Compact Single-Shot Hyperspectral Imaging Using a Prism

(Supplemental Material #1)

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

Fig. 1. Overview of our reconstruction algorithm. We take a dispersed RGB image as an input. We align the dispersed image and extract edge locations so that we can locate dispersion around edges to use it as evident spectral cues (Section 3.1). Gradient values on the edge pixels are then effectively estimated from dispersion (Section 3.2.1). We estimate a hyperspectral image using the gradient information as spectral cues (Section 3.2.2). Note that geometric distortion introduced by refraction of a prism is corrected for the final hyperspectral image.

ACM Reference format:
Seung-Hwan Baek, Incheol Kim, Diego Gutierrez, and Min H. Kim. 2017. Compact Single-Shot Hyperspectral Imaging Using a Prism (Supplemental Material #1) . *ACM Trans. Graph.* 36, 6, Article 217 (November 2017), 4 pages. https://doi.org/10.1145/3130800.3130896

In this supplemental material, we provide the validation of our dispersion model, the details of our calibration method and reconstruction method. For the dispersion model, we provide the experimental results comparing the prediction of our model and the simulation of a professional ray tracing software. For calibration, we describe the experimental details of obtaining the spectrum-to-RGB matrix Ω and the dispersion model Φ. For reconstruction, solution of each subproblem (edge restoration, gradient reconstruction and intensity reconstruction) is described.

1 DISPERSION MODEL VALIDATION

We validate the accuracy of our dispersion model Φ using the professional ray tracing software for optics engineering, Zemax. We simulate our imaging setup in Zemax, where we place a 30-60-90 prism made of N-BK7, in front of a Canon 5D Mark III camera equipped with a 50 mm lens. We first capture a point at a distance of 700 mm using the camera-prism system. We then simulate the dispersion of seven visible wavelengths from 400 nm to 700 nm, using both our dispersion model Φ, and Zemax. Figure 2 shows a strong agreement between both predictions.

2 CALIBRATION

We propose a compact hyperspectral imaging setup consisting of a prism in front of an RGB camera. We describe the calibration method of our system in details including practical issues that we encountered in calibration. For the completeness of the supplemental material, we first rewrite our image formation model as follows:

$$\mathbf{j} = \Omega\Phi\mathbf{i}, \tag{1}$$

where $\mathbf{j}$ is the vectorized RGB image and $\mathbf{i}$ is a vectorized hyperspectral image. Ω is the conversion matrix from spectrum to RGB channels, and Φ describes the dispersion effect represented as a matrix relating corresponding pixels of different spectrums.

Obtaining Ω. We first describe how to obtain Ω, the conversion matrix from spectrum to RGB. This process is called as radiometric calibration. We place a sample of Spectralon illuminated by a Xenon light source (Thorlab HPLS-30-4). We capture the spectral distribution function $R(\lambda)$ of the reflected radiances from the sample with a spectroradiometer (Jeti Specbos 1200), of which has spectral

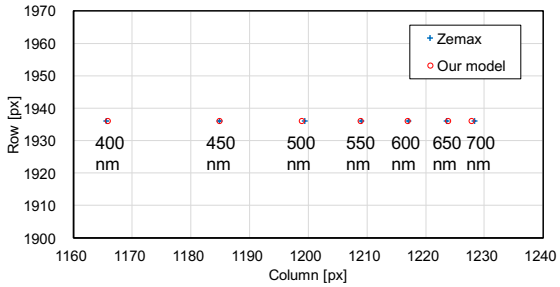

Fig. 2. We compare the predictions of our dispersion model Φ with those of a professional optics simulation software, Zemax. We place a 30-60-90 prism in front of a 50 mm lens with a Canon 5D Mark III camera, and capture a point at a distance of 700 mm. Dispersion is accurately predicted by our method (seven wavelengths from 400 nm to 700 nm are shown), with a strong agreement with the professional software.

© 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.

resolution as 10 nm bandwidth from 400 nm to 700 nm. Then we capture linear hyperspectral images of the same sample with an RGB camera (Canon EOS 5D Mark III) equipped with an liquid crystal tunable filter (Varispec VIS LCTF) in front of the camera. Note that we capture the images with three different exposures to obtain valid signals without under and over saturations for the visible spectrum range. Captured image $M(c,\lambda)$ is the linear hyperspectral image of wavelength λ for the channel $c \in \{red, green, blue\}$. We also measured the transmission function of the liquid crystal tunable filter $T_\lambda(\lambda')$, where it describes the transmission of wavelength λ' when we set the LCTF to capture wavelength λ. From the measurements, we can compute the spectral response for each channel, $\Omega(c,\lambda)$ as follows:

$$\Omega(c,\lambda) = \frac{M(c,\lambda)}{\sum_{\lambda'} R(\lambda')\, T_\lambda(\lambda')}. \tag{2}$$

By representing the spectral response in a matrix form, we obtain $\mathbf{\Omega} \in \mathbb{R}^{23\times 3}$ for 23 wavelengths from 430 nm to 650 nm in 10 nm intervals.

Obtaining $\mathbf{\Phi}$. Our current prototype has two practical issues for performing the dispersion calibration. First, to obtain the dispersion model $\mathbf{\Phi}$, we captured images of different spectral channels by employing bandpass filters in front of our system. However, since the bandpass filters only cover a part of the captured image due to the small size of the bandpass filter, we captured multiple images of a particular channel by translating the bandpass filters which is mounted on a holder. Second, the prism changes view direction of the camera by refracting light rays making the corner points of a same checkerboard cannot be captured with and without the prism. We therefore selected some corner points in the checkerboard to be used for camera calibration, and other corner points in the checkerboard will be used for prism calibration.

3 RECONSTRUCTION

Our reconstruction method consists of three stages: edge restoration, gradient reconstruction, and hyperspectral-image reconstruction. The edge-restoration stage (Section 3.1) spatially aligns an input dispersed RGB image to obtain edge inforamtion without dispersion. In the gradient-domain reconstruction stage (Section 3.2.1), we effectively extract spectral cues at edges in the gradient domain. We then combine this gradient information as a spectral cue to recover accurate spectral intensities in the hyperspectral reconstruction stage (Section 3.2.2). The overall pipeline is described in Algorithm 1. In this supplemental material, we describe additional details of each stage.

ALGORITHM 1: Hyperspectral reconstruction

1: $\mathbf{i}_{\text{aligned}} = \arg\min_{\mathbf{i}} \|\mathbf{\Omega\Phi i} - \mathbf{j}\|_2^2 + \alpha_1 \left\|\nabla_{xy}\mathbf{i}\right\|_1 + \beta_1 \left\|\nabla_\lambda \nabla_{xy}\mathbf{i}\right\|_1$

2: $\hat{\mathbf{g}}_{xy} = \arg\min_{\mathbf{g}_{xy}} \left\|\mathbf{\Omega\Phi g}_{xy} - \nabla_{xy}\mathbf{j}\right\|_2^2 + \alpha_2 \left\|\nabla_\lambda \mathbf{g}_{xy}\right\|_1 + \beta_2 \left\|\nabla_{xy}\mathbf{g}_{xy}\right\|_2^2$

3: $\mathbf{i}_{\text{opt}} = \arg\min_{\mathbf{i}} \|\mathbf{\Omega\Phi i} - \mathbf{j}\|_2^2 + \alpha_3 \left\|\mathbf{W}_{xy} \odot \left(\nabla_{xy}\mathbf{i} - \hat{\mathbf{g}}_{xy}\right)\right\|_2^2 + \beta_3 \left\|\Delta_\lambda \mathbf{i}\right\|_2^2$

4: $\mathbf{i}_{\text{fin}} \leftarrow \text{REFINE}(\mathbf{i}_{\text{opt}}, \mathbf{\Omega i}_{\text{aligned}})$

3.1 Restoring Edges from Dispersion

The edge restoration stage estimates the latent edge location for a given dispersed RGB image by estimating a spatially aligned hyperspectral image. We solve the equation as below:

$$\mathbf{i}_{\text{aligned}} = \arg\min_{\mathbf{i}} \underbrace{\|\mathbf{\Omega\Phi i} - \mathbf{j}\|_2^2}_{\text{data term}} + \underbrace{\alpha_1 \left\|\nabla_{xy}\mathbf{i}\right\|_1 + \beta_1 \left\|\nabla_\lambda \nabla_{xy}\mathbf{i}\right\|_1}_{\text{prior terms}}. \tag{3}$$

The explanation of each term is described in Section 5.1 of the main paper.

Optimization. We employ the alternating direction method of multipliers (ADMM) to split Equation (3) into l_2 and l_1 terms. We introduce slack variables $\mathbf{z}_1$ and $\mathbf{z}_2$ to split our problem into three subproblems:

$$f(\mathbf{i}) = \|\mathbf{\Omega\Phi i} - \mathbf{j}\|_2^2,\ g(\mathbf{z}_1) = \alpha_1\|\mathbf{z}_1\|_1,\ h(\mathbf{z}_2) = \beta_1\|\mathbf{z}_2\|_1,$$

so that our optimization problem described as Equation (3) is reformulated as:

$$\begin{aligned} &\min_{\mathbf{i},\mathbf{z}_1,\mathbf{z}_2} f(\mathbf{i}) + g(\mathbf{z}_1) + h(\mathbf{z}_2) \\ &\text{subject to } \nabla_{xy}\mathbf{i} - \mathbf{z}_1 = 0,\ \nabla_\lambda\nabla_{xy}\mathbf{i} - \mathbf{z}_2 = 0. \end{aligned} \tag{4}$$

Refer to [Boyd et al. 2011] for more detailed description of the ADMM framework. The ADMM solves this problem by optimizing Equation (4) with respect to each variable $\mathbf{i}, \mathbf{z}_1$, and $\mathbf{z}_2$ iteratively. The scaled form of ADMM subproblems are:

$$\begin{aligned} \mathbf{i}^{(k+1)} &= \arg\min_{\mathbf{i}} \begin{array}{l} f(\mathbf{i}) + \frac{\rho_1}{2}\left\|\nabla_{xy}\mathbf{i} - \mathbf{z}_1^{(k)} + \mathbf{u}_1^{(k)}\right\|_2^2 + \\ \frac{\rho_2}{2}\left\|\nabla_\lambda\nabla_{xy}\mathbf{i} - \mathbf{z}_2^{(k)} + \mathbf{u}_2^{(k)}\right\|_2^2 \end{array} \\ \mathbf{z}_1^{(k+1)} &= \arg\min_{\mathbf{z}_1} g(\mathbf{z}_1) + \frac{\rho_1}{2}\left\|\nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_1 + \mathbf{u}_1^{(k)}\right\|_2^2 \\ \mathbf{z}_2^{(k+1)} &= \arg\min_{\mathbf{z}2} h(\mathbf{z}_2) + \frac{\rho_2}{2}\left\|\nabla_\lambda\nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_2 + \mathbf{u}_2^{(k)}\right\|_2^2 \\ \mathbf{u}_1^{(k+1)} &= \mathbf{u}_1^{(k)} + \nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_1^{(k+1)} \\ \mathbf{u}_2^{(k+1)} &= \mathbf{u}_2^{(k)} + \nabla_\lambda\nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_2^{(k+1)}, \end{aligned} \tag{5}$$

where $\mathbf{u}_1$ and $\mathbf{u}_2$ are Lagrange multipliers. The variable $\mathbf{i}^{(k+1)}$ is updated by solving only l_2 terms, which is done by the conjugate gradient method. The slack variables $\mathbf{z}_1$ and $\mathbf{z}_2$ are updated by using a soft-thresholding operator, which is given as:

$$S_\theta(\mathbf{x}) = \begin{cases} \mathbf{x} - \theta & \mathbf{x} > \theta \\ 0 & |\mathbf{x}| \leqslant \theta \\ \mathbf{x} + \theta & \mathbf{x} < -\theta, \end{cases} \tag{6}$$

where θ is a parameter that determines the strength of sparsity of gradients. The Lagrange multipliers $\mathbf{u}_1$ and $\mathbf{u}_2$ are then updated via gradient ascent to relate the objective function with constraints in Equation (4). See Algorithm 2 as an overview.

3.2 Reconstructing Spectral Information

In this stage, we first obtain spectral signature in a gradient domain (Section 3.2.1), and we use this information for spectral image reconstruction (Section 3.2.2). The key insight of this stage is described in Section 5.2 in the main paper.

ALGORITHM 2: ADMM solution of Equation (5)

repeat
$\mathbf{i}^{(k+1)} = \arg\min_{\mathbf{i}} f(\mathbf{i}) + \frac{\rho_1}{2}\left\|\nabla_{xy}\mathbf{i} - \mathbf{z}_1^{(k)} + \mathbf{u}_1^{(k)}\right\|_2^2 + \frac{\rho_2}{2}\left\|\nabla_\lambda\nabla_{xy}\mathbf{i} - \mathbf{z}_2^{(k)} + \mathbf{u}_2^{(k)}\right\|_2^2$
$\mathbf{z}_1^{(k+1)} = S_{\frac{\alpha_1}{\rho_1}}\left(\nabla_{xy}\mathbf{i}^{(k+1)} + \mathbf{u}_1^{(k)}\right)$
$\mathbf{z}_2^{(k+1)} = S_{\frac{\beta_1}{\rho_2}}\left(\nabla_\lambda\nabla_{xy}\mathbf{i}^{(k+1)} + \mathbf{u}_2^{(k)}\right)$
$\mathbf{u}_1^{(k+1)} = \mathbf{u}_1^{(k)} + \nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_1^{(k+1)}$
$\mathbf{u}_2^{(k+1)} = \mathbf{u}_2^{(k)} + \nabla_\lambda\nabla_{xy}\mathbf{i}^{(k+1)} - \mathbf{z}_2^{(k+1)}$
until *the stopping criterium is satisfied.*

3.2.1 Gradient-based Reconstruction. We first estimate a stack of spatial gradients $\mathbf{g}_{xy}$ for each wavelength by solving a following problem:

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

See Section 5.2.1 of the main paper for more detailed explanation of the formulation.

Edge-based Objective Reformulation. Prior to solving the objective, we reduce the region of the optimization problem to edge pixels only (Equation (7)). Since the spectral signature is only observed around edges, solving the objective in the entire pixel deteriorates the performance of optimization.

We first describe how to narrow down the region of interests in the gradient domain. Given the edge information obtained from the edge restoration stage (Section 3.1), we define forward and backward edge-pixel extracting matrices in a gradient domain: $\mathbf{M}_f \in \mathbb{R}^{2E\Lambda\times 2XY\Lambda}$ and $\mathbf{M}_b \in \mathbb{R}^{2XY\Lambda\times 2E\Lambda}$, where X and Y are the width and height of an image, and $E \ll XY$ is the number of edge pixels. The forward and backward operations with the matrices are written

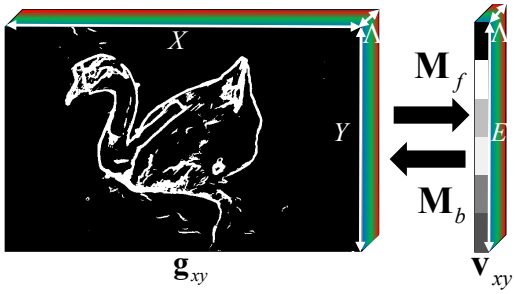

Fig. 3. The forward and backward edge extraction matrices denoted by Equation (8). The left figure shows a stack of gradient images for each spectral channel $\mathbf{g}_{xy}$, and the right figure describes the extracted gradient values on edges denoted by $\mathbf{v}_{xy}$.

as below:

$$\mathbf{v}_{xy} = \mathbf{M}_f\mathbf{g}_{xy},\ \mathbf{g}_{xy} = \mathbf{M}_b\mathbf{v}_{xy}, \tag{8}$$

where $\mathbf{v}_{xy} \in \mathbb{R}^{2E\Lambda}$ denotes gradient values at the edge pixels. The forward matrix $\mathbf{M}_f$ extracts values at the edge pixels of a hyperspectral image in a gradient domain denoted by $\mathbf{g}_{xy}$, while the backward matrix $\mathbf{M}_b$ expands the resolution of the gradient values on edge pixels $\mathbf{v}_{xy}$ to the original resolution by locating the values at the original positions and filling zero in non-edge pixels. See Figure 3 for graphical description of the operators. Based on the notation above, we reformulate the gradient reconstruction problem (Equation (7)) so that we only need to optimize edge pixels:

$$\hat{\mathbf{v}}_{xy} = \arg\min_{\mathbf{v}_{xy}} \underbrace{\left\|\Omega\Phi\mathbf{M}_b\mathbf{v}_{xy} - \nabla_{xy}\mathbf{j}\right\|_2^2}_{\text{data term}} + \underbrace{\alpha_2\left\|\nabla_\lambda\mathbf{v}_{xy}\right\|_1 + \beta_2\left\|\nabla_{xy}\mathbf{M}_b\mathbf{v}_{xy}\right\|_2^2}_{\text{prior terms}}. \tag{9}$$

The first term is a data term that ensures the image formation model (Equation (1)) in gradient domain over edge pixels. Since the term $\Omega\Phi\mathbf{M}_b\mathbf{v}_{xy}$ has zero values in non-edge pixels, we assign zero at non-edge pixels in the observation $\nabla_{xy}\mathbf{j}$ as well. The two other terms are prior terms for gradient values. The first prior is the spectral sparsity of gradient used in the spatial alignment stage (Equation (3)). It enforces sparse change of gradient values along the spectral domain. The second prior is a spatial smoothness for smooth change of gradients in the spatial domain, which reduces the artifacts of gradient values.

Optimization. We solve Equation (9) with ADMM, splitting l_2 and l_1 terms, and employing a slack variable $\mathbf{z}_3$. The problem is then splitted into two subproblems:

$$f\left(\mathbf{v}_{xy}\right) = \left\|\Omega\Phi\mathbf{M}_b\mathbf{v}_{xy} - \nabla_{xy}\mathbf{j}\right\|_2^2 + \beta_2\left\|\nabla_{xy}\mathbf{M}_b\mathbf{v}_{xy}\right\|_2^2,\ g\left(\mathbf{z}_3\right) = \alpha_2\|\mathbf{z}_3\|_1.$$

We then combine constraints induced by employing the slack variable so that we formulate an ADMM objective:

$$\begin{aligned} &\min_{\mathbf{v}_{xy},\mathbf{z}_3} f\left(\mathbf{v}_{xy}\right) + g\left(\mathbf{z}_3\right) \\ &\text{subject to } \nabla_\lambda\mathbf{v}_{xy} - \mathbf{z}_3 = 0. \end{aligned} \tag{10}$$

The scaled-form solution of the ADMM objective is:

$$\begin{aligned} \mathbf{v}_{xy}^{(k+1)} &= \arg\min_{\mathbf{v}_{xy}} f\left(\mathbf{v}_{xy}\right) + \frac{\rho}{2}\left\|\nabla_\lambda\mathbf{v}_{xy} - \mathbf{z}_3^{(k)} + \mathbf{u}_3^{(k)}\right\|_2^2 \\ \mathbf{z}_3^{(k+1)} &= \arg\min_{\mathbf{z}_3} g\left(\mathbf{z}_3\right) + \frac{\rho}{2}\left\|\nabla_\lambda\mathbf{v}_{xy}^{(k+1)} - \mathbf{z}_3 + \mathbf{u}_3^{(k)}\right\|_2^2 \\ \mathbf{u}_3^{(k+1)} &= \mathbf{u}_3^{(k)} + \nabla_\lambda\mathbf{v}_{xy}^{(k+1)} - \mathbf{z}_3^{(k+1)}, \end{aligned} \tag{11}$$

where $\mathbf{u}_3$ is a Lagrange multiplier. The $\mathbf{v}_{xy}$-subproblem is solved by the conjugate gradient method, and the solution of the $\mathbf{z}_3$-subproblem is obtained by soft-thresholding (Equation (6)). The optimization process of this stage is described in Algorithm 3. After obtaining the gradient values at edge pixels as $\hat{\mathbf{v}}_{xy}$, we compute $\hat{\mathbf{g}}_{xy}$, a stack of gradient images for each spectral channel: $\hat{\mathbf{g}}_{xy} = \mathbf{M}_b\hat{\mathbf{v}}_{xy}$.

ALGORITHM 3: ADMM solution of Equation (11)

repeat
$\mathbf{v}_{xy}^{(k+1)} = \arg\min_{\mathbf{v}_{xy}} f\left(\mathbf{v}_{xy}\right) + \frac{\rho}{2}\left\|\nabla_\lambda\mathbf{v}_{xy} - \mathbf{z}_3^{(k)} + \mathbf{u}_3^{(k)}\right\|_2^2$
$\mathbf{z}_3^{(k+1)} = S_{\frac{\alpha_2}{\rho}}\left(\nabla_\lambda\mathbf{v}_{xy}^{(k+1)} + \mathbf{u}_3^{(k)}\right)$
$\mathbf{u}_3^{(k+1)} = \mathbf{u}_3^{(k)} + \nabla_\lambda\mathbf{v}_{xy}^{(k+1)} - \mathbf{z}_3^{(k+1)}$
until *the stopping criterium is satisfied.*

3.2.2 Reconstructing Spectral Images. The spectral cue $\hat{\mathbf{g}}_{xy}$ is obtained in the previous stage (Section 3.2). We use this information to reconstruct a hyperspectral image so that we can properly rely on spectral signature. The reconstruction is performed by solving the equation below:

$$\mathbf{i}_{\text{opt}} = \arg\min_{\mathbf{i}} \underbrace{\|\Omega\Phi\mathbf{i} - \mathbf{j}\|_2^2 + \alpha_3 \left\|\mathbf{W}_{xy} \odot \left(\nabla_{xy}\mathbf{i} - \hat{\mathbf{g}}_{xy}\right)\right\|_2^2}_{\text{data terms}} + \underbrace{\beta_3 \|\Delta_\lambda \mathbf{i}\|_2^2}_{\text{prior term}}. \tag{12}$$

See Section 5.2.2 in the main paper for more detailed explanation of the formulation. Since Equation (12) consists only of l_2-norm terms, we solve it using a conjugate gradient method iteratively by clamping intensity values from zero to one.

Detail-Guided Filtering. We enhance the spatial details of the estimated hyperspectral image $\mathbf{i}_{\text{opt}}$ by applying an edge-aware filtering [He et al. 2013] to each reconstructed hyperspectral channel, using the aligned image $\Omega\mathbf{i}_{\text{aligned}}$ as guide. This leads to our final reconstructed image $\mathbf{i}_{\text{fin}}$, with high spectral and spatial accuracy.

REFERENCES

Stephen P. Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. 2011. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine Learning* 3, 1 (2011), 1–122.

Kaiming He, Jian Sun, and Xiaoou Tang. 2013. Guided image filtering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35, 6 (2013), 1397–1409.

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