers in the parenthesis on top of the pictures show pixel-wise spectral differences between the result and the ground-truth (PSNR and SSIM). The middle row shows reconstructed reflectances for five colors: blue, green, red, yellow, and pink. The bottom row shows RMSEs of the reconstructed spectra for the 24 patches. Our algorithm outperforms all the other methods on average, and on most of the individual color patches. Moreover, our spatial reconstruction is free from spatial artifacts visible in the other methods. The spatial accuracy of the reconstruction is further analyzed in Figure 8.

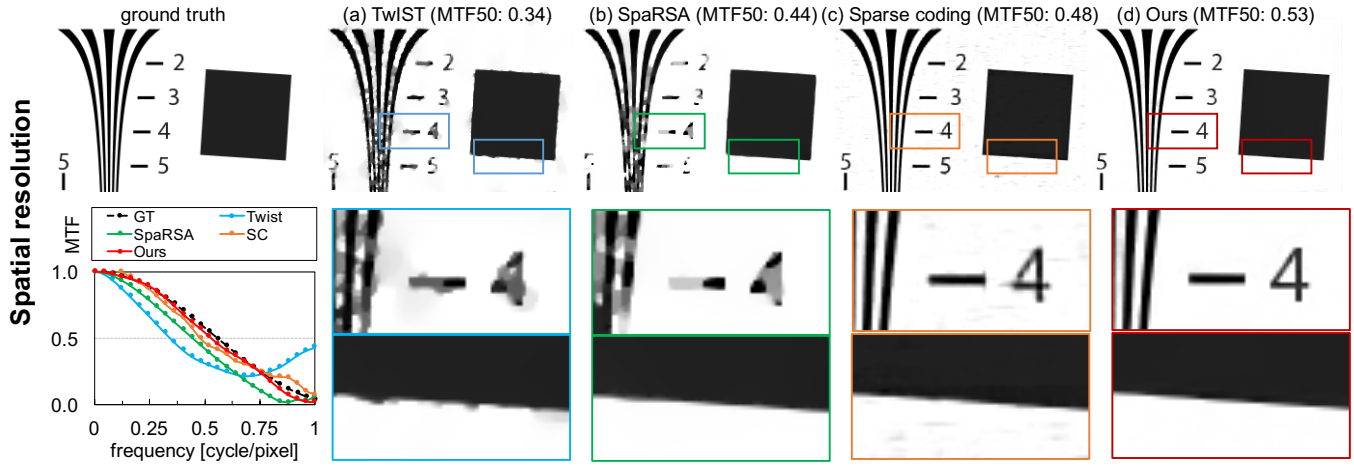


Fig. 8. Comparison of the spatial accuracy of our reconstruction against three state-of-the-art methods: (a) TwIST, (b) SpaRSA, (c) sparse coding (SC), and (d) our method. We calculate modulation transfer functions (MTF) for each method using the bottom region of the square for a wavelength of 550 nm. The numbers in the parenthesis of each method are MTF50 values. The spatial resolutions of TwIST (a) and SpaRSA (b) are clearly suboptimal, like other traditional optimization-based algorithms. Our method outperforms all three state-of-the-art approaches, including the sparse coding method.

2010] spectral image datasets[2]. In addition, we have created a *new dataset* (see Figure 12 and supplementals) by simulating the imaging process with the three main types of encoding architectures: SD-CASSI, DD-CASSI and SS-CASSI, as described in Section 3 (note that the modulation matrix Φ in Equation (11) changes depending on the image formation model[3]).

We compare our results against three other state-of-the-art methods, representing the three different encoding architectures: TwIST [Bioucas-Dias and Figueiredo 2007], SpaRSA [Wright et al. 2009], and sparse coding [Lin et al. 2014]. We choose the best imaging architecture for each method to produce these results: DD-CASSI for TwIST and SpaRSA, and SS-CASSI for sparse coding and ours. As we show in this section, our reconstructions show a significant improvement in both spectral and spatial accuracy. Moreover, our method is the fastest of the three. Figure 6 shows average results over the Columbia image dataset. Additionally, we provide an analysis of the parameter space, compare our method against ground truth and a straightforward learning-based reconstruction [Kulkarni et al. 2016], introduce our new hyperspectral dataset, and present results with a real hyperspectral imaging system. Last, we propose two applications of our method, without any hardware modifications: from multi- to hyperspectral interpolation, and hyperspectral demosaicing. Refer to supplemental materials and our project website for more results and materials not included in the rest of the section.

5.1 Spectral Accuracy vs. Spatial Resolution

Existing reconstruction techniques share an intrinsic tradeoff between spectral accuracy and spatial resolution, which defines the quality of the final image. As shown in Figure 1, traditional optimization approaches such as TwIST and SpaRSA yield good results in spectral accuracy, but at the cost of suboptimal spatial resolution. On the other hand, the data-driven approach based on sparse coding offers good spatial resolution, but sacrificing spectral accuracy. In contrast, our method yields high-quality results in both domains. We compare here the performance of our reconstruction algorithm in terms of spectral accuracy and spatial resolution, respectively.

Spectral Accuracy. We evaluate the spectral accuracy of our reconstructed spectral images by calculating peak signal-to-noise ratios (PSNR), and structural similarity (SSIM). Figure 7 shows side-by-side comparisons for the ColorChecker hyperspectral image from the Columbia dataset. We converted the results of spectral images to sRGB via the revised 2-degree CIE color matching functions [Vos 1978] for visualization. The averaged PSNR and SSIM of our result (38.87dB/0.98) across the 31 wavelength channels outperforms all the reconstructions of TwIST (31.57dB/0.94), SpaRSA (30.59dB/0.94) and sparse coding (28.85db/0.92). In addition, we evaluate the reconstructed spectral reflectances of five primary colors in the chart: blue, green, red, yellow, and pink; our results are consistently closer to the ground truth than the rest of the methods. The table at the bottom shows root-mean-squared errors (RMSEs) for each color patch, as well as the average.

[2]We use visible spectral wavelengths between 420 nm and 720 nm to avoid the inaccuracy between 400 nm and 410 nm in common LCTF measurements [Imai et al. 2002].
[3]For SD-CASSI and DD-CASSI, the amount of dispersion is set to one pixel per 10 nm. For SS-CASSI, we set the spectrum shift ratio to 0.1 following [Lin et al. 2014].

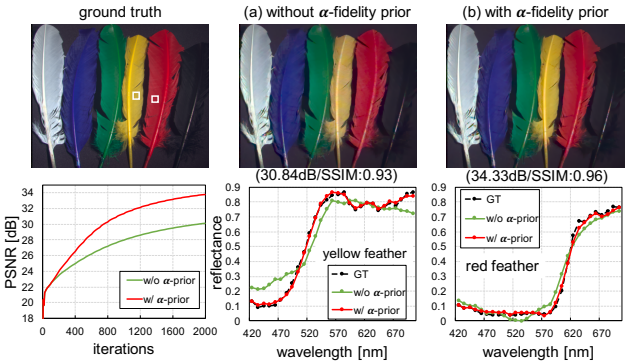


Fig. 9. Impact of our novel α-fidelity prior. (a) shows the reconstruction result without the α prior, while (b) shows the result with the prior. Insets indicate PSNR and SSIM compared to ground truth. The α prior increases both PSNR and SSIM significantly. In the second row, left, we show how the α prior increases accuracy over the number of iterations. The last two charts plot spectral accuracy comparisons for the yellow and the red feathers.

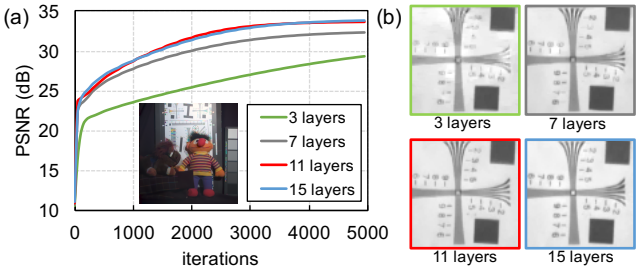


Fig. 10. Impact of the number of hidden layers. (a) For more than eleven layers, there is no significant improvement in the accuracy of the reconstruction. (b) Insets of the resulting reconstructions. Given the practical tradeoff between performance and memory, we set the number of hidden layers in our autoencoder to eleven.

Spatial Resolution. We evaluate the spatial resolution of our reconstructed spectral images by calculating spatial frequency responses as modulation transfer functions (MTFs). We reconstruct the standard spatial frequency measurement chart (ISO 12233), again using TwIST, SpaRSA, sparse coding, and our method. Figure 8 shows the results. Like other existing optimization methods, TwIST and SpaRSA show suboptimal reconstruction of spatial frequencies. While the recent data-driven approach based on sparse coding [Lin et al. 2014] improves this spatial resolution, our method clearly yields the best results.

5.2 Analysis of Parameters

Impact of the Fidelity Prior. One of the key novelties of our optimization formulation is our α-fidelity prior in Equation (11), relating the nonlinear representations from the trained autoencoder with the reconstruction problem. In Figure 9, this novel prior has a large impact on the accuracy of the reconstruction. PSNR increases significantly from 30.84dB to 34.33dB, while SSIM also increases from 0.93 to 0.96. Moreover, in the second row we show how the prior influences the PSNR with the number of iterations, as well as the reflectance accuracy for the yellow and the red feathers.

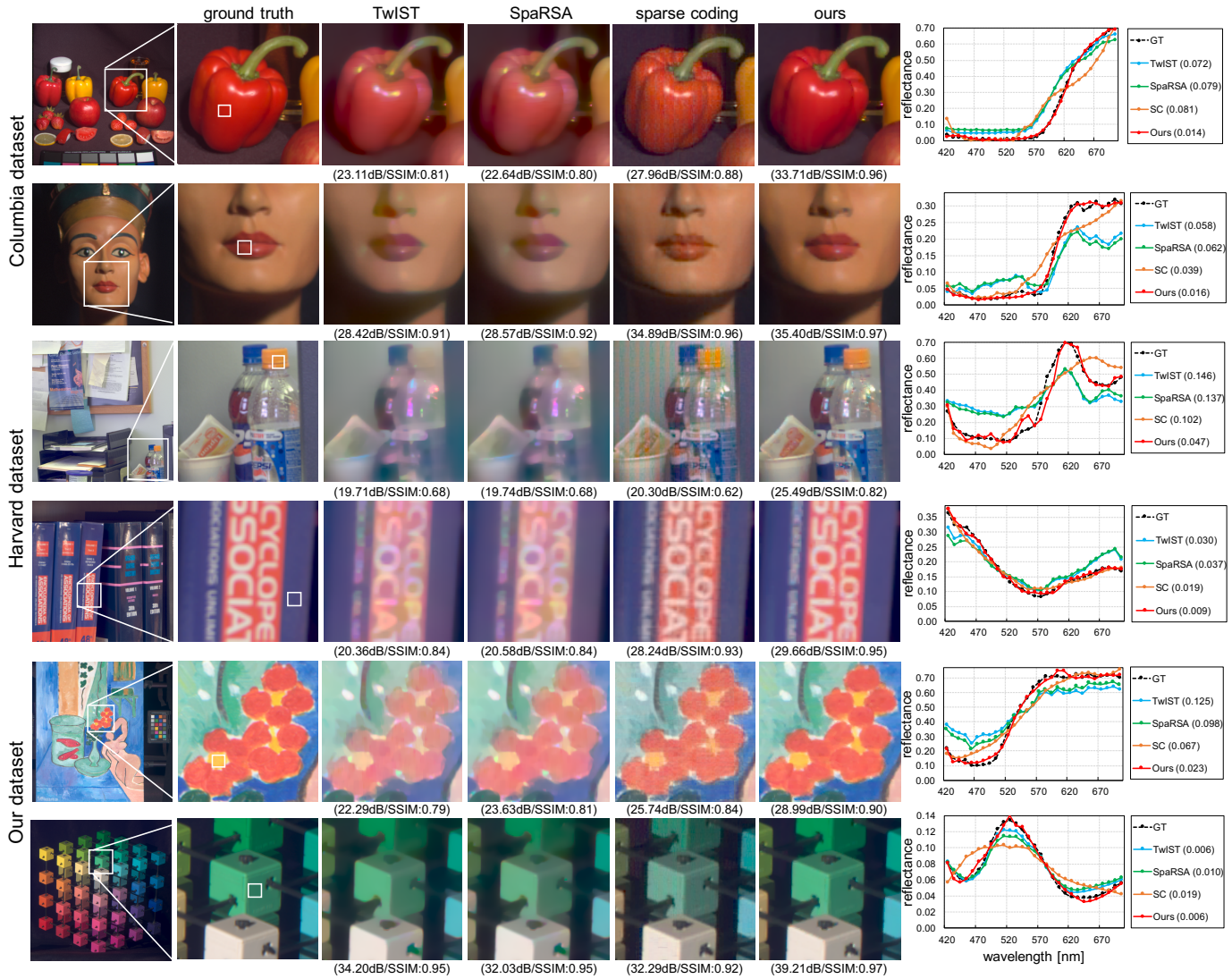


Fig. 11. Reconstruction results: TwIST, SparRSA, sparse coding, and our method, on three datasets: Columbia, Harvard, and our new dataset. The overlaid numbers in parenthesis show the average PSNRs and SSIMs of reflectance. The numbers in the plot legends indicate the spectral RMSEs of the reconstructed spectra. Our reconstruction method outperforms all three methods in terms of spatial resolution and spectral accuracy. Refer to supplemental materials for more results, including images for each reconstructed wavelength.

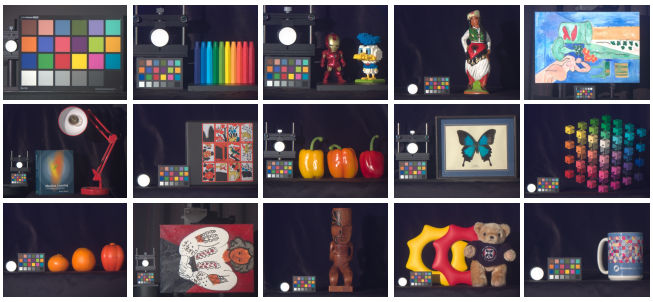


Fig. 12. Representative thumbnails of our new high-resolution, hyperspectral dataset (http://vclab.kaist.ac.kr/siggraphasia2017p1/).

Impact of Hidden Layers. Figure 10 shows the impact of the number of hidden layers on the spatial resolution of the reconstruction. We found that after eleven layers there is no significant increase of spatial resolution. Given the tradeoff between performance and memory, we set the number of hidden layers to eleven.

5.3 Additional Comparisons

We conducted an additional experiment to provide further comparisons with the beta process factor analysis (BPFA) [Rajwade et al. 2013] and a low-rank reconstruction method [Fu et al. 2016]. Our quantitative evaluations in this subsection are computed for reflectance, rather than radiance, to ensure color fidelity; this typically lowers PSNR values by 2.0 to 3.0 dB, compared to radiance. In BPFA,

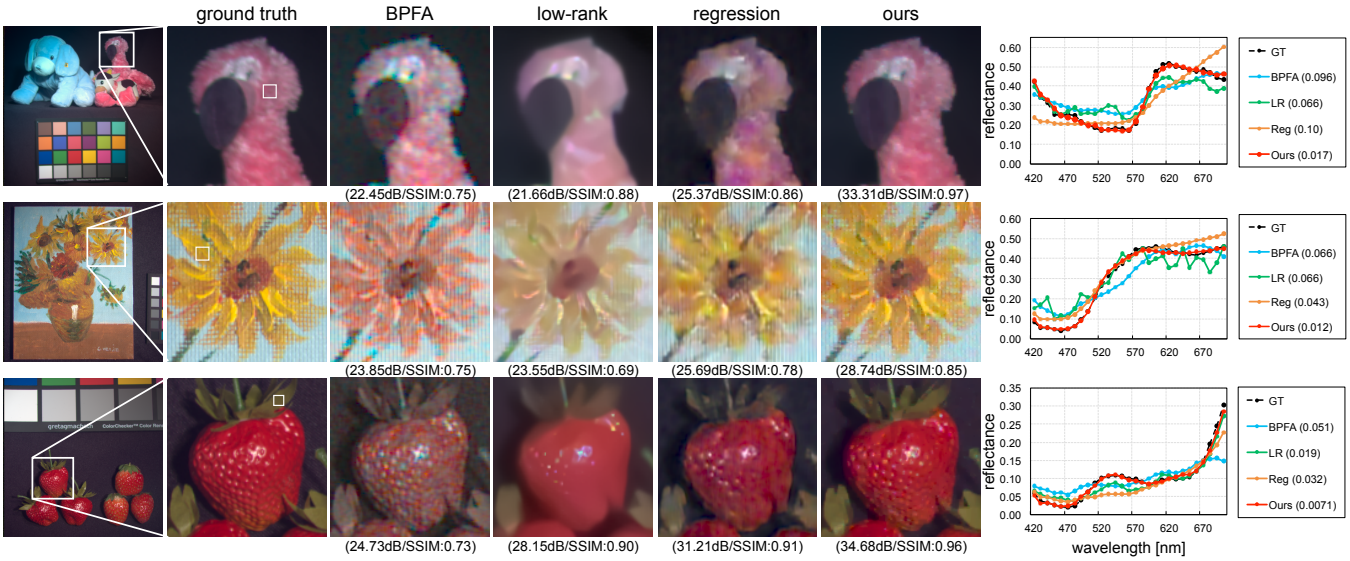


Fig. 13. Comparison of our results to BPFA [Rajwade et al. 2013], low-rank [Fu et al. 2016], and the reconstruction using a deep regression network. The plots on the right present the spectra of the area marked by small squares in the ground truth close-ups; the values indicate spectral RMSEs. Our reconstruction is significantly more accurate.

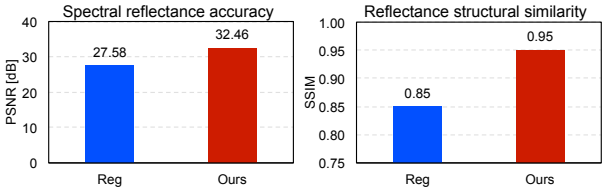


Fig. 14. Comparison of spectral reflectance accuracy (left) and spatial structure accuracy (right), for our method and a regression-based approach [Kulkarni et al. 2016]. The average PSNRs and SSIMs of our method are significantly higher (32.46 dB and 0.95), than the regression-based reconstruction (27.58 dB and 0.86).

hyperspectral images are reconstructed from coded inputs, adopting the reconstruction method used in blind compressive sensing. The latter method refines initial estimations of hyperspectral images exploiting the spectral-spatial correlation that exists in similar, non-local hyperspectral image patches. For the 32 hyperspectral images from the Columbia dataset, the average PSNR and SSIM measurements were 21.71 dB and 0.69 for BPFA, and 24.48 dB and 0.85 for the low-rank reconstruction, while for our method we obtained 32.46 dB and 0.95, respectively (see Figure 13). Note that BPFA is designed for multi-frame CASSI, but we only used a single input for fairness of comparison. As mentioned, the quality of the low-rank reconstruction depends on initial estimations, which is a TwIST reconstruction in this case. Therefore, the PSNR and SSIM values of the low-rank reconstruction are higher or equal to those of TwIST (23.74 dB/0.85) shown in Figure 6.

A straightforward learning-based reconstruction would train an end-to-end regression network taking compressive measurements as input, and outputting the corresponding original images; the modulation matrix Φ would be implicitly encoded in the regression model. We compare our reconstruction to a recent regression-based network [Kulkarni et al. 2016]. Since it had not been originally designed for hyperspectral imaging, we modified it to train a deep convolutional regression network that directly estimates a hyperspectral image from a compressive input. The revised deep regression network consists of eleven hidden convolutional layers with 64 feature maps. The convolutions are performed with 3×3 kernels, using ReLU activation functions. For training, we used the Columbia dataset, where the size of the images is 512×512. Note that this deep regression network is restricted to 512×512 images, since the model implicitly encodes the fixed modulation matrix Φ in the network. We trained the network using the ADAM optimizer. Packing eight images as a batch, training was carried out for 3000 epochs.

For comparisons, we used 32 spectral images from the Columbia dataset. Since the regression-based reconstruction does not need an optimization step, it is very fast (average of 0.14 sec.); however, it yields significantly less accurate reconstructions both in the spectral and spatial domains. Figure 13 (two rightmost close-ups) shows representative results, while Figure 14 shows PSNR and SSIM values averaged across the whole dataset. Besides the lower quality of the regression-based reconstruction, learning an end-to-end regression requires training a *different* model each time the image setup changes (image size, mask patterns, lens, or the pixel pitch of the sensor), which is highly impractical.

5.4 A New Hyperspectral Image Dataset

During our experiments, we found that the Columbia dataset offers images with a wide range of spectral information, but at low spatial resolution and slightly out-of-focus. Similarly, the Harvard dataset provides high spatial resolution, but limited spectral range. To improve this, we have additionally captured a new high-resolution dataset, consisting of 30 hyperspectral images covering a wide spectral range. A detailed description of the system appears in the supplemental material. Figures 11 and 12 show some examples; the complete dataset can be downloaded from our project website.

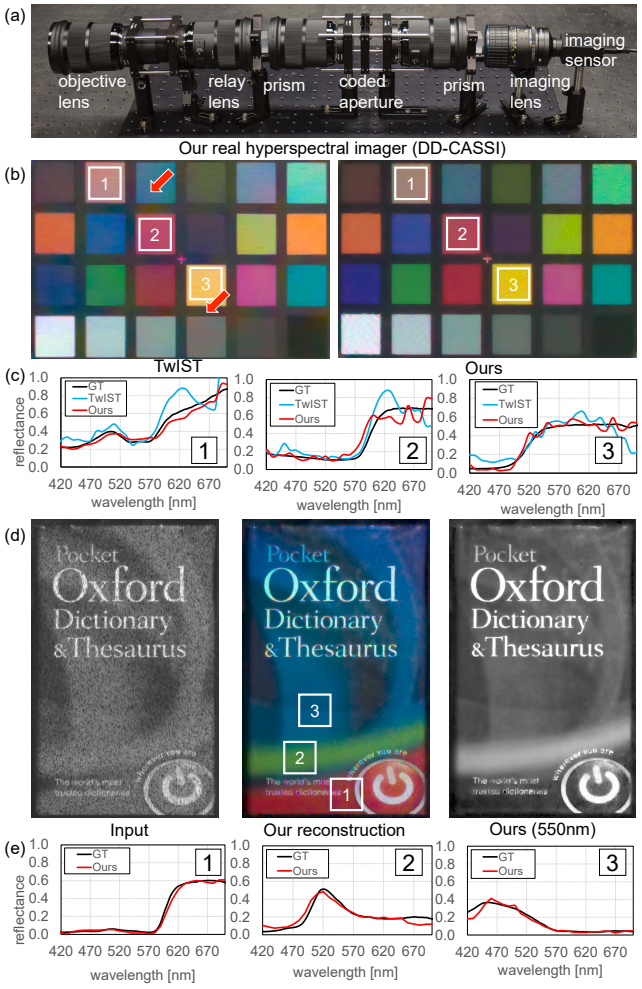


Fig. 15. (a) Our prototype DD-CASSI imaging system (b) and (c) Comparison between TwIST and our reconstruction. TwIST suffers from spatial artifacts (see arrows) and less accurate spectral reconstruction. (d) and (e) Additional result of our reconstruction.

5.5 Results on a Real Hyperspectral Camera

To further validate our reconstruction algorithm, we built a prototype of a spatial-spectral encoded DD-CASSI imaging system, shown in Figure 15(a). The system is made up of an apochromatic objective lens, relay lenses, two prisms (made of NBK-7, 2-degree angles, producing 13-pixel dispersion), a coded aperture, and a CCD imaging sensor. All the relay lenses (Sigma A, f/1.4) have the same focal length (50 mm) for one-to-one imaging. The camera is a Point-Grey Grasshopper (GS3 9.1MP Mono) with pixel pitch 3.69 μm. The coded aperture mask includes random binary patterns made through lithographic chrome etching on a quartz plate, where the pixel pitch of the binary patterns is 7.40μm. A pixel in the mask corresponds to two-by-two pixels in the CCD sensor of the imager. Scenes are captured under a solid-state plasma light source. We calibrated the optical properties of the system, such as the binary mask pattern $T(x, y)$ and the wavelength-dependent pixel shift function $\phi(\lambda)$ in Equation (2).

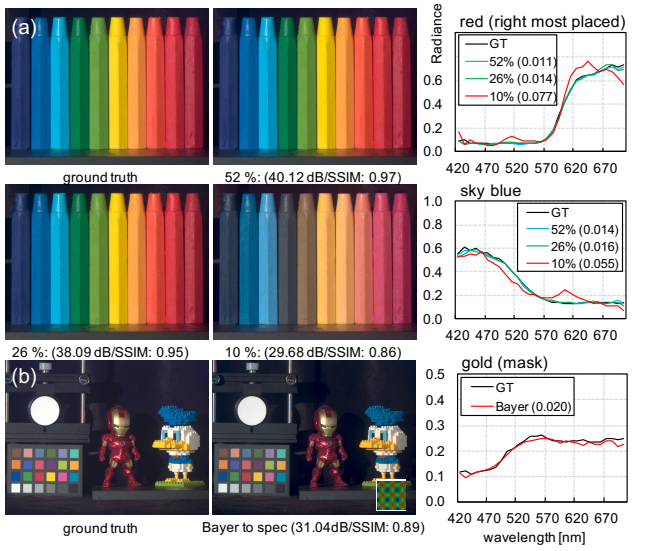


Fig. 16. (a) Hyperspectral reconstructions from subsampled wavelength information, using only 52 %, 26 % and 10 % of the samples (16, 8 and 3 spectral channels, respectively). Our interpolated reconstructions are very accurate, although performance predictably decays in the most extreme case. (b) Hyperspectral demosaicing from spatially-spectrally subsampled input in the Bayer-pattern. We use only four spectral wavelengths of 450 nm, 520 nm, 580 nm, and 650 nm, as shown in the inset of the second image. Our demosaiced hyperspectral image shows a good agreement with ground truth. The values in the legends of the plots indicate the spectral RMSEs.

Learning Illumination Invariance. To handle real-world input from our prototype, we retrain our model with additional datasets under various illuminations of different color temperatures. We created an additional training dataset of 192 hyperspectral images, using 32 hyperspectral reflectance images from the Columbia [Yasuma et al. 2010] dataset under five different color temperatures of 2000°K (CIE A, tungsten), 4000°K (fluorescent light), 5000°K (CIE D50), 6500°K (CIE D65), and 13,000°K (plasma), in addition to original reflectance images. This dataset is further augmented for scale invariance [Simonyan and Zisserman 2015], resulting in 384 new hyperspectral images in total. We sampled 19,200 tensor patches of size 96×96×31 from this augmented dataset.

Results on Real Data. Figure 15(b) compares reconstructions using TwIST[4] and our method. Plots in (c) compare spectral accuracy on the selected patches, for both methods and ground truth measured with a spectroradiometer. Our spectral reconstruction outperforms the conventional approach: TwIST reconstruction suffers from spatial artifacts (see arrows), and is less accurate in the spectral domain. (d) and (e) show another result of our reconstruction.

5.6 Applications

Hyperspectral Interpolation. Taking advantage of our spectral prior, our method allows to interpolate a multispectral image into a

[4]Since TwIST without any spectral prior requires a wider dispersion than our prior-based approach, we use 15-degree prisms to disperse light for TwIST in 89 pixels to provide a fair comparison.

hyperspectral image of higher spectral resolution, without any hardware modification. We simply substitute the measurement matrix Φ in Equation (11) with a wavelength subsampling matrix.

Figure 16(a) shows interpolated results by subsampling 52%, 26% and 10% of the original spectral wavelengths, which translates into 16, 8 and 3 channels, respectively. We compare our interpolated reconstructions for 31 wavelengths with ground truth. The accuracy of the reconstructions remains high, although it predictably decays when using only 10% of the information.

Hyperspectral Demosaicing. We further extend our interpolated reconstruction to enable hyperspectral demosicing, assuming that our input corresponds only to wavelengths of 450 nm, 520 nm, 580 nm, and 650 nm, according to conventional Bayer-patterns. We replace the Φ matrix with the spatially and spectrally subsampling matrix in Equation (11) We account for diffraction blur as a Gaussian blur for the Φ matrix, and set τ_2 to a very small value (10^{-8}) to avoid reconstruction blur. Figure 16(b) shows how our spectral reconstruction is remarkably accurate. This technique enables single-shot hyperspectral imaging using a Bayer-patterned multispectral input, analogous to demosaicing in a digital camera.

6 LIMITATIONS

Our reconstruction algorithm includes a total variation term to favor sparsity of gradients in the spatial domain, which relates spectral information to neighboring pixels. This can lead to suboptimal reconstruction of very fine image structures if the input is not of sufficient quality. This can be seen in Figure 17, where the input image is slightly out of focus: although our method still produces better results than other approaches, the reconstruction of the small details in the printed words is not perfect.

Fig. 17. Limitation example: out-of-focus blur in the input images leads to suboptimal reconstruction of fine details. Nevertheless, our result produces better results than existing approaches.

7 CONCLUSION AND FUTURE WORK

We have presented a novel hyperspectral image reconstruction method, which outperforms current state-of-the art methods for both spatial resolution and spectral accuracy. The two main steps are: (1) we train a natural spectrum prior as nonlinear representations using a convolutional autoencoder, and (2) we formulate a novel nonlinear optimization by using the autoencoder representations as spectral priors. Our reconstruction method can be applied to any compressive imaging architecture. Moreover, compared to the best performing method based on sparse coding, computational complexity is reduced by two orders of magnitude.

We have also built a prototype camera, and showed how our method outperforms other state-of-the-art reconstruction methods with real input. Last, we have presented two novel, high-accuracy spectral interpolation applications, which can be beneficial for many bandpass-filter based multispectral imaging systems. Further exploitation of the possibilities of these interpolation schemes remains an interesting avenue of future work. Automatically adjusting the total variation prior for better reconstruction of fine details is also an attractive problem. To foster further research on compressive imaging, we make our trained model, source code, and our new hyperspectral image dataset available through our project website.

ACKNOWLEDGMENTS

The authors would like to thank the anonymous reviewers, and Seung-Hwan Baek, Incheol Kim, Adrian Jarabo, and Paz Hernando for help and proof-reading. Min H. Kim acknowledges Korea NRF grants (2016R1A2B2013031, 2013M3A6A6073718) and additional support by Korea Creative Content Agency (KOCCA) in Ministry of Culture, Cross-Ministry Giga KOREA Project (GK17P0200), Sports and Tourism (MCST), Samsung Electronics (SRFC-IT1402-02), and an ICT R&D program of MSIT/IITP of Korea (R7116-16-1035). Diego Gutierrez acknowledges funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (CHAMELEON project, grant agreement No 682080), and from the Spanish Ministerio de Economía y Competitividad (project TIN2016-78753-P).

REFERENCES

M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. 2016. TensorFlow: A System for Large-scale Machine Learning. In *Proc. USENIX Conf. Operating Systems Design and Implementation (OSDI'16)*. 265–283.

Manya V. Afonso, José M. Bioucas-Dias, and Mário A. T. Figueiredo. 2011. An Augmented Lagrangian Approach to the Constrained Optimization Formulation of Imaging Inverse Problems. *IEEE Trans. Image Processing (TIP)* 20, 3 (2011), 681–695.

Sara Alvarez, Timo Kunkel, and Belen Masia. 2016. Practical Low-Cost Recovery of Spectral Power Distributions. *Computer Graphics Forum* 35, 1 (2016), 166–178.

Michael Attas, Edward Cloutis, Catherine Collins, Douglas Goltz, Claudine Majzels, James R Mansfield, and Henry H Mantsch. 2003. Near-infrared spectroscopic imaging in art conservation: investigation of drawing constituents. *J. Cultural Heritage* 4, 2 (2003), 127–136.

Seung-Hwan Baek, Incheol Kim, Diego Gutierrez, and Min H. Kim. 2017. Compact Single-Shot Hyperspectral Imaging Using a Prism. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2017)* 36, 6 (2017).

Jose M. Bioucas-Dias and Mario A. T. Figueiredo. 2007. A new TwIST: two-step iterative shrinkage/thresholding for image restoration. *IEEE Trans. Image Processing (TIP)* 16, 12 (2007), 2992–3004.

N. Brusco, S. Capeleto, M. Fedel, A. Paviotti, L. Poletto, G. M. Cortelazzo, and G. Tondello. 2006. A system for 3D modeling frescoed historical buildings with multispectral texture information. *Machine Vision and Applications* 17, 6 (2006), 373–393.

Xun Cao, Hao Du, Xin Tong, Qionghai Dai, and Stephen Lin. 2011. A prism-mask system for multispectral video acquisition. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)* 33, 12 (2011), 2423–2435.

A. Chakrabarti and T. Zickler. 2011. Statistics of Real-World Hyperspectral Images. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*. 193–200.

Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. 2014. Deep Learning-Based Classification of Hyperspectral Data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 7, 6 (2014), 2094–2107.

B. Du, W. Xiong, J. Wu, L. Zhang, L. Zhang, and D. Tao. 2016. Stacked Convolutional Denoising Auto-Encoders for Feature Representation. *IEEE Trans. Cybernetics* PP, 99 (2016), 1–11.

Ying Fu, Yinqiang Zheng, Imari Sato, and Yoichi Sato. 2016. Exploiting Spectral-Spatial Correlation for Coded Hyperspectral Image Restoration. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Nahum Gat. 2000. Imaging spectroscopy using tunable filters: a review. In *Proc. AeroSense 2000*. 50–64.

M E Gehm, R John, D J Brady, R M Willett, and T J Schulz. 2007. Single-shot compressive spectral imaging with a dual-disperser architecture. *OSA Optics Express (OE)* 15, 21 (2007), 14013–27.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Aistats*, Vol. 9. 249–256.

Mohammad Golbabaee, Simon Arberet, and Pierre Vandergheynst. 2013. Compressive source separation: Theory and methods for hyperspectral imaging. *IEEE Transactions on Image Processing (TIP)* 22, 12 (2013), 5096–5110.

Ralf Habel, Michael Kudenov, and Michael Wimmer. 2012. Practical spectral photography. In *Computer Graphics Forum (Proc. EUROGRAPHICS 2012)*, Vol. 31. Wiley Online Library, 449–458.

Geoffrey E Hinton and Ruslan R Salakhutdinov. 2006. Reducing the dimensionality of data with neural networks. *Science* 313, 5786 (2006), 504–507.

Gudrun Hoye and Andrei Fridman. 2013. Mixel camera – a new push-broom camera concept for high spatial resolution keystone-free hyperspectral imaging. *OSA Optics Express (OE)* 21, 9 (2013), 11057–11077.

Francisco Imai, Lawrence Taplin, and Ellen Day. 2002. *Comparison of the accuracy of various transformations from multi-band images to reflectance spectra.* Technical Report ISO 12233:2000. Rochester Institute of Technology.

Daniel S. Jeon, Inchang Choi, and Min H. Kim. 2016. Multisampling Compressive Video Spectroscopy. *Computer Graphics Forum* 35, 2 (2016), 467–477.

Min H Kim. 2013. 3D Graphics Techniques for Capturing and Inspecting Hyperspectral Appearance. In *Ubiquitous Virtual Reality (ISUVR), 2013 Int. Symp. on*. IEEE, 15–18.

Min H. Kim, Todd Alan Harvey, David S. Kittle, Holly Rushmeier, Julie Dorsey, Richard O. Prum, and David J. Brady. 2012a. 3D Imaging Spectroscopy for Measuring Hyperspectral Patterns on Solid Objects. *ACM Trans. Graph. (TOG)* 31, 4 (2012), 38:1–11.

Min H. Kim, Holly Rushmeier, John ffrench, and Irma Passeri. 2012b. Developing Open-Source Software for Art Conservators. In *VAST12: The 13th International Symposium on Virtual Reality, Archaeology and Intelligent Cultural Heritage*. Eurographics Association, Brighton, England, 97–104.

Min H. Kim, Holly Rushmeier, John ffrench, Irma Passeri, and David Tidmarsh. 2014. Hyper3D: 3D Graphics Software for Examining Cultural Artifacts. *ACM Journal on Computing and Cultural Heritage* 7, 3 (2014), 1:1–19.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. In *The International Conference on Learning Representations (ICLR)*.

David Kittle, Kerkil Choi, Ashwin Wagadarikar, and David J Brady. 2010. Multiframe image estimation for coded aperture snapshot spectral imagers. *Applied Optics (AO)* 49, 36 (2010), 6824–6833.

Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok. 2016. ReconNet: Non-Iterative Reconstruction of Images From Compressively Sensed Measurements. In *The IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*.

Haebom Lee and Min H. Kim. 2014. Building a Two-Way Hyperspectral Imaging System with Liquid Crystal Tunable Filters. In *Proc. Int. Conf. Image and Signal Processing (ICISP)*. 26–34.

Biao Leng, Shuang Guo, Xiangyang Zhang, and Zhang Xiong. 2015. 3D object retrieval with stacked local convolutional autoencoder. *Signal Processing* 112 (2015), 119–128.

Chengbo Li, Ting Sun, Kevin F. Kelly, and Yin Zhang. 2012. A Compressive Sensing and Unmixing Scheme for Hyperspectral Data Processing. *IEEE Trans. Image Processing (TIP)* 21, 3 (2012), 1200–1210.

Yunsong Li, Weiying Xie, and Huaqing Li. 2017. Hyperspectral image reconstruction by deep convolutional neural network for classification. *Pattern Recognition (PR)* 63 (2017), 371 – 383.

Xing Lin, Yebin Liu, Jiamin Wu, and Qionghai Dai. 2014. Spatial-spectral Encoded Compressive Hyperspectral Imaging. *ACM Trans. Graph. (TOG)* 33, 6 (2014).

Zhouhan Lin, Yushi Chen, Xing Zhao, and Gang Wang. 2013. Spectral-spatial classification of hyperspectral image using autoencoders. In *Proc. Int. Conf. Signal Process. Commun.* 1–5.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully Convolutional Networks for Semantic Segmentation. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*. 3431–3440.

Xiaorui Ma, Hongyu Wang, and Jie Geng. 2016. Spectral-Spatial Classification of Hyperspectral Image Based on Deep Auto-Encoder. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 9, 9 (2016), 4073–4085.

Emmanuel Maggiori, Yuliya Tarabalka, Guillaume Charpiat, and Pierre Alliez. 2016. Convolutional Neural Networks for Large-Scale Remote Sensing Image Classification. *IEEE Trans. Geosci. Remote Sens.* (2016), 645 – 657.

Alkhazur Manakov, John F Restrepo, Oliver Klehm, Ramon Hegedüs, Elmar Eisemann, Hans-Peter Seidel, and Ivo Ihrke. 2013. A reconfigurable camera add-on for high dynamic range, multispectral, polarization, and light-field imaging. *ACM Trans. Graph. (Proc. SIGGRAPH 2013)* 32, 4 (2013), 47.

Julio Marco, Quercus Hernandez, Adolfo MuÃśoz, Yue Dong, Adrian Jarabo, Min Kim, Xin Tong, and Diego Gutierrez. 2017. DeepToF: Off-the-Shelf Real-time Correction of Multipath Interference in Time-of-Flight Imaging. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2017)* 36, 6 (2017).

Gabriel Martin, Jose M. Bioucas-Dias, and Antonio J. Plaza. 2015. HYCA: A New Technique for Hyperspectral Compressive Sensing. *IEEE Trans. Geoscience and Remote Sensing* 53, 5 (2015), 2819–2831.

Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. 2011. Stacked convolutional auto-encoders for hierarchical feature extraction. In *Proc. Int. Conf. Artificial Neural Networks (ICANN)*. 52–59.

Yi Peng, Deyu Meng, Zongben Xu, Chenqiang Gao, Yi Yang, and Biao Zhang. 2014. Decomposable Nonlocal Tensor Dictionary Learning for Multispectral Image Denoising. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*.

Wallace M Porter and Harry T Enmark. 1987. A system overview of the airborne visible/infrared imaging spectrometer (AVIRIS). In *Technical Symposium*. International Society for Optics and Photonics, 22–31.

Ajit Rajwade, David Kittle, Tsung-Han Tsai, David Brady, and Lawrence Carin. 2013. Coded Hyperspectral Imaging and Blind Compressive Sensing. *SIAM Journal on Imaging Sciences* 6, 2 (2013). https://doi.org/10.1137/120875302

Konstantinos Rapantzikos and Costas Balas. 2005. Hyperspectral imaging: potential in non-destructive analysis of palimpsests. In *Proc. IEEE Int. Conf. Image Processing (ICIP)*, Vol. 2. IEEE, II–618.

K. Simonyan and A. Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Proc. Int. Conf. Learning Representation (ICLR)*.

Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. 2014. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*.

Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. 2010. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *J. Mach. Learn. Res.* 11 (2010), 3371–3408.

J. J. Vos. 1978. Colorimetric and photometric properties of a 2-deg fundamental observer. *Color Res. Appl.* 3 (1978), 125–128.

Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. 2008. Single disperser design for coded aperture snapshot spectral imaging. *OSA Applied Optics (AO)* 47, 10 (2008), B44–B51.

Lizhi Wang, Zhiwei Xiong, Dahua Gao, Guangming Shi, Wenjun Zeng, and Feng Wu. 2015. High-Speed Hyperspectral Video Acquisition With a Dual-Camera Architecture. In *Proc. IEEE Conf. Comput. Vis. Pattern. Recognit. (CVPR)*.

S.J. Wright, R.D. Nowak, and M.A.T. Figueiredo. 2009. Sparse reconstruction by separable approximation. *IEEE TSP* 57, 7 (2009), 2479–2493.

Yuehao Wu, Iftekhar O Mirza, Gonzalo R Arce, and Dennis W Prather. 2011. Development of a digital-micromirror-device-based multishot snapshot spectral imaging system. *OSA Optics Letters (OL)* 36, 14 (2011), 2692–4.

Chen Xing, Li Ma, and Xiaoquan Yang. 2016. Stacked Denoise Autoencoder Based Feature Extraction and Classification for Hyperspectral Images. *Journal of Sensors* 2016 (2016), 1–10.

F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar. 2010. Generalized Assorted Pixel Camera: Postcapture Control of Resolution, Dynamic Range, and Spectrum. *IEEE Trans. Image Processing (TIP)* 19, 9 (Sept 2010), 2241–2253.

Jaime Zabalza, Jinchang Ren, Jiangbin Zheng, Huimin Zhao, Chunmei Qing, Zhijing Yang, Peijun Du, and Stephen Marshall. 2016. Novel Segmented Stacked Autoencoder for Effective Dimensionality Reduction and Feature Extraction in Hyperspectral Imaging. *Neurocomputing* 214 (2016), 1–10.

Qiang Zhang, Robert Plemmons, David Kittle, David Brady, and Sudhakar Prasad. 2011. Joint segmentation and reconstruction of hyperspectral data with compressed measurements. *OSA Applied Optics (AO)* 50, 22 (2011), 4417–4435.

Yuting Zhang, Kibok Lee, and Honglak Lee. 2016. Augmenting Supervised Neural Networks with Unsupervised Objectives for Large-scale Image Classification. In *Proc. International Conference on Machine Learning (ICML)*.

Received May 2017; revised August 2017; accepted August 2017; final version November 2017