

# Compact Single-Shot Hyperspectral Imaging Using a Prism

SEUNG-HWAN BAEK and INCHEOL KIM, KAIST  
DIEGO GUTIERREZ, Universidad de Zaragoza, I3A  
MIN H. KIM, KAIST



Fig. 1. We propose a compact single-shot spectral imaging method that enables us to capture hyperspectral images using a conventional DSLR camera equipped with an ordinary prism only. Our computational method reconstructs full spectral information of a scene from dispersion over edges. Therefore, our setup requires no coded aperture mask and no collimating optics, which are necessary for traditional hyperspectral imaging. Our method allows for simple and cost-efficient hyperspectral imaging with high accuracy. (a) Our acquisition setup. (b) Input image. (c) Reconstructed hyperspectral image (d) Corresponding captured spectral channels. (e) Spectral plots of two patches from the captured ColorChecker, compared to ground truth.

We present a novel, compact single-shot hyperspectral imaging method. It enables capturing hyperspectral images using a conventional DSLR camera equipped with just an ordinary refractive prism in front of the camera lens. Our computational imaging method reconstructs the full spectral information of a scene from dispersion over edges. Our setup requires no coded aperture mask, no slit, and no collimating optics, which are necessary for traditional hyperspectral imaging systems. It is thus very cost-effective, while still highly accurate. We tackle two main problems: First, since we do not rely on collimation, the sensor records a projection of the dispersion information, distorted by perspective. Second, available spectral cues are sparse, present only around object edges. We formulate an image formation model that can predict the perspective projection of dispersion, and a reconstruction method that can estimate the full spectral information of a scene from sparse dispersion information. Our results show that our method compares well with other state-of-the-art hyperspectral imaging systems, both in terms of spectral accuracy and spatial resolution, while being orders of magnitude cheaper than commercial imaging systems.

CCS Concepts: • Computing methodologies → Hyperspectral imaging;

Additional Key Words and Phrases: Hyperspectral imaging, image reconstruction

## ACM Reference format:

Seung-Hwan Baek, Incheol Kim, Diego Gutierrez, and Min H. Kim. 2017. Compact Single-Shot Hyperspectral Imaging Using a Prism . *ACM Trans. Graph.* 36, 6, Article 217 (November 2017), 12 pages.

Authors' addresses: Seung-Hwan Baek and Incheol Kim, KAIST, School of Computing, 291 Daehak-ro, Yuseong-gu, Daejeon, South Korea 34141; Diego Gutierrez, Universidad de Zaragoza, I3A, María 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.

© 2017 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3130800.3130896>.

<https://doi.org/10.1145/3130800.3130896>

## 1 INTRODUCTION

Hyperspectral imaging has wide applicability in many fields such as physically-accurate material appearance, automatic segmentation and matching, material classification, or material inspection for mining and geology [Kim 2013; Kim et al. 2012a,b, 2014; Lin et al. 2014; Wagadarikar et al. 2008]. However, hyperspectral imaging systems are still very expensive, with starting prices falling in the range of \$ 25,000 – \$ 100,000. They require specialized hardware such as collimating optics, or a lithographic coded aperture with microscale patterns and a diffraction grating [Choi et al. 2017; Jeon et al. 2016; Lee and Kim 2014; Lin et al. 2014; Wagadarikar et al. 2008], in addition to professional engineering skills for handling and assembling such components. In general, these systems are built for specific purposes such as aerial remote sensing, or military applications; as such, they are not affordable nor practical for ordinary users.

Traditional scanning systems isolate measurements for each wavelength using filters, which results in a slow process; moreover, the spectral resolution is limited by the type and number of filters used. Hyperspectral imaging techniques such as coded aperture snapshot spectral imaging (CASSI) employ a coded aperture as an essential element to capture spectral information. Subsequent spectral reconstruction of the image relies on a spatially-invariant dispersion model, for which collimating optical setups are required. This alters significantly the form factor of the system.

To overcome these limitations, we present a novel single-shot technique for hyperspectral imaging, which requires just a conventional DSLR camera and a simple glass prism placed in front of the lens. It thus avoids expensive or specialized hardware, requires

no advanced skills, and has a minimal impact on the form factor, allowing general users to freely capture hyperspectral information.

Using a simple prism presents two main **technical challenges**: First, existing spatially-invariant dispersion models cannot be applied to our system. This is due to the absence of collimating optics (common in professional hyperspectral imagers) and the resulting wavelength-dependent, nonlinear refractive distortion created by the prism, as shown in Figure 1(b). Second, since our setup includes no coded aperture mask, available spectral cues are sparsely limited as dispersion over edges. In addition, since our setup captures a scene in its full resolution without using a diffraction grating, spectral cues are also sparse in a single direction. The new reconstruction algorithm must be able to reconstruct the full spectral information of a scene from sparse dispersion over edges, without relying on spectral signatures from a coded aperture.

To solve these challenges, we make the following **contributions**: First, we introduce a novel image formation model that predicts the perspective projection of dispersion, yielding the dispersion direction and magnitude of each wavelength at every pixel. Second, we propose a novel calibration method to estimate the *spatially-varying* dispersion of the prism, given the absence of collimating optics in our setup. Last, we develop a novel reconstruction algorithm leveraging sparse dispersion over edges. Our reconstruction algorithm consists of three main steps: edge restoration, gradient estimation, and spectral reconstruction. See Figure 2 for an overview of our algorithm.

In summary, we introduce a novel, single-shot hyperspectral imaging method, using only a single prism placed in front of a conventional camera. Our system requires no additional hardware, keeping its form factor compact and simple. We obtain hyperspectral images at virtually the full resolution of the imaging sensor, while making hyperspectral imaging practical.

## 2 RELATED WORK

Hyperspectral imaging has been researched extensively in the last decade, e.g., [Kim 2013]. Existing methods can be categorized into three different groups: spectral scanning, computed tomography imaging, and compressive hyperspectral imaging.

**Spectral Scanning.** The most straightforward approach in hyperspectral imaging is to isolate measurements using different bandpass or liquid crystal tunable filters, scanning the entire spectrum to form the final image [Gat 2000; Lee and Kim 2014; Rapantzikos and Balas 2005]. In addition, using dispersive optics such as a prism or a diffraction grating, scanning-based approaches can image each wavelength in isolation through a slit, using for instance whiskbroom or pushbroom scanners [Brusco et al. 2006; Porter and Enmark 1987]. While scanning yields high spatial resolution, the spectral resolution of this approach is limited by the number of filters used. In contrast, our hyperspectral imaging system captures an image with continuous dispersion with a single shot, resulting in higher spectral resolution without severely sacrificing spatial resolution.

**Compressive Imaging.** Coded aperture snapshot spectral imaging (CASSI) [Gehm et al. 2007; Wagadarikar et al. 2008] was developed in an effort to overcome the limitations of spectral scanning systems. A coded aperture is placed in front of or behind a prism (or a

diffraction grating) via collimation in the optical path of the imaging system. The coded aperture is used to encode spectral signatures, which are later used to reconstruct the compressive input into a complete hyperspectral image. Multiple sampling techniques have been introduced to further improve accuracy using a micro-translation stage [Kittle et al. 2010], or a kaleidoscope [Jeon et al. 2016]. Lin et al. [2014] and Choi et al. [2017] introduced learning-based solutions for enhancing spatial and spectral resolution of reconstructed images. Traditional compressive imaging systems are large and expensive due to additional elements such as collimating optics and coded masks, making them bulky and hard to handle in practice. In contrast, our system is extremely compact, needing just a prism in front of a regular camera lens, without any coded aperture nor complicated collimating optics.

Recently, Cao et al. [2011a] proposed a low-cost spectral imaging method, a so-called prism-mask multispectral video imaging system (PMVIS). The authors isolate spectral dispersion by placing a large mask of pinholes in front of a prism, thus creating a simple setup with affordable elements. However, such a large mask must be carefully installed in front of a large black box, at a relatively large distance from the prism, in order to properly isolate dispersed regions. This results in a large form factor, again hindering usability. Moreover, the number of holes in the mask determines the spatial resolution of the image, which is much lower than the image sensor resolution (less than 10%), while overall light throughput is reduced significantly. Alvarez et al. [2016] proposed a simple system based on a cheap, off-the-shelf diffraction grating. The system was designed to recover the spectral signature of visible light sources, although it cannot reconstruct the full hyperspectral image.

**Computed Tomography Imaging.** Computed tomography imaging spectrometry (CTIS) [Johnson et al. 2007; Okamoto et al. 1993] uses a diffraction grating to split incident light rays into a number of spectral projections on an image plane. Since multiple sub-images need to be captured with a single sensor, the effective spatial resolution of the reconstructed images is less than 10% of the original sensor resolution. Habel et al. [2012] introduced a cheaper solution for hyperspectral imaging by applying CTIS to a conventional DSLR camera. However, the system suffers from a similar loss of spatial resolution. In contrast, our system reconstructs hyperspectral images at virtually the full spatial resolution of the image sensor.

Snapshot image mapping spectrometry [Gao et al. 2010] employs a micro-mirror array as an image mapper to split incoming rays into strips, followed by dispersion and imaging with a prism and lens array. The spectral intensity of a scene point is directly measured by a single pixel on the camera sensor similar to PMVIS [Cao et al. 2011a]. However, the image mapping sacrifices spatial resolution, while the system setup requires a prism, a mirror, and a lens array, making it complex. In contrast, our proposed system utilizes a simple setup with specially designed computational algorithms, allowing for compact hyperspectral imaging without sacrificing sensor resolution.

Recently, Takatani et al. [2017] employed faced reflectors with color filters by exploiting multiple reflections through the filters. However, the system is limited to flat objects, and the spatial resolution is reduced because multiple reflections of the objects must be

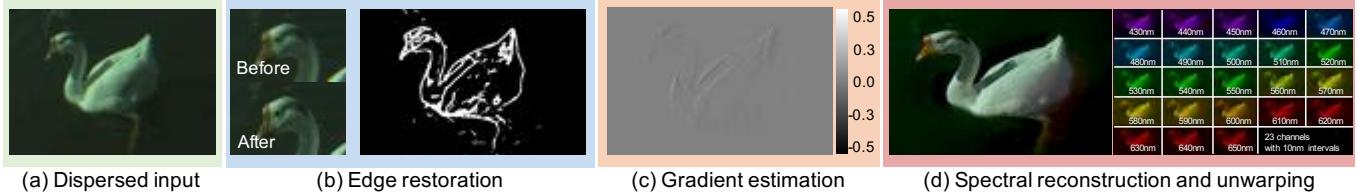


Fig. 2. Overview of our reconstruction algorithm. (a) We take a dispersed RGB image as an input. (b) We align the dispersed image and reconstruct edges (Section 5.1). (c) Gradient values are estimated from dispersion (Section 5.2.1). (d) We reconstruct the final hyperspectral image using the gradient information as spectral cues (Section 5.2.2). The geometric distortion introduced by the refraction through the prism is corrected in the final hyperspectral image.

captured in a single image. In order to obtain high-resolution spectral images, hybrid imaging systems [Cao et al. 2011b; Kawakami et al. 2011] have been proposed by employing an additional high-resolution, trichromatic imaging device with red, green, and blue (RGB) filters. However, it increases the size of the system, and introduces calibration issues between the heterogeneous systems.

*Estimating Spectrum from RGB.* Several works approximate spectral information from a single RGB input. Smit [1999] proposed a regression-based method that approximates a spectrum as a linear combination of basis functions of the standard XYZ matching functions. The estimation algorithm relies on piecewise smoothness constraints along the spectrum. Nguyen et al. [2014] proposed a data-driven approach that learns spectral distributions of the reflectance of natural objects. While regressing the spectral information from an RGB input, they use the reflectance model, trained with a hyperspectral image dataset, as a prior for optimization. However, these regression-based approaches can only approximate the actual spectra. The spectral accuracy of these methods is limited by the metameric input of the RGB camera, for which filter bandwidth is about 100 nm. The spectral information measured by our method is more accurate than these regression-based approximations.

### 3 IMAGE FORMATION MODEL

A hyperspectral image can be described as a three-dimensional cube  $I(p, \lambda) \in \mathbb{R}^{X \times Y \times \Lambda}$ , where  $X$ ,  $Y$  and  $\Lambda$  represent the horizontal, vertical and spectral axes, respectively;  $p$  represents a sensor pixel  $(x, y)$ , and  $\lambda$  represents wavelength. Different from traditional hyperspectral imagers, we just use a conventional RGB camera, plus a prism in front of the lens. Our spectral image formulation is therefore different from classic compressive imaging architectures [Gehm et al. 2007; Wagadarikar et al. 2008]. Since the prism disperses incident rays of light, resulting in shifts of the spectral cube  $I$  along  $\lambda$ , we can describe our image formation model as:

$$J(p, c) = \int \Omega(c, \lambda) I(\Phi_\lambda(p), \lambda) d\lambda, \quad (1)$$

where  $J(p, c)$  is the linear RGB image captured by the camera (with  $c \in \{R, G, B\}$ ),  $\Omega(c, \lambda)$  is the transmission function encoding the camera response for channel  $c$  and wavelength  $\lambda$ , and  $\Phi_\lambda(p)$  represents the spatially-varying, nonlinear dispersion caused by the prism, modeled as a shift operator at each pixel  $p$  for each wavelength  $\lambda$ . We can reformulate this model in a discrete form as  $J(p, c) = \sum_\lambda \Omega(c, \lambda) I(\Phi_\lambda(p), \lambda)$ , which in matrix-vector form

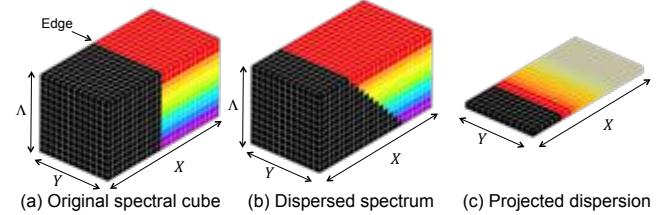


Fig. 3. (a) 3D spectral data cube  $i$ : the original data cube consists of two spatial axes ( $X$  and  $Y$ ) and one spectral axis ( $\Lambda$ ). (b) Dispersed spectrum  $\Phi i$ : the prism shifts the spectral information along  $X$ . (c) Projected dispersion  $\Omega \Phi i$ : the dispersed spectrum is then integrated along the spectral axis, following the camera response functions. The spectral information of the original data cube is embedded around edges in the projected dispersion. Our method takes the projected information with dispersion as input, and reconstructs the original spectral data cube by analyzing the dispersion around edges.

becomes:

$$j = \Omega \Phi i, \quad (2)$$

where  $j \in \mathbb{R}^{XY \times 1}$  and  $i \in \mathbb{R}^{XY \Lambda \times 1}$  are vectorized linear RGB and hyperspectral images, respectively, and  $\Omega$  is an operator converting spectral information to RGB (refer to Section 6). Last,  $\Phi \in \mathbb{R}^{XY \Lambda \times XY \Lambda}$  is a matrix describing the direction and magnitude of dispersion per pixel. Figure 3 shows a schematic diagram of the spectral data cube at each step:  $i$ ,  $\Phi i$  and  $\Omega \Phi i$ .

In the following sections, we will first derive our dispersion function  $\Phi_\lambda(p)$  in Equation (1) (Section 4); we will then describe our reconstruction method (Section 5). The camera and prism parameters will be estimated after calibration in Section 6.

### 4 SPATIALLY-VARYING DISPERSION

Figure 4 presents a schematic view of our prism-based hyperspectral capture model. Without a prism, a ray of light travels undisturbed from scene point  $q$ , and reaches the sensor at a point  $p_d$  through the optical center  $o$  of the camera lens. When the prism is placed in front of the lens,  $q$  instead projects through  $o$  to new wavelength-dependent positions  $p_\lambda$  on the sensor, after two refractions at  $s_\lambda^i$  and  $s_\lambda^o$ . Our goal in this section is to obtain the dispersion function  $\Phi_\lambda$  (Equation (1)), which models the dispersion created by the prism placed in front of the lens. Traditional compressive hyperspectral imagers employ complex collimating optics, which allows to model dispersion as a simple linear translation [Wagadarikar et al. 2008]. However, since we do not rely on optical collimation in

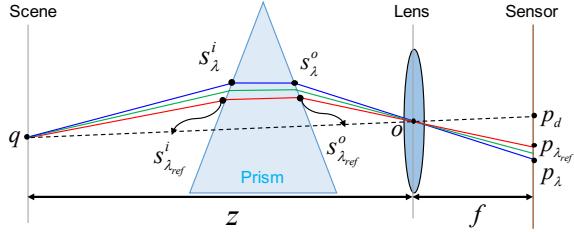


Fig. 4. Without a prism, a scene point  $q$  is imaged to  $p_d$  on the sensor plane, through the optical center  $o$  (dotted line). Introducing a prism, wavelength-dependent dispersion affects the imaging process. A ray from  $q$  of wavelength  $\lambda$  intersects the surfaces of the prism at  $s_{\lambda}^i$  and  $s_{\lambda}^o$ , being finally projected on the sensor at pixel  $p_{\lambda}$  (e.g., blue line). For our hyperspectral imaging model, we first derive the refraction function  $\Psi_{\lambda}$  relating  $p_{\lambda}$  and  $p_d$  for every wavelength in the visible spectrum. Then, we obtain the dispersion function  $\Phi_{\lambda}$  relating  $p_{\lambda}$  to a reference-wavelength pixel  $p_{\lambda_{ref}}$  ( $\lambda_{ref}=550$  nm, see text for the full details).

our compact setup, dispersion in our imaging model becomes both *spatially-varying* and *nonlinear*.

To derive  $\Phi_{\lambda}$ , we follow a two-stage approach: First, we develop a *refraction* function  $\Psi_{\lambda}$  for each wavelength, which predicts each pixel's wavelength-dependent refractive shift; in other words, it describes the relation between the direct-ray point  $p_d$  and  $p_{\lambda}$  at the sensor. Second, we find per-pixel correspondences for all wavelengths in the captured image, resulting in our *dispersion* function  $\Phi_{\lambda}$ . We describe these steps in the following subsections. Note that our single-shot approach does not require a direct capture without a prism, so  $p_d$  is never imaged. However, we will rely on  $p_d$  in our derivations, and eliminate this dependency when developing the final dispersion model (Section 4.2). To explicitly take depth  $z$  into account, we redefine points in three dimensional space  $\mathbb{R}^3$ , so that a pixel  $(x, y)$  on the sensor is now represented as a 3D-vector  $[x, y, -f]^T$ , where  $f$  represents focal length (see Figure 4).

#### 4.1 Refraction Model

We first seek a refraction model  $\Psi$  in matrix form, describing the refractive shift function  $\Psi_{\lambda}$  for every pixel as the relation between points  $p_d$  and  $p_{\lambda}$  at the sensor, for which we will describe both a refracted and a direct ray through  $o$ . Note that our refraction model concerns itself with refracted rays through  $o$ , which is different from the traditional light-through-a-prism model commonly used to illustrate dispersion.

*Refracted Ray*. From  $p_{\lambda}$ , we obtain the intersection points  $s_{\lambda}^o$  and  $s_{\lambda}^i$  with simple geometric optics, and using Snell's law at the surface of the prism. From  $s_{\lambda}^i = [s_x, s_y, s_z]^T$ , using again Snell's law, we obtain the refraction direction vector  $\overline{s_{\lambda}^i q} = [v_x, v_y, v_z]^T$ . We can then formulate a scene point  $q$  as a function of depth  $z$  and wavelength  $\lambda$ , as follows:

$$q = \Theta_{\lambda}(z; p_{\lambda}) \equiv \left[ s_x + \frac{z - s_z}{v_z} v_x, s_y + \frac{z - s_z}{v_z} v_y, z \right]^T. \quad (3)$$

*Direct Ray vs. Refracted Ray*. We can now formulate the depth-dependent relation between points  $p_d$  and  $p_{\lambda}$  by simple perspective

projection, as our function  $\Psi_{\lambda}$  describing the refractive shift:

$$p_d = \Psi_{\lambda}(z; p_{\lambda}) \equiv -\frac{f}{z} \Theta_{\lambda}(z; p_{\lambda}). \quad (4)$$

This function  $\Psi_{\lambda}$  will enable us to derive our final dispersion model  $\Phi$  as a vector field, which will in turn allow us to establish per-pixel correspondences along the visible spectrum in the captured image, as we explain next.

#### 4.2 Dispersion Model

We now seek a dispersion model  $\Phi$  in matrix form, describing the dispersion function  $\Phi_{\lambda}$  for every pixel as the relation between the points  $p_{\lambda_{ref}}$  and  $p_{\lambda}$ , in terms of the magnitude and direction of dispersion in the captured image. For this, we leverage our refraction function  $\Psi_{\lambda}$ , relating the position of  $p_d$  (from the direct ray through  $o$ ) to the captured  $p_{\lambda}$  (see Equation (4)). We first select a reference wavelength  $\lambda_{ref}=550$  nm for our measurements. Using our refraction function  $\Psi_{\lambda_{ref}}$ , we obtain pixel  $p_d$  corresponding to the reference pixel  $p_{\lambda_{ref}}$ , as a function of depth  $z$ :  $p_d = \Psi_{\lambda_{ref}}(z; p_{\lambda_{ref}})$ .

Then, for any other target wavelength  $\lambda$ , we calculate the inverse refraction function  $\Psi_{\lambda}^{-1}$  mapping from  $p_d$  to  $p_{\lambda}$ :  $p_{\lambda} = \Psi_{\lambda}^{-1}(z; p_d)$ ; Figure 5 illustrates this process<sup>1</sup>. Finally, we obtain our dispersion function  $\Phi_{\lambda}$ , which encodes the magnitude and direction of dispersion, and thus enables us to locate any pixel  $p_{\lambda}$  from the reference-wavelength pixel  $p_{\lambda_{ref}}$  as a function of depth  $z$ :

$$p_{\lambda} = \Phi_{\lambda}(z; p_{\lambda_{ref}}) \equiv \Psi_{\lambda_{ref}}^{-1}(z; \Psi_{\lambda_{ref}}(z; p_{\lambda_{ref}})). \quad (5)$$

#### 4.3 Analysis

Figure 6(a) shows an example of per-wavelength changes in dispersion magnitude with varying depth. In particular we measure the displacement from  $p_{700\text{ nm}}$  to  $p_{\lambda}$ , at a center pixel. We observe that the magnitude of dispersion increases rapidly with depth  $z$  up to approximately 350 mm, then converges after approximately 700 mm. Figure 6(b) shows the resulting spatially-varying dispersion vector field for all sensor pixels, with the image plane at  $z=700$  mm, using Equation (5). The arrows indicate main directions of dispersion per pixel, while color codes magnitude. Note that dispersion is more pronounced to the left due to the varying thickness of the prism. Other prisms, such as an Amici prism, can be used to mitigate this effect.

In practice, since a DSLR camera with an ordinary lens can capture information farther than 350 mm, we can simplify our dispersion model removing its depth-dependency for larger distances; the resulting matrix  $\Phi$  at  $z=700$  mm can then be applied in general situations where the scene is at  $z>700$  mm (see Section 5 for more

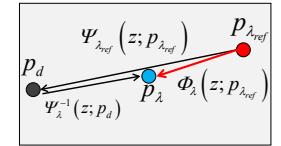


Fig. 5. We derive our dispersion function  $\Phi_{\lambda}$  (red arrow) by using our refraction function  $\Psi_{\lambda_{ref}}$  from a reference point  $p_{\lambda_{ref}}$  to obtain  $p_d$ , then computing the inverse function from  $p_d$  to any other point  $p_{\lambda}$ .

<sup>1</sup>We obtain the inverse function  $\Psi_{\lambda}^{-1}$  by interpolation with Delaunay triangulation.

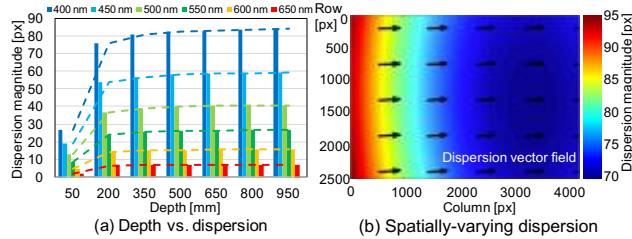


Fig. 6. (a) Dispersion magnitude changes with depth, as our simulations show. For all wavelengths, this magnitude increases rapidly up to  $\sim 200$  mm, being approximately constant at around  $\sim 700$  mm. (b) An example of the dispersion magnitude and direction of every sensor pixel, at a distance of 700 mm.

details). This is important, since it allows us to reconstruct spectral information without the need for depth estimation.

## 5 SPECTRUM RECONSTRUCTION

Given our image formation model (Section 3) and our spatially-varying dispersion model (Section 4), we now describe how to reconstruct a hyperspectral image  $\mathbf{i}$  from a dispersed RGB image  $\mathbf{j}$ . The spectrum-to-RGB operator  $\Omega$  and the dispersion model  $\Phi$  for every pixel in Equation (2) are obtained from the calibration of the camera system (described in Section 6).

Previous reconstruction methods, used in both CASSI and CTIS approaches, rely on rich spectral cues that exist over all the input image. In CASSI, using a coded mask and collimation, dispersion patterns are obtained independently of image structures [Kim et al. 2012a; Wagadarikar et al. 2008]. In CTIS, several dispersed images along different directions are captured, in addition to a sharp image, at the cost of resolution loss [Habel et al. 2012; Johnson et al. 2007]. Different from these systems, we only use a prism with a DSLR camera. As a result, our input is a single RGB image with *overlapping* dispersion information, and *sparse* spectral signatures only at the edges (Figure 7(a)). Thus, our reconstruction problem is more severely ill-posed. Simply adopting existing reconstruction algorithms results in poor spectral accuracy, as Figure 18 shows.

We thus develop a novel reconstruction framework for compact single-shot hyperspectral imaging. Our reconstruction consists of three main stages: First, an edge restoration stage aligns an input dispersed RGB image, in order to obtain clear edge information without dispersion. Second, we estimate spectral information in the gradient domain using dispersion over the extracted edges. Last, we recover the hyperspectral image by using the sparse spectral information of gradients.

### 5.1 Restoring Edges from Dispersion

Restoring accurate edge information is critical for our reconstruction algorithm, since edges are the main source to analyze dispersion. We estimate the spatially-aligned hyperspectral image  $\mathbf{i}_{\text{aligned}} \in \mathbb{R}^{XY\Lambda \times 1}$  from an input dispersed RGB image  $\mathbf{j}$ , by solving the following convex optimization problem:

$$\mathbf{i}_{\text{aligned}} = \arg \min_{\mathbf{i}} \underbrace{\|\Omega \Phi \mathbf{i} - \mathbf{j}\|_2^2}_{\text{data term}} + \underbrace{\alpha_1 \|\nabla_{xy} \mathbf{i}\|_1 + \beta_1 \|\nabla_\lambda \nabla_{xy} \mathbf{i}\|_1}_{\text{prior terms}} \quad (6)$$

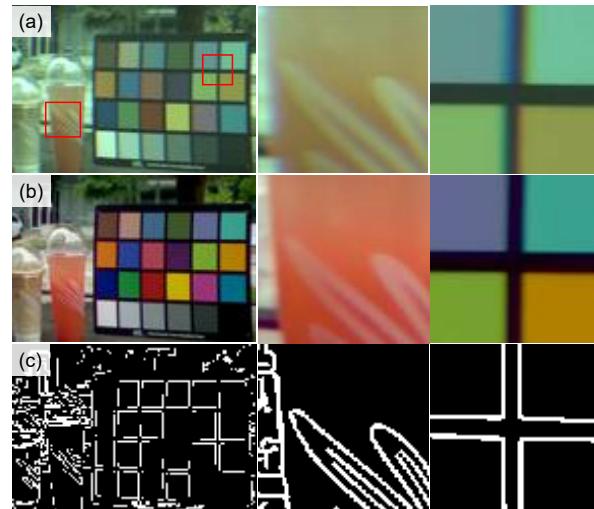


Fig. 7. (a) A RAW RGB image [ $\mathbf{j}$  in Equation (6)] with dispersion (gamma corrected for visualization). (b) A recovered sRGB image with edge restoration. (c) Detected edges from the restored image without dispersion.

where  $\nabla_{xy}$  is a spatial gradient operator, and  $\nabla_\lambda$  is a spectral gradient operator. Specifically, the first term describes the data residual of our image formation model (Equation (2)), while the other terms are priors. The first prior is a traditional total variation (TV) term, ensuring sparsity of spatial gradients, and removal of spatial artifacts. We introduce a second prior, our modified cross-channel term, penalizing edge misalignment across spectral channels by pursuing sparseness in the change of spatial gradients. Note that while the original cross-channel prior of Heide et al. [2013] computes the difference between normalized gradient values for every pair of color channels, we compute the difference between *unnormalized* gradient values of *adjacent* spectral channels, assuming that spectral signals are locally smooth in adjacent channels. Our cross-channel prior thus enables us to achieve spatial alignment between spectral channels. We solve Equation (6) with the alternating direction method of multipliers (ADMM) [Afonso et al. 2011]. Refer to the supplemental material for more details on the optimization.

Our spatial alignment stage yields a hyperspectral image without edge dispersion (see Figure 7(b)). However, the accuracy of the spectral information in  $\mathbf{i}_{\text{aligned}}$  is still incomplete, and therefore unreliable. To accurately locate edges, instead of applying an edge detection algorithm directly on the aligned spectral channels in  $\mathbf{i}_{\text{aligned}}$ , we first project the aligned spectral image  $\mathbf{i}_{\text{aligned}}$  onto RGB channels via the camera response functions  $\Omega \mathbf{i}_{\text{aligned}}$ , then apply a multi-scale edge detector [Dollár and Zitnick 2015]. This achieves more robust edge detection results, as shown in Figure 7(c). This extracted edge information will then be used to reconstruct spectral information from dispersion over edges in the next stage.

### 5.2 Reconstructing Spectral Information

The **main insight** of our spectral reconstruction algorithm is based on the observation that dispersion over edges can be used as spectral reconstruction cues, since it relates *spectral* information with *spatial* blur (Figure 8(a)). In particular, we account for spatial gradients in the dispersed regions around edges to reconstruct a complete

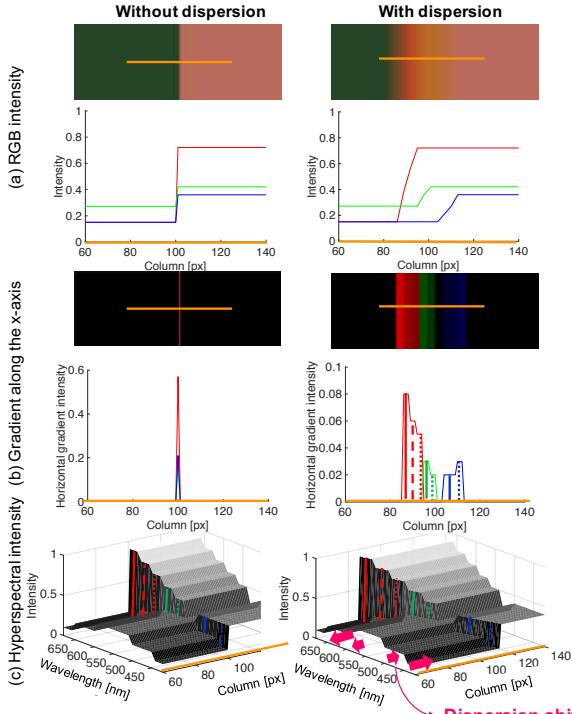


Fig. 8. (a) Left: image without dispersion; right: the same image with dispersion. The orange lines refer to the plots of pixel intensities below. (b) X-axis gradient images, without and with dispersion. With dispersion, the spatial gradients reveal the spectral components. (c) Hyperspectral information around edges. The marginal projection of the spectrum along the wavelength axis yields the RGB intensities through the camera functions, as shown in (a).

hyperspectral image. With a dispersed image with overlapping spectral information as input, as opposed to the rich per-pixel spectral information provided by a coded mask, the  $\|\Omega\Phi\mathbf{i} - \mathbf{j}\|_2^2$  data term in Equation (6) is insufficient to reconstruct actual spectral intensities. Since our spatially-varying dispersion model  $\Phi$  describes the relationship between spectral information and dispersed spatial information (Section 4), we can leverage *spatial gradients* of the dispersed image as input to reconstruct spectral intensity around edges.

Figure 8 illustrates an example. The left column shows pixel intensities (a), horizontal gradients (b), and its hyperspectral representation (c), along a row in an image without dispersion (orange line). The right column presents the same information, but with dispersion through a prism. The spectral information in the dispersion-free image is projected directly into RGB values, so we cannot trace back metameristic spectra from the given input. However, in the dispersed image, even though the spectral information appears as *blur* (which is a projection of sheared spectral information), its spatial gradients reveal information about spectral power distributions along edges. As shown in the hyperspectral representation (c) on the right column, different wavelengths are sheared by a different magnitude and direction due to dispersion. Marginal projection of these dispersed spectra to RGB values along the wavelength axis still

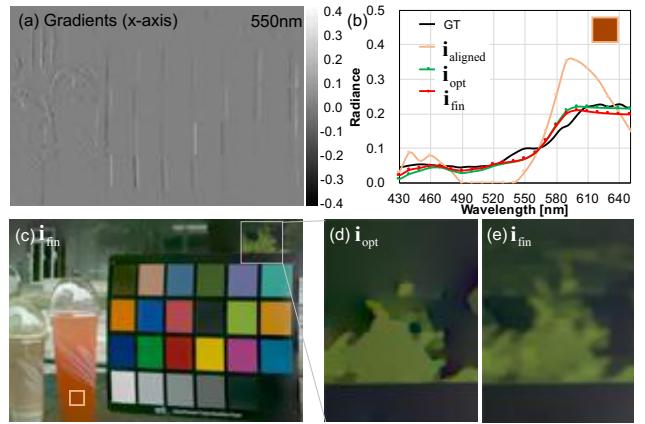


Fig. 9. (a) Reconstructed horizontal gradients  $\mathbf{g}_x$  for a wavelength of 550 nm. (b) Spectral power distributions of the orange patch for each stage of our method, compared to the ground truth. (c) Final reconstructed sRGB image. (d) and (e) Comparison between the initial reconstruction  $\mathbf{i}_{\text{opt}}$  (d), and after detail enhancement  $\mathbf{i}_{\text{fin}}$  (e).

leaves cues for spectral reconstruction. We can clearly see the spectral cues in the gradient domain rather than the intensity domain; each gradient profile of a pixel contains the gradient information of a specific spectral channel depending on the dispersion predicted by our model  $\Phi$ . This motivates us to formulate our reconstruction problem in the *spatial gradient* domain.

**5.2.1 Gradient-based Reconstruction.** We first estimate a stack of spatial gradients  $\hat{\mathbf{g}}_{xy}$  for each wavelength, by finding out the spatial gradients  $\mathbf{g}_{xy}$  that are close to the spatial gradients of the captured image  $\nabla_{xy}\mathbf{j}$ , formulating the gradient reconstruction problem as:

$$\hat{\mathbf{g}}_{xy} = \arg \min_{\mathbf{g}_{xy}} \underbrace{\|\Omega\Phi\mathbf{g}_{xy} - \nabla_{xy}\mathbf{j}\|_2^2}_{\text{data term}} + \underbrace{\alpha_2 \|\nabla_\lambda \mathbf{g}_{xy}\|_1 + \beta_2 \|\nabla_{xy} \mathbf{g}_{xy}\|_2^2}_{\text{prior terms}}, \quad (7)$$

where the first term is a data term describing our image formation model (Equation (1)) in the gradient domain. The following two terms are prior terms for gradients. The first prior is equivalent to the spectral sparsity of gradients used in the spatial alignment stage (Equation (6)), enforcing sparse changes of gradients along the spectral dimension. The second prior imposes smooth changes of gradients in the spatial domain, which reduces artifacts.

Given that the spectral signature exists solely over edges, we only take into account edge pixels for the optimization in Equation (7), which we solve using ADMM. Refer to the supplemental material for the details of our edge-based objective reformulation and optimization. Figure 9(a) shows the estimated horizontal gradient image for a wavelength of 550 nm extracted from the dispersion over edges.

**5.2.2 Reconstructing the Spectral Image.** After obtaining from dispersion a stack of spatial gradients  $\hat{\mathbf{g}}_{xy}$  for each wavelength, we utilize gradient information as strong spectral cues to reconstruct

the hyperspectral image  $\mathbf{i}_{\text{opt}}$ . Since we do not have any known hyperspectral intensity, we cannot directly apply Poisson reconstruction [Pérez et al. 2003] with boundary conditions. Instead, we formulate the following optimization problem to estimate  $\mathbf{i}_{\text{opt}}$  from  $\hat{\mathbf{g}}_{xy}$ :

$$\mathbf{i}_{\text{opt}} = \arg \min_{\mathbf{i}} \underbrace{\|\Omega \Phi \mathbf{i} - \mathbf{j}\|_2^2 + \alpha_3 \|\mathbf{W}_{xy} \odot (\nabla_{xy} \mathbf{i} - \hat{\mathbf{g}}_{xy})\|_2^2}_{\text{data terms}} + \underbrace{\beta_3 \|\Delta_\lambda \mathbf{i}\|_2^2}_{\text{prior term}}, \quad (8)$$

where  $\Delta_\lambda$  is a Laplacian operator for the spectral image  $\mathbf{i}$  along the spectral axis, and  $\mathbf{W}_{xy}$  is an element-wise weighting matrix that determines the level of confidence of the estimated gradients in the previous step. To take into account the directional dependency of spectral cues, we build our confidence matrix  $\mathbf{W}_{xy}$  based on the extracted edge information, and the dispersion direction  $\mathbf{n} = [n_x, n_y]$ . For non-edge pixels, we assign a high confidence for zero gradient values. For edge pixels, we assign different confidence levels for horizontal and vertical components respectively, so that gradient directions similar to the dispersion direction have a high confidence value. Specifically, a confidence value,  $W_{k \in \{x,y\}}(p, \lambda)$ , is an element of the matrix  $\mathbf{W}_{xy}$  for the horizontal and vertical gradient components of a pixel  $p$  of wavelength  $\lambda$ :

$$W_{k \in \{x,y\}}(p, \lambda) = \begin{cases} |n_{k \in \{x,y\}}| & \text{if } p \text{ is an edge pixel} \\ 1 & \text{otherwise} \end{cases} \quad (9)$$

where  $|\cdot|$  denotes absolute value.

The first data term minimizes errors in our image formation model (Equation (2)). The second data term minimizes differences between the gradients  $\nabla_{xy} \mathbf{i}$ , and the gradients  $\hat{\mathbf{g}}_{xy}$  from the previous stage. The prior term in Equation (8) favors spectral curvature smoothness  $\|\Delta_\lambda \mathbf{i}\|_2^2$ . Note that we do not optimize any spectral gradients along the wavelength axis. Instead, we improve the stability of our spectral estimation by accounting for such curvature smoothness along the different wavelengths. Since Equation (8) consists only of  $l_2$ -norm terms, we solve it using a conjugate gradient method.

**Detail-Guided Filtering.** While the solution from Equation (8) yields high spectral accuracy, the lack of gradient information on smooth surfaces may lead to loss of spatial details (see Figure 9(b)). To restore these details, we apply a guided image filter [He et al. 2013] for detail enhancement. It allows us to enhance structural details of the aligned image  $\Omega \mathbf{i}_{\text{aligned}}$  to each reconstructed hyperspectral channel, resulting in the detail-enhanced hyperspectral channel. This leads to our final reconstructed image  $\mathbf{i}_{\text{fin}}$ , with high spectral and spatial accuracy (Figure 9(c)).

## 6 CALIBRATION

To reconstruct the spectral information from dispersion, we need to obtain the camera's intrinsic parameters, its radiometric response function  $\Omega$  (spectrum-to-RGB operator), and the spatially-varying dispersion matrix  $\Phi$  of the prism. We calibrate the intrinsic properties of the camera, which include the focal length  $f$ , its optical center  $o$ , and distortion coefficients, using Zhang's method [2000]. In the following, we describe how to calibrate  $\Omega$  and  $\Phi$ .

### 6.1 Radiometric Response of the Camera

To calibrate the spectral response functions of a camera's RGB channels, we illuminate a standard reflectance tile (Spectralon) with a solid-state plasma light (Thorlab HPLS-30-4). The reflected energy of each wavelength  $R(\lambda)$  is

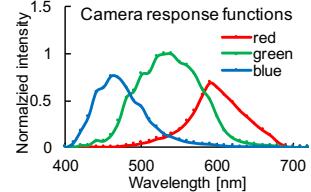


Fig. 10. Measured camera response functions (Canon 5D Mark III).

then measured by a spectroradiometer (Jeti Specbos 1200) from 400 nm to 700 nm, in 10 nm intervals. At the same time, we capture band-wise spectral images filtered by an electronic bandpass filter (VariSpec VIS LCTF) using a DSLR camera (Canon EOS 5D Mark III) with a 50 mm lens (to avoid under- or over-exposure, we capture raw images with five different exposures). As a result, we obtain the spectral response of the camera for each band, as a trichromatic response function  $\Omega \in \mathbb{R}^{3 \times \Lambda}$  in matrix form (Figure 10). Note that if these calibration instruments are not available, we can instead use publicly available datasets of the spectral sensitivity functions of other DSLR cameras [Jiang et al. 2013].

### 6.2 Spatially-varying Dispersion of the Prism

This calibration is based on the image formation model described in Section 4, and consists of three steps: (a) obtaining the extrinsic position and orientation of the prism with respect to a camera; (b) obtaining our refraction model  $\Psi$  (Section 4.1); and (c) obtaining our dispersion model  $\Phi$  (Section 4.2).

**Setup.** Figure 11(a) depicts our calibration setup. We insert a bandpass filter between a prism and a checkerboard target, to isolate dispersion per wavelength. We use five 10 nm bandpass filters from 450 nm to 650 nm in 50 nm intervals<sup>2</sup>. For each captured spectral image of wavelength  $\lambda$ , we obtain shifted positions  $p_\lambda$  of the corner pixels in the checkerboard (Figure 11(b)), and corresponding direct-ray positions  $p_d$  without the prism at a known distance  $z$ . Note that the camera calibration method [Zhang 2000] yields not only the intrinsic camera matrix, but also three-dimensional coordinates of feature points at corners. In the end, we use a set of positions  $p_\lambda$ ,  $p_d$  and depth  $z$  of each checkerboard corner for prism calibration.

**Prism Parameters.** We use an Edmund Littrow dispersion prism made of N-BK7 with 30-60-90 angles to disperse light. Figure 12 shows its transmission and refractive indices per wavelength. In addition to these intrinsic parameters, we define its extrinsic parameters (position and orientation) as a six-dimensional vector  $\xi$  that includes the parameters,  $s_x, s_y, s_z, v_x, v_y$ , and  $v_z$  in Equation (3). We therefore can rewrite our refraction function (Equation (4)) as  $p_d = \Psi_\lambda(p_\lambda, z; \xi)$ . Using the refraction function  $\Psi_\lambda$  and the captured positions  $p_\lambda$ ,  $p_d$  and depth  $z$ , we can estimate  $\xi$  by nonlinear optimization of the following objective function [Waltz et al. 2006]:

$$\min_{\xi} \sum_{\lambda \in \Gamma} \sum_{\{p_\lambda, p_d\} \in \Pi_\lambda} \|p_d - \Psi_\lambda(p_\lambda, z; \xi)\|_2^2, \quad (10)$$

where  $\Gamma$  is a subset of the wavelengths (450 nm, 500 nm, 550 nm, 600 nm, and 650 nm) used for calibration in our experiment, and  $\Pi_\lambda$

<sup>2</sup>We employ bandpass filters instead of a LCTF, since the LCTF alters the light path through optical components inside the filter.

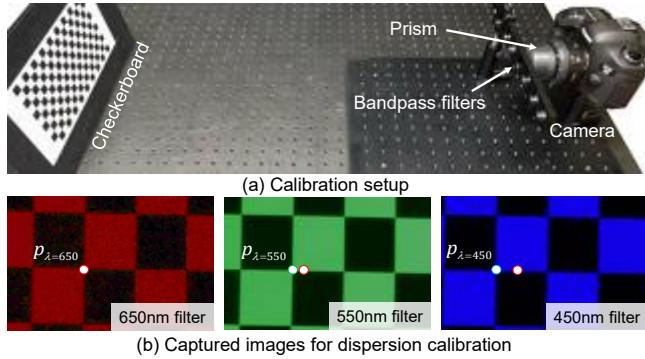


Fig. 11. (a) Our calibration setup to estimate spatially-varying dispersion. (b) Captured spectral images present clear wavelength-dependent shifts. For a corner pixel on an image at 650 nm, corresponding pixels at 550 nm and 450 nm are shifted by 21 px and 50 px, respectively. We calibrate the prism parameters  $\xi$  from the corresponding corners on the checkerboard using Equation (10).

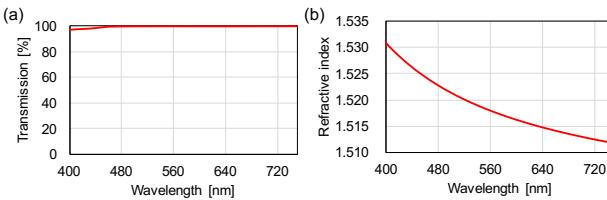


Fig. 12. (a) Light transmission of our dispersion prism (N-BK7, measured at 30-60-90 angles). (b) Refractive indices.

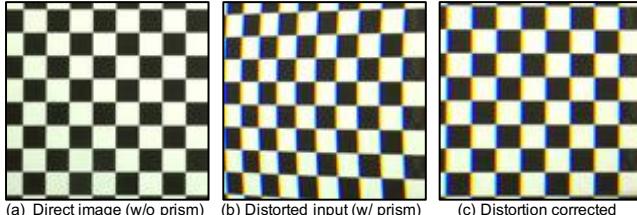


Fig. 13. (a) Ground-truth direct image of a checkerboard captured without a prism. (b) Captured image with a prism, resulting in geometric distortion. (c) Distortion-corrected image using our refraction model.

is a set of corresponding pairs  $p_d$  and  $p_\lambda$  at a depth  $z$ . This yields highly accurate prism parameters; the average reprojection error from  $p_\lambda$  to  $p_d$  is  $\sim 0.51$  pixel in our implementation.

**Refraction Model.** Once we know our prism parameters  $\xi$  and our refraction function  $\Psi_\lambda$  (Equation (4)), we can build a refraction model  $\Psi$  in matrix form, describing per-pixel relations between a refracted pixel  $p_\lambda$  and its corresponding direct-ray pixel  $p_d$ , per wavelength (see Section 4.1). The refraction model  $\Psi$  allows us not only to calculate the dispersion model  $\Phi$  for every pixel, but also to correct geometric distortions caused by refraction (see Figure 13). Figure 13(a) shows the captured checkerboard without a prism. Once we have a prism in front of the camera, refraction through the prism introduces geometric distortion in addition to dispersion, as shown in Figure 13(b). We can correct this geometric distortion using the refraction model  $\Psi$ , which warps the captured refracted image into a direct-ray image (without the prism). Note that we use an inverse

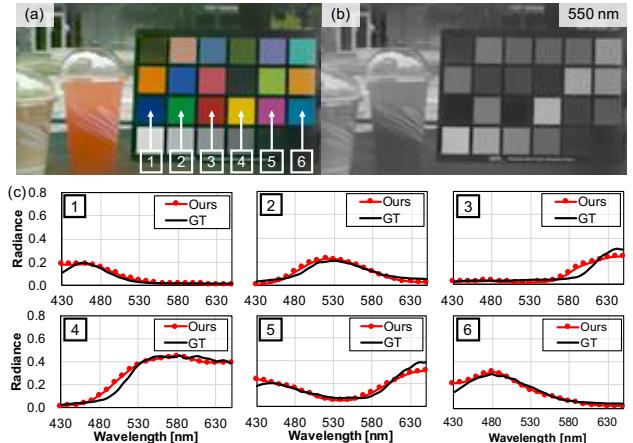


Fig. 14. Quantitative evaluation of our compact hyperspectral imaging on a real-world scene. (a) Reconstructed hyperspectral image with geometric-distortion correction (displayed as an sRGB image). (b) Reconstructed spectral channel of 550 nm. (c) Spectral plots of six patches in the ColorChecker. Refer to the supplemental material for complete plots of every patch.

mapping of the refraction model in order to avoid holes similar to inverse texture mapping.

**Dispersion Model.** We define our dispersion function  $\Phi_\lambda$  (Equation (5)) using the refraction function  $\Psi_\lambda$  (Equation (4)). Therefore, once we have obtained our refraction model  $\Psi$ , we can compute a dispersion model  $\Phi$  for every pixel; we warp a pixel  $p_{\lambda,ref}$  to a pixel  $p_d$  via the refraction model  $\Psi$ , then warp the pixel  $p_d$  to a pixel  $p_\lambda$  via its inverse model  $\Psi^{-1}$  (see Figure 5 in Section 4.2). The dispersion model  $\Phi$  then relates a pixel  $p_{\lambda,ref}$  to a pixel  $p_\lambda$ . Note that we had previously fixed a distance (700 mm) for the dispersion model where dispersion does not depend on depth (see Figure 6).

## 7 RESULTS AND VALIDATION

We validate the performance of our method by using both real data, and synthetic data with ground truth. As shown in Figure 1(a), we have built a prototype with a Canon 5D Mark III camera with a 50 mm lens (Canon EF f/1.8 STM). We position a 30-60-90 prism in front of the lens by installing it in a supporting structure made by a 3D printer. The dispersion model and camera response functions are calibrated as described in Section 6. For the synthetic dataset to be used for ground-truth evaluation, we use a public hyperspectral dataset [Yasuma et al. 2010], and simulate dispersion with approximately one pixel shift per 10 nm wavelength (by taking into account our optical component parameters, camera response functions, and based on the refraction phenomenon as described in Section 4).

We reconstruct the hyperspectral images on an Intel i7-3770 CPU 3.40 GHz with 32 GB of memory. Our MATLAB implementation takes approximately 45 minutes in total to process a 512-by-512 hyperspectral image, with 23 wavelengths from 430 nm to 650 nm, in 10 nm intervals. In particular, our edge restoration takes about 9 minutes, the gradient-based optimization about 4 minutes, and the final hyperspectral image reconstruction about 30 minutes.

For the real scenes, from the captured raw image of 3870-by-5796 resolution, we first discard boundary pixels due to occlusion by

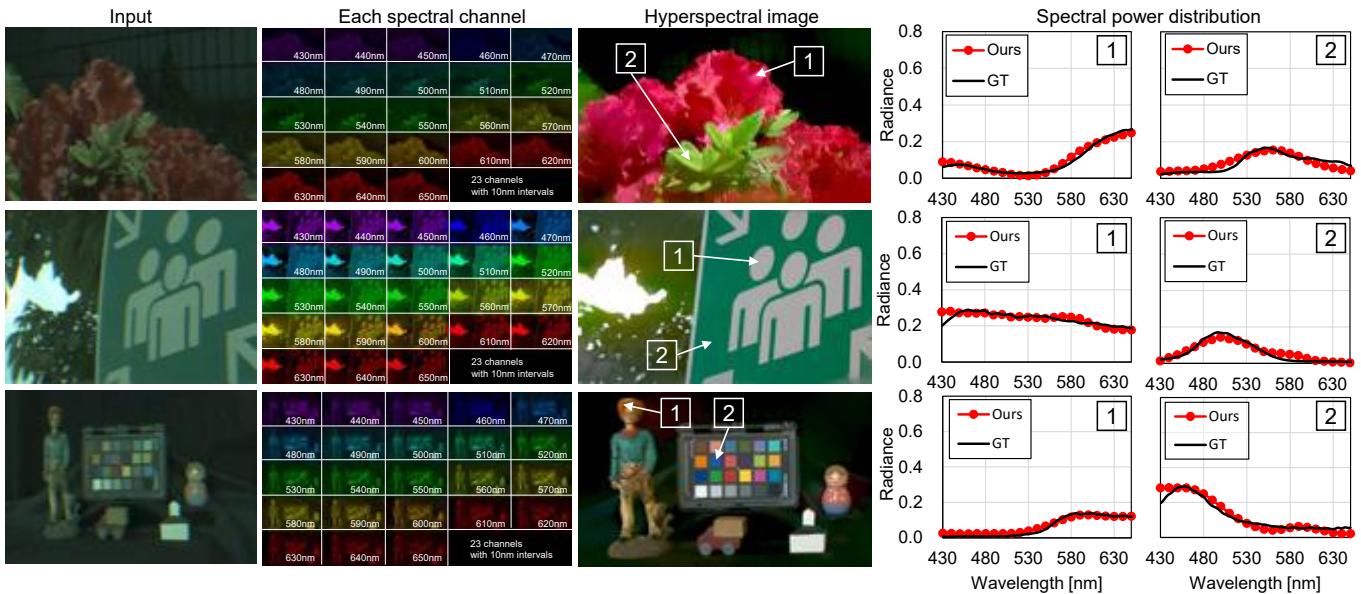


Fig. 15. Reconstructed hyperspectral images of real-world, casual scenes. For each dispersed RAW-RGB input, we show a reconstructed hyperspectral image for each wavelength, displayed as an sRGB image. The spectral plots compare our reconstruction with the ground truth, measured by a spectroradiometer for the patches shown.

the prism holder, resulting in 3300-by-4800 resolution. Then, we downscale the captured high-resolution image to 600-by-860 with bicubic interpolation for computational efficiency. We empirically set the reconstruction parameters (Section 5) as:  $\alpha_1 = 10^{-5}$ ,  $\beta_1 = 10^{-1}$ ,  $\alpha_2 = 10^{-4}$ ,  $\beta_2 = 0.5 \times 10^{-3}$ ,  $\alpha_3 = 0.5 \times 10^2$ , and  $\beta_3 = 0.5 \times 10^{-1}$ .

**Spectral Accuracy.** In order to evaluate the performance of our system, we capture a scene with a ColorChecker under daylight. As shown in Figure 14 and the supplemental material, our results closely match the spectral power distributions of every patch, as measured by a spectroradiometer.

**Casual Hyperspectral Imaging.** Our compact system enables casual hyperspectral imaging of indoor and outdoor scenes, as shown in Figures 1, 2, 14, 21 and 19. Figure 15 shows additional results for three different scenes. For each input RAW image with dispersion, we present the reconstructed spectral channels, as well as the reconstructed hyperspectral image (displayed as an sRGB image). Comparison with the ground truth data measured by a spectroradiometer is also presented.

**Comparison with Existing Hyperspectral Imaging Systems.** Figure 16 compares our system with other conventional hyperspectral imaging systems (CASSI [Kim et al. 2012a], CTIS [Habel et al. 2012] and PMVIS [Cao et al. 2011a]). They all are significantly larger and more expensive than our system, since they rely on a coded mask or collimating optics.

We use simulated data based on the image formation model of each system. The top row shows the different systems, and the second row the input. The third row shows the corresponding reconstructions. For the conventional CASSI system, we simulated a coded mask to embed spectral signatures using a monochromatic camera. The results from CASSI show high spectral accuracy at

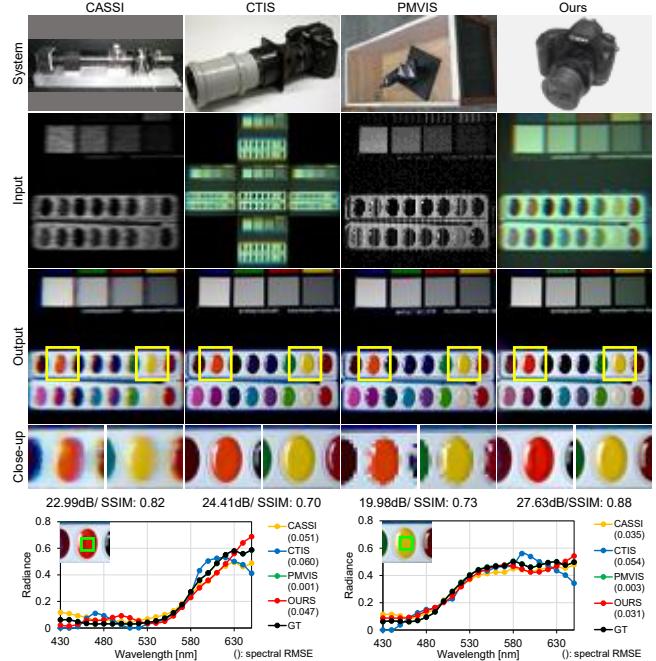


Fig. 16. We compare our system with other conventional hyperspectral imaging systems: CASSI [Kim et al. 2012a], CTIS [Habel et al. 2012], and PMVIS [Cao et al. 2011a]. We evaluate image quality and spectral accuracy by simulating the image formation model of each system. Since they rely on a coded mask or collimating optics, they all are significantly larger and more expensive than our system. The combined spectral and spatial accuracy of our method is superior to the other systems, even though our system takes a single dispersed RGB image as an input with the simplest setup.

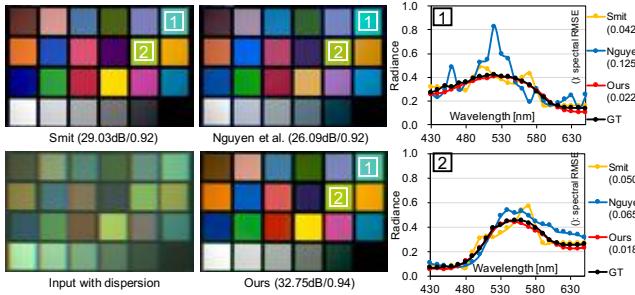


Fig. 17. Comparison with other RGB-to-spectrum methods ([Smit 1999] and [Nguyen et al. 2014]). We show the reconstructed sRGB image of each method, along with the average PSNR and SSIM values, and plot normalized radiance for two color patches on the right, including RMSE values. Note that we provide an RGB image *without* dispersion as input for Smit [1999] and Nguyen et al. [2014].

the cost of poor spatial resolution, due to the coded mask pattern carved in the input image. CTIS uses a diffraction grating to obtain multiple dispersed RGB images along with a clear RGB image, thus sacrificing spatial resolution. The system shows consistent and moderately accurate performance; however, it utilizes only about 10% of the total image sensor area to capture input dispersion, resulting in clear stair-step artifacts. In PMVIS, the system projects each wavelength on a single pixel by using a dispersive unit and a large-sized mask on a box, so the user can simply extract the pixel to obtain a particular spectrum. In terms of spectral accuracy, PMVIS produces very accurate results. However, it is the largest system, and the spatial resolution is significantly reduced, due to the sub-sampling mask. In comparison, our system takes a single dispersed RGB image as an input with the simplest and most compact setup, while the combined spectral and spatial accuracy is superior to the other systems.

**Comparison with RGB-to-Spectrum Methods.** We compare our method with other RGB-to-spectrum methods from a single RGB image. Smit [1999] regresses a spectrum from the RGB input as a combination of camera response basis functions. Nguyen et al. [2014] introduced a data-driven function that converts RGB to spectrum by training a radial basis function network with ground truth hyperspectral data. As shown in Figure 17, our method yields superior results in terms of spectral accuracy.

**Spectral Smoothness Priors.** We employ spectral smoothness priors in our reconstruction (the second and the third prior in Equations (7) and (8), respectively). We explicitly tested impacts of the priors for the ColorChecker scene (Figure 17) by testing results without the priors in our reconstruction formulation. The average PSNR drops about 8% from 32.75dB to 30.17dB.

**Comparison with Inverse System Solvers.** Inverse solvers such as TwIST [Dias and Figueiredo 2007] or TVAL3 [Li 2011] have been commonly used for reconstructing hyperspectral images in traditional systems. In principle, these solvers could also be used in our method, by formulating our reconstruction problem ( $\mathbf{j} = \Omega\Phi\mathbf{i}$  in Equation (2)) as an inverse problem. However, since those solvers rely heavily on rich spectral signatures all over the observed image

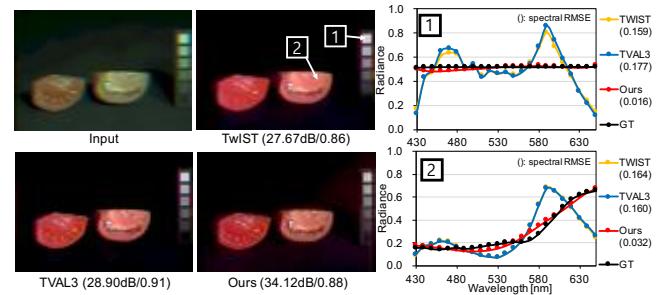


Fig. 18. Comparison with other inverse system solvers (TwIST and TVAL3). All the methods are given a dispersed RGB image as shown in the upper-left. The sRGB images projected from each hyperspectral estimate and their average PSNR and SSIM values are presented. TwIST and TVAL3 fail to achieve high spectral accuracy due to the sparseness of spectral cues and ambiguity of metameristic spectra, while our method closely matches the ground truth.

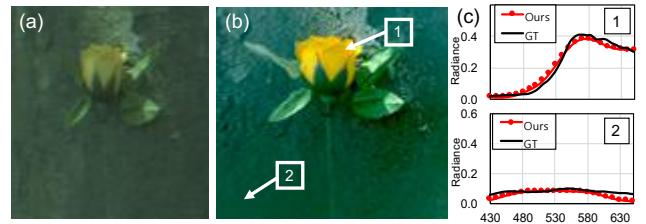


Fig. 19. If large areas of the input image (a) lack sufficient edge information, our reconstruction may blur details around them (b). Spectral accuracy could be degraded due to missing edges (c).

(achieved through additional optical elements), they fail to differentiate metameristic spectra due to the sparse spectral cues in our input (see Figure 18). In contrast, our method avoids the complexity of such inverse problem by working in two different domains: aligning the input in the spatial domain, and reconstructing spectral information in the gradient domain. This strategy allows us to exploit sparse cues more effectively, leading to higher spatial and spectral accuracy.

## 8 LIMITATIONS AND DISCUSSION

In this section, we further evaluate our system in the presence of suboptimal input. Refer to the supplemental material for more results.

**Edge Frequency.** Our reconstruction quality depends on the edge frequency of an input image. If a scene has not enough edge information, reconstruction quality degrades as shown in Figure 19. Additionally, we have examined the impact of increasingly higher frequency edges, as shown in Figure 20. Our reconstruction remains relatively stable as the frequency increases; however, performance degrades significantly when the dispersed edges overlap (the space between edges becomes less than the half of the dispersion amount).

**Edge Blurriness.** To examine the impact of blurred edges, we generate a synthetic blurred ColorChecker (Gaussian convolution with  $\sigma=1$ ). The accuracy of the reconstructed spectrum (PSNR) is 30.50dB, compared to the original 32.75dB shown in Figure 17.

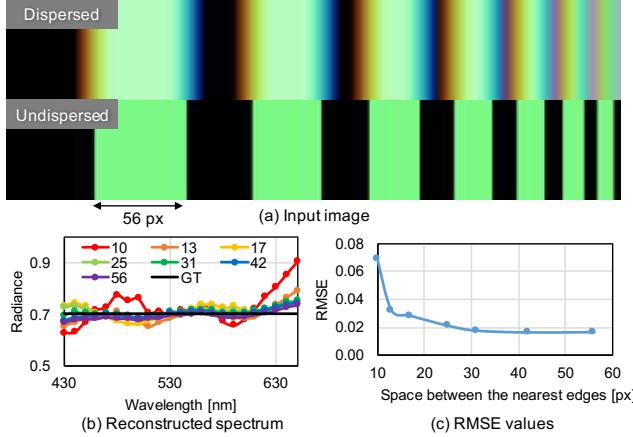


Fig. 20. Evaluation with varying edge frequency. (a) Synthetic RAW images of seven bright stripes with different spacing between edges; 56, 42, 31, 25, 17, 13 and 10 in pixels. We set the amount of dispersion to 23 pixels from 430 nm to 650 nm (10 nm per band). (b) Reconstructed spectrum for each stripe, compared with ground truth. (c) Our reconstruction accuracy degrades gradually (RMSE increases) as the space between edges decreases. When the space becomes smaller than half of the dispersion magnitude (in this example,  $23/2 = 11.5$  px), reconstruction quality degrades severely.

**Illumination Environments.** We tested our system under different illumination environments other than daylight: halogen, and solid-state plasma illumination. Figure 21 shows the two corresponding reconstruction results. While halogen illumination is spectrally smooth, plasma light has many high-frequency changes which violate our assumption of spectral smoothness. Although our method fails to recover these high-frequency spectral changes from the plasma illumination, we can still approximate its main low-frequency components.

**Fluorescent Lighting.** While our method performs robustly under most illuminations, its performance under the strong-peak illumination of fluorescent light decreases. We have tested this by simulating a synthetic illumination with three strong peaks (red, green and blue), and reconstructing the spectra of 24 color patches for each one. Compared to our results in Figure 17 (using the standard D65 illuminant), the accuracy of our reconstruction decreases about 15%, from 32.75dB (PSNR) to 27.85dB.

**Dependency on the RGB Sensor.** Since we target casual spectral imaging using a regular DSLR camera, we presume an RGB-filtered camera by default; this broadens the applicability of our method, although in turn, it makes it dependent on the particular transmission functions of the sensor’s RGB channels. Our approach leverages RGB channel separation, allowing us to locate edges accurately by using our cross-channel prior (third term in Equation (6)). We confirm this by testing our method simulating a monochrome sensor with a ColorChecker target: in the absence of separate RGB information, our edge-restoration stage yields visibly worse results (average PSNR drops 33.4%, from 32.75dB to 21.81dB).

**Influence of Noise.** Since our reconstruction method operates in the gradient domain, it is sensitive to sensor noise. To minimize this, we consistently use a low ISO speed (ISO 100). Moreover, we

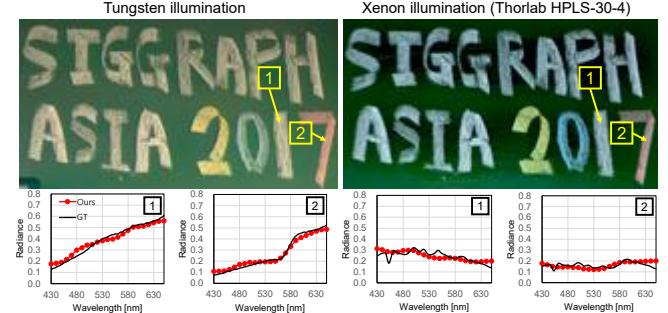


Fig. 21. Reconstructed hyperspectral images under different illuminations of halogen and solid-state plasma lights. The top row presents the reconstructed sRGB images for the two different light sources without white balancing. The bottom row shows the spectral plot of two sample patches of different color chalks, compared with ground truth data measured by a spectroradiometer. While we can capture the low-frequency tendency of the spectrum for the halogen case, high-frequency details introduced by the plasma illumination cannot be captured.

introduce TV-terms (L1-norm of spatial gradients) in Equations (6) and (7) to handle noise when restoring edges and reconstructing the spectrum from spatial-spectral gradients.

**Depth from Dispersion.** Since dispersion increases proportionally with depth within a certain range, as shown in Figure 6(a), it should be possible to estimate depth by analyzing the amount of dispersion, similar to depth from refraction [Baek et al. 2016]. However, this relationship between dispersion and depth only holds for close distances within 200 mm, while an ordinary lens cannot capture sharp images at such a short distance, and a macro lens has an extremely shallow depth-of-field. Based on these observations, developing a single-shot depth-from-dispersion algorithm remains an interesting avenue of future work.

## 9 CONCLUSION

We have presented a novel compact hyperspectral imaging system, requiring only a conventional DSLR camera and a prism, thus enabling low-budget hyperspectral photography. In addition, we have provided a novel image formation model of dispersion, and a novel calibration method based on our formulation. We have demonstrated results on various natural scenes; the combined spatial and spectral accuracy of our technique is superior to previous approaches, which use more complex setups and may cost orders of magnitude more.

As we have seen, our system is sensitive to noise, and its performance may drop depending on the edge properties of the scene, or its illumination. Addressing these issues, and deriving a complete depth-from-dispersion model, are interesting avenues for future work.

## ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insights as well as Daniel S. Jeon, Yeong Beum Lee, Giljoo Nam, Inchang Choi, Joo Ho Lee, Adrian Jarabo, and Belen Masia for helpful discussions 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,

Sports and Tourism (MCST), Cross-Ministry Giga KOREA Project (GK17P0200), 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

- 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* 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.
- Seung-Hwan Baek, Diego Gutierrez, and Min H Kim. 2016. Birefractive stereo imaging for single-shot depth acquisition. *ACM Transactions on Graphics* 35, 6 (2016), 194.
- Nicola Brusco, S Capeletto, M Fedel, Anna Paviotti, Luca Poletto, Guido Maria Cortelazzo, and G Tonello. 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. 2011a. A prism-mask system for multispectral video acquisition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 12 (2011), 2423–2435.
- Xun Cao, Xin Tong, Qionghai Dai, and S. Lin. 2011b. High Resolution Multispectral Video Capture with a Hybrid Camera System. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, 297–304.
- Inchang Choi, Daniel S. Jeon, Giljoo Nam, Diego Gutierrez, and Min H. Kim. 2017. High-Quality Hyperspectral Reconstruction Using a Spectral Prior. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2017)* 36, 6 (2017).
- J. M. Bioucas Dias and M. A. T. Figueiredo. 2007. A New TwIST: Two-Step Iterative Shrinkage/Thresholding Algorithms for Image Restoration. *IEEE Trans. Image Processing* 16, 12 (Dec. 2007), 2992–3004.
- Piotr Dollár and C Lawrence Zitnick. 2015. Fast edge detection using structured forests. *IEEE transactions on pattern analysis and machine intelligence* 37, 8 (2015), 1558–1570.
- Liang Gao, Robert T. Kester, Nathan Hagen, and Tomasz S. Tkaczyk. 2010. Snapshot Image Mapping Spectrometer (IMS) with high sampling density for hyperspectral microscopy. *Opt. Express* 18, 14 (Jul 2010), 14330–14344.
- Nahum Gat. 2000. Imaging spectroscopy using tunable filters: a review. In *AeroSense 2000*. International Society for Optics and Photonics, 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 OE* 15, 21 (2007), 14013–27.
- Ralf Habel, Michael Kudnenov, and Michael Wimmer. 2012. Practical spectral photography. In *Computer graphics forum*, Vol. 31. Wiley Online Library, 449–458.
- Kaiming He, Jian Sun, and Xiaowei Tang. 2013. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 6 (2013), 1397–1409.
- Felix Heide, Mushfiqur Rouf, Matthias B Hullin, Björn Labitzke, Wolfgang Heidrich, and Andreas Kolb. 2013. High-quality computational imaging through simple lenses. *ACM Transactions on Graphics* 32, 5 (2013), 149.
- Daniel S Jeon, Inchang Choi, and Min H Kim. 2016. Multisampling Compressive Video Spectroscopy. In *Computer Graphics Forum*, Vol. 35. Wiley Online Library, 467–477.
- Jun Jiang, Dengyu Liu, Jinwei Gu, and Sabine Süstrunk. 2013. What is the space of spectral sensitivity functions for digital color cameras?. In *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*. IEEE, 168–179.
- William R Johnson, Daniel W Wilson, Wolfgang Fink, Mark Humayun, and Greg Bearman. 2007. Snapshot hyperspectral imaging in ophthalmology. *Journal of biomedical optics* 12, 1 (2007), 014036–014036.
- R. Kawakami, Y. Matsushita, J. Wright, M. Ben-Ezra, Y. W. Tai, and K. Ikeuchi. 2011. High-resolution hyperspectral imaging via matrix factorization. In *CVPR 2011*. 2329–2336.
- 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 Transactions on Graphics* 31, 4 (2012), 38.
- 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.
- David Kittle, Kerkil Choi, Ashwin Wagadarikar, and David J Brady. 2010. Multiframe image estimation for coded aperture snapshot spectral imagers. *Applied Optics* 49, 36 (2010), 6824–6833.
- 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 2014) (LNCS)*, Vol. 8509. Springer, Normandy, France, 26–34.
- Chengbo Li. 2011. *Compressive sensing for 3D data processing tasks: applications, models and algorithms*. Ph.D. Dissertation. Rice University.
- Xing Lin, Yebin Liu, Jiamin Wu, and Qionghai Dai. 2014. Spatial-spectral encoded compressive hyperspectral imaging. *ACM Transactions on Graphics* 33, 6 (2014), 233.
- Rang MH Nguyen, Dilip K Prasad, and Michael S Brown. 2014. Training-Based Spectral Reconstruction from a Single RGB Image. In *Computer Vision–ECCV 2014*. Springer, 186–201.
- Takayuki Okamoto, Akinori Takahashi, and Ichiro Yamaguchi. 1993. Simultaneous Acquisition of Spectral and Spatial Intensity Distribution. *Appl. Spectrosc.* 47, 8 (Aug 1993), 1198–1202.
- Patrick Pérez, Michel Gangnet, and Andrew Blake. 2003. Poisson image editing. In *ACM Transactions on Graphics*, Vol. 22. ACM, 313–318.
- Wallace M Porter and Harry T Enmark. 1987. A system overview of the airborne visible/infrared imaging spectrometer (AVIRIS). In *31st Annual Technical Symposium. International Society for Optics and Photonics*, 22–31.
- Konstantinos Rapantzikos and Costas Balas. 2005. Hyperspectral imaging: potential in non-destructive analysis of palimpsests. In *IEEE International Conference on Image Processing 2005*, Vol. 2. IEEE, II–618.
- Brian Smits. 1999. An RGB-to-spectrum conversion for reflectances. *Journal of Graphics Tools* 4, 4 (1999), 11–22.
- Tsuyoshi Takatani, Takahito Aoto, and Yasuhiro Mukaigawa. 2017. One-shot Hyperspectral Imaging using Faced Reflectors. In *Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. TBD.
- Ashwin Wagadarikar, Renu John, Rebecca Willett, and David Brady. 2008. Single disperser design for coded aperture snapshot spectral imaging. *Applied optics* 47, 10 (2008), B44–B51.
- Richard A Waltz, José Luis Morales, Jorge Nocedal, and Dominique Orban. 2006. An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Mathematical programming* 107, 3 (2006), 391–408.
- Fumihito Yasuma, Tomoo Mitsunaga, Daisuke Iso, and Shree K Nayar. 2010. Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum. *IEEE Transactions on Image Processing* 19, 9 (2010), 2241–2253.
- Zhengyou Zhang. 2000. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence* 22, 11 (2000), 1330–1334.

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